A Game-Theoretic Approach for Optimal Distributed Cooperative Hybrid Caching in D2D Networks

Yuli Zhang, *Student Member, IEEE*, Yuhua Xu, *Member, IEEE*, Qihui Wu, *Senior Member, IEEE*, Xin Liu, Kailing Yao, and Alagan Anpalagan, *Senior Member, IEEE*

***Abstract*—The distributed cooperative hybrid caching problem based on content-awareness in device-to-device networks is studied in this letter. Besides caching from the base station, nodes also can cache files from nearby nodes. We also consider the content similarity between caching nodes, which would reduce the cost further through caching similar traffic from one source cooperatively. We model the cost reducing problem as a local cooperative game, and prove it to be an exact potential game, which has at least one pure Nash equilibrium (NE). Fortunately, the potential function is just the aggregate cost of the network, which means the NE point minimizes the total cost. We modified the log-linear learning algorithm and designed a half-fixed action to reduce the strategy space, and with random action to pursue better performance. The simulation results show that the modified log-linear learning algorithm achieves better performance compared with other algorithms, and the content-aware hybrid caching reduces the cost.**

***Index Terms*—Cooperative hybrid cache, potential game, content-awareness, log-linear learning algorithm.**

I. Introduction

THE GROWTH of mobile traffic data is rapidly increasing due to the explosion of smart phones. Many technologies have been proposed to achieve high channel rates, improve the spectrum efficiency and increase network capacity. Besides traditional technologies, caching has drawn researchers' attention in recent years due to the high utilization of popular data and good quality of service.

While providing a possible way to solve heavy traffic data, caching also brings many challenging problems. Taghizadeh *et al.* [1] introduced cooperative caching policies for minimizing electronic content provisioning cost in social wireless networks and minimized content provisioning cost in networks with demands. The energy efficiency of caching was also studied through a game-theoretic approach in [2]. Guo *et al.* [3] maximized the probability of successful caching and formulated the problem as a weighted sum utility maximization problem. Das and Abouzeid [4] considered cooperation between primary and secondary users in shared spectrum radio networks via caching. Some other researchers also focused on the caching cost problem. Chen *et al.* [5] studied the relationship between offloading gain of the system and energy cost of caching. And Araldo *et al.* [6] designed a cost-aware cache decision to reduce cost. A new peer-to-peer cooperative caching scheme to minimize the load on the infrastructure was proposed in [7].

Among these works, the most common problem is deciding which file should be cached. Besides, to decide where files should be cached is also an interesting problem, especially considering the content-aware relationship among caching nodes. Generally, the caching contents are related to some popular issues, which brings about the similarity between two caching nodes. However, the traditional cooperative caching did not focus on the content-aware issue. The cost will be smaller when two nodes are caching some similar files from one source together, compared with two different sources. Our previous work focused on the group buying in D2D networks through an overlapping coalition formation game approach [8]. But the basic cost and sharing cost among nodes and the influence of multiple file caching were not considered. Furthermore, caching by each file independently may also bring about more cost, due to ignoring the files content relationship among caching nodes. Therefore, designing a better caching strategy needs to be studied, which is based on the content-awareness to improve the caching efficiency and reduce cost.

In this letter, we consider a hybrid caching in D2D networks, which means nodes can download file data from not only base stations but also from nearby nodes. When nodes have files cached from the base station, it brings about a basic cost to connect with the base station. Considering popularity of cached files, there may be similar file requirements among all nodes. If each node caches data from base stations, there will be a waste in both the node side and the base station side. To solve the problem, when two nodes have the same file requirement, they can cooperate and download the file only once and then share the content. In the cooperative caching, the total

Manuscript received September 6, 2017; revised October 13, 2017; accepted November 13, 2017. Date of publication November 23, 2017; date of current version June 19, 2018. This work was supported in part by the National Science Foundation of China under Grant 61771488, Grant 61671473, Grant 61631020, and Grant 61401508, in part by the Natural Science Foundation for Distinguished Young Scholars of Jiangsu Province under Grant BK20160034, and in part by the Open Research Foundation of Science and Technology on Communication Networks Laboratory. The associate editor coordinating the review of this paper and approving it for publication was J. Mietzner. *(Corresponding author: Yuhua Xu.)*

Y. Zhang, Y. Xu, and K. Yao are with the College of Communications Engineering, PLA Army Engineering University, Nanjing 210007, China and also with the Science and Technology on Communication Networks Laboratory, Shijiazhuang 050002, China (e-mail: yulipkueecs08@sina.com; yuhuaenator@gmail.com; kailing_yao@126.com).

