Equilibria for Broadcast Range Assignment Games in Ad-Hoc Networks

P. Crescenzi[1] M. Di Ianni[2] A. Lazzoni[1] P. Penna[3]
G. Rossi[2] P. Vocca[4]

[1]University of Florence

[2]University of Rome II

[3]University of Salerno

[4]University of Lecce

May 2005

Outline

1. Introduction
   - Ad-Hoc Networks
   - Model and assumptions
   - Related works
2. Our Contribution
   - Analytic Results
   - Experimental Results
3. Conclusion
   - Open Question

Ad-Hoc networks: main features

- `Lack of fixed infrastructure`: *self-organized* network with highly cooperative nodes
- `Lack of central authority`: *altruistic* behavior of the nodes cannot be assumed
- `Transmission power`:

$$P_v \geq d(v,t)^{\alpha} \times \gamma$$

where α is the distance-power gradient (usually, between 1 and 6) and $\gamma \geq 1$ is transmission quality parameter

Social behavior

- `Social cost`: the overall power consumption
- `Selfish behavior`: each station prefers to reduce its own costs
- `Cooperation via payments`
  - Consider n stations equally spaced on a line and the leftmost station s willing to perform a *broadcast* operation
  - A single-hop transmission would cost $O(n^\alpha)$ to s, while a multi-hop transmission would globally cost $O(n)$ ($O(1)$ to each station)
  - s may decide to “pay” the energy spent for forwarding the message

Managing the mobility

- `Using traces`
  - Advantages: realistic movement behavior
  - Disadvantages: confinement to a specific scenario, tracing of users is complicated
- `Mobility models`
  - Random way-point model, random walk, and Brownian motion: assume that each node moves freely and independently, and are based on rather simple assumptions regarding the movement behavior
  - Obstacle model: tries to take into account pathways and obstacles, and is based on the construction of the Voronoi diagram corresponding to the vertices of a set of polygonal obstacles

Broadcast Range Assignments

- `Range assignment`: function $r : S \to \mathbb{R}^+$, that specifies the *transmission range* of each station (that is, the maximum distance at which a station can transmit)
- `Transmission graph`: $G_r = (S, E_r)$, where $(v, t) \in E_r$ if and only if $d(v, t) \leq r(v)$
- `Broadcast range assignment`: G_r contains a directed spanning tree rooted at source station
- `Cost of BRA`:

$$\texttt{cost}(r) = \sum_{u \in S} r(u)^\alpha$$

BRA games and Nash equilibria

- `Station strategy`: choosing its own transmission range
- `Station benefit`: due, for example, to the implementation of the required connectivity or to the payments from other stations
- `Utility function`:

$$u_v(r) = b_v(r) - r(v)^\alpha$$

  (observe that it depends on the strategy of all stations)
- `Nash equilibrium`:

$$u_v(r) \geq u_v(r')$$

  for every v and every r' obtained from r by varying $r(v)$
  - ϵ-approximate if $\epsilon \cdot u_v(r) \geq u_v(r')$

Payment policies

- `Payment-free`: no payments are allowed (clearly, a broadcast range assignment will be a Nash equilibrium if at least one station is penalized)
- `Who is paid`
  - `Edge-payments`: only the last station in the path
  - `Path-payments`: all the stations in the path
- `How much is paid`
  - `No-profit`: the cost of station u is shared among all the stations using u
  - `Profit`: each station using u pays the cost of u
- Payment ϵ-approximate Nash equilibrium

$$p_v(r) \leq p_v(r')$$

for every v and every r' obtained from r by varying $r(v)$

Broadcast Range Assignment

- `Complexity`: NP-hard for all $\alpha > 1$ [Clementi et al., 2001] (trivially in P, if $\alpha = 1$)
- `MST-based algorithm`: 6-approximation algorithm, for $\alpha \geq 2$ (tight analysis) [Ambühl, 2005]
  - No approximation algorithm is known for $1 < \alpha < 2$
- `Random instances`: [Ephremides et al., 2000], [Klasing et al., 2004], [Penna and Ventre, 2004]
- `Other range assignments problems`: *strongly connected* communication graphs, bounded number of hops, stations located on the *d*-dimensional Euclidean space, for $d > 2$, more general settings considering non-geometric instances modeled by arbitrary weighted graphs, and symmetric wireless links

