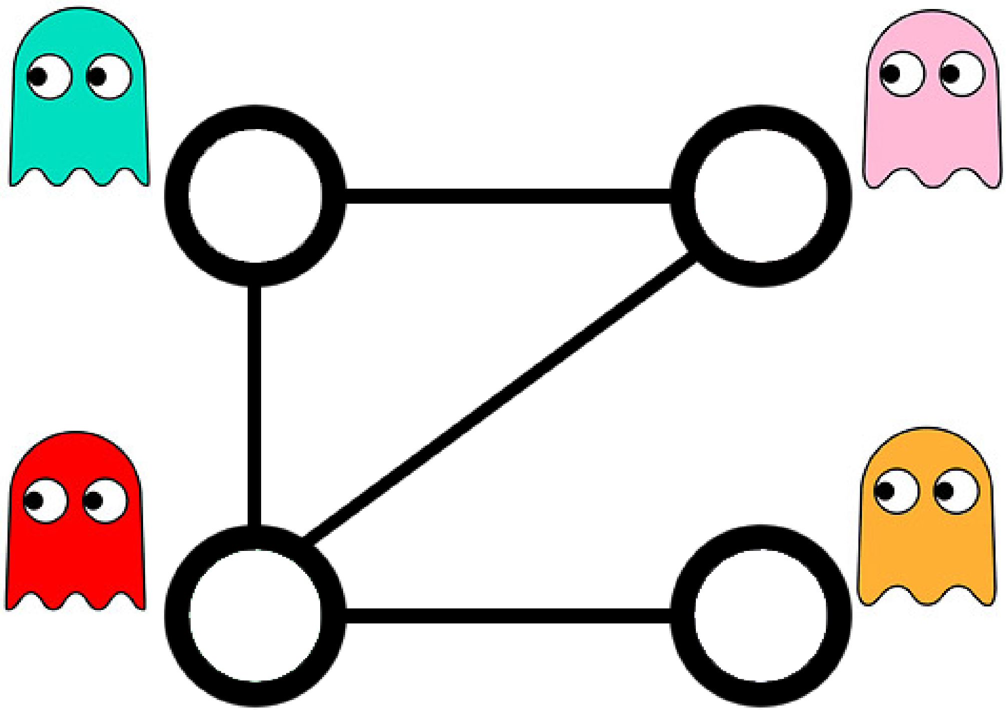
Anti-Coordination Games and Stable Graph Colorings

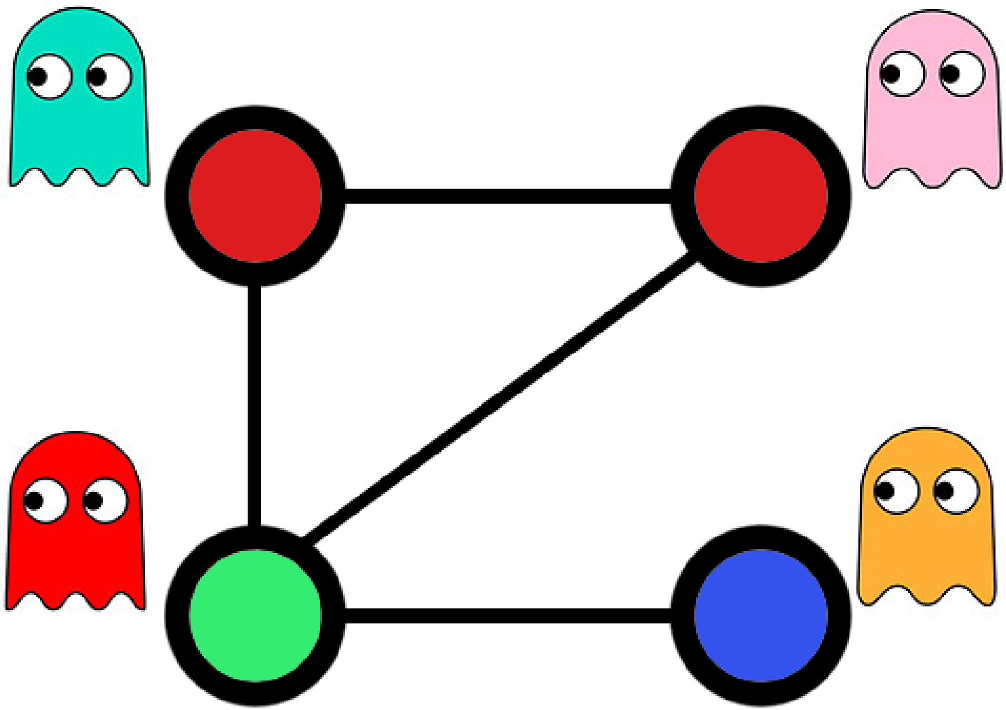
Jeremy Kun, Brian Powers, and Lev Reyzin

