Link-preserving channel assignment game for wireless mesh networks

Li-Hsing Yen, Bo-Rong Ye

SIMULATION PROGRAM

Outline

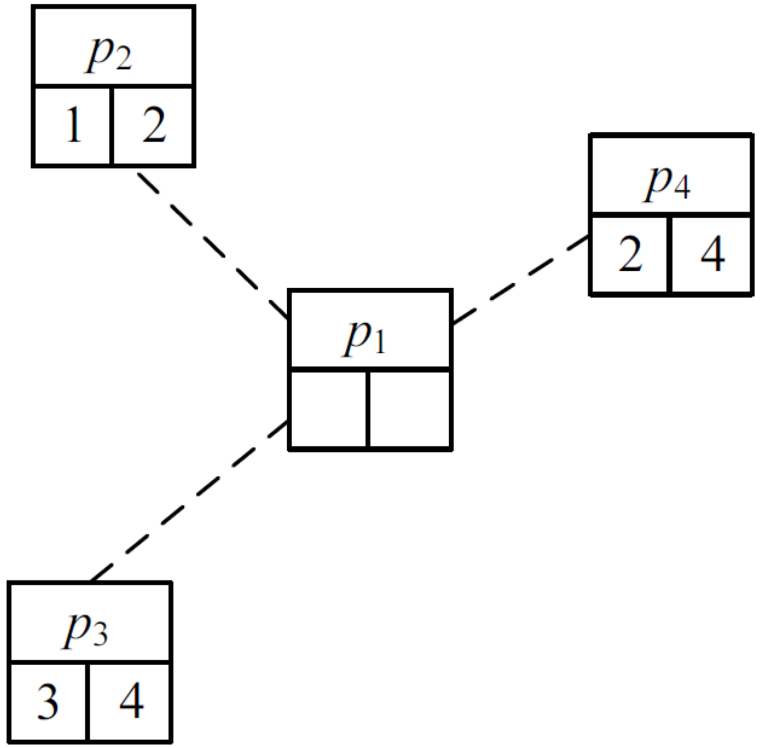
- Introduction
- Implementation of LPIM
- Simulation
- Conclusion

Introduction

Link-Preserving Interference-Minimization (LPIM)

- A channel assign game
- Exact potential game

  => always stabilizes
- Link-Preserving



Define of the game

Players: the set of mesh stations, $P = \{p_1,\ p_2, \dots, p_n\}$

Channels: set of orthogonal channels, $C = \{k_1, k_2, \dots, k_m\}$

Strategy: $s_i = \left(c_1^i, c_2^i, \dots, c_m^i\right)$ for p_i, where c_j^i = 1 or 0 indicating whether p_i assigns channel k_j to one of its

Number of interface: $r_i = \min(r_{max}, |N_i|)$, N_i number of neighbor, r_{max} max number of interface

Assume the number of available channels is at least as many as the number of interfaces

Impact of co-channel interference.

s_i . s_j : the number of common channels assigned by both p_i and p_j

$$I_i(S) = -\sum_{p_j \in N_i} \left(\boxed{s_i \cdot s_j} \right)$$

Gains of connectivity

$$L_{i(S)} = \Sigma_{p_j \in N_i} C_i(s_i \,.\, s_j),$$

Where $C_i(\mathbf{s}_i, \mathbf{s}_j) = \begin{cases} -|N_i| & \text{if } \mathbf{s}_i \cdot \mathbf{s}_j = 0 \\ 0 & \text{otherwise} \end{cases}$

For a particular strategy profile S

Combining previous two formula

β : a constant, where $\beta > r_{max}$

$$t_i(S) = \beta L_i(S) + I_i(S)$$

Utility and Potential function

Utility:

$$u_i(S) = t_i(S) + \sum_{p_j \in N_i} t_j(S)$$

Potential function:

