Testing Consumer Rationality using Perfect Graphs and Oriented Discs

Shant Boodaghians and Adrian Vetta

McGill University

WINE 2015

Commodity Market

Consider an agent in an n-commodity market:

- Each commodity has price/unit p_i
- Market price vector $\mathbf{p} \in \mathbf{R}^n_+$
- Agent requests bundle $\mathbf{x}^* \in \mathbf{R}^n_+$

Multiple observations at multiple price-points

$$\text{Data-set} = \{(\mathbf{p}_1, \mathbf{x}_1), (\mathbf{p}_2, \mathbf{x}_2), \ldots, (\mathbf{p}_m, \mathbf{x}_m)\}$$

At prices $\mathbf{p}_i$, agent demands bundle $\mathbf{x}_i$, for $i = 1$ to m.

Commodity Market

Consider an agent in an n-commodity market:

- Each commodity has price/unit p_i
- Market price vector $\mathbf{p} \in \mathbf{R}_+^n$
- Agent requests bundle $\mathbf{x}^* \in \mathbf{R}_+^n$

Multiple observations at multiple price-points

$$\text{Data-set} = \{(\mathbf{p}_1, \mathbf{x}_1), (\mathbf{p}_2, \mathbf{x}_2), \ldots, (\mathbf{p}_m, \mathbf{x}_m)\}$$

At prices $\mathbf{p}_i$, agent demands bundle $\mathbf{x}_i$, for $i = 1$ to m.

Revealed Preference

Consider two pairs, $(\mathbf{p}_i, \mathbf{x}_i)$ and $(\mathbf{p}_j, \mathbf{x}_j)$

- If $\mathbf{p}_j \cdot \mathbf{x}_i \leq \mathbf{p}_j \cdot \mathbf{x}_j$,
- then $\mathbf{x}_i$ was more affordable than $\mathbf{x}_j$ at prices $\mathbf{p}_j$. $\Rightarrow \quad \mathbf{x}_j \succeq \mathbf{x}_i$

Demanded bundle is **revealed preferred** to every more-affordable bundle.

Note. Geometrically, $\mathbf{x}_j$ revealed preferred to every point below the hyperplane due to $\mathbf{p}_j$

Revealed Preference

Consider two pairs, $(\mathbf{p}_i, \mathbf{x}_i)$ and $(\mathbf{p}_j, \mathbf{x}_j)$

- If $\mathbf{p}_j \cdot \mathbf{x}_i \leq \mathbf{p}_j \cdot \mathbf{x}_j$,
- then $\mathbf{x}_i$ was more affordable than $\mathbf{x}_j$ at prices $\mathbf{p}_j$. $\Rightarrow \quad \mathbf{x}_j \succeq \mathbf{x}_i$

Demanded bundle is **revealed preferred** to every more-affordable bundle.

Note. Geometrically, $\mathbf{x}_j$ revealed preferred to every point below the hyperplane due to $\mathbf{p}_j$

Revealed Preference

Consider two pairs, $(\mathbf{p}_i, \mathbf{x}_i)$ and $(\mathbf{p}_j, \mathbf{x}_j)$

- If $\mathbf{p}_j \cdot \mathbf{x}_i \leq \mathbf{p}_j \cdot \mathbf{x}_j$,
- then $\mathbf{x}_i$ was more affordable than $\mathbf{x}_j$ at prices $\mathbf{p}_j$. $\Rightarrow \quad \mathbf{x}_j \succeq \mathbf{x}_i$

Demanded bundle is **revealed preferred** to every more-affordable bundle.

Note. Geometrically, $\mathbf{x}_j$ revealed preferred to every point below the hyperplane due to $\mathbf{p}_j$

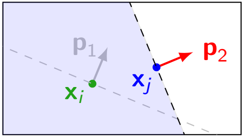
Revealed Preference

Consider two pairs, $(\mathbf{p}_i, \mathbf{x}_i)$ and $(\mathbf{p}_j, \mathbf{x}_j)$

- If $\mathbf{p}_j \cdot \mathbf{x}_i \leq \mathbf{p}_j \cdot \mathbf{x}_j$,
- then $\mathbf{x}_i$ was more affordable than $\mathbf{x}_j$ at prices $\mathbf{p}_j$. $\Rightarrow \quad \mathbf{x}_j \succeq \mathbf{x}_i$

Demanded bundle is **revealed preferred** to every more-affordable bundle.

Note. Geometrically, $\mathbf{x}_j$ revealed preferred to every point below the hyperplane due to $\mathbf{p}_j$



Possible Configurations

The possible geometric relations between two items fall in 3 categories:

No relation

One preferred

Mutual Preference

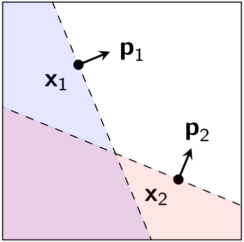
Can we determine if preference data is consistent?

- Clearly a 2-cycle ($\mathbf{a} \succ \mathbf{b} \succ \mathbf{a}$) is invalid. .... Non truthful
- 3-cycle also invalid ($\mathbf{a} \succ \mathbf{b} \succ \mathbf{c} \succ \mathbf{a}$) .... Non truthful

Possible Configurations

The possible geometric relations between two items fall in 3 categories:

No relation



One preferred

Mutual Preference

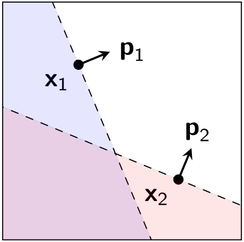
Can we determine if preference data is consistent?

- Clearly a 2-cycle ($\mathbf{a} \succ \mathbf{b} \succ \mathbf{a}$) is invalid. .... Non truthful
- 3-cycle also invalid ($\mathbf{a} \succ \mathbf{b} \succ \mathbf{c} \succ \mathbf{a}$) .... Non truthful

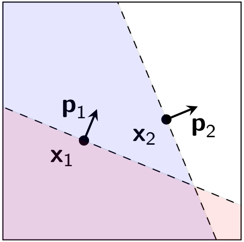
Possible Configurations

The possible geometric relations between two items fall in 3 categories:

No relation



One preferred



Mutual Preference

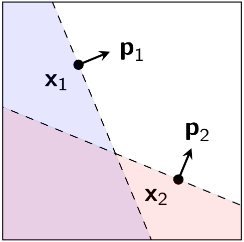
Can we determine if preference data is consistent?

- Clearly a 2-cycle ($\mathbf{a} \succ \mathbf{b} \succ \mathbf{a}$) is invalid. .... Non truthful
- 3-cycle also invalid ($\mathbf{a} \succ \mathbf{b} \succ \mathbf{c} \succ \mathbf{a}$) .... Non truthful

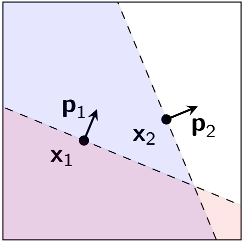
Possible Configurations

The possible geometric relations between two items fall in 3 categories:

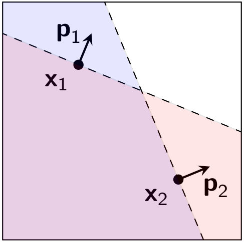
No relation



One preferred



Mutual Preference

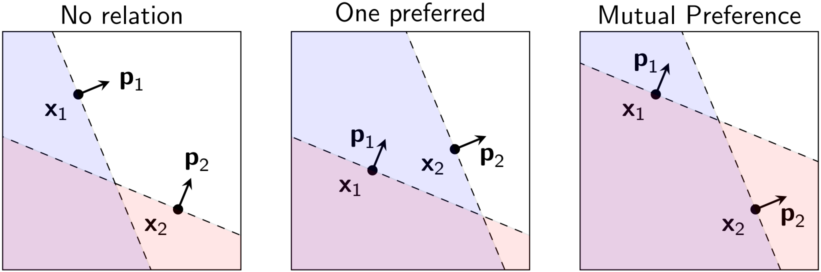


Can we determine if preference data is consistent?

