

Bluetooth Time Division Duplex – Analysis as a Polling System

Gil Zussman, Adrian Segall
Dept. of Electrical Engineering
Technion–Israel Institute of Technology
Haifa 32000, Israel
{gilz@tx, segall@ee}.technion.ac.il

Uri Yechiali
Dept. of Statistics and Operations Research
School of Mathematical Sciences, Tel Aviv University
Tel Aviv 69978, Israel
uriy@post.tau.ac.il

Abstract—Efficient communication in Bluetooth networks requires design of intra and inter-piconet scheduling algorithms, and therefore numerous algorithms have been proposed. However, due to complexities of the Bluetooth MAC, the performance of these algorithms has been analyzed mostly via simulation. We present exact analytic results regarding the exhaustive, gated, and limited (pure round robin) scheduling algorithms in piconets with unidirectional traffic. We show that, surprisingly, in symmetrical piconets with only uplink traffic, the mean waiting time is the same for the exhaustive and limited algorithms. This observation is extended for Time-Division-Duplex systems with arbitrary packet lengths. Furthermore, we show that the mean waiting time in a piconet with only uplink traffic is significantly higher than its corresponding value in a piconet with only downlink traffic. We demonstrate the difficulties in analyzing the performance of the exhaustive and gated algorithms in a piconet with bi-directional traffic. Finally, we numerically compare the exact results to approximate results, presented in the past.

I. INTRODUCTION

Bluetooth is a Personal Area Network (PAN) technology, which enables devices to connect and communicate wirelessly via short-range ad-hoc networks [2]. The basic network topology (referred to as a *piconet*) is a collection of slave devices operating together with one master. A multihop ad hoc network of piconets in which some of the devices are present in more than one piconet is referred to as a *scatternet*. A device that is a member of more than one piconet (referred to as a *bridge*) must schedule its presence in all the piconets in which it is a member.

The master uses *intra-piconet* scheduling algorithms to schedule the traffic within a *piconet*. *Inter-piconet* scheduling algorithms are used to schedule the presence of the bridges in different piconets. Numerous *intra* and *inter-piconet* scheduling algorithms have been proposed (see [3],[5],[8],[9],[20], and references therein).

Analytical performance evaluation of intra and inter-piconet scheduling algorithms has great importance, since it may provide insight on their design and optimization. However, as mentioned in [3], due to the special characteristics of the Bluetooth Medium Access Control (MAC) which is based on Time-Division-Duplex (TDD), the performance of these algorithms has been analyzed mostly via simulation. *In this paper we present overlooked connections between Bluetooth*

*piconets and polling systems*¹. We show that these connections can be utilized in order to obtain (in a straightforward manner) analytic results regarding the performance of the algorithms.

We show that a piconet with *unidirectional uplink* traffic operated according to the *exhaustive scheduling algorithm* is equivalent to a specific *exhaustive polling system* and derive exact analytic results regarding intra-piconet waiting times.² We also show that a piconet with *unidirectional uplink* traffic operated according to the *limited (pure round robin) scheduling algorithm* can be modeled as a *1-limited polling system* and derive exact results.

Following this analysis, a surprising result is obtained: the mean waiting times for the *limited and exhaustive* algorithms are equal in a piconet with only uplink traffic, where all arrival rates are statistically equal. Moreover, the mean waiting time when such a piconet is operated according to the gated algorithm is higher than in the exhaustive or limited algorithms. These observations are extended to arbitrary Time-Division-Duplex systems, where the packets are not necessarily 1, 3, and 5 slots long (as required in Bluetooth [2]). This extension is important, since the TDD mechanism will be used by other technologies. For example, 3.5G and 4G cellular systems are expected to use a combination of TDD and CDMA [6].

It is shown that the result regarding the equality of the mean waiting times is a specific case of a result that holds in polling systems. This observation yields a decomposition result for the mean waiting time in symmetrical 1-limited polling systems.

Furthermore, we show that a piconet with *unidirectional downlink* traffic operated according to the *exhaustive scheduling algorithm* is equivalent to an exhaustive polling system with *zero-switchover* periods³. A similar equivalence holds for the gated and the limited algorithms. It is shown that the mean waiting time in a piconet with only uplink traffic is significantly higher than in a piconet with only downlink

¹A polling system consists of several queues served by a single server according to a set of rules (polling scheme) [1, p. 195],[18].

²If analytic results are exact under the assumption of a Poisson arrival process, we refer to them as *exact results*.

³In a polling system, the time required for the server to shift from one queue to another is referred to as the switchover time.

traffic.

We argue that when the traffic is *bi-directional*, it seems that there is no closed form expression for the Probability Generating Function of the time to exhaust the two related queues, at a given slave and the master. Finally, we note that approximate results regarding the performance of various intra and inter-piconet scheduling regimes have been recently presented (e.g. [11],[15],[16],[17]). We conclude by numerically comparing our exact results to the approximate results presented in [15] and [16].

In [22] we have derived *exact* analytic results for *symmetrical* piconets operated according to the *limited* algorithm. To the best of our knowledge, the results presented in this paper and in [22] are the only available *correct exact* analytic results regarding the performance of Bluetooth scheduling algorithms. The results presented in [22] have been extended by Miorandi and Zanella for an asymmetrical arrival process [12] and for fading channels [13]. Similarly, we argue that the results presented in this paper, regarding specific scenarios (e.g. Poisson arrival process), can be easily extended to different scenarios by utilizing the vast amount of research regarding polling systems.

The rest of the paper is organized as follows. Section II gives a brief introduction to the Bluetooth technology, while Section III presents the model. In Section IV we analyze the scheduling algorithms in piconets and in general TDD systems with unidirectional uplink traffic. In Section V we discuss general polling systems and present a simple decomposition result. In Section VI we analyze piconets with unidirectional downlink traffic. Section VII analyzes the gated algorithm in a piconet with bi-directional traffic and Section VIII compares numerical results to results obtained in the past. Section IX summarizes the main results.

II. BLUETOOTH TECHNOLOGY

In a *piconet* one unit acts as a *master* and the others act as *slaves* (a master can have up to 7 slaves). Bluetooth channels use a Frequency-Hop/Time-Division-Duplex (FH/TDD) scheme in which the time is divided into 625- μ sec intervals called *slots*. The master-to-slave transmission starts in even-numbered slots, while the slave-to-master transmission starts in odd-numbered slots. Masters and slaves are allowed to send 1, 3 or 5-slot *packets*, which are transmitted in consecutive slots. Packets can carry synchronous information (voice link) or asynchronous information (data link).⁴ Information can only be exchanged between a master and a slave.

A slave is allowed to start transmission in a given slot, if the master has addressed it in the preceding slot. The master addresses a slave by sending a data packet or, if it has no data to send, a 1-slot *POLL packet*. The slave must respond by sending a data packet or, if it has nothing to send, a 1-slot *NULL packet*. We refer to the master-to-slave communication as *downlink* and to the slave-to-master communication as

⁴We concentrate on networks in which only data links are used.