$$\phi(S) = \sum_i t_i(S)$$

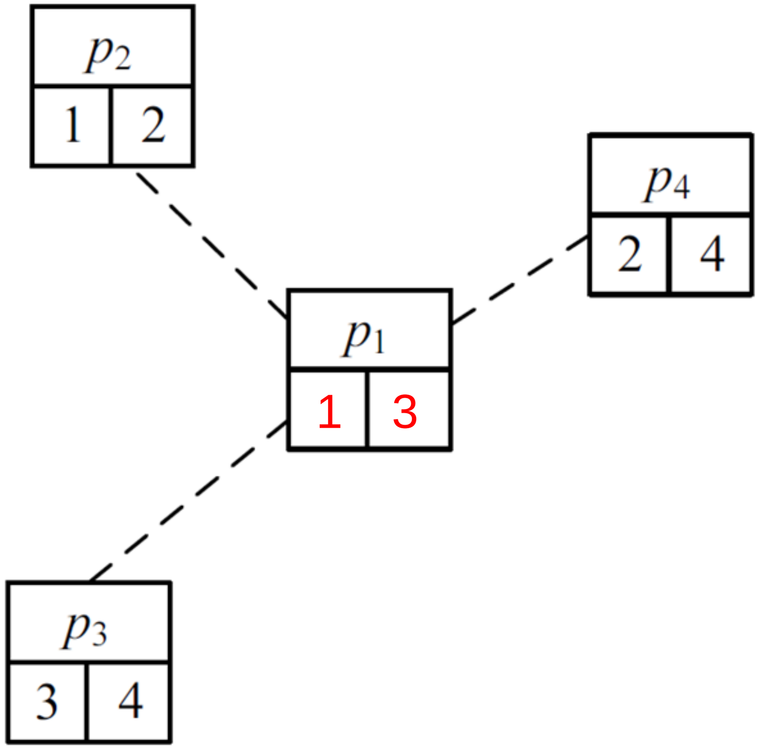
Utility example (1/2)

$L_1(S) = 0 + 0 + (-3) = -3$

$I_1(S) = (-1) + (-1) + 0 = -2$

$t_1(S) = 5 \times (-3) + (-2) = -17, \beta = 5$

$u_i(S) = -17 + (-1) + (-1) + (-5) = -24$



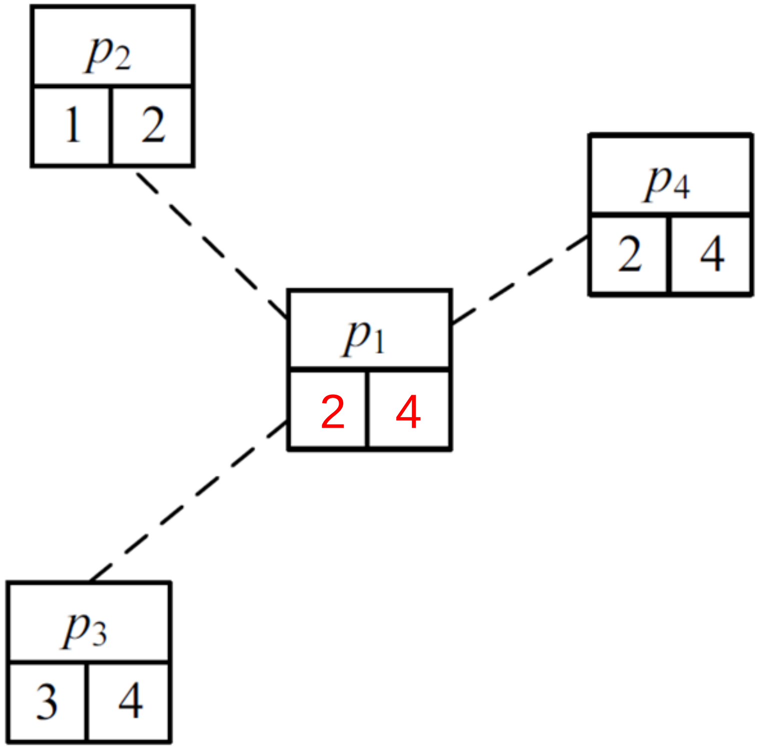
Utility example (2/2)

$L_1(S) = 0 + 0 + 0 = 0$

$I_1(S) = (-1) + (-1) + (-2) = -4$

$t_1(S) = 5 \times (0) + (-4) = -4, \beta = 5$

$u_i(S) = -4 + (-1) + (-1) + (-2) = -8$



Exact potential game

Not affected by p_i's move

$$\phi(\bar{S}) - \phi(S) = t_i(\bar{S}) - t_i(S) + \sum_{p_j \in N_i} (t_j(\bar{S}) - t_j(S)) + \boxed{\sum_{p_j \in P \setminus N_i} (t_j(\bar{S}) - t_j(S))}$$

$$= t_i(\bar{S}) - t_i(S) + \sum_{p_j \in N_i} (t_j(\bar{S}) - t_j(S))$$

$$= u_i(\bar{S}) - u_i(S)$$

Best-response rule and Nash equilibrium

Player p_i selects strategy s_i^* only if

$$s_i^* = \arg\max_{s_i \in S_i} u_i(s_i, s_{-i})$$

Nash equilibrium $S = (s_1, s_2, \ldots, s_n)$ Not unique

If $u_i(s_i, s_{-i}) \geq u_i(s_i^*, s_{-i})$, $\forall i \in \{1, 2, \ldots, n\}$

$$\forall s_i^* \in S_i$$

Link-Preserving

Highest interference value of p_i:

$r_i = \min(r_{max}, |N_i|)$

$$I_i(S) = -\sum_{p_j \in N_i} \min(r_i, r_j) \geq -r_{\max}|N_i|$$

P_i no common channel on some neighbors :

$$L_i(S) \leq -|N_i| \Rightarrow \beta L_i(S) \leq -\beta|N_i| < -r_{\max}|N_i|$$

$\beta > r_{max}$

LPIM(PP)

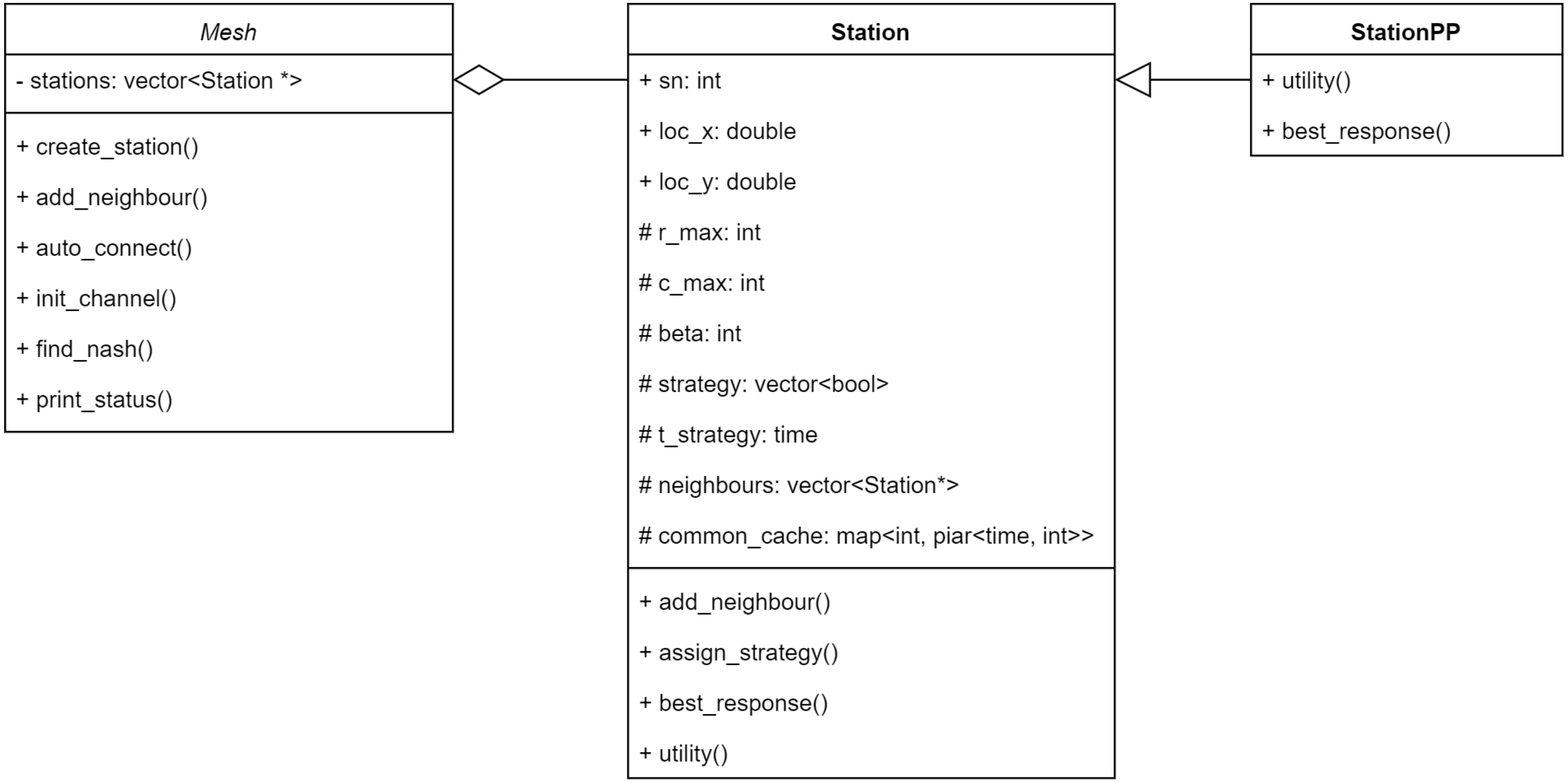
- A variant of LPIM
- Considers only the impact of interference
- Adheres to the Pigeonhole principle
- Exact potential game
- New utility function:

$$u_i(S) = -\sum_{p_j \in N_i} (s_i \cdot s_j)$$

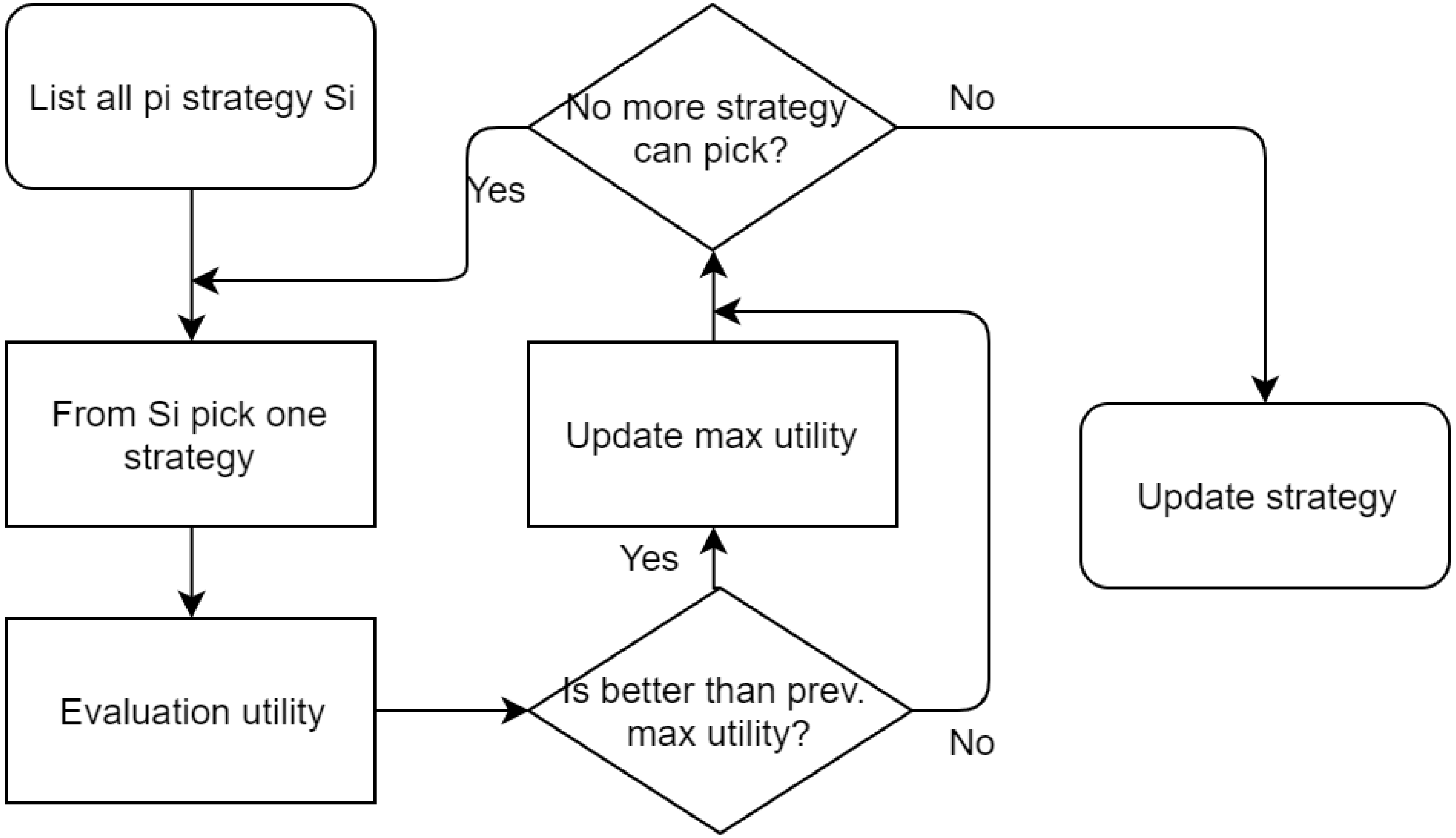
$$c_k^i = 0, \forall\, p_i \in P \ and\ k > \min_{p_j \in N_i} \{r_i + r_j - 1\}$$

