頂点被覆とマッチングに対する最適なパラメータ化量子クエリ計算量

寺尾 樹哉[1], 森 立平[2]

1. 京都大学 数理解析研究所 M2
2. 名古屋大学 多元数理科学研究科

冬のLAシンポジウム＠京都大学
2月20日(火)

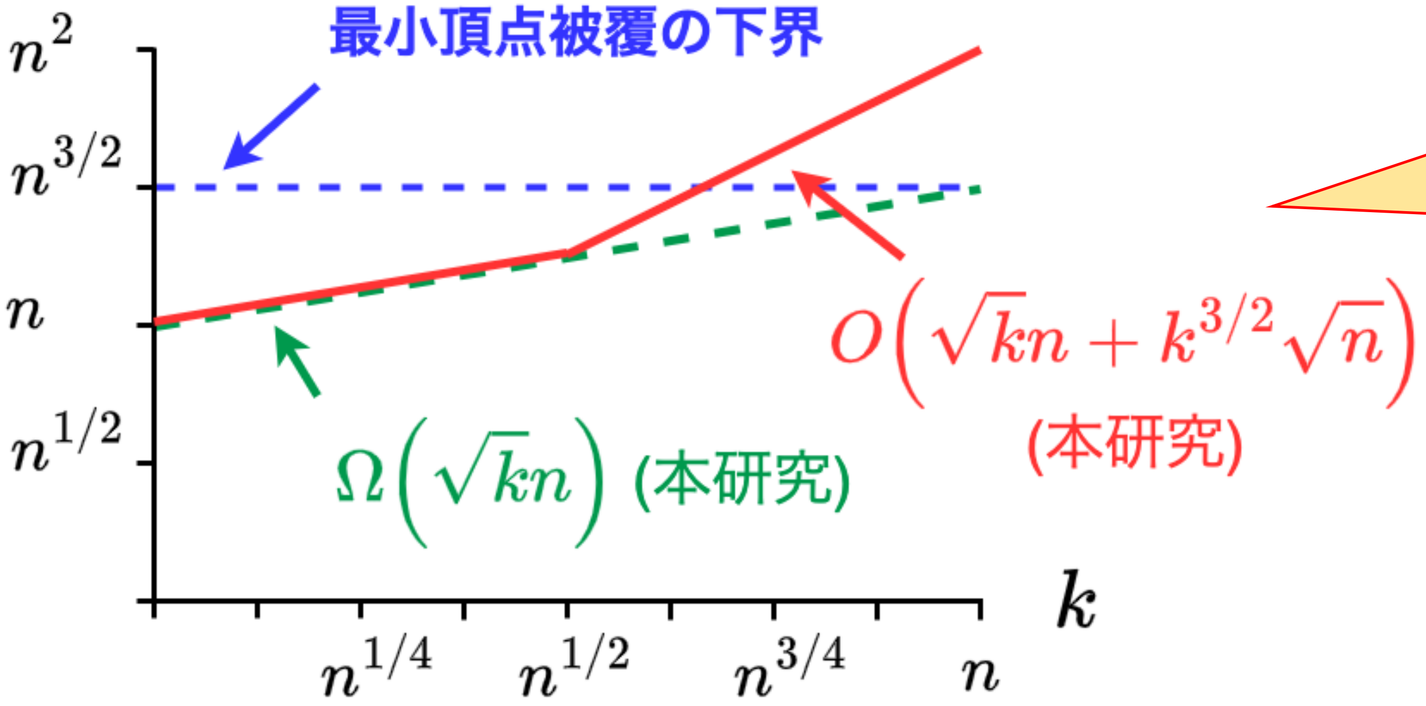
発表概要

主結果

大きさ k 以下の頂点被覆を持つかの判定問題の量子クエリ計算量

上界：$O(\sqrt{k}n + k^{3/2}\sqrt{n})$

下界：$\Omega(\sqrt{k}n)$ （$k \leq (1-\epsilon)n$ の時）



$k = O(\sqrt{n})$のとき
$\Theta(\sqrt{k}n)$で最適！！

発表概要

主結果

大きさ k 以下の頂点被覆を持つかの判定問題の量子クエリ計算量

上界：$O(\sqrt{k}n + k^{3/2}\sqrt{n})$

下界：$\Omega(\sqrt{k}n)$ （$k \leq (1-\epsilon)n$ の時）

カーネル化(kernelization)という古典アルゴリズムの手法の、
量子クエリver.を考えた！

目次

- **準備**
  - クエリ計算量
  - グラフの問題の量子クエリ計算量
- **本研究：パラメータ化量子クエリ計算量**
  - 頂点被覆
  - マッチング
- **アイデア**
  - カーネル化 (kernelization)
  - 量子クエリカーネル化
  - 頂点被覆問題のパラメータ化量子クエリアルゴリズム
- **結論**

クエリ計算量

クエリ計算量

関数fが陽にではなく、
オラクルで与えられてる！

オラクル $\mathcal{O}_f$

$f: \{1, \ldots, N\} \to \{0, 1\}$

クエリ計算量

アルゴリズム

$\boldsymbol{i}$ での値をクエリ

オラクル $\boldsymbol{O_f}$

$\boldsymbol{f}: \{\mathbf{1}, \ldots, \boldsymbol{N}\} \to \{\mathbf{0}, \mathbf{1}\}$

クエリ計算量

アルゴリズム

$\boldsymbol{i}$ での値
をクエリ

$\boldsymbol{f(i)}$を返す

オラクル $\boldsymbol{O_f}$

$\boldsymbol{f: \{1, \ldots, N\} \to \{0, 1\}}$

クエリ計算量

アルゴリズム

$\boldsymbol{i}$ での値をクエリ

$\boldsymbol{f(i)}$を返す

オラクル $\boldsymbol{O_f}$

$\boldsymbol{f: \{1, \ldots, N\} \to \{0, 1\}}$

問題例）$\boldsymbol{f(1), \ldots, f(N)}$に1になるものはあるか？

クエリ計算量

アルゴリズム

$\boldsymbol{i}$ での値
をクエリ

$\boldsymbol{f(i)}$を返す

オラクル $\boldsymbol{O_f}$

$\boldsymbol{f\colon\{1,\ldots,N\}\to\{0,1\}}$

☞ **オラクルへのクエリ回数を評価**

量子クエリ計算量

量子アルゴリズム

$|\boldsymbol{i}, b\rangle$

$|i, b \oplus \boldsymbol{f}(\boldsymbol{i})\rangle$

量子オラクル $\mathcal{O}_f$

$\boldsymbol{f}: \{\mathbf{1}, \dots, \boldsymbol{N}\} \to \{\mathbf{0}, \mathbf{1}\}$

☞ オラクルへのクエリ回数を評価

量子クエリ計算量

$$\sum_i \alpha_i \, |\boldsymbol{i}, b\rangle$$

量子アルゴリズム

$$\sum_i \alpha_i \, |i, b \oplus \boldsymbol{f}(\boldsymbol{i})\rangle$$

量子オラクル $\boldsymbol{O_f}$

$\boldsymbol{f}: \{\mathbf{1}, \dots, \boldsymbol{N}\} \to \{\mathbf{0}, \mathbf{1}\}$

☞ **オラクルへのクエリ回数を評価**

Groverのアルゴリズム

入力：**量子オラクルでアクセスする** $\boldsymbol{f}: \{\mathbf{1}, \dots, \boldsymbol{N}\} \to \{\mathbf{0}, \mathbf{1}\}$
出力：$f(i) = 1$なる$i \in \{1, \dots, N\}$を一つ

Groverのアルゴリズム

入力：**量子オラクルでアクセスする** $\boldsymbol{f}: \{\mathbf{1}, \dots, \boldsymbol{N}\} \to \{\mathbf{0}, \mathbf{1}\}$
出力：$f(i) = 1$なる$i \in \{1, \dots, N\}$を一つ

古典

$\boldsymbol{\Theta(N)}$ クエリは
誤り確率高々1/3には必要

Groverのアルゴリズム

入力：**量子オラクルでアクセスする** $\boldsymbol{f}\colon\{\mathbf{1},\ldots,\boldsymbol{N}\}\to\{\mathbf{0},\mathbf{1}\}$
出力：$f(i)=1$なる$i\in\{1,\ldots,N\}$を一つ

古典

$\boldsymbol{\Theta}(\boldsymbol{N})$ クエリは
誤り確率高々1/3には必要

量子

$\boldsymbol{O}(\sqrt{\boldsymbol{N}})$ クエリで誤り確率高々1/3
[Grover '96]

Groverのアルゴリズム

入力：**量子オラクルでアクセスする** $\boldsymbol{f}$: $\{\mathbf{1}, \ldots, \boldsymbol{N}\} \rightarrow \{\mathbf{0}, \mathbf{1}\}$
出力：$f(i) = 1$なる$i \in \{1, \ldots, N\}$を一つ

古典

$\boldsymbol{\Theta}(\boldsymbol{N})$ クエリは
誤り確率高々1/3には必要

量子

$\boldsymbol{O}(\sqrt{\boldsymbol{N}})$ クエリで誤り確率高々1/3
[Grover '96]

$\boldsymbol{\Omega}(\sqrt{\boldsymbol{N}})$ クエリは必要
[Bennett-Bernstein-Brassard-Vazirani '97]

Groverのアルゴリズム

入力：**量子オラクルでアクセスする** $\boldsymbol{f}: \{\mathbf{1}, \dots, \boldsymbol{N}\} \to \{\mathbf{0}, \mathbf{1}\}$
出力：$f(i) = 1$なる$i \in \{1, \dots, N\}$を一つ

古典

$\boldsymbol{\Theta}(\boldsymbol{N})$ クエリは
誤り確率高々1/3には必要

量子

意義①: 量子計算の優位性！

$\boldsymbol{O}(\sqrt{\boldsymbol{N}})$ クエリで誤り確率高々1/3
[Grover ’96]

$\boldsymbol{\Omega}(\sqrt{\boldsymbol{N}})$ クエリは必要
[Bennett-Bernstein-Brassard-Vazirani ’97]

Groverのアルゴリズム

入力：**量子オラクルでアクセスする** $\boldsymbol{f}: \{\mathbf{1}, \dots, \boldsymbol{N}\} \to \{\mathbf{0}, \mathbf{1}\}$
出力：$f(i) = 1$なる$i \in \{1, \dots, N\}$を一つ

古典

$\boldsymbol{\Theta}(\boldsymbol{N})$ クエリは
誤り確率高々1/3には必要

量子

意義①: 量子計算の優位性！

$\boldsymbol{O}(\sqrt{\boldsymbol{N}})$ クエリで誤り確率高々1/3
[Grover '96]

意義②: 量子計算の限界！

$\boldsymbol{\Omega}(\sqrt{\boldsymbol{N}})$ クエリは必要
[Bennett-Bernstein-Brassard-Vazirani '97]