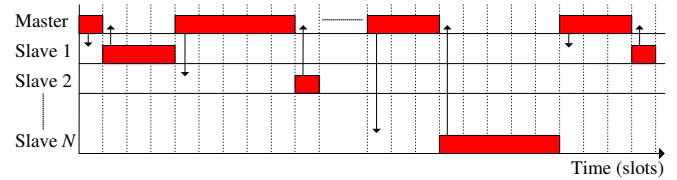


Fig. 1. An example of the TDD scheme in a Bluetooth piconet.

uplink. An example of the TDD scheme in a piconet with N slaves is given in Fig. 1.

The master schedules the traffic in a *piconet* according to an intra-piconet scheduling algorithm. We focus on the following algorithms in which the master communicates with the slaves according to a *fixed cyclic order*:

- Limited (Pure) Round Robin – At most a single packet is sent in each direction (downlink and uplink) whenever a master-slave queue pair is served.
- Exhaustive Round Robin – The master does not switch to the next master-slave queue pair until both the downlink and the uplink queues are empty.
- Gated Round Robin – Only the packets that are found in the uplink and downlink queues when the master starts serving the queue pair are transmitted.

III. THE MODEL

The number of slaves is denoted by N and we assume that each node has an infinite buffer. We assume that the packets are generated at the uplink and downlink queues according to a Poisson arrival process. Since the arrival process in real networks is not Poisson, throughout the paper we briefly point out the additional steps that have to be taken in order to analyze systems with a compound Poisson arrival process (i.e. systems in which *batches* of packets arrive according to a Poisson arrival process).

We consider different packet generation scenarios:

- *Symmetrical piconet* – The arrival rate to every downlink and uplink queue is λ (packets/slot).
- *Half-symmetrical piconet* – The arrival rate to all the downlink queues is the same (denoted by λ_d). Similarly, the arrival rate to all the uplink queues is λ_u , but $\lambda_d \neq \lambda_u$.
- *Asymmetrical piconet* – The arrival rate to the uplink queue at slave i is λ_u^i and the arrival rate into the master of packets intended for slave i is λ_d^i .

We assume that the master is the final destination of all packets generated at the slaves. The probabilities of a packet length being 1, 3, or 5 slots are p_1 , p_3 , and p_5 , respectively.⁵ The mean and second moment of the packet length are denoted by \bar{L} and \bar{L}^2 . The *waiting time* is the time a packet waits in the uplink or the downlink queue before it is served. The mean waiting times in the uplink and downlink queues are

⁵Although we assume that the packet lengths are random, in practice, these length depend on the Segmentation and Reassembly (SAR) of higher layer packets (see the discussion in [5]). The SAR policy can also affect the arrival process (i.e. in practice, it is likely that batches of packets will arrive at once).

denoted by \overline{W}_u and \overline{W}_d , respectively. In case the piconet is asymmetrical, the mean waiting time in the uplink queue of slave i is denoted by \overline{W}_u^i .

Some of the scheduling algorithms proposed in the past (e.g. [5]) assume that the master has some information regarding the status of the slaves' queues. However, obtaining such information requires changing the Bluetooth specifications [2] or using a proprietary algorithm in all the devices participating in a piconet. Thus, we assume that the master does not have any information about the state of the uplink queues. This assumption complies with the assumptions made in several previous analyses of intra-piconet scheduling algorithms (e.g. [3],[11],[14],[15],[16],[17]).

We note that whenever we refer to results regarding general (non-Bluetooth) symmetrical polling systems we follow the notation of Takagi [18]. Namely, the polling system is composed of N queues served by a single server. The packet arrival process to each queue is Poisson with intensity λ . The mean and second moment of the packet service times are denoted by b and $b^{(2)}$. The mean and variance of the switchover times are denoted by r and δ^2 .

IV. UPLINK TRAFFIC

A. Analysis of the Exhaustive Algorithm

Consider a piconet with only *uplink* traffic operated according to the *exhaustive* scheduling algorithm. First, we analyze a *half-symmetrical* piconet (i.e. a piconet in which $\lambda_u = \lambda > 0$ and $\lambda_d = 0$). Then, we show that an *asymmetrical* piconet (in which the arrival rates to the uplink queues are not necessarily equal) can be analyzed in a similar manner.

Since $\lambda_d = 0$, when the master communicates with a particular slave it sends only POLL packets. The slave replies with data packets until its queue is empty. Then, it sends a NULL packet which signals the end of the exhaustive communication with that particular slave⁶.

In order to model the piconet as an exhaustive polling system, we define the service time of a k -slot data packet as $(k + 1)$ slots which are composed of the k slots of data, augmented by the *following* POLL packet. Hence, the service time of a 1-slot packet is defined as 2 slots, for 3-slot packet it is 4 slots, and for 5-slot packet it is 6 slots. The switchover time is defined as 2 slots, composed of the NULL packet ending the exchange with a particular slave and the POLL packet starting the exchange with the next slave.

For a *half-symmetrical* piconet ($\lambda_u = \lambda > 0$, $\lambda_d = 0$), we apply the model for a symmetrical discrete-time exhaustive polling system described in [18, p. 68]. Accordingly, we apply eq. (3.63b) in [18], where the number of queues is N , the arrival process is Poisson with intensity λ , the switchover time is two slots ($r = 2$) with zero variance ($\delta^2 = 0$), the mean

⁶The termination of the master-slave exchange with a POLL-NULL exchange results from the fact that the master has no information about the slaves' queues and complies with the assumptions made in previous analyses of the exhaustive algorithm (e.g. [14],[15],[16]).

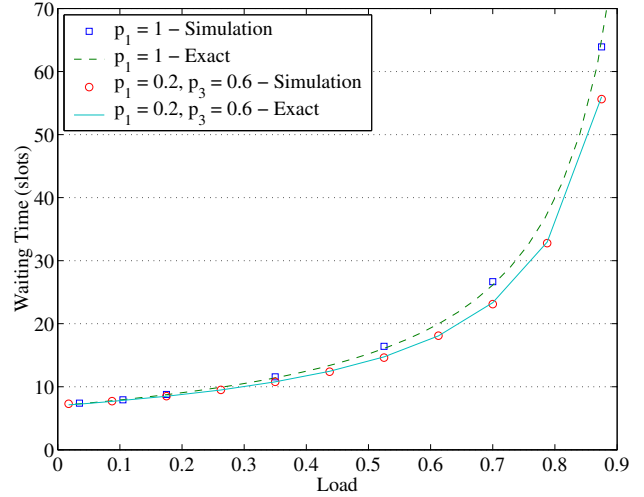


Fig. 2. The exact mean waiting time (calculated according to (1)) and the average waiting time values, computed by simulation, in piconets composed of 7 slaves and with only uplink traffic, operated according to the exhaustive algorithm.

service time is $b = \overline{L} + 1$, and the second moment of the service time is $b^{(2)} = 4p_1 + 16p_3 + 36p_5$. By adding 0.5 slot, we obtain the mean waiting time (in slots)⁷:

$$\overline{W}_u = \frac{N \left[1 + 4\lambda(p_3 + 3p_5) \right]}{1 - N\lambda(\overline{L} + 1)}. \quad (1)$$

We shall refer to $N\lambda(\overline{L} + 1)$ as the load in the uplink exhaustive system.

