An energy-conserving wireless protocol for mobile environments

Imrich Chlamtac[a], Chiara Petrioli[b], Jason Redi[c]

[a] The University of Texas at Dallas
Erik Johnsson School of Engineering and Computer Science
E-mail: chlamtac@utdallas.edu

[b] Università di Roma "La Sapienza"
Dipartimento di Scienze dell'Informazione
E-mail: petrioli@dsi.uniroma1.it

[c] Boston University
Department of Electrical and Computer Engineering
E-mail: redi@bu.edu

Abstract

One of the most important factors which will influence the future success of ubiquitous mobile computing is the convenience and usefulness of the mobile devices. Major defining factors of the utility of these nodes will be the lifetime, size and weight of the power source. Various methods have been proposed for reducing battery consumption. In this paper, we consider a method to substantially reduce the amount of power necessary for reception of packets at the mobile devices. This energy-conserving protocol consists of a set of dynamic wakeup schedules at the nodes which are correlated with the scheduling of packets to be transmitted by a base station. We compare this protocol with a generalized version of a "directory" scheme similar to those presented in related research on energy-conserving protocols. It is shown that the proposed *On-line Grouped TDMA* protocol shows an improvement of 25-80% over the traditional schemes for uniform traffic distributions, and up to a 30% improvement in performance for heterogeneous traffic distributions when either low-energy or low-delay is of exclusive concern. The protocol maintains a high level of simplicity and flexibility which allows it to be used in a wide range of mobile computing applications.

1 Introduction

Wireless networking is rapidly gaining in popularity and breadth of applications. Many predict ubiquitous wireless networks to soon become "standard equipment" throughout the home, office and public areas. Two-way pagers, mobile computers, palm-top computers, and RFID are only a few examples of the wide variety of wireless computing devices which are expected to become commonplace. However, as with any consumer item, the success of wireless networking depends upon the cost and convenience of the end user system. Of paramount importance to both these areas is the design of devices that are energy-efficient. Of all the different systems and applications being developed, each characterized by different uses, processing and communication capabilities, bandwidth, cost and node location density, energy-conservation is a major factor which directs the design, usability, and feasibility of the entire wireless system.

A wealth of research has been performed into the areas of VLSI miniaturization, low power/sleep modes for processors, spin down algorithms and energy-efficient mechanisms for hard disks, as well as novel formulations for increased energy storage in batteries. However, as the wireless network transceiver can typically use 15-30% of the power of a typical mobile computer [2] (with the display the only part using more power), it is of great importance to include this portion of the mobile computing device when considering methods of energy-conservation.

The battery's energy is indeed a limited and scarce resource, which is not expected to increase in potential more than 30% in the near future [5]. Energy efficient solutions have many advantages:

Size and weight: Smaller, lighter batteries facilitate more innovative and convenient designs while not sacrificing "talk-time".

Maintenance: A more efficient use of batteries reduces the rate at which they need to be either replaced or recharged.

Environmental: Every battery that is disposed of is an environmental hazard and contains significant amounts of toxic compounds.

Recently, research has been performed into various methods for conserving energy by switching off the transceiver during the times it is expected not to be in use [1, 4]. These methods are all concerned with reducing the amount of energy necessary for reception. Typically uplink transmission can use twice as much energy as reception [2], however uplink transmission is usually dictated by the specific usage of the mobile user. As such, it is not simple to predict or schedule the use of the uplink of multiple mobile nodes. Additionally, since the uplink traffic is completely known at the mobile node, it can simply power down the transmitter when not in use. This is generally not the case for reception, where mobile users are passive entities which do not know if and when they will be destinations of a broadcast message, resulting in the need to keep the receiver in the ON state most of the time, and yielding therefore a higher energy consumption.

The protocols proposed in [4] address the problem of saving energy in systems where large amounts of data, such as stock exchange quotes, are periodically broadcast on a radio channel to a very large number of mobile users. Since in these applications mobile users autonomously decide when to retrieve data and the base station has no knowledge of when or what data the mobile users are retrieving, these solutions concentrate on the efficient organization of both the data and index information in the frame, in order to minimize both the energy and time required to access the data of interest. In contrast, this paper concentrates on the downlink communication of the base station sending packets to individual mobile nodes where each packet is expected to be received by an awake destination node when it is sent. The paradigm shifts from a base station "displaying" its information for anyone listening to the channel, to one where specific nodes are targeted for each transmission, such that approaches different from [4] have to be used.