グラフの問題の量子クエリ計算量

隣接行列モデル
量子オラクルで隣接行列 $E_M: \{1, \ldots, n\} \times \{1, \ldots, n\} \rightarrow \{0,1\}$にアクセス

$$\boldsymbol{E_M(u,v) = 1 \Leftrightarrow (u,v) \in E(G)}$$

n =頂点数

グラフの問題の量子クエリ計算量

隣接行列モデル

量子オラクルで隣接行列 $E_M: \{1,\ldots,n\}\times\{1,\ldots,n\} \to \{0,1\}$にアクセス

2頂点u, vの間に辺があるかをクエリ

n =頂点数

先行研究:グラフの問題の量子クエリ計算量

古典では$\boldsymbol{\Theta(n^2)}$の問題でも…

n =頂点数

先行研究:グラフの問題の量子クエリ計算量

古典では$\boldsymbol{\Theta(n^2)}$の問題でも…

- k-クリーク: $\boldsymbol{\tilde{O}(n^{2-2/k})}$ [Magniez-Santha-Szegedy ’05]
- 連結性判定: $\boldsymbol{\Theta(n^{3/2})}$ [Dürr-Heiligman-Høyer-Mhalla ’06]
- 平面性判定: $\boldsymbol{\Theta(n^{3/2})}$ [Ambainis et al. ’08]
- 最大マッチング: $\boldsymbol{O(n^{7/4})}$ [Kimmel-Witter ’21], $\Omega(n^{3/2})$ [Zhang ’04]
- 最小カット: $\boldsymbol{\Theta(n^{3/2})}$ [Apers-Lee ’21]

n =頂点数

k-頂点被覆問題

入力：グラフ G と整数 k
問題：大きさ k 以下の**頂点被覆** $S \subseteq V$ が G に存在するか？

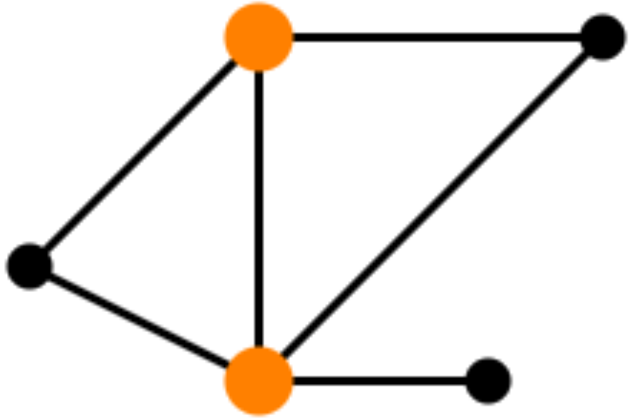
Gのどの辺も端点の少なくとも一方はSに属す

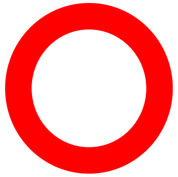
k-頂点被覆問題

入力：グラフ G と整数 k
問題：大きさ k 以下の**頂点被覆** $S \subseteq V$ が G に存在するか？

Gのどの辺も端点の少なくとも一方はSに属す

例)



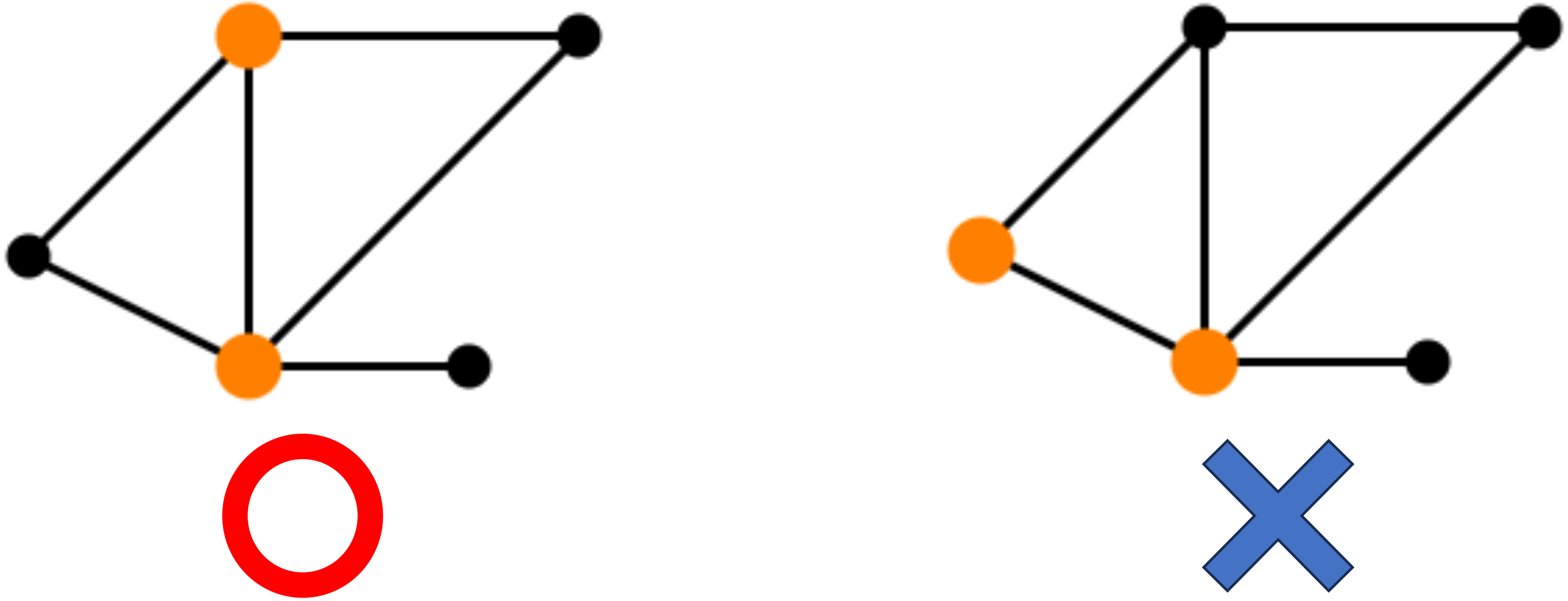


k-頂点被覆問題

入力：グラフ G と整数 k
問題：大きさ k 以下の**頂点被覆** $S \subseteq V$ が G に存在するか？

Gのどの辺も端点の少なくとも一方はSに属す

例)