In a piconet with a single slave ($N = 1$) there is no difference between the exhaustive and the limited scheduling algorithms. As a special case, consider a piconet with unidirectional traffic of 1-slot packets (i.e. $p_1 = 1$) operated according to the limited algorithm. Its mean waiting time is given in eq. (2) in [22] as:

$$\overline{W}_u^i = \frac{N}{1 - 2N\lambda_u^i}. \quad (2)$$

It readily follows that for such a piconet ($\lambda_u^i = \lambda$, $N = 1$, and $p_1 = 1$) (1) and (2) coincide.

The result presented in (1) was also verified by a simulation model based on OPNET (the simulation model is described in [8]). For example, Fig. 2 compares the exact mean waiting time to the computed (by simulation) average waiting time, in piconets with 7 slaves in which (i) all packets are 1-slot long and (ii) $p_1 = 0.2$, $p_3 = 0.6$, and $p_5 = 0.2$. For each load value, the results have been computed after 230,000 slots.

We now consider an *asymmetrical* piconet with only *uplink traffic* (i.e. $\lambda_d^i = 0 \forall i$, and $\lambda_u^i > 0$ not all necessarily equal). Such a piconet can be analyzed in a similar manner

⁷We add 0.5 slot, since we are interested in the waiting time from the time of arrival, whereas in [18] the waiting time is counted from the end of the slot in which a packet arrives. We note that using eq. (3.69) in [1, p. 200] with the same parameters does not require adding 0.5 slot and yields the same result.

to a half-symmetrical piconet. Namely, it can be modeled as an asymmetrical exhaustive *polling* system composed of N queues, with 2-slot switchover time and with service time of $(k + 1)$ slots for a k -slot data packet. Accordingly, the mean waiting time in each uplink queue can be obtained by any of the methods for analyzing exhaustive polling systems described in [18] and [19]. Since some of these methods require solving $O(N^2)$ equations and since $N \leq 7$, the computational complexity is negligible. We note that results can be obtained even for the case in which the probabilities of a packet length being 1, 3, or 5 slots vary in different uplink queues.

Finally, we note that a piconet with batch arrivals can be analyzed by directly applying the methods for the analysis of exhaustive polling systems with a compound Poisson arrival process (see for example [18]).

B. Analysis of the Gated Algorithm

A piconet with only *uplink* traffic operated according to the *gated* algorithm is similar to a piconet operated according to the exhaustive algorithm. The main difference is that a slave replies to the master *only* with the data packets that were present in the uplink queue when it received the first POLL packet from the master. In order to signal the end of the gated communication, the slave sends a NULL packet. We note that since we assume that the master and the slave do not exchange queue status information, the last POLL-NULL exchange is required. Yet, by slightly modifying the protocol this exchange could be avoided.

This algorithm can be modeled as a gated *polling* system in a similar manner to the modeling of the exhaustive algorithm. Namely, we define the service time of a k -slot data packet as $(k + 1)$ slots and the switchover time is defined as 2 slots. For a *half-symmetrical* piconet, we apply the model for a symmetrical gated polling system described in [18, p. 104]. Accordingly, using eq. (5.23) in [18], we obtain the mean waiting time (in slots):

$$\bar{W}_u = \frac{N \left[1 + 4\lambda(p_3 + 3p_5) \right]}{1 - N\lambda(\bar{L} + 1)} + \frac{2N\lambda(\bar{L} + 1)}{1 - N\lambda(\bar{L} + 1)}. \quad (3)$$

An *asymmetrical* piconet with only *uplink traffic* can be similarly analyzed by one of the methods described in [18] and [19].

C. Analysis of the Limited Algorithm

In this section we show that a piconet with only *uplink traffic* operated according to the *limited* (pure round robin) scheduling algorithm can be modeled as a 1-limited polling system. In such a piconet the master continuously sends POLL packets to the slaves. Even if the slave has nothing to send, one slot must be used during the uplink communication (by the NULL packet).

We define the beginning of the switchover to a queue as the instance in which the *preceding* slave starts transmitting

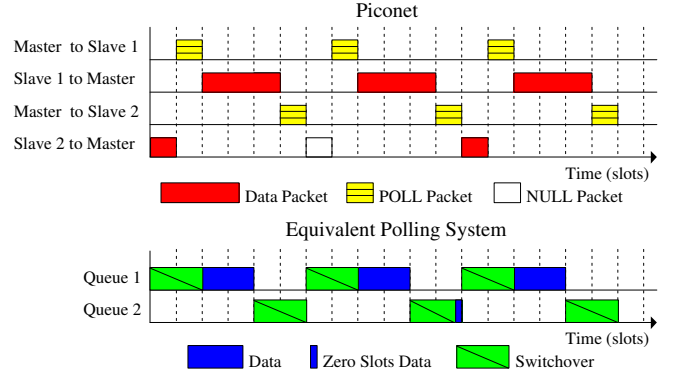


Fig. 3. An example of a piconet operated according to the limited algorithm and of the equivalent polling system.

the last slot of a data packet or a NULL packet. A switchover ends when the master completes the transmission of the POLL packet *intended to the slave* (if at the end of the switchover the queue is empty, the switchover to the next queue is immediately started). We define the switchover time to each of the queues as 2 slots:

- If the preceding slave sends a 3 or 5-slot data packet, the 2 switchover slots are composed of the last slot of the packet and the following POLL packet.
- If the preceding slave sends a 1-slot data packet or a NULL packet, these 2 slots are composed of the packet sent by the preceding slave and the following POLL packet.

Consequently, when data packets are sent, some of the data is actually sent during the “switchover” to the next queue. Therefore, the service time of a k -slot data packet is defined as $(k - 1)$ slots. Note that this implies that a 1-slot packet has a service time of 0 slots. Fig. 3 illustrates an example of the operation of a piconet and of the equivalent polling system.