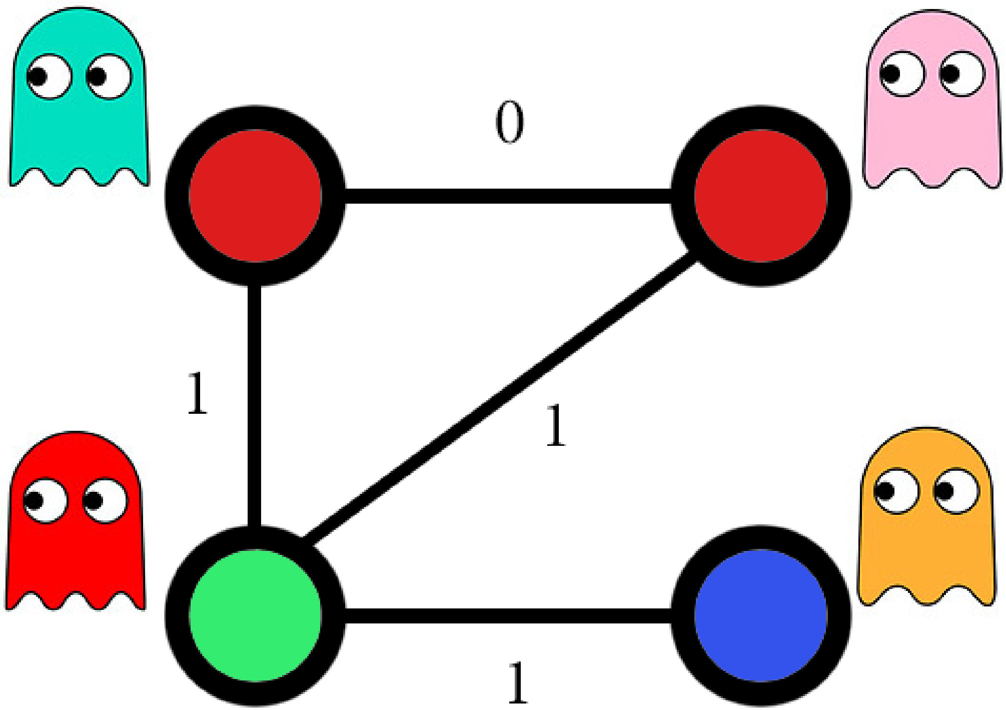
University of Illinois at Chicago
Department of Mathematics, Statistics, and Computer Science

