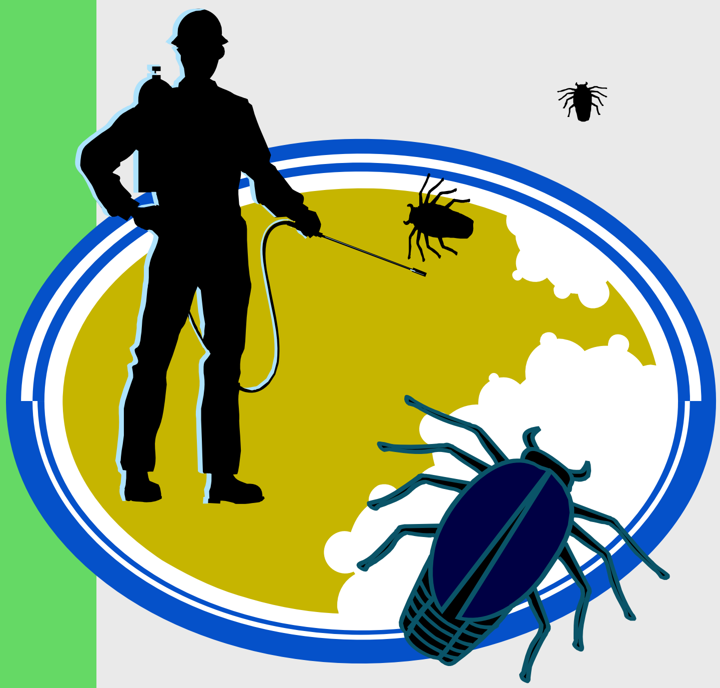
# The Power of the Defender




**M. Gelastou***

M. Mavronicolas*

V. Papadopoulou*

A. Philippou*

P. Spirakis§


IBC' 06, Lisbon, Portugal


*University of Cyprus, Cyprus

§University of Patras and RACTI, Greece

# Outline

- Introduction
- Model
- Previous Work and Motivation
- Results
- Conclusions
- Future work

# General Motivation

- Network Security: a critical issue in networks
  - Large network size
  - Dynamic nature of current networks
  - Economic factors
  - Low performance of protected nodes

# ⇒ Realistic Assumption:

*a Partially Secure Network*

security provided to a limited part of the network

# A Network Security Problem

- A partially secure network
  - *Defender* (firewall): protects the network
  - *Attackers* (viruses): damage the network (avoid the defender)

⇒ Attackers and defender have conflicting objectives

⇒ A *strategic game* with attacker players and a defender player

# Research Approach

- Algorithmic Game Theory
- Graph Theory

# Related Work

- [Mavronicolas, Papadopoulou, Philippou, Spirakis; ISAAC 2005]
  - Defender cleans a single edge:
    - *Edge model*
  - Pure Nash equilibria:
    - Non existence
  - Mixed Nash equilibria:
    - characterization
  - Matching Nash equilibria:
    - characterization and computation for bipartite graphs

# Related Work (cont.)

- [Mavronicolas, Papadopoulou, Philippou, Spirakis; WINE 2005]
  - Matching Nash equilibria:
    - computation for other classes of graphs
- [Mavronicolas, Michael, Papadopoulou, Philippou, Spirakis; MFCS 2006]
  - *Price of Defense*
  - Guarantees on Price of Defense for structured Nash equilibria

# Less Related Work

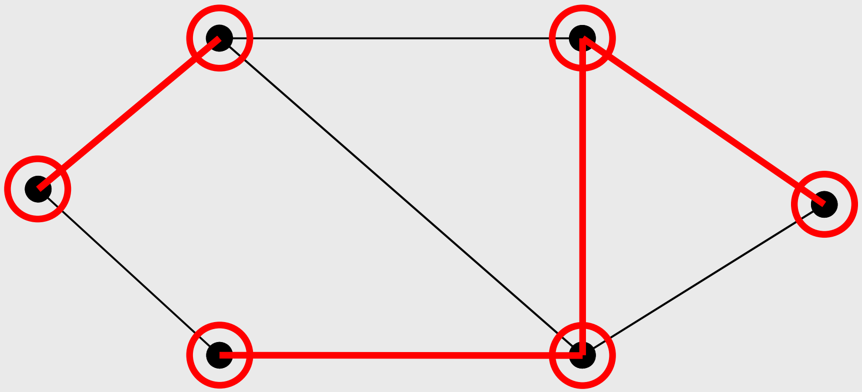
[Aspnes, Chang, Yampolskiy; SODA 2005]

