Nash Equilibrium and Stability in Network Selection Games

Mohit Hota(✉) and Sanjiv Kapoor

Illinois Institute of Technology, Chicago, IL 60616, USA
mhota@hawk.iit.edu, kapoor@iit.edu

Abstract. We study network selection games in wireless networks. Each client selects a base station to maximize her throughput. We utilize a model which incorporates client priority weight and her physical rate on individual Base Stations. The network selection behavior considered is atomic, implying that a client connects to exactly one Base Station.

We formulate a non-cooperative game and study its convergence to a pure Nash equilibrium, if it exists, or prove non-existence otherwise, and present algorithms to discover pure Nash equilibrium for multiple cases.

1 Introduction

Enhancements in wireless connectivity involve the ability to choose the best available network connection. This is evident in recently put forth proposals and implementations, where a wireless device selects the provider (base station) and type of access (Wi-Fi, WiMax or GPRS schemes, femto etc.) which permits the best speed or rate, on the basis of location and availability (Google Fi services is an example). Moreover, priority weights, ensuring individual user priorities according to fixed agreements, are being increasingly suggested by providers. These priority agreements would serve to provide Quality of Service (QoS).

Throughput analysis of accessing heterogeneous radio technologies has been studied in [1–4] where clients utilize information from the access networks (termed RAT or RAN) to determine the choice of network access points (also referred to as Base stations). The standard approach is to consider the clients to be autonomous agents. Alternately, rules can be imposed on the RAN clients to regulate traffic.

The key decision for users in such a model is the selection of the network access point. The system of autonomous agents competing for a limited set of resources gives rise to a congestion game. Such a system leads to the formation of a complex system model where a user (client) would select, based on priority weights, a provider's base station and an instantaneous PHY rate provided by the base station, as has been utilized in [4], depending on current physical conditions like base station load, location or even radio bandwidth congestion. All such factors would determine the throughput that a client would be able to obtain on a base station.

Every client seeks to maximize her own total throughput without regard for how other clients are affected by her actions and thus, we formulate a game

L. Duan et al. (Eds.): GameNets 2017, LNICST 212, pp. 40–50, 2017.
DOI: 10.1007/978-3-319-67540-4_4

where each client behaves selfishly to maximize her throughput. Such a game-theoretic model has previously been studied also in [3,4]. Additional throughput or utility models can be found in the survey paper [1]. We term the above model as an *atomic throughput game*, also termed as a RAN selection game in previous papers. These previous papers leave a number of unresolved issues regarding the existence of pure Nash equilibrium (interchangeably, for simplicity, referred to as *Nash equilibrium* in this paper) in the defined games.

The RAN selection game falls into the class of congestion games. Atomic congestion games with a cost function dependent on the number of clients occupying a resource were first studied in [5] with consequent work on client specific utilities in [6]. The computational complexity of determining Nash equilibrium in these games was studied in [7], where they showed that atomic congestion games with arbitrary cost functions are $\mathcal{PLS}$-Complete. Wireless congestion games and cost network specific cost functions have been studied in [2,8,9]. Unlike the prior studies, the model in [3,4] utilizes the throughput itself as a metric of performance. Additional game theoretic models using evolutionary games [10,11] have been studied but are not relevant as these models correspond to non-atomic versions of the game with large number of users, each with infinitesimal impact.

In this paper we consider the RAN selection game:

- We first show that pure Nash equilibrium does not always exist for the RAN selection game with non-uniform weights and rates, implying that the system might not stabilize at all. This resolves a question left unanswered in [3,4] where Aryafar et al. alluded to such a result. Resolving the existence and complexity of Nash equilibrium is considered important as it characterizes the convergence towards stability of such autonomous systems. We consider interesting practical cases and prove that pure Nash Equilibrium always exists if the user has uniform or identical priorities over all base stations. We provide an ϵ-approximate Nash Equilibrium algorithm which runs in polynomial time in this case, as well as a polynomial algorithm to compute pure Nash equilibrium when, additionally, rates are uniform.
- We consider *priority regulated games*, where priority can be used to regulate the throughput rate and disprove a conjecture from [4] which states that a Nash Equilibrium always exists in games where the priority weights is a polynomial function of the rates. On the positive side, we provide a simple fairness rule that ensures convergence to a solution which is stable, i.e. no further improvements are possible. This stable point may not be a Nash equilibrium of the original strategy space but the system is stable under the rule.