Implementation of LPIM

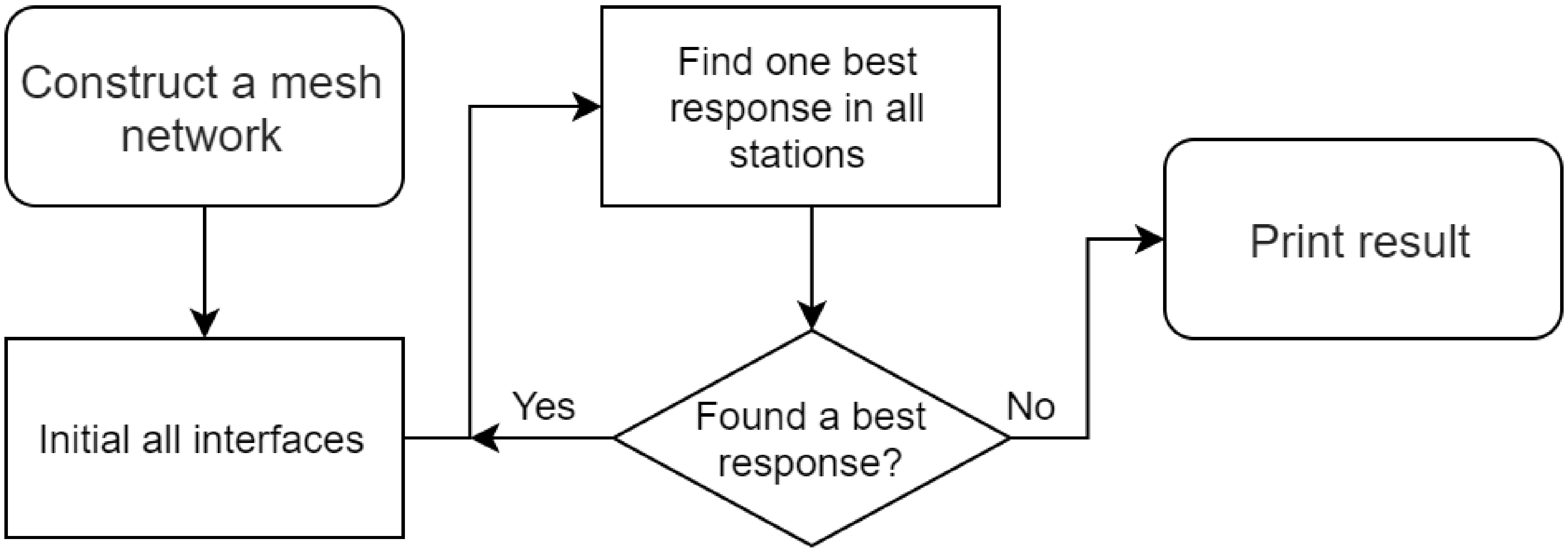
UML


Mesh
- stations: vector<Station *>
+ create_station()
+ add_neighbour()
+ auto_connect()
+ init_channel()
+ find_nash()
+ print_status()
Station
+ sn: int
+ loc_x: double
+ loc_y: double
# r_max: int
# c_max: int
# beta: int
# strategy: vector<bool>
# t_strategy: time
# neighbours: vector<Station*>
# common_cache: map<int, piar<time, int>>
+ add_neighbour()
+ assign_strategy()
+ best_response()
+ utility()
StationPP
+ utility()
+ best_response()

