PAN-on-Demand: Leveraging multiple radios to build self-organizing, energy-efficient PANs

Manish Anand
Nokia Research Center
Palo Alto, CA
manish.anand@nokia.com

Jason Flinn
University of Michigan
Ann Arbor, MI
jflinn@eecs.umich.edu

ABSTRACT

We present PAN-on-Demand, a self-organizing wireless personal-area network (PAN) that balances performance and energy concerns by scaling the structure of the network to match the demands of applications. PAN-on-Demand autonomously organizes co-located mobile devices with one or more commodity radios such as Bluetooth and Wi-Fi to form a network that enables data sharing among those devices. It improves performance and extends battery life by switching between interfaces and opportunistically exploiting available power-saving modes. When applications are actively using the network, PAN-on-Demand offers high-bandwidth, low-latency communication; when demand is light, it adapts the network structure to minimize energy usage. Our results show that PAN-on-Demand reduces the average response time of PAN activities such as MP3 playing, e-mail viewing, and photo sharing by up to 92% and extends battery life by up to 47% compared to current PAN communication strategies.

1. INTRODUCTION

As the number of mobile computing devices owned by the average user grows, personal data is increasingly distributed among multiple locations. For example, photos, music, video, and text files are stored on both traditional computing devices such as laptops and handhelds, as well as on consumer-electronic devices such as MP3 players, digital cameras, and personal video players. Although many recent models of such devices have built-in Bluetooth or Wi-Fi (802.11b) wireless interfaces, transferring data between devices remains a tedious process that requires explicit user involvement. Thus, fast, energy-efficient mechanisms are needed to automatically share data among personal devices.

Personal-area networks (PANs) are a promising technology for minimizing the user distraction incurred while sharing data. When devices come within wireless radio range of each other, they can self-organize to form a PAN. The resultant PAN can present the user with a common, distributed namespace for all of the user's personal data located on its members [10, 18]. For example, an MP3 player with PAN support could offer to play not just the songs located on its local storage, but also the songs stored on a nearby laptop or portable media player owned by the same user.

Unfortunately, these advantages come at a price: PAN participation can substantially reduce the battery lifetime of mobile devices. Wireless interfaces consume a significant portion of the battery energy of a small computer. For instance, our results show that maintaining a single Bluetooth connection decreases the battery lifetime of an iPAQ 3970 handheld computer by over 14% and that keeping a Wi-Fi interface active shortens battery lifetime by 53%. Thus, energy considerations often prohibit maintaining continual PAN connections for sharing data among co-located mobile devices.

The solution to this dilemma is to scale the network to meet the demands of applications. When applications are actively sharing data, the PAN should operate in a high-power mode that allows high-bandwidth, low-latency communication. As PAN usage decreases, the PAN should automatically enter modes that require less power to operate (but offer less efficient communication). During idle periods when the network is unused, the PAN should expend only the minimal amount of energy required to maintain connectivity.

In this paper, we present PAN-on-Demand, a system that allows mobile devices to self-organize into a PAN and share data without explicit user involvement. PAN-on-Demand adopts a centralized approach to organization and communication that reduces the energy demands for poorly-provisioned devices within the network. It uses distributed self-tuning power management (DSTPM) to adapt its operating behavior to match the needs of applications. PAN-on-Demand switches between different radio technologies and employs power-saving strategies to offer high-bandwidth, low-latency communication when network usage is high, while requiring minimal energy when network use is low.

PAN-on-Demand reorganizes its network structure to match the communication patterns of its members. If a device is actively communicating, it is migrated to the hub of the network. This improves performance and extends battery life by shortening routes for data transfers. Reorganization also allows PAN members that are not actively communicating to save power by turning off network interfaces and using power-saving strategies. PAN-on-Demand's adaptive cost/benefit policies perform much better for MP3, e-mail, and photo sharing scenarios; compared to these prior policies, PAN-on-Demand extends battery lifetime by up to 47% and improves average response time by up to 92%.

2. PAN BACKGROUND/ASSUMPTIONS

A personal-area network connects mobile computers and consumer electronics devices owned by the same user through short-range wireless radios. Since PAN members are usually co-located, we assume that all PAN members are within wireless range of each other; thus, any two members can communicate directly if they

have the same type of wireless interface. We have validated that this assumption holds using our experimental testbed, which places PAN members up to 20 feet apart.

