INTERNATIONAL JOURNAL OF COMMUNICATION SYSTEMS
Int. J. Commun. Syst. (2013)
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/dac.2648

Performance fairness across multiple applications in wireless push systems

P. Nicopolitidis*,†

Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki, Greece

SUMMARY

This letter addresses the area of performance fairness in wireless data broadcasting. It proposes an adaptive wireless push system capable of fairly allocating performance to multiple applications receiving content from a Broadcast Server. While requiring only minimal changes at the Broadcast Server, the proposed approach provides performance fairness to clients running different applications and manages to alleviate the performance difference that typically appears in wireless push systems, when different applications access different-sized subsets of data items, via a possibly different demand pattern. Copyright © 2013 John Wiley & Sons, Ltd.

Received 20 May 2013; Revised 19 August 2013; Accepted 19 August 2013

KEY WORDS: Data broadcasting; fairness; mobile computing; learning automata

1. INTRODUCTION

Adaptive data broadcasting (e.g., [1, 2]) has emerged as an efficient way for dissemination of information over wireless environments, in which the demands of client devices for data items substantially overlap and are unknown to the Broadcast Server (BS).

In data broadcasting, the primary performance metric is the mean time a client waits to receive a data item (mean access time), which is desirable to be as low as possible. However, another equally important metric is fairness of the performance offered to the various client applications. To this end, Kakali *et al*. [1] proposed an adaptive wireless push-based system that achieves performance fairness for different applications that run on client groups of unequal sizes. Specifically, the problem addressed in [1] is that applications run by client groups having a few members have much lower performance than applications run by groups with many clients, because the performance of each application depends on the size of the group running it. It proposes a fair push system where the offered performance for each application is independent of the total number of clients that run it. Using different weight coefficients for the clients' feedbacks of each group, [1] succeeds in providing fairness in terms of response time and the percentage of broadcasts for the various applications in the system.

Nevertheless, performance fairness also depends on (i) the number of data items accessed by each application and (i) the demand skewness for each application, which is essentially the amount of commonality characterizing the data item demands of clients that run the same application. When the aforementioned two parameters are not the same for all applications, the mean access time for each application will be different, despite the use of the method of [1].

This letter proposes a method of alleviating the problem of performance unfairness across multiple applications, caused by the two aforementioned parameters. Our method requires additional functionality only at the BS; thus, it can constitute a simple and effective means of

*Correspondence to: Petros Nicopolitidis, Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki, Greece.
†E-mail: petros@csd.auth.gr

supporting fairness by wireless data broadcasting providers. Apart from [1], it is the only approach to our knowledge dealing with fairness in push-based broadcasting, as other recent approaches (e.g., [3]) concern on-demand (pull) systems running at special environments.

2. THE PROPOSED SYSTEM

The proposed system uses an S-model Learning Automaton (LA) at the BS, whose probability distribution vector p contains the server's estimate p_i of the demand probability d_i for each data item i demanded by the client population. LA are artificial intelligence tools that can acquire knowledge regarding the environment in which they operate and have extensively been applied in wireless networking [4–7] and wireless data broadcasting specifically (e.g., [8–10]). The clients run a number of different applications, each demanding items from a different subset of the BS's database. Each client acknowledges reception of an item it is waiting via Code Division Multiple Access [1, 2].

For each item broadcast, the BS selects to broadcast the item i that maximizes the cost function

$$G(i) = (T - R(i))^2 \frac{w_i p_i}{l_i} \tag{1}$$

[11], where T is the current time, $R(i)$ is the time when i was last broadcast, l_i is the length of item i and w_i is its weight. After the broadcast of item i, the BS waits for an acknowledging feedback from clients that were waiting for item i. For items that have not been previously broadcast, $R(i)$ is initialized to -1. If the maximum value of $G(i)$ is shared by more than one item, the algorithm selects one of them arbitrarily. Upon the broadcast of item i at time T, $R(i)$ is changed so that $R(i) = T$. If this was the kth broadcast, the item estimation vector p is updated according to the re-enforcement scheme of the S-model LA:

$$\begin{aligned} p_j(k+1) &= p_j(k) - L(1-\beta(k))\big(p_j(k) - a\big), \forall j \neq i \\ p_i(k+1) &= p_i(k) + L(1-\beta(k))\textstyle\sum_{j\neq i}\big(p_j(k) - a\big) \end{aligned} \tag{2}$$