- Clearly a 2-cycle ($\mathbf{a} \succ \mathbf{b} \succ \mathbf{a}$) is invalid. .... Non truthful
- 3-cycle also invalid ($\mathbf{a} \succ \mathbf{b} \succ \mathbf{c} \succ \mathbf{a}$) .... Non truthful

Possible Configurations

The possible geometric relations between two items fall in 3 categories:

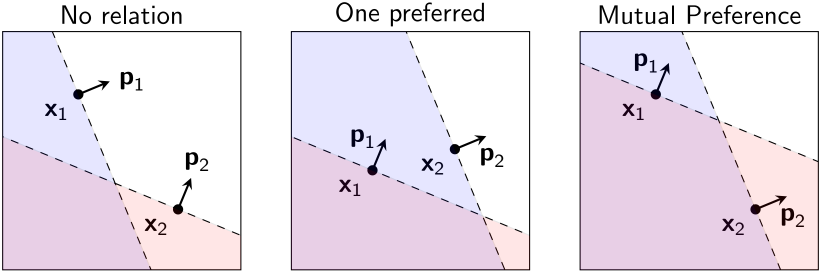


Can we determine if preference data is consistent?

- Clearly a 2-cycle ($\mathbf{a} \succ \mathbf{b} \succ \mathbf{a}$) is invalid. .... Non truthful
- 3-cycle also invalid ($\mathbf{a} \succ \mathbf{b} \succ \mathbf{c} \succ \mathbf{a}$) .... Non truthful

Possible Configurations

The possible geometric relations between two items fall in 3 categories:



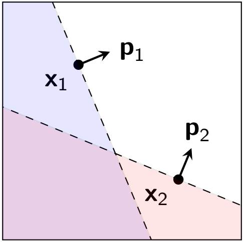
Can we determine if preference data is consistent?

- Clearly a 2-cycle ($\mathbf{a} \succ \mathbf{b} \succ \mathbf{a}$) is invalid. .... Non truthful
- 3-cycle also invalid ($\mathbf{a} \succ \mathbf{b} \succ \mathbf{c} \succ \mathbf{a}$) .... Non truthful

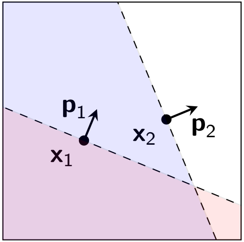
Possible Configurations

The possible geometric relations between two items fall in 3 categories:

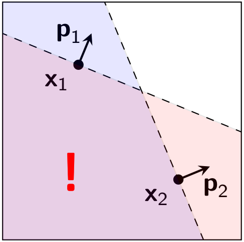
No relation



One preferred



Mutual Preference

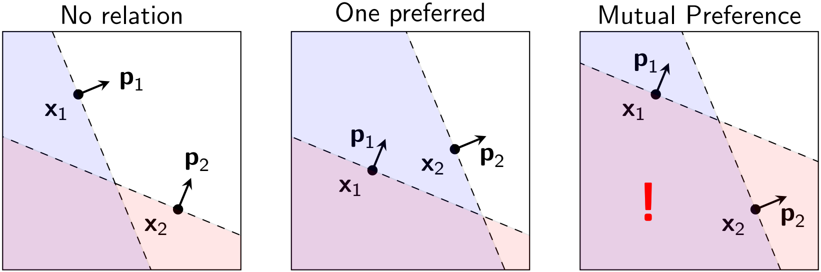


Can we determine if preference data is consistent?

- Clearly a 2-cycle ($\mathbf{a} \succ \mathbf{b} \succ \mathbf{a}$) is invalid. .... Non truthful
- 3-cycle also invalid ($\mathbf{a} \succ \mathbf{b} \succ \mathbf{c} \succ \mathbf{a}$) .... Non truthful

Possible Configurations

The possible geometric relations between two items fall in 3 categories:



Can we determine if preference data is consistent?

- Clearly a 2-cycle ($\mathbf{a} \succ \mathbf{b} \succ \mathbf{a}$) is invalid. . . . . Non truthful
- 3-cycle also invalid ($\mathbf{a} \succ \mathbf{b} \succ \mathbf{c} \succ \mathbf{a}$) . . . . Non truthful

Preference Graph

We can build a **Preference (di-)Graph** to check for cycles

- Vertex set is data: $(\mathbf{p}_1, \mathbf{x}_1), (\mathbf{p}_2, \mathbf{x}_2), \ldots, (\mathbf{p}_m, \mathbf{x}_m)$
- Arc from $(\mathbf{p}_i, \mathbf{x}_i)$ to $(\mathbf{p}_j, \mathbf{x}_j)$ if $\mathbf{x}_i \succeq_{\mathbf{p}_i} \mathbf{x}_j$

Preference Graph

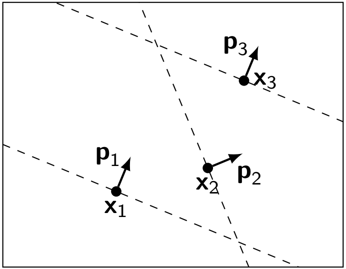
We can build a **Preference (di-)Graph** to check for cycles

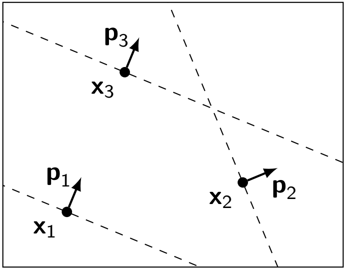
- Vertex set is data: $(\mathbf{p}_1, \mathbf{x}_1)$, $(\mathbf{p}_2, \mathbf{x}_2), \ldots,$ $(\mathbf{p}_m, \mathbf{x}_m)$
- Arc from $(\mathbf{p}_i, \mathbf{x}_i)$ to $(\mathbf{p}_j, \mathbf{x}_j)$ if $\mathbf{x}_i \succeq_{\mathbf{p}_i} \mathbf{x}_j$

Preference Graph

We can build a **Preference (di-)Graph** to check for cycles

- Vertex set is data: $(\mathbf{p}_1, \mathbf{x}_1)$, $(\mathbf{p}_2, \mathbf{x}_2)$, $\ldots$, $(\mathbf{p}_m, \mathbf{x}_m)$
- Arc from $(\mathbf{p}_i, \mathbf{x}_i)$ to $(\mathbf{p}_j, \mathbf{x}_j)$ if $\mathbf{x}_i \succeq_{\mathbf{p}_i} \mathbf{x}_j$

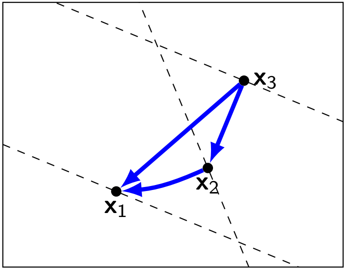


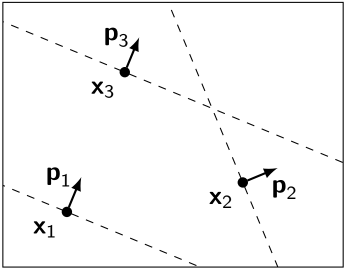


Preference Graph

We can build a **Preference (di-)Graph** to check for cycles

- Vertex set is data: $(\mathbf{p}_1, \mathbf{x}_1)$, $(\mathbf{p}_2, \mathbf{x}_2), \ldots,$ $(\mathbf{p}_m, \mathbf{x}_m)$
- Arc from $(\mathbf{p}_i, \mathbf{x}_i)$ to $(\mathbf{p}_j, \mathbf{x}_j)$ if $\mathbf{x}_i \succeq_{\mathbf{p}_i} \mathbf{x}_j$