- A different security game
- Connection to the Graph Partition problem

# Graph Theory Background

**A graph** G=(V,E)

- **Vertex Cover**
- **Edge Cover**
- **Independent Set**
- **Matching**



# A Strategic Game

For $1 \leq k \leq |E|$, consider the strategic game

$$\Pi_k(G) = \langle \mathcal{N}, \{S_i\}_{i \in \mathcal{N}}, \{\mathrm{IP}\}_{i \in \mathcal{N}} \rangle$$

- $\mathcal{N} = \mathcal{N}_{VP} \cup \mathcal{N}_{TP}$
- v *attackers* or ***vertex players*** $\mathbf{vp_i}$, with strategy set $S_{vp_i} = V$
- a *defender* or the ***tuple edge player*** **tep**, with strategy set $S_{tep} = E^k$ (all sets of *k* edges)

# Pure Strategies and Profiles

- ***Pure Strategy*** for player i:

  *a single* strategy from its strategy set

- ***Pure Profile***:

  a collection of pure strategies for all players

# Individual Profits

***Individual Profits*** in $\mathbf{s} = \langle s_1, \ldots s_\nu, s_{tep} \rangle \in \mathcal{S}$

- Vertex player $vp_i$:

  $$\text{IP}_i(\mathbf{s}) = 0 \text{ if } s_i \in s_{tep} \text{ or } 1 \text{ otherwise}$$

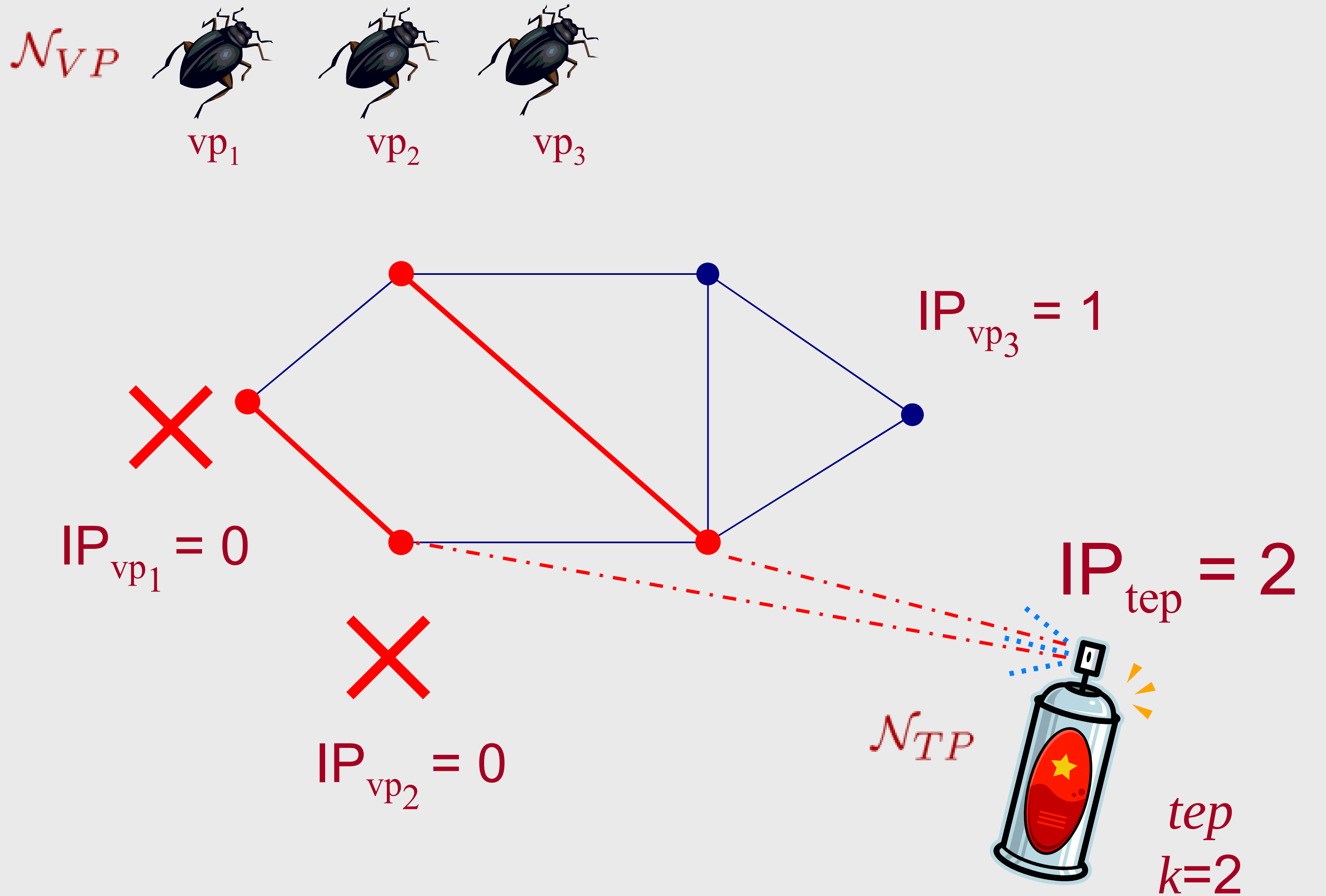
  gains 1 if it is not caught by the tuple edge player, and 0 otherwise