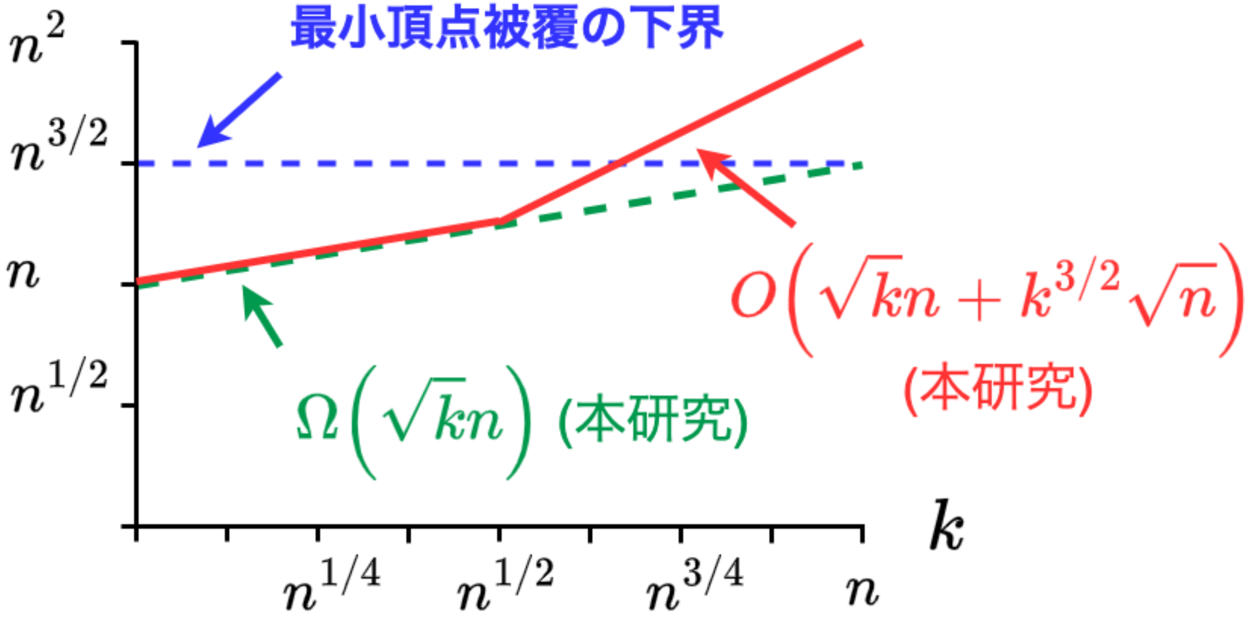
主結果①: 頂点被覆問題に対する パラメータ化量子クエリ計算量

定理

大きさ k 以下の頂点被覆を持つかの判定問題の量子クエリ計算量

上界：$O(\sqrt{kn} + k^{3/2}\sqrt{n})$

下界：$\Omega(\sqrt{kn})$ （$k \leq (1-\epsilon)n$ の時）



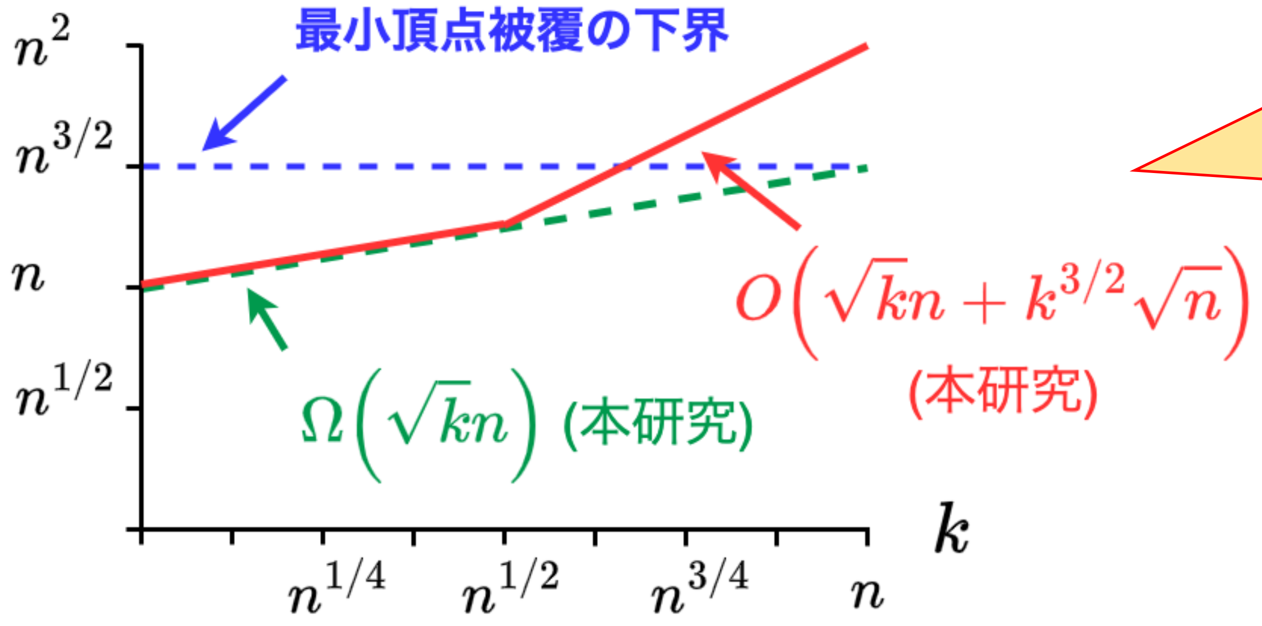
主結果①: 頂点被覆問題に対する パラメータ化量子クエリ計算量

<u>定理</u>

大きさ k 以下の頂点被覆を持つかの判定問題の量子クエリ計算量

上界：$O(\sqrt{kn} + k^{3/2}\sqrt{n})$

下界：$\Omega(\sqrt{kn})$ （$k \le (1-\epsilon)n$ の時）



$k = O(\sqrt{n})$のとき
$\Theta(\sqrt{kn})$で最適！！

主結果①: 頂点被覆問題に対する パラメータ化量子クエリ計算量

定理

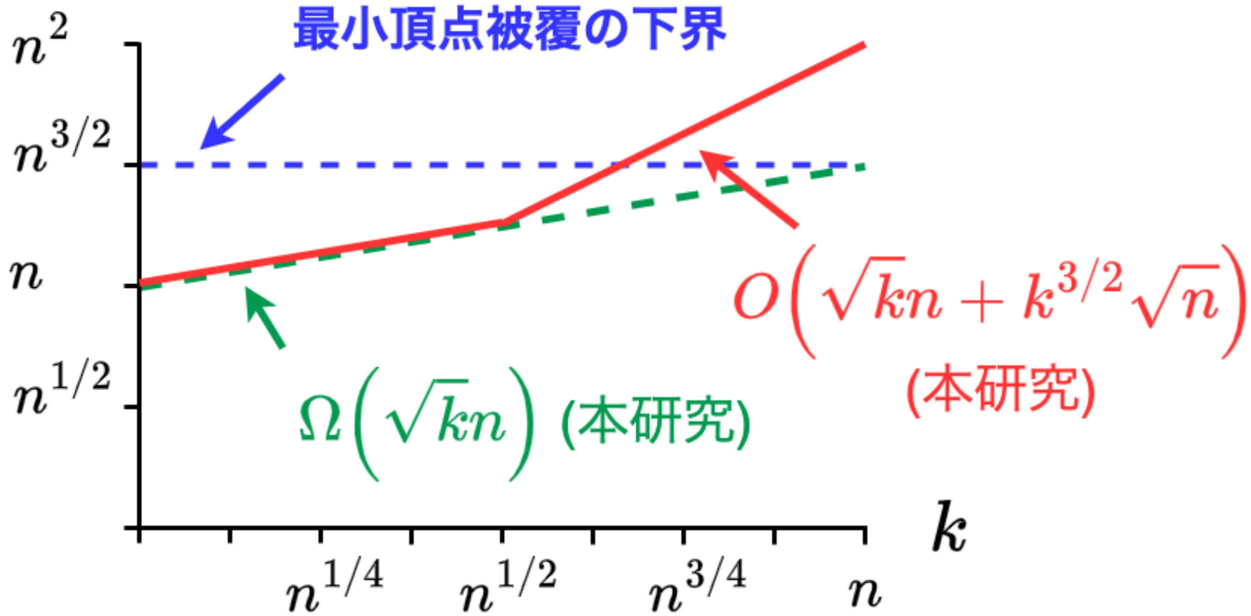
大きさ k 以下の頂点被覆を持つかの判定問題の量子クエリ計算量

上界：$O(\sqrt{kn} + k^{3/2}\sqrt{n})$

下界：$\Omega(\sqrt{kn})$（$k \leq (1-\epsilon)n$ の時）

意義

- **最小頂点被覆の上界$O(n^2)$, 下界$\Omega(n^{3/2})$** [Zhang '04]**を パラメータ化により改善**



主結果①: 頂点被覆問題に対する パラメータ化量子クエリ計算量

定理

大きさ k 以下の頂点被覆を持つかの判定問題の量子クエリ計算量

上界：$O(\sqrt{kn} + k^{3/2}\sqrt{n})$

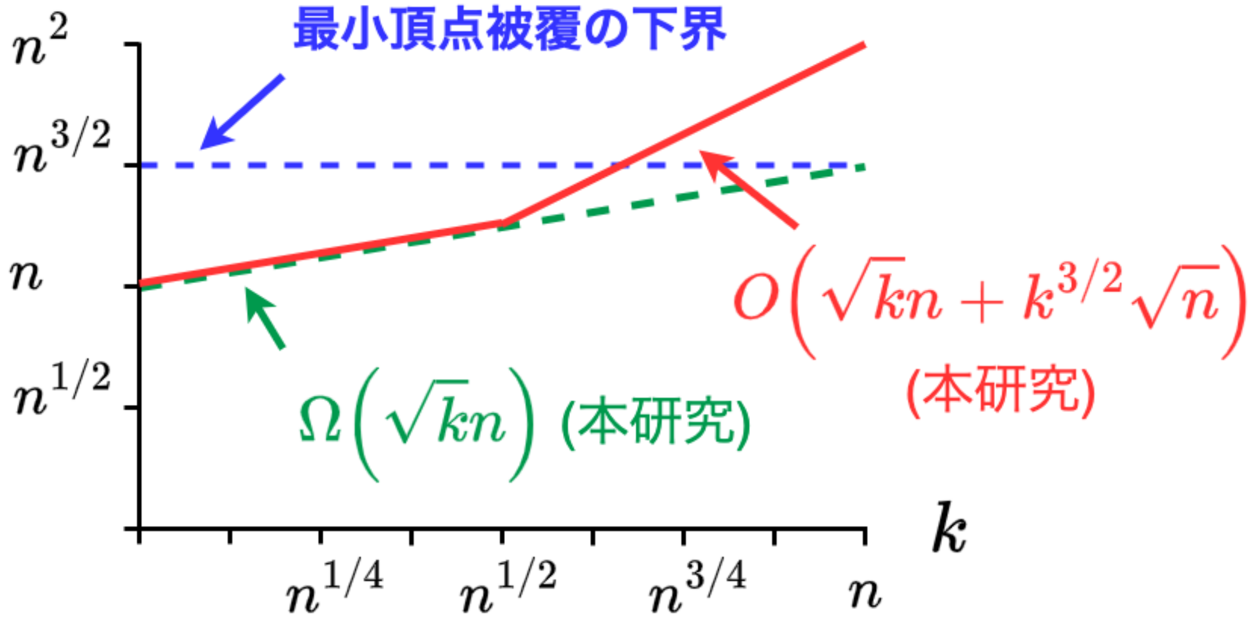
下界：$\Omega(\sqrt{kn})$ （$k \leq (1-\epsilon)n$ の時）

意義

- **最小頂点被覆の上界$O(n^2)$, 下界$\Omega(n^{3/2})$** [Zhang '04]**をパラメータ化により改善**

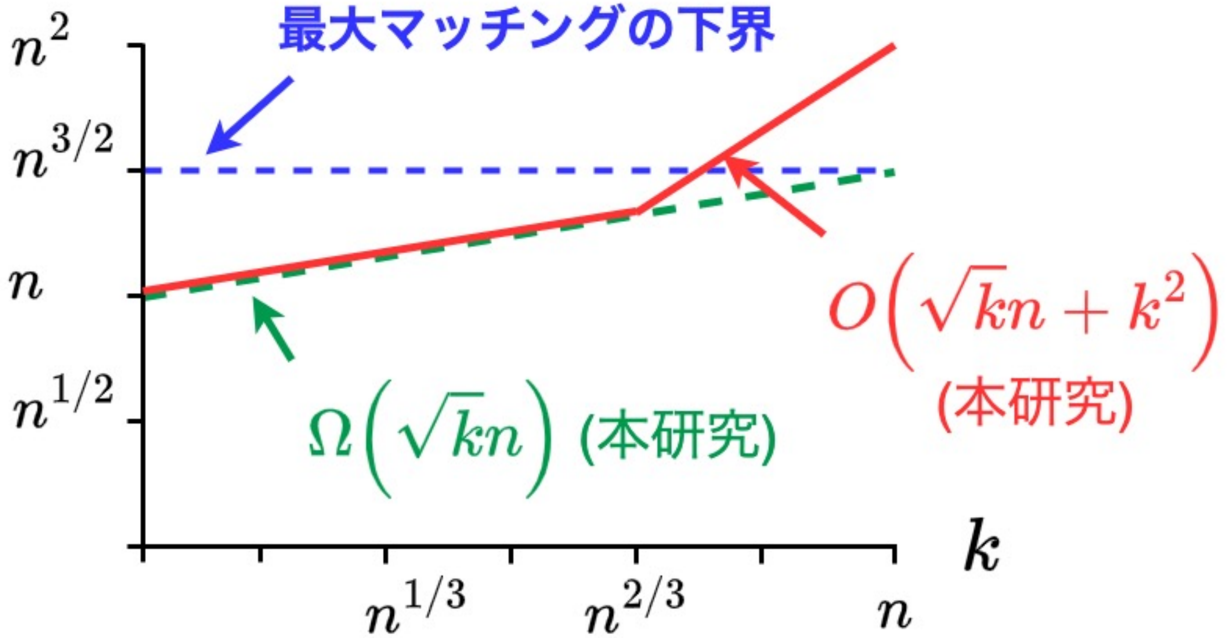
手法

- **量子クエリカーネル化**



主結果②:マッチング問題に対する パラメータ化量子クエリ計算量

定理
大きさ k 以上のマッチングを持つかの判定問題の量子クエリ計算量
上界：$O(\sqrt{k}n + k^2)$
下界：$\Omega(\sqrt{k}n)$