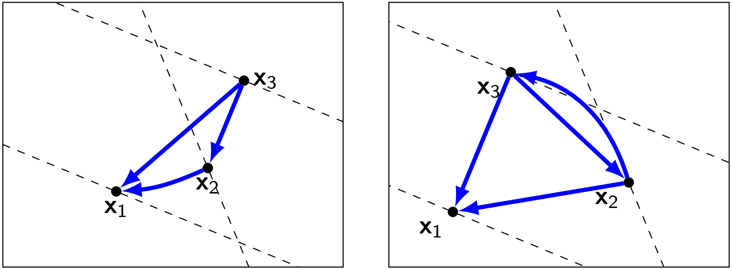




Preference Graph

We can build a **Preference (di-)Graph** to check for cycles

- Vertex set is data: $(\mathbf{p}_1, \mathbf{x}_1)$, $(\mathbf{p}_2, \mathbf{x}_2), \ldots,$ $(\mathbf{p}_m, \mathbf{x}_m)$
- Arc from $(\mathbf{p}_i, \mathbf{x}_i)$ to $(\mathbf{p}_j, \mathbf{x}_j)$ if $\mathbf{x}_i \succeq_{\mathbf{p}_i} \mathbf{x}_j$



Degree of Rationality

[Afriat, '67] The agent is rational $\iff$ preference graph is acyclic.

We ask: *how* rational is agent?

Definition. An agent's *degree of rationality* is the least number of data pairs to ignore for the agent to seem rational. Rational *iff* DoR $= 0$.

Note 1. This was defined by Houtman and Maks in '88.

Note 2. Equivalent to *Min. Feedback Vertex Set* on the Pref. Graph.

Example. In the following preference graph, the degree of rationality is 1.

Degree of Rationality

[Afriat, '67] The agent is rational $\iff$ preference graph is acyclic.

We ask: *how* rational is agent?

Definition. An agent's *degree of rationality* is the least number of data pairs to ignore for the agent to seem rational. Rational *iff* DoR = 0.

Note 1. This was defined by Houtman and Maks in '88.

Note 2. Equivalent to *Min. Feedback Vertex Set* on the Pref. Graph.

Example. In the following preference graph, the degree of rationality is 1.

Degree of Rationality

[Afriat, '67] The agent is rational $\iff$ preference graph is acyclic.

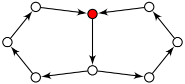
We ask: *how* rational is agent?

Definition. An agent's *degree of rationality* is the least number of data pairs to ignore for the agent to seem rational. Rational *iff* DoR $= 0$.

Note 1. This was defined by Houtman and Maks in '88.

Note 2. Equivalent to *Min. Feedback Vertex Set* on the Pref. Graph.

Example. In the following preference graph, the degree of rationality is 1.



Degree of Rationality, Complexity

What is the complexity of MFVS in preference graphs?

- If market size (n) large, then any graph is a preference graph

$$p^i_j = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \qquad x^i_j = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } (i, j) \in D \\ 2 & \text{if } (i, j) \notin D \end{cases}$$

Our Results

- **Theorem 1.** If $n = 2$, *Degree of Rationality* is in P
- **Theorem 2.** If $n \geq 3$, *DoR* decision problem is NP-complete

Degree of Rationality, Complexity

What is the complexity of MFVS in preference graphs?

- If market size (n) large, then any graph is a preference graph

$$p^i_j = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \qquad x^i_j = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } (i,j) \in D \\ 2 & \text{if } (i,j) \notin D \end{cases}$$

Our Results

- **Theorem 1.** If $n = 2$, *Degree of Rationality* is in P.
- **Theorem 2.** If $n \geq 3$, *DoR* decision problem is NP-complete.

Degree of Rationality, Complexity

What is the complexity of MFVS in preference graphs?

- If market size (n) large, then any graph is a preference graph

$$p_j^i = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \qquad x_j^i = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } (i,j) \in D \\ 2 & \text{if } (i,j) \notin D \end{cases}$$

Our Results

- **Theorem 1.** If $n = 2$, *Degree of Rationality* is in P.
- **Theorem 2.** If $n \geq 3$, *DoR* decision problem is NP-complete.

Degree of Rationality, Complexity

What is the complexity of MFVS in preference graphs?

- If market size (n) large, then any graph is a preference graph
  So complexity and approximation complexity same as that of MFVS.
- What if n small? Set of attainable graphs is strictly smaller.
  Approximation complexity likely improves.

Our Results

- **Theorem 1.** If $n = 2$, *Degree of Rationality* is in P.
- **Theorem 2.** If $n \geq 3$, *DoR* decision problem is NP-complete.

Degree of Rationality, Complexity

What is the complexity of MFVS in preference graphs?

- If market size (n) large, then any graph is a preference graph
  So complexity and approximation complexity same as that of MFVS.
- What if n small? Set of attainable graphs is strictly smaller.
  Approximation complexity likely improves.

Our Results

- **Theorem 1.** If $n = 2$, *Degree of Rationality* is in P.
- **Theorem 2.** If $n \geq 3$, *DoR* decision problem is NP-complete.

Degree of Rationality, Complexity

What is the complexity of MFVS in preference graphs?

- If market size (n) large, then any graph is a preference graph
  So complexity and approximation complexity same as that of MFVS.
- What if n small? Set of attainable graphs is strictly smaller.
  Approximation complexity likely improves.

Our Results

- **Theorem 1.** If $n = 2$, *Degree of Rationality* is in P.
- **Theorem 2.** If $n \geq 3$, *DoR* decision problem is NP-complete.

Degree of Rationality, Complexity

What is the complexity of MFVS in preference graphs?

- If market size (n) large, then any graph is a preference graph
  So complexity and approximation complexity same as that of MFVS.
- What if n small? Set of attainable graphs is strictly smaller.
  Approximation complexity likely improves.

Our Results

- **Theorem 1.** If $n = 2$, *Degree of Rationality* is in P.
- **Theorem 2.** If $n \geq 3$, *DoR* decision problem is NP-complete.

2-Commodity Markets

Theorem 1. *Degree of rationality* is poly-time for $n = 2$.

Proof Sketch.

1. **[Rose, '58]**: If $n = 2$, every cycle contains a digon (2-cycle)
2. **Corollary.** MFVS is Min. Vertex Cover on the set of digons
3. **Lemma.** Digon graphs of pref. graphs are perfect for $n = 2$. (New class of perfect graphs?)
4. **[GLS, '84]**: Min Vertex Cover on a Perfect Graph is polytime.

So **Theorem 1.** follows as a corollary. □

2-Commodity Markets

Theorem 1. *Degree of rationality* is poly-time for $n = 2$.

Proof Sketch.

1. **[Rose, '58]**: If $n = 2$, every cycle contains a digon (2-cycle)
2. **Corollary.** MFVS is Min. Vertex Cover on the set of digons
3. **Lemma.** Digon graphs of pref. graphs are perfect for $n = 2$. (New class of perfect graphs?)
4. **[GLS, '84]**: Min Vertex Cover on a Perfect Graph is polytime.

So **Theorem 1.** follows as a corollary. □

2-Commodity Markets

Theorem 1. *Degree of rationality* is poly-time for $n = 2$.

Proof Sketch.