- Tuple edge player tep:

  $$\text{IP}_{tep}(\mathbf{s}) = |\{i : s_i \in V(s_{tep})\}|$$

  gains the number of vertex players incident to its selected tuple

# Game Example



# Mixed Strategies and Profiles

- ***Mixed strategy*** $\mathbf{s}_i$ for player $i$:

  a probability distribution over its strategy set

- ***Mixed profile*** $\mathbf{s}$:

  a collection of mixed strategies for all players

- ***Support*** of player $i$:

  set of pure strategies receiving positive probability

- ***Expected Individual Profit*** $IP_i$ :

  expectation of Individual Profit of player $i$ in profile $\mathbf{s}$

# Notation

- ***tuple*** **t**: a set of $k$ edges
- **V(t)**: vertices incident to the edges of tuple **t**
- **E(S)**: distinct edges of the set of tuples **S**

In a profile **s**,

- $s_{tep}(t)$: probability that tep chooses tuple **t**

# Notation (cont.)

In a profile **s**,

- ***Support$_s$(i)***:

  the support of player $i$

- ***Support$_s$(VP)***:

  the support of all vertex players

- ***Tuples$_s$($\upsilon$)*** $= \{ t : \upsilon \in V(t), t \in \text{Support}_s(tep) \}$:

  set of tuples of the support of the tuple edge player that contain vertex $\upsilon$

Model

# Notation (cont.)

In a profile **s**,

- **Hit(υ)**:

  the event that the tuple edge player chooses tuple that contains vertex $\upsilon$

  **$P_s$(Hit(υ))** = $\sum_{t \in Tuples_{\mathbf{s}}(\upsilon)} \mathbf{s}_{tep}(t)$

- **$VP_s$(υ)**:

  expected number of vertex players choosing vertex $\upsilon$

- **$VP_s$(t)**:

  expected number of vertex players on vertices of the tuple $t$

# Profiles

- ***Uniform***:

  uniform probability distribution on each player’s support

- ***Attacker Symmetric***:

  all vertex players have the same distribution

# Nash Equilibrium (NE)

No player can unilaterally improve its Individual Profit by switching to another strategy.

# Edge Model

- Edge Model [MPPS´05] = Tuple model for $k$ =1
- In a *Covering* profile **s** [MPPS´05]:
  - $\text{Support}_s$(ep) is an Edge Cover
  - $\text{Support}_s$(VP) is a Vertex Cover of G($\text{Support}_s$(ep))

# Edge Model (cont.)

- [MPPS´05]. An *Independent Covering* profile is a Uniform, Attacker Symmetric Covering profile such that:
  - $Support_s(VP)$ is Independent Set
  - Each vertex of $Support_s(VP)$ is incident to only *one* edge of $Support_s(ep)$

# Edge Model (cont.)

- **Theorem** [MPPS'05].

  An Independent Covering profile is a Nash equilibrium.

# Motivation

- Extend the Edge model $\Rightarrow$ Tuple model
  - Increased power to the defender
  - Increased quality of the protection provided in the network

# Summary

- Graph-theoretic characterization of Nash Equilibria
- Necessary conditions for Nash Equilibria

  $\Rightarrow$ *k*-Covering profiles
- Independent *k*-Covering profiles
  - are Nash equilibria
    - called *k*-Matching Nash equilibria

