Connections between discrete and continuous optimization

Euiwoong Lee
University of Michigan
“CS Theory”

• “Theoretical Computer Science” (TCS) / “Theory of Computing” (ToC)

• What do we do?
  • We prove theorems about computation
STOC/FOCS/SODA community

• Sometimes called “algorithms and complexity”
  • STOC (Symp. on Theory of Computing, since 1969)
  • FOCS (symp on Foundations of Computer Science, since 1960)
  • SODA (Symp. on Discrete Algorithms, since 1990)

• Other significant communities of TCS
  • Logic and theory of programming languages
  • Optimization/numerical analysis
  • Information/coding/control/systems performance theory
Influence from Mathematics

• Everything you claim should be proved.

• Authors are ordered alphabetically.

• Some papers are published in math journals

• Avi Wigderson and László Lovász won 2021 Abel Prize.
  • Avi’s PhD is from Computer Science!
A “typical paper” looks like...

• “Task/problem” that you want to do with computers
  • Finding shortest path, factoring integer, finding optimal parameters for neural nets
  • Any well-defined “function” specifying (valid input) => (desired output)

• “Model of computation”
  • (poly-time/randomized/non-deterministic) Turing machine
  • streaming/online/dynamic
  • distributed/parallel
  • quantum
  • crypto/communication
  • market/brain/evolution (computational lens)

• Can do it (there’s an algorithm) or cannot do it (there’s no algorithm)!
Some cool recent things

- Computing Maximum Flow of graph $G = (V, E)$
  - [Ford-Fulkerson 56] $O(|E| f)$ where $f$ is max flow --- not poly-time
  - [Dinitz 70, Edmonds-Karp 72] $O(|V|^2 |E|)$
  - ...
  - [CKLPPS 22] $O(|E|^{1.0001})$
    - Combination of interior point method (continuous) + graph theory (discrete)

- Traveling Salesperson Problem (TSP)
  - Given $G = (V, E)$, find shortest tour that visits every vertex at least once.
  - [Christofides 76] 1.5-approximation
  - [KKO 21] $(1.5 - 10^{-36})$-approximation!
    - Combination of graph cut representation (discrete) + real-stable polynomials (continuous).
<table>
<thead>
<tr>
<th>Abstraction</th>
<th>Algorithms</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objects</td>
<td><img src="image1.png" alt="Diagram" /></td>
<td><img src="image2.png" alt="Diagram" /></td>
</tr>
<tr>
<td>Discrete</td>
<td><img src="image3.png" alt="Diagram" /></td>
<td></td>
</tr>
<tr>
<td>Continuous</td>
<td><img src="image4.png" alt="Diagram" /></td>
<td></td>
</tr>
<tr>
<td>Plan</td>
<td>Abstraction</td>
<td>Algorithms</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
<td>------------</td>
</tr>
<tr>
<td></td>
<td>Objects</td>
<td>Discrete</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[Image]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[Image]</td>
</tr>
</tbody>
</table>
Example 1: Vertex Cover

- Minimization
- Input
  - (Undir.) Graph $G = (V, E)$
- Output
  - Subset $U$ of $V$ such that
    - $U$ intersects (covers) every edge!
- Objective Function
  - Cardinality of $U$
Example 1: Vertex Cover

- Choose an edge in $G$
- Add both endpoints to $U$.
- Delete these two vertices from $G$,
  - Including all edges incident on them
- Repeat until $G$ has no edge.
Example 1: Vertex Cover

- Choose an edge in $G$
- Add both endpoints to $U$.
- Delete these two vertices from $G$,
  - Including all edges incident on them
- Repeat until $G$ has no edge.
Example 1: Vertex Cover

- Choose an edge in G
- Add both endpoints to U.
- Delete these two vertices from G,
  - Including all edges incident on them
- Repeat until G has no edge.
Example 1: Vertex Cover

• Choose an edge in G
• Add both endpoints to U.
• Delete these two vertices from G,
  – Including all edges incident on them
• Repeat until G has no edge.
Example 1: Vertex Cover

- Choose an edge in $G$
- Add both endpoints to $U$.
- Delete these two vertices from $G$,
  - Including all edges incident on them
- Repeat until $G$ has no edge.
Example 1: Vertex Cover

- Let $p = \#$ of blue (chosen) edges
  - $|U| = 2p$
  - Every Vertex Cover has to contain at least one vertex from each blue edge!
  - All blue edges do not share a vertex
  - $OPT \geq p$.
  - Therefore, $|U| \leq 2OPT$. 
Example 2: Max Cut

- Maximization
- Input
  - Graph $G = (V, E)$
- Output
  - Coloring $V \rightarrow \{B, R\}$
- Objective Function
  - # of edges between Blue and Red
Example 2: Max Cut

• For each v, randomly (and independently) color B or R.