New generation of both mobile computers and consumer-electronic devices such as handhelds [16], cell phones [17], and digital cameras [15] are now being shipped with both Wi-Fi and Bluetooth. These interfaces have different characteristics that give substantial opportunity for optimization. For instance, Bluetooth provides lower throughput than Wi-Fi and increases network latency; yet, its power consumption is typically much less. PAN-on-Demand uses the multiple interfaces to adapt network behavior to meet application demand. It also uses available power-saving strategies for each interface. For example, it reduces the power required by a Bluetooth interface by dropping active connections and re-establishing them when needed.

PAN-on-Demand allows any member of the PAN to be either the initiator or recipient of network communication. This policy requires that each member keep at least one network interface active at all times so that it can receive incoming traffic. To save energy, a PAN member may turn off all of its interfaces except the one that uses the least power. For the remaining interface, a member may employ power-saving strategies such as dropping active connections. Ideally, during idle periods when a member is not actively communicating, it should expend only the minimal energy needed to allow it to be signaled when another member wishes to communicate.

In developing PAN-on-Demand, we felt it was important to support current mobile computers that have commodity wireless interfaces. Thus, PAN-on-Demand currently assumes that all members have at least a Bluetooth interface, and that members may optionally have Wi-Fi interfaces capable of ad-hoc communication. It uses Bluetooth for signaling, control traffic, and device discovery. Our implementation uses Linux's BlueZ Bluetooth protocol stack [4]. However, PAN-on-Demand's algorithms are general and can support other network technologies as they become available. To use a new network technology, we perform a characterization similar to that described in Section 8 for Bluetooth and Wi-Fi. Potentially, such characterization could be done by the manufacturer of each wireless interface and specified in the network device driver.

3. RELATED WORK

PAN-on-Demand distinguishes itself from previous work on PANs and use of multiple radios [19, 21, 23] by using adaptive policies to the choose between interfaces, routes, power-management modes, and network topologies. While previous systems have tackled one or two of these policy decisions in isolation, PAN-on-Demand takes a more holistic view by applying a uniform cost metric to all power to performance tradeoff that occur during PAN operation. Another major contribution of this work lies in showing that previously-proposed static policies that always choose low-power interfaces for transmission in preference to high-power ones and hierarchical policies that always use low-power radios to wake up and use high-power interfaces perform poorly in many application scenarios.

Bahl et al. [3] proposed splitting wireless communication into low-power radio (LPR) tasks and high-power radio (HPR) tasks to leverage the benefits offered by multi-radio systems. Hierarchical radio strategies [20, 23, 25] use the LPR to signal the HPR for data transmission, while CoolSpots [19] uses passive bandwidth monitoring to switch between the LPR and the HPR. These approaches are applicable to specific usage scenarios. For instance, a hierarchical strategy is ideal if the cost of transitioning radios is negligible or if devices intermittently exchange large amounts of data. On the other hand, the bandwidth-X policy used by CoolSpots is useful when latency is critical. Our experiments show that these static approaches perform poorly when application load is non-ideal. In contrast, PAN-on-Demand's adaptive policies can perform substantially better in mixed application scenarios, and adapt to perform approximately as well as each static strategy in its ideal scenarios.

Contact Networking [6] ignores the differences in the data-rate offered by different radios and only employs the communication range to choose the right communication interface. Thus, it always chooses the Bluetooth for short-range PAN communication. Similar to PAN-on-Demand, CoolSpots [19] and PPM [21] also consider the bandwidth and latency characteristics of different interfaces to switch between them for improved device power management. Unlike PAN-on-Demand, they focus only on the reduction of energy consumed by the network interface on a device and are not directly applicable to a PAN with more than two members. Further, they do not deal with device discovery, routing, or the topology of the network.

MOPED [5] allows a mobile computer to aggregate transmission bandwidth across multiple non-interfering wireless channels. Aggregation is not a good choice in the PAN environment with Bluetooth and 802.11b as their transmissions interfere with each other [8, 12]. Thus, multiplexing data transmissions across the two interfaces would decrease throughput and increase energy usage.

4. DESIGN PRINCIPLES

We begin by describing the principles we followed while designing PAN-on-Demand.