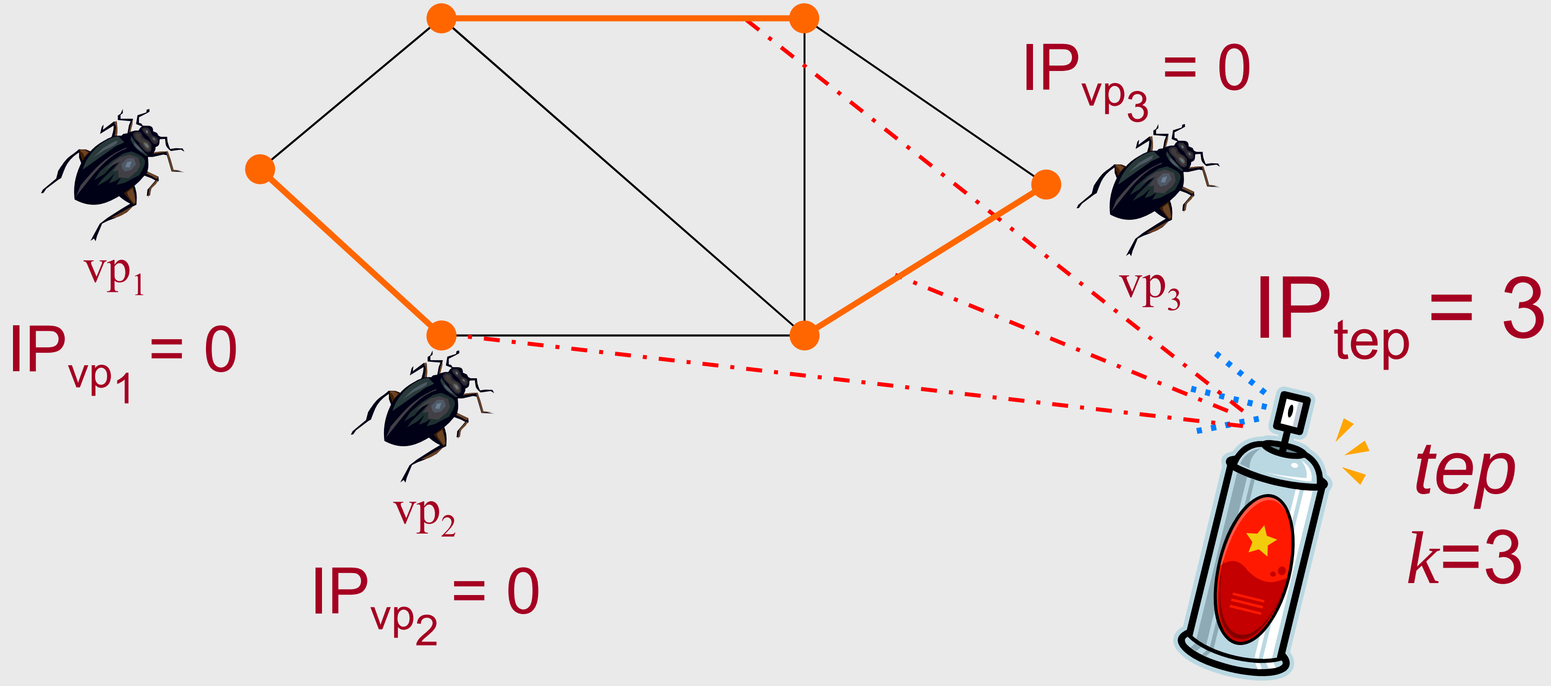
# Summary (cont.)

- Characterization of graphs admitting *k*-Matching Nash equilibria
- Polynomial-time algorithm for computing a *k*-Matching Nash equilibrium
- The Individual Profit of the defender is multiplied by *k* compared to the Edge model

# Pure Nash Equilibria

- **Theorem 1.**

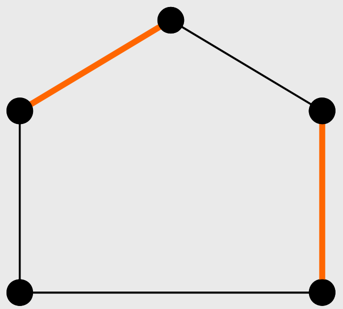
  G admits a pure Nash equilibrium if and only if G has an Edge Cover of size *k*.