We now focus on half-symmetrical systems in which the arrival rates to all uplink queues are equal (i.e. $\lambda_u = \lambda > 0$ and $\lambda_d = 0$) and apply the model for a symmetrical discrete-time 1-limited polling system described in [18, p. 140]. We use [18] eq. (6.60), where the switchover time is two slots ($r = 2$) with zero variance ($\delta^2 = 0$), the mean service time is $b = \bar{L} - 1$, and the second moment of the service time is $b^{(2)} = 4p_3 + 16p_5$. By adding 0.5 slot, we obtain the mean waiting time (in slots):

$$\bar{W}_u = \frac{N \left[1 + 4\lambda(p_3 + 3p_5) \right]}{1 - N\lambda(\bar{L} + 1)}. \quad (4)$$

As a special case, consider a half-symmetrical piconet with unidirectional traffic of 1-slot packets (i.e. $p_1 = 1$) operated according to the limited algorithm. Its mean waiting time has been derived in [22] and it is given by (2). It readily follows that for such a piconet ($\lambda_u^i = \lambda$ and $p_1 = 1$), (4) coincides with (2). Moreover, in a piconet with a single slave ($N = 1$), there is no difference between the exhaustive and the limited

scheduling algorithms. Indeed, for $N = 1$, the result presented in (4) coincides with (1), which presents the mean waiting time in a piconet operated according to the exhaustive algorithm. Finally, the result presented in (4) was also verified by a simulation model based on OPNET.

We note that an *asymmetrical* piconet with unidirectional uplink traffic can be modeled as a 1-limited polling system with N queues in a similar manner. Since there are no closed form results for the latter case, approximation methods reviewed in [19] can be used. Moreover, a piconet with batch arrivals can be analyzed by applying the methods for the analysis of 1-limited polling systems with a compound Poisson arrival process [19].

D. Equality of Mean Waiting Times

Eq. (1) and (4) lead to the following.

Corollary 1: The mean waiting time in a half-symmetrical piconet with only uplink traffic is the same for the exhaustive and for the limited scheduling algorithms.

It is well known [18],[19] that in the classical symmetrical polling systems, where switchover time is incurred *whenever* the server moves from one channel to the next, the mean waiting time in the exhaustive regime is *smaller* than its counterpart in the 1-limited regime. When the piconet is operated according to the exhaustive algorithm, switchover time is incurred at the end of a slave-master session. On the other hand, in the limited algorithm, when two adjacent slaves' queues are non-empty, no *real* switchover time is incurred. Real switchover times are paid only when a slave has nothing to transmit. Thus, this limited procedure is more efficient than the classical one. In the next section we extend Corollary 1 to general TDD systems and in section V we show that it is a specific case of a phenomenon occurring in symmetrical (non-Bluetooth) polling systems.

The mean waiting time in a piconet using the gated regime (3) is higher than the corresponding value in piconets using the exhaustive and limited algorithms. Again, this observation differs from the situation in classical polling systems. Usually, one can use the gated algorithm in order to provide some fairness to the different queues, while maintaining relatively low delay. It seems that in a symmetrical piconet with unidirectional traffic, the limited algorithm provides both the desirable fairness and the lowest delay.

E. Extension to General TDD Systems

In the following lemma we extend the result presented in Corollary 1 to general systems operated according to the TDD mechanism. First, we define a *general TDD system* as follows.

Definition 1: A *general TDD system* is composed of a master (base station) and at least one slave (station), operated in a similar manner to a Bluetooth piconet. Namely:

- The channel is slotted.
- A slave is allowed to start transmission in a slot, if the master has addressed it in the preceding slot.

- The master addresses a slave by sending a data packet or a 1-slot *empty packet*.
- The slave must respond by sending a data packet or a 1-slot *empty packet*.
- The *master and slaves are allowed to send data packets of any length*. The probability of a packet length being i slots long is denoted by p_i .

Lemma 1: In a symmetrical general TDD system with only uplink traffic, the two mean waiting times, in the exhaustive and limited scheduling algorithms, are equal to each other for any given packet length distribution.

Proof: According to Definition 1, the mean packet length is $\bar{L} = \sum_{i=1}^{\infty} p_i i$ and the second moment of the packet length distribution is $\bar{L}^2 = \sum_{i=1}^{\infty} p_i i^2$. Similarly to the analysis in Section IV-A, it can be shown that the considered TDD system operated according to the *exhaustive* scheduling algorithm is equivalent to the symmetrical exhaustive polling system. In the equivalent polling system, the service time of a k -slot packet (of the TDD system) is defined as $k + 1$ slots and the switchover time is defined as two slots with zero variance. Accordingly, using the notation of [18], the mean service time is $b = \bar{L} + 1$, the mean switchover time is $r = 2$, the variance of the switchover time is $\delta^2 = 0$, and the second moment of the service time is: $b^{(2)} = \sum_{i=1}^{\infty} (i + 1)^2 p_i = \bar{L}^2 + 2\bar{L} + 1$. Using [18], eq. (3.63b) and adding 0.5 slot, we obtain the mean waiting time (in slots):

$$\bar{W}_u = \frac{N(\lambda\bar{L}^2 - \lambda + 2)}{2(1 - N\lambda(\bar{L} + 1))}. \quad (5)$$

The considered TDD system operated according to the *limited* scheduling algorithm is equivalent to the symmetrical 1-limited polling system. In the equivalent polling system, the service time of a k -slot packet (of the TDD system) is defined as $(k - 1)$ slots and the switchover time is defined as two slots with zero variance. Accordingly, $b = \bar{L} - 1$, $r = 2$, $\delta^2 = 0$, and $b^{(2)} = \sum_{i=1}^{\infty} (i - 1)^2 p_i = \bar{L}^2 - 2\bar{L} + 1$. Using [18], eq. (6.60) and adding 0.5 slot, we again obtain (5). ■

It can be shown that in a symmetrical general TDD system, the waiting time in the gated algorithm is *higher* than the waiting times in the exhaustive and limited algorithms.

V. GENERALIZATION TO POLLING SYSTEMS

In this section we extend the result presented in Lemma 1 to continuous time (non-Bluetooth) polling systems. We show that the mean waiting time in a symmetrical 1-limited polling system with constant switchover times is equal to the mean waiting time in a corresponding exhaustive polling system with extended service time. Furthermore, in [4] and [7] it has been shown that the mean waiting times in exhaustive and gated polling systems decompose into a sum of two terms, one being a function of the switchover times and the other the mean waiting time in the corresponding model with *zero-switchover* times. We shall show that for symmetrical polling

systems operated according to the 1-limited regime, a similar decomposition result to the one obtained by Fuhrman [7] for the exhaustive regime holds as well.

We denote the mean waiting times in the symmetrical exhaustive and 1-limited (non-Bluetooth) polling systems with constant switchover times by \overline{W}^{Ex} and \overline{W}^L . We denote the server utilization at a queue by ρ_1 and the total server utilization by $\rho = N\rho_1$. The *corresponding extended service* polling system is defined as follows⁸.

Definition 2: A *corresponding extended-service* polling system differs from the basic polling system only by the fact that the service time of a packet whose original length is k is extended to $k + r$.