where $p_i(k) \in (a..1)\ \forall i$ and $L, a \in (0,1)$. L sets the rate of LA convergence, while the role of parameter α is to prevent the probabilities of non-popular items from taking values very close to zero in order to increase the adaptivity of the LA [10]. This is because if the probability estimate p_i of an item i approaches zero, then $G(i)$ will take a value very close to zero. However, item i, even if unpopular, still needs to be transmitted because some clients may request it. Furthermore, the dynamic nature of client demands might make this item popular in the future so its probability estimate should not equal zero. $(1-\beta(k)) \in [0..1]$ is the normalized environmental response for the server's k^{th} broadcast. It is essentially the percentage of clients acknowledging the kth broadcast item. It has been shown [10] that the probability distribution vector p maintained by the Automaton estimates the demand probability of each information item i. For the next broadcast, the server chooses which item to transmit by using the updated vector p.

The item weight parameter w_i has not been used so far for achieving fairness, as all items were considered to have equal weights. In the proposed system, the BS will regularly use its vector p to estimate the performance S_z of each application z as $S_z = \frac{1}{2}\left(\sum_{i=1}^{M_z} \sqrt{p_i^z l_i}\right)^2$. This is the optimal overall mean access time of an application z that accesses a subset of M_z items [11], with a demand probability vector of p^z. p^z is computed from the respective subset of the overall demand probability vector p and then normalized so that $\sum_{i=1}^{M_z} \sqrt{p_i^z l_i} = 1$. Thus, for any two items in positions a1, a2 in the database, with respective positions a1′ and a2′ in the item subset accessed by application z, after the weighting procedure, it will hold that $\frac{p_{a1}}{p_{a2}} = \frac{p^z_{a1'}}{p^z_{a2'}}$.

After having obtained the mean access time estimates for each application z, the BS will compute the weight w_z for every item i in the item set demanded by each application z to be S_z / S_{zmin}, where

DOI: 10.1002/dac

S_{zmin} is the highest application optimal overall mean access time estimate and corresponds to the application z_{min} having the lowest performance. It can be easily seen that this approach assigns weights to the items demanded by an application in a manner proportional to the overall mean access time estimate for this application. Thus, items accessed from a certain application will be broadcasted with an increased probability compared to items of other applications originally having lower mean access times, thus resulting to an increased bandwidth assignment and thus a performance increase for the 'slow' applications. It can also be seen that the complexity for computing the weights of the data items in a subset accessed by each application is linear to the number of the items in the subset; thus, the procedure does not increase the complexity of [2, 10, 11], which is also linear to the number of data items.

The novelty of the proposed approach is that it addressed performance fairness in an adaptive wireless data broadcasting environment. Apart from [1], it is the only approach to our knowledge dealing with fairness, as other recent approaches (e.g., [3]) concern on-demand (pull) systems running at special environments.

3. PERFORMANCE EVALUATION

We consider a BS that broadcasts data items from a set of N equally-sized items having initial probability estimates of $1/N$. We also consider four different applications $z \in [1..4]$ running on Cl clients, with each client running one application. Each application accesses different database subsets of size Num_z items each. The demand probability d_i for an item in place i in a subset is computed via the Zipf distribution [1, 2, 11]: $d(i) = q(1/i)^{\theta}$, $q = 1/\sum_k (1/k)^{\theta}$, $k \in [1..Num_z]$. The data skew coefficient θ is a parameter that when increased, increases demand skewness. The number of clients that run each application z equals the parameter NCl_z. The BS estimates the weights of data items every Est item broadcasts.

The simulation results were obtained via an event-driven simulator coded in the C language. The BS, the mobile clients and the wireless links are modeled as separate entities that interact via events. The simulation runs until each E data items are broadcast by the BS and uses the following parameters: $N = 300$, $Cl = 10000$, $E = 1000000$, $L = 0.015$, $\alpha = 10^{-6}$, $Num_1 = 9$, $Num_2 = 27$, $Num_3 = 81$, $Num_4 = 183$. $Est = 300$.