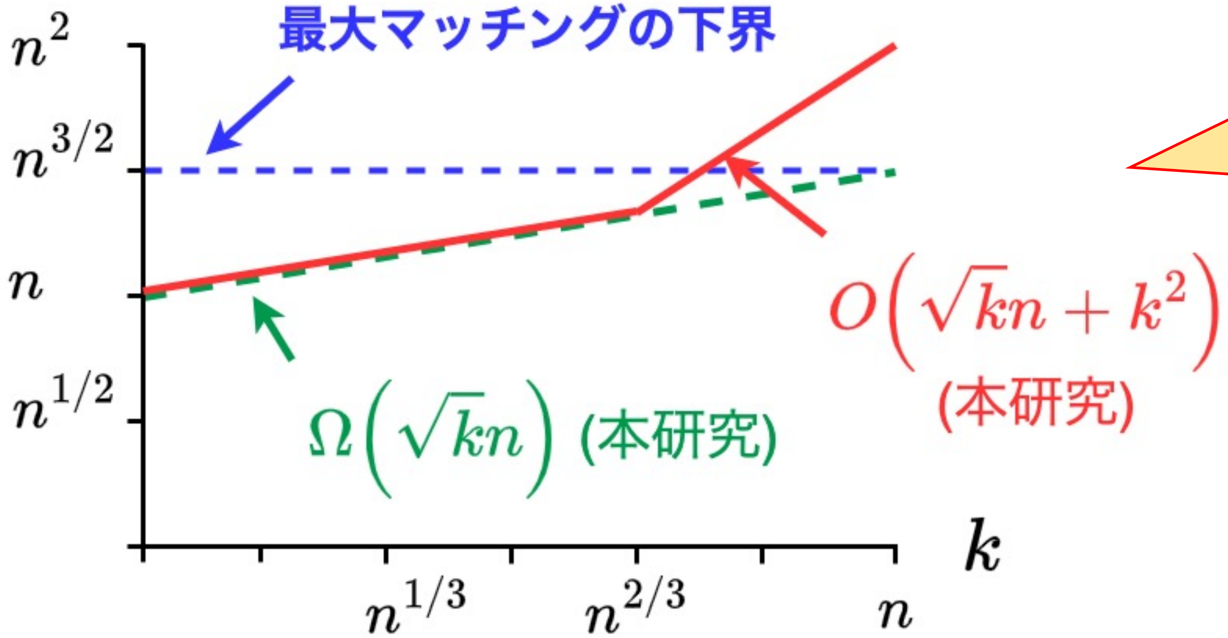
主結果②:マッチング問題に対する パラメータ化量子クエリ計算量

定理

大きさ k 以上のマッチングを持つかの判定問題の量子クエリ計算量

上界：$O(\sqrt{k}n + k^2)$

下界：$\Omega(\sqrt{k}n)$



$k = O(n^{2/3})$のとき
$\Theta(\sqrt{k}n)$で最適！！

主結果②:マッチング問題に対する パラメータ化量子クエリ計算量

定理

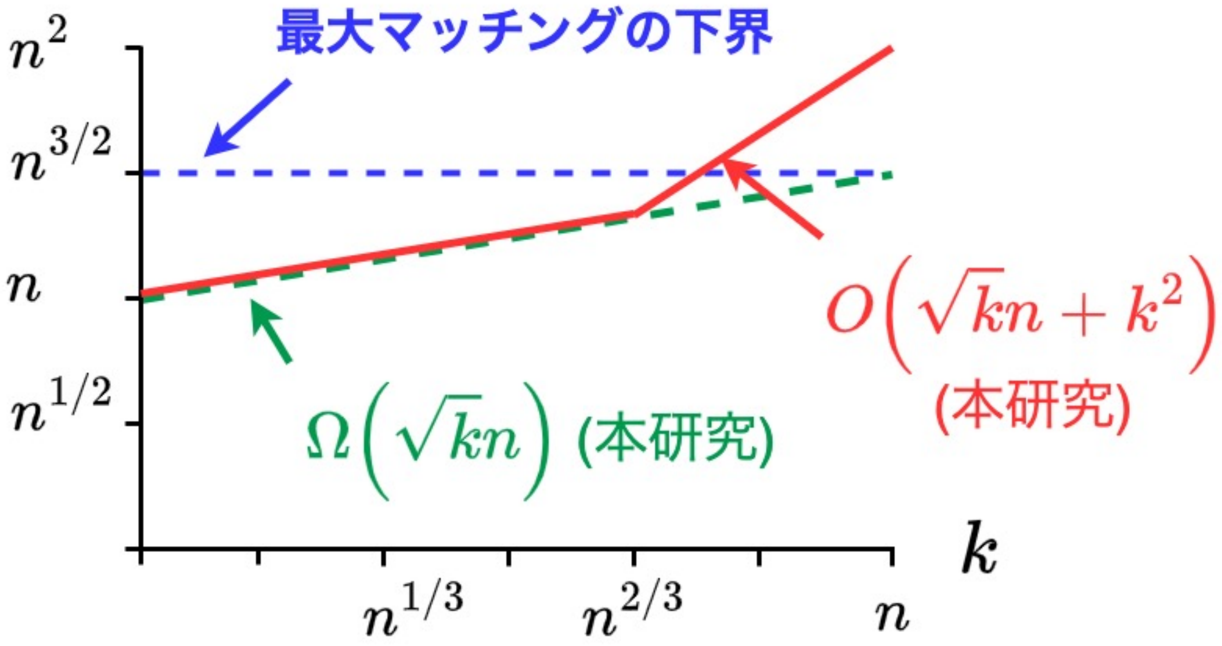
大きさ k 以上のマッチングを持つかの判定問題の量子クエリ計算量

上界：$O(\sqrt{k}n + k^2)$

下界：$\Omega(\sqrt{k}n)$

意義

- **最大マッチングの上界$O(n^{7/4})$** [Kimmel-Witter '21], **下界$\Omega(n^{3/2})$** [Zhang '04]**を パラメータ化により改善**



主結果②:マッチング問題に対する パラメータ化量子クエリ計算量

定理

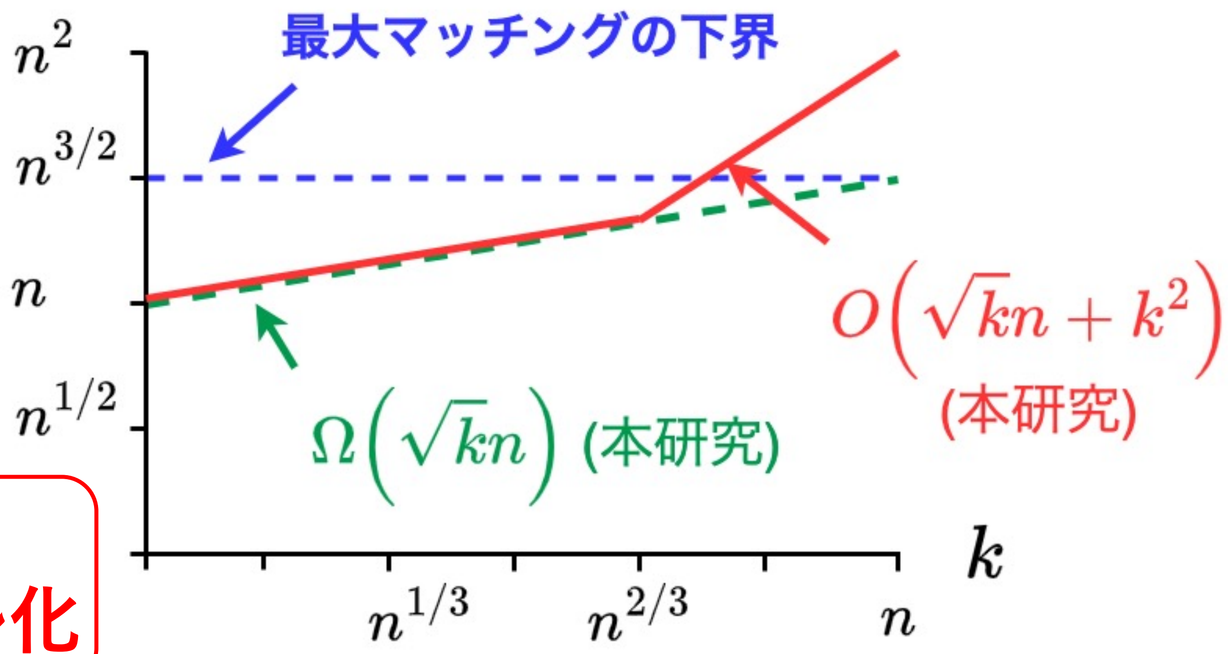
大きさ k 以上のマッチングを持つかの判定問題の量子クエリ計算量

上界：$O(\sqrt{k}n + k^2)$

下界：$\Omega(\sqrt{k}n)$

意義

- **最大マッチングの上界$O(n^{7/4})$** [Kimmel-Witter '21], **下界$\Omega(n^{3/2})$** [Zhang '04]**をパラメータ化により改善**



手法

- **増加路探索 ＋ 量子クエリカーネル化**

カーネル化 (kernelization)

入力：インスタンス (G, k)
出力：別の**等価で小さな**インスタンス (G', k')

カーネル化 (kernelization)

入力：インスタンス (G, k)
出力：別の**等価で小さな**インスタンス (G', k')

- **(G, k)がYesインスタンス $\Leftrightarrow$ (G', k')がYesインスタンス**

カーネル化 (kernelization)

入力：インスタンス (G, k)
出力：別の**等価で****小さな**インスタンス (G', k')

- **(G, k)がYesインスタンス $\Leftrightarrow$ (G', k')がYesインスタンス**
- **G'のサイズ $\leq f(k)$**
- $k' \leq g(k)$