Q. Wu is with the College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China (e-mail: wuqihui2014@sina.com).

X. Liu is with the College of Information Science and Engineering, Guilin University of Technology, Guilin 541006, China (e-mail: leo_nanjing@126.com).

A. Anpalagan is with the Department of Electrical and Computer Engineering, Ryerson University, Toronto, ON M5B 2K3, Canada (e-mail: alagan@ee.ryerson.ca).

Digital Object Identifier 10.1109/LWC.2017.2776920

cost contains one time download cost and sharing cost, which may be smaller than that in the non-cooperative situation. With the download cost and the basic cost afforded by cooperative nodes, cost of each node can be reduced. Furthermore, if one node caches all files from nearby nodes, it will not access the base station and the basic cost can be saved, with cost reduced. To summarize, the contributions of this letter are as follows:

- We consider a cooperative hybrid caching based on content awareness in D2D networks. The hybrid caching mechanism with data cost and sharing cost is formulated, and the coupled file caching strategies are discussed.
- The cache cost reducing problem with constraint is modeled as a local cooperative game, and is proved to be a constraint potential game. The existence of the generalized Nash equilibrium (GNE) is guaranteed.
- The log-linear learning algorithm is modified with an extra half-fixed strategy to speed up the convergence. The algorithm converges to a Nash equilibrium (NE) which approaches the optimal NE approximately.

Note that some social-aware caching problem can be found in our previous work [8], [9]. The main differences are: (i) the caching cost are formulated with different caching mechanisms and parameters. The caching files are coupled in this letter but independent in [8]. The sharing cost is considered in this letter but not in [8] and [9]. (ii) a local cooperative game is modeled in this letter whereas coalition formation games were modeled in [8] and [9]. The corresponding algorithms are also different.

II. System Model and Problem Formulation

In this letter, we consider a distributed cooperative caching problem in D2D networks, as shown in Fig. 1. We consider two ways of caching, one is from the base station directly, and the other is from neighbors. Due to the final caching strategy, nodes can be classified into two kinds. The active nodes download the file data from the base station and share with neighbors. The silent nodes cache from nearby active nodes instead of downloading from the base station. In this way, resources between the base station and cache nodes are saved. At the same time, caching from nearby nodes is more effective and less costly than from the base station, and the caching cost is reduced. Sometimes, due to the similarity of the cache content, the hierarchical caching may reuse the same content, which also contributes to reducing cost.

Assume that the cache node set is $\mathcal{N} = \{1, 2, \ldots, N\}$. Each node has a file queue to be downloaded. For node n, the file queue is denoted as $\mathbf{d}_n = \{d_{n1}, d_{n2}, \ldots, d_{nl_n}\}$ and the length of the queue is l_n. Considering that each node has a capacity for cache, denoted as c_n, the capacity should be no smaller than the file queue length l_n. Assume that there are total L_{max} potential caching files in the network. For caching node n, it generates l_n caching files from the total L_{max} files, randomly. It is assumed that the cache cost from the base station is α_{n0} per file, which may be related with the distance, the channel quality or some other issues about node n. For the second situation, node n might cache files from its neighbors. Due to the limited power, nodes can only transmit data to their nearby neighbors instead of to all nodes in the network. The neighbor node set of node n is $\mathcal{J}_n$. The cost of the second cache way can be divided into two parts. For example, node n caches one unit file data from its neighbor node j. In this situation, node j has a cost α_{j0} caching from the base station and a cost α_{jn} sharing the data with node n. Beside the above cost related to file caching, to access the base station, the active node may pay extra cost to achieve the channel access opportunity, which is denoted as BC_n and is afforded by all its file owners.

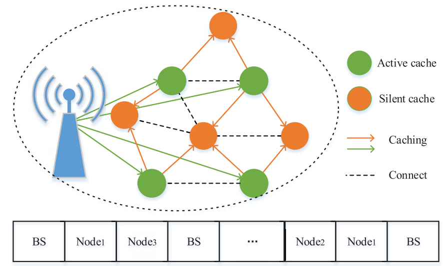

Fig. 1. The system model of hybrid cooperative caching and node caching strategy.