1.1 Network Model

The wireless selection problem has a set of clients P accessing a set of wireless access points, which we refer to as Base Stations, K. The base stations represent the range of wireless access points, Wi-Fi and GPRS etc. Each client accesses a base station and negotiates a rate of access. The wireless selection problem is that of scheduling clients to base stations to optimize throughput. We represent the wireless selection problem by a network model where the underlying graph is a bipartite graph represented by $G = (P, K, E)$. The clients are represented by one (independent) vertex set P and the set of Base Stations (BS), K, the second (independent) vertex set. The set of edges E represents the base stations available between the clients in the set P and the base stations. An edge $e = (i, k), i \in P, k \in K$ exists if and only if the client i can access base station k. Each client i is characterized by two parameters, the weight $\phi_{i,k}$ that provides her a priority on a base station $k \in K$ and the PHY rate $R_{i,k}$ that she can obtain on that base station k. The throughput that the clients acquire from the base station k is dependent on the other clients that utilize the base station.

Throughput Model. The throughput model we use is based on the model in [3,4] that defines the throughput client i obtains on base station k as

$$\omega_{i,k} = \frac{\phi_{i,k}}{\sum_{j \in s(k)} \frac{\phi_{j,k}}{R_{j,k}}}$$

where $s(k)$ is the set of clients that are currently accessing base station k.

Since each client has an independent choice of scheduling her traffic on the available base stations, the rational autonomous decisions of the client can be modeled by a game:

A **Throughput Game** is denoted by $TG(P, K, \phi, R)$ where P are the clients (clients) in the game, K is the set of base stations, $\phi : P \times K \to \mathbb{R}^+$ is a function representing the priorities (weights) of clients on the base stations K, $R : P \times K \to \mathbb{R}^+$ is a function representing the rates the clients have obtained on the base stations. In a throughput game, a client selects one base station to transfer data, and given the selection of the other clients, selfishly selects the base station on which she receives maximum throughput. We also consider restricted models defined below:

1. Different types of traffic require a priority that is dictated by their type, e.g., video traffic requires a certain priority level, and do not depend on the base stations, leading to **Uniform Priority Throughput** games, denoted by $TG_P(P, K, \phi, R)$, where the priority levels are independent of the base stations, i.e. $\phi_{i,k} = \phi_{i,k'} = \phi_i, \forall k, k'$.
2. Furthermore, devices may only be able to communicate at a particular rate, leading to **Uniform Rate Throughput** games, denoted by $TG_R(P, K, \phi, R)$, where the rates achieved by a client i is independent of the base stations, i.e. $R_{i,k} = R_{i,k'} = R_i, \forall k, k'$.

We define the *Load* on a base station k, when a set $s(k)$ of clients are scheduled on base station k, to be $G_{s(k)} = \sum_{j \in s(k)} \frac{\phi_{j,k}}{R_{j,k}}$ where the contribution of a client j to the load is $\frac{\phi_{j,k}}{R_{j,k}}$. Note that when i is the only client on a base station k, she gets throughput $\omega_{i,k} = R_{i,k}$.

A *pure Nash equlibrium* is defined as an assignment of clients to base stations such that no client can unilaterally improve her throughput by switching to a different base station.

2 Nash Equilibrium in Throughput Games

We first resolve the question of existence of Nash equilibrium in throughput games, left unanswered in [4].

Theorem 1. *There exists a throughput game, $TG(P, K, \phi, R)$, for which there is no pure Nash equilibrium.*

Proof. In order to determine an example for a game, a Monte Carlo algorithm was used to generate the base station rates and priorities. In a game involving 3 clients and 3 base stations, the following values of ϕ and R present a scenario such that no configuration of assignments result in any client being satisfied on the base station she occupies. The matrix Φ represents the priorities $\phi_{i,k}$ and the matrix R represents the rates $R_{i,k}$.