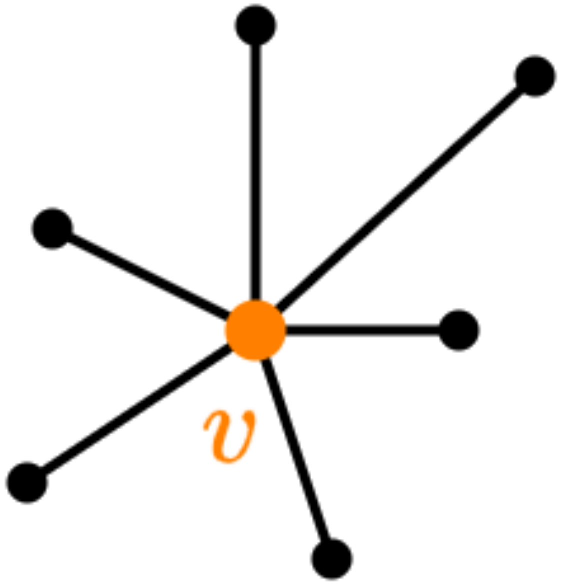
Buss-Goldsmithによるk-頂点被覆のカーネル化

Rule 1. 孤立点 v がある時、$(G,k) \to (G - v, k)$

Buss-Goldsmithによるk-頂点被覆のカーネル化

Rule 1. 孤立点 v がある時、$(G,k) \rightarrow (G - v, k)$

Rule 2. **次数 $k+1$ 以上の頂点 v** がある時、

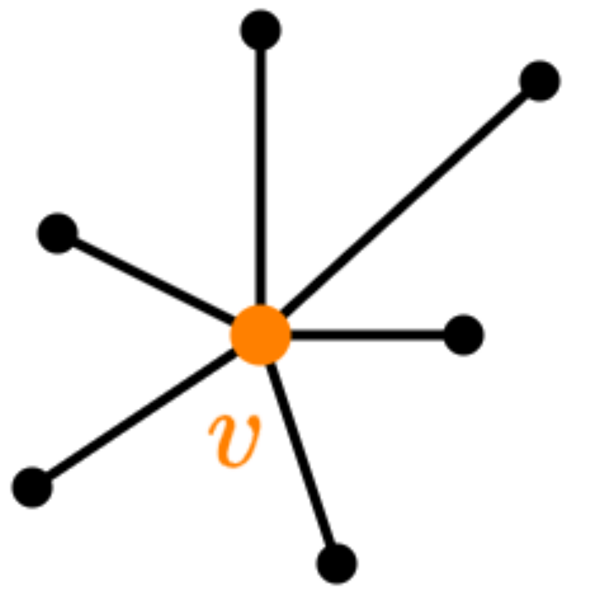


Buss-Goldsmithによるk-頂点被覆のカーネル化

Rule 1. 孤立点 v がある時、$(G, k) \to (G - v, k)$

Rule 2. **次数 $k+1$ 以上の頂点 v** がある時、

vを選ばなければ、
vと接続する頂点を全て
選ばないといけない

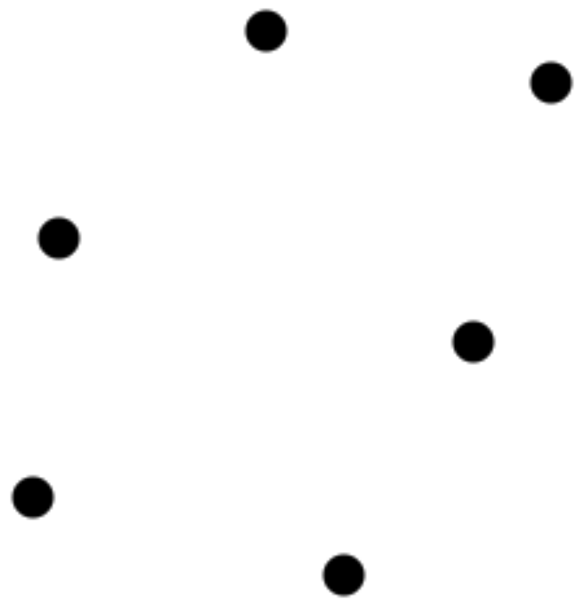


Buss-Goldsmithによるk-頂点被覆のカーネル化

Rule 1. 孤立点 v がある時、$(G, k) \to (G - v, k)$

Rule 2. **次数 $k+1$ 以上の頂点 v** がある時、$\boldsymbol{(G, k) \to (G - v, k - 1)}$

vを選ばなければ、
vと接続する頂点を全て
選ばないといけない

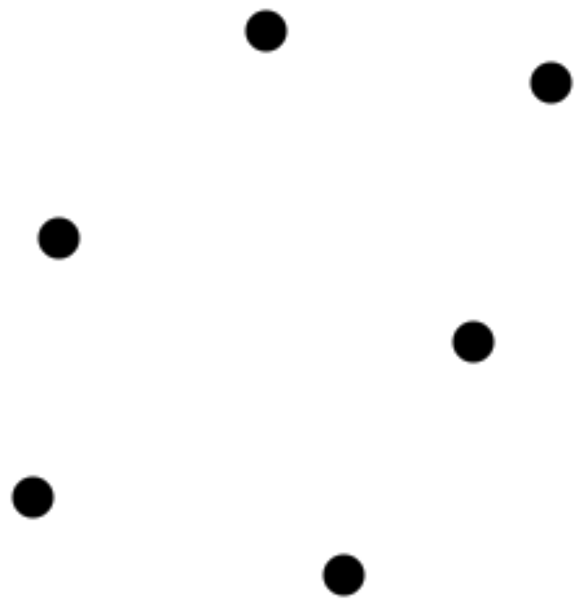


Buss-Goldsmithによるk-頂点被覆のカーネル化

Rule 1. 孤立点 v がある時、$(G, k) \to (G - v, k)$

Rule 2. **次数 $\boldsymbol{k+1}$ 以上の頂点 $\boldsymbol{v}$** がある時、$(G, k) \to (G - v, k-1)$

**$\boldsymbol{v}$を選ばなければ、
$\boldsymbol{v}$と接続する頂点を全て
選ばないといけない**



**事実: Rule1, 2の適用後 $\boldsymbol{G}$ の辺の本数が
$\boldsymbol{k^2}$本より多いならNoインスタンス**

新しいアプローチ: 量子クエリカーネル化

入力：**量子オラクルでアクセスする**インスタンス (G,k)

新しいアプローチ: 量子クエリカーネル化

入力：**量子オラクルでアクセスする**インスタンス (G, k)
出力：**ビット列として**別の**等価な**インスタンス (G', k')を得る

新しいアプローチ: 量子クエリカーネル化

入力：**量子オラクルでアクセスする**インスタンス (G, k)
出力：**ビット列として**別の**等価な**インスタンス (G', k')を得る

- **(G, k)がYesインスタンス $\Leftrightarrow$ (G', k')がYesインスタンス**

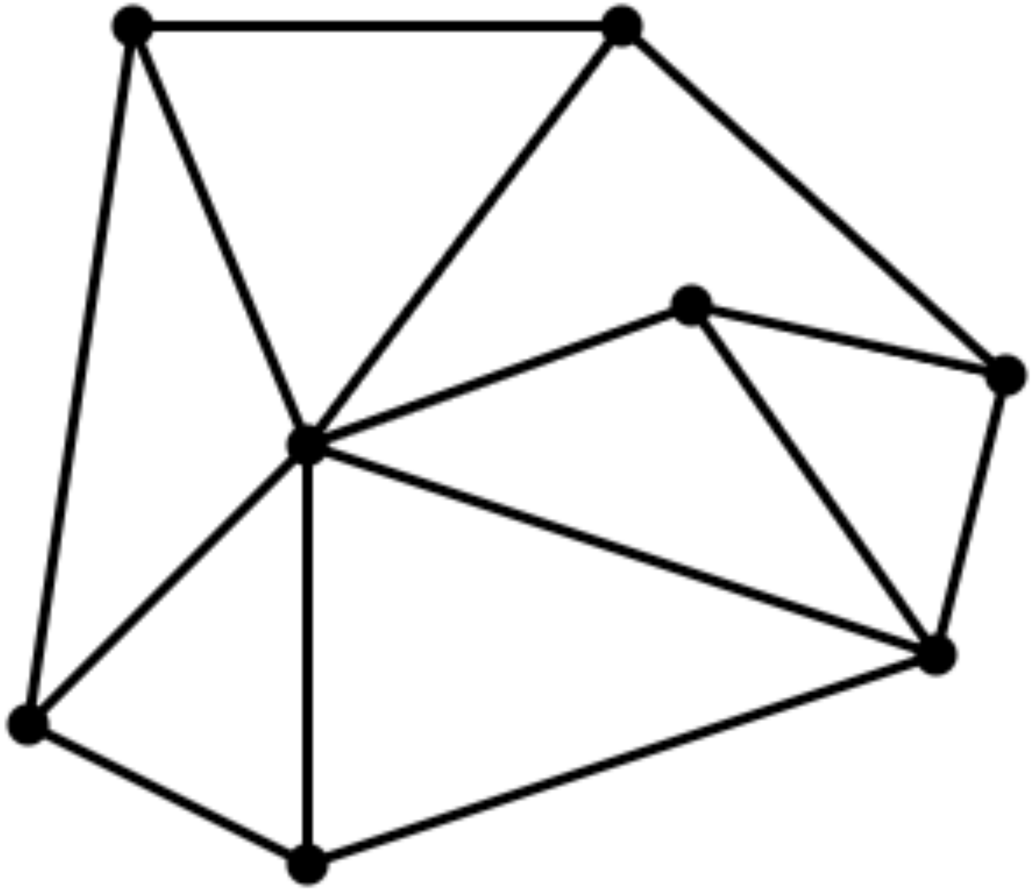
新しいアプローチ: 量子クエリカーネル化

入力：**量子オラクルでアクセスする**インスタンス (G, k)
出力：**ビット列として**別の**等価な**インスタンス (G', k')を得る

- (G, k)**がYesインスタンス** $\Leftrightarrow$ (G', k')**がYesインスタンス**

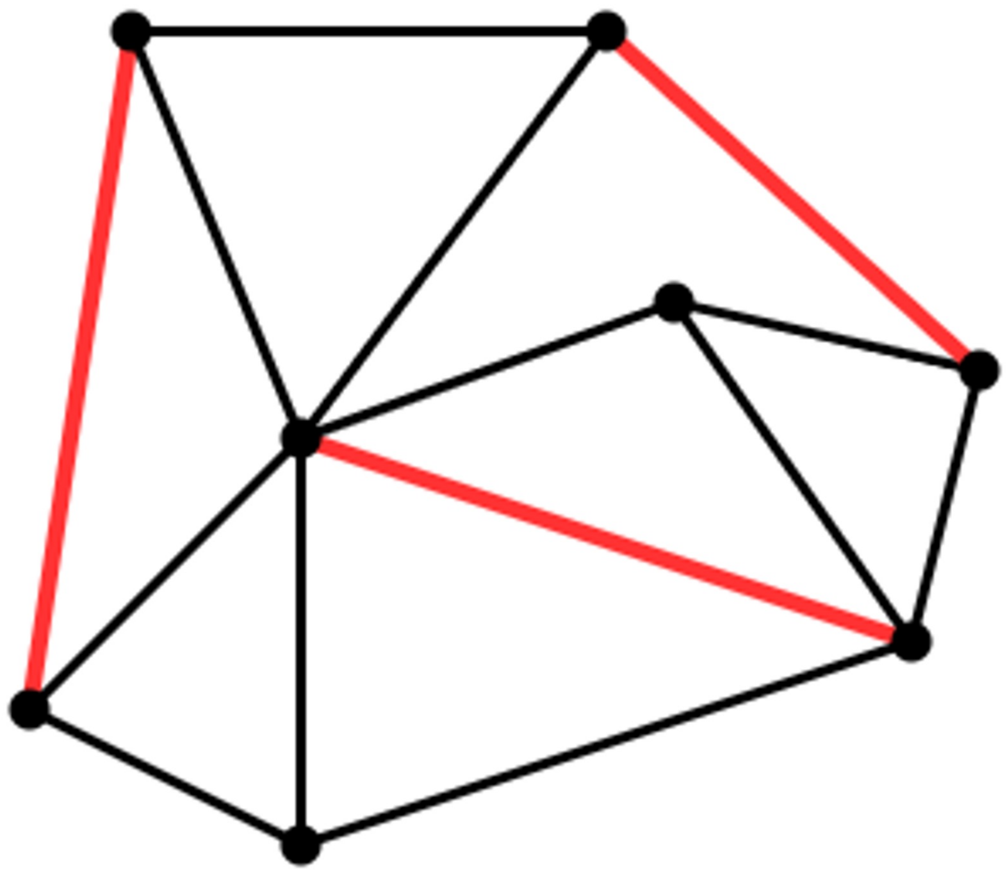
☞ **量子クエリカーネル化後は、(G', k')に**
古典のアルゴリズムを適用するだけ！

