

# Fairness Assessment of the Adaptive Token Bank Fair Queuing Scheduling Algorithm

Feroz A. Bokhari\*, Halim Yanikomeroglu\*, William K. Wong<sup>†</sup>, and Mahmudur Rahman\*

\*Broadband Communications and Wireless Systems Centre, Department of System and Computer Engineering  
Carleton University, Ottawa, Ontario K1S 5B6, Canada  
Email: {fbokhari,halim,mrahman}@sce.carleton.ca

<sup>†</sup> Communication Research Center Canada  
Ottawa, Ontario K2H 8S2, Canada  
Email: william.wong@crc.ca

**Abstract**—Adaptive Token Bank Fair Queuing (ATBFQ) algorithm has been proposed as a cross-layer scheduling technique for 4G wireless systems recently. This algorithm takes higher layer Quality of Service (QoS) attributes such as priorities, interflow fairness, and delay constraints into account. By selecting the user terminals (UTs) in a certain prioritized manner derived from QoS attributes, the performance of the UTs, suffering from high interference and/or shadowing in particular, can be improved. The ATBFQ algorithm has been tested in a multicell environment in the presence of intercell interference by comparing with reference Score Based (SB) and Round Robin (RR) algorithms. In this paper, we further analyze ATBFQ primarily with regard to fairness along with other performance metrics accessed in a more elaborate system considering varying interference and loading conditions. Furthermore, an adaptive method for the allocation of resources is proposed for ATBFQ parameter selection, and is shown to have better performance in various loading conditions. It is observed from simulation results that ATBFQ with adaptive parameter selection outperforms the reference schemes in terms of queuing delay and UT throughput for different network loading cases.

**Keywords:** Radio resource management, QoS, Cross layer scheduling, Fairness.

## I. INTRODUCTION

Research on packet scheduling for wire-line networks has matured extensively during the last two decades. Much of this research has focused on scheduling algorithms similar to the Weighted Fair Queuing (WFQ) algorithm [1] which is a packet-based version of Generalized Processor Sharing (GPS) [2]. This is because GPS can guarantee to the different applications (sessions) that the network resources are allocated fairly and independently of the behavior of the other applications [3]. Most literature on such packet scheduling is based on the fact that throughput of the channel is constant.

On the other hand, the research for wireless networks has mainly concentrated on allocating radio resources, e.g., time-slots, frequencies, powers and/or codes, to different UTs. Most of these scheduling algorithms do not take the UTs' QoS

requirements into account and mainly focus on how to exploit the time-varying nature of the wireless channels in order to increase the throughput in an opportunistic manner.

The scheduling problem becomes more complex for future generation systems, such as 4G wireless networks, due to the diverse nature of the anticipated traffic (on demand video streaming, voice over Internet protocol (VoIP), file transfer, and web browsing) as well as the unpredictable nature of the wireless channel where mobile speeds of up to 100 km/hr have to be supported in different propagation environments. This segregation between packet scheduling and radio resource scheduling is not efficient. It is necessary to merge packet scheduling and resource allocation to design cross-layer scheduling algorithms that meet QoS requirements for diverse traffic as well as exploit time varying nature of the wireless channel [4]. Wireless World Initiative New Radio (WINNER) is a 4G wireless research consortium where one of the key focuses is to design an efficient cross-layer scheduling scheme [5].

It is crucial for a scheduler to consider the characteristics of the packet load of the buffers at the mobile UTs and the BS containing packets waiting to be transmitted for uplink and downlink scheduling, respectively [6]. Adaptive Token Bank Fair Queuing (ATBFQ) algorithm, proposed in [7], provides fairness without sacrificing the network throughput by taking both queue and channel states into account. It is adapted from the Token Bank Fair Queuing (TBFQ) algorithm [8], originally proposed for single carrier Time Division Multiple Access (TDMA) systems, to suit WINNER Orthogonal Frequency Division Multiple Access (OFDMA) air-interface. In this paper, we refine the ATBFQ algorithm by introducing an adaptive method for parameter selection which shows improved performance. The primary focus of this paper is on fairness assessment of the enhanced ATBFQ in the downlink of the WINNER system with varying interference and loading scenarios.

