ANALYTICAL STUDY OF BROADCASTING IN 802.11 AD HOC NETWORKS

Andrey Lyakhov and Vladimir Vishnevsky
Institute for Information Transmission Problems, RAS
B. Karetny 19, Moscow, 127994, Russia

Pavel Poupyrev
ATR Adaptive Communications Research Laboratories
2-2-2 Hikaridai, Keihanna Science City, Kyoto, Japan

Keywords: Broadcasting, IEEE 802.11, Markov models, performance study.

Abstract: IEEE 802.11 technology becomes attractive with the implementation of various broadcast applications. For these applications, one of the main performance indices is the mean notification time namely the interval between consecutive successful receipts of the same source's packets. In this paper, we develop an analytical method to study the ad hoc 802.11 network performance with broadcasting stations. The method based on a Markov model allows the estimation of the mean notification time and the optimization of the packet generation time. The developed method has been validated by simulations and has demonstrated a high accuracy of notification time estimation as well as the its efficiency in broadcasting optimization.

1 INTRODUCTION

IEEE 802.11 (IEEE 802.11, 1999) is one of the most popular technologies for wireless ad hoc and mobile networking. The fundamental access mechanism in the IEEE 802.11 protocol is the Distributed Coordination Function (DCF), which implements the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) method.

In the majority of previous related papers, the 802.11 DCF performance was studied with unicast transmissions: see (Bianchi, 2000; Cali et al., 2000; Vishnevsky et al., 2002), for example. However, the 802.11 standard specifies both unicast transmissions, and multicast or broadcast ones. Broadcasting in ad hoc 802.11 networks is of our main interest in the paper.

As an example of such broadcast applications, we can consider the Whizbe project (Poupyrev et al., 2002) developed in ATR, Japan. Whizbe is an application for discovering resources such as users, product information discounts and etc. in ad hoc networks using broadcasting. The main goal of the application is to provide mutual presence awareness for acquaintances that are in close physical proximity. Whereas an online friend list application tells if a friend or acquaintance is available for online chat, Whizbe lets the user know if a friend or acquaintance is in physical proximity to facilitate a physical meeting or other interaction. A number of PDAs supporting such sort of message broadcasts and located within the same line-of-sight area can be large and reach hundreds and thousands (for example, in a stadium).

Another type of application, where 802.11 ad hoc broadcasting can be successfully used, is wireless sensor networks with distributed control. For example,

- Wireless home temperature sensors, each of them transmitting information to receivers of a climatic system, fire fighting service and the home owner work station (Cheng et al., 2002);
- Gas escape sensors in gas pipelines, where measured data are to be delivered to repair service controllers. To improve reliability, several data receivers are exploited in these systems, what makes broadcasting attractive.

In such broadcast systems, the key point is to optimize the mean message generation time, that is, the time interval separating the instances when a data source puts its data packets into the broadcast device's queue. As the optimization criterion, we can consider the mean notification time namely the interval between consecutive successful receipts of the same source's packets. Obviously, the 802.11 DCF peculiarities described below will affect the performance index.

In the 802.11 DCF, no ARQ mechanisms are used

Lyakhov A., Vishnevsky V. and Poupyrev P. (2005).
ANALYTICAL STUDY OF BROADCASTING IN 802.11 AD HOC NETWORKS.
In *Proceedings of the Second International Conference on e-Business and Telecommunication Networks*, pages 86-91
DOI: 10.5220/0001410000860091

with packet broadcasting. A device starts transmitting a DATA frame containing a broadcast packet with the following conditions:

1) The channel was free for DIFS (Distributed InterFrame Space) since the latest transmission in the network;

2) The device's backoff time has been expired;

3) There are packets to be transmitted in the device's queue.