• Each edge is (B-R) with probability 0.5.
  – 0.5-approximation

• [GW94] Semidefinite programming

\[ \min_{0 \leq x \leq 1} \frac{1}{\pi} \arccos(1 - 2x) \approx 0.878 \]
Big Open Questions

• Can we do better?
  – 1.99-approximation for Vertex Cover
  – 0.879 for Max-Cut
Probably not!

<table>
<thead>
<tr>
<th></th>
<th>Vertex Cover</th>
<th>Max Cut</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithm</td>
<td>2</td>
<td>0.878 (GW 94)</td>
</tr>
<tr>
<td>NP-Hardness</td>
<td>1.36 (DS 05)</td>
<td>0.941(TSSW01)</td>
</tr>
<tr>
<td>UG-Hardness</td>
<td>2 (KR 08)</td>
<td>0.878(KKMO07)</td>
</tr>
</tbody>
</table>

- [Khot 02] Unique Games Conjecture (UGC).
- [KR08, KKMO07] Tight hardness for Vertex Cover and Max-Cut assuming UGC
• [Khot 02] Unique Games Conjecture (UGC).

• [KR08, KKMO07] Tight hardness for Vertex Cover and Max-Cut assuming UGC.
<table>
<thead>
<tr>
<th>Abstraction</th>
<th>Algorithms</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Objects</strong></td>
<td><strong>Vertex Cover</strong></td>
<td><strong>Discrete</strong></td>
</tr>
<tr>
<td><strong>Continuous</strong></td>
<td><strong>Max Cut</strong></td>
<td><strong>NP</strong></td>
</tr>
</tbody>
</table>

Plan

- Abstraction
- Objects
  - Discrete
  - Continuous
- Algorithms
- Complexity
  - Discrete
  - Continuous
<table>
<thead>
<tr>
<th>Abstraction</th>
<th>Algorithms</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objects</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Discrete</td>
<td>Vertex Cover</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Max Cut</td>
</tr>
<tr>
<td></td>
<td>Continuous</td>
<td></td>
</tr>
</tbody>
</table>

Plan
Small Set Expansion

- [RS 10] Small Set Expansion Hypothesis (SSEH)
  - SSEH $\Rightarrow$ UGC $\Rightarrow$ (optimal hardness for ...)
  - Don’t know whether SSEH $\Leftarrow$ UGC

- Will focus on SSEH..
• Let $G = (V, E)$ be a $d$-regular graph. ($n = |V|$)
• Given $S \subseteq V$,
  - $\Phi(S) := \frac{|E(S, V \setminus S)|}{d|S|}$
  - Note that $0 \leq \Phi(S) \leq 1$
  - $S$ is “expanding” when $\Phi(S) \approx 1$.
  - Fair to consider $S$ with $|S| \leq n/2$.
• $\Phi_G = \min_{|S| \leq n/2} \Phi(S)$
  - Expansion of “the least expanding set”
Spectral Graph Theory 101

Let $G = (V, E)$ be a $d$-regular graph. ($n = |V|$)

Given $S \subseteq V$,
- $\Phi(S) := \frac{|E(S, V \setminus S)|}{d|S|}$
- Note that $0 \leq \Phi(S) \leq 1$
- $S$ is “expanding” when $\Phi(S) \approx 1$.
- Fair to consider $S$ with $|S| \leq n/2$.

$\Phi_G = \min_{|S|\leq n/2} \Phi(S)$
- Expansion of “the least expanding set”
Spectral Graph Theory 101

- Let $G = (V, E)$ be a $d$-regular graph. ($n = |V|$)
- Given $S \subseteq V$,
  - $\Phi(S) := \frac{|E(S, V \setminus S)|}{d|S|}$
  - Note that $0 \leq \Phi(S) \leq 1$
  - $S$ is “expanding” when $\Phi(S) \approx 1$.
  - Fair to consider $S$ with $|S| \leq n/2$.
- $\Phi_G = \min_{|S| \leq n/2} \Phi(S)$
  - Expansion of “the least expanding set”
SGT 101

• “Expansion problem”
  – Given $G$ with $\Phi_G \leq \epsilon$,
  – Find $T$ s.t.
    • $|T| \leq n/2$
    • $\Phi(T) \leq \epsilon$
  – Want $\epsilon' \to 0$ as $\epsilon \to 0$.