In [3] Imielinski and al. considered this difference and proposed an energy-conserving protocol, the *preview/edition* protocol for e-mail and newsgroup broadcasting. In this protocol time is divided into cycles of preview and edition periods. During each preview, the identities of the destinations of the packets scheduled for the next edition period are announced, and the edition periods host the actual transmissions of the packets. This is very similar to the method for energy conservation used in the IEEE 802.11 draft standard for wireless LANs. In that system mobile users listen to a periodically broadcast TIM (Traffic Indication Map). This is essentially a directory of the packets to be transmitted, such that if they discover their ID broadcast in the TIM, they will stay awake during the following transmission period, but only until the last packet addressed to them is received. Otherwise, they will go into a sleep mode until next TIM transmission.

When considering these protocols for the sake of comparison to our protocol, it is important to note that neither of explicitly assume either slotted time or fixed size packets. Therefore, we introduce a new protocol called a *generalized directory protocol*, which summarizes many of the ideas presented in [1, 3] and represents an intuitive version of these protocols as they might be adapted to our system model.

In this paper, we present a new energy-conserving protocol, termed *On-line Grouped TDMA*, which dynamically assigns to groups of users the right to receive packets in a slot (i.e. awake slots), based on the current characteristics of the traffic. It will be shown that the proposed protocol performs fairly well, in many cases achieving a substantial improvement over protocols previously presented in the literature.

In the remainder of the paper we first describe the network model, followed by a description of both the On-line Grouped TDMA and directory protocols. We follow with a discussion of the simulation results, where the behavior of the two protocols under both uniform and heterogeneous destination distributions is compared. A summary concludes the paper.

2 Network model and protocols description

We consider a single cell system where N mobile users receive traffic from the base station.

Although both downlink and uplink communication is usually possible, as mentioned in the previous section, we restrict our attention to the downlink channel to address the problem of saving energy at the receiver when each transmission is a unique packet for a unique destination, and transmissions are base station initiated.

The downlink communication is broadcast by the base station through a radio channel of bandwidth B. The communication is packet-oriented. We assume the time to be slotted and the base stations transmissions to be synchronized to the beginning of slots. The packet length c is constant, and exactly one packet can be transmitted during one slot. In this analytic model, we do not explicitly treat transmission errors.

We define an *energy conserving* protocol as consisting of two components: a *transmission scheduling strategy* at the base station, where in each slot the base station selects a packet for transmission from the arrival queue, and a *wake-up schedule* associated with each mobile user, which determines which slots the mobile user's receiver will be in the "awake" mode and available to receive the transmission.

We introduce the following performance measures for comparing energy conserving protocols:

access delay: the average waiting time in slots experienced by a packet in the system between arrival at the base station and successful reception at the destination.

energy consumption: the average percentage of slots in which a mobile user is awake.

Our goal is the design of energy conserving protocols which minimize the energy consumption, while meeting the application delay constraints. In the protocol we propose, both the transmission scheduling strategy and the wake-up schedules are computed at the base station since it is there that all the current traffic is known. In choosing the order in which packets should be scheduled several parameters can be taken into account: the number of packets in the queue, the packets' ages, priorities, etc. In the protocols presented the 'oldest packet' criterion is generally adopted to help meet the application delay constraints. We next present and compare the following two classes of protocols for constructing efficient wake-up schedules:

- On-line grouped TDMA protocols
- Directory protocols

The latter class of protocols summarizes the main features of the energy-saving protocols at the receiver currently used in mobile environments and is introduced for sake of comparison.

2.1 On-line grouped TDMA protocols

In the protocols proposed, mobile users are divided into m groups and downlink transmission is organized in frames of length m, such that each group is initially assigned to a different single slot in the frame, similar to classical TDMA. Packets addressed to a particular destination will be transmitted in the slot to which that group is assigned. Each destination is associated with one and only one group. Among different packets to transmit in a slot, the base station always chooses the oldest, in order to minimize the worst case access delay.

We observe that, by choosing an optimum value for m, such a protocol would show a good energy-consumption, but the access delay would strongly degrade under even a moderately loaded heterogeneous traffic distribution, due to the fixed structure of the schedule. In particular, a slot is 'wasted' when no packets are transmitted because there are no packets for that particular group, but there are other groups associated with other slots which are heavily loaded. In order to improve the performance we, therefore, introduce a method that allows the borrowing of slots between different groups, with the aim of limiting the number of wasted slots. At the beginning of each slot, the base station decides whether the group associated with this slot should give up or "lend" some of it's future slots so other groups can wake up more frequently. It is required that there is a guarantee that there will be an available packet for transmission in the borrowed slot.