We simulated three network scenarios, N_1, N_2 and N_3. In N_1, the demand skewness (θ) of all applications are all equal, ranging together from 0.0 to 1.4, and the number of clients NCl_z running each application $z \in [1..4]$ is 2500. In N_2, the demand skewness characteristics are as in N_1, and

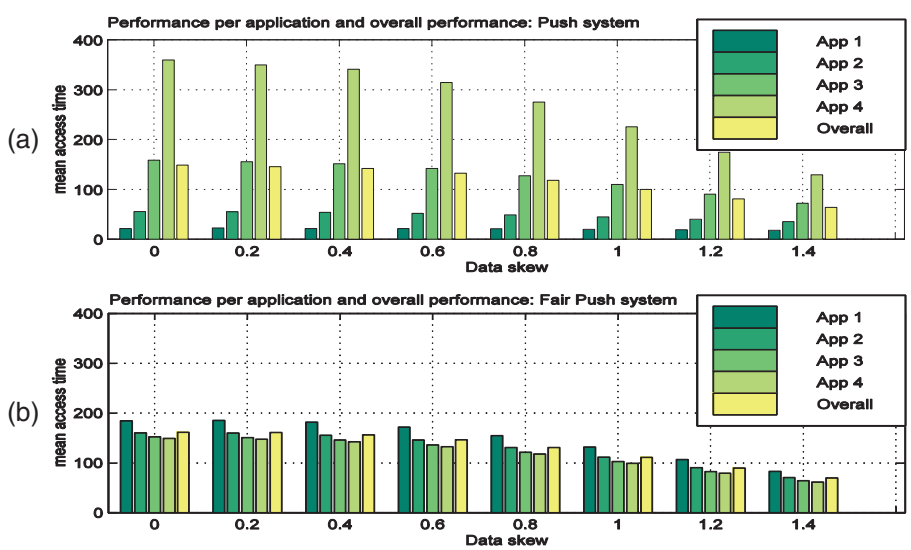

Figure 1. Scenario N_1: performance for applications 1–4 and overall performance in the system of [2] (a) and the proposed fair one (b).

DOI: 10.1002/dac

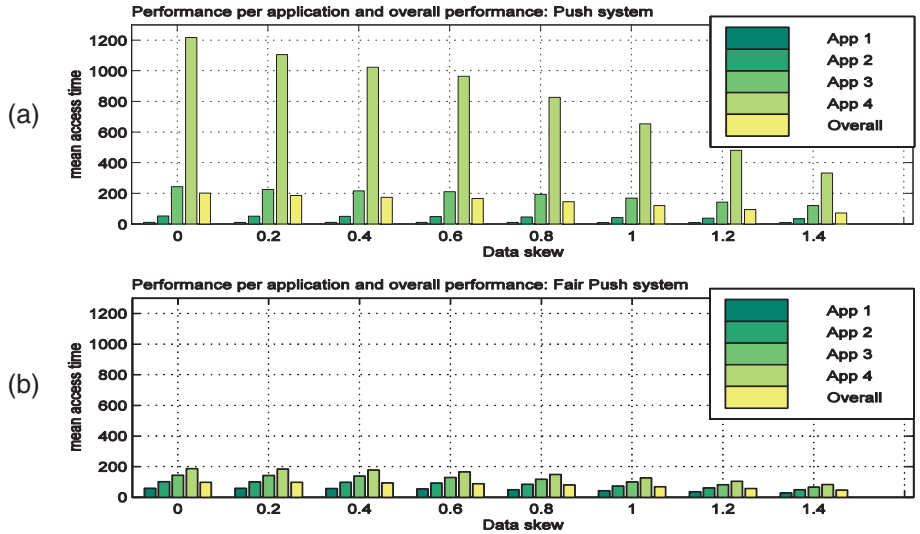

Figure 2. Scenario N_2: Performance for applications 1–4 and overall performance in the system of [2] (a) and the proposed fair one (b).

$NCl_1 = 4800$, $NCl_2 = 2400$, $NCl_3 = 1600$, $NCl_4 = 1200$. In N_3, the demand skewness for each application is random in [1..1.4], and the number of clients running NCl_z are as in N_2.