• [Cheeger’s inequality] Can solve “Expansion problem” with $\epsilon' = \sqrt{\epsilon}$
Given $d$-regular $G$, consider normalized adjacency matrix $A$.
- $A_{i,j} = 1/d$ if $(i, j) \in E$.

Let $\lambda_1 \geq \cdots \geq \lambda_n$ eigenvalues of $A$
- $\lambda_1 = 1$ ($Ax = x$ when $x = (1,1,\ldots,1)$).

\[ \Phi(S) = \frac{|E(S, V \setminus S)|}{d|S|}, \Phi_G = \min_{|S| \leq n/2} \Phi(S) \]

[Cheeger] Given $G$ with $\Phi_G = \epsilon$, can find $S$ with $\Phi(S) \leq \sqrt{\epsilon}$
• Given $d$-regular $G$, consider normalized adjacency matrix $A$.
  $A_{i,j} = 1/d$ if $(i, j) \in E$.

• Let $\lambda_1 \geq \cdots \geq \lambda_n$ eigenvalues of $A$.
  $\lambda_1 = 1$ ($Ax = x$ when $x = (1,1, \ldots, 1)$).

[Cheeger] Given $G$ with $\Phi_G = \epsilon$, can find $S$ with $\Phi(S) \leq \sqrt{\epsilon}$. 

\[\Phi(S) = \frac{|E(S, V \setminus S)|}{d|S|}, \quad \Phi_G = \min_{|S| \leq n/2} \Phi(S)\] 

\[\Phi_G = \min \Phi(S) = \epsilon\]

\[
\begin{align*}
    a & b & c & d \\
    a & 0 & \frac{1}{2} & 0 & \frac{1}{2} \\
    b & \frac{1}{2} & 0 & \frac{1}{2} & 0 \\
    c & 0 & \frac{1}{2} & 0 & \frac{1}{2} \\
    d & \frac{1}{2} & 0 & \frac{1}{2} & 0 \\
\end{align*}
\]

\[
\begin{pmatrix}
    a & b & c & d \\
    1 & 1 & 1 & 1 \\
\end{pmatrix}
\]

$\lambda_1 = 1,$
• Given $d$-regular $G$, consider normalized adjacency matrix $A$.
  - $A_{i,j} = 1/d$ if $(i, j) \in E$.

• Let $\lambda_1 \geq \cdots \geq \lambda_n$ eigenvalues of $A$
  - $(Ax)_i = \frac{1}{d} \sum_{(i,j) \in E} x_j$
    • “Average $x$-value of nbrs”

\[
\Phi(S) = \frac{|E(S,V\setminus S)|}{|S|}, \quad \Phi_G = \min_{|S| \leq n/2} \Phi(S)
\]

[Cheeger] Given $G$ with $\Phi_G = \epsilon$, can find $S$ with $\Phi(S) \leq \sqrt{\epsilon}$
SGT 101 - Intuition

- \((Ax)_i = \frac{1}{d} \sum_{(i,j) \in E} x_j\)
  - "Average value of nbrs"

- If \(x\) is indicator vector of "expanding set" \(S\),
  - \(Ax\) will have not much values in \(S\)
  - \(x\) and \(Ax\) are "far"

\[
\Phi(S) = \frac{|E(S, V \setminus S)|}{d|S|}, \quad \Phi_G = \min_{|S| \leq n/2} \Phi(S)
\]

[Cheeger] Given \(G\) with \(\Phi_G = \epsilon\), can find \(S\) with \(\Phi(S) \leq \sqrt{\epsilon}\)
If $x$ is indicator vector of “non-expanding set” $S$,

- $Ax$ will have some values in $S$
- $x$ and $Ax$ are “close”
  - Large eigenvalue and its eigenvector!

[Cheeger] Converse of above
- If $x \approx Ax$ and far from $(1, ..., 1)$,
  - E.g., $\lambda_2 x = Ax$,
- $x$ is “close” to indicator vector of non-expanding small set $S$. 

\[
\Phi(S) = \frac{|E(S, V \setminus S)|}{d|S|}, \quad \Phi_G = \min_{|S| \leq n/2} \Phi(S)
\]

[Cheeger] Given $G$ with $\Phi_G = \epsilon$, can find $S$ with $\Phi(S) \leq \sqrt{\epsilon}$
Small Set Expansion

- $\Phi_{\delta,G} = \min_{|S| \leq \delta n} \Phi(S)$.
  - So $\Phi_G = \Phi_{1/2,G}$.