1. **[Rose, '58]**: If $n = 2$, every cycle contains a digon (2-cycle)
2. **Corollary.** MFVS is Min. Vertex Cover on the set of digons
3. **Lemma.** Digon graphs of pref. graphs are perfect for $n = 2$. (New class of perfect graphs?)
4. **[GLS, '84]**: Min Vertex Cover on a Perfect Graph is polytime.

So **Theorem 1.** follows as a corollary. □

2-Commodity Markets

Theorem 1. *Degree of rationality* is poly-time for $n = 2$.

Proof Sketch.

1. **[Rose, '58]**: If $n = 2$, every cycle contains a digon (2-cycle)
2. **Corollary.** MFVS is Min. Vertex Cover on the set of digons
3. **Lemma.** Digon graphs of pref. graphs are perfect for $n = 2$. (New class of perfect graphs?)
4. **[GLS, '84]**: Min Vertex Cover on a Perfect Graph is polytime.

So **Theorem 1.** follows as a corollary. □

2-Commodity Markets

Theorem 1. *Degree of rationality* is poly-time for $n = 2$.

Proof Sketch.

1. **[Rose, '58]**: If $n = 2$, every cycle contains a digon (2-cycle)
2. **Corollary.** MFVS is Min. Vertex Cover on the set of digons
3. **Lemma.** Digon graphs of pref. graphs are perfect for $n = 2$. (New class of perfect graphs?)
4. **[GLS, '84]**: Min Vertex Cover on a Perfect Graph is polytime.

So **Theorem 1.** follows as a corollary. □

2-Commodity Markets

Theorem 1. *Degree of rationality* is poly-time for $n = 2$.

Proof Sketch.

1. **[Rose, '58]**: If $n = 2$, every cycle contains a digon (2-cycle)
2. **Corollary.** MFVS is Min. Vertex Cover on the set of digons
3. **Lemma.** Digon graphs of pref. graphs are perfect for $n = 2$. (New class of perfect graphs?)
4. **[GLS, '84]**: Min Vertex Cover on a Perfect Graph is polytime.

So **Theorem 1.** follows as a corollary. □

2-Commodity Markets

Theorem 1. *Degree of rationality* is poly-time for $n = 2$.

Proof Sketch.

1. **[Rose, '58]**: If $n = 2$, every cycle contains a digon (2-cycle)
2. **Corollary.** MFVS is Min. Vertex Cover on the set of digons
3. **Lemma.** Digon graphs of pref. graphs are perfect for $n = 2$.
   (New class of perfect graphs?)
4. **[GLS, '84]**: Min Vertex Cover on a Perfect Graph is polytime.

So **Theorem 1.** follows as a corollary. □

The digon graph

[Rose, '58]: If $n = 2$, every cycle contains the vertex set of a digon.

Therefore, hitting every digon hits every cycle.

Definition. The *digon graph* of a directed graph $D = (V, A)$ is $G = (V, E)$, where E is exactly the set of digons in D.

Example.

Corollary. *Degree of Rationality reduces to Min. Vertex Cover on* G.

The digon graph

[Rose, ‘58]: If $n = 2$, every cycle contains the vertex set of a digon.

Therefore, hitting every digon hits every cycle.

Definition. The *digon graph* of a directed graph $D = (V, A)$ is $G = (V, E)$, where E is exactly the set of digons in D.

Example.

Corollary. *Degree of Rationality* reduces to Min. Vertex Cover on G.

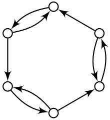
The digon graph

[Rose, '58]: If $n = 2$, every cycle contains the vertex set of a digon.

Therefore, hitting every digon hits every cycle.

Definition. The *digon graph* of a directed graph $D = (V, A)$ is $G = (V, E)$, where E is exactly the set of digons in D.

Example.



Corollary. *Degree of Rationality* reduces to Min. Vertex Cover on G.

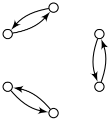
The digon graph

[Rose, '58]: If $n = 2$, every cycle contains the vertex set of a digon.

Therefore, hitting every digon hits every cycle.

Definition. The *digon graph* of a directed graph $D = (V, A)$ is $G = (V, E)$, where E is exactly the set of digons in D.

Example.



Corollary. *Degree of Rationality* reduces to Min. Vertex Cover on G.

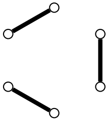
The digon graph

[Rose, '58]: If $n = 2$, every cycle contains the vertex set of a digon.

Therefore, hitting every digon hits every cycle.

Definition. The *digon graph* of a directed graph $D = (V, A)$ is $G = (V, E)$, where E is exactly the set of digons in D.

Example.



Corollary. *Degree of Rationality* reduces to Min. Vertex Cover on G.

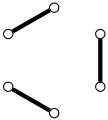
The digon graph

[Rose, '58]: If $n = 2$, every cycle contains the vertex set of a digon.

Therefore, hitting every digon hits every cycle.

Definition. The *digon graph* of a directed graph $D = (V, A)$ is $G = (V, E)$, where E is exactly the set of digons in D.

Example.



Corollary. *Degree of Rationality* reduces to Min. Vertex Cover on G.

Examples of Digon Graphs

Which graphs can be constructed as digon graphs? Some examples

Complete Graphs,

Complete Bip. Graphs,

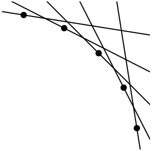
Complete r-part. Graphs,

- Complements of complete r-partite graphs,
- Many small graphs,
- etc.

Examples of Digon Graphs

Which graphs can be constructed as digon graphs? Some examples

Complete Graphs,



Complete Bip. Graphs,

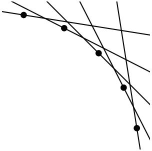
Complete r-part. Graphs,

- Complements of complete r-partite graphs,
- Many small graphs,
- etc.

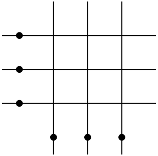
Examples of Digon Graphs

Which graphs can be constructed as digon graphs? Some examples

Complete Graphs,



Complete Bip. Graphs,



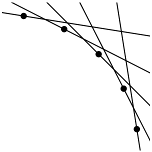
Complete r-part. Graphs,

- Complements of complete r-partite graphs,
- Many small graphs,
- etc.

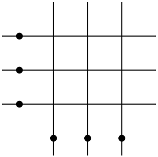
Examples of Digon Graphs

Which graphs can be constructed as digon graphs? Some examples

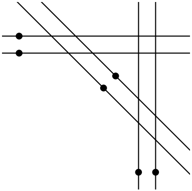
Complete Graphs,



Complete Bip. Graphs,



Complete r-part. Graphs,

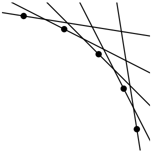


- Complements of complete r-partite graphs,
- Many small graphs,
- etc.

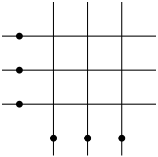
Examples of Digon Graphs

Which graphs can be constructed as digon graphs? Some examples

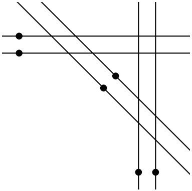
Complete Graphs,



Complete Bip. Graphs,



Complete r-part. Graphs,



- Complements of complete r-partite graphs,
- Many small graphs,
- etc.