Figures 1–3 and Table I show simulation results for N_1–N_3. In these Figures, the proposed fair system is compared to [2] in terms of performance offered to applications 1–4 as well as overall performance. The main conclusions drawn from the Figures are summarized as follows:

- When every application is run by the same number of clients (N_1), the proposed system manages to alleviate the fairness problem caused by applications accessing unequally-sized data item sets, as it yields a much more fair balance between the overall mean access time offered to each application (Figure 1(a and b)). To show this numerically, we computed the Jain Fairness Index (JFN) [12] for each result set in N_1. As seen in Table I, the JFN for N_1 approaches the optimum of 1 for all result sets of the proposed approach, whereas it is much less for the system of [2].

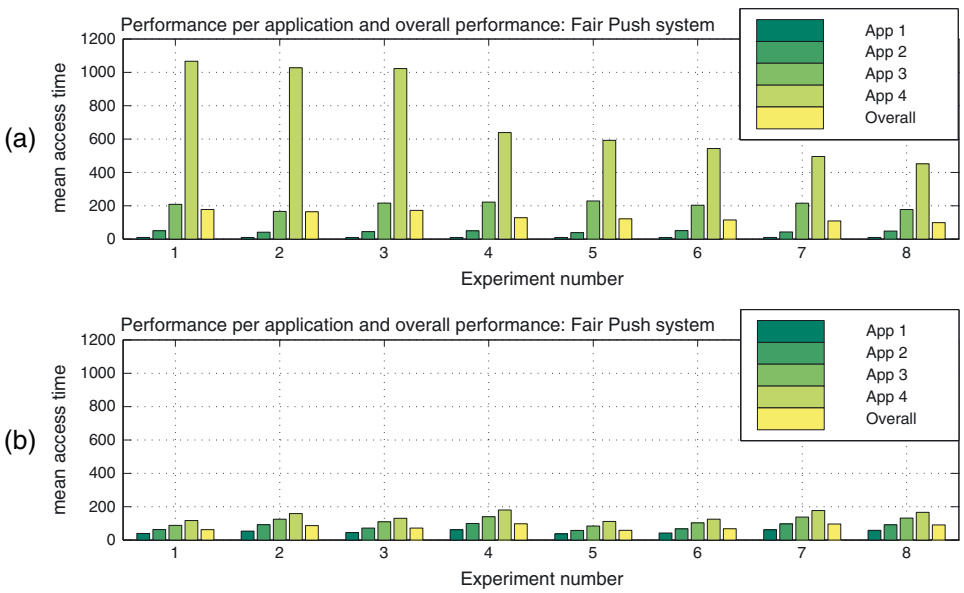

Figure 3. Scenario N_3: Performance for applications 1–4 and overall performance in the system of [2] (a) and the proposed fair one (b).

DOI: 10.1002/dac

Table I. Fairness indices for the system of [2] and the proposed fair one, in N_1–N_3.

		Fair	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
N_1	JFN	[2]	0.56	0.57	0.57	0.57	0.59	0.61	0.65	0.69
		Fair	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
N_2	JFN	[2]	0.38	0.38	0.39	0.39	0.40	0.42	0.45	0.49
		Fair	0.87	0.87	0.87	0.88	0.88	0.88	0.89	0.89
	WJFN	[2]	0.45	0.47	0.46	0.47	0.49	0.56	0.56	0.63
		Fair	0.99	0.99	0.99	0.99	0.99	0.99	0.98	0.98
N_3	JFN	[2]	0.38	0.36	0.38	0.46	0.47	0.48	0.5	0.5
		Fair	0.88	0.88	0.88	0.88	0.87	0.88	0.88	0.88
	WJFN	[2]	0.45	0.42	0.46	0.57	0.57	0.6	0.61	0.63
		Fair	0.98	0.99	0.98	0.98	0.98	0.99	0.98	0.98

JFN, Jain Fairness Index; WJFN weighed JFN.

- The benefit described previously also holds in N_2 (Figure 2(a and b)). Again, the JFN is seen from Table I to be superior for the proposed approach. However, as in N_2, the number of clients running the same application is different, it would be normal to expect mean access times for each application inversely proportional to the number of clients running the application. This is desirable in data broadcasting systems, as more popular data are supposed to be broadcast more frequently. As this proportional fairness is not directly apparent from Figure 2 visually, we also computed the weighed JFN (WJFN) for each result set in N_2. This was carried out by weighting the mean access time of each application with the percentage of the clients that run the application. As seen from Table I for $N_{2,}$, it approaches the optimum value of 1 for the proposed approach, whereas it is much less for the system of [2].
- The proposed system also alleviates the problem of applications accessing unequally-sized data item sets with different demand skewness per each application (N_3). The application performances are shown in Figure 3(a and b). Table I shows that performance fairness across the four applications is nearly optimal, as the WJFN for each result set in N_3 reaches 1, whereas it is much less for the system of [2].
- It can be seen from Figures 1–3 that the overall system performance is not significantly affected in a negative manner. Moreover, it is actually improved in N_2 and N_3, as the fourth application is alleviated from the starvation caused by the facts that it accesses the largest set of data items and is run by the smallest number of clients in the system.