According to Definition 2, the mean and the second moment of the service time in the corresponding extended-service system are $x = b + r$ and $x^{(2)} = b^{(2)} + 2rb + r^2$, respectively. We denote the mean waiting time in the corresponding extended-service system by \overline{W}_{b+r} .

Observation 1: The mean waiting times in the symmetrical 1-limited polling system with constant switchover times and in the *corresponding extended-service* exhaustive polling system are equal. Namely:

$$\overline{W}^L = \overline{W}_{b+r}^{Ex}. \quad (6)$$

Proof: Applying [18, eq. (4.33b)] with service time $b + r$ and second moment of service time $b^{(2)} + 2rb + r^2$ to get \overline{W}_{b+r}^{Ex} yields the same result as \overline{W}^L given in [18, eq. (6.19)]. ■

According to Observation 1, the result about the equality of the mean waiting times in general TDD systems operated according to the exhaustive and the limited algorithms (i.e. Lemma 1) is a specific case of a result that holds in symmetrical polling systems.

We shall now define a *corresponding zero-switchover* system, differing from the basic polling system only by the fact that the switchover time is zero. We denote by \overline{W}^0 the mean waiting time in the corresponding zero-switchover system.

In [7, Prop. 4] it has been shown that in an exhaustive polling system with constant switchover times:

$$\overline{W}^{Ex} = \overline{W}^0 + \frac{Nr(1 - \rho_1)}{2(1 - \rho)}. \quad (7)$$

The result derived in [7] holds for asymmetrical systems and was extended by Cooper et al. [4] for the case in which the switchover times are random variables. Since in continuous-time symmetrical systems with zero-switchover periods the mean waiting time is the same disregarding the polling regime [10], the following corollary immediately follows from Observation 1 and (7).

Corollary 2: The mean waiting time in the 1-limited polling system decomposes into two terms: (i) the mean

waiting time in the corresponding extended-service zero-switchover 1-limited polling system and (ii) a function of the switchover and service times. Namely:

$$\overline{W}^L = \overline{W}_{b+r}^0 + \frac{Nr(1 - \rho_1)}{2(1 - \rho)}, \quad (8)$$

where ρ_1 and ρ are the server utilization values in the corresponding extended-service system.

This Corollary resembles the decomposition result derived in [7] for the exhaustive algorithm (i.e. (7)).

VI. DOWNLINK TRAFFIC

A. Analysis of the Exhaustive Algorithm

Consider a piconet with only *downlink* traffic operated according to the exhaustive algorithm. In such a piconet traffic flows only from the master to the slaves and the master has complete information on the status of its downlink queues. Thus, there is no reason to send a POLL packet in order to end a master-slave exchange. Yet, in case all queues are empty, the master should transmit POLL packets (and receive NULL packets) until a data packet arrives to one of its downlink queues.

We define the operation model of the piconet as follows. The master serves the downlink queues in a fixed cyclic order. When serving queue i , the master sends all data packets present in the queue and the slave replies with NULL packets. When the master empties queue i , it immediately switches, in a cyclic manner to the next non-empty downlink queue. In case all queues are empty, the master sends a POLL packet to one of the slaves which replies with a NULL packet. If after the NULL packet at least one of the queues becomes non-empty, the master randomly selects one of the N queues and proceeds from there in a cyclic manner until it finds a non-empty queue which is immediately served.

Fig. 4 illustrates an example of the operation of such a piconet. In this example, when the master empties the downlink queue of packets intended to slave 1, the queue of slave 2 is empty. When it empties the queue of slave 3, all the queues are empty, and therefore, it sends POLL packets until at least one packet arrives. In the scenario described in the figure, packets arrive to queues 1 and 4 during the transmission of the NULL packet by slave 2. The master randomly selects queue 4, serves it, and continues to queue 1 in a cyclic manner.

The piconet can be modeled as a *discrete time* polling system with zero-switchover periods [10]. In order to obtain results for a discrete time exhaustive polling system with zero-switchover periods, Levy and Kleinrock [10] define the AZSOP (*Almost Zero Switchover Period*) *polling system*. In that system it is assumed that the switchover period is nonzero with probability p ($p > 0$) and zero with probability $1 - p$. Then, the system is analyzed as an exhaustive polling system with mean switchover times defined as p and the variances of switchover time defined as $p(1 - p)$. It is shown that when $p \rightarrow 0$, the waiting time in the AZSOP system approaches the waiting time in the zero-switchover period system.

⁸Recall that r is the mean switchover time in a polling system.

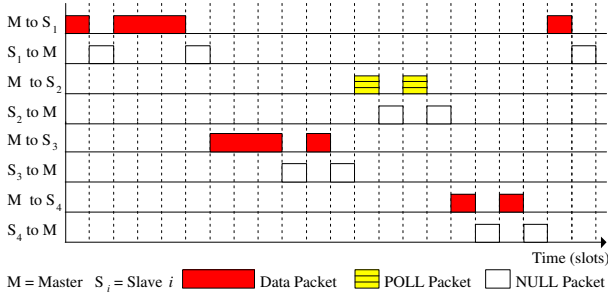


Fig. 4. An example of the operation of the exhaustive algorithm in a piconet with only downlink traffic.

We note that the *continuous time* polling system with zero-switchover periods [18, p. 142] *does not comply* with the operation model of a piconet, due to the following reason. In the continuous time model it is assumed that if a packet arrives while the server is idle, its service starts immediately. On the other hand, in a piconet, if a packet arrives while a POLL or a NULL packet is sent, it could be served only after the transmission of the NULL packet.

In order to model the piconet as an AZSOP polling system, we define a single slot in the AZSOP system as two slots in a Bluetooth piconet. To this end, we define the service time of a k -slot data packet in a Bluetooth piconet as $(k + 1)/2$ slots in the AZSOP system, which are composed of the k slots of data, augmented by the following NULL packet. Thus, in the corresponding AZSOP polling system, the service time of a 1-slot Bluetooth packet is defined as 1 slot, for 3-slot Bluetooth packet it is 2 slots, and for 5-slot Bluetooth packet it is 3 slots. The switchover time in the AZSOP system is defined as 1 slot, composed of POLL and NULL packets. As mentioned above, the length of this period is 1 slot with probability p .