In particular, when a packet arrives a device either starts the packet transmission immediately, if the channel is free and the device is in the idle state at the arrival time, or switches to the backoff state otherwise. We call these immediate transmissions the *asynchronous* ones to distinguish them from other (*synchronous*) transmissions carried out after the backoff. After passing to the backoff state, the backoff counter is reset to the initial value b, which is called the backoff time, measured in units of backoff slots of duration σ, and chosen uniformly from a set $(0, \ldots, W-1)$. W is called the contention window and it does not depend on the number of retries (contrary of unicast transmissions), because broadcast transmissions are not acknowledged and hence there are no retries at the MAC layer.

Backoff intervals are reckoned only as long as the channel is free: the backoff counter is decreased by one only if the channel was free for the entire previous slot. Counting the backoff slots stops when the channel becomes busy, and backoff time counters of all stations decrement only when the channel is sensed idle for DIFS. When the backoff counter attains its zero value, the station starts transmission. A collision happens when backoff counters of two or more stations are zeroed simultaneously. All packets of stations involved in the collision are lost.

A device that has completed a packet transmission returns to the backoff state. If the device's queue appears to be empty at the end of the backoff time, the device becomes idle.

Further, in Sections 2 and 3 we develop an analytical model, which considers all significant DCF features and provides the mean notification time estimation. In Section 4, we give some numerical research results obtained by both our analytical method and simulations, which allows us to validate the developed method. The obtained results are summarized in Section 5.

2 ANALYTICAL MODEL

Let us consider an ad hoc 802.11 network of N statistically homogeneous stations generating broadcast packets of the same fixed length with an identical rate $\lambda \ll (N\sigma)^{-1}$. Generation intervals are assumed to be distributed exponentially. Every station's queue can contain no more than B packets. The channel is ideal, that is, there are no noise-induced distortions. Signal propagation time is assumed to be negligible.

The analytical model to be developed is intended for the determination of the optimal generation rate λ_{opt}, when the mean notification time T_{not} is minimal.

As in (Bianchi, 2000) and (Vishnevsky et al., 2002), let us subdivide the time of the network operation into non-uniform virtual slots such that:

- Every station with non-empty queue changes its backoff counter at the start of a virtual slot and begins a synchronous transmission if the counter value becomes zero.
- If a transmission happens within a given virtual slot, the slot is ended by the DIFS closing the transmission.
- If nobody transmits during σ since the current slot beginning, this slot duration is equal to σ.

Such a virtual slot is either (a) an "empty" slot σ in which no station transmits, or (b) a "synchronous" slot when one or more stations transmit synchronously or (c) an "asynchronous" slot when a station transmits asynchronously. Note that an asynchronous transmission can be considered as always successful since, firstly, it can be performed only in a slot when no other stations transmit synchronously (otherwise it will sense the channel busy at the slot beginning and defer from its transmission), and secondly, we can neglect the probability that two or more packets arrive to queues for σ because $\lambda \ll (N\sigma)^{-1}$.

Following (Bianchi, 2000), we describe a station state by the couple (i, k), where i = 0, 1 is an indicator of non-empty queue and k = 0, …, W-1 is a backoff counter value, and a station behavior by the Markov chain shown in Fig.1. In the chain, all transitions happen at virtual slot borders.

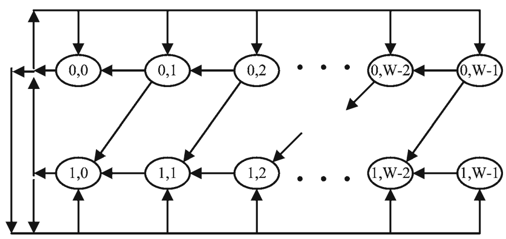

Figure 1: Markov chain.

State (0,0) corresponds to the idle state of the considered station. In states (0, 1)…(0,W-1), the station is in the backoff state, but its queue is empty. At last, (1, 0),…, (1,W-1) are the states when the station's queue is not empty.

Let us introduce the following notation:

$P\{i_2,k_2|i_1,k_1\}$ is the probability of one-step transition from (i_1,k_1) to (i_2,k_2).