4.1 Some peers are more equal than others

Each member should contribute to the PAN in proportion to its abilities. At first glance, an architecture in which PAN members act as peers, with equal roles and duties, seems promising. However, mobile computers currently exhibit a wide disparity in provisioning. Some mobile devices, such as laptops, have large battery capacity, ample storage, and substantial processing power. Other devices, such as MP3 players and cell phones, are poorly provisioned, often having smaller batteries and less storage capacity. An architecture that treats all mobile computers equally can exhaust the resources of poorly-provisioned devices while barely taxing the resources of well-provisioned devices.

For example, maintaining PAN membership requires a constant power expenditure. If all devices contribute an equal amount of power, a handheld computer with a small battery would quickly run out of power after joining a PAN. At the same time, a laptop with several more orders of magnitude of energy in its battery would be relatively unaffected by the power drain of PAN membership.

PAN-on-Demand adopts a centralized network architecture that asks more from well-provisioned devices while demanding less from poorly-provisioned ones. At any given time, a single device acts as the network *manager*, while the other devices in the network act as *workers*. When possible, power-intensive tasks are shifted to the manager to minimize the power drain of PAN membership for workers.

4.2 Scale the network with demand

The characteristics of a PAN should match the immediate needs of its applications. Since current wireless technologies offer a trade-off between power and performance, selecting the correct interface can substantially affect system behavior. When an interactive application is actively using the PAN, a network that delivers crisp performance to the user seems best. When the network is idle, a low-power option that extends the battery life of member devices

is preferable. Clearly, no static choice for network communication can be best in both scenarios. Therefore, PAN-on-Demand uses an *adaptive* strategy that dynamically chooses a network interface and power management strategy that matches the activity of the applications using the network.

In PAN-on-Demand, each worker independently chooses which network interface it will employ for communication and which power saving mode it will use for that interface — each worker makes this choice based on the network traffic that it expects to see in the near future. The manager allows this flexibility by supporting multiple communication modes. For instance, a PAN-on-Demand manager can communicate with workers that prefer high quality using its Wi-Fi interface, while simultaneously communicating with workers that desire energy-efficiency using its Bluetooth interface. Section 7 describes how these decisions are made.

4.3 Minimize user distraction

A PAN should require minimal supervision from its user. User attention is a scarce resource in a mobile environment in which people may be walking, driving, or performing other critical activities while using their computers.

Potentially, a PAN member could save energy by turning off all its network interfaces. However, before another device can initiate communication, its user would have to explicitly turn an interface back on. In effect, this would mirror the manual synchronization process currently used by many cameras and MP3 players since the user would need to initiate the transfer at both communication endpoints. This process is worse for applications such as search that involve many devices. For instance, to locate a particular MP3 file within the PAN, the user would need to turn on an interface for all devices that might store the file.

In contrast, PAN-on-Demand allows mobile computers to self-organize without user intervention. Computers owned by the same user that are within wireless radio range detect each others' presence and automatically form a PAN with one computer being the master and the others acting as workers. The resultant PAN requires some additional energy to maintain connectivity, but requires no explicit user interaction to transfer or search for data. The advantage of this design is that the user is not distracted by the need to explicitly enable communication on any single device since PAN-on-Demand keeps at least one interface on each member active to receive communication requests.

5. COST METRIC

PAN-on-Demand uses a common cost metric to evaluate different options for network topology and communication strategies. The cost metric balances the principles of maximizing performance and battery lifetime while minimizing user distraction. It measures performance as the time the user waits for data to be transferred. This includes both the time to activate network interfaces, as well as the time to transfer the data over the PAN. Background transfers have no impact on performance since the user is not waiting for them to complete.

Our cost metric also considers how much an activity would reduce the battery lifetime of the computers participating in the PAN. For each computer, we divide the amount of energy that the activity is predicted to use by the average power usage of the machine when it is idle. The result is the estimated decrease in the computer's battery lifetime as a result of performing the activity. The cost metric sums this value for all PAN members.

In the sensor and ad-hoc networking communities, cost metrics often maximize the lifetime of the entire network [7, 24] — this is appropriate when the primary objective of the network is to perform a collaborative activity. However, in a PAN, the primary function of mobile computers is not to participate in the network — instead, it is to play music, take pictures, display e-mail, etc. Thus, it is appropriate to subordinate the needs of the network as a whole in preference to the needs of individual devices. Our cost metric captures this behavior by minimizing the impact of network participation on each computer.