Digon Graphs of Pref. Graphs are Perfect when $n = 2$

Sub-Lemma. If $\mathbf{x} \lessgtr \mathbf{y}$, $\mathbf{x} \lessgtr \mathbf{z}$ and $\mathbf{y} \not\lessgtr \mathbf{z}$, then either

Proof: Suppose not,

□

Digon Graphs of Pref. Graphs are Perfect when $n = 2$

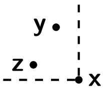
Sub-Lemma. If $\mathbf{x} \lesseqgtr \mathbf{y}$, $\mathbf{x} \lesseqgtr \mathbf{z}$ and $\mathbf{y} \not\lesseqgtr \mathbf{z}$, then either

Proof: Suppose not,

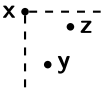
□

Digon Graphs of Pref. Graphs are Perfect when $n = 2$

Sub-Lemma. If $\mathbf{x} \lesseqgtr \mathbf{y}$, $\mathbf{x} \lesseqgtr \mathbf{z}$ and $\mathbf{y} \not\lesseqgtr \mathbf{z}$, then either



or



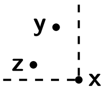
not

Proof: Suppose not,

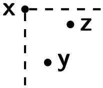
□

Digon Graphs of Pref. Graphs are Perfect when $n = 2$

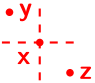
Sub-Lemma. If $\mathbf{x} \lessgtr \mathbf{y}$, $\mathbf{x} \lessgtr \mathbf{z}$ and $\mathbf{y} \not\lessgtr \mathbf{z}$, then either



or



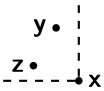
not



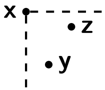
Proof: Suppose not,

Digon Graphs of Pref. Graphs are Perfect when $n = 2$

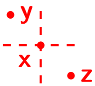
Sub-Lemma. If $\mathbf{x} \lessgtr \mathbf{y}$, $\mathbf{x} \lessgtr \mathbf{z}$ and $\mathbf{y} \not\lessgtr \mathbf{z}$, then either



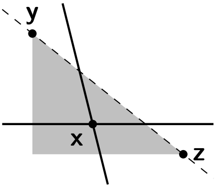
or

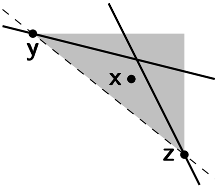


not



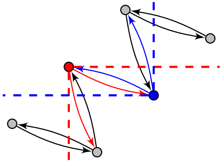
Proof: Suppose not,





□

Corollary. Long paths "alternate" in the digon graph



Corollary.

1. The digon graph contains no **odd** holes on ≥ 5 vertices.
2. The digon graph contains no antiholes on ≥ 5 vertices.

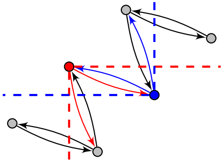
[CRST, '06] Strong Perfect Graph Theorem

A graph is perfect $\iff$ contains no odd holes and no odd antiholes.

Corollary. Digon Graphs of Pref. Graphs are perfect when $n = 2$.

"So **Theorem 1.** follows as a corollary." □

Corollary. Long paths "alternate" in the digon graph



Corollary.

1. The digon graph contains no **odd** holes on ≥ 5 vertices.
2. The digon graph contains no antiholes on ≥ 5 vertices.

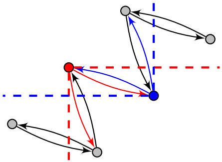
[CRST, '06] Strong Perfect Graph Theorem

A graph is perfect $\iff$ contains no odd holes and no odd antiholes.

Corollary. Digon Graphs of Pref. Graphs are perfect when $n = 2$.

"So **Theorem 1.** follows as a corollary." □

Corollary. Long paths "alternate" in the digon graph



Corollary.

1. The digon graph contains no **odd** holes on ≥ 5 vertices.
2. The digon graph contains no antiholes on ≥ 5 vertices.

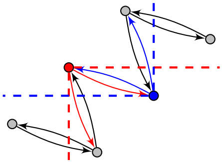
[CRST, '06] Strong Perfect Graph Theorem

A graph is perfect $\iff$ contains no odd holes and no odd antiholes.

Corollary. Digon Graphs of Pref. Graphs are perfect when $n = 2$.

"So **Theorem 1.** follows as a corollary." □

Corollary. Long paths "alternate" in the digon graph



Corollary.

1. The digon graph contains no **odd** holes on ≥ 5 vertices.
2. The digon graph contains no antiholes on ≥ 5 vertices.

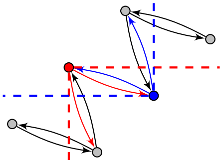
[CRST, '06] Strong Perfect Graph Theorem

A graph is perfect $\iff$ contains no odd holes and no odd antiholes.

Corollary. Digon Graphs of Pref. Graphs are perfect when $n = 2$.

"So **Theorem 1.** follows as a corollary." □

Corollary. Long paths "alternate" in the digon graph



Corollary.

1. The digon graph contains no **odd** holes on ≥ 5 vertices.
2. The digon graph contains no antiholes on ≥ 5 vertices.

[CRST, '06] Strong Perfect Graph Theorem

A graph is perfect $\iff$ contains no odd holes and no odd antiholes.

Corollary. Digon Graphs of Pref. Graphs are perfect when $n = 2$.

"So **Theorem 1.** follows as a corollary." □

Digon Graphs as a Class of Perfect Graphs

Open problem:

Characterize the class of digon graphs of pref. graphs for $n = 2$.

We know:

- DG $\nleq$ Complements of bipartite graphs
- DG $\nleq$ Complements of line gr. of bipartite graphs
- DG $\not\subset$ Bipartite graphs
- DG $\not\subset$ Line gr. of bipartite graphs

And maybe DG $\subsetneq$ Comparability graphs

3-Commodity Markets

Theorem 2. The decision problem is NP-complete for $n \geq 3$.

Proof Sketch.

1. **[WK, '88]**: Reduce planar 3-SAT to vert. cover on "gadget graphs"
2. **Remark.** Can reduce vert. cover to MFVS on a graph of digons.
3. **Lemma.** This graph of digons is an "oriented-disk graph"
4. **Lemma.** Any ODG is a valid preference graph for $n = 3$.

So **Theorem 2.** follows as a corollary. □

3-Commodity Markets

Theorem 2. The decision problem is NP-complete for $n \geq 3$.

Proof Sketch.

1. **[WK, '88]**: Reduce planar 3-SAT to vert. cover on "gadget graphs"
2. **Remark.** Can reduce vert. cover to MFVS on a graph of digons.
3. **Lemma.** This graph of digons is an "oriented-disk graph"
4. **Lemma.** Any ODG is a valid preference graph for $n = 3$.

So **Theorem 2.** follows as a corollary. □

3-Commodity Markets

Theorem 2. The decision problem is NP-complete for $n \geq 3$.

Proof Sketch.

1. **[WK, '88]**: Reduce planar 3-SAT to vert. cover on "gadget graphs"
2. **Remark.** Can reduce vert. cover to MFVS on a graph of digons.
3. **Lemma.** This graph of digons is an "oriented-disk graph"
4. **Lemma.** Any ODG is a valid preference graph for $n = 3$.

So **Theorem 2.** follows as a corollary. □

3-Commodity Markets

Theorem 2. The decision problem is NP-complete for $n \geq 3$.

Proof Sketch.

1. **[WK, '88]**: Reduce planar 3-SAT to vert. cover on "gadget graphs"
2. **Remark.** Can reduce vert. cover to MFVS on a graph of digons.
3. **Lemma.** This graph of digons is an "oriented-disk graph"
4. **Lemma.** Any ODG is a valid preference graph for $n = 3$.

So **Theorem 2.** follows as a corollary. □

3-Commodity Markets

Theorem 2. The decision problem is NP-complete for $n \geq 3$.

Proof Sketch.

1. **[WK, '88]**: Reduce planar 3-SAT to vert. cover on "gadget graphs"
2. **Remark.** Can reduce vert. cover to MFVS on a graph of digons.
3. **Lemma.** This graph of digons is an "oriented-disk graph"
4. **Lemma.** Any ODG is a valid preference graph for $n = 3$.