We now focus on half-symmetrical systems (i.e. $\lambda_d = \lambda > 0$ and $\lambda_u = 0$). By applying the model for a discrete-time exhaustive polling system described in [18] and using the methodology described in [10], we can obtain the mean waiting time. Accordingly, we apply [18] eq. (3.63b), where the arrival process is Poisson with intensity 2λ the switchover time is $r = p$, the variance of the switchover time is $\delta^2 = p(1 - p)$, the mean service time is $b = (\bar{L} + 1)/2$, and the second moment of the service time is $b^{(2)} = p_1 + 4p_3 + 9p_5$. Letting $p \rightarrow 0$, adding 0.5 slot (since in [18] the waiting time is counted from the end of the slot), and multiplying by 2 (since the obtained result is the number of slots in the AZSOP system and we are interested in waiting time measured in Bluetooth slots), we obtain the mean waiting time (in Bluetooth slots):

$$\bar{W}_d = \frac{1 + 4N\lambda(p_3 + 3p_5)}{1 - N\lambda(\bar{L} + 1)}. \quad (9)$$

A similar approach can be used for the analysis of *asymmetrical* piconets with only *downlink traffic* (i.e. $\lambda_u^i = 0 \forall i$, and $\lambda_d^i > 0$, not all necessarily equal). That is, it can be modeled as an asymmetrical AZSOP polling system operated according to the exhaustive regime and composed of N queues, with

1-slot switchover time and with service time of $(k + 1)/2$ slots for a k -slot data packet. Accordingly, a relatively good approximation of the waiting times in each downlink queue can be computed by solving $O(N^3)$ equations as described in [10, Section 3.6].

B. Gated and Limited Algorithms

Half-symmetrical piconets operated according to the gated and limited algorithms can be modeled as AZSOP polling systems similarly to the modeling of exhaustive algorithm. By applying the models for discrete-time gated and 1-limited polling systems described in [18] and using the methodology described in [10], we obtain the mean waiting times for the two schemes. It turns out that all 3 mean waiting times, for the exhaustive, gated, and limited, are equal and given by (9).

C. Comparison

The fact that in a half-symmetrical piconet with only downlink traffic, the mean waiting time is the same for all algorithms is expected. Such a result was obtained in [10] for a symmetric discrete-time polling system with fixed service times and zero switchover times. Similarly, it is well known [10],[18] that the mean waiting time in symmetric continuous time polling systems with zero switchover time is equal to the mean waiting time in an M/G/1 system with the combined inputs of all queues, regardless of the polling regime (exhaustive, gated, or 1-limited). Yet, it is interesting to compare the results obtained for systems with only downlink traffic (i.e. (9)) to the results for systems with only uplink traffic (i.e. (1), (3), and (4)). For clarity of the presentation, we use in the following equations the superscript to denote the scheduling algorithm. It can be seen that

$$\bar{W}_d^L = \bar{W}_d^G = \bar{W}_d^{Ex} = \bar{W}_u^{Ex} - \frac{N - 1}{1 - N\lambda(\bar{L} + 1)}, \quad (10)$$

where it has been shown in Section IV that

$$\bar{W}_u^{Ex} = \bar{W}_u^L = \bar{W}_u^G - \frac{2N\lambda(\bar{L} + 1)}{1 - N\lambda(\bar{L} + 1)}. \quad (11)$$

Moreover, for the special case in which the traffic is composed of only 1-slot packets (i.e. $p_1 = 1$), there is a significant difference between the values in only downlink and only uplink piconets. Namely,

$$\bar{W}_u^L = \bar{W}_u^{Ex} = N\bar{W}_d^L = N\bar{W}_d^{Ex} = N\bar{W}_d^G. \quad (12)$$

The above results can be useful for developing piconet and scatternet topology construction algorithms (see [20] for a review of the scatternet topology construction problem). When the traffic is mostly unidirectional, allowing the node that generates most of the traffic to be the master would significantly decrease the delay.

VII. BI-DIRECTIONAL TRAFFIC

Analyzing the performance of scheduling regimes such as the exhaustive and gated in a piconet with bi-directional traffic requires obtaining the PGF of the exchange time of a single master-slave queue pair (channel). This analysis is significantly complicated by the TDD mechanism and the use of POLL and NULL packets by the master and the slaves. In order to demonstrate the difficulties in analyzing the exhaustive algorithm, we discuss a less complicated case, namely a single master-slave channel in a piconet, operated in the *gated algorithm*.

In the gated algorithm, only the packets that are found in the uplink and downlink queues when the master starts serving the master-slave queue pair are transmitted. If the number of downlink packets exceeds the number of uplink packets, the slave sends NULL packets as a response to some data packets. On the other hand, if the number of uplink packets exceeds the number of downlink packets, the master sends some POLL packets in order to allow the slave to reply with data packets. We assume that at the end of the master-slave exchange, the slave has to respond with a NULL packet to a POLL packet.

Let X_G denote the total time (number of slots) required for the exchange duration of a single master-slave channel in the gated algorithm. Namely, it is the number of slots required to serve all packets present in both downlink and uplink queues at the instance when the master starts serving the queue pair plus 2 slots (the last POLL-NULL exchange). The PGF and the mean of X_G are denoted by $X_G(x)$ and $\overline{X_G}$. For simplicity, we assume that *all packets are 1 slot long* ($p_1 = 1$) and that packets have accumulated in both queues for some T slots before the gated service starts. We define U and D as the number of packets accumulated in the uplink and downlink queues, respectively, during T slots ($U, D \sim \text{Poisson}(\lambda T)$).

Thus, given that $p_1 = 1$, X_G equals twice the maximum of U and D plus 2 slots. Namely, it is a function of the maximum of two Poisson random variables. Accordingly, the PGF of the time to serve a single master-slave channel is given by: $X_G(x) = x^2 \sum_{m=0}^{\infty} x^{2m} \text{Prob}(\max[U, D] = m)$, where

$$\text{Prob}(\max[U, D] = m) = 2 e^{-\lambda T} \frac{(\lambda T)^m}{m!} \left(\sum_{j=0}^{m-1} e^{-\lambda T} \frac{(\lambda T)^j}{j!} \right) + \left(e^{-\lambda T} \frac{(\lambda T)^m}{m!} \right)^2. \quad (13)$$

Unfortunately, it appears that in view of (13) there is no closed form expression for $X_G(x)$ and consequently, it seems that there is no closed form expression for the waiting time in a piconet with bi-directional traffic operated according to the gated algorithm. It is clear that the exact analysis of the exhaustive algorithm is more complicated.

The mean time to serve a single master-slave channel is given by: $\overline{X_G} = 2E(\max[U, D]) + 2$. In order to bound the value of $\overline{X_G}$, we observe that for $U, D \sim \text{Poisson}(\gamma)$ and $\gamma > 0$: $1 < E(\max[U, D])/\gamma < 2$. To illustrate the behavior

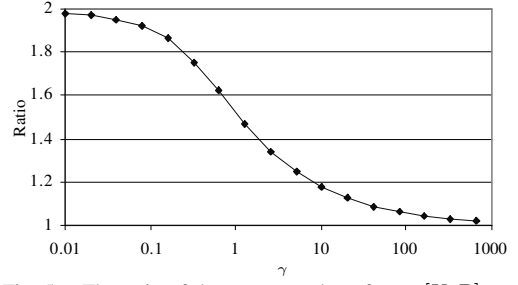


Fig. 5. The ratio of the average value of $\max[U, D]$ to γ .

of this ratio, we have randomly generated 300,000 different values of U and D (for 17 various values of γ) and computed the average value of $\max[U, D]$ and its ratio to γ . The results are depicted in Fig. 5.