The performance of the ATBFQ algorithm is compared to the Score Based (SB) scheduling algorithm (which was the baseline scheduling scheme in WINNER) [9] and the Round Robin (RR) scheme by extensive simulations. The rest of

---

This work was supported by the Natural Sciences and Engineering Research Council (NSERC) of Canada under participation in the Wireless World Initiative New Radio (WINNER) project – [www.ist-winner.org](http://www.ist-winner.org).

the paper is organized as follows. In Section II, the ATBFQ algorithm is described in detail along with its parameter selection and methods of fairness assessment. The system model and the simulation parameters are presented in Section III. Simulation results are provided in Section IV followed by conclusions in Section V.

## II. ATBFQ SCHEDULING ALGORITHM

### A. Description of ATBFQ Algorithm

An overview of the ATBFQ scheme is shown in Fig. 1. It highlights the scheduling operation that includes the arrangement of per-flow queuing (PFQ) of packets belonging to different service classes, and the scheduling procedure based on channel feedback, service prioritization and other QoS parameters. As the access scheme being used is OFDMA, there are many chunks to be scheduled in each scheduling frame. A chunk is defined as a frequency-time unit defined by fixed number of subcarriers and OFDMA time symbols [10].

The ATBFQ (as well as TBFQ) scheduling principle is based on the leaky bucket mechanism. Each traffic flow  $i$  is characterized by a packet arrival rate  $\lambda_i$ , token generation rate  $r_i$ , token pool size  $p_i$ , and a counter  $E_i$  to keep track of the number of tokens borrowed from or given to the token bank. Each  $L$ -byte packet consumes  $L$  tokens. As tokens are generated at rate  $r_i$ , the tokens overflowing from the token pool are added to the token bank, and  $E_i$  is incremented by the same amount. When the token pool is depleted and there are still packets to be served, tokens are withdrawn from the bank by flow  $i$ , and  $E_i$  is decremented by the same amount. A debt limit  $d_i$  is set as a threshold to limit the amount a UT can borrow from the bank. It also acts as a measure to prevent malicious UTs (transmitting at unusually high transmission rates) from borrowing extensively. The scheduling algorithm is defined by the following steps.

- 1) Using the *getActiveUsers()* function, information regarding backlogged UTs is retrieved from the Radio Link Controller (RLC) layer residing above [10].
- 2) The *highestBorrow Priority()* function returns the UT  $i^*(t_k)$  with the highest priority  $P_i$  among the backlogged users in the current scheduling frame determined by  $i^*(t_k) = \arg \max_{1 \leq i \leq N_{act}} (P_i)$ , where  $P_i = \frac{E_i}{r_i}$  and  $N_{act}$  is the number of active users.
- 3) The *borrowBudget()* function calculates the budget for the selected UT  $i^*$  (i.e., flow  $i$ ) based upon the amount of tokens the UT has contributed to the bank, the debt limit it has incurred from previous round of scheduling, and the channel feedback received by the UT.
- 4) The chunk allocation is done based on the calculated budget using the *maxSINR()* function. Chunks are allocated based on the maximum Signal-to-Noise-plus-Interference Ratio (SINR) principle, where the chunk  $j^*$  with the highest SINR is given to the selected user  $i^*$  [11] according to the following:

$$j^*(t_k) = \arg \max_{1 < j < N_{chunks}} (\gamma_{i^*j}(t_k)), \quad (1)$$

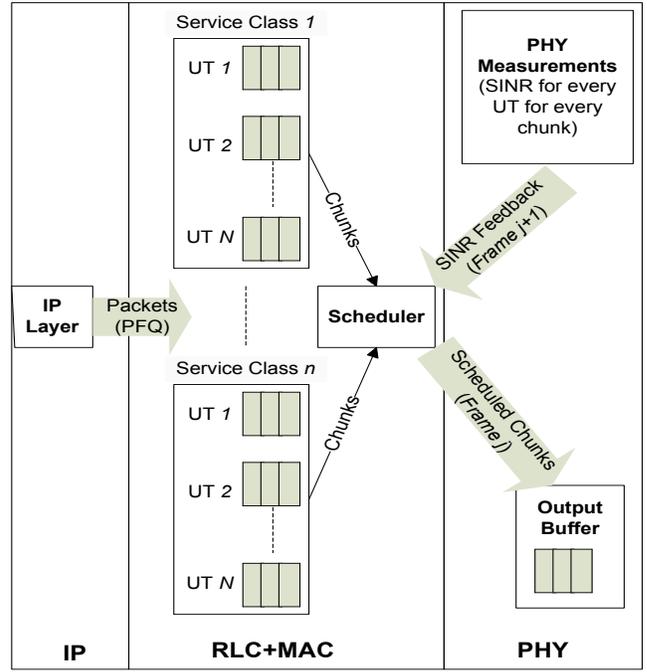


Fig. 1. Overview of scheduling operation

where  $\gamma_{i^*j}$  is the SINR of the selected user  $i^*$  on chunk  $j$  and  $N_{chunks}$  is the number of available chunks in the current frame.

- 5) The *resourceMap()* function determines the amount of bits that can be mapped to the chunk depending on the type of modulation and coding used.
- 6) Each time a chunk resource is allocated, the *updateCounter()* function is called. This function updates the bank, counter  $E_i$ , and the allocated budget.

The selected user  $i$  gets to transmit as long as its queue is backlogged or the allocated budget is less than the total bank size and more than the number of bits that can be supported for the smallest modulation and coding scheme (BPSK rate 1/2). If either of these conditions is not satisfied, the UT is classified as inactive. A new priority is calculated on the updated active UTs and the above steps are repeated. This procedure is carried out till there are no chunk resources available or no active UTs remain.

### B. ATBFQ Parameter Selection

The performance of the ATBFQ scheduler depends on its parameters that define the debt limit, the burst credit  $BC$ , and the token generation rate. The token generation rate is detrimental to the extent to which the burstiness of the UT traffic can be accommodated. A UT in its burst mode transmits more data in a short interval of time than its actual statistics, and hence, requires more resources in order to maintain a certain QoS level. Thus, the token generation rate is set to be considerably large. In simulations, this has been taken to be three times the packet arrival rate.

The burst credit for flow  $i$  ( $BC_i$ ) determines the amount of bits selected user  $i^*$  can receive in a frame. In the TBFQ

TABLE I  
BURST CREDIT FOR ATBFQ FOR LOW LOADING (8 USERS)

Burst Credit (BC)	Queuing Delay (sec)	Packets Dropped (per frame)	Throughput (Byte per frame)	Spectral Efficiency (bits/sec/Hz)
BC = 1000	0.025	4.36	815.4	2.37
BC = 5000	0.017	0.76	1473.3	2.05
BC = 10000	0.015	0.42	1546.6	1.98
<b>Adaptive BC</b>	<b>0.012</b>	<b>0.30</b>	<b>1551.1</b>	<b>2.34</b>

TABLE II  
BURST CREDIT FOR ATBFQ FOR HIGH LOADING (20 USERS)

Burst Credit (BC)	Queuing Delay (sec)	Packets Dropped (per frame)	Throughput (Byte per frame)	Spectral Efficiency (bits/sec/Hz)
BC = 1000	0.044	3.19	2299.4	2.09
BC = 5000	0.036	3.98	2094.0	1.88
BC = 10000	0.033	4.00	2090.4	1.87
<b>Adaptive BC</b>	<b>0.038</b>	<b>2.01</b>	<b>2497.1</b>	<b>2.29</b>

algorithm, this quantity was a fixed measure. In this research, this has been modified to be adaptive. It is observed through simulations that for low loading cases, a higher value for  $BC_i$  is shown to perform better as illustrated in Table I. On the other hand, for high loading conditions, a lower value for  $BC_i$  is desired as it makes use of multiuser diversity as shown in Table II. It is further shown that this can be achieved for both low and high loading conditions by calculating the  $BC_i$  for the selected UT in an adaptive manner. For UT  $i$ , this adaptive value depends on the past spectral efficiency  $\eta_i$ , the number of available chunks  $N_{chunks}$ , the amount of time-frequency resource in a chunk,  $M$ , and the number of active UTs  $N_{act}$  in that particular scheduling frame. It can be formulated as follows:

$$BC_i = \frac{\eta_i(\text{bits/sec/Hz}) * M(\text{Hz} * \text{sec}) * N_{chunks}}{N_{act}}. \quad (2)$$