We observe that the information needed for a dynamic handling of the TDMA schedule (number and destinations of packets in the queue, wake-up schedules of all the groups, etc.) is known at the base station. The protocol, therefore, consists in an on-line organization of the wake-up schedules at the base-station. The mobile users only need to be informed, during their wake-up slots, of the length of the next sleep interval. We allow this quantity to be bounded by r frames, where r is an application-dependent parameter. The choice of a small value for r results in a shorter variance of the access delay and a quicker response to destination distribution changes, but increases the average access delay for a certain energy consumption. We note that the energy conservation protocol we propose is computationally simple, and would add minimal complexity to the base station. Additionally, utilizing the centralized approach of the base station for making the sleep-time decisions makes this protocol suitable for a wide range of wireless applications, including those in which mobile devices have severely limited processing capabilities.

In the following, we will first describe the *On-line Grouped TDMA* protocol for the case $r = 1$ and then extend it to the case of a general r. We introduce the following notation:

$(i, h)-$ the i-th slot in the h-th frame.

$P^1_{i,h}-$ the set of packets in the queue that can be scheduled for transmission in $(i, h + 1)$, as estimated one frame before. This quantity is used in evaluating whether group i should lend its slot during future frames.

$$\begin{aligned} P^1_{i,h} &= \text{packets in the queue} - \text{packets scheduled for transmission in (i,h),(i,h+1)} \\ &-\text{packets for asleep groups which will wake up after } (i,h+1) \end{aligned}$$

$P^2_{i,h}-$ set of candidate packets for transmission in $(i,h+1)$ (if borrowed).

$$P^2_{i,h} = P^1_{i,h} - \{p_g | p_g \text{oldest packet for } g \in P^1_{i,h}, \forall g = i, ..., m\}$$

$V_{i,h}-$ set of groups in $i, ..., m$ which are awake in frame h and do not have packets to be transmitted in frame $h+1$.

2.1.1 On-line grouped TDMA with r=1

We logically divide slots in a frame into two groups depending on whether they are "borrowed" or not. In the former case, the group originally associated with a slot is sent to sleep for one frame and a group with more traffic is assigned to wakeup in that slot instead. The only scheduling information which has to be transmitted to the group in the borrowed slot is the length of the next sleeping interval, while during slots which have not been borrowed, the base station additionally has to decide whether the group associated with that slot should be awake in the next frame or not. If there is at least one packet addressed to i in $P^1_{i,h}$, then group i will be awake in $(i,h+1)$. Otherwise the probability that a packet for i will be received before $(i,h+1)$ is computed and used to make the decision. We call a group i 'underloaded' if it is awake during the current frame h but it has not yet received a packet to transmit in (i,h) yet. We call $p^v_{i,h}$ the probability that a group $v \in V_{i,h}$ will receive a packet for transmission during frame $h+1$. This probability can be easily computed as follows:

$$\begin{cases} p^v_{i,h} &= A^2_{(i,h),(v,h)} + (1 - A^2_{(i,h),(v,h)}) A^1_{(v,h),(v,h+1)} \quad v\ underloaded \\ p^v_{i,h} &= A^1_{(i,h),(v,h+1)} \quad otherwise \end{cases}$$

, where $A^k_{(j,h),(j',h')}$ is the probability of at least k arrivals for v in the interval $(j,h)-(j',h')$. If we assume that the inter-arrival process is Poisson of parameter λ_v then:

$$\begin{cases} p^v_{i,h} &= 1 - e^{-\lambda_v(v-i)} - \lambda_v(v-i)e^{-\lambda_v(v-i)} + \\ & (e^{-\lambda_v(v-i)} + \lambda_v(v-i)e^{-\lambda_v(v-i)})(1 - e^{-\lambda_v m}) \quad v\ underloaded \\ p^v_{i,h} &= 1 - e^{-\lambda_v(m-i+v)} \quad otherwise \end{cases}$$

The quantity λ_v is estimated based on the number of packets received by the group in the recent past. A group i in $V_{i,h}$ will be awake during next frame $h+1$ if there are at least $|P^2_{i,h}|$ other groups in $V_{i,h}$ with a lower probability of getting a packet for the next frame. Finally, we observe that if a group i was asleep during frame h, then it will wake up in $(i,h+1)$, since there is no way to inform it of staying asleep any longer. The algorithm for computing the transmission schedule is as follows:

Step 0:

In frame 0 all the groups behave according to a classical TDMA method and decide whether to go to sleep or to be awake in frame 1.