k-頂点被覆問題に対する
量子に適した(古典の)カーネル化



k-頂点被覆問題に対する量子に適した(古典の)カーネル化

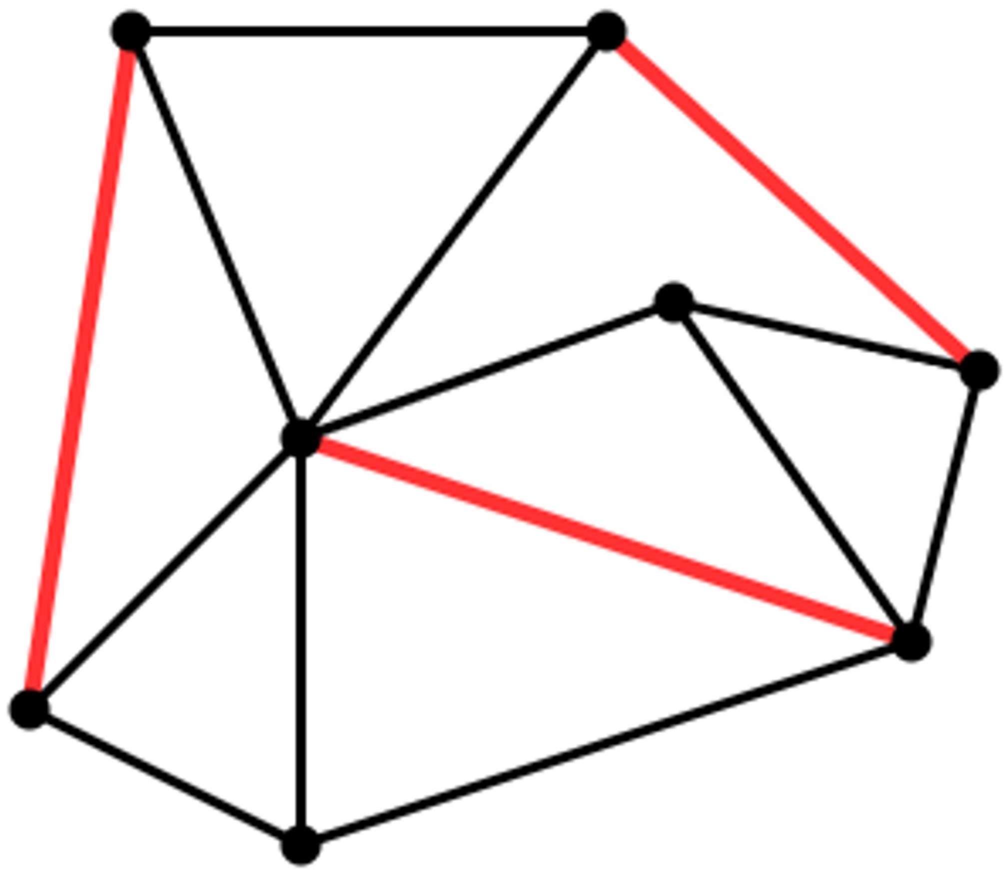
Step1　**極大マッチングM**をみつける



k-頂点被覆問題に対する量子に適した(古典の)カーネル化

Step1　**極大マッチングM**をみつける

if $|M| > k$: then Noインスタンス



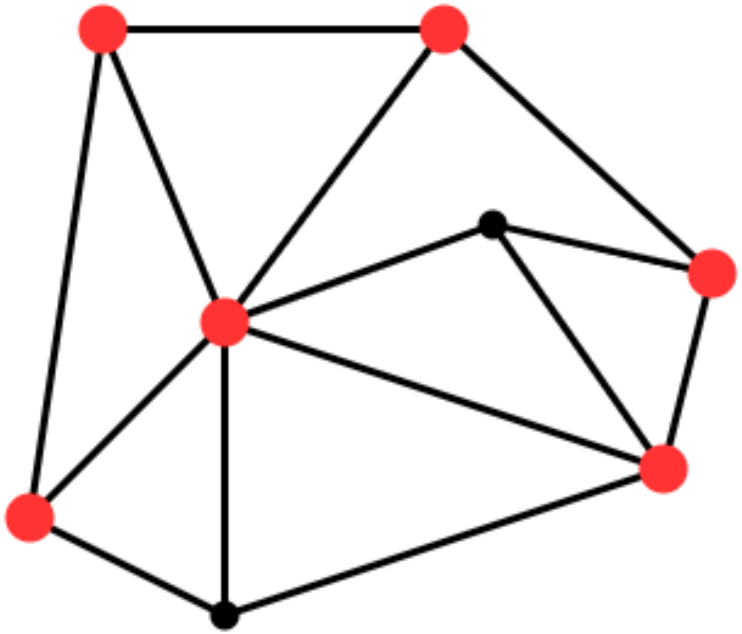
k-頂点被覆問題に対する量子に適した(古典の)カーネル化

Step1　極大マッチングMをみつける

if $|M| > k$: then Noインスタンス

Step2　**Mに属す辺の各端点vのみ**に対して、**Rule 2**を適用

Rule 2 **次数 $k+1$ 以上の頂点 v** がある時、$(G, k) \to (G - v, k-1)$



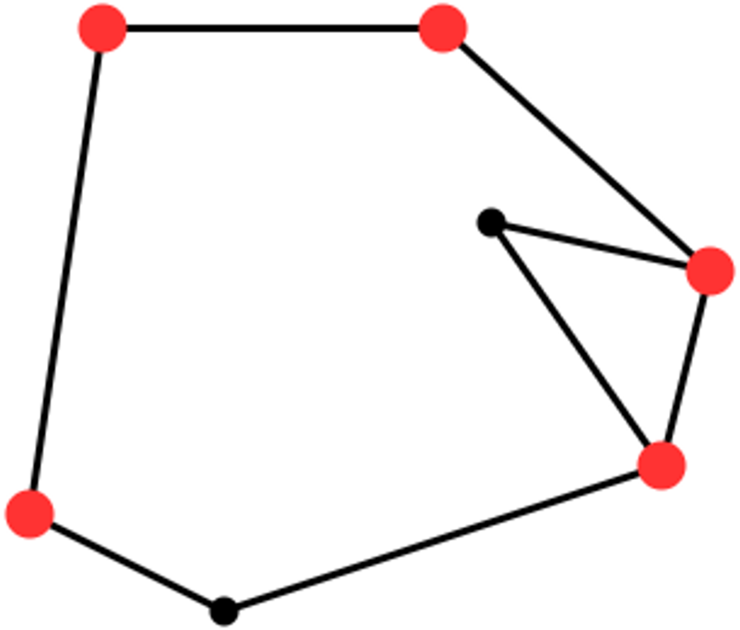
k-頂点被覆問題に対する量子に適した(古典の)カーネル化

Step1　極大マッチングMをみつける

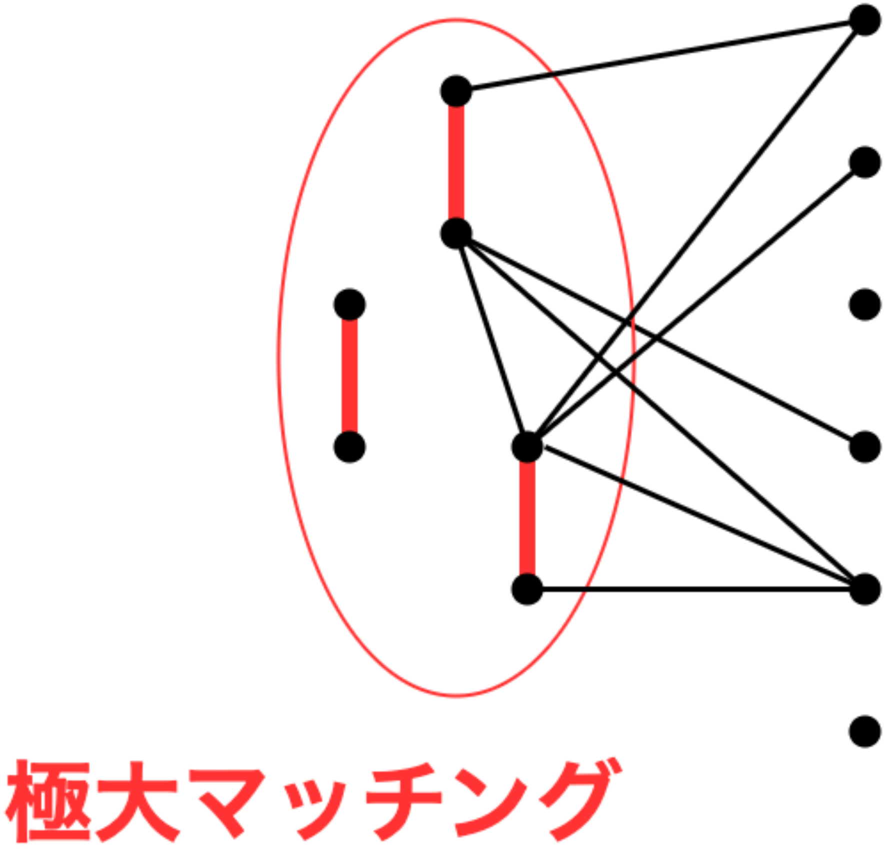
if $|M| > k$: then Noインスタンス

Step2　**Mに属す辺の各端点vのみ**に対して、**Rule 2**を適用

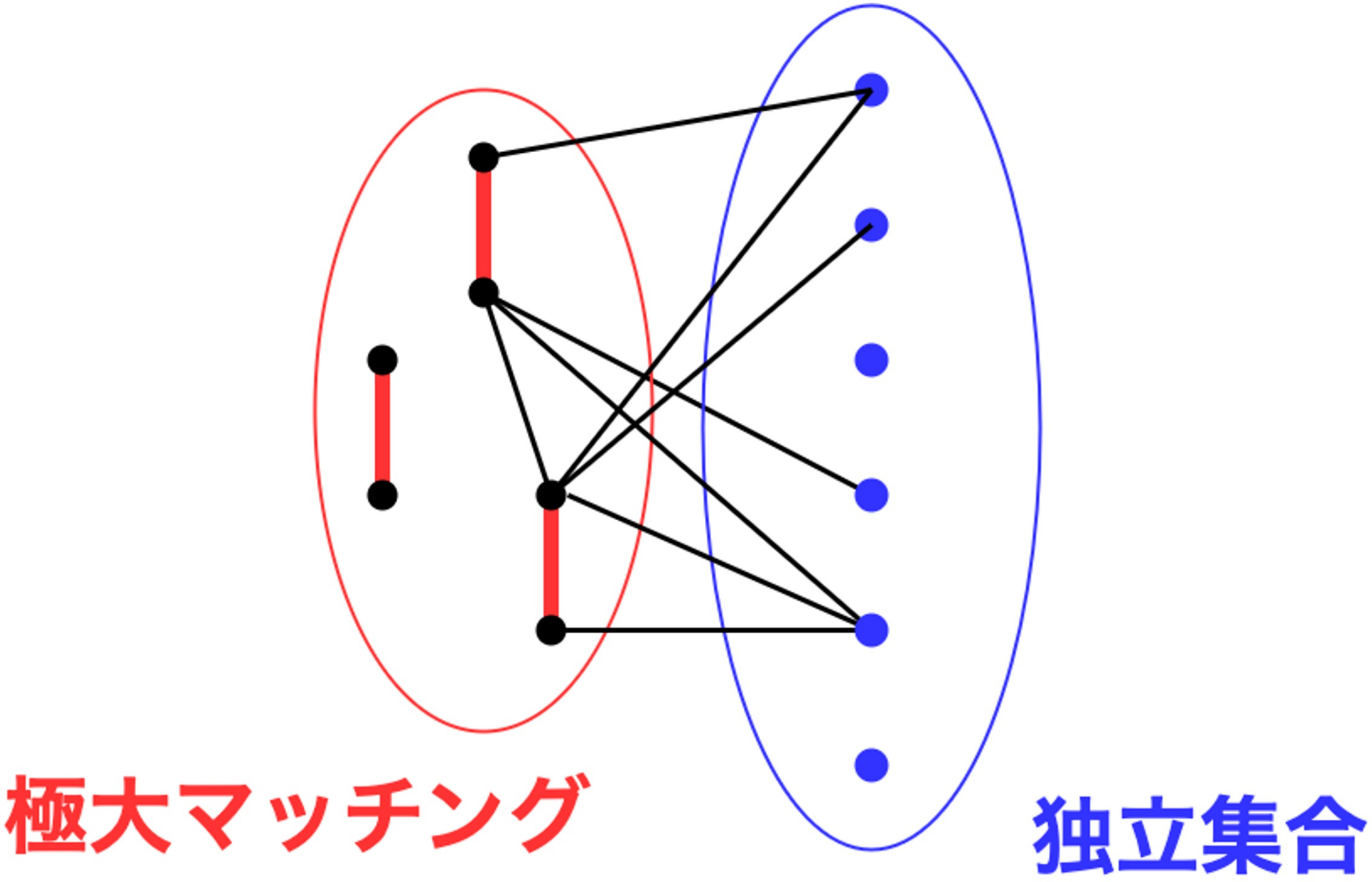
Rule 2 **次数 $\boldsymbol{k+1}$ 以上の頂点 $\boldsymbol{v}$** がある時、$(G, k) \rightarrow (G - v, k-1)$