### C. ATBFQ Fairness Study

There are scheduling algorithms which work to achieve better performance for UTs having good channel conditions (closer to the BS) whereas UTs further from the BS suffer tremendously. In such circumstances, the overall throughput of the system is maximized but the fairness amongst UTs is greatly affected. Therefore, it is essential to design a performance metric which is an appropriate indicator of the fairness. One such index is the *Jain's fairness index* proposed in [12]. This fairness index is bounded between zero and unity, and has been widely used [13][14]. If a system allocates resources to  $n$  contending UTs such that the  $i^{th}$  user receives an allocation  $x_i$ , then this fairness index  $f_I(x)$  is given by:

$$f_I(x) = \frac{[\sum_{i=1}^n x_i]^2}{n \sum_{i=1}^n x_i^2}, \quad (3)$$

where  $x_i \geq 0$ . This index measures the equality of UT allocation  $x$ . If  $x_i$ 's are equal for all UTs, then the fairness index is 1 and the system is 100% fair and vice versa. In this

paper, the allocation metric 'x' is defined as the ratio of UT throughput and queue size, and is given by

$$x_i = \left[ \frac{TP_i}{Q_i} \right]_{(t_1, t_2)}, \quad (4)$$

where  $TP_i$  is the transmitted throughput in bits for UT  $i$  during the time interval  $[t_1, t_2]$  and  $Q_i$  is the total queue size for UT  $i$  given by  $Q_i = Q_i^{t_2} - Q_i^{t_1}$ . Here  $Q_i^{t_2}$  and  $Q_i^{t_1}$  are the queue sizes of UT  $i$  at time instant  $t_1$  and  $t_2$ , respectively. We chose  $t_2 - t_1$  to be 16 frame time duration.

In (4), the throughput is normalized to avoid the ambiguity as the throughput alone as a metric does not provide an insight into the overall fairness of the system when using modeled traffic.

We also use another method of fairness assessment (proposed in WiMAX standard [15]) which is determined by the normalized cumulative distributive furniture (CDF) of throughput per UT. The normalized UT throughput with respect to the average throughput,  $\tilde{T}_i$  for UT  $i$  is given by:

$$\tilde{T}_i = \frac{T_i}{\frac{1}{n} \sum_{j=1}^n T_j}, \quad (5)$$

where  $T_i$  is the instantaneous throughput of UT  $i$  in a particular frame and  $n$  is the total number of UTs. As stated in the standard, the CDF of this normalized throughput should lie to the right of the coordinates (0.1, 0.1), (0.2, 0.2), and (0.5, 0.5).

The results using both of these fairness assessment methods are discussed in details in Section IV.

## III. SYSTEM MODEL

The ATBFQ scheme is evaluated in the WINNER wide area down link scenario. To reduce the simulation complexity, the bandwidth is reduced to 15 MHz from the original 45 MHz. The chunk dimension is given as 8 subcarriers by 12 OFDMA symbols or  $312.5 \text{ KHz} \times 345.6 \mu\text{s}$ . The frame duration is defined as  $691.2 \mu\text{s}$ , i.e., there are a total of 96 chunks available to the scheduler in a frame.

Rayleigh channel samples correlated both in time and frequency are obtained from the power delay profile for WINNER wide area scenario. The UT speed is considered to be 70 km/hr and the inter site distance is 1 km. The following exponential path-loss (PL) model has been used [16]

$$PL = 38.4 + 35.0 \log_{10}(d)[dB], \quad (6)$$

where  $d$  is the distance in meters between the BS and UT. Three-sectored cells with hexagonal sectors have been considered in simulations. Available frequency band is reused in each sector. Interference from the first-tier of cells is calculated using the central cell approach. UTs are assumed to be distributed uniformly in the sector of interest in the central cell from where the performance statistics are collected. To model the impact of the varying interference, we have introduced an Activity Factor (AF) which is defined as a probability for a particular interfering link to be active. For example, AF of 1 denotes the highest level of interference

where all interferer links are active. In this case, for a UT in the sector of interest, the interference will comprise of 18 inter-cell (6 BS x 3 sectors) and 2 intra-cell links.