Step h**:**

During the i-th slot in frame h we make a decision about group i in frame $h+1$.

- group i was asleep during frame h. It will be awake during frame $h+1$.
- group i was awake during frame h.
  - at least one packet in $P^1_{i,h}$ has i as its destination. Group i will be awake during frame $h+1$
  - no packet in $P^1_{i,h}$ has i as its destination.
    - if there are at least $|P^2_{i,h}|$ groups $v_{j_1}, ..., v_{j_{|P^2_{i,h}|}}$ in $V_{i,h}$ whose probability $p^{v_{j_k}}_{i,h}$ is lower than $p^i_{i,h}$, group i will be awake
    - otherwise, group i will go to sleep for one frame and slot $(i, h+1)$ will be associated with the destination of the oldest packet in $P^2_{i,h}$

Since all the decisions on whether a group should go to sleep or not are made during the slot associated with that group, it is easy to see that the base station is always aware of the identity of the next wake-up slot. Observe that, under a uniform association of mobile users to the groups, the borrowing mechanism introduced does not affect the average energy consumption (since exactly one group is awake during each slot). Also the fact that a slot is lent only if we are sure that a packet will be sent during that slot suggests that the average delay shouldn't increase because of the dynamic handling of the schedule, independently of the destination distribution.

2.1.2 On-line grouped TDMA: generic r

The improvement we can achieve by using the on-line grouped TDMA described in the previous section is partially limited by the fact that groups cannot go to sleep for more than one frame. We now extend this bound to a generic integer r. We mark as 'borrowed' the slots whose group has been forced to go to sleep and indicate with $B[h]$ the number of slots borrowed during frame h. We observe that the maximum lookahead is r frames, since this is the maximum number of frames which may have been affected by a decision made in the past. As before, no slot can be borrowed if there is no guaranteed transmission for it. Packets that can be transmitted in a borrowed slot can be divided into two groups, depending on whether their destination was awake at the time the slot was borrowed or not. This reflects the fact that even this second group of packets should be considered in the estimation of the sleep interval for group i if the next wake-up slot for i will occur after their destinations wake up. To do this, we first schedule the packets associated with currently asleep destinations in the future frames and compute the number $S[h]$ of packets which would be transmitted in frame h and have not been associated with a borrowed slot yet. We also denote with P the number of packets in $P^1_{i,h}$ which have not been associated with future borrowed slots. When in slot (i,h) the base station decides whether group i should go to sleep during any future frames and, if so, the length of the sleeping interval. If there is at least one packet for i in $P^1_{i,h}$, group i will be awake in $(i, h+1)$. Otherwise, the base station computes a lower bound on the needed sleep time, calculated as the maximum number of frames which could be completely filled by the transmission of packets in $P^1_{i,h}$(plus some of the packets addressed to currently asleep destinations if they will be available), under the assumption of no more packet arrivals. The length of the asleep interval is then computed (in frames) as the minimum between this quantity and r. As before we associate to a borrowed slot $(i, h+1)$ the destination of the oldest packet in $P^2_{i,h}$, which is not the first packet in the queue for groups $i, ..., m$. For a certain fixed value j, we require that:

$$\begin{aligned} \text{number of schedulable packets} \quad & \geq jm - \text{borrowed slots in } h, ..., h+j \\ & -\text{packets to asleep groups schedulable in } h, ..., h+j \\ P \quad & \geq jm - \sum_{q=1}^{j} B[h+q] - \sum_{q=1}^{j} S[h+q] \end{aligned}$$

, where we include in B even the first awake slot after a sleep period, and do not include in S the first packet which arrived for a certain group. It is easy to see that, if the above inequality is not satisfied for a certain j, then it will not be satisfied for any $j' > j$. The maximum required sleep time can, therefore, be computed by doing a binary search on the interval $1, ..., r$ and choosing the highest value satisfying the inequality, zero if no such value exists. The values of P and $S[h]$ have to be updated when either new packets arrive or a group is sent to sleep. In the latter case, the introduction of a new borrowed slot in frame h results in a decrease of $S[h]$, unless this value is already zero or all the scheduled packets are after (i, h), in which case P is decreased. A detailed description of the protocol follows:

Step h:

during the i-th slot in frame h we decide about frames in the interval $h+1, ..., h+r$.