Flow Chart for finding BP of p_i



Flow Chart for finding Nash equilibrium



Simulation

A five-station mesh network example

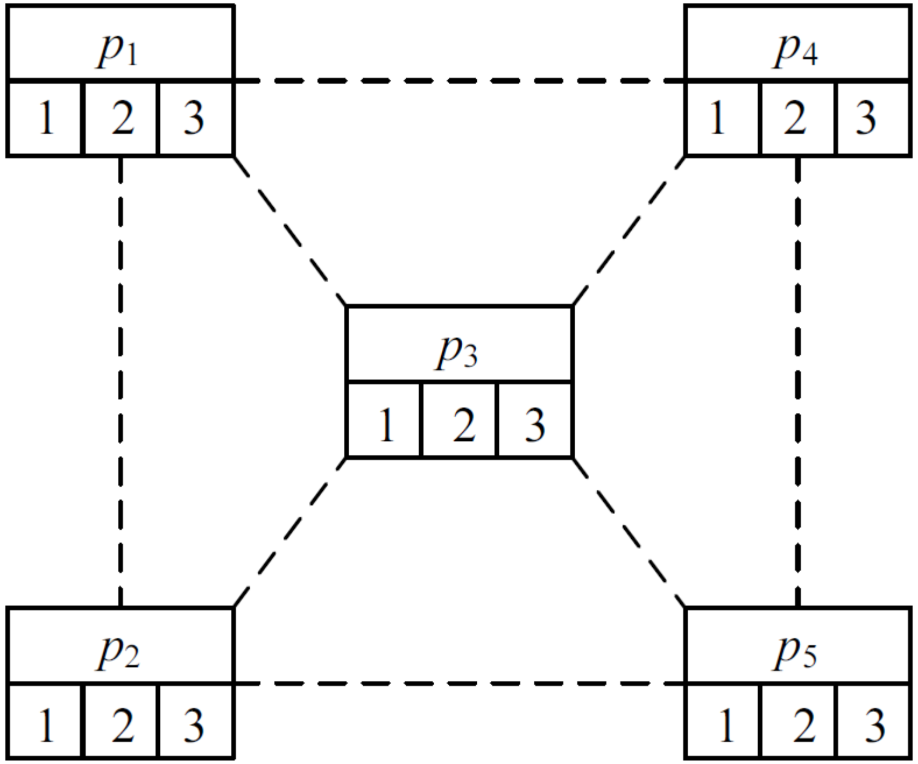


Fig. A five-station mesh network

- 5 stations(p_1 、 p_2 … p_5)
- Max number of interfaces: 3
- Number of channel: 7

Best-reply path

Step	s_1	s_2	s_3	s_4	s_5
0	$\{1,2,3\}$	$\{1,2,3\}$	$\{1,2,3\}$	$\{1,2,3\}$	$\{1,2,3\}$
1	$\{1,2,3\}$	$\{1,2,3\}$	$\{1,4,5\}$	$\{1,2,3\}$	$\{1,2,3\}$
2	$\{1,2,3\}$	$\{1,2,3\}$	$\{1,4,5\}$	$\{2,4,6\}$	$\{1,2,3\}$
3	$\{1,2,3\}$	$\{1,2,3\}$	$\{1,4,5\}$	$\{2,4,6\}$	$\{3,5,6\}$
4	$\{2,5,7\}$	$\{1,2,3\}$	$\{1,4,5\}$	$\{2,4,6\}$	$\{3,5,6\}$

Both strategy profile are Nash equilibrium

```
ceMinimization\x64\Release>LinkPreservingInterferenceMinimization.exe
{ 1 2 3}  { 1 2 3}  { 1 2 3}  { 1 2 3}  { 1 2 3}
{ 3 4 7}  { 1 2 3}  { 1 2 3}  { 1 2 3}  { 1 2 3}
{ 3 4 7}  { 2 4 6}  { 1 2 3}  { 1 2 3}  { 1 2 3}
{ 3 4 7}  { 2 4 6}  { 1 4 5}  { 1 2 3}  { 1 2 3}
{ 3 4 7}  { 2 4 6}  { 1 4 5}  { 1 6 7}  { 1 2 3}
{ 3 4 7}  { 2 4 6}  { 1 4 5}  { 1 6 7}  { 1 2 3}
```

Fig. Best-reply path in LPIM

Best-reply path(PP)

```
{ 1 2 3}  { 1 2 3}  { 1 2 3}  { 1 2 3}  { 1 2 3}
{ 1 4 5}  { 1 2 3}  { 1 2 3}  { 1 2 3}  { 1 2 3}
{ 1 4 5}  { 2 4 5}  { 1 2 3}  { 1 2 3}  { 1 2 3}
{ 1 4 5}  { 2 4 5}  { 3 4 5}  { 1 2 3}  { 1 2 3}
{ 1 4 5}  { 2 4 5}  { 3 4 5}  { 1 2 3}  { 1 2 3}

D:\user\Documents\Visual Studio 2017\project\Link-Preservi
```

Fig. Best-reply path in LPIM(PP)

Simulation

- Simulations on unit disk graphs
- Area: 1000 m X 1000 m = 1000,000 m^2
- Randomly placed n mesh nodes
- $r\_max = 3$
- Communication range = 200 m
- Total number of Channel: m