We shall now provide a simple explanation for an observation made via simulation in [3] and [8]. According to [3] and [8], in piconets with bi-directional traffic and high loads, the limited algorithm outperforms the exhaustive algorithm. Consider a symmetrical piconet operated according to the exhaustive or gated algorithms, with only 1-slot packets. According to the above analysis, for an arrival rate of λ (packets/slot), a node will have to transmit on average $E(\max[U, D])$ packets per slot (where $U, D \sim \text{Poisson}(\lambda)$). Thus, the arrival rate λ should be set such that $2NE(\max[U, D]) < 1$. As we have shown in Fig. 5, $E(\max[U, D])$ can approach 2λ . Hence, a necessary condition for stability is $\lambda < 1/(\alpha N)$, where $2 < \alpha < 4$. On the other hand, when the same piconet is operated according to the limited algorithm, a necessary condition for stability is $\lambda < 1/(2N)$. When λ approaches the stability limit, the waiting time approaches infinity. Thus, in a piconet using the exhaustive or gated algorithm, the waiting time approaches infinity for lower values of λ than in a piconet using the limited algorithm. Therefore, for high values of load the waiting time in the limited piconet will be lower than in the exhaustive or gated piconet.

VIII. NUMERICAL RESULTS

Approximate results regarding the performance of various intra and inter-piconet scheduling algorithms have been presented in [11],[15],[16], and [17]. The analysis of the exhaustive algorithm in [15] is based on 2 stages: (i) the derivation of the PGF of the time to exhaust a single master-slave queue pair, and (ii) modeling the piconet as an M/G/1 queue with vacations. In [21] we have shown that the PGF of the time to exhaust a queue pair, derived in [15], does not take into account the complexities discussed in Section VII. Thus, it differs from the correct PGF. Moreover, we have argued that the direct application of results from the M/G/1 queue with vacations model to the piconet system ignores important statistical dependencies that exist in the piconet operation model. In this section we compare our exact numerical results to numerical results computed according to [15] and [16].

The model presented in Section III is a specific case of the

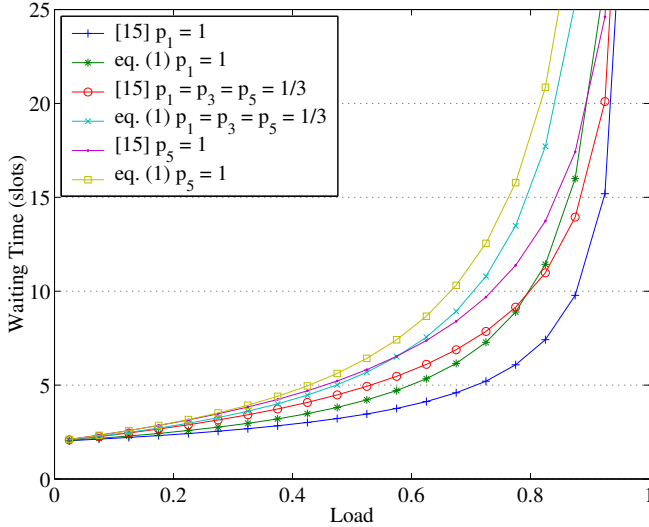


Fig. 6. The mean waiting time derived according to [15] and the exact mean waiting time (derived according to (1)) in half-symmetrical piconets with only uplink traffic, composed of 2 slaves, and operated according to the exhaustive algorithm.

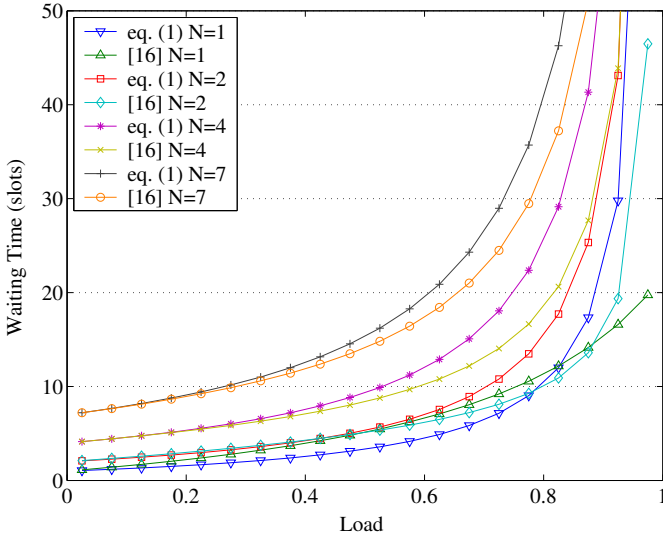


Fig. 7. The mean waiting time derived according to [16] and the exact mean waiting time (derived according to (1)) in half-symmetrical piconets with only uplink traffic, operated according to the exhaustive algorithm, in which $p_1 = p_3 = p_5 = 1/3$.

piconet model presented in [15]. Thus, Fig. 6, compares the mean waiting time computed according to the analysis of the exhaustive regime in [15] when $\lambda_d = 0$ to the mean waiting time computed according to our analysis (i.e. according to (1)). The figure presents the waiting time (in slots) as a function of the load in the uplink exhaustive system ($N\lambda(\bar{L} + 1)$) in half-symmetrical piconets with 2 slaves.

In [16] the intra-piconet exhaustive scheduling algorithm is analyzed in a somewhat different methodology than the analysis described in [15]. In Fig. 7 we compare the exact mean waiting time to the mean waiting time computed according to [16], in half-symmetrical piconets with only unlik traffic

($\lambda_u = \lambda, \lambda_d = 0$) in which the probabilities of 1, 3, and 5-slot packets are equal.

It is clear that in all cases shown the results presented in [15] and [16] underestimates or overestimate the mean waiting time. Thus, we conjecture that for complicated scenarios, deriving approximate results, which are based on the relationship between Bluetooth piconets and polling systems, will yield significantly better approximations than those that are based on M/G/1 queue with vacations.

IX. CONCLUSIONS

This work reveals overlooked connections between *Bluetooth piconets* and *polling systems* that enable to obtain exact and approximate analytical results regarding the performance of Bluetooth scheduling algorithms. First, we have analyzed piconets with unidirectional uplink traffic. We have obtained exact results for the symmetric limited, gated, and exhaustive regimes, and shown that exact results can also be obtained for asymmetrical piconets operated according to the gated and exhaustive algorithms.