$$\Phi = \begin{bmatrix} & L_1 & L_2 & L_3 \\ P_1 & 9.8 & 1.6 & 5.1 \\ P_2 & 8.1 & 0.2 & 8.6 \\ P_3 & 4.6 & 3.9 & 8.8 \end{bmatrix} \quad R = \begin{bmatrix} & L_1 & L_2 & L_3 \\ P_1 & 98.3 & 80.8 & 12.6 \\ P_2 & 27.6 & 32.6 & 21.2 \\ P_3 & 65.8 & 14.9 & 9.8 \end{bmatrix}$$

Any configuration in this instance results in cycling. To illustrate one such cycle, consider an initial configuration $(2, 1, 1)$ denoting that client 1 is on link 2, client 2 is on link 1 and client 3 is on link 1. Client 3 can obtain a higher throughput than what she already has by moving from link 1 to link 2, and does so. The configuration is thus $(2, 1, 2)$, following which, Client 1 then switches to link 1 to obtain a higher throughput, yielding the configuration $(1, 1, 2)$. The configuration keeps changing and eventually cycles back to a previous state. The cycle is shown in Fig. 1.

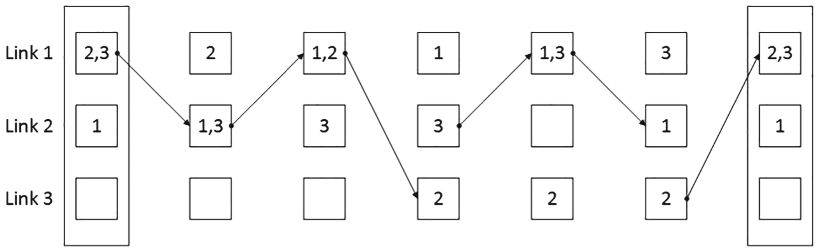

Fig. 1. Client cycling in a throughput game where Nash equilibrium does not exist

2.1 Nash Equilibrium in Uniform Priority Throughput Games

Since we have shown that a Nash equilibrium may not always exist in throughput games, we study its existence in Uniform Priority Throughput Games.

Theorem 2. *Every instance of a Uniform Priority Throughput game, $TG_U(P, K, \phi, R)$ (where $\phi_{ik} = \phi_{il} = \phi_i$), has a pure Nash equilibrium.*

Proof. Given an assignment of clients to base stations, characterized by specifying $s(k)$, the set of clients on base station k, we first establish an inequality, which provides a condition under which a client switches to another base station. Consider a client i who chooses to make a move from k to k' to get a higher throughput. For this move to occur, we must have

$$\frac{\phi_i}{\sum_{j \in s(k')} \frac{\phi_j}{R_{j,k'}} + \frac{\phi_i}{R_{i,k'}}} > \frac{\phi_i}{\sum_{j \in s(k)} \frac{\phi_j}{R_{j,k}}} \tag{1}$$

The load on base station k is $\sum_{j \in k} \frac{\phi_j}{R_{j,k}} = G_{s(k)}$, as defined in the network model. Inequality (1) then becomes

$$G_{s(k')} + \frac{\phi_i}{R_{i,k'}} < G_{s(k)} \text{ or equivalently, } G_{s'(k')} < G_{s(k)} \tag{2}$$

where $G_{s'(k')} = G_{s(k')} + \frac{\phi_i}{R_{i,k'}}$ is the load of base station k after i moves.

This expresses the fact that when client i moves from base station k to k' to increase her throughput, the load on base station k' after the move must be less than the pre-move load of k, otherwise client i would have had no incentive to move.

We then consider a vector $L = \{\{G_{s(k_1)}, \cdots, G_{s(k)}, G_{s(k')}, \cdots, G_{s(k_K)}\}$ s.t. $\{G_{s(k_1)} > \cdots > G_{s(k)} > G_{s(k')} > \cdots > G_{s(k_K)}\}\}$, i.e., k_1 is the base station with the highest load and k_K is the base station with the smallest load. Our claim is that the load vector L, which is the sorted loads of base stations, decreases (in lexicographic ordering) for every move that client i makes to increase her throughput. We prove our claim below:

We define the position (increasing from left to right) of the load of a base station k in a load vector L by $\pi_L(k)$. Let L be the load vector before client i moves and L' be the load vector after i has moved. Note that $\pi_L(k) \leq \pi_{L'}(k)$. There are two cases:

1. $\pi_{L'}(k) < \pi_{L'}(k')$: Since $G_{s'(k)} < G_{s(k)}$, the lexicographic value of L' will be less than L.
2. $\pi_{L'}(k') < \pi_{L'}(k)$: Since $G_{s'(k')} < G_{s(k)}$ from inequality (2), the lexicographic value of L' will be less than L.