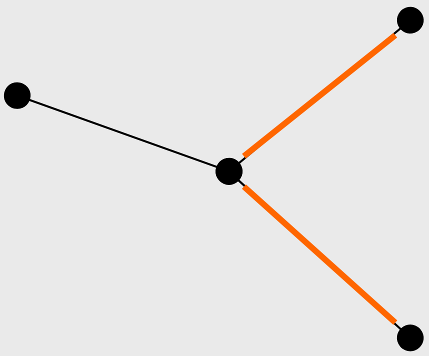


# Pure Nash Equilibria (cont.)

- If $|V(G)| \geq 2k+1$, then G admits no pure NE.
- If $|V(G)| \leq 2k$, G does not necessarily admit a Nash equilibrium.

k=2





# Characterization of Nash Equilibria

- **Theorem 2.**

  A profile **s** is a Nash Equilibrum if and only if:

  – For any vertex $v \in \mathrm{Support}_{\mathbf{s}}(VP)$,

  $$P_{\mathbf{s}}(\mathrm{Hit}(v)) = \min_{v} P_{\mathbf{s}}(\mathrm{Hit}(v))$$

  – For any tuple $\mathbf{t} \in \mathrm{Support}_{\mathbf{s}}(tep)$,

  $$VP_{\mathbf{s}}(t) = \max_{\mathbf{t}} VP_{\mathbf{s}}(\mathbf{t})$$

# Necessary Conditions for NE

- **Definition 2.**

  A *k-Covering* profile **s** of $\Pi_k(G)$ satisfies:

  – $Support_s(tep)$ is an Edge Cover

  – $Support_s(VP)$ is a Vertex Cover of $G(Support_s(tep))$

# Necessary Conditions for NE

- **Proposition 3.**

  A Nash equilibrium is a *k*-Covering profile.

# Independent *k*-Covering Profiles

**Definition 3.** An *Independent k-Covering* profile is a Uniform, Attacker Symmetric Covering profile such that:

- $Support_s(VP)$ is an Independent Set
- Each vertex of $Support_s(VP)$ is incident to only *one* edge of $E(Support_s(tep))$.
- Each edge in $E(Support_s(tep))$ belongs to an equal number of distinct tuples of $Support_s(tep)$.

# *k*-Matching Nash Equilibria

- **Theorem 3.**

  An Independent *k*-Covering profile is a Nash Equilibrium.

- Call it a *k-Matching* Nash Equilibrium

# The Power of the Defender

- **Proposition 3.**

  Computing a Matching Nash equilibrium $\mathbf{s}^1$ for $\Pi_1(G)$ and computing a *k*-Matching Nash equilibrium $\mathbf{s}^k$ of $\Pi_k(G)$ are polynomial time equivalent.

# The Power of the Defender

- **Theorem 4.**

  Assume that G admits a Matching Nash Equilibrium $\mathbf{s}^1$ for $\Pi_1(G)$. Then G admits a $k$-Matching Nash Equilibrium $\mathbf{s}^k$ for $\Pi_k(G)$ with $\mathsf{IP}_{tep}(s^k) = k \cdot \mathsf{IP}_{ep}(s^1)$.

# Characterization of *k*-Matching NE

- **Definition 4.**

  The graph G is a *U-Expander* graph if for each set U'µ U,

  $$|U'| \cdot |\, \mathrm{Neigh}_G (U') \text{ Å } (V \setminus U)\, |.$$

# Characterization of *k*-Matching NE

- **Theorem 5.**

  A G admits a *k*-Matching Nash Equilibrium if and only if G contains an Independent Set IS such that G is a (V\IS)-Expander graph.

# Polynomial Time Algorithm $A_{tuple}$

**INPUT**: A game $\Pi_k(G)$, with an Independent set of G such that G is a V\IS-Expander graph.

**OUTPUT**: A Nash equilibrium $\mathbf{s}^k$ for $\Pi_k(G)$

1. Compute a Matching Nash equilibrium $\mathbf{s}^1$ for $\Pi_1(G)$ [MPPS, ISAAC 2005]
2. Compute a tuple set $T$
3. Construct a Uniform, Attacker Symmetric profile $\mathbf{s}^k$ with:
   - $\text{Support}_{s^k}(\text{tep}) = T$