22-10-2013

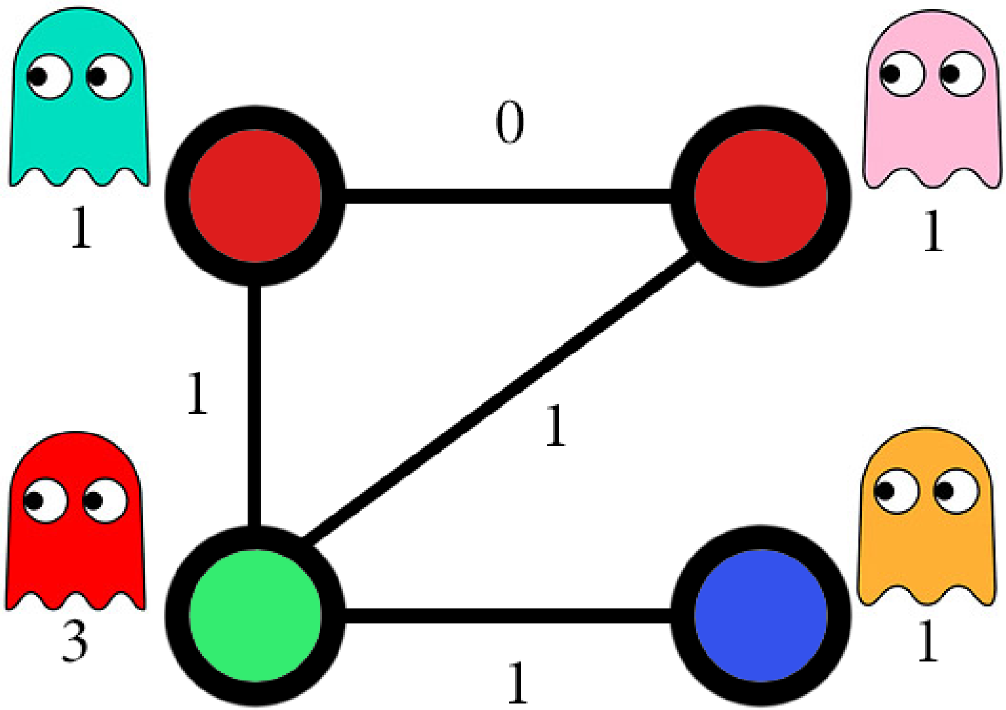
1. Model
2. Previous Work and Our Results
3. Undirected Graphs
4. Directed Graphs







0
1
1
1


1
0
1
1
1
1
3
1
1

Equilibrium Types

Pure strategy Nash equilibrium: no player can gain utility by changing colors.

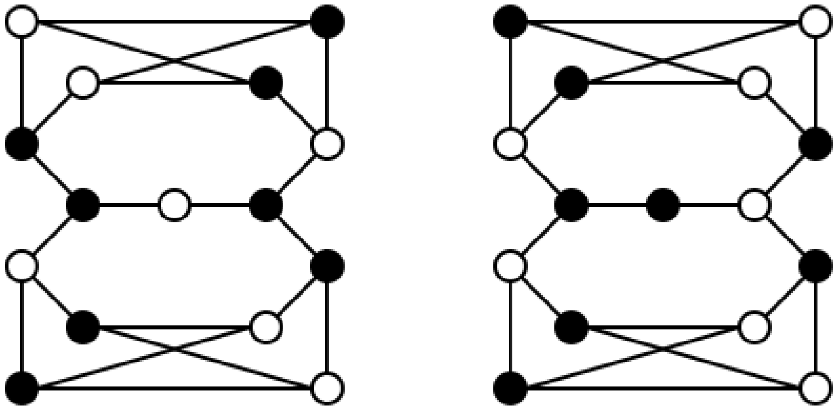
Strict Nash equilibrium: switching colors necessarily costs utility.

Our Goal

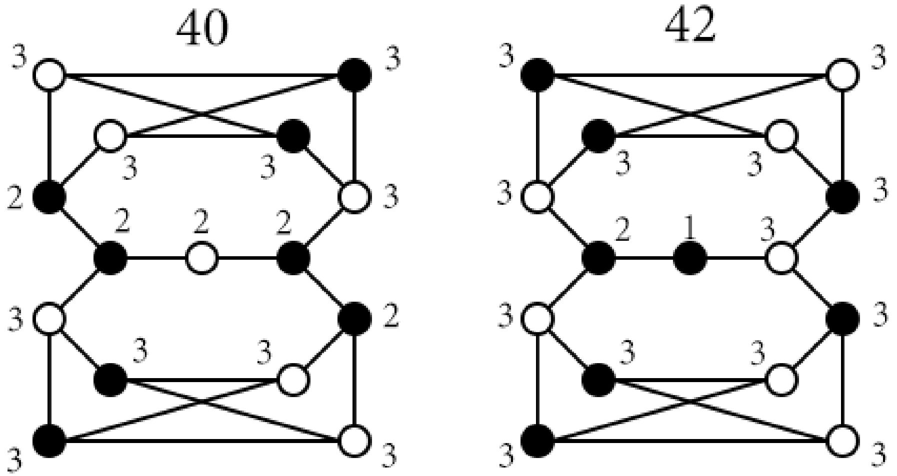
We analyze the decision problem for pure strategy Nash equilibria.

