A Simple FPTAS for Counting Edge Covers

Chengyu Lin[1] Jingcheng Liu[1] Pinyan Lu[2]

[1]Shanghai Jiao Tong University

[2]Microsoft Research Asia

ACM-SIAM Symposium on Discrete Algorithms, 2014

Overview

1. Introduction
   - Definition
   - Counting Problems
   - Edge Covers and Matching
   - Edge Covers and Rtw-Mon-CNF Formulae
2. Main Result
   - Previous Results and Our Result
   - Counting via Marginal Probability
   - Computation Tree Recursion
   - Correlation Decay

Overview

1. Introduction
   - Definition
   - Counting Problems
   - Edge Covers and Matching
   - Edge Covers and Rtw-Mon-CNF Formulae
2. Main Result
   - Previous Results and Our Result
   - Counting via Marginal Probability
   - Computation Tree Recursion
   - Correlation Decay

Edge cover

Definition

For an undirected input graph $G = (V, E)$, an **edge cover** of G is a set of edges C covering all vertices.

Example

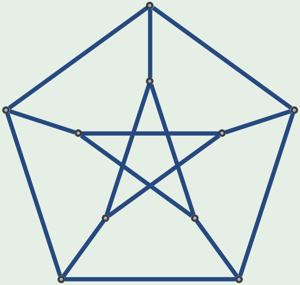


Figure : An edge cover for Petersen graph

Edge cover

Definition

For an undirected input graph $G = (V, E)$, an **edge cover** of G is a set of edges C covering all vertices.

Example

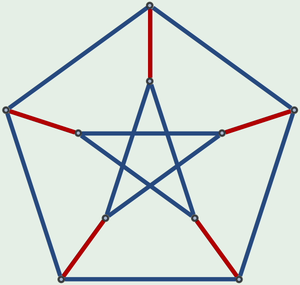


Figure : An edge cover for Petersen graph, with edges chosen being highlighted in red. Note that this is also a perfect matching.

Overview

1. Introduction
   - Definition
   - Counting Problems
   - Edge Covers and Matching
   - Edge Covers and Rtw-Mon-CNF Formulae
2. Main Result
   - Previous Results and Our Result
   - Counting via Marginal Probability
   - Computation Tree Recursion
   - Correlation Decay

Why counting?

Remark (Counting edge covers)

- *As the name suggests, the problem of counting edge covers simply asks for the number of edge covers in a graph.*
- *While both decision and optimization (minimum edge covers) version of edge cover is easy, the counting version is more challenging and interesting.*

Besides theoretical computer science, counting problems could also be related to many problems from other discipline such as:

- Partition function in statistical physics.
- Graph polynomials.
- Sampling, learning and inference in probabilistic graphical models.
- Pricing in combinatorial prediction markets.
- Query evaluations of probabilistic database.
- ....

Why counting?

Remark (Counting edge covers)

- *As the name suggests, the problem of counting edge covers simply asks for the number of edge covers in a graph.*
- *While both decision and optimization (minimum edge covers) version of edge cover is easy, the counting version is more challenging and interesting.*

Besides theoretical computer science, counting problems could also be related to many problems from other discipline such as:

- Partition function in statistical physics.
- Graph polynomials.
- Sampling, learning and inference in probabilistic graphical models.
- Pricing in combinatorial prediction markets.
- Query evaluations of probabilistic database.
- ....

Why counting?

Remark (Counting edge covers)

- *As the name suggests, the problem of counting edge covers simply asks for the number of edge covers in a graph.*
- *While both decision and optimization (minimum edge covers) version of edge cover is easy, the counting version is more challenging and interesting.*

Besides theoretical computer science, counting problems could also be related to many problems from other discipline such as:

- Partition function in statistical physics.
- Graph polynomials.
- Sampling, learning and inference in probabilistic graphical models.
- Pricing in combinatorial prediction markets.
- Query evaluations of probabilistic database.
- ....

Why counting?

Remark (Counting edge covers)

- *As the name suggests, the problem of counting edge covers simply asks for the number of edge covers in a graph.*
- *While both decision and optimization (minimum edge covers) version of edge cover is easy, the counting version is more challenging and interesting.*

Besides theoretical computer science, counting problems could also be related to many problems from other discipline such as:

- Partition function in statistical physics.
- Graph polynomials.
- Sampling, learning and inference in probabilistic graphical models.
- Pricing in combinatorial prediction markets.
- Query evaluations of probabilistic database.
- ....

Why counting?

Remark (Counting edge covers)

- *As the name suggests, the problem of counting edge covers simply asks for the number of edge covers in a graph.*
- *While both decision and optimization (minimum edge covers) version of edge cover is easy, the counting version is more challenging and interesting.*

Besides theoretical computer science, counting problems could also be related to many problems from other discipline such as:

- Partition function in statistical physics.
- Graph polynomials.
- Sampling, learning and inference in probabilistic graphical models.
- Pricing in combinatorial prediction markets.
- Query evaluations of probabilistic database.
- ....

Approximate Counting

Many interesting problems in the exact counting regimes, including counting edge cover, is hard (#P-complete). We look for their approximate version.

Definition (FPTAS)

For given parameter $\varepsilon > 0$ and an instance of a particular problem class, if the algorithm outputs a number $\hat{N}$ such that

$$(1-\varepsilon)N \leq \hat{N} \leq (1+\varepsilon)N,$$

where N is the accurate answer of the problem instance, and the running time is bounded by $poly(n, 1/\varepsilon)$ with n being the size of instance, this is called the **FPTAS (fully polynomial time approximation scheme)**.

A randomized relaxation of FPTAS is known as **FPRAS (fully polynomial time randomized approximation scheme)**, which uses random bits and only outputs $\hat{N}$ to the desired precision with high probability.

Approximate Counting

Many interesting problems in the exact counting regimes, including counting edge cover, is hard (#P-complete). We look for their approximate version.

Definition (FPTAS)

For given parameter $\varepsilon > 0$ and an instance of a particular problem class, if the algorithm outputs a number $\hat{N}$ such that

$$(1-\varepsilon)N \leq \hat{N} \leq (1+\varepsilon)N,$$

where N is the accurate answer of the problem instance, and the running time is bounded by $poly(n, 1/\varepsilon)$ with n being the size of instance, this is called the **FPTAS (fully polynomial time approximation scheme)**.

A randomized relaxation of FPTAS is known as **FPRAS (fully polynomial time randomized approximation scheme)**, which uses random bits and only outputs $\hat{N}$ to the desired precision with high probability.

Edge cover

Edge cover is related to many other problems such as:

- Matching problem.
- Rtw-Mon-CNF. (read twice monotone CNF)
- Holant problem.
- . . . .

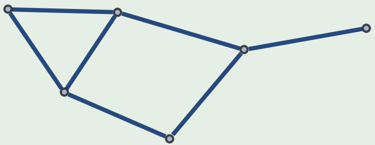
Overview

Relation to Matching

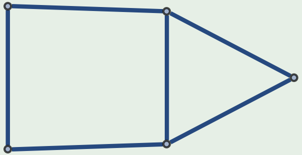
Counting perfect matchings could be solved with an oracle that counts the minimum edge covers.

Example (Minimum edge covers)

Find a minimum edge cover by maximal matching?



(a) G has a perfect matching.

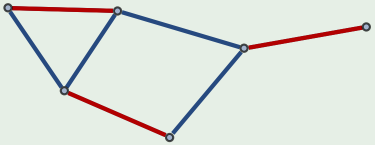


(b) G doesn't have a perfect matching.

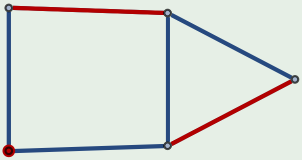
Counting perfect matchings could be solved with an oracle that counts the minimum edge covers.

Example (Minimum edge covers)

Find a minimum edge cover by maximal matching?



(a) G has a perfect matching.



(b) G doesn't have a perfect matching.

Overview

Relation to Rtw-Mon-CNF

Definition

A formula is **read twice** if every variables appears at most twice.
A formula is **monotone** if every variables appears positively.

By treating edges as variables, and vertices as clauses, every graph G has a $\phi(G)$ in Rtw-Mon-CNF, and every Rtw-Mon-CNF ϕ has a $G(\phi)$, in such a way that $SAT(\phi(G)) \equiv EC(G)$ and $SAT(\phi) \equiv EC(G(\phi))$.

Figure : Graph representation for $\phi = (e_1 \vee e_2 \vee e_3) \wedge (e_1 \vee e_4) \wedge (e_4 \vee e_5 \vee e_2) \wedge (e_3 \vee e_5)$.

Relation to Rtw-Mon-CNF

Definition

A formula is **read twice** if every variables appears at most twice.
A formula is **monotone** if every variables appears positively.

By treating edges as variables, and vertices as clauses, every graph G has a $\phi(G)$ in Rtw-Mon-CNF, and every Rtw-Mon-CNF ϕ has a $G(\phi)$, in such a way that $SAT(\phi(G)) \equiv EC(G)$ and $SAT(\phi) \equiv EC(G(\phi))$.

Figure : Graph representation for $\phi = (e_1 \vee e_2 \vee e_3) \wedge (e_1 \vee e_4) \wedge (e_4 \vee e_5 \vee e_2) \wedge (e_3 \vee e_5)$.

Relation to Rtw-Mon-CNF

Definition

A formula is **read twice** if every variables appears at most twice.
A formula is **monotone** if every variables appears positively.

By treating edges as variables, and vertices as clauses, every graph G has a $\phi(G)$ in Rtw-Mon-CNF, and every Rtw-Mon-CNF ϕ has a $G(\phi)$, in such a way that $SAT(\phi(G)) \equiv EC(G)$ and $SAT(\phi) \equiv EC(G(\phi))$.

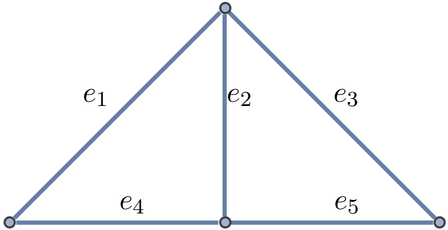


Figure : Graph representation for
$\phi = (e_1 \vee e_2 \vee e_3) \wedge (e_1 \vee e_4) \wedge (e_4 \vee e_5 \vee e_2) \wedge (e_3 \vee e_5)$.

Overview

Previous Results

- For counting edge covers, only an Markov Chain Monte Carlo (MCMC) based FPRAS is known for graphs with maximum degree 3.
- For #DNF and counting matchings, also only FPRAS is known.
- For counting perfect matchings, it's still open whether or not it admits FPRAS (or FPTAS).
- For anti-ferromagnetic 2-spins systems (e.g. counting independent sets), an FPTAS is known, and it's correlation decay based, and goes beyond the best known MCMC based FPRAS and achieves the boundary of approximability (in terms of maximum degree).

Previous Results

- For counting edge covers, only an Markov Chain Monte Carlo (MCMC) based FPRAS is known for graphs with maximum degree 3.
- For #DNF and counting matchings, also only FPRAS is known.
- For counting perfect matchings, it's still open whether or not it admits FPRAS (or FPTAS).
- For anti-ferromagnetic 2-spins systems (e.g. counting independent sets), an FPTAS is known, and it's correlation decay based, and goes beyond the best known MCMC based FPRAS and achieves the boundary of approximability (in terms of maximum degree).

Previous Results

- For counting edge covers, only an Markov Chain Monte Carlo (MCMC) based FPRAS is known for graphs with maximum degree 3.
- For #DNF and counting matchings, also only FPRAS is known.
- For counting perfect matchings, it's still open whether or not it admits FPRAS (or FPTAS).
- For anti-ferromagnetic 2-spins systems (e.g. counting independent sets), an FPTAS is known, and it's correlation decay based, and goes beyond the best known MCMC based FPRAS and achieves the boundary of approximability (in terms of maximum degree).

Previous Results

- For counting edge covers, only an Markov Chain Monte Carlo (MCMC) based FPRAS is known for graphs with maximum degree 3.
- For #DNF and counting matchings, also only FPRAS is known.
- For counting perfect matchings, it's still open whether or not it admits FPRAS (or FPTAS).
- For anti-ferromagnetic 2-spins systems (e.g. counting independent sets), an FPTAS is known, and it's correlation decay based, and goes beyond the best known MCMC based FPRAS and achieves the boundary of approximability (in terms of maximum degree).

Our Result

- An FPTAS for counting edge covers in *general* graphs.
- Our algorithm is correlation decay based.
- This provides another example where the tractable range of correlation decay based FPTAS exceeds the sampling based FPRAS.

Proof Sketch

Let $EC(G)$ be the set of edge covers. Here is an overall work-flow:

- Relate $|EC(G)|$ with a marginal probability $P(G, e)$.
- Derive a computation tree recursion for $P(G, e)$.
- $P(G, e, L)$: Truncate the tree at depth L for some notion of depth.
- Show exponential correlation decay with respect to that tree depth:

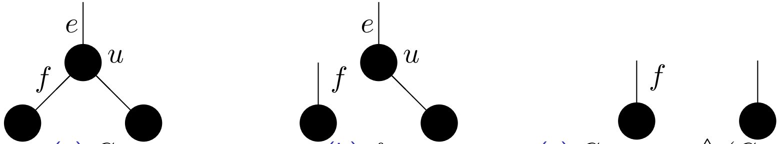
$$|P(G, e, L) - P(G, e)| \leq exp(-\Omega(L))$$

Definition (Dangling edge)

A **dangling edge** $e = (u, \_)$ of a graph is such a singleton edge with exactly one end-point vertex u, as shown in the Figure 4a.

$$G - e \triangleq (V, E \setminus e)$$

$$G - u \triangleq (V \setminus u, E - u)$$



(a) G (b) $f - u$ (c) $G - e - u \triangleq (G - e) - u$

Figure : Dangling edges examples.

Proof Sketch

Let $EC(G)$ be the set of edge covers. Here is an overall work-flow:

- Relate $|EC(G)|$ with a marginal probability $P(G, e)$.
- Derive a computation tree recursion for $P(G, e)$.
- $P(G, e, L)$: Truncate the tree at depth L for some notion of depth.
- Show exponential correlation decay with respect to that tree depth:

$$|P(G, e, L) - P(G, e)| \leq exp(-\Omega(L))$$

Counting v.s. Marginal Probability

Problem

Goal: estimate $|EC(G)|$.

Let X be an edge cover sampled uniformly from $EC(G)$, consider the following marginal probability:
for an edge e, we write $P(G, e) \triangleq \Pr(e \notin X)$.

Solution: estimate $P(G, e)$.

Why $P(G, e)$?

Recall that the set of all edges E is an edge cover. For a randomly sampled edge cover X, what is $\Pr(X = E)$?

Let $E = \{e_i\}$, and $e_i = (u_i, v_i)$.

$$\Pr(X = E) = \frac{1}{|EC(G)|}$$

Why $P(G, e)$?

Recall that the set of all edges E is an edge cover. For a randomly sampled edge cover X, what is $\Pr(X = E)$?
Let $E = \{e_i\}$, and $e_i = (u_i, v_i)$.

$$\Pr(X = E) = \frac{1}{|EC(G)|}$$

Why $P(G, e)$?

Recall that the set of all edges E is an edge cover. For a randomly sampled edge cover X, what is $\Pr(X = E)$?
Let $E = \{e_i\}$, and $e_i = (u_i, v_i)$.

$$\Pr(X = E) = \frac{1}{|EC(G)|}$$

$$\Pr(X = E) = \Pr(\forall i, e_i \in X)$$

Why $P(G, e)$?

Recall that the set of all edges E is an edge cover. For a randomly sampled edge cover X, what is $\Pr(X = E)$?
Let $E = \{e_i\}$, and $e_i = (u_i, v_i)$.

$$\Pr(X = E) = \frac{1}{|EC(G)|}$$

$$\begin{aligned} \Pr(X = E) &= \Pr(\forall i, e_i \in X) \\ &= \Pr(e_1 \in X) \Pr(e_2 \in X \mid e_1 \in X) \cdots \end{aligned}$$

Why $P(G, e)$?

Recall that the set of all edges E is an edge cover. For a randomly sampled edge cover X, what is $\Pr(X = E)$?
Let $E = \{e_i\}$, and $e_i = (u_i, v_i)$.

$$\Pr(X = E) = \frac{1}{|EC(G)|}$$

$$\begin{aligned}\Pr(X = E) &= \Pr(\forall i, e_i \in X) \\ &= \Pr(e_1 \in X)\Pr(e_2 \in X \mid e_1 \in X) \cdots \\ &= \prod_i \Pr(e_i \in X \mid \{e_j\}_{j=1}^{i-1} \subseteq X)\end{aligned}$$

Why $P(G, e)$?

Recall that the set of all edges E is an edge cover. For a randomly sampled edge cover X, what is $\Pr(X = E)$?
Let $E = \{e_i\}$, and $e_i = (u_i, v_i)$.

$$\Pr(X = E) = \frac{1}{|EC(G)|}$$

$$\begin{aligned}
\Pr(X = E) &= \Pr(\forall i, e_i \in X) \\
&= \Pr(e_1 \in X) \Pr(e_2 \in X \mid e_1 \in X) \cdots \\
&= \prod_i \Pr(e_i \in X \mid \{e_j\}_{j=1}^{i-1} \subseteq X) \\
&= \prod_i \left(1 - P(G_i, e_i)\right),
\end{aligned}$$

where $G_1 = G, G_{i+1} = G_i - e_i - u_i - v_i$.

Why $P(G, e)$?

Recall that the set of all edges E is an edge cover. For a randomly sampled edge cover X, what is $\Pr(X = E)$?
Let $E = \{e_i\}$, and $e_i = (u_i, v_i)$.

$$\Pr(X = E) = \frac{1}{|EC(G)|}$$

$$\begin{aligned}\Pr(X = E) &= \Pr(\forall i, e_i \in X) \\ &= \Pr(e_1 \in X) \Pr(e_2 \in X \mid e_1 \in X) \cdots \\ &= \prod_i \Pr(e_i \in X \mid \{e_j\}_{j=1}^{i-1} \subseteq X) \\ &= \prod_i (1 - P(G_i, e_i)),\end{aligned}$$

where $G_1 = G, G_{i+1} = G_i - e_i - u_i - v_i$.
Therefore,

$$\frac{1}{|EC(G)|} = \prod_i (1 - P(G_i, e_i)).$$

Proof Sketch

Let $EC(G)$ be the set of edge covers. Here is an overall work-flow:

- Relate $|EC(G)|$ with a marginal probability $P(G, e)$.
- Derive a computation tree recursion for $P(G, e)$.
- $P(G, e, L)$: Truncate the tree at depth L for some notion of depth.
- Show exponential correlation decay with respect to that tree depth:

$$|P(G, e, L) - P(G, e)| \leq exp(-\Omega(L))$$

Computation Tree Recursion

We focus on e is dangling.

$$P(G, e) = \frac{1 - \prod_{i=1}^{d} P(G_i, e_i)}{2 - \prod_{i=1}^{d} P(G_i, e_i)},$$

where $G_1 \triangleq G - e - u$, and $G_{i+1} \triangleq G_i - e_i$.

Truncate with a modified recursion depth:

$$P(G, e, L) = \begin{cases} \frac{1}{2}, & \text{if } L \le 0; \\ \frac{1 - \prod_{i=1}^{d} P(G_i, e_i, L - \lceil \log_6 (d+1) \rceil)}{2 - \prod_{i=1}^{d} P(G_i, e_i, L - \lceil \log_6 (d+1) \rceil)}, & \text{otherwise.} \end{cases}$$

Computation Tree Recursion

We focus on e is dangling.

$$P(G, e) = \frac{1 - \prod_{i=1}^{d} P(G_i, e_i)}{2 - \prod_{i=1}^{d} P(G_i, e_i)},$$

where $G_1 \triangleq G - e - u$, and $G_{i+1} \triangleq G_i - e_i$.
Truncate with a modified recursion depth:

$$P(G, e, L) = \begin{cases} \frac{1}{2}, & \text{if } L \leq 0; \\ \frac{1 - \prod_{i=1}^{d} P(G_i, e_i, L - \lceil \log_6 (d+1) \rceil)}{2 - \prod_{i=1}^{d} P(G_i, e_i, L - \lceil \log_6 (d+1) \rceil)}, & \text{otherwise.} \end{cases}$$

Proof Sketch

Let $EC(G)$ be the set of edge covers. Here is an overall work-flow:

- Relate $|EC(G)|$ with a marginal probability $P(G, e)$.
- Derive a computation tree recursion for $P(G, e)$.
- $P(G, e, L)$: Truncate the tree at depth L for some notion of depth.
- Show exponential correlation decay with respect to that tree depth:

$$|P(G, e, L) - P(G, e)| \leq exp(-\Omega(L))$$

Correlation Decay

Proposition

Given graph G, edge e and depth L,

$$|P(G,e,L) - P(G,e)| \leq 3 \cdot (\frac{1}{2})^{L+1}$$

Now we just take $L = \log_2\left(\frac{6m}{\varepsilon}\right)$, this gives the desired FPTAS.

Conclusions and Upcoming Results

As a conclusion, we have shown an FPTAS for counting edge covers in general graphs.

We later generalized our techniques for establishing correlation decay based FPTAS and obtained the following:

- FPTAS for counting weighted edge covers. (Arbitrary edge weights).
- FPTAS for counting Read-5-Mon-CNF, while Read-6-Mon-CNF does not admit FPTAS unless $P = NP$.
- FPTAS for counting hypergraph matching in certain range (e.g. 3D-Matching with maximum degree 4).
- ...

Conclusions and Upcoming Results

As a conclusion, we have shown an FPTAS for counting edge covers in general graphs.

We later generalized our techniques for establishing correlation decay based FPTAS and obtained the following:

- FPTAS for counting weighted edge covers. (Arbitrary edge weights).
- FPTAS for counting Read-5-Mon-CNF, while Read-6-Mon-CNF does not admit FPTAS unless $P = NP$.
- FPTAS for counting hypergraph matching in certain range (e.g. 3D-Matching with maximum degree 4).
- ...

THANK YOU!

Q & A.