- group i was asleep during frame h. It will be awake during frame $h+1$ only if its sleeping interval just expired.
- group i was awake during frame h.
  - at least one packet in $P^1_{i,h}$ has i as destination. Group i will be awake during frame $h+1$.
  - no packet in $P^1_{i,h}$ has i as destination.
    Request sleep time (1,r)
    Go to sleep for *sleep time* frames.
    If *sleep time* is zero then use $P^2_{i,h}$, the arrival rates and $V_{i,h}$ to determine whether to go to sleep, as for the case $r = 1$.

```
Requested sleep time $(i, j)$
Begin
if $i = j$ then
    if $(\sum_{h=1}^{i} B[h] + \sum_{h=1}^{i} S[h] - im + P > 0)$ then
        sleep time:=i
    else
        sleep time:=0;
else if $(j = i + 1)$ then
    if $(\sum_{h=1}^{i+1} B[h] + \sum_{h=1}^{i+1} S[h] - (i+1)m + P > 0)$ then
        sleep time:=i+1
    else if $(\sum_{h=1}^{i} B[h] + \sum_{h=1}^{i} S[h] - im + P > 0)$ then
        sleep time:=i
    else
        sleep time:=0;
else
    if $(\sum_{h=1}^{i+\lfloor\frac{j-i}{2}\rfloor} B[h] + \sum_{h=1}^{i+\lfloor\frac{j-i}{2}\rfloor} S[h] \geq (i + \lfloor\frac{j-i}{2}\rfloor)m - P)$ then
        Requested sleep time $(i + \lfloor\frac{j-i}{2}\rfloor, j)$
```

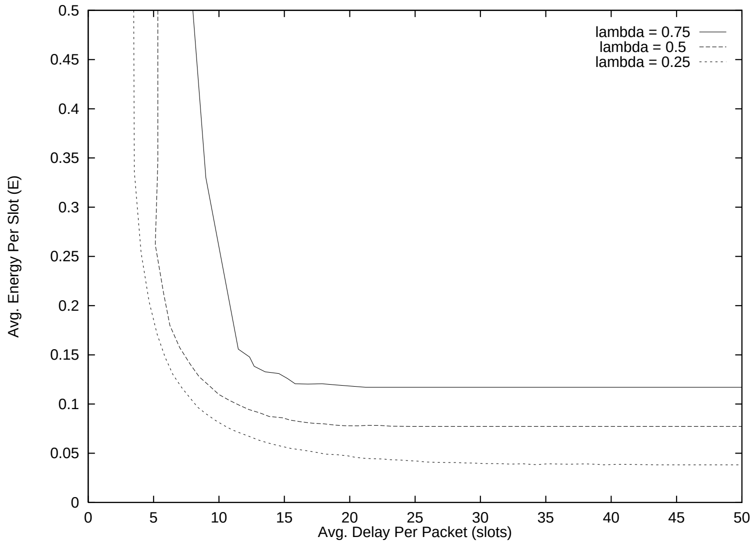

Figure 1: Generalized Directory protocol, uniform destination distribution, optimal directory and frame lengths, varying lambda

```
  else
     Requested sleep time $(i, i + \lfloor \frac{j-i}{2} \rfloor - 1)$
End.
```

2.2 Directory protocols

In this protocol time is divided into fixed length transmission cycles. Each cycle consists of two parts: a directory transmission, when the IDs of the mobile users which will receive a packet within the cycle are announced, and the actual transmission of the packets. The length of the directory transmission (and, therefore, the number of different destinations that can be listed) is fixed. We denote with d the maximum number of destinations listed in a directory, while up to k packets can be sent during a single cycle. All the mobile users are awake during the directory transmission. If their ID is broadcast, they wake up in the slots where the packets addressed to them are scheduled. Otherwise, they go to sleep until the beginning of the next cycle. The number and identity of the packets transmitted in a cycle are decided by the base station on the basis of the packets in the queue. In particular, at the beginning of a cycle the queue is examined in a FCFS order, and the oldest k packets addressed to the first d different destinations are scheduled for transmission. This corresponds to a FCFS strategy, such that certain packets are skipped if their destinations couldn't be informed of their future transmission.