Adaptive Modulation and Coding (AMC) is used with Block Low-Density Parity-Check (B-LDPC) code. Thresholds for transmission schemes are determined assuming a block length of 1704 bits and 10% Block Error Rate (BLER) [16]. A chunk using Quadrature Phase Shift Keying (QPSK) with rate 1/2 can carry 96 information bits. Retransmissions for erroneous packets are not considered.

Real-time traffic, such as streaming video is used (proposed for WiMAX). This traffic is modeled as two superimposed Interrupted Renewal Process (2IRP) sources as proposed in [17]. Packets are dropped from the queue when the queuing delay exceeds 0.19 sec. Table III summarizes the simulation parameters.

#### IV. RESULTS

Performance indicators such as average packets dropped, average UT throughput, and average UT queuing delays have been considered to evaluate ATBFQ by comparing with the reference SB and RR schemes.

Figures 2, 3, and 4 show the performance results for average UT queuing delay, average packets dropped per frame, and the total sector throughput, respectively, in varying loading conditions for ATBFQ, SB, and RR. The curves are plotted for two different AF's of 0.5 and 0.7 to model moderate and high interference situations, respectively. ATBFQ outperforms the reference SB and RR algorithms in terms of the above mentioned performance parameters for all loading conditions when the AF is 0.5. In this case, the UTs experience better channel conditions. Hence fewer chunks are used to transmit data as compared to a higher AF. This is the reason why RR performs better than SB at a lower loading.

For low to medium loading with an AF of 0.7, we again observe that ATBFQ outperforms the reference schemes in terms of all observed parameters. This trend changes as network loading increases to 20 UTs per sector. In such a high loading case, SB outperforms ATBFQ and RR in terms of average UT queuing delay, average packets dropped per frame, and the total sector throughput, respectively. This is because SB is opportunistic in nature whereas ATBFQ tries to maintain fairness. As the number of UTs increase, SB takes advantage of the multi-user diversity to achieve higher throughput.

The CDF of the Jain's fairness index given by (3) is shown in Fig. 5. These curves represent network loading of 20 UTs per sector with an AF of 0.7. It is observed that ATBFQ is better in fairness as compared to SB and RR at all times.

Figure 6 shows the CDF plot of the normalized throughput given by (5) for 20 UTs per sector with an AF of 0.7. It is again observed that a higher normalized throughput is achieved for ATBFQ as compared to SB and the curve lies to the right of the above mentioned coordinates.

#### V. CONCLUSION

In this study, the performance of the ATBFQ scheduling algorithm with adaptive parameter selection is investigated in

TABLE III  
SUMMARY OF SIMULATION PARAMETERS

Parameter	Used Value/Model
Scenario	Wide area DL (frequency adaptive)
Channel model	WINNER C2 channel
Shadowing	Independent lognormal random variables (standard deviation 8dB)
Sector Tx antenna	120 <sup>o</sup> directional with baseline antenna pattern
UT receive antenna	Omni-directional
Inter-site distance	1000 meters
Signal bandwidth	15 MHz (i.e., 48 chunks which is 1/3 <sup>rd</sup> of the baseline assumptions)
Mobility	70 km/hr
Sector Tx power	46 dBm
Scheduler	Adaptive Token Bank Fair Queuing, Score Based, Round Robin
Interference model	Brute force method (central cell is considered with interference from the 1 <sup>st</sup> tier)
Antenna configuration	SISO
Coding	B-LDPC
AMC modes	BPSK (rate 1/2 and 2/3), QPSK (rate 1/2, 2/3, and 3/4), 16QAM (rate 1/2, 2/3, and 3/4), and 64QAM (rate 2/3 and 3/4)
AMC thresholds	With FEC block of 1728 bits and 10% BLER
Frame duration	0.6912 ms (scheduling interval)
Traffic model	1.9Mbps 2IRP model for MPEG video
Packet size	188 Bytes
Packet drop criterion	Delay > 0.19 sec
Simulation time	60 sec
Simulation tools	MATLAB, OPNET