We have shown that in symmetrical piconets with only uplink traffic, the mean waiting times are the same for the limited and exhaustive algorithms. This observation has been extended for general TDD systems and for specific (non-Bluetooth) polling systems. The extension to polling systems yields a delay decomposition result for symmetrical 1-limited polling systems.

Furthermore, we have shown that a piconet with unidirectional downlink traffic is equivalent to a polling system with zero-switchover times. The mean waiting times in such a piconet can be significantly lower than in piconets with only uplink traffic. The complications in analyzing the gated scheduling algorithm in piconets with bi-directional traffic have been described, indicating that the corresponding analysis of the exhaustive regime is even more complex. Finally, numerical results have been compared to approximate results derived in the past.

The presented analysis can be extended in various directions (e.g. batch arrivals, asymmetrical arrival processes, retransmissions, etc.) by directly applying various results regarding the performance of polling systems (see for example the extensions in [12] and [13] to our work regarding the limited algorithm [22]).

The exact results presented in this paper can be utilized in order to validate and evaluate simulation models and approximate analytic models. They also provide a few important insights regarding the design and the performance of Bluetooth piconets and scatternets. For example, since the mean waiting times are equal for the exhaustive and limited algorithms, it seems that when the traffic is mostly unidirectional, the limited algorithm, which provides some degree of fairness, is preferable. Moreover, topology construction algorithms can exploit the observation that when the traffic is mostly unidirectional, allowing the node that generates most of the traffic to be the master would significantly decrease the delay. Finally, the

effect of the packet length distribution on the waiting time has been revealed (the effect on the piconet throughput has been quite clear).

Due to the TDD mechanism, algorithms that tend to optimize the performance of polling systems are not necessarily optimal for piconets. Therefore, a future research direction is the development of optimal piconet scheduling algorithms. For example, we wish to analyze variations of the K -Limited scheduling algorithm in which the master exchanges up to K packets with each slave in every cycle. Furthermore, due to the inherent complexities in analyzing the gated and exhaustive algorithms, a future research direction is to obtain a good (at least approximate) analysis of such regimes.

ACKNOWLEDGMENTS

This research (No. 148/03) was supported by THE ISRAEL SCIENCE FOUNDATION, by a grant from the Ministry of Science, Israel, and by Intel. We thank the reviewers for helpful comments. We thank Daniele Miorandi and Hanoch Levy for helpful discussions and comments.

REFERENCES

- [1] D. P. Bertsekas and R. Gallager, *Data Networks*, Prentice-Hall Inc., 1992.
- [2] Bluetooth Special Interest Group, *Specification of the Bluetooth System – Version 1.2*, Nov. 2003.
- [3] A. Capone, M. Gerla, and R. Kapoor, “Efficient polling schemes for Bluetooth picocells”, *Proc. IEEE ICC’01*, June 2001.
- [4] R. B. Cooper, S.-C. Niu, and M. M. Srinivasan, “A decomposition theorem for polling models: the switchover times are effectively additive”, *Operations Research*, Vol. 44, pp. 629–633, July-Aug. 1996.
- [5] A. Das, A. Ghose, A. Razdan, H. Saran, and R. Shorey, “Enhancing performance of asynchronous data traffic over the Bluetooth wireless ad-hoc network”, *Proc. IEEE INFOCOM’01*, Apr. 2001.
- [6] R. Esmailzadeh, M. Nakagawa, and A. Jones, “TDD-CDMA for the 4th generation of wireless communications”, *IEEE Wireless Comm.*, Vol. 10, pp. 8–15, Aug. 2003.
- [7] S. W. Fuhrman, “A decomposition result for a class of polling models”, *Queueing Systems*, Vol. 11, pp. 109–120, 1992.
- [8] L. Har-Shai, R. Kofman, A. Segall, and G. Zussman, “Load adaptive inter-piconet scheduling in small-scale Bluetooth scatternets”, *IEEE Comm.*, Vol. 42, pp. 136–142, July 2004.
- [9] R. Kapoor, A. Zanella, and M. Gerla, “A fair and traffic dependent scheduling algorithm for Bluetooth scatternets”, *ACM/Kluwer MONET*, Vol. 9, pp. 9–20, Feb. 2004.
- [10] H. Levy and L. Kleinrock, “Polling systems with zero switch-over periods: a general method for analyzing the expected delay”, *Performance Evaluation*, Vol. 13, pp. 97–107, Oct. 1991.
- [11] D. Miorandi, C. Caimi, and A. Zanella, “Performance characterization of a Bluetooth piconet with multi-slot packets”, *Proc. WiOpt’03*, Mar. 2003.
- [12] D. Miorandi and A. Zanella, “Achievable rate regions for Bluetooth piconets in fading channels”, *Proc. IEEE VTC’04 Spring*, May 2004.
- [13] D. Miorandi and A. Zanella, “Performance analysis of limited-1 polling in a Bluetooth piconet”, *Proc. IEEE PIMRC’04*, Sep. 2004.
- [14] D. Miorandi, A. Zanella, and G. L. Pierobon, “Performance evaluation of Bluetooth polling schemes: an analytical approach”, *ACM/Kluwer MONET*, Vol. 9, pp. 63–72, Feb. 2004.
- [15] J. Mistic and V. B. Mistic, “Modeling Bluetooth piconet performance”, *IEEE Comm. Letters*, Vol. 7, pp. 18–20, Jan. 2003.
- [16] J. Mistic and V.B. Mistic, “Bridges of Bluetooth county: topologies, scheduling, and performance”, *IEEE J. on Selected Areas in Comm.*, Vol. 21, pp. 240–258, Feb. 2003.
- [17] V. B. Mistic and J. Mistic, “Performance of Bluetooth slave/slave bridge”, *Telecommunications Systems*, Vol. 22, pp. 221–239, Jan.-Apr. 2003.
- [18] H. Takagi, *Analysis of Polling Systems*, MIT Press, 1986.
- [19] H. Takagi, “Queueing analysis of polling models: progress in 1990–1994”, In *Frontiers in Queueing* (ed.: J. H. Dshalalow), CRC Press, pp. 119–146, 1997.
- [20] R. M. Whitaker, L. Hodge, and I. Chlamtac, “Bluetooth scatternet formation: a survey”, To appear in *Ad Hoc Networks*, Vol. 2, 2004.
- [21] G. Zussman, A. Segall, and U. Yechiali, “Bluetooth time division duplex – exact analysis as a polling system”, *CCIT Report No. 414*, Dept. of Electrical Engineering, Technion, Available at http://www.comnet.technion.ac.il/~gilz/pub_files/ccit_414.pdf, Feb. 2003.
- [22] G. Zussman, U. Yechiali, and A. Segall, “Exact probabilistic analysis of the limited scheduling algorithm for symmetrical Bluetooth piconets”, *Proc. IFIP-TC6 PWC’03, LNCS Vol. 2775* (eds: M. Conti et al.), Springer, Sep. 2003.