To equate the relative impact of performance and battery lifetime, we turn to the user. PAN-on-Demand has a single knob, ranging in value from 0 to 1, that can be tuned to bias decisions toward performance or energy conservation. The simplicity of this interface is driven by the principle of minimizing user distraction. More complex interfaces are certainly possible; e.g., separate knobs or different weights for each PAN member — however, our interface is chosen to give the user the most control with the least effort.

In summary, PAN-on-Demand uses the following metric:

$$C = kT + (1-k) \sum_{j=1}^{members} E(j)/P_{base}(j) \qquad (1)$$

where C is the calculated cost, k is the global knob that is adjusted by the user, T is the time the user waits for the activity to complete, $E(j)$ is the amount of energy used by PAN member j to perform the activity, and $P_{base}(j)$ is the *base power*, the amount of power consumed by member j when it is turned on but running no additional activities.

6. SELF-ORGANIZATION

PAN-on-Demand provides an automated, energy-efficient mechanism that enables co-located personal devices to self-organize into a PAN. Self-organization consists of three phases: discovery, network entry, and reorganization. The next three subsections describe these phases in more detail. Section 6.4 describes how members leave the PAN.

6.1 Discovery

Isolated devices and PAN managers periodically initiate the discovery phase by performing a Bluetooth inquiry to locate nearby devices owned by the same user. PAN workers do not need to perform inquiries since their network manager performs this activity on their behalf. This is one benefit of centralizing functionality at the manager: only the manager expends battery energy to perform common activities.

All Bluetooth-enabled devices within radio range respond to an inquiry if they do not have an active connection in which they are acting as a slave device. The inquiry response contains the Bluetooth device address of the responder — this address is sufficient to uniquely identify each device. All isolated devices and PAN managers respond to an inquiry. Since PAN-on-Demand does not require workers to maintain an active connection, a worker may respond to an inquiry if it does not have an active Bluetooth connection.

Each device stores a *device list* that contains all personal devices owned by its user. The device list is sorted by capabilities, with well-provisioned devices that are most likely to serve as managers at the top of the list and poorly-provisioned devices at the bottom. The device list allows each device to identify which set of devices within its radio range belong to the same user. If a response is received that includes an identifier not on the device list, then the responding device is assumed to belong to a different user and is ignored.

The device list is sorted so that the devices with greater base power precede devices with lower base power. Devices with the

highest base power tend to minimize the cost metric when they serve as manager, since the constant power expenditure of being the manager has the least relative impact on their battery lifetime. For instance, if serving as manager consumes an extra 0.5 Watts, the battery lifetime of a handheld with base power usage of 1 Watt will be reduced by 33% if it is the manager. Alternatively, a laptop with base power usage of 30 Watts will see its battery lifetime reduced by only 1.7% if it is the manager.

We currently assume that the device list is statically configured on each device owned by a common user. We plan to eventually store this list in the EnsemBlue distributed file system [18], which can replicate changes made on one device to the other devices owned by a user.

6.2 Network entry

During the network entry phase, an *initiating device* attempts to form a PAN with a *discovered device*. The discovered device is the highest-ranking device on the device list that responded to the initiating device. We first describe network entry assuming the initiating device is an isolated device; we then extend the protocol to the case where the initiating device is a PAN manager.

If the initiating device is an isolated device, it connects to the discovered device as a Bluetooth slave. The discovered device may itself be an isolated device — in this case, the discovered device accepts the connection, sends back an acknowledgment, and acts as a manager for the newly formed PAN. If the discovered device is already the manager of an existing PAN, it accepts the connection, sends back an acknowledgment, and adds the initiating device as another worker in its PAN.

Infrequently, the discovered device may be a worker in an existing PAN. This case occurs when the Bluetooth connection between the worker and its manager has been previously dropped to save power. In this case, the discovered device sends back a negative acknowledgment that includes the Bluetooth device address of its manager. The initiating device then connects to the discovered device's manager. If the initiating device is unable to establish a connection at any point during this process, the network entry phase ends. If that happens, the initiating device must re-run the discovery protocol before it joins a PAN.