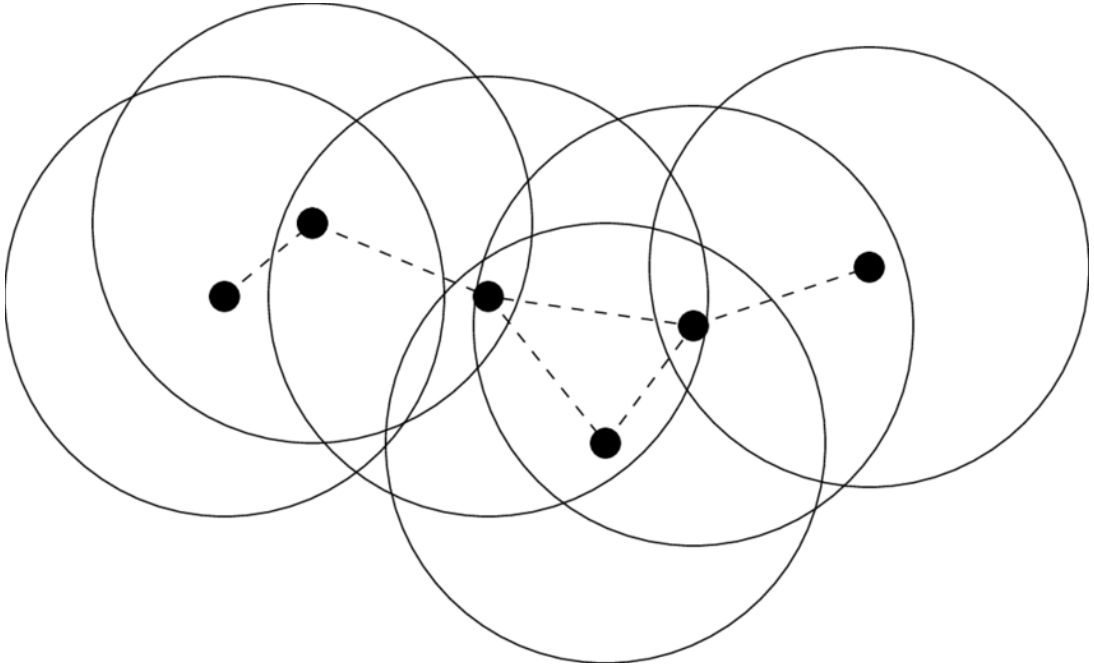
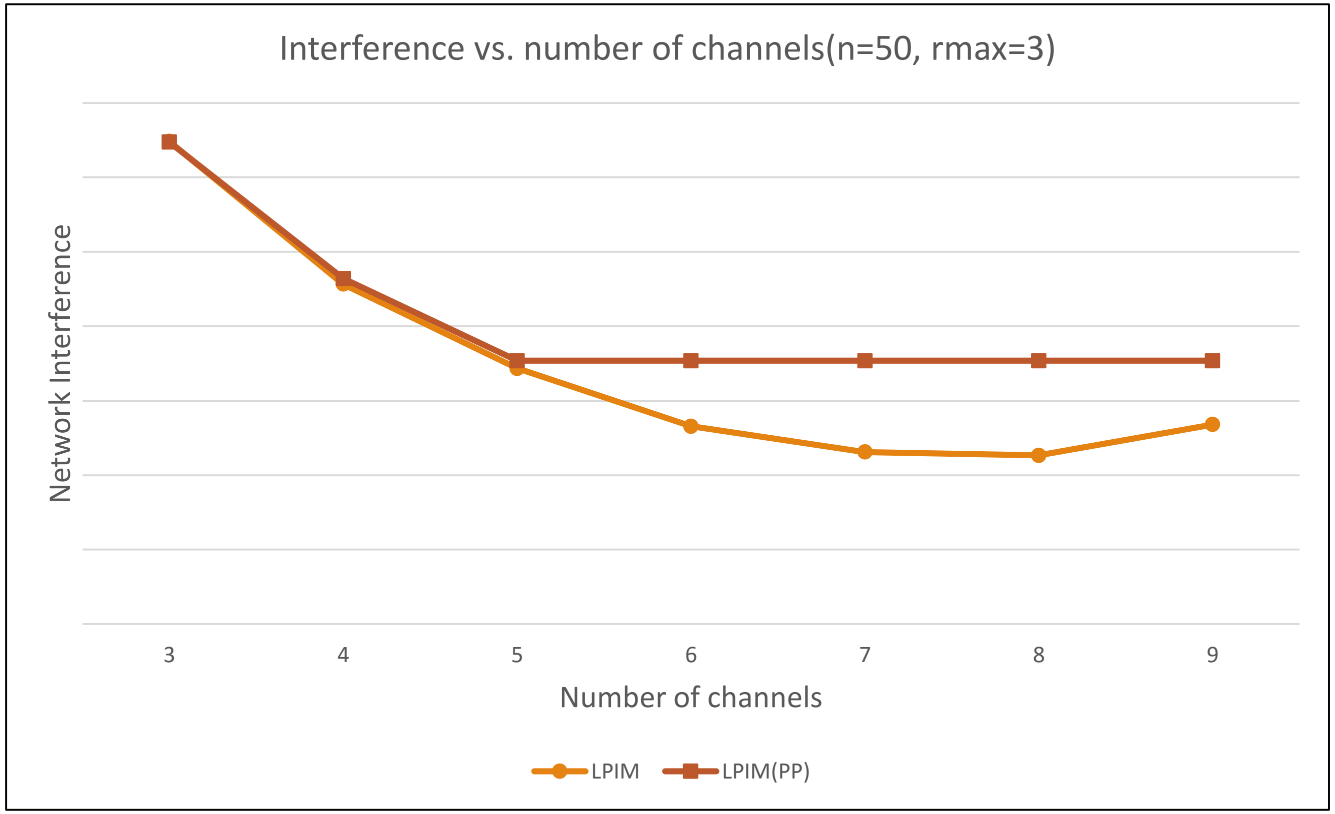