Theme: strictness behaves strangely in undirected graphs.

Strictness behaves strangely



Strictness behaves strangely



Non-strict equilibria can achieve higher social welfare (total payoff) than strict.

Previous Work and Our Results

Previous Work

- Experiments on humans [Kearns-Suri-Montfort '06]
- Clear relationship to MAX-CUT
- Unfriendly graph 2-partitions [Aharoni-Milner-Prikry '90, Hoefer '07, Gourvès-Monnot '09]
- Strict equilibria (variant) NP-hard for 2 strategies [Cao-Yang '12]

Generalization to directed graphs is natural, but largely unaddressed.

Our Results

Edge Type	Equilibrium Type: Non-Strict	Equilibrium Type: Strict
Undirected	in P	variant NP-hard (2 colors)
Directed		

Our Results

Edge Type		Equilibrium Type: Non-Strict	Equilibrium Type: Strict
	Undirected	in P	NP-hard
	Directed	NP-hard	NP-hard

Our Results

Edge Type	Equilibrium Type: Non-Strict	Equilibrium Type: Strict
Undirected	in P	NP-hard
Directed	NP-hard	NP-hard

Preliminary Analysis

Upper bound

Stable equilibria can be found in polynomial time for all k.

Algorithm: while there are vertices that can improve their current payoff, pick such a vertex and improve its payoff.

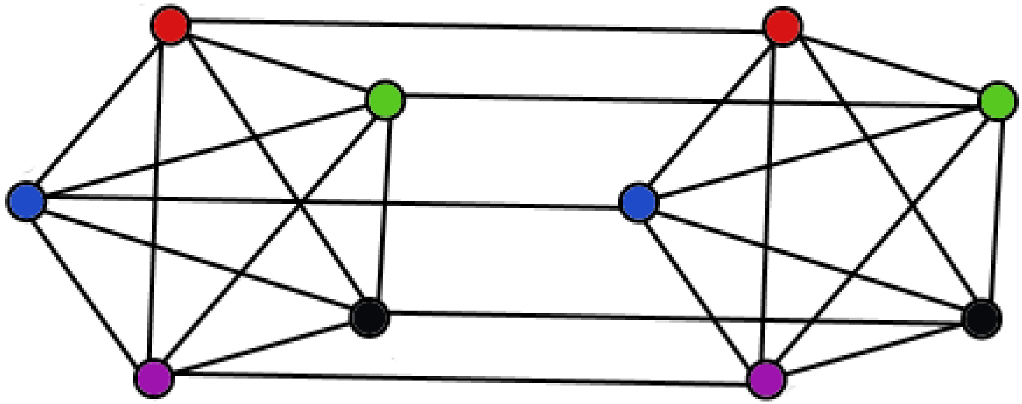
Social welfare is a bounded potential function, increasing by at least 1 each time any vertex improves its payoff.

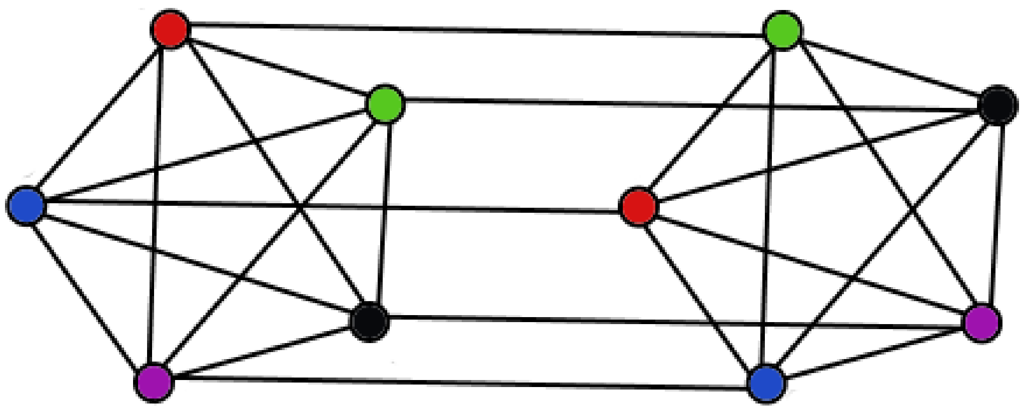
Price of Anarchy

Recall the *price of anarchy* is the ratio of the best social welfare of an equilibrium to the worst.