Nash equilibria and network design games

- `Network design games`: each station offers to pay an *arbitrary* fraction of the cost of building/maintaining a link of a network, and the corresponding link "exists" if and only if enough money is collected from all agents [Anshelevich et al., 2003-2004]
- `NDG and wireless networks`
    - Point-to-point and strong connectivity requirements [Eidenbenz et al., 2003]
    - Multicast games in general ad-hoc networks [Bilò et al., 2004]

Summary of the results

	Profit	**No-profit**
Edge-Payment	A P-time computable Nash equilibrium that is a 6-approximation of the optimum	
Path-Payment	A P-time computable payment ϵ-approximate Nash equilibrium that is a $6(1+\frac{2}{1-\epsilon})$-approximation of the optimum	A P-time computable payment 6-approximated Nash equilibrium that is a 6-approximation of the optimum

Algorithm for no-profit models

Computes a directed minimum spanning tree of S rooted at s. Then, every station, in turn, tries to decrease the amount of its payments

procedure findNE(S, s)
    $T_0 \leftarrow$ mst(S);
    compute T by rooting T_0 at s and by orienting all its edges towards the leaves;
    for $v \in S - \{s\}$ **do**
        $p_T(v) \leftarrow$ the sum of all payments due by v according to T and to the payment model;
    while T does not represent a Nash equilibrium **do** {
        choose $v \in S - \{s\}$;
        $m \leftarrow p_T(v)$;
        $T_2 \leftarrow T$;
        for $x \in S - \{s\}$ **and** x not belonging to the subtree of T rooted at v {
            let u be the father of v in T;
            $T_1 \leftarrow E(T) - \{(u,v)\} \cup \{(x,v)\}$
            if $p_{T_1}(v) < m$ **then**
                $m \leftarrow p_{T_1}(v)$;
                $T_2 \leftarrow T_1$;
        }
        if $p_T(v) < m$ **then**
            $T \leftarrow T_2$;
    }
    return T;

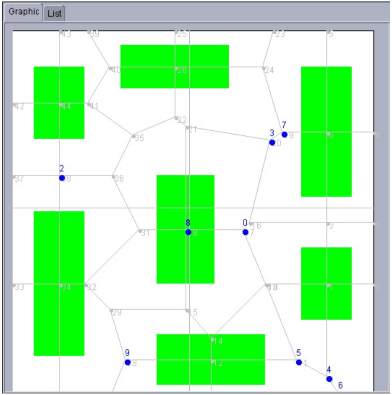
Convergence speed results: random instances

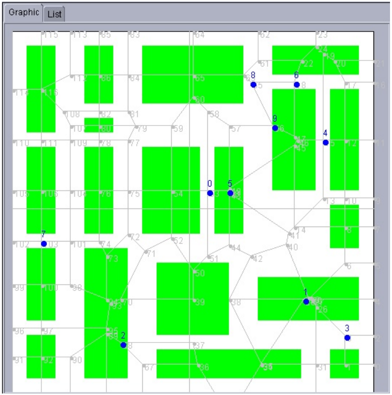
For each *n*, 1000 instances have been randomly generated according to the uniform distribution.

	1		2		3		4		5		6		...
n	*e*	*p*	*e*	*p*	*e*	*p*	*e*	*p*	*e*	*p*	*e*	*p*	...
10	40.9	12.0	50.9	69.5	7.5	16.8	0.6	1.5	0.0	0.1	0	0	...
100	0	0	46.4	5.2	48.9	65.9	4.6	25.4	0.1	3.3	0	0.2	...
200	0	0	24.1	0.1	67.9	50.5	7.8	40.8	0.2	7.2	0	1.3	...
300	0	0	10	0	77.2	33.9	12.3	54	0.4	9.6	0.1	1.7	...
400	0	0	4.4	0	79.6	23.8	15.5	55.4	0.5	16.5	0	3.7	...
500	0	0	3.1	0	76.9	15.5	19.1	61.6	0.9	17.8	0	3.5	...
1000	0	0	0.1	0	62.4	2.6	34.7	58.1	2.7	30.3	0.1	6.9	...
1500	0	0	0	0	50.9	1.3	46.3	41.8	2.7	45.3	0.1	10.4	...
2000	0	0	0	0	41.3	0.2	54	33.4	4.3	45.7	0.4	14.9	...