So **Theorem 2.** follows as a corollary. □

3-Commodity Markets

Theorem 2. The decision problem is NP-complete for $n \geq 3$.

Proof Sketch.

1. **[WK, '88]**: Reduce planar 3-SAT to vert. cover on "gadget graphs"
2. **Remark.** Can reduce vert. cover to MFVS on a graph of digons.
3. **Lemma.** This graph of digons is an "oriented-disk graph"
4. **Lemma.** Any ODG is a valid preference graph for $n = 3$.

So **Theorem 2.** follows as a corollary. $\square$

[WK, '88]: Reduce planar 3-SAT to vertex cover on "gadget graphs":

- Add a triangle per clause, labelled with variable type
- Add a cycle per variable, alternating labels
- Connect each clause to the cycles, opposite label.

Example. Graph for $\varphi = (\bar{u}_1 \vee u_2 \vee u_3) \wedge (u_1 \vee \bar{u}_2 \vee u_3)$

[WK, '88]: Reduce planar 3-SAT to vertex cover on "gadget graphs":

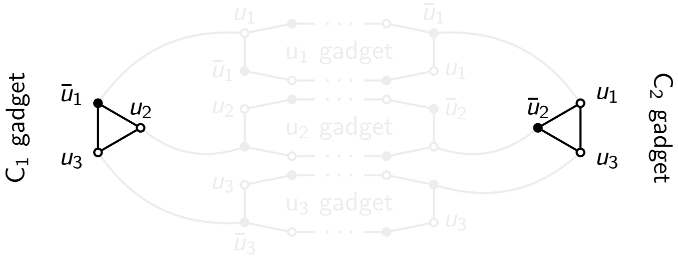
- Add a triangle per clause, labelled with variable type
- Add a cycle per variable, alternating labels
- Connect each clause to the cycles, opposite label.

Example. Graph for $\varphi = (\bar{u}_1 \vee u_2 \vee u_3) \wedge (u_1 \vee \bar{u}_2 \vee u_3)$

[WK, ‘88]: Reduce planar 3-SAT to vertex cover on “gadget graphs”:

- Add a triangle per clause, labelled with variable type
- Add a cycle per variable, alternating labels
- Connect each clause to the cycles, opposite label.

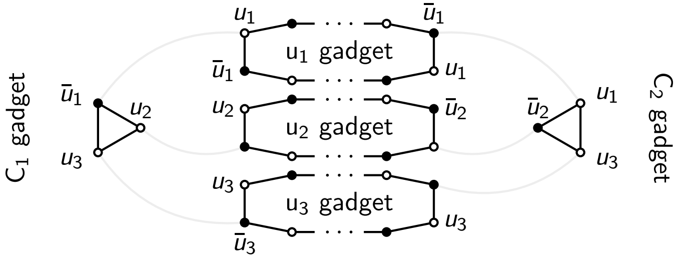
Example. Graph for $\varphi = (\bar{u}_1 \vee u_2 \vee u_3) \wedge (u_1 \vee \bar{u}_2 \vee u_3)$



[WK, '88]: Reduce planar 3-SAT to vertex cover on "gadget graphs":

- Add a triangle per clause, labelled with variable type
- Add a cycle per variable, alternating labels
- Connect each clause to the cycles, opposite label.

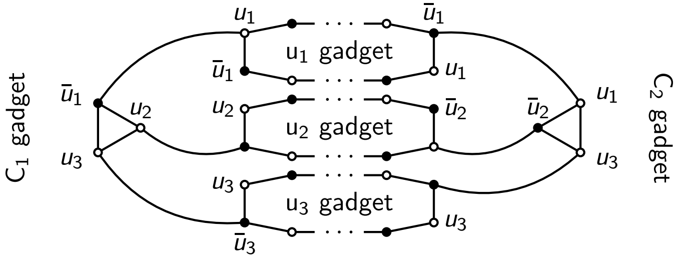
Example. Graph for $\varphi = (\bar{u}_1 \vee u_2 \vee u_3) \wedge (u_1 \vee \bar{u}_2 \vee u_3)$



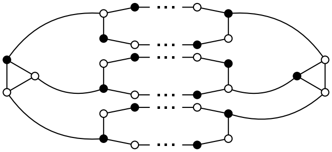
[WK, '88]: Reduce planar 3-SAT to vertex cover on "gadget graphs":

- Add a triangle per clause, labelled with variable type
- Add a cycle per variable, alternating labels
- Connect each clause to the cycles, opposite label.

Example. Graph for $\varphi = (\bar{u}_1 \vee u_2 \vee u_3) \wedge (u_1 \vee \bar{u}_2 \vee u_3)$



Reduction To Vertex Cover

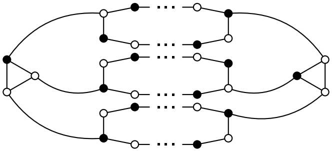


Note. Have a lower bound on vertex cover:

- Every clause needs at least 2 vertices to cover
- Every cycle needs at least half

[WK,'88] This bound is attainable if and only if the expression is satisfiable. $\Longrightarrow$ 3-SAT reduces to Vertex Cover on these graphs.

Reduction To Vertex Cover



Note. Have a lower bound on vertex cover:

- Every clause needs at least 2 vertices to cover
- Every cycle needs at least half

[WK,'88] This bound is attainable if and only if the expression is satisfiable. $\Longrightarrow$ 3-SAT reduces to Vertex Cover on these graphs.

3-Commodity Markets

Theorem 2. The decision problem is NP-complete for $n \geq 3$.

Proof Sketch.

1. **[WK, '88]**: Reduce planar 3-SAT to vert. cover on "gadget graphs"
2. **Remark.** Can reduce vert. cover to MFVS on a graph of digons.
3. **Lemma.** This graph of digons is an "oriented-disk graph"
4. **Lemma.** Any ODG is a valid preference graph for $n = 3$.

So **Theorem 2.** follows as a corollary. □

Oriented Disk Graphs

Definition. Let $x_1, \ldots, x_n$ be points on the plane, and $D_1, \ldots D_n$ be disks such that x_i is on D_i's boundary. Add an arc from x_i to x_j if $x_j \in D_i$. Any such graph is an *oriented-disk graph.*

Example.

Oriented Disk Graphs

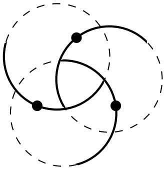
Definition. Let $x_1, \ldots, x_n$ be points on the plane, and $D_1, \ldots D_n$ be disks such that x_i is on D_i's boundary. Add an arc from x_i to x_j if $x_j \in D_i$. Any such graph is an *oriented-disk graph.*

Example.

Oriented Disk Graphs

Definition. Let $x_1, \ldots, x_n$ be points on the plane, and $D_1, \ldots D_n$ be disks such that x_i is on D_i's boundary. Add an arc from x_i to x_j if $x_j \in D_i$. Any such graph is an *oriented-disk graph.*

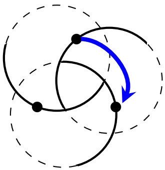
Example.



Oriented Disk Graphs

Definition. Let $x_1, \ldots, x_n$ be points on the plane, and $D_1, \ldots D_n$ be disks such that x_i is on D_i's boundary. Add an arc from x_i to x_j if $x_j \in D_i$. Any such graph is an *oriented-disk graph.*

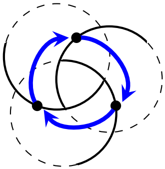
Example.



Oriented Disk Graphs

Definition. Let $x_1, \ldots, x_n$ be points on the plane, and $D_1, \ldots D_n$ be disks such that x_i is on D_i's boundary. Add an arc from x_i to x_j if $x_j \in D_i$. Any such graph is an *oriented-disk graph.*

Example.