If the initiating device is a PAN manager with one or more workers, it waits for existing communication within its PAN to quiesce. Once no data transfers are in progress, it sends a message to its workers informing them of the Bluetooth device address of the discovered device. It terminates its existing Bluetooth connections and attempts to connect to the discovered device using the protocol described above for isolated devices. Its workers also attempt to connect to the discovered device using the same protocol.

6.3 Reorganization

Once a PAN is formed, the reorganization phase begins. During this phase, the manager monitors PAN traffic to analyze communication patterns. If it was not the ideal manager for recent communication, the current manager triggers a network reorganization during which the more optimal member assumes the role of manager. The reorganization process is based on the assumption that future traffic will resemble the traffic seen on the network in the recent past. Of course, this assumption could be wrong. In that case, another node will likely be selected as manager in the future, triggering a subsequent reorganization.

The PAN manager logs the size of each transfer and the state of all network devices along the possible routes for that transfer. This is another benefit of centralization — since the manager sees all communication requests and responses, no further communication

is required to get information for the log. The transfer data are written to a circular buffer that holds a maximum of 50 entries — after the buffer fills, the oldest record is evicted when space is needed. For each PAN member, the current manager calculates the amount of time the transfer would take if that member had been the manager of the network at the time the transfer began. It also calculates the amount of energy that would be expended by each participating member. It uses the metric in Equation 1 to determine the cost that would have been incurred for that transfer based on the calculated time and energy values. The manager calculates the total cost over the period covered by the transfer log, assuming that a particular device had been the PAN manager:

$$C_{total}(M) = \sum_{i=1}^{transfers} C_i(M) + (1-k)\frac{P_{manager}(t_n - t_1)}{P_{base}(M)} \quad (2)$$

C_i is the cost of the i'th transfer if M had been the manager. $P_{manager}$ is the constant power drain that a member incurs by serving as the manager. $t_n - t_1$ is the time between the first and last transfers recorded in the log.

If the manager determines that C_{total} is smaller for one of its workers than it is for itself and if the difference between the two values is greater than a threshold, it triggers a reorganization. We set the threshold to be the cost of the reorganization — this is the additional energy expended by all other workers to disconnect and reconnect their Bluetooth interfaces, as well as the energy required to transfer the manager role between the two devices.

The manager waits for all current data transfers to end. It sends a message to all of its workers informing them of the reorganization and the Bluetooth device address of the new manager. It disconnects from its workers and reconnects to the new manager as a Bluetooth slave. All other devices within the PAN reconnect to the newly chosen master.

6.4 Leaving the PAN

Network partitions and device disconnections are discovered passively, when a communication request fails to complete. When a PAN manager fails to contact a worker to satisfy a request, it assumes that the worker has left the PAN. The manager removes the worker from its list of members. When a worker fails to contact its manager, it becomes an isolated device. It periodically initiates device discovery and attempts to form a new PAN when it next locates a nearby device owned by the same user.

7. DISTRIBUTED STPM

PAN-on-Demand uses distributed self-tuning power management (DSTPM) to set the power states of the network interfaces on each member. These algorithms are based on the STPM algorithms [1, 2] that we previously developed to manage a single storage and/or network device on a single computer. In this paper, we extend these algorithms to a distributed environment in which we manage multiple network interfaces on multiple computers.

Extending self-tuning power management to the PAN environment presented several challenges. First, we defined an appropriate cost metric that expressed performance and energy conservation goals for all the members of a network, not just a single computer. This metric has already been described in Section 5. Next, as discussed in Section 7.1, we defined the possible strategies for transmitting data within the PAN and developed a reactive strategy for dynamically choosing the best transmission strategy and network route. Finally, we developed proactive strategies that switch between multiple interfaces and manage control connections. These strategies are discussed in Sections 7.2 and 7.3.

7.1 Choosing the best transfer strategy

When a PAN-on-Demand member initiates a new data transfer, it uses DSTPM to choose the best communication strategy. This is a purely reactive decision that minimizes the cost of the transfer given the expected size of the transfer, the performance and energy characteristics of the available network interfaces, and the present state of the network. The questions to be answered are:

- What route should be used to transfer the data?
- What wireless technology should be used for each hop along that route?

The initiator, which may either be sending or receiving data, sends a communication request to the target device via the PAN manager. This message contains the expected size of the transfer (if known to the initiator), the type and power state of the initiator's network interfaces, and the initiator's base power. The manager appends its interface state and base power to the message.