P_0 is the probability that a queue becomes empty after completing its synchronous transmission.

P_S is the probability that at least one packet arrives in the considered station's queue for a virtual slot under the condition that the queue was empty at the slot beginning. Obviously, the probability consists of two items: $P_S = P_S^F + P_S^E$, where P_S^F and P_S^E are the arrival probabilities for non-empty and empty slots, respectively, under the same condition as above.

P_T is the probability that at least one packet arrives in the considered station's queue for a packet transmission time, including the closing DIFS.

To find stationary probabilities $\alpha(i,k)$ of the states in Fig. 1, let us define non-null transition probabilities:

$P\{1,k|1,k+1\} = 1, \quad k \in [0, W-2]$: backoff counter decrement, the queue is not empty.

$P\{1,k|1,0\} = \frac{1-\overline{P}_0}{W}, \quad k \in [0, W-1]$: synchronous transmission of the station, after which its queue remains to be non-empty. Here, $\overline{P}_0 = P_0 e^{-\lambda \text{DIFS}}$ is the probability that a queue becomes empty after completing its synchronous transmission and keeps to be empty after DIFS closing the transmission.

$P\{0,k|1,0\} = \frac{\overline{P}_0}{W}, \quad k \in [0, W-1]$: synchronous transmission of the station, after which its queue becomes empty.

$P\{1,k|0,k+1\} = P_S, \quad k \in [0, W-2]$: backoff counter decrement, and a packet arrives in the station's empty queue.

$P\{0,k|0,k+1\} = 1 - P_S, \quad k \in [0, W-2]$: backoff counter decrement, the queue remains to be empty.

$P\{1,k|0,0\} = \frac{P_S^F + P_S^E P_T}{W}, \quad k \in [0, W-1]$: the station switches from the idle state to the backoff state. The switch takes place, if either a packet arrives when the channel was busy, or one more packet arrives during the station's asynchronous transmission.

$P\{0,k|0,0\} = \frac{P_S^E(1-P_T)}{W}, \quad k \in [1, W-1]$: the transition corresponds to the station's asynchronous transmission, after which the station's queue is empty and the backoff counter $b = k > 0$.

$P\{0,0|0,0\} = 1 - P_S + \frac{P_S^E(1-P_T)}{W}$: there are no packet arrivals in the queue, or the station's asynchronous transmission takes place, after which the station's queue is empty and the backoff counter $b = 0$.

Now we can determine stationary probabilities $\alpha(i,k)$. Using global balance equations and the normalizing condition $\sum\limits_{(i,k)} \alpha(i,k) = 1$, we obtain:

$$\alpha(0,0) = \left\{ 1 - P_S + \frac{W+1}{2} \times \left(P_S + \frac{\frac{W P_S^2}{1-(1-P_S)^W} - P_S^E(1-P_T)}{\overline{P}_0} \right) \right\}^{-1}, \tag{1}$$

$$\alpha(1,0) = \frac{\frac{W P_S^2}{1-(1-P_S)^W} - P_S^E(1-P_T)}{\overline{P}_0} \cdot \alpha(0,0), \tag{2}$$

$$\alpha(0,j) = P_S \cdot \frac{1-(1-P_S)^{W-j}}{1-(1-P_S)^W} \cdot \alpha(0,0), \quad j = 1, \ldots, W-1, \tag{3}$$

$$\alpha(1,j) = \frac{W-j}{W}\left[P_S \cdot \alpha(0,0) + \alpha(1,0)\right] - \alpha(0,j), \quad j = 1, \ldots, W-1, \tag{4}$$

Let τ be the probability of the station's synchronous transmission for a virtual slot. Similarly to (Bianchi, 2000; Cali et al., 2000), we assume that the probability depends neither on the previous history, nor on the behavior of other stations. Obviously, $\tau = \alpha(1,0)$. Moreover, the probability of synchronous transmission failure due to a collision is $P_C = 1-(1-\tau)^{N-1}$, and the probability of the station's asynchronous transmission for a virtual slot is $\tau_a = \alpha(0,0)P_S^E$, where $P_S^E = (1-\tau)^{N-1}[1-\exp\{-\lambda\sigma\}]$ since we have neglected the case, when two or more packets arrive for σ.