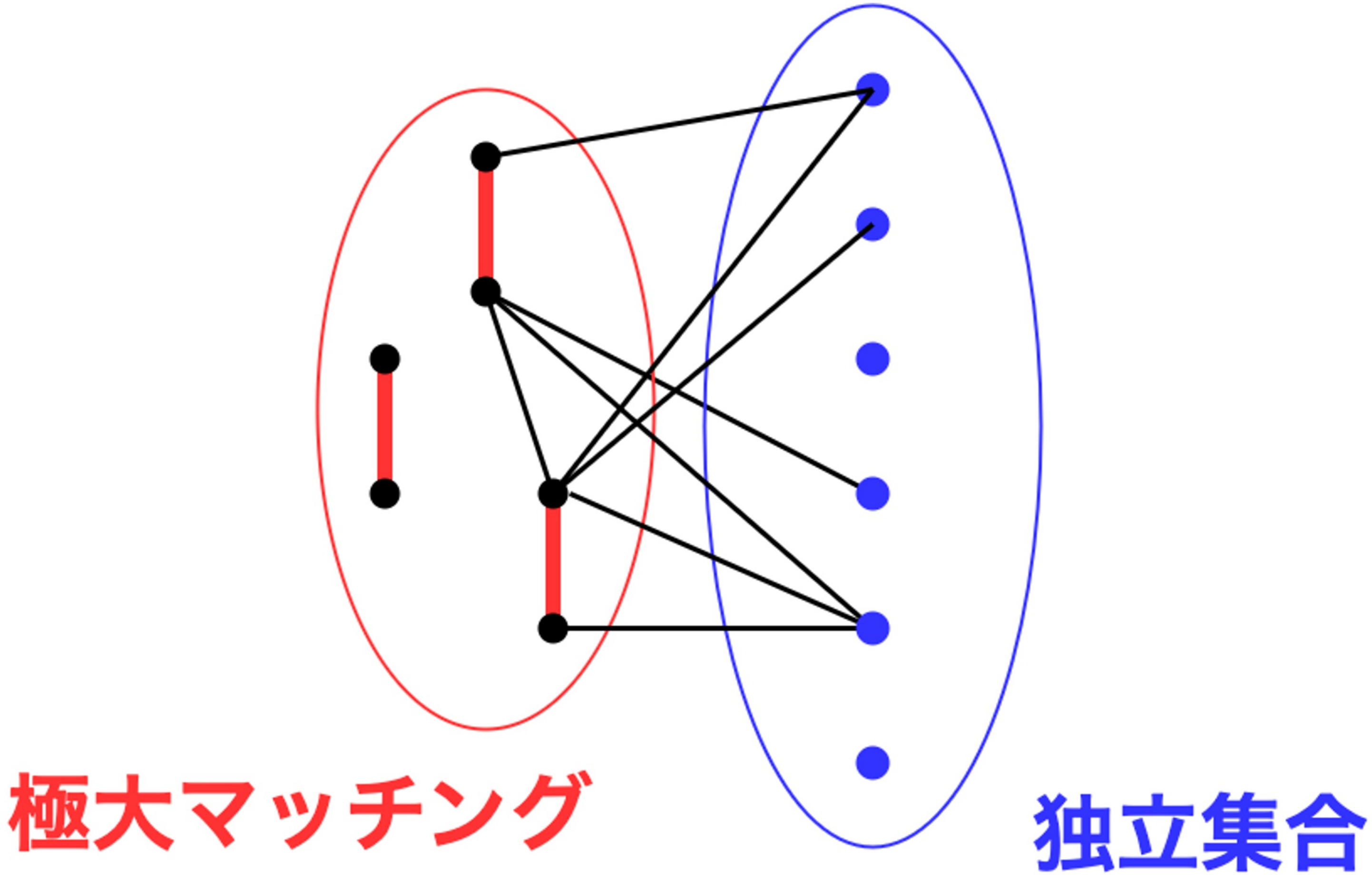
重要な観察: 極大マッチングの構造



重要な観察: 極大マッチングの構造



重要な観察: 極大マッチングの構造



☞ すべての辺は、極大マッチングの端点を端点としてもつ！

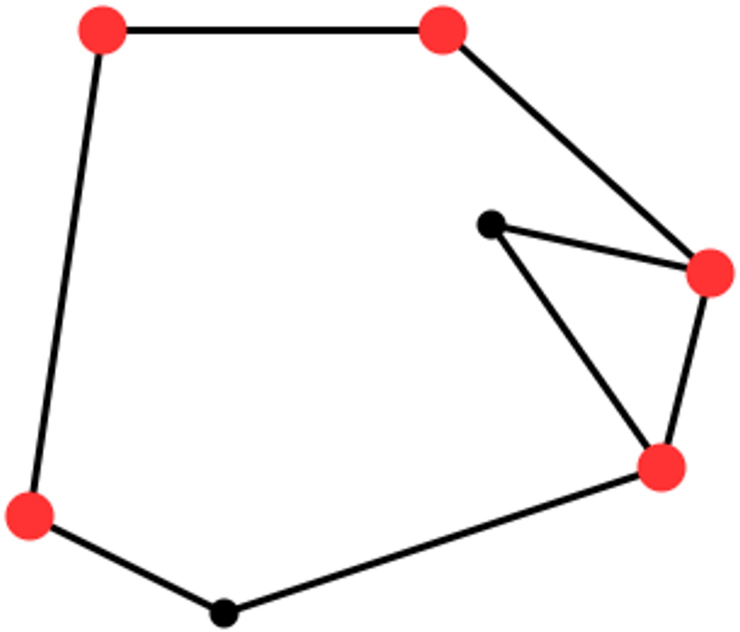
k-頂点被覆問題に対する量子に適した(古典の)カーネル化

Step1　極大マッチングMをみつける

if $|M| > k$:　then Noインスタンス

Step2　**Mに属す辺の各端点vのみ**に対して、**Rule 2**を適用

Rule 2 **次数 $k+1$ 以上の頂点 v** がある時、$(G, k) \rightarrow (G - v, k-1)$



補題:
Step1, 2の後、
Gの辺の本数は$2k^2$本以下

k-頂点被覆問題に対する量子クエリカーネル化

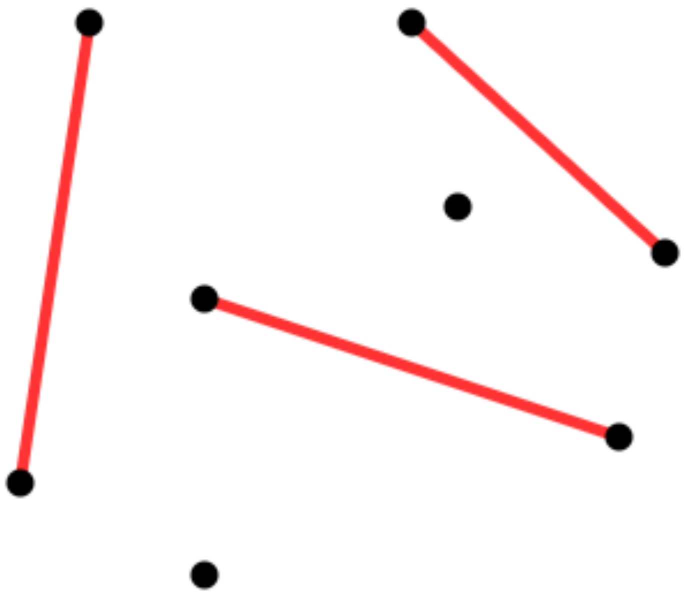
k-頂点被覆問題に対する量子クエリカーネル化

?

クエリしないと、グラフについて
何も分からない…

k-頂点被覆問題に対する量子クエリカーネル化

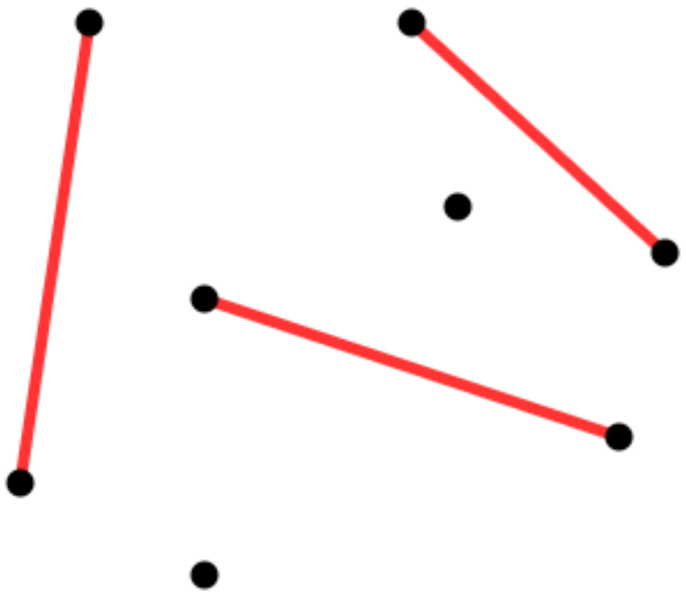
Step1 サイズ $k+1$ 以上のマッチング or サイズ k 以下の極大マッチング をみつける