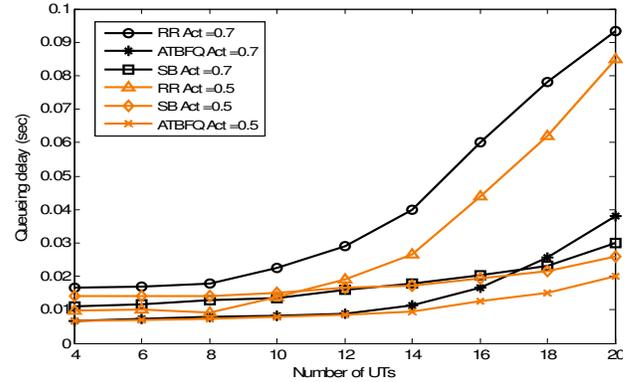


Fig. 2. Average UT queuing delay vs. number of UTs

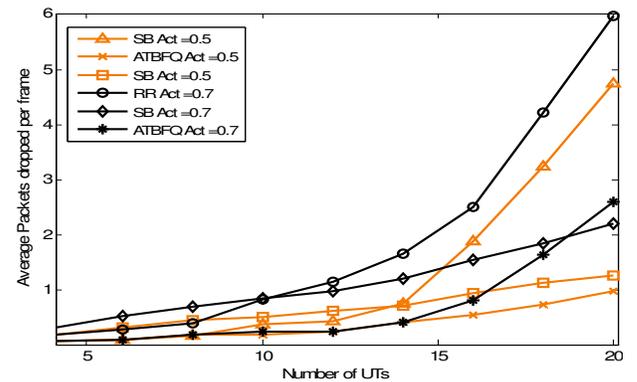


Fig. 3. Average UT packets dropped per frame vs. number of UTs

the context of WINNER wide area downlink scenario. This algorithm is refined to suit to the 4G wireless networks with OFDMA air-interface. The scheme is evaluated by comparing with the reference SB and RR schedulers. Being a queue and channel aware scheduling algorithm, ATBFQ always maintains fairness among UTs. In particular, ATBFQ outperforms the reference schemes in terms of throughput, queueing delay, and dropped packets for low to medium loading conditions in varying interference situations. The same trend is observed for high loading conditions when interference is low. At a high network loading with high interference, SB marginally outperforms

ATBFQ and RR with regard to throughput, queueing delay, and dropped packets. However, this loss in the aforementioned performance metrics is compensated by significantly better fairness performance the ATBFQ algorithm offers.

#### ACKNOWLEDGMENT

The authors would like to thank Mr. Jiangxin Hu for his technical support. They would also like to acknowledge OPNET Technologies Inc. for providing research license for the simulations.

#### REFERENCES

- [1] A. J. Demers, S. Keshav, and S. Shenker, "Analysis and simulation of a fair queueing algorithm," *Proc. ACM Symposium on Communications Architectures and Protocols (SIGCOMM'89)*, pp. 1–12, September 1989.
- [2] A. K. Parekh and R. G. Gallager, "A generalized processor sharing approach to flow control in integrated services networks: The single-node case," *IEEE/ACM Trans. on Networking*, vol. 1, pp. 344–357, March 1993.
- [3] A. Stamoulis and G. B. Giannakis, "Packet fair queueing scheduling based on multirate multipath-transparent cdma for wireless networks," in *Proc. IEEE Joint Conference of the Computer and Communications Societies (INFOCOM'00)*, vol. 3, pp. 1067–1076, March 2000.
- [4] Q. Liu, X. Wang, and G. B. Giannakis, "A cross-layer scheduling algorithm with QoS support in wireless networks," *IEEE Trans. on Vehicular Technology*, vol. 55, pp. 839–847, May 2006.
- [5] *Project Presentation*, WINNER Deliverable D8.1, Mar. 2004. [Online]. Available: <http://www.ist-winner.org>
- [6] D. Wu and R. Negi, "Effective capacity: A wireless link model for support of quality of service," *IEEE Trans. on Wireless Communications*, vol. 2, pp. 630–643, July 2003.
- [7] F. A. Bokhari, W. K. Wong, and H. Yanikomeroglu, "Adaptive token bank fair queuing scheduling in the downlink of 4G wireless networks," in *Proc. IEEE Vehicular Technology Conference (VTC'08)*, May.
- [8] W. K. Wong, H. Y. Tang, and V. C. M. Leung, "Token bank fair queuing: a new scheduling algorithm for wireless multimedia services," *International Journal of Communication Systems*, vol. 17, pp. 591–614, 2004.
- [9] T. Bonald, "A score-based opportunistic scheduler for fading radio channels," *European Wireless Conference*, February 2004.
- [10] *Final report on identified RI key technologies, system concept, and their assessment*, WINNER I Deliverable D2.10, Nov. 2005. [Online]. Available: <http://www.ist-winner.org>
- [11] R. Knopp and P. Humblet, "Information capacity and power control in single cell multiuser communications," in *Proc. IEEE Int. Conference on Communications (ICC'95)*, pp. 331–335, June 1995.
- [12] R. Jain, D. Chiu, and W. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer systems," *DEC Research Report TR-301*, Digital Equipment Corporation, September 1984.
- [13] H. Sirisena, A. Haider, M. Hassan, and K. Fawlikowski, "Transient fairness of optimized end-to-end window control," in *Proc. IEEE Global Communications Conference (GLOBECOM'03)*, pp. 3979–3983, December 2003.
- [14] G. Berger-Sabbatel, A. Duda, O. Gaudoin, M. Heusse, , and F. Rousseau, "Fairness and its impact on delay in 802.11 networks," in *Proc. of the IEEE Global Communications Conference (GLOBECOM'04)*, pp. 2967–2973, December 2004.
- [15] *Evaluation Methodology Document*, IEEE 802.16 Work-in-progress document C802.16m-07/080r2, 2007. [Online]. Available: <http://ieee802.org/16>
- [16] *Test Scenarios and Calibration Cases Issue 2*, WINNER II Deliverable D6.13.7, Dec. 2006. [Online]. Available: <http://www.ist-winner.org>
- [17] *Traffic model for 802.16 TG3 MAC/PHY simulations*, IEEE 802.16 Work-in-progress document 802.16.3c-01/30r1, Mar. 2001. [Online]. Available: <http://ieee802.org/16>