For node n, the decision $\mathbf{a}_n$ is to schedule the cache strategy for each file in the queue, as shown in Fig. 1:

$$\mathbf{a}_n = \left\{a_{n1}, a_{n2}, \ldots, a_{nl_n}\right\}, a_{ni} \in \mathcal{J}_n \cup \{0\} \quad (1)$$

which means file d_{ni} would cache from the node a_{ni}. Case $a_{ni} = 0$ stands for downloading data from the base station. For file d_{ni} of node n, if $a_{ni} = j$, the cost of the file can be formulated as follows:

$$f_{ni}(\mathbf{a}_n, \mathbf{a}_{-n}) = \begin{cases} \frac{BC_j + K_j\alpha_{j0}}{K_j}\frac{1}{\delta_j(d_{ni})} + \alpha_{jn}, j > 0 \\ \frac{BC_n + K_n\alpha_{n0}}{K_n}\frac{1}{\delta_n(d_{ni})}, j = 0, \end{cases} \quad (2)$$

where $j > 0$ means file d_{ni} is cached from a nearby node and there is an extra transmitting cost α_{jn} for transmitting data from node j to node n. K_j is the total file number of node j downloaded from the base station. Then the basic cost BC_j would be afforded by all K_j downloaded files. $\delta_j(d_{ni})$ means the number of the same file with d_{ni}. In this way, the basic cost of node j is afforded by all other nodes in a Shapley value [10]. The case $j = 0$ means that the file is downloaded from base station by itself.

The data cost includes two parts. One is the unit cost for downloading one file from the base station, which is afforded by the number of the same files. This indicates that when two nodes have the same caching file, the better way is to cooperate to download with half cost for each node. This motivates nodes to make content awareness decisions. The other one is the basic cost, which is also shared equally with the total number caching files of the corresponding node. The mechanism decides that the more files, the cheaper the cost will be. Even if two nodes are not very close in the content awareness degree, it is still possible for them to cooperate to reduce the basic cost. The total cost of node n is the sum of all its file costs. Formally,

$$r_n(\mathbf{a}_n, \mathbf{a}_{-n}) = \sum_{i=1}^{l_n} f_{ni}(\mathbf{a}_n, \mathbf{a}_{-n}). \quad (3)$$

Considering that the total downloaded number of each node should be smaller than the capacity, the objective of this letter is to minimize the total cost of all the nodes. Formally,

$$\begin{aligned}&\mathbf{A} = \arg\min \textstyle\sum_{i\in\mathcal{N}} r_i(\mathbf{a}_n, \mathbf{a}_{-n})\\ &\text{s.t. } K_n \le c_n, \forall n \in \mathcal{N}\end{aligned} \tag{4}$$

III. Cooperative Hybrid Caching Game

A. Generalized Local Cooperative Caching Game

Game theory is a useful tool in solving interaction between multiple players, especially in distributed systems. We formulate the cooperative caching problem as an extensive local cooperative game. The game is denoted as $\mathcal{G} = \{\mathcal{N}, \mathcal{A}, \mathcal{E}, u_n\}$, where $\mathcal{N}$ is the node set, $\mathcal{A}$ is the action space, and $\mathcal{E}$ is the connective relationship which represents the neighbor set of each node. The utility function is denoted as $u_n(\mathbf{a}_n, \mathbf{a}_{-n})$. We define the utility function as follows:

$$u_n(\mathbf{a}_n, \mathbf{a}_{-n}) = r_n + \sum_{i\in\mathcal{J}_n} r_i + \sum_{i\in\mathcal{J}_{\mathcal{J}_n}} r_i, \tag{5}$$

where the second part is the total cost of node n's neighbor set and the third part is the total cost of neighbor's neighbor not including the second part. It is little different from the original cooperative model in [11], which only contains the neighbors' rewards. The proposed cooperative caching game is as follows:

$$\mathcal{G}\colon \min u_n(\mathbf{a}_n, \mathbf{a}_{-n}), \forall n \in \mathcal{N}. \tag{6}$$

B. Analysis of the Nash Equilibrium

In this subsection, we first define the Nash equilibrium (NE) and exact potential game of game $\mathcal{G}$.