Each of the cases indicate that the vector L will lexicographically decrease for every move that improves the throughput of a client. To show that the minimum load on each base station is lower bounded by a positive value, let $\phi_{\min} = \min_{i \in P} \phi_i$ and $R_{\max} = \max_{i \in P, k \in K} R_{i,k}$. The minimum load on each base station, which is occupied by at least one client, is then at least $\frac{\phi_{\min}}{R_{\max}}$. Therefore, the *uniform throughput priority game* will always converge to a pure Nash equilibrium.

2.2 ϵ-Approximate Nash Equilibrium for Uniform Priority Models

Based on the proof of Theorem 2, we observe that Nash equilibrium can be determined by allowing clients to improve their throughput by switching base stations. The number of improvement steps of the lexicographic ordering in vector L is upper bounded by $O\Big(|P| \times \frac{\sum_{i\in P,k\in K}\frac{\phi_i}{R_{i,k}}}{\delta}\Big)$, where δ is the minimum change in lexicographic value caused by a switch. When the values of ϕ_{ij} and R_{ij} are integers, $\delta \geq \frac{1}{R^2_{max}}$. Similar bounds can be established for rationals. Finding a polynomial time algorithm for determining Nash equilibrium appears difficult. Therefore, we specialize certain parameters to obtain faster algorithms for achieving a near-Nash equilibrium state.

We first define an **ϵ-approximate Nash equilibrium**: A throughput game is at an ϵ-approximate Nash equilibrium if for every client, a switch to another base station improves her throughput by a factor of at most $(1+\epsilon)$.

Algorithm 1. Finding ϵ-approximate Nash equilibrium in $TG_P(P, K, \phi, R)$

1: Start with any random assignment of clients, where the set of clients on base station k is given by $s(k), \forall k$.
2: **while** $\exists$ clients who can improve their throughput by a factor of $(1+\epsilon)$ by switching to another base station **do**
3: Select client i s.t. $(i, k'_i) = \arg\min_{(i,k'(i))}(G_{s'(k_i)} + G_{s'(k'_i)} + \sum_{k\in K, k\neq k_i, k'_i} G_{s(k)})$ where i is assigned to k_i and moves to k'_i, and $s'(k)$ denotes the set of clients on k after the movement of i.
4: Move i from k_i to k'_i.
5: **end while**

Theorem 3. *Given an instance of a Uniform Priority Throughput Game, Algorithm 1 finds a ϵ-approximate Nash equilibrium in time $O(t \times P \times K)$, where $t = \log_{1+\epsilon}\frac{R_{\max}}{\phi_{\min}} + \log_{1+\epsilon}\sum_{i\in P}(\frac{\phi}{R})_{i_{\max}}$ is the upper bound on the number of steps a client moves, where $(\frac{\phi}{R})_{i_{\max}} = \max_{k,k'\in K}\frac{\phi_k}{R_{k'}}$.*

Proof. We have already established that a switch implies a lexicographic decrease of vector L, as shown in Theorem 2, and therefore, assured that the approximate Nash equilibrium is achieved by the algorithm.

To calculate the time complexity, we provide an upper bound t on the number of times a client would have to switch to reach her final choice of base station. Since a client i can only move if she gains a factor of $(1+\epsilon)$ on her current throughput, t can be calculated by comparing the lower bound and upper bound, termed $\omega_{i_{\min}}$ and $\omega_{i_{\max}}$, respectively, on her possible throughput.