3 Simulation results

Figures 1 through 5 describe the simulated performance of the generalized directory as well as the On-line Grouped TDMA protocol for both uniform and heterogeneous destination distributions. Each simulation consisted of a single base station generating 15,000 packets according to a Poisson process with arrival rates of $\lambda = 0.25, 0.5$ and 0.75 arrivals per slot. The heterogeneous distribution was simulated by a Gaussian distribution with a mean of $\frac{N}{2}$ and a variance of N. These packets are

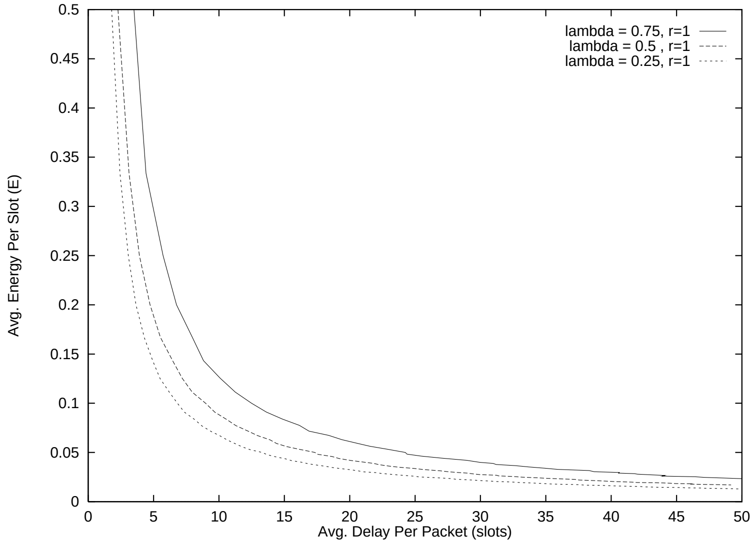

Figure 2: On-line Grouped TDMA protocol with r=1, uniform destination distribution, varying group sizes, varying lambda

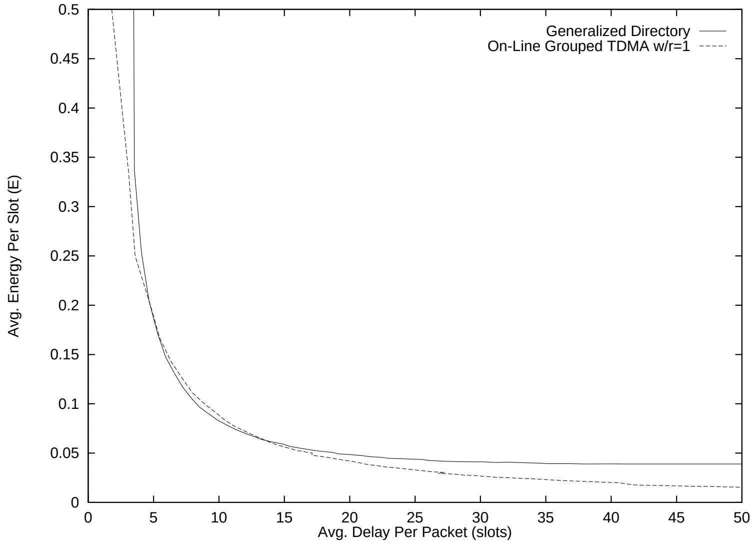

Figure 3: On-line Grouped TDMA and Generalized Directory protocol performance under a Gaussian destination distribution ($\mu = \frac{N}{2}, \sigma^2 = N$) and traffic arrival rate lambda = 0.25