4. CONCLUSION

This letter proposed an adaptive wireless data broadcasting system of push nature, capable of providing a fair allocation of bandwidth to multiple client applications, each accessing different-sized subsets of data items, with a possibly different data demand pattern per application. It requires additional functionality only at the BS; thus, it can constitute a simple and effective means of supporting performance fairness by wireless data broadcasting providers.

REFERENCES

1. Kakali V, Papadimitriou GI, Nicopolitidis P, Pomportsis AS. A new class of wireless push systems. *IEEE Transactions on Vehicular Technology* 2009; **58**(8):4529–4539.
2. Nicopolitidis P, Papadimitriou GI, Pomportsis AS. Continuous flow wireless data broadcasting for high-speed environments. *IEEE Transactions on Broadcasting* 2009; **55**(2):260–269.
3. Hu C-L. Fair scheduling for on-demand time-critical data broadcast. *Proceedings of IEEE ICC*, Taipei, Taiwan, 2007; 5831–5836.
4. Nicopolitidis P, Papadimitiou GI, Sarigiannidis PG, Obaidat MS, Pomportsis AS. Adaptive wireless networks using learning automata. *IEEE Wireless Communications Magazine* 2011; **18**(2):75–81.
5. Sarigiannidis A, Nicopolitidis P, Papadimitriou G, Louta M, Pomportsis AS. On the use of learning automata in tuning the channel split ratio of WiMax networks. *IEEE Systems Journal* 2013; accepted.
6. Torkestani A, Meybodi M. An efficient cluster-based CDMA/TDMA scheme for wireless mobile ad-hoc networks: a learning automata approach. *Journal of Network and Computer Applications, Elsevier* 2010; **33**(4):477–490.

DOI: 10.1002/dac

7. Torkestani A, Meybodi M. Mobility-based multicast routing algorithm for wireless mobile ad-hoc networks: a learning automata approach. *Computer Communications, Elsevier* 2010; **33**(6):721–735.
8. Nicopolitidis P, Chrysostomou C, Papadimitriou G, Pitsillides A, Pomportsis AS. On the efficient use of multiple channels by single-receiver clients in wireless data broadcasting. *International Journal of Communication Systems, Wiley* 2012; DOI:10.1002/dac.2375.
9. Polatoglou M, Nicopolitidis P, Papadimitiou GI. On low-complexity adaptive wireless push-based data broadcasting. *International Journal of Communication Systems, Wiley* 2012; DOI:10.1002/dac.2343.
10. Nicopolitidis P, Papadimitriou GI, Pomportsis AS. Using learning automata for adaptive push-based data broadcasting in asymmetric wireless environments. *IEEE Transactions on Vehicular Technology* 2002; **51**(6):1652–1660.
11. Vaidya NH, Hameed S. Scheduling data broadcast in asymmetric communication environments. *ACM Wireless Networks* 1999; **55**(3):171–182.
12. Jain R, Chiu DM, Hawe W. A quantitative measure of fairness and discrimination for resource allocation in shared computer systems. *DEC Research Report TR-301*.

AUTHORS' BIOGRAPHY

Petros Nicopolitidis received BS and PhD degrees in computer science from the Department of Informatics, Aristotle University of Thessaloniki, Greece, in 1998 and 2002, respectively. From 2004 to 2009, he was a lecturer in the same department where since 2009 he serves as assistant professor. He has published more than 80 papers in international refereed journals and conferences. He is co-author of the book Wireless Networks (Wiley, 2003). His research interests are in the areas of wireless networks and mobile communications. He serves as an associate editor for the International Journal of Communication Systems and Security and Communications Journal, both published by Wiley. He is a senior member of IEEE.

DOI: 10.1002/dac