After the target device receives the communication request, it has all the information it needs to evaluate the cost metric described in Section 5. The target device first computes the expected time and energy to perform the transfer for each possible communication strategy. Such a calculation is straightforward given the size of the data to be transferred and a characterization of the network interface such as the ones in Section 8.

If either the initiating or target device is the PAN manager, only a one-hop route is considered. In this case, the decision is whether to employ Bluetooth or Wi-Fi for the transmission. If both devices are workers, then the three strategies are possible. The data may be transferred using a Bluetooth connection, in which case a two-hop route is needed (Bluetooth is a master-slave protocol). The data may also be transferred using a one-hop Wi-Fi connection. Alternatively, data may be routed through the manager with one hop using Bluetooth and the other Wi-Fi. A hybrid Bluetooth/Wi-Fi strategy is most useful when one worker has disabled its Wi-Fi interface at the start of the transfer while the other has an active interface. The hybrid strategy can also be used when one member does not have a Wi-Fi interface. While it is feasible to transfer data using a two-hop Wi-Fi route through the manager, PAN-on-Demand always chooses the one-hop route since it offers better performance and energy conservation.

The target device evaluates the cost metric for all feasible transfer strategies. It chooses the strategy with the smallest cost and communicates its decision back to the initiator via the manager. The target device appends the state of its interfaces to its response so that the manager can log the information and use it to determine when reorganization is needed. Both the manager and initiator examine the reply to determine if they need to enable a network interface for the upcoming data transfer.

7.2 Proactive energy savings

DSTPM reduces the energy cost of PAN membership by turning off network interfaces and employing power-saving strategies when applications are not using the PAN. Currently, a worker can be in one of four states:

- **WiFi-only**. Both the Bluetooth and WiFi interfaces are active, and a Bluetooth connection exists with the manager. This state uses the most power, but also offers the best performance since no mode transition is required to send either control or data traffic.

- **BT-connected**. The worker's WiFi interface is disabled, its Bluetooth interface is active, and an active connection exists with the manager. This state uses less power than the WiFi
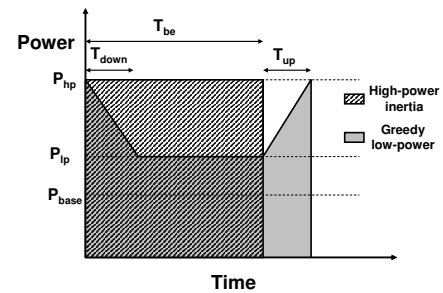

Figure 1: Break-even illustration

state, but offers poorer performance since data must either be sent via Bluetooth or the transfer must wait for the WiFi interface to be turned on.

- **BT-sniff**. The WiFi interface is disabled. The Bluetooth connection is placed in a power saving *sniff* mode. In this mode, connected devices initiate network communication with one another only at pre-determined periods, allowing them to disable their radios at other times. Sniff mode adds latency to Bluetooth communication but offers better energy conservation.

- **BT-unconnected**. The WiFi interface is disabled. The Bluetooth interface is active but no connection exists between the worker and the manager. This state uses the least energy but offers the poorest performance since the connection must be re-established before the next transfer can begin.

While most WiFi interfaces support a power-saving mode (PSM) when communicating with an AP, we could find no commercial hardware that currently supports PSM for ad-hoc communication. Were such a mode to become available in the future, our algorithms are general enough to incorporate WiFi-PSM as another data transmission alternative.

PAN members can enter a high-power state either as the result of a data transfer (as described in the previous section) or due to a proactive transition (as described in the next section). If the time and energy cost of transitioning between power modes were negligible, then each member would simply enter the minimum-power mode when it is not participating in data transfers. However, our results show that it often takes several seconds to enable network interfaces and transition between power modes. Since data transfers are often closely correlated in time, a new transfer is more likely to occur if transfers have occurred in the recent past. In such circumstances, a proactive strategy that keeps interfaces in high-power states for a short time after a transfer ends has been shown to work best [1, 14].

DSTPM uses a break-even timeout strategy [9] to decide when to transition from a high-power state to a low-power state. This strategy compares a *high-power inertia* policy that continues to employ the current high-power state with a *greedy low-power* policy that immediately switches to a low-power state. The break-even time, T_{be}, is the inter-arrival time for data transfers for which these two strategies yield equivalent results. If the next transfer starts before the break-even time, the high-power strategy is superior; otherwise the low-power strategy is best.