A tight bound for the price of anarchy for k strategies: $k/(k-1)$.

Price of anarchy $\leq k/(k-1)$: equilibirum implies each vertex must achieve at least a $(k-1)/k$ fraction of its max payoff.





Hardness for Strict Equilibria in Undirected Graphs

Hardness of undirected strict equilibria

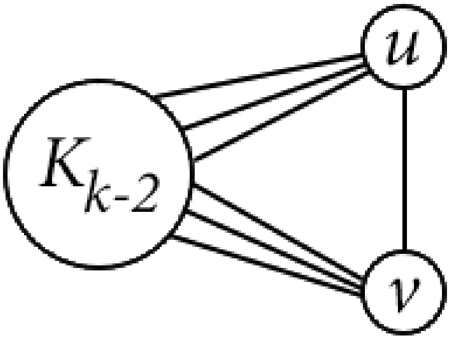
Theorem: For all $k \geq 2$, determining whether a graph has a strict Nash equilibrium with k colors is NP-complete.

Hardness of undirected strict equilibria

Theorem: For all $k \geq 2$, determining whether a graph has a strict Nash equilibrium with k colors is NP-complete.

$k \geq 3$ first, we reduce from k-coloring (2-coloring is not hard).

Given a 3-coloring instance G, attach a copy of the following subgraph for each edge $e = (u, v)$



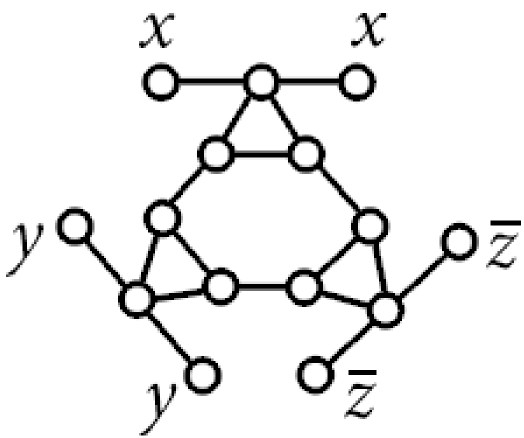
Hardness of undirected strict equilibria

For $k = 2$, reduce from 3-CNF satisfiability.

A literal x or $\overline{x}$ becomes a pair of vertices

Hardness of undirected strict equilibria

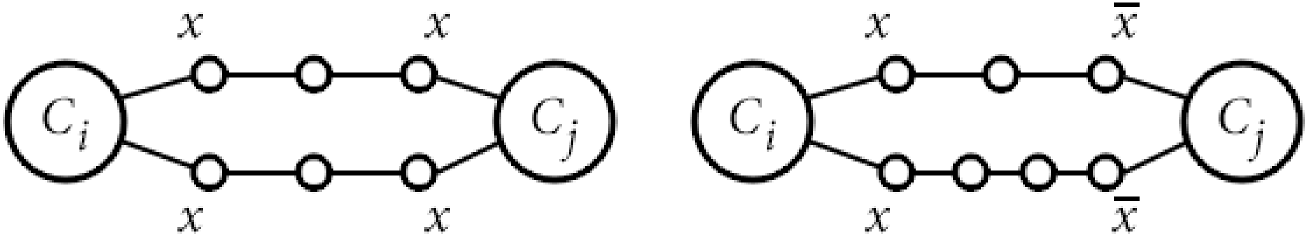
A clause (e.g.) $(x \vee y \vee \overline{z})$ becomes



Lemma: the clause gadget has a strict equilibrium iff some literal is colored true

Hardness of undirected strict equilibria

Connect literals across gadgets to be consistent.

