

# A SIMPLE FPTAS FOR COUNTING EDGE COVERS

Chengyu Lin<sup>1</sup>   Jingcheng Liu<sup>1</sup>   Pinyan Lu<sup>2</sup>

<sup>1</sup>SHANGHAI JIAO TONG UNIVERSITY

<sup>2</sup>MICROSOFT RESEARCH ASIA

ACM-SIAM SYMPOSIUM ON DISCRETE ALGORITHMS, 2014

## 1 INTRODUCTION

- Definition
- Edge Covers and Matching
- Edge Covers and Rtw-Mon-CNF Formulae
- Counting Problems

## 2 MAIN RESULT

- Previous Results and Our Result
- Counting via Marginal Probability
- Computation Tree Recursion
- Correlation Decay

## 1 INTRODUCTION

- Definition
- Edge Covers and Matching
- Edge Covers and Rtw-Mon-CNF Formulae
- Counting Problems

## 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

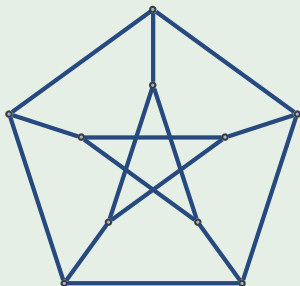


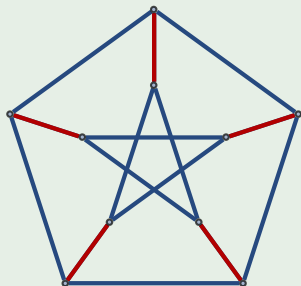
Figure : An edge cover for Petersen graph

# 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.

Edge cover is related to many other problems such as:

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

## 1 INTRODUCTION

- Definition
- **Edge Covers and Matching**
- Edge Covers and Rtw-Mon-CNF Formulae
- Counting Problems

## 2 MAIN RESULT

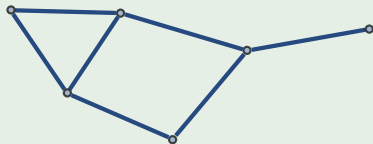
- Previous Results and Our Result
- Counting via Marginal Probability
- Computation Tree Recursion
- Correlation Decay

# Relation to Matching

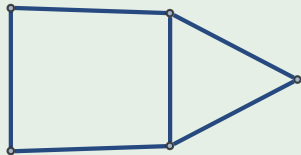
For a graph with a perfect matching, enumerating (sampling) perfect matchings is equivalent to enumerating (sampling) minimum edge covers.

## 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.

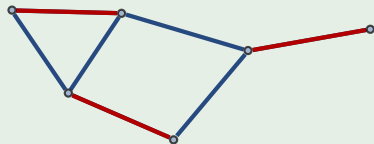


# Relation to Matching

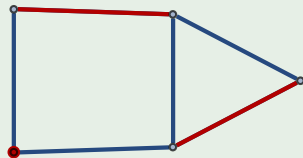
For a graph with a perfect matching, enumerating (sampling) perfect matchings is equivalent to enumerating (sampling) minimum edge covers.

## 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.

## 1 INTRODUCTION

- Definition
- Edge Covers and Matching
- **Edge Covers and Rtw-Mon-CNF Formulae**
- Counting Problems

## 2 MAIN RESULT

- Previous Results and Our Result
- Counting via Marginal Probability
- Computation Tree Recursion
- Correlation Decay

# 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.

Consider the following Rtw-Mon-CNF formula, its satisfying assignments are exactly edge covers of its graph representation, where we write edges as variables, and vertices as clauses.

$$\phi = (e_1 \vee e_2 \vee e_3) \wedge (e_1 \vee e_4) \wedge (e_4 \vee e_5 \vee e_2) \wedge (e_3 \vee e_5).$$

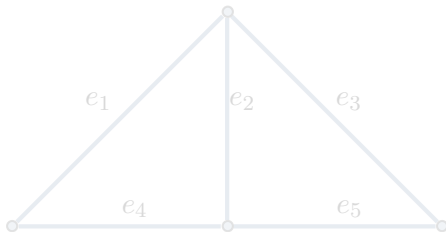


Figure : Graph representation for  $\phi$ .