We obtain the value of $\omega_{i_{\max}}$ for a client i by placing her alone on the base station where she has the maximum PHY rate $R_{\max} = \max_{i\in P,k\in K} R_{i,k}$, since the load on the base station increases as soon as she shares a base station with another client. Therefore, $\omega_{i_{\max}} = R_{\max}$. Similarly, we get $\omega_{i_{\min}}$ by placing the client with the minimum ϕ_i ($\phi_{\min}$) with all the other clients in the game, and then by selecting the maximum load contribution of each client, $(\frac{\phi}{R})_{i_{\max}} = \max_{k\in K}(\frac{\phi_i}{R_{i,k}})$, giving a total load of $\sum_{i\in P}(\frac{\phi}{R})_{i_{\max}}$. Therefore,

$\omega_{i_{min}} = \frac{\phi_{min}}{\sum_{i \in P}(\frac{\phi}{R})_{i_{max}}}$. We then obtain the bound on t by using the fact that when the algorithm terminates, the maximum throughput is at most $\frac{\omega_{i_{max}}}{\omega_{i_{min}}}$. Therefore, $(1+\epsilon)^t \leq \frac{\omega_{i_{max}}}{\omega_{i_{min}}}$, which implies $t \leq \log_{1+\epsilon} \frac{R_{\max}}{\phi_{\min}} + \log_{1+\epsilon} \sum_{i \in P}(\frac{\phi}{R})_{i_{\max}}$ which then leads to our result.

2.3 Finding Equilibrium in Uniform Priority-and-Rate Games

While a Nash equilibrium is not easily (in polynomial time) found in Uniform Priority games, we show that by altering the uniform priority game to include uniform rates, denoted by $TG_{P,R}(P, K, \phi, R)$, a Nash equilibrium can be discovered by a polynomial time algorithm.

Algorithm 2. Finding a Pure Nash equilibrium in $TG_{P,R}(P, K, \phi, R)$

Sort clients in non-increasing order of $\frac{\phi_i}{R_i}$
for Client $i = 1 \cdots |P|$ **do**
  $k_i = \arg\min_{k \in K}(G_{s(k)} + \frac{\phi_i}{R_i})$
  Assign client i to base station k_i.
end for

Theorem 4. *Given an instance of a Uniform Priority-and-Rate Throughput game $TG_{P,R}(P, K, \phi, R)$, Algorithm 2 correctly finds a pure Nash equilibrium in time $O(|P|(|K| + log|P|))$.*

Proof. The algorithm assigns a new client to a base station and ensures that the client gets maximum throughput, given the current system configuration. For our algorithm to be correct, an addition of a new client to the system should not induce any moves.

First, we use contradiction to show that after addition of a new client to a base station, other clients from that base station do not have an incentive to move to other links. Let $G_{s(k)}$ and $G_{s(k')}$ be the loads of base stations k and k' respectively before either client i or i' have been introduced. Suppose that on addition of the i^{th} client to base station k, client i' wants to switch from using base station k to k', implying inequality (3),

$$\frac{\phi_{i'}}{G_{s(k)} + \frac{\phi_i}{R_i} + \frac{\phi_{i'}}{R_{i'}}} < \frac{\phi_{i'}}{G_{s(k')} + \frac{\phi_{i'}}{R'_i}} \Rightarrow G_{s(k')} < G_{s(k)} + \frac{\phi_i}{R_i} \quad (3)$$

and from the fact that client i was previously assigned to base station k, we have inequality (4)

$$\frac{\phi_i}{G_{s(k)} + \frac{\phi_i}{R_i} + \frac{\phi_{i'}}{R_{i'}}} > \frac{\phi_i}{G_{s(k')} + \frac{\phi_i}{R_i}} \Rightarrow G_{s(k')} > G_{s(k)} + \frac{\phi_{i'}}{R_{i'}} \quad (4)$$

From the sorting performed $\frac{\phi_i}{R_i}$ in step 1, we have

$$\phi_{i'}/R_{i'} > \phi_i/R_i \tag{5}$$

Using the above inequalities, we get the following contradiction

$$G_{s(k')} < G_{s(k)} + \frac{\phi_i}{R_i} < G_{s(k)} + \frac{\phi_{i'}}{R_{i'}} \text{ and } G_{s(k')} > G_{s(k)} + \frac{\phi_{i'}}{R_{i'}} > G_{s(k)} + \frac{\phi_i}{R_i}$$

In the second case, we show that when a client p is added to a base station k, clients p' from other base stations k'(say) do not move to k. Prior to adding p, p' had chosen base station k' over base station k.

$$\therefore G_{s(k)} + \frac{\phi_{p'}}{R_{p'}} > G_{s(k')} + \frac{\phi_{p'}}{R_{p'}} \Rightarrow G_{s(k)} > G_{s(k')} \tag{6}$$

If it were beneficial for p' to switch to base station k now, the following inequality must be true

$$G_{s(k)} + \frac{\phi_p}{R_p} + \frac{\phi_{p'}}{R_{p'}} < G_{s(k')} + \frac{\phi_{p'}}{R_{p'}} \Rightarrow G_{s(k)} < G_{s(k')} \tag{7}$$

which is contradictory to inequality (6), thus proving our claim.