# Computation of Tuple Set *T*

1. Label the edges of $\text{Support}_{s1}(ep)$

   $e_0, e_1, \ldots, e_{E_{num}}$

2. Do

   a) Construct a tuple $\mathbf{t}_i$ of $k$ edges such that

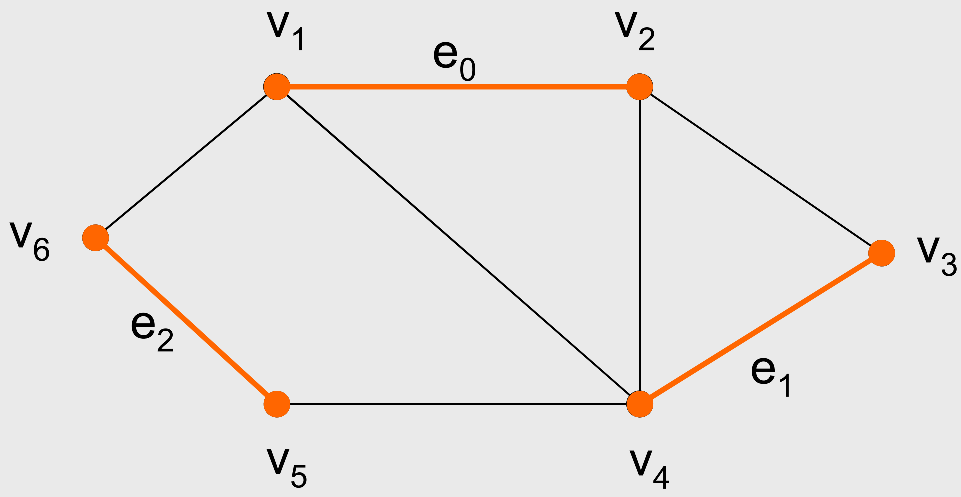
   $$t_i = \langle e_{((i-1)\cdot k) mod(E_{num})}, \ldots, e_{(i\cdot k-1) mod(E_{num})} \rangle$$

   b) $\boldsymbol{T} = \boldsymbol{T} \cup \{\mathbf{t}_i\}$

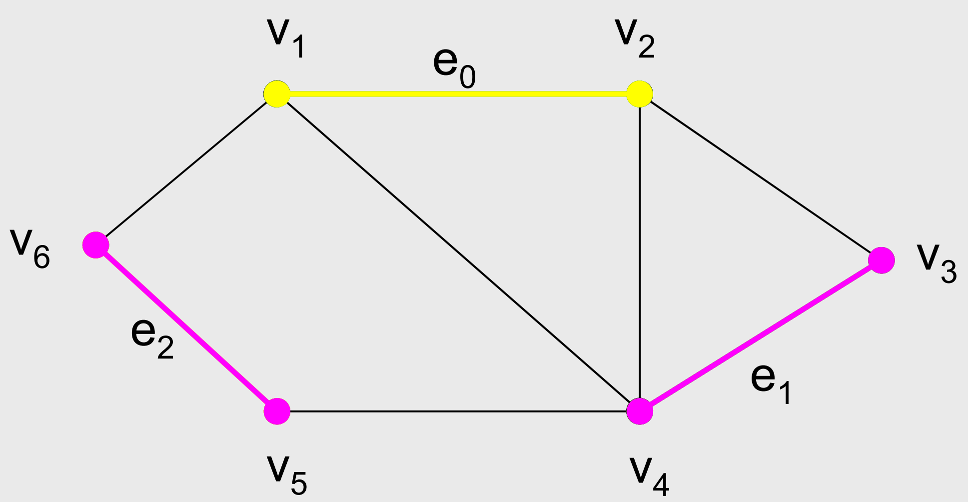
   while $|T| = \frac{E_{num}}{GCD(E_{num}, k)}$

# Example

- $\text{Support}_{s1}(ep) = <e_0, e_1, e_2>$



# Example (cont)




- $k$=2

$$|T| = \frac{E_{num}}{GCD(E_{num},k)} = \frac{3}{GCD(3,2)} = 3$$

$$\Rightarrow \mathbf{T} = \{ \langle e_0,e_1\rangle, \langle e_2,e_0\rangle, \langle e_1,e_2\rangle \}$$

# Polynomial Time Algorithm (cont.)

- **Theorem 6.**

  Algorithm $A_{tuple}$ computes a $k$-Matching Nash equilibrium in time

  $$\mathbf{O(k \cdot n + T(G))}$$

  $T(G)$: the time needed to compute a Matching Nash equilibrium for the Edge model.

# Application

- **Corollary 1.**

  A bipartite graph G admits a *k*-Matching Nash equilibrium which can be computed in polynomial time

$$O\left(\sqrt{n} \cdot m \cdot \log_n \frac{n^2}{m}\right).$$

# Conclusions

- Characterized Pure and Mixed Nash Equilibria
- Polynomial-time algorithm for computing $k$-Matching Nash equilibria
- Increased protection of the network through the increased power of the defender

# Future Work

- Other families of structured Nash equilibria
- *Path model*: The defender protects a path of length $k$

# Thank you !