Let us find durations of non-empty slots:

- "Synchronous" slot: $t_S = t_P$+DIFS, where t_P is a DATA frame transmission time;
- "Asynchronous" slot: the mean duration is $t_A = \frac{\sigma}{2} + t_P$+DIFS.

The probability P_T that at least one packet arrives in the station's queue for t_S is $P_T = 1-\exp(-\lambda t_S)$.

To find the probability P_S^F, we need to know probabilities of all possible kinds of slots under the condition that the considered station does not transmit in the slot. Under the condition, the "empty" slot probability is $Q_E = (1-\tau-\tau_a)^{N-1}$, the "synchronous" slot probability is $Q_S = 1-(1-\tau)^{N-1}$, and the "asynchronous" slot probability is $Q_A = 1 - Q_E - Q_S$. So we have

$$P_S^F = (Q_S + Q_A) P_T.$$

To complete the model definition, it remains to find probability P_0 that a queue becomes empty after finishing the current packet service. A packet service starts at the instance of either the packet arrival in an empty queue or completing the transmission of the previous packet in the queue. If a packet arrives in an empty queue and is transmitted asynchronously, we deal with the asynchronous service case, when a packet is served for the fixed time t_P. With synchronous service, the service time is random.

Let us describe a device's queue change process by the auxiliary model shown in Fig. 2. With the model, we assume that a packet arriving at a device not engaged in servicing other packets is serviced asynchronously (and hence successfully) with probability p_a. Otherwise, the packet is put into a queue of size B. After standing in the queue, it is serviced synchronously for a random time distributed exponentially with the average value T_S.

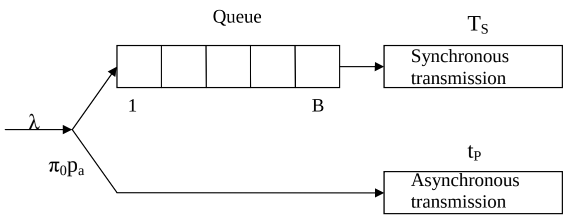

Figure 2: Queue change process.

Under these assumptions, we can use a birth-and-death process to describe how the length i of the synchronously serviced packets queue changes. Since the birth rate is $\lambda_0 = (1 - p_a)\,\lambda$ with $i = 0$ and λ with $0 < i < B$, while the the death rate is $1/T_S$, we obtain the following formula for the steady-state probabilities:

$$\pi_i = \pi_0 \lambda_0 \lambda^{i-1} T_S^i, \quad i = 1, \ldots, B. \tag{5}$$

Obviously, π_B is the probability of packet rejection because of buffer overflow. Therefore, the probability that a queue becomes empty after completing its synchronous transmission is equal to $P_0 = \pi_1/(1 - \pi_0)$. Moreover, since

$$\pi_0 = \frac{1}{1 + (1 - p_a) \sum\limits_{i=1}^{B} (\lambda T_S)^i}, \tag{6}$$

we obtain

$$P_0 = \frac{1}{\sum\limits_{i=1}^{B} (\lambda T_S)^{i-1}} = \frac{1 - \lambda T_S}{1 - (\lambda T_S)^B}. \tag{7}$$

In the next section, we will estimate the probability p_a and the average time T_S of synchronous service, as well as the main performance index T_{not}.

3 ESTIMATION OF PERFORMANCE INDICES

Let us subdivide all packets, which arrive at some station A during all possible virtual slots identified by couple (i, k) and are serviced synchronously, into the following 4 categories:

1. Packets arriving for slots $(1, k)$;
2. Packets arriving for slots $(0, k > 0)$;
3. Packets arriving for slot (0,0), when a device different from A transmits;
4. Packets arriving during an asynchronous transmission of device A.