The running time of step 1 is $O(|P| \log |P|)$ for sorting the values. Step 2 has $|P|$ iterations of steps 3 and 4, which perform $|K|$ comparisons, thus giving a total running time of $O(|P|(\log |P| + |K|))$.

3 Nash Equilibrium in Rate-Dependent Priority Throughput Games

Throughput games where the priorities are a function of the rates have been investigated in [4]. In this model, $\phi_{i,k} = R_{i,k}^{\beta}$, and the game is denoted by $TG_\beta(P, K, \phi, R)$. Properties of this model have been studied in [4] where it was conjectured that Nash equilibrium exists for any value of β. We first disprove this conjecture and then provide a set of rules under which we prove that a stable point exists.

Theorem 5. *There exists an instance of a Rate-dependent Throughput Game,* $TG_\beta(P, K, \phi, R)$*, with* $\phi_{ij} = R_{ij}^{-1.5}$ *for which a pure Nash equilibrium does not exist.*

Proof. Similar to Theorem 1, we use a Monte Carlo algorithm to obtain the values of ϕ_{ij} and R_{ij} where $\phi_{ij} = R_{ij}^{\beta}$ and $\beta = -1.5$. The matrix R is

$$R = \begin{bmatrix} & L_1 & L_2 & L_3 \\ P_1 & 8.719 & 3.755 & 4.927 \\ P_2 & 5.802 & 1.361 & 5.783 \\ P_3 & 4.824 & 1.094 & 4.643 \\ P_4 & 3.340 & 9.648 & 8.743 \\ P_5 & 2.818 & 9.543 & 4.325 \end{bmatrix}$$

Values of ϕ can be generated using matrix R and $\beta = -1.5$. In fact, such examples were found for multiple values of β where $\beta < 0$. We illustrate an instance of a cycle of configurations in Fig. 2. Each configuration of the above instance results in similar cycles, yielding a system where no pure Nash equilibrium exists.

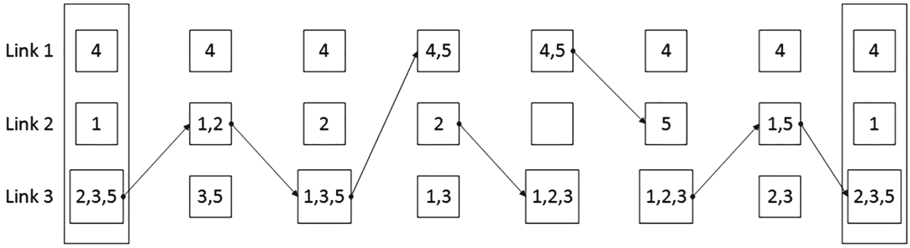

Fig. 2. Client cycling in a rate dependent throughput game where Nash equilibrium does not exist

3.1 Conditions for Convergence to a Stable Point

Having established that Nash equilibrium need not always exist, we now establish a protocol that ensures convergence to a stable point, thus preventing thrashing in the system.

Fair-Movement Protocol:

The following rules shall apply:

1. When a new client joins a system, she will be automatically assigned to the base station she has the highest rate on.
2. For every client, say i, switching from base station k to k' is permitted, only if $R_{i,k'} \leq R_{i,k}$. This is termed as the *Fair Movement Rule*

The purpose to the *Fair Movement Rule* is that since a client has already sought to reject a base station she was assigned a higher rate on, she must not be allowed to act selfishly with respect to her base rate $R_{i,k}$ and prevent the system from stabilizing.