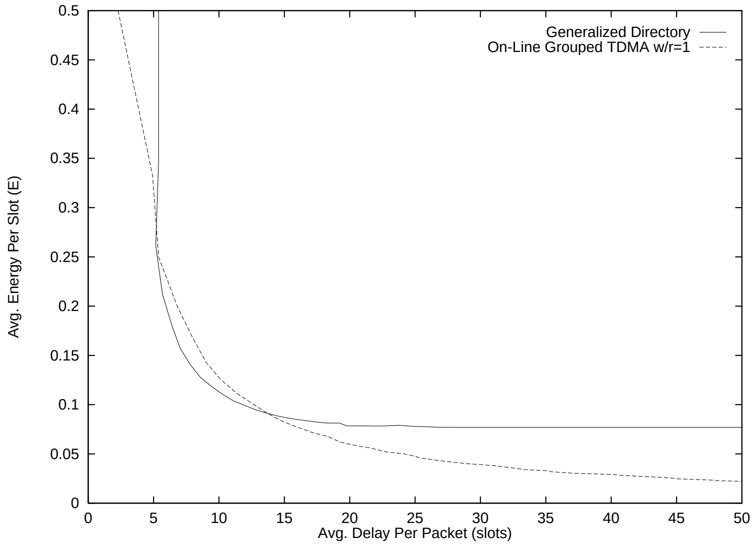

Figure 4: On-line Grouped TDMA and Generalized Directory protocol performance under a Gaussian destination distribution ($\mu = \frac{N}{2}, \sigma^2 = N$) and traffic arrival rate lambda = 0.5

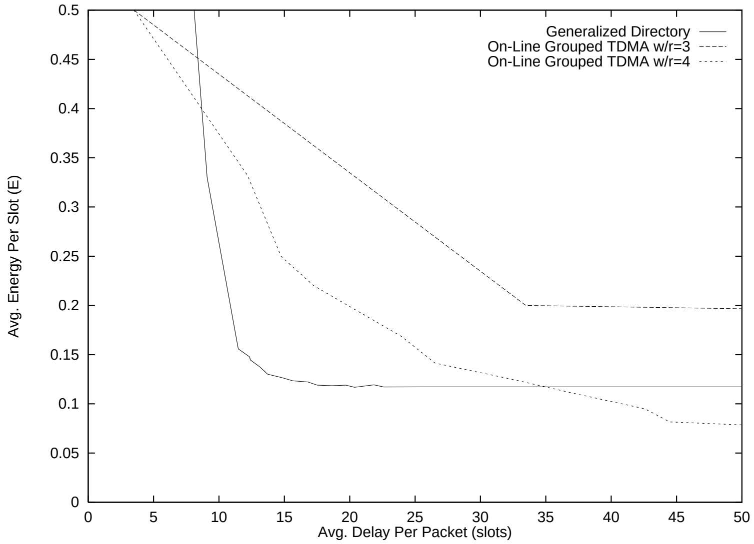

Figure 5: On-line Grouped TDMA and Generalized Directory protocol performance under a Gaussian destination distribution ($\mu = \frac{N}{2}, \sigma^2 = N$) and traffic arrival rate lambda = 0.75

sent to an arrival queue where they are selected for transmission to one of the 100 nodes according to the appropriate protocol.

Figures 1 and 2 plot the performance of the protocols for three arrival rates from a uniform destination distribution. From these two plots it is easy to see that not only does the On-line Grouped TDMA protocol perform better for all arrival rates than the generalized directory protocol, but there is a much smaller variation between the performance of various arrival rates. Specifically, we can see performance improvements from 25% in the mid-energy, mid-delay regions of the $\lambda = 0.25$ plots, up to an 80% improvement in the low-energy regions of the $\lambda = 0.75$ plots. As in all the plots of the directory protocol's performance, the line is generally characterized by a dramatic vertical line, a quick turn and then a nearly-flat horizontal line. The reason for these characteristics is the strict and regular wake-up pattern of the protocol. Since the directory can only send packets to nodes which had packets queued at the time the most recent directory was broadcast, the average delay consists first of the time between the moment a packet arrives until the time the next directory is broadcast, and second of the time between a directory broadcast and the slot when its destination receives the packet. In comparison, On-line Grouped TDMA's minimum average delay is only the average amount of time between arrival of a packet and the next wake-up slot of it's group ($\frac{m}{2}$ for $r = 0$). It is important to note that this minimum delay is not related to the arrival rate, only the number of groups. The directory's minimum average delay increases as the arrival rate increases since each slot used for the directory itself cannot transmit any packets. For example, for an arrival rate of 0.75, we need at least three transmission slots for every directory slot to maintain stability, while for $\lambda = 0.5$ we only need one transmission slot per directory slot. For Figure 2 it should be noted that the lines all correspond to the On-line Grouped TDMA with $r = 1$. Utilizing the generalized r version of the protocol yields nearly the exact same results with the uniform destination distribution as $r = 1$ since it is rare for any one group to have a large backlog of packets.