Vertex Cover Reduces to FVS on a Digon Graph

Remark.

- Take any undirected graph, and replace each edge with a digon.
- Add an arbitrary acyclic set of arcs.
- Min. Vertex Cover on the underlying graph is equivalent to MFVS on this digraph. (Any new cycle passes through a digon)

Claim. There exists an **oriented-disk graph** which is a graph of digons for the gadget graph, plus an acyclic set of edges.

This would give a reduction from planar 3-SAT to MFVS on ODG's.

Clause Gadgets, Long Paths and Cycles

Vertex Cover Reduces to FVS on a Digon Graph

Remark.

- Take any undirected graph, and replace each edge with a digon.
- Add an arbitrary acyclic set of arcs.
- Min. Vertex Cover on the underlying graph is equivalent to MFVS on this digraph. (Any new cycle passes through a digon)

Claim. There exists an **oriented-disk graph** which is a graph of digons for the gadget graph, plus an acyclic set of edges.

This would give a reduction from planar 3-SAT to MFVS on ODG's.

Clause Gadgets, Long Paths and Cycles

Vertex Cover Reduces to FVS on a Digon Graph

Remark.

- Take any undirected graph, and replace each edge with a digon.
- Add an arbitrary acyclic set of arcs.
- Min. Vertex Cover on the underlying graph is equivalent to MFVS on this digraph. (Any new cycle passes through a digon)

Claim. There exists an **oriented-disk graph** which is a graph of digons for the gadget graph, plus an acyclic set of edges.

This would give a reduction from planar 3-SAT to MFVS on ODG's.

Clause Gadgets, Long Paths and Cycles

Vertex Cover Reduces to FVS on a Digon Graph

Remark.

- Take any undirected graph, and replace each edge with a digon.
- Add an arbitrary acyclic set of arcs.
- Min. Vertex Cover on the underlying graph is equivalent to MFVS on this digraph. (Any new cycle passes through a digon)

Claim. There exists an **oriented-disk graph** which is a graph of digons for the gadget graph, plus an acyclic set of edges.

This would give a reduction from planar 3-SAT to MFVS on ODG's.

Clause Gadgets, Long Paths and Cycles

Vertex Cover Reduces to FVS on a Digon Graph

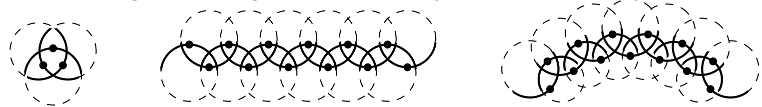
Remark.

- Take any undirected graph, and replace each edge with a digon.
- Add an arbitrary acyclic set of arcs.
- Min. Vertex Cover on the underlying graph is equivalent to MFVS on this digraph. (Any new cycle passes through a digon)

Claim. There exists an **oriented-disk graph** which is a graph of digons for the gadget graph, plus an acyclic set of edges.

This would give a reduction from planar 3-SAT to MFVS on ODG's.

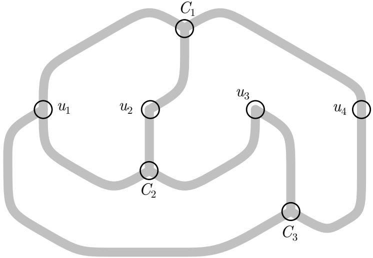
Clause Gadgets, Long Paths and Cycles



The Gadget Graph is Planar

What does this oriented-disk drawing look like?

1. Take a planar drawing of the variable-clause graph.
2. Replace the clause vertices with clause gadgets.*
3. Trace around the edges incident to each variable with its cycle.
4. * **How do we connect the cycles to the clause gadgets?**

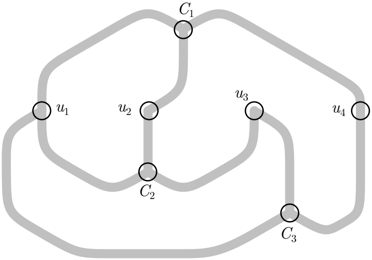


E.g. $\varphi = (u_1 \vee u_2 \vee u_4) \wedge (u_1 \vee u_2 \vee u_3) \wedge (u_1 \vee u_3 \vee u_4)$

The Gadget Graph is Planar

What does this oriented-disk drawing look like?

1. Take a planar drawing of the variable-clause graph.
2. Replace the clause vertices with clause gadgets.*
3. Trace around the edges incident to each variable with its cycle.
4. * **How do we connect the cycles to the clause gadgets?**

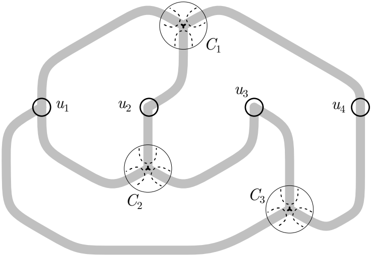


E.g. $\varphi = (u_1 \vee u_2 \vee u_4) \wedge (u_1 \vee u_2 \vee u_3) \wedge (u_1 \vee u_3 \vee u_4)$

The Gadget Graph is Planar

What does this oriented-disk drawing look like?

1. Take a planar drawing of the variable-clause graph.
2. Replace the clause vertices with clause gadgets.*
3. Trace around the edges incident to each variable with its cycle.
4. * **How do we connect the cycles to the clause gadgets?**

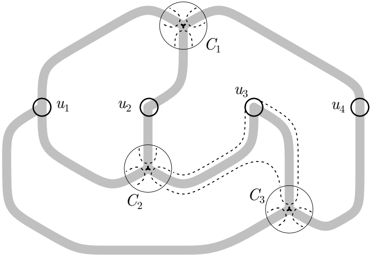


E.g. $\varphi = (u_1 \vee u_2 \vee u_4) \wedge (u_1 \vee u_2 \vee u_3) \wedge (u_1 \vee u_3 \vee u_4)$

The Gadget Graph is Planar

What does this oriented-disk drawing look like?

1. Take a planar drawing of the variable-clause graph.
2. Replace the clause vertices with clause gadgets.*
3. Trace around the edges incident to each variable with its cycle.
4. * How do we connect the cycles to the clause gadgets?

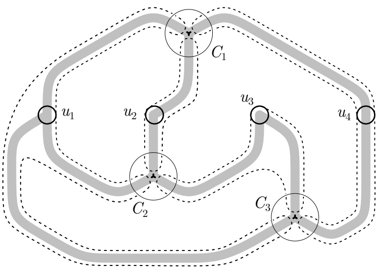


E.g. $\varphi = (u_1 \vee u_2 \vee u_4) \wedge (u_1 \vee u_2 \vee u_3) \wedge (u_1 \vee u_3 \vee u_4)$

The Gadget Graph is Planar

What does this oriented-disk drawing look like?