補題: Step1は $\boldsymbol{O(\sqrt{kn})}$ クエリでできる

k-頂点被覆問題に対する量子クエリカーネル化

Step1　サイズ $k+1$ 以上のマッチング or サイズ k 以下の極大マッチング をみつける



補題: Step1は $\boldsymbol{O(\sqrt{kn})}$ クエリでできる

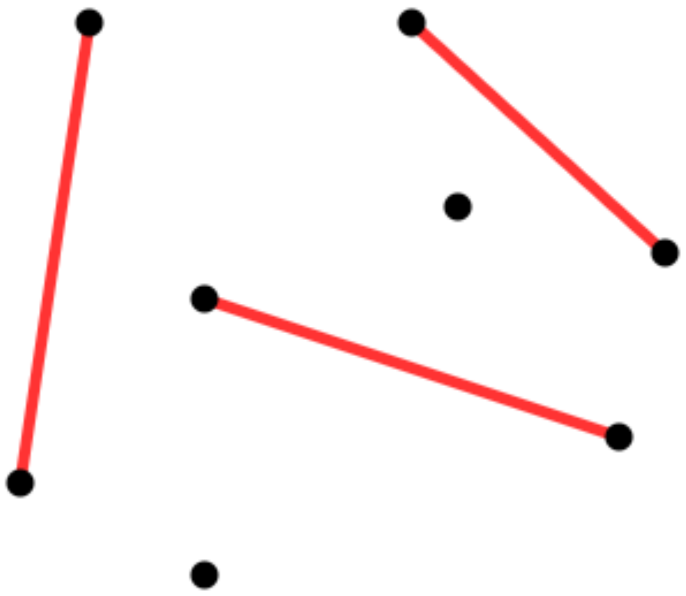
k-頂点被覆問題に対する量子クエリカーネル化

サイズ $k+1$ 以上のマッチング

Step1 or をみつける

サイズ k 以下の極大マッ

Noインスタンス



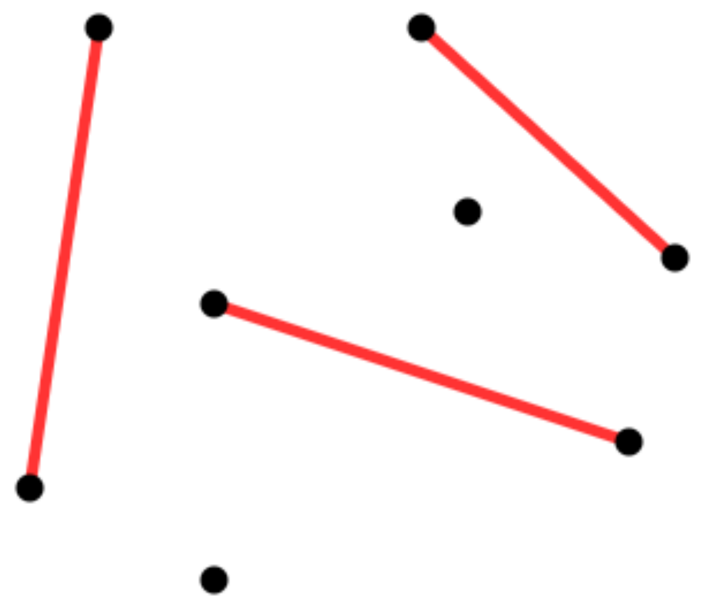
補題: Step1は$O(\sqrt{kn})$クエリでできる

k-頂点被覆問題に対する量子クエリカーネル化

サイズ $k+1$ 以上のマッチング

Step1　　　or　　　をみつける

サイズ k 以下の極大マッチングM



補題: Step1は$\boldsymbol{O(\sqrt{kn})}$クエリでできる

k-頂点被覆問題に対する量子クエリカーネル化

サイズ $k+1$ 以上のマッチング

Step1　or　をみつける

サイズ k 以下の極大マッチング M

Step2　**M に属す辺の各端点 v のみ**に対して、

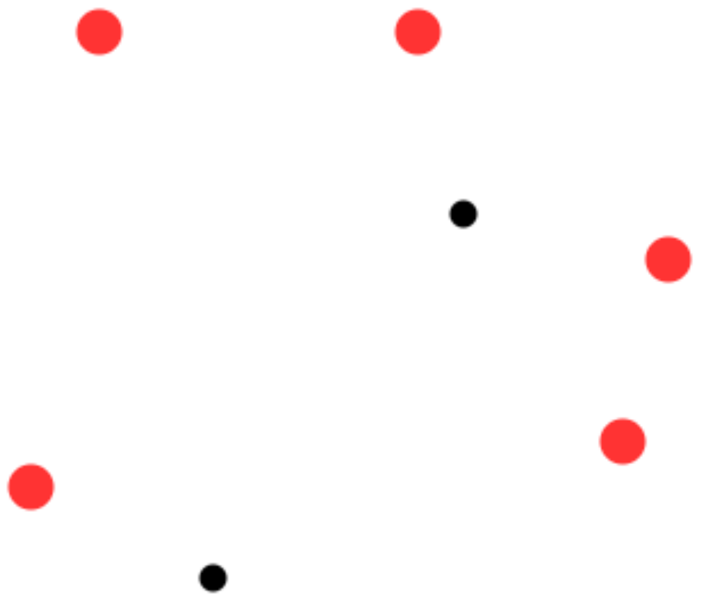
k-頂点被覆問題に対する量子クエリカーネル化

Step1　サイズ $k+1$ 以上のマッチング or **サイズ k 以下の極大マッチングM** をみつける

Step2　**Mに属す辺の各端点vのみ**に対して、

if vの次数 $> k$: then vを削除, $k \leftarrow k-1$

else: vと接続する辺を全てみつける



k-頂点被覆問題に対する量子クエリカーネル化

サイズ $k+1$ 以上のマッチング

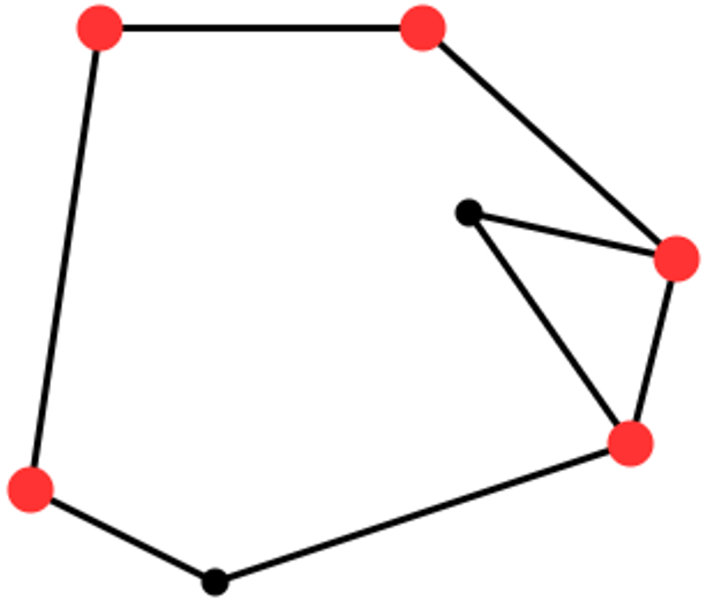
Step1　　or　　をみつける

サイズ k 以下の極大マッチングM

Step2　**Mに属す辺の各端点vのみ**に対して、

if vの次数 $> k$: then vを削除, $k \leftarrow k-1$

else: vと接続する辺を全てみつける



k-頂点被覆問題に対する量子クエリカーネル化

サイズ $k+1$ 以上のマッチング

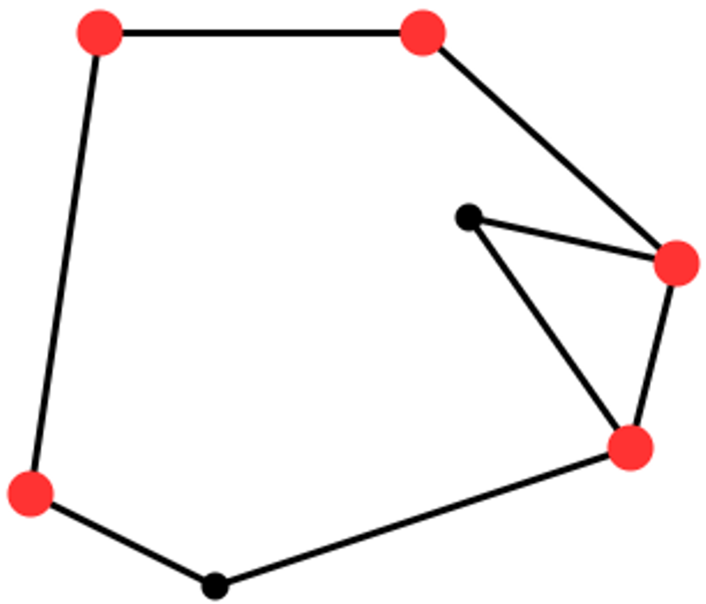
Step1　　　　　　　　or　　　　　　　　をみつける

サイズ k 以下の極大マッチングM

Step2　**Mに属す辺の各端点vのみ**に対して、

if vの次数 $> k$: then vを削除, $k \leftarrow k-1$

else: vと接続する辺を全てみつける



古典のカーネル化と同じカーネルを得た！

k-頂点被覆問題に対する量子クエリカーネル化

サイズ $k+1$ 以上のマッチング

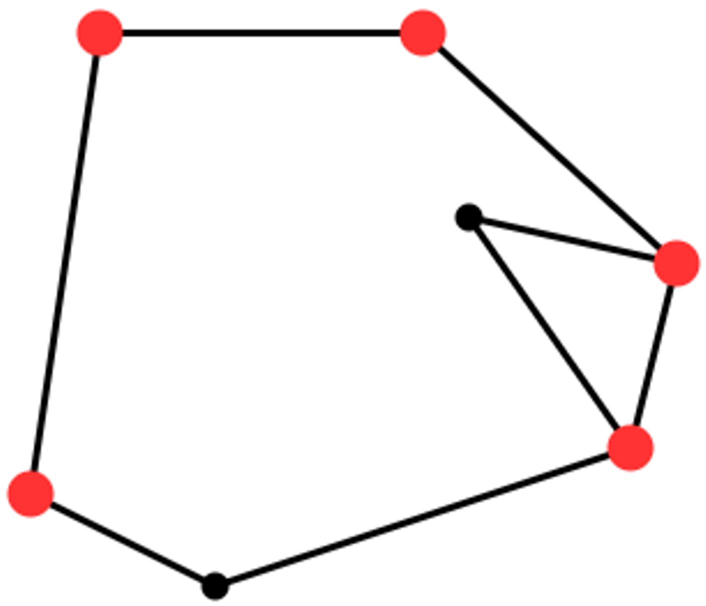
Step1　　　　　or　　　　　をみつける

サイズ k 以下の極大マッチングM

Step2　**Mに属す辺の各端点vのみ**に対して、

if vの次数 $> k$: then vを削除

else: vと接続する辺を全てみつける



補題:
Step2は$O(k^{3/2}\sqrt{n})$クエリでできる

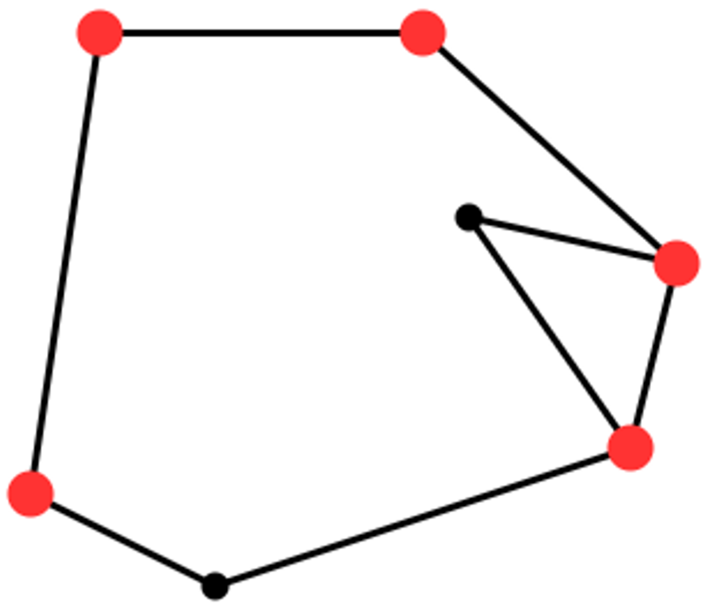
k-頂点被覆問題に対する量子クエリカーネル化

Step1　サイズ $k+1$ 以上のマッチング or **サイズ k 以下の極大マッチング M** をみつける ⇐ $O(\sqrt{kn})$クエリ

Step2　**Mに属す辺の各端点vのみ**に対して、 ⇐ $O(k^{3/2}\sqrt{n})$クエリ

if vの次数 $> k$: then vを削除

else: vと接続する辺を全てみつける



結論

- **頂点被覆問題**と**マッチング問題**に対して、
  パラメーター*k*が小さい時に最適な、**量子クエリ計算量**を導出

量子クエリカーネル化
☞**カーネル化**という古典アルゴリズムの考え方が、
量子クエリ計算量に役に立つ！

結論

- **頂点被覆問題**と**マッチング問題**に対して、
パラメーターkが小さい時に最適な、**量子クエリ計算量**を導出

量子クエリカーネル化
☞**カーネル化**という古典アルゴリズムの考え方が、
量子クエリ計算量に役に立つ！

Q. 本結果はkが大きい時に改善できるか？
Q. 量子クエリカーネル化は他の問題にも適用できるか?