For each of the categories, we will count the average number n_i of these packets arriving for a virtual slot and the average number n_i^0 of these packets arriving for a virtual slot in empty queue. It is easy to show that

$$T_S = \left[(T_S^* + \text{DIFS}) \sum_{i=1}^{4} \left(n_i - n_i^0 \right) + \sum_{i=1}^{4} T_S^i n_i^0 \right] / \\ / \sum_{i=1}^{4} n_i, \tag{8}$$

$$p_a = \tau_a / \left(\tau_a + \sum_{i=1}^{4} n_i^0 \right), \tag{9}$$

where

$$T_S^* = \frac{W-1}{2} t_{VS} + t_P$$

and $t_{VS} = Q_E \sigma + Q_S t_S + Q_A t_A$ is the average duration of a virtual slot, when the given device A does not transmit. Moreover, T_S^i is the average service time for category i packets arriving in empty queue.

To estimate T_S and p_a, using (8) and (9), it remains to find n_i, n_i^0 and T_S^i for each of introduced categories. Let us do it.

Category 1. In the case, packets can arrive in empty queue of the considered station A only for the DIFS closing the station's transmission and only, if the transmitted packet was the last in the queue. So we have

$$n_1^0 = \left(1 - e^{-\lambda \cdot \text{DIFS}}\right) P_0 \cdot \alpha(1, 0),$$

$$n_1 = \lambda t_{VS} \sum_{k=1}^{W-1} \alpha(1, k) + \lambda t_S \cdot \alpha(1, 0),$$

and

$$T_S^1 = T_S^* + \frac{\text{DIFS}}{2}.$$

Category 2. Since only the first of packets arriving in the queue for each of the slots can find the queue empty, we have

$$n_2^0 = Q^* \sum_{k=1}^{W-1} \alpha(0, k),$$

$$n_2 = \lambda t_{VS} \sum_{k=1}^{W-1} \alpha(0,k),$$

and

$$T_S^2 = t_P + \frac{t_{VS} Q^*}{n_2^0} \sum_{k=1}^{W-1} \left(k - \frac{1}{2}\right) \alpha(0,k),$$

where

$$Q^* = Q_E \left(1 - e^{-\lambda\sigma}\right) + Q_S \left(1 - e^{-\lambda t_S}\right) + Q_A \left(1 - e^{-\lambda t_A}\right).$$

Category 3. For this case, we have

$$n_3^0 = \left[Q_S \left(1 - e^{-\lambda t_S}\right) + Q_A \left(1 - e^{-\lambda t_A}\right)\right] \alpha(0,0),$$

$$n_3 = \lambda \left(Q_S t_S + Q_A t_A\right) \alpha(0,0),$$

$$T_S^3 = T_S^* + \frac{Q_S t_S + Q_A t_A}{2\left(1 - Q_E\right)}.$$

Category 4. As for Category 2, in this case we should distinguish the first of these arriving packets and others. So

$$n_4^0 = P_S^E \left(1 - e^{-\lambda t_S}\right) \alpha(0,0) = P_T \tau_a,$$

$$n_4 = \tau_a \lambda t_S, \quad T_S^4 = T_S^* + \frac{t_S}{2}.$$

Finally, taking into account of asynchronous transmissions, we obtain the mean notification time:

$$T_{not} = \frac{1}{\lambda\left[\pi_0 p_a + \left(1 - \pi_0 p_a\right)\left(1 - P_C\right)\left(1 - \pi_B\right)\right]}. \tag{10}$$

We adopt the following iterative procedure to calculate this performance index.

Step 0. Define an initial value for P_0.

Step 1. Define initial values for transmission probabilities τ and τ_a.