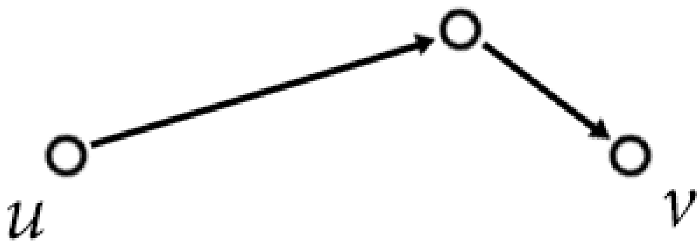


Non-strict Equilibria in Directed Graphs

Directed Graphs

A directed edge (u, v) means "u wants to anticoordinate with v."

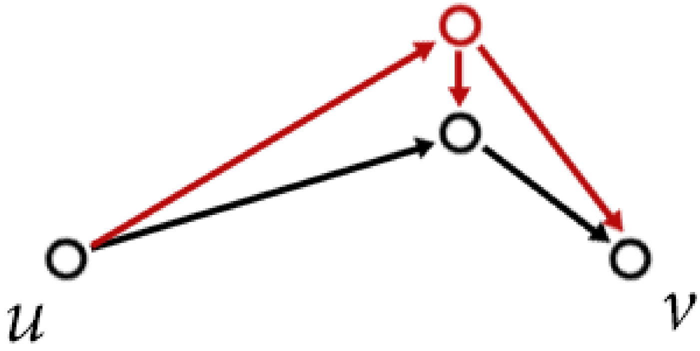
Here anticoordination captures coordination.



Directed Graphs

A directed edge (u, v) means "u wants to anticoordinate with v."

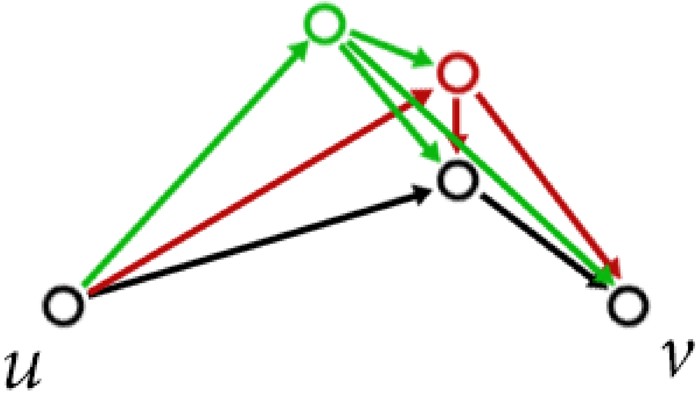
Here anticoordination captures coordination.



Directed Graphs

A directed edge (u, v) means "u wants to anticoordinate with v."

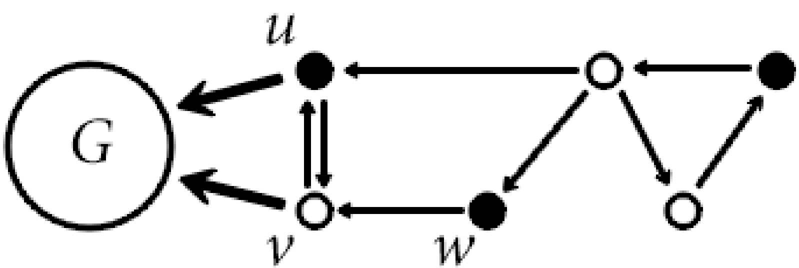
Here anticoordination captures coordination.



Hardness for directed graphs

Theorem: For all $k \geq 2$ determining whether a graph has a non-stirct equilibrium is NP-complete.

Idea: for $k = 2$ a directed 3-cycle has no equilibrium, so stabilize it with the solution to an NP-hard problem.



For $k \geq 3$, we can reduce to $k = 2$

Open problems:

- More interesting payoff schemes (e.g., channel assignment)
- Generalization to hypergraphs
- Better upper bounds for special graph classes
- Distributed algorithms

Questions?