Figures 3 and 4 show the performance of the protocols for a Gaussian destination distribution with $\lambda = 0.25$ and 0.5 respectively. It can be seen that the directory protocol is generally unaffected by heterogeneous destination distributions. This behavior occurs since the protocol announces the packets' destinations individually in the directory slot, and therefore the wakeup strategies can potentially change from frame to frame based on the traffic. This ability is limited by d, the number of unique packet destinations to be transmitted in the frame. In contrast, the On-line Grouped TDMA performance decreases in the mid-energy, mid-delay region. The decrease in comparable performance begins where the arrival rate of packets to a particular group is greater than the average amount of time that the group is awake. This is the region where the lending of slots begins show an advantage over a standard grouped TDMA since the base station can begin to have groups change their wakeup schedule based on the traffic distribution. However, this "online" adaptation ability does not bring the performance below that of the directory until the point where there are enough groups in the system such that the base station can more efficiently "borrow" the slots from groups with very little traffic. Since we are assuming groups consist of an equal number of destination nodes, the point where this performance increase occurs could be improved through either a greater number of nodes or by performing an on-line reassignment of the number of nodes in each group. We do not consider the latter solution here since that would most likely reduce the ability of the protocol to adapt to very quick changes in the destination distribution.

For the arrival rates of 0.25 and 0.5 shown in the figures, $r = 1$ is the optimal parameter for the On-line Grouped TDMA. As with graphs of the uniform distribution, allowing groups to lend their slots for more than one frame does not show an improvement since the arrival rate is low enough that there is rarely a large backlog of packets for any one group. In contrast, Figure 5 shows how the performance of the On-line Grouped TDMA improves dramatically when we move from the $r = 1$ case to a generalized r. The graph depicts two lines, $r = 3$ and $r = 4$, for the On-Line Grouped TDMA protocol. For values of r less than 3, the protocol is unstable for this arrival rate and traffic distribution. For values of r greater than 4, we do not see any significant improvement beyond that of the $r = 4$ case. In comparing the On-line Grouped TDMA with the generalized directory protocol, it can be seen that similar to the arrival rates of 0.25 and 0.5 for the heterogeneous distribution, the arrival rate of 0.75 shows the directory approach has an advantage for moderate arrival rates and delays, but the On-line Grouped TDMA consistently has a smaller lower bound in it's delay and energy-conservation performance. Specifically, when delays of 50 slots can be tolerated, the On-line Grouped TDMA uses 30% less energy than that of the generalized directory and when a low delay is of greater importance, the On-line Grouped TDMA has less than half the average delay.

4 Conclusions

This paper considered the design of a general protocol for wireless mobile devices which allows a reduction in the amount of power necessary to receive transmissions while keeping the access delay low. The proposed protocol, termed *On-line Grouped TDMA* was compared with a generalized directory protocol which consisted of the intuitive adaptation of other current research [1, 3] to the slotted, half-duplex channel of our model.

It was shown that the On-line Generalized TDMA protocol consistently outperforms the directory protocol for various arrival rates of a uniform traffic distribution, as well as for low delay or low energy regions when utilizing a heterogeneous destination distribution. The directory approach was shown to be a successful approach for moderate arrival rates and energy-saving requirements. However the On-line Grouped TDMA consistently showed the ability to maintain a higher degree of energy conservation regardless of traffic distribution or rate with only a small cost in delay.

By allowing an "on-line" approach to lend slots between the various groups we obtain a flexible method for adapting quickly to changes in the traffic distribution while not adding significantly to the complexity of the mobile nodes or the base station.

References

[1] The editors of IEEE Wireless LAN Media Access Control (MAC) and Physical Layer (PHY) Specification IEEE 802.11 Draft Version 4.0; May 1996.

[2] Harris, E.P., Depp, S.W., Pence,E., Kirkpatrick, S., Sri-Jayantha, M., and Troutman, R.R. Technology directions for portable computers. *Proceedings of the IEEE*, 83(4):636–58, April 1995.

[3] Imielinski, T., Gupta, M., and Peyyeti, S. Energy efficient data filtering and communication in mobile wireless computing. In *Proceedings of the Second USENIX Symposium on Mobile and Location-Independent Computing*, pages 109–119, Berkeley, CA, USA, April 1995. USENIX Assoc, USENIX Assoc.

[4] Imielinski, T., Viswanathan, S., and Badrinath, B.R. Energy efficient indexing on air. In *Proceedings of the International conference on Management of Data - ACM-SIGMOD (Special Interest Group on Management Of Data) (SIGMOD Record)*, pages 25–36. ACM, May 1994.

[5] Powers, R.A. Batteries for low power electronics. *Proceedings of the IEEE*, 83(4):687–93, April 1995.