Step 2. Calculate, firstly, probabilities P_S^F and P_S^E and then the steady-state probabilities $\alpha(i,k)$. Using $\alpha(0,0)$ and $\alpha(1,0)$, calculate modified values for transmission probabilities. If differences of initial and modified values are greater than a predefined limit Δ_t, repeat this Step, using new initial values for τ and τ_a – the half-sums of their old initial values and the modified values.

Step 3. Calculate, firstly, T_S and p_a, using (8) and (9), secondly, the steady-state probabilities π_i and then the modified value of probability P_0. If the difference of initial and modified values of P_0 is greater than a predefined limit Δ_P, return to Step 1.

Step 4. Calculate the mean notification time by (10).

We do not prove exactly the convergence of this iterative technique due to its complexity. In practice, numerous examples of adopting the technique with various values of network and traffic parameters have shown that this technique provides very fast convergence to the solution and high speed of T_{not} calculation.

4 NUMERICAL RESULTS

To validate our model, we have compared its results with that obtained by GPSS (General Purpose Simulation System) simulations (Schriber, 1974; GPSS World, 1998). The object of our numerical investigations was an ad hoc 802.11b (with Short Preamble) network (IEEE 802.11b, 1999) of N statistically homogeneous stations generating broadcast packets of the same length L. The every station's buffer size B was chosen equal to 100 packets. In our simulation model, we took into account of all real features of the 802.11 MAC protocol and did not adopt the assumptions used in analytical modelling and described in Section 2. In each run (it took about an hour on the average) of the simulation model, we observed values of the measured performance indices and stopped the simulation when their fluctuations became quite small (within 0.5%).

In Fig. 3 we show how the mean notification time depends on the average generation time $T_{gen} = \lambda^{-1}$ with the packet length $L = 1000$ bytes that corresponds to the DATA transmission time $t_P = 850$ μs. The dependency has been obtained analytically (the "Math" curve) and validated by simulation (the "GPSS" circles). In particular, the comparison has shown a high accuracy of the analytical model: the errors never exceed 5% with the mean notification time estimation.

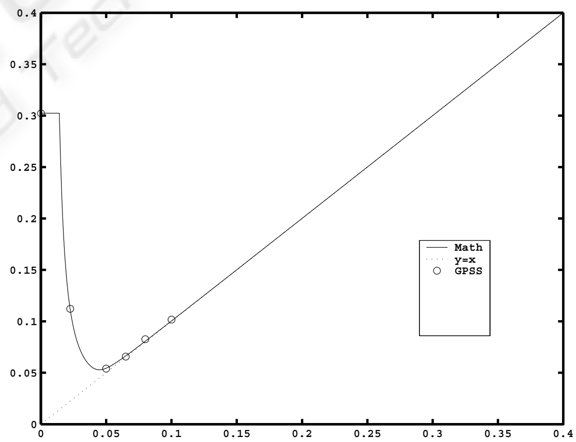

Figure 3: Mean notification time T_{not} (s) versus the average generation time T_{gen} (s) with $L = 1000$ bytes and $N = 50$: Math – analytical method; GPSS – simulation.

The curve $T_{not}(T_{gen})$ in Fig. 3 has a clear minimum at $T_{gen} = T_{gen}^{opt} \approx 0.48$ s. When T_{gen} exceeds T_{gen}^{opt}, most of packets are transmitted asynchronously (and hence, successfully) and the mean notification time is approximately equal to T_{gen}. However, in this case there are wide gaps between packet arrivals, and the network capacity is used ineffectively. With high generation rates ($T_{gen} < T_{gen}^{opt}$), most of packets are transmitted synchronously, what causes frequent collisions and hence T_{not} growth. Moreover, there is a threshold (≈ 0.015 s in Fig. 3) of T_{gen}, below which the network becomes saturated and buffers are almost always overflowed. In this saturation area, T_{not} does not change.