- Q] Is there Cheeger-type algorithm?
  - Given $\Phi_{\delta,G} = \epsilon$, find $T$ with
    - $|T| \leq \delta n, \Phi(T) \leq \epsilon'$.

- [RS10] (SSE Hypothesis)
  - $\forall \epsilon > 0, \exists \delta > 0$ s.t. given $G$ with
    $\Phi_{\delta,G} = \epsilon$,
  - NP-hard to find $T$ with $|T| \leq \delta n, \Phi(T) \leq 1 - \epsilon$
Small Set Expansion

- $\Phi_{\delta,G} = \min_{|S| \leq \delta n} \Phi(S)$.
  - So $\Phi_G = \Phi_{1/2,G}$.
- Q] Is there Cheeger-type algorithm?
  - Given $\Phi_{\delta,G} = \epsilon$, find $T$ with
    - $|T| \leq \delta n, \Phi(T) \leq \epsilon'$.
- [RS10] (SSE Hypothesis)
  - $\forall \epsilon > 0$, $\exists \delta > 0$ s.t. given $G$ with $\Phi_{\delta,G} = \epsilon$,
  - NP-hard to find $T$ with $|T| \leq \delta n$, $\Phi(T) \leq 1 - \epsilon$

[Cheeger] Given $G$ with $\Phi_G = \epsilon$, can find $S$ with $\Phi(S) \leq \sqrt{\epsilon}$
Small Set Expansion

• Try previous algorithm.
  – Recall $1 = \lambda_1 \geq \cdots \geq \lambda_n$ with eigenvectors $v_1, \ldots, v_n$
  – Take $v_2$ s.t. $Av_2 = \lambda_2 v_2$.

• If $v_2$ is **sparse indicator**
  – E.g., indicator vector of some set $|S| \leq \delta n$.
  – $S$ is a small non-expanding set!

• Even if $\text{span}(v_2, \ldots, v_k)$ contains a sparse indicator vector for small $k$, we are good.

\[
\Phi_{\delta,G} = \min_{|S| \leq \delta n} \Phi(S)
\]

[SSEH] Given $G$ with $\Phi_{G,\delta} = \epsilon$, cannot find $S$ with $\Phi(S) \leq 1 - \epsilon$. 
Finding “Sparse” Vector

• Now if want to find a “sparse” vector in linear space $span(v_2, \ldots, v_k)$.
Finding “Sparse” Vector

• Now if want to find a “sparse” vector in linear space $V \subseteq \mathbb{R}^n$.

• What is good notion of “sparsity”?  
  – # of nonzero entries is too susceptible to noise.
Finding “Sparse” Vector

• Now if want to find a “sparse” vector in linear space $V \subseteq \mathbb{R}^n$.

• What is good notion of “sparsity”?
  – # of nonzero entries is too susceptible to noise.
<table>
<thead>
<tr>
<th>Abstraction</th>
<th>Algorithms</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objects</td>
<td>Discrete</td>
<td>Continuous</td>
</tr>
<tr>
<td></td>
<td>Vertex Cover</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Max Cut</td>
<td>Unique Games</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Small-Set Expansion</td>
</tr>
</tbody>
</table>
## Plan

<table>
<thead>
<tr>
<th>Abstraction</th>
<th>Algorithms</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Objects</strong></td>
<td><strong>Discrete</strong></td>
<td><strong>Unique Games</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Vertex Cover</strong></td>
<td><strong>Small-Set Expansion</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Max Cut</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Continuous</strong></td>
<td></td>
<td><strong>Operator Norms</strong></td>
</tr>
</tbody>
</table>
Norms

• Given a vector $x = (x_1, \ldots, x_n)$ and $p \geq 1$,
  - $||x||_p := (\sum_{i=1}^{n}|x_i|^p)^{1/p}$.
  - $||x||_\infty := \max_i|x_i|$.