Theorem 6. *Under the* **Fair-Movement Rule***, every Rate-dependent Throughput Game has a stable point; i.e., no client gains by unilaterally changing to a different assignment.*

Proof. Consider a vector L of loads on base stations $L = \{Gs_{(k_1)}, Gs_{(k_2)}, \cdots, Gs_{(k_K)}\}$ s.t. $\{G_{s(k_1)} > ... > G_{s(k)} > G_{s(k')} > ... > G_{s(k_K)}\}\}$. Note that the load of a base station is now given by $G_{s(k)} = \sum \frac{1}{R_{i,k}^{1-\beta}}$

Now, client i moves from base station k to k' ($G_{s(k)}$ is inclusive of client i) when she gets a higher throughput on k',

$$\frac{Gs_k}{R_{i,k}^{\beta}} > \frac{1}{R_{i,k'}^{\beta}}(Gs_{k'} + \frac{1}{R_{i,k'}^{1-\beta}}) \tag{8}$$

Using rule 2 of the Fair-Movement Protocol, we have $R_{i,k}^{\beta} - R_{i,k'}^{\beta} > 0$, therefore

$$\frac{1}{R_{i,k'}^{\beta}}(Gs_{k'} + \frac{1}{R_{i,k'}^{1-\beta}}) > \frac{1}{R_{i,k}^{\beta}}(Gs_{k'} + \frac{1}{R_{i,k'}^{1-\beta}}) \tag{9}$$

So, from (8) and (9), $\frac{1}{R_{i,k}^{\beta}}Gs_k > \frac{1}{R_{i,k}^{\beta}}(Gs_{k'} + \frac{1}{R_{i,k'}^{\beta}}) \Rightarrow Gs_k > Gs_{k'} + \frac{1}{R_{i,k'}^{1-\beta}}$, implying that the vector L decreases in lexicographic ordering every time a client i switches from base station k to k'. Thus, the game will be stable.

4 Conclusions and Acknowledgements

This paper has presented results for pure Nash equilibrium in a wireless game model where throughput has been used as the measure of the payoff. It would be of further interest to include link access costs, client budgets and general utility functions in the model.

The research was supported in part by NSF grant: CCF-1451574.

References

1. Kuo, G.S.: Mathematical modeling for network selection in heterogeneous wireless networks - a tutorial. IEEE Commun. Surv. Tutorials **15**(1), 271–292 (2013)
2. Malanchini, I., Cesana, M., Gatti, N.: Network selection and resource allocation games for wireless access networks. IEEE Trans. Mob. Comput. **12**(12), 2427–2440 (2013)
3. Aryafar, E., Keshavarz-Haddad, A., Wang, M., Chiang, M.: Rat selection games in hetnets. In: INFOCOM, pp. 998–1006. IEEE (2013)
4. Monsef, E., Keshavarz-Haddad, A., Aryafar, E., Saniie, J., Chiang, M.: Convergence properties of general network selection games. In: INFOCOM, pp. 1445–1453. IEEE (2015)
5. Rosenthal, R.W.: A class of games possessing pure-strategy nash equilibria. Int. J. Game Theory **2**, 65–67 (1973)
6. Milchtaich, I.: Weighted congestion games with separable preferences. Games Econ. Behav. **67**(2), 750–757 (2009)
7. Fabrikant, A., Papadimitriou, C., Talwar, K.: The complexity of pure nash equilibria. In: Proceedings of the Thirty-Sixth Annual ACM Symposium on Theory of Computing, pp. 604–612. ACM (2004)
8. Tekin, C., Liu, M., Southwell, R., Huang, J., Ahmad, S.H.A.: Atomic congestion games on graphs and their applications in networking. IEEE/ACM Trans. Netw. **20**(5), 1541–1552 (2012)

9. Ibrahim, M., Khawam, K., Tohm, S.: Congestion games for distributed radio access selection in broadband networks. In: GLOBECOM, pp. 1–5. IEEE (2010)
10. Shakkottai, S., Altman, E., Kumar, A.: Multihoming of users to access points in wlans: a population game perspective. IEEE J. Sel. Areas Commun. **25**(6), 1207–1215 (2007)
11. Niyato, D., Hossain, E.: Dynamics of network selection in heterogeneous wireless networks: an evolutionary game approach. IEEE Trans. Veh. Technol. **58**(4), 2008–2017 (2009)