Fig. 4 shows dependencies $T_{not}(T_{gen})$ for various packet lengths. It proves that the optimal generation rate depends essentially on the packet length (as well as on the number N of stations – these results are not shown). Concerning the form of this dependency, it appears that the optimal average generation time depends on both the packet length and the number N of stations almost linearly.

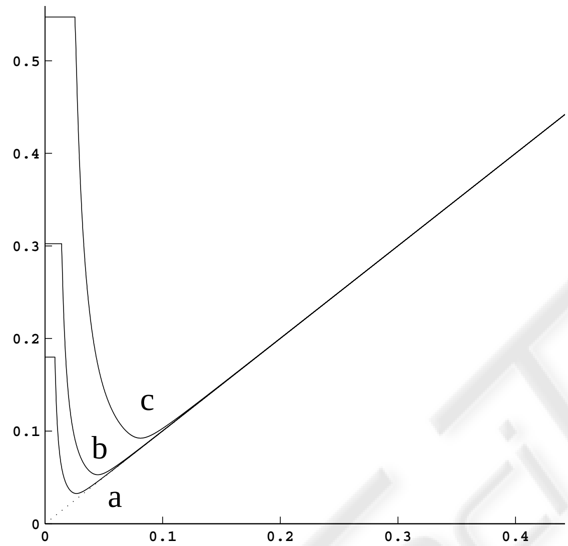

Figure 4: Mean notification time T_{not} (s) versus the average generation time T_{gen} (s) with $N = 50$ and various packet lengths L: (a) $L = 500$ bytes; (b) $L = 1000$ bytes; (c) $L = 2000$ bytes.

5 CONCLUSION

In this paper, we have developed a novel analytical method to study the performance of an ad hoc 802.11 network of stations generating broadcast packets. The method based on Markov models allows estimating the main performance index of such a network: the mean notification time namely is the interval between consecutive successful receipts of the same source's packets. Using the method, we succeeded in optimizing the packet generation time to minimize the mean notification time.

According to numerical results obtained by both the developed method and simulations, our method is quite exact: the errors never exceed 5% with the notification time estimation. This method provides a high speed of calculating the values of performance indices, which has allowed us to perform the exhaustive search of optimal packet generation time.

As a future research activity in this direction, we propose extensions of this method to take into account the cases when (a) the wireless channel is not ideal; and (b) broadcasting stations can be hidden from each other.

ACKNOWLEDGMENT

This work was partially supported by Russian Science Support Foundation.

REFERENCES

ANSI/IEEE Std 802.11 (1999). *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.* IEEE Press, New York, 1999 edition.

Bianchi, G. (2000). Performance Analysis of the IEEE 802.11 Distributed Coordination Function. In *IEEE Journal on Selected Areas in Communications* 18(3):535-547.

Calí, F., Conti, M., Gregory E. (2000). Dynamic Tuning of the IEEE 802.11 Protocol to Achieve a Theoretical Throughput Limit. In *IEEE/ACM Transactions on Networking* 8(6):785-799.

Vishnevsky V.M., Lyakhov A.I. (2002). IEEE 802.11 Wireless LAN: Saturation Throughput Analysis with Seizing Effect Consideration. In *Cluster Computing* 5(2):133-144.

Poupyrev P., Kosuga M., Davis P. (2002). Analysis of Wireless Message Broadcast in Large Ad Hoc Networks of PDAs. In *Proceedings of Forth IEEE conference on Mobile and Wireless Communications Networks*, pp. 299-303.

Cheng L., Marsic I. (2002). Piecewise Network Awareness Service for Wireless/Mobile Pervasive Computing. In *Mobile Networks and Applications (MONET)* 7(4):299-303.

Schriber, T.J. (1974). *Simulation using GPSS.* John Wiley & Sons.

GPSS World (1998). *www.minutemansoftware.com*.

IEEE 802.11b: Supplement to ANSI/IEEE Std 802.11 (1999). (IEEE 802.11, 1999). *Higher-Speed Physical Layer Extension in the 2.4 GHz Band.* IEEE Press, New York.