Definition 1 (NE): The file cache decision of all nodes $(\mathbf{a}_1^*, \mathbf{a}_1^*, \ldots, \mathbf{a}_1^*)$ is a pure strategy NE if and only if no user can improve its utility by deviating unilaterally, i.e.,

$$u(\mathbf{a}_n^*, \mathbf{a}_{-n}^*) \le u(\mathbf{a}_n', \mathbf{a}_{-n}^*), \forall n \in \mathcal{N}, \forall \mathbf{a}_n \in \mathcal{A}, \mathbf{a}_n' \ne \mathbf{a}_n^* \tag{7}$$

Definition 2 (Exact Potential Game) [11]: A game $\mathcal{G} = [\mathcal{N}, \mathcal{A}_n, \mathcal{J}_n, u_n]$ is an exact potential game if there exists a function Φ such that

$$\begin{aligned}&\Phi(\mathbf{a}_n^*, \mathbf{a}_{-n}^*) - \Phi(\mathbf{a}_n', \mathbf{a}_{-n}^*) = u_n(\mathbf{a}_n^*, \mathbf{a}_{-n}^*) - u_n(\mathbf{a}_n', \mathbf{a}_{-n}^*),\\ &\forall \mathbf{a}_n^* \in \mathcal{A}_n, \mathbf{a}_n' \ne \mathbf{a}_n^*\end{aligned} \tag{8}$$

The function Φ is the potential function for the game $\mathcal{G}$.

Theorem 1: The proposed cooperative cache game $\mathcal{G}$ is an exact potential game, which has at least one pure NE.

Proof: Assume the potential function of the game is as follows:

$$\Phi(\mathbf{a}_n^*, \mathbf{a}_{-n}^*) = \sum_{i\in\mathcal{N}} r_i(\mathbf{a}_n^*, \mathbf{a}_{-n}^*), \tag{9}$$

which is just the total cost of all nodes in the network.

From the formulation of the cost r_n, when node n changes its decision, its neighbors are influenced directly. What's more, the nodes with two hops, neighbor's neighbors, still have an influence due to affording cost of the same file with n in its neighbors. Hence, we design the utility function including three partial costs in (5). Then, the three hops nodes are independent from $\mathbf{a}_n$.

Assume that node n changes its action from $\mathbf{a}_n^*$ to $\mathbf{a}_{-n}'$, then the change in potential function is

$$\begin{aligned}&\Phi(\mathbf{a}_n', \mathbf{a}_{-n}^*) - \Phi(\mathbf{a}_n^*, \mathbf{a}_{-n}^*) = \sum_{i\in\mathcal{N}} r_i(\mathbf{a}_n', \mathbf{a}_{-n}^*) - \sum_{i\in\mathcal{N}} r_i(\mathbf{a}_n^*, \mathbf{a}_{-n}^*)\\ &= \sum_{i\in n\cup\mathcal{J}_n\cup\mathcal{J}_{\mathcal{J}_n}} r_i(\mathbf{a}_n', \mathbf{a}_{-n}^*) - \sum_{i\in n\cup\mathcal{J}_n\cup\mathcal{J}_{\mathcal{J}_n}} r_i(\mathbf{a}_n^*, \mathbf{a}_{-n}^*)\\ &= u_n(\mathbf{a}_n', \mathbf{a}_{-n}^*) - u_n(\mathbf{a}_n^*, \mathbf{a}_{-n}^*).\end{aligned} \tag{10}$$

Hence, the proposed game is proved to be an exact potential game and has at least one pure Nash equilibrium. ■

Due to the property of the exact potential function, the Nash equilibrium of the proposed game is also the solution to the potential function. The proposed potential function Φ in this letter is just the minimum of the total cost. Hence, the NE in the cooperative caching game approximately minimizes the total cost at the same time.

Remark: Due to the limited buffer, the strategy space is varying with other nodes' caching decisions. The corresponding Nash equilibrium becomes a generalized NE (GNE) and the GNE existence proof is a little different from the above. (i) Prove the game with constraint turns into an exact constraint potential game, which still keeps the finite improvement property (ii) Find a point $\mathbf{A}_0$ in the constraint action space. Due to the finite improvement property games, improving the performance step-by-step, $\mathbf{A}_0, \mathbf{A}_1, \ldots, \mathbf{A}_{NE}$, a final GNE would be found in the constraint action space. Therefore, the GNE existence is proved. More details can be found in [12].