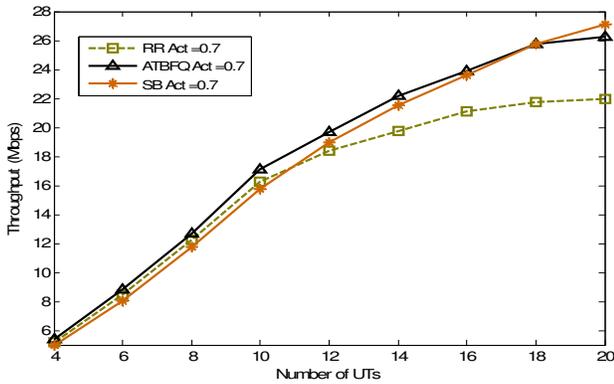


Fig. 4. Sector throughput

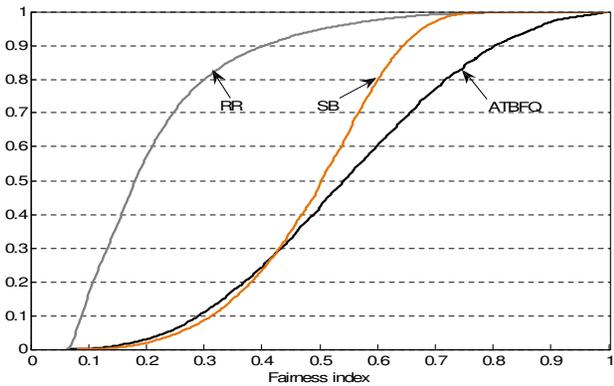


Fig. 5. CDF of fairness index

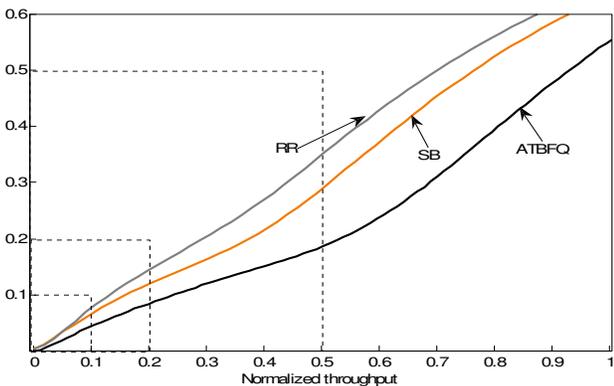


Fig. 6. CDF of normalized throughput (zoomed in)