• Facts
  - $||x||_p \geq ||x||_q$ if $p \leq q$
  - For $q > p$, $||x||_q/||x||_p$ is maximized when $x$ has only one nonzero entry.
  - For $q < p$, $||x||_q/||x||_p$ is maximized when it is well-spread (i.e., $|x_1| = |x_2| = \cdots = |x_n|$)
Norms

• Given a vector $x = (x_1, \ldots, x_n)$ and $p \geq 1$,
  
  \begin{itemize}
  \item $||x||_p := (\sum_{i=1}^{n} |x_i|^p)^{1/p}$.
  \item $||x||_\infty := \max_i |x_i|$.
  \end{itemize}

• Facts
  
  \begin{itemize}
  \item $||x||_p \geq ||x||_q$ if $p \leq q$
  \item For $q > p$, $||x||_q/||x||_p$ is maximized when $x$ has only one nonzero entry.
  \item For $q < p$, $||x||_q/||x||_p$ is maximized when it is well-spread (i.e., $|x_1| = |x_2| = \cdots = |x_n|$)
  \end{itemize}

\[
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
\frac{||x||_1}{||x||_2} = 1, & \frac{||x||_4}{||x||_2} = 1 \\
\frac{1/\sqrt{5}}{1/\sqrt{5}} & \frac{1/\sqrt{5}}{1/\sqrt{5}} & \frac{1/\sqrt{5}}{1/\sqrt{5}} & \frac{1/\sqrt{5}}{1/\sqrt{5}} \\
\frac{||x||_1}{||x||_2} = \sqrt{5}, & \frac{||x||_4}{||x||_2} = 1/\sqrt{5} \\
\frac{||x||_1}{||x||_2} = 2.24, & \frac{||x||_4}{||x||_2} = \frac{1}{\sqrt{5}} = 0.66
\end{bmatrix}
\]
Small Set Expansion

• Now if want to find a “sparse” vector in linear space $V \subseteq \mathbb{R}^n$.

• What is good notion of “sparsity”?
  – # of nonzero entries is too susceptible to noise.
Small Set Expansion

• Now if want to find a “sparse” vector in linear space $V \subseteq \mathbb{R}^n$.

• What is good notion of “sparsity”?
  – # of nonzero entries is too susceptible to noise.
  – $||x||_q / ||x||_p$ is large when $q > p$.

$||v||_q / ||v||_p$ must be large!
Small Set Expansion

• Now if want to find a “sparse” vector in linear space $V \subseteq \mathbb{R}^n$.

• What is good notion of “sparsity”?  
  – # of nonzero entries is too susceptible to noise.  
  – $\frac{||x||_q}{||x||_p}$ is large when $q > p$.

• ($p = 2$) Let $A$ be matrix whose columns form orthonormal basis of $V$.  
  – $\forall x$: $||Ax||_2 = ||x||_2$.  
  – $Ax \in V$

• So, finding a “sparse vector” in $V$ became finding $x$ that maximizes  
  \[ \frac{||Ax||_q}{||Ax||_2} = \frac{||Ax||_q}{||x||_2} \]
Operator Norms

- For matrix $A$, and $p, q$,
  $$
  \|A\|_{p\to q} := \max_{x \neq 0} \frac{\|Ax\|_q}{\|x\|_p} = \max_{x: \|x\|_p=1} \|Ax\|_q
  $$
  - Can we (approximately) compute it?

- Connections to machine learning, quantum computing, etc.
  - When $q$ is even integer, $\|Ax\|_q^q$ is degree-$q$ polynomial in $x$!

- [BBHKSZ 12] When $p = 2$, $q > 2$, a good (constant) approx. algorithm to compute $\|A\|_{p\to q}$ solves Small Set Expansion.
Approximating $||A||_{p \to q}$

- $p = q = 2$: spectral norm.
- $p \geq q$: well-understood.
  - $p \geq 2 \geq q$: $K_G$-approx. ($1.67 \leq K_G \leq 1.79$)
  - Otherwise: No $c$-approx. is possible for any $c > 1$.
- [BGGLT 19] If $2 < p < q$ or $p < q < 2$,
  - No $c$-approx. is possible for any $c > 1$.
  - First NP-hardness for $p < q$
  - Don’t cover $p = 2, q > 2$. 
Approximating $\|A\|_{p \to q}$

- $p = q = 2$: spectral norm.
- $p \geq q$: well-understood.
  - $p \geq 2 \geq q$: $K_G$-approx. ($1.67 \leq K_G \leq 1.79$)
  - Otherwise: No $c$-approx. is possible for any $c > 1$.
- [BGGLT 19] If $2 < p < q$ or $p < q < 2$,
  - No $c$-approx. is possible for any $c > 1$.
  - First NP-hardness for $p < q$
  - Don’t cover $p = 2, q > 2$. 
One-line intuitions for proofs

• $p \geq q$ : Closer to “discrete problems”!
  – When $p = \infty$, $||x||_\infty \leq 1$ means $x_i \in [-1, 1]$ for every $i$
    • WLOG, can even assume $x_i \in \{-1, 1\}$
  – Tools from discrete problems work when $p \geq q$.