C. Modified Log-Linear Learning Algorithm

To approach the Nash equilibrium, we modify the log-linear learning algorithm [13] into the distributed cooperative cache game, shown in Algorithm 1.

Theorem 2: The modified log-linear learning algorithm converges to the NE points.

Proof: Compared with the traditional log-linear learning algorithm, our modified algorithm has an extra half-fixed action and comparison before updating. The half-fixed action is designed to reduce the solution space with a better average performance. The comparison is used to keep the probability to achieve global optimization through random actions. Hence, the modified mechanisms do not influence the convergence of the traditional algorithm, which has been proved in [13]. ■

IV. Simulation Results and Discussions

In this section, three algorithms are compared, the random caching which selects the source randomly, the existing non-overlapping coalition formation algorithm [9], and the modified log-linear learning algorithm. The existing non-overlapping algorithm regards all files as an entirety and caches all files from one source. The basic cost $BC = 5$ and the length of the demand queue is 10. The cache capacity is assumed to be twice the demand length. The cost $\alpha_{0i} = 1$ and $\alpha_{ij} = 0.3$. In Fig. 2, the average cost of different user

Algorithm 1 Modified Log-Linear Learning Algorithm

Initialization In $j = 0$ slot, every node makes the decision that all files are cached from the base station, $\mathbf{a}_n = 0$.
Loop $j = 1, 2, ...$
Node Selection: Select one node to update its strategy randomly. With the help of the information exchange, node n can calculate its cost r_n and utility function u_n.
Exploration: For node n, it randomly changes its strategy to $\mathbf{a}'_n(j+1)$. Achieve the new utility function u'_n. Then, user n generates a half-fixed action $\mathbf{a}''_n(j+1)$, when all files are cached from one node or base station. Achieve the new utility function u''_n. All other nodes keep their strategy unchanged during this step.
Update : Node n updates its strategy through the following rule:

$$\begin{aligned} Pr[\mathbf{a}_n(j+1) = \mathbf{a}_n(j)] &= \frac{\exp(\beta u_n)}{X} \\ Pr[\mathbf{a}_n(j+1) = \mathbf{a}_n^{min}] &= \frac{\exp(\beta u_n^{min})}{X} \end{aligned} \quad (11)$$

where $u_n^{min} = \min(u'_n, u''_n)$, $X = \exp(\beta u_n) + \exp(\beta u_n^{min})$ and β is the learning parameter. $\mathbf{a}_n^{min}$ is the corresponding action of the utility u_n^{min}. All other nodes keep their strategy, $\mathbf{a}_{-n}(j+1) = \mathbf{a}_{-n}(j)$. **End**

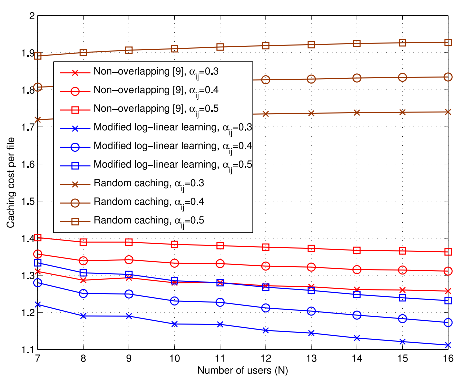

Fig. 2. The average cost of different algorithms and transmitting costs.

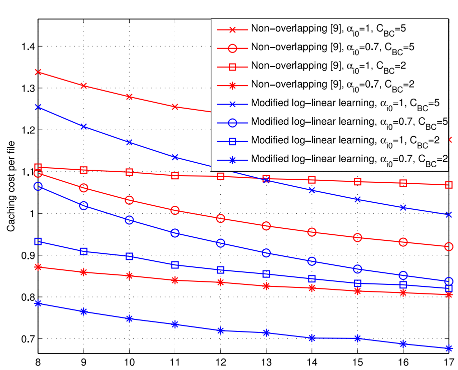

Fig. 3. The average cost of different demand queue lengths, basic cost and downloading cost.

numbers N and transmitting cost α_{ij} is shown. The modified log-linear learning algorithm achieves the best results and the non-overlapping in [9] is close. The costs increase with transmitting cost α_{ij} but decrease with user numbers N. This is because the probability of potential cooperation is increasing for more users. In Fig. 3, we compared the different demand queue lengths with basic cost and downloading cost α_{0i}. The average costs decrease with increasing demand queue lengths, and decreasing α_{0i}, BC. The modified algorithm is always the best. Furthermore, from these results, the cost has different influence to these algorithms. The downloading cost has a more significant effect in the non-overlapping algorithm than the modified log-linear algorithm.