1. Take a planar drawing of the variable-clause graph.
2. Replace the clause vertices with clause gadgets.*
3. Trace around the edges incident to each variable with its cycle.
4. * **How do we connect the cycles to the clause gadgets?**

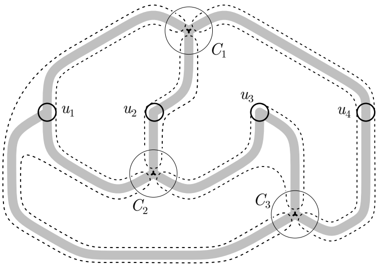


E.g. $\varphi = (u_1 \vee u_2 \vee u_4) \wedge (u_1 \vee u_2 \vee u_3) \wedge (u_1 \vee u_3 \vee u_4)$

The Gadget Graph is Planar

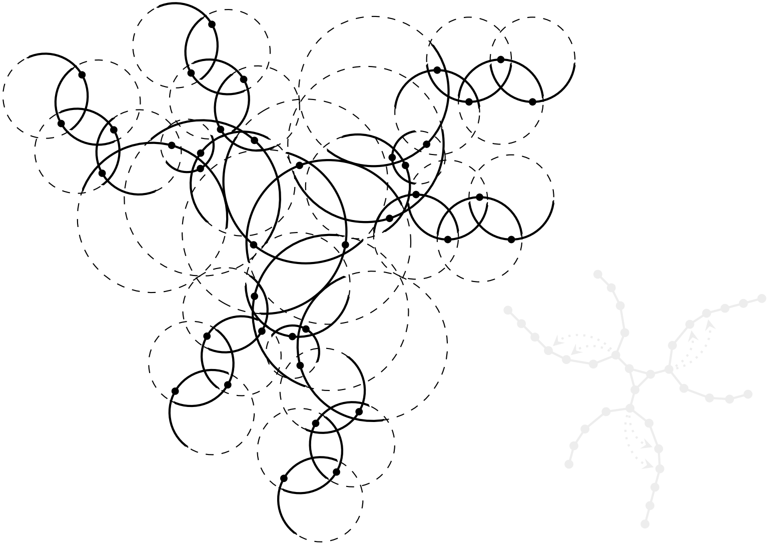
What does this oriented-disk drawing look like?

1. Take a planar drawing of the variable-clause graph.
2. Replace the clause vertices with clause gadgets.*
3. Trace around the edges incident to each variable with its cycle.
4. * **How do we connect the cycles to the clause gadgets?**

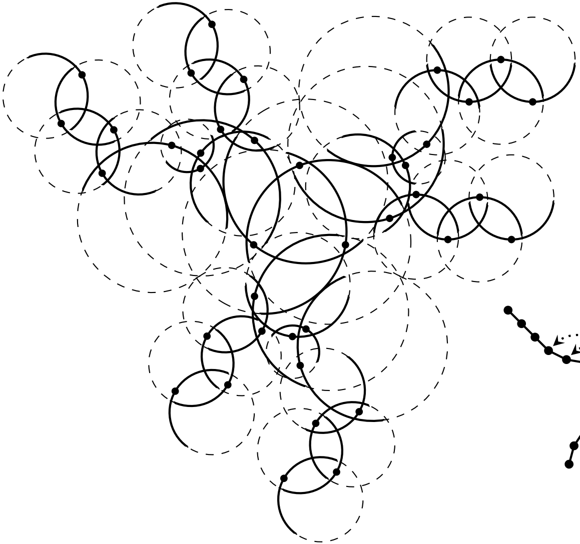


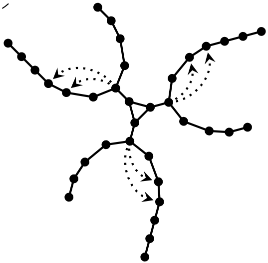
E.g. $\varphi = (u_1 \vee u_2 \vee u_4) \wedge (u_1 \vee u_2 \vee u_3) \wedge (u_1 \vee u_3 \vee u_4)$

Clause Gadget to Cycle Connection



Clause Gadget to Cycle Connection





3-Commodity Markets

Theorem 2. The decision problem is NP-complete for $n \geq 3$.

Proof Sketch.

1. **[WK, '88]**: Reduce planar 3-SAT to vert. cover on "gadget graphs"
2. **Remark.** Can reduce vert. cover to MFVS on a graph of digons.
3. **Lemma.** This graph of digons is an "oriented-disk graph"
4. **Lemma.** Any ODG is a valid preference graph for $n = 3$.

So **Theorem 2.** follows as a corollary. □

Lemma. Oriented-disk graphs are valid preference graphs for $n = 3$.

Can map any planar drawing onto a small sphere section

Map points onto sphere, cut through sphere with plane to get disks (Price vector is plane normal)

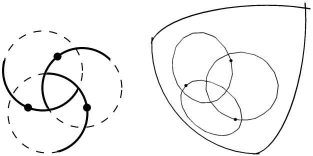
Vertex revealed preferred to any point inside disk

So any ODG is a preference graph,

and "**Theorem 2.** follows as a corollary." □

Lemma. Oriented-disk graphs are valid preference graphs for $n = 3$.

Can map any planar drawing onto a small sphere section



Map points onto sphere, cut through sphere with plane to get disks (Price vector is plane normal)

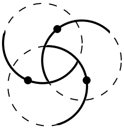
Vertex revealed preferred to any point inside disk

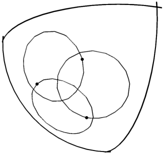
So any ODG is a preference graph,

and "**Theorem 2.** follows as a corollary." □

Lemma. Oriented-disk graphs are valid preference graphs for $n = 3$.

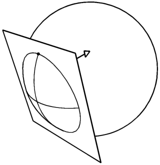
Can map any planar drawing onto a small sphere section





Map points onto sphere, cut through sphere with plane to get disks (Price vector is plane normal)

Vertex revealed preferred to any point inside disk

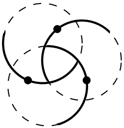


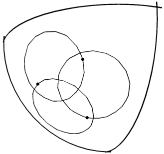
So any ODG is a preference graph,

and "**Theorem 2.** follows as a corollary." □

Lemma. Oriented-disk graphs are valid preference graphs for $n = 3$.

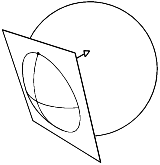
Can map any planar drawing onto a small sphere section





Map points onto sphere, cut through sphere with plane to get disks (Price vector is plane normal)

Vertex revealed preferred to any point inside disk



So any ODG is a preference graph,

and "**Theorem 2.** follows as a corollary." □

Further Research

We have shown that $n = 2$ is the threshold for poly-time solvability

- What about approximation complexity?
- Can we provide combinatorial algorithm for $n = 2$?
- Can we characterise preference graphs for fixed n?
- Does bounding DoR affect welfare in a combinatorial auction? (RP constraints are used to impose truthfulness.)

Thank you.

Further Research

We have shown that $n = 2$ is the threshold for poly-time solvability

- What about approximation complexity?
- Can we provide combinatorial algorithm for $n = 2$?
- Can we characterise preference graphs for fixed n?
- Does bounding *DoR* affect welfare in a combinatorial auction? (RP constraints are used to impose truthfulness.)

Thank you.

Further Research

We have shown that $n = 2$ is the threshold for poly-time solvability

- What about approximation complexity?
- Can we provide combinatorial algorithm for $n = 2$?
- Can we characterise preference graphs for fixed n?
- Does bounding *DoR* affect welfare in a combinatorial auction? (RP constraints are used to impose truthfulness.)

Thank you.

Further Research

We have shown that $n = 2$ is the threshold for poly-time solvability

- What about approximation complexity?
- Can we provide combinatorial algorithm for $n = 2$?
- Can we characterise preference graphs for fixed n?
- Does bounding *DoR* affect welfare in a combinatorial auction? (RP constraints are used to impose truthfulness.)

Thank you.

Further Research

We have shown that $n = 2$ is the threshold for poly-time solvability

- What about approximation complexity?
- Can we provide combinatorial algorithm for $n = 2$?
- Can we characterise preference graphs for fixed n?
- Does bounding DoR affect welfare in a combinatorial auction? (RP constraints are used to impose truthfulness.)

Thank you.

Further Research

We have shown that $n = 2$ is the threshold for poly-time solvability

- What about approximation complexity?
- Can we provide combinatorial algorithm for $n = 2$?
- Can we characterise preference graphs for fixed n?
- Does bounding *DoR* affect welfare in a combinatorial auction? (RP constraints are used to impose truthfulness.)

Thank you.