Figure 1 illustrates how DSTPM calculates the break-even time. The y-axis on this graph is power usage, and the x-axis is time.

The energy expended by each strategy is the area under its curve. Thus, the energy used by the high-power inertia strategy is given by the area of the striped region: specifically, it is product of P_{hp} (the power used in the high-power mode) and T_{be} (the time until the next activity begins). The cost of the high-power inertia strategy is:

$$C_{hpi} = (1-k)\frac{P_{hp}*T_{be}}{P_{base}} \qquad (3)$$

Since the high-power strategy does not add any delay, its cost is measured solely in the energy cost of keeping the member in the high power mode.

In contrast, the cost of the greedy low power strategy consists of both the energy used by the PAN member and the time the user must wait for the transition back to the high-power mode before the next transfer begins (T_{up}). The shaded area of Figure 1 shows the energy used by the greedy low-power strategy — this consists of the sum of the energy used in three distinct regions. The energy used in the first region, E_{down}, is the energy used by the device to transition from the high-power to the low-power mode. The second region is the period the device operates in the low-power mode — the energy of this region is the product of P_{lp} (the power used in the low-power mode) and $T_{be} - T_{down}$. The final region is the transition to the high-power mode before the next activity begins — the energy used in this region is given by E_{up}. According to the cost metric of Equation 1, the cost of the greedy low-power strategy is:

$$D_M = \frac{E_{up} + E_{down} - P_{lp}*T_{down}}{P_{base}}$$
$$C_{glp} = k*T_{up} + (1-k)*\left[\frac{P_{lp}*T_{be}}{P_{base}} + D_M\right] \qquad (4)$$

D_M is the extra energy expended by the greedy low-power strategy beyond that required to keep the device in the low-power mode for the break-even time.

The value of T_{be} is calculated by equating C_{glp} and C_{hpi}. Solving for T_{be} yields:

$$s_M = \frac{P_{hp} - P_{lp}}{P_{base}}$$
$$T_{be} = \frac{k*T_{up} + (1-k)*(D_M)}{(1-k)*(s_M)} \qquad (5)$$

s_M is the surplus rate of battery consumption by the high-power mode compared to the low-power mode.

The above calculation of T_{be} can be applied to any transition that affects only a single machine. For instance, in our current implementation, it applies to the transition from the Wi-Fi mode to the BT-connected mode. In contrast, the transition from the BT-connected mode to the BT-unconnected mode affects both the worker and manager since the Bluetooth connection is dropped at both ends. The decision must therefore account for the energy used by both devices. The calculation of the break-even time for this transition is:

$$T_{be} = \frac{k*T_{up} + (1-k)*(D_{wkr} + D_{mgr})}{(1-k)*(s_{wkr} + s_{mgr})} \qquad (6)$$

Each worker uses DSTPM to independently change the power states of its interfaces. Thus, a worker that is actively participating in data transfers may be in its high-power mode, while another worker that has been idle for a substantial time may be in its lowest-power mode. The manager independently decides to turn off its Wi-Fi interface. However, the manager defers the decision of when to drop Bluetooth connections to its workers.

7.3 Proactive performance improvement

In order to transfer data, a PAN worker must leave the unconnected state since a Bluetooth connection is required to send or receive control messages from the manager. However, a worker need not ever enter the Wi-Fi state to transfer data. If each transfer is small, the transition cost of turning on its Wi-Fi interface will be greater than the potential performance and energy benefit of performing the transfer using Wi-Fi instead of Bluetooth. In this case, the reactive strategy described in Section 7.1 will always use Bluetooth to transfer the data.

However, if a member were to perform several short transfers in short succession, the cost of the transition could be amortized over many transfers and lead to an overall reduction in the cost metric. To deal with such instances, a proactive strategy is needed to transition to the Wi-Fi state.

When a worker uses Bluetooth, it calculates the time and energy reduction (if any) that it would have seen had its Wi-Fi interface been active at the start of the transfer. Using the cost metric of Section 5, it calculates the opportunity cost of being in the BT-connected state. It adds the opportunity cost (if any) to a running total of such costs every time it performs a Bluetooth transfer. From this total, it subtracts the cost of the additional energy that would have been expended keeping the Wi-Fi interface active during the idle periods between transfers. The running total is not allowed to go below zero.