V. Conclusion

The distributed cooperative hybrid caching problem based on content awareness in D2D networks was studied in this letter. Besides the base station, cache nodes can also cache files from nearby nodes according to the content. Nodes can cooperatively cache files from nearby nodes to achieve a lower caching cost. The cost reducing potential problem is modeled as a local cooperative game, and proved to be an exact potential game, which has at least one pure Nash equilibrium. Fortunately, the potential function is just the total cost of all nodes, which means the NE point minimizes the total cost approximately. The log-linear learning algorithm is also modified to converge rapidly and keep a good performance in large strategy space situations. Simulation results show that the content-aware hybrid caching achieves a good performance.

References

[1] M. Taghizadeh, K. Micinski, S. Biswas, C. Ofria, and E. Torng, "Distributed cooperative caching in social wireless networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 6, pp. 1037–1053, Jun. 2013.

[2] C. Fang, Y. F. Richard, H. Tao, L. Jiang, and L. Yunjie, "A game theoretic approach for energy-efficient in-network caching in content-centric networks," *China Commun.*, vol. 11, no. 11, pp. 135–145, Nov. 2014.

[3] Y. Guo, L. Duan, and R. Zhang, "Cooperative local caching under heterogeneous file preferences," *IEEE Trans. Commun.*, vol. 65, no. 1, pp. 444–457, Jan. 2017.

[4] D. Das and A. A. Abouzeid, "Co-operative caching in dynamic shared spectrum networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 7, pp. 5060–5075, Jul. 2016.

[5] B. Chen, C. Yang, and A. F. Molisch, "Cache-enabled device-to-device communications: Offloading gain and energy cost," *IEEE Trans. Wireless Commun.*, vol. 16, no. 7, pp. 4519–4536, Jul. 2017.

[6] A. Araldo, D. Rossi, and F. Martignon, "Cost-aware caching: Caching more (costly items) for less (ISPs operational expenditures)," *IEEE Trans. Parallel Distrib. Syst.*, vol. 27, no. 5, pp. 1316–1330, May 2016.

[7] N. Kumar and J.-H. Lee, "Peer-to-peer cooperative caching for data dissemination in urban vehicular communications," *IEEE Syst. J.*, vol. 8, no. 4, pp. 1136–1144, Dec. 2014.

[8] L. Ruan *et al.*, "Context-aware group buying in D2D networks: An overlapping coalition formation game approach," in *Proc. IEEE ICCT*, Chengdu, China, Oct. 2017, pp. 1–5.

[9] Y. Zhang, Y. Xu, and Q. Wu, "Group buying based on social aware in D2D networks: A game theoretic approach," in *Proc. IEEE/CIC ICCC*, Qingdao, China, Sep. 2017, pp. 1–6.

[10] W. Saad, Z. Han, M. Debbah, A. Hjorungnes, and T. Basar, "Coalitional game theory for communication networks," *IEEE Signal Process. Mag.*, vol. 26, no. 6, pp. 77–97, Sep. 2009.

[11] Y. Xu, J. Wang, Q. Wu, A. Anpalagan, and Y.-D. Yao, "Opportunistic spectrum access in cognitive radio networks: Global optimization using local interaction games," *IEEE J. Sel. Topics Signal Process.*, vol. 6, no. 2, pp. 180–194, Apr. 2012.

[12] J. Zheng, Y. Cai, X. Chen, R. Li, and H. Zhang, "Optimal base station sleeping in green cellular networks: A distributed cooperative framework based on game theory," *IEEE Trans. Wireless Commun.*, vol. 14, no. 8, pp. 4391–4406, Aug. 2015.

[13] Y. Xu, Q. Wu, L. Shen, J. Wang, and A. Anpalagan, "Opportunistic spectrum access with spatial reuse: Graphical game and uncoupled learning solutions," *IEEE Trans. Wireless Commun.*, vol. 12, no. 10, pp. 4814–4826, Oct. 2013.