For a negligible number of instances the required rounds are in the interval $7 - 12$. No instance require more than 13 rounds.

The two scenarios of the obstacle mobility model



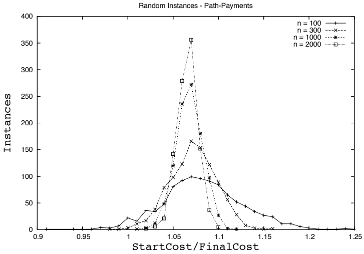


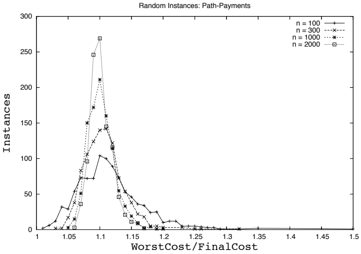
Convergence speed results: mobility model instances

For each *n*, 100 instances have been generated according to the obstacle mobility model.

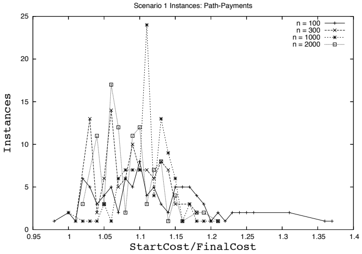
		1		2		3		4		5		6		7		8	
n		*e*	*p*	*e*	*p*	*e*	*p*	*e*	*p*	*e*	*p*	*e*	*p*	*e*	*p*	*e*	*p*
10	Scen. 1	45	15	50	66	5	19	0	0	0	0	0	0	0	0	0	0
	Scen. 2	44	12	50	72	5	15	1	1	0	0	0	0	0	0	0	0
100	Scen. 1	1	0	75	42	20	50	4	8	0	0	0	0	0	0	0	0
	Scen. 2	1	0	72	8	26	67	1	20	0	4	0	1	0	0	0	0
200	Scen. 1	0	0	65	39	28	55	6	5	1	1	0	0	0	0	0	0
	Scen. 2	1	0	61	4	33	65	5	27	0	3	0	1	0	0	0	0
300	Scen. 1	0	0	70	37	25	58	3	5	2	0	0	0	0	0	0	0
	Scen. 2	0	0	65	4	28	64	6	25	1	7	0	0	0	0	0	0
400	Scen. 1	0	0	67	29	27	56	6	14	0	1	0	0	0	0	0	0
	Scen. 2	0	0	60	1	35	55	4	39	1	4	0	1	0	0	0	0
500	Scen. 1	0	0	93	22	7	64	0	13	0	1	0	0	0	0	0	0
	Scen. 2	0	0	53	1	46	57	1	35	0	7	0	0	0	0	0	0
1000	Scen. 1	0	0	69	28	23	66	8	5	0	0	0	1	0	0	0	0
	Scen. 2	0	0	88	0	12	51	0	39	0	9	0	0	0	0	0	1
1500	Scen. 1	0	0	91	20	7	76	1	4	1	0	0	0	0	0	0	0
	Scen. 2	0	0	66	1	33	45	1	41	0	13	0	0	0	0	0	0
2000	Scen. 1	0	0	68	69	22	26	8	5	2	0	0	0	0	0	0	0
	Scen. 2	0	0	3	0	56	1	41	70	0	25	0	3	0	1	0	0

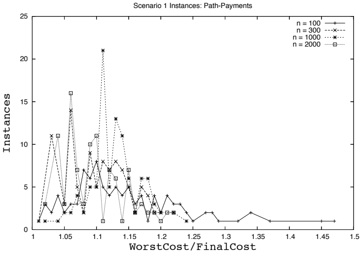
Quality of the solution: random instances





Quality of the solution: mobility model instances





No-profit edge-payment model

There exists a polynomial time computable (approximated) Nash equilibrium that is an approximation of the optimal solution