When the accumulated opportunity cost exceeds a threshold, the PAN member turns on its Wi-Fi interface. This decision reflects the assumption that it is likely to see several more short transfers if it has seen several in the recent past. PAN-on-Demand chooses the threshold based on the break-even calculation in Section 7.2. Since the member will remain in the Wi-Fi mode for T_{be} if it sees no more data transfers, the cost of the incorrect transition, C_{thresh}, is:

$$C_{thresh} = (1-k)*\left[T_{be}*s_M + D_M - \frac{P_{lp}*T_{up}}{P_{base}}\right] \qquad (7)$$

8. CHARACTERIZING THE PAN USAGE

The DSTPM algorithms are general; they do not assume particular performance and energy usage values for any network interface. Thus, before using them, one must first characterize the network interfaces that will be employed. In this section, we report on our characterization of two interfaces: Bluetooth and Wi-Fi.

8.1 Methodology

We deployed PAN-on-Demand on an experimental testbed consisting of several HP iPAQ 3970 handheld computers running Linux 2.4.18-rmk3 kernels. Each iPAQ has a built-in Bluetooth interface and a Cisco 350 802.11 PCMCIA card. The iPAQs are deployed approximately 10–20 feet apart in an office environment. The transmit power of each Cisco card is set to 1 mW. We found this value to be sufficient to transmit the distances required of our testbed, despite the presence of several nearby Wi-Fi access points.

We measured performance using the `gettimeofday` system call. We measured energy used by the iPAQs with an Agilent 34401A digital multimeter. We removed the batteries from each iPAQ and sampled current drawn through the iPAQ's external power supply approximately 50 times per second. We calculated system power usage by multiplying each current sample by the mean voltage drawn by the iPAQ — separate voltage samples are not necessary since the variation in voltage drawn through the external power supply is very small. We calculated total energy usage by multiplying the average power drawn during benchmark execution by the time needed to complete execution.

Interface State	Idle Power (Watts)	Latency (seconds)	Throughput (Kbps)	Transmit Power (Watts)	Receive Power (Watts)
$BT_{connected}$	0.24 (0.00)	0.032 (0.003)	520 (4)	0.69 (0.01)	0.49 (0.00)
Wi-Fi	1.44 (0.00)	0.002 (0.000)	4429 (64)	1.72 (0.01)	1.86 (0.02)

This figure shows the time and energy characteristics of Bluetooth and Wi-Fi interfaces on an iPAQ 3970 handheld. Each value shows the mean of five measurements with standard deviation in parentheses.

Figure 2: Performance and energy characteristics of wireless interfaces

Interface	Activity	Time (seconds)	Energy (J)
Bluetooth	Connection	3.18 (2.69)	1.33 (1.19)
	Disconnection	3.24 (0.37)	1.13 (0.14)
Wi-Fi	Switch On	3.04 (0.02)	3.99 (0.37)
	Switch Off	2.06 (0.13)	2.93 (0.03)

This figure shows the transition costs of Bluetooth and Wi-Fi interfaces in on iPAQ 3970 handheld. Each value shows the mean of five measurements with standard deviation in parentheses.

Figure 3: Transition costs of wireless interfaces

8.2 Measurements

With no network interfaces active, an idle iPAQ uses 1335 mW of power. If the Bluetooth interface is active, the iPAQ uses an additional 125 mW. Since the Bluetooth interface must be kept continually active for PAN-on-Demand, this 125 mW represents the minimal cost of PAN participation. It is encouraging that the relative power consumption is less than 10% of the idle power of a small, mobile device such as the iPAQ.

Figure 2 shows the additional power consumed by the Bluetooth and Wi-Fi interfaces. Maintaining a Bluetooth connection between manager and worker uses an additional 240 mW on the worker and 120 mW on the manager. For each further connection, the manager expends an additional 20 mW. The substantial power cost of maintaining connections motivates our decision to drop Bluetooth connections during periods of inactivity. Thus, an idle iPAQ in PAN-on-Demand expends only the 1460 mW of base power required to enable other devices to initiate communication.

Figure 3 shows the time and energy required to transition between the various modes of PAN operation. There is substantial variation in the time required to create a Bluetooth connection due to the Bluetooth frequency synchronization delay between two machines [11