• [BGGLT 19] Hardness for $2 < p < q$
  – It is hard to find $x$ with $||x||_p = 1$ with large $||Ax||_2$.
  – Dvoretzky’s theorem: If $B$ is a “random matrix”, then $||By||_q \approx ||y||_2$
  – It is hard to find $x$ with $||x||_p = 1$ with large $||B Ax||_q$!
<table>
<thead>
<tr>
<th>Plan</th>
<th>Abstraction</th>
<th>Algorithms</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objects</td>
<td>Discrete</td>
<td>Vertex Cover</td>
<td>Unique Games</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Max Cut</td>
<td>Small-Set Expansion</td>
</tr>
<tr>
<td></td>
<td>Continuous</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Operator Norms</td>
</tr>
<tr>
<td>Plan</td>
<td>Abstraction</td>
<td>Algorithms</td>
<td>Complexity</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td></td>
<td>Objects</td>
<td>Vertex Cover</td>
<td>Unique Games</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Max Cut</td>
<td>Small-Set Expansion</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Discrete</td>
<td>Quadratic Maximization</td>
<td>Operator Norms</td>
</tr>
<tr>
<td></td>
<td>Continuous</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Quadratic optimization over general norm

- $\|A\|_{p \rightarrow 2}$: Easy when $p \geq 2$ ($K_G$-approx. $1.67 \leq K_G \leq 1.79$) but (believed to be) hard when $p < 2$.
- Can we generalize “$p$”? 
  • Let $B \subseteq \mathbb{R}^n$ be a symmetric ($B = -B$) convex set.
    • Defines a general “norm”: $\|x\|_B = (\text{smallest } t > 0 \text{ s.t. } x/t \in B)$
    • Example: Unit $\ell_p$ ball $\{x: \|x\|_p \leq 1\}$.
  • Input: $m \times n$-matrix $A$.
  • Output: $x \in B$ that maximizes $\|Ax\|_2$.
  • Which $B$ allows $O(1)$-approximation?
Quadratic optimization over general norm

- Let $B \subseteq \mathbb{R}^n$ be a symmetric ($B = -B$) convex set.
- Input: $m \times n$-matrix $A$.
- Output: $x \in B$ that maximizes $||Ax||_2$.
- Which $B$ allows $O(1)$-approximation?
- Answer [BLN 21]: When $B$ is “fatter” than $\ell_2$ ball!
  - Using notion of “type” from functional analysis
  - Can handle more general norms (e.g., when $x$ is matrix)!

Tools from discrete optimization

Reduction from $\ell_p$ with $p<2$
## Plan

<table>
<thead>
<tr>
<th>Abstraction</th>
<th>Algorithms</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objects</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Discrete</td>
<td>Vertex Cover</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Max Cut</td>
</tr>
<tr>
<td></td>
<td>Continuous</td>
<td>Quadratic Maximization</td>
</tr>
</tbody>
</table>

- **Discrete**
  - Vertex Cover
  - Max Cut

- **Continuous**
  - Quadratic Maximization

- **Unique Games**
  - Small-Set Expansion
  - Operator Norms
Conclusion

• Continuous and Discrete optimization
  – This talk: through operator norms
  – Other connections (e.g., Max-flow, TSP)

• (Approximately) finding global optimum for non-convex functions
  – Many viewpoints: Statistics, Optimization, Machine learning, Physics, Pure math, CS, etc.
  – Important to build “bridges”.
    • Recently started to talk to each other.
  – Exciting time to study!
<table>
<thead>
<tr>
<th>Abstraction</th>
<th>Algorithms</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objects</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Discrete</td>
<td>Vertex Cover</td>
<td>Unique Games</td>
</tr>
<tr>
<td></td>
<td>Max Cut</td>
<td>Small-Set</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Expansion</td>
</tr>
<tr>
<td>Continuous</td>
<td>Quadratic Maximization</td>
<td>Operator</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Norms</td>
</tr>
</tbody>
</table>