Fig. Unit disk graph model

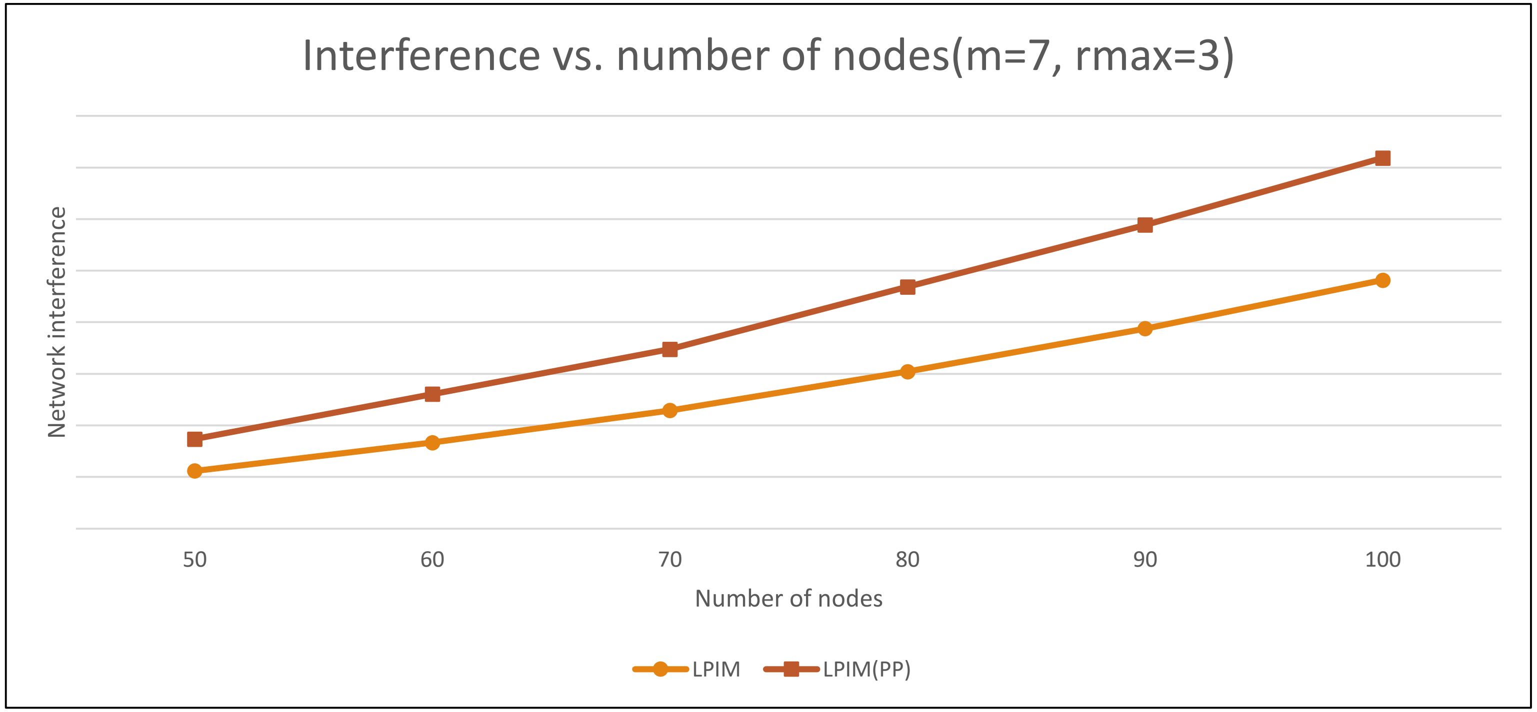
Simulation – interference vs. channels

- Node: 50
- r_{max}: 3
- Channel: 3-9
- lower is better



Simulation – interference vs. nodes

- Node: 50-100 (step 10)
- r_{max}: 3
- Channel: 7
- lower is better



Conclusions

- ✓A non-cooperative game design for the channel assignment problem in WMNs
- ✓First non-cooperative game approach that ensures link connectivity
- ✓Eventually enters a Nash equilibrium regardless of initial channel configuration
- ✓Link-preserving
- ✓This approach generally better than the other approaches when a moderate number of channels are available

Thank you for your attention