# 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.

Consider the following Rtw-Mon-CNF formula, its satisfying assignments are exactly edge covers of its graph representation, where we write edges as variables, and vertices as clauses.

$$\phi = (e_1 \vee e_2 \vee e_3) \wedge (e_1 \vee e_4) \wedge (e_4 \vee e_5 \vee e_2) \wedge (e_3 \vee e_5).$$

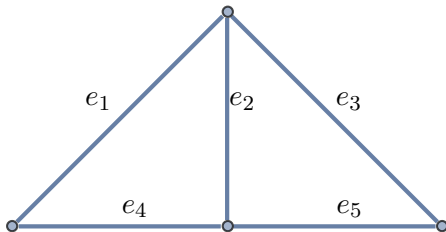


Figure : Graph representation for  $\phi$ .

## 1 INTRODUCTION

- Definition
- Edge Covers and Matching
- Edge Covers and Rtw-Mon-CNF Formulae
- Counting Problems

## 2 MAIN RESULT

- Previous Results and Our Result
- Counting via Marginal Probability
- Computation Tree Recursion
- Correlation Decay

# Counting Problems

A list of problems in their search, optimization, and counting versions.

## Search problems:

- SAT.
- Find a (perfect) matching.
- Find an edge cover.
- ....

## Optimizations:

- MAX-SAT.
- Find a maximum matching.
- Find a minimum edge cover.
- ....

## Counting problems:

- #SAT.
- Counting matchings.
- Counting edge covers.
- ....

# Counting Problems

A list of problems in their search, optimization, and counting versions.

## Search problems:

- SAT.
- Find a (perfect) matching.
- Find an edge cover.
- ....

## Optimizations:

- MAX-SAT.
- Find a maximum matching.
- Find a minimum edge cover.
- ....

## Counting problems:

- #SAT.
- Counting matchings.
- Counting edge covers.
- ....

# Counting Problems

A list of problems in their search, optimization, and counting versions.

## Search problems:

- SAT.
- Find a (perfect) matching.
- Find an edge cover.
- ....

## Optimizations:

- MAX-SAT.
- Find a maximum matching.
- Find a minimum edge cover.
- ....

## Counting problems:

- #SAT.
- Counting matchings.
- Counting edge covers.
- ....



# Why counting?

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

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

# Why counting?

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

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

# Why counting?

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

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

# Why counting?

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

- Partition function in statistical physics.
- Graph polynomials.
- Sampling, learning and inference.
- 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). Instead we look for these two types of polynomial time approximation scheme:

## 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  $\text{poly}(n, 1/\varepsilon)$  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). Instead we look for these two types of polynomial time approximation scheme:

## 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  $\text{poly}(n, 1/\varepsilon)$  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.

## 1 INTRODUCTION

- Definition
- Edge Covers and Matching
- Edge Covers and Rtw-Mon-CNF Formulae
- Counting Problems

## 2 MAIN RESULT

- Previous Results and Our Result
- Counting via Marginal Probability
- Computation Tree Recursion
- Correlation Decay

# Previous Results

- For counting edge covers, only an MCMC-based FPRAS is known graphs with maximum degree 3.
- For  $\#$ DNF and counting matchings, 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.



# Previous Results

- For counting edge covers, only an MCMC-based FPRAS is known graphs with maximum degree 3.
- For  $\#$ DNF and counting matchings, 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.

# Previous Results

- For counting edge covers, only an MCMC-based FPRAS is known graphs with maximum degree 3.
- For  $\#$ DNF and counting matchings, 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.

# Previous Results

- For counting edge covers, only an MCMC-based FPRAS is known graphs with maximum degree 3.
- For  $\#$ DNF and counting matchings, 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.

# 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.

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:

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

# Devising sub-problems

## Definition (Dangling edge)

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

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

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

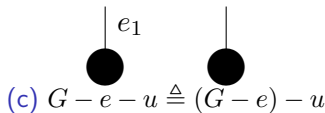
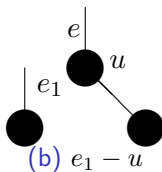
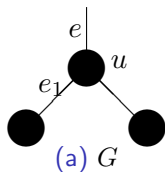


Figure : Dangling edges examples.

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  $e_i = (u_i, v_i)$ .

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

Therefore,

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

# 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  $e_i = (u_i, v_i)$ .

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

Therefore,

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

# 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  $e_i = (u_i, v_i)$ .

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

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

Therefore,

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

# 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  $e_i = (u_i, v_i)$ .

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

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

Therefore,

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

# 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  $e_i = (u_i, v_i)$ .

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

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

Therefore,

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

# 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,

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

# 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,

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

# 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,

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



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:

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

# 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:

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

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:

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

## Proposition

*Given graph  $G$ , edge  $e$  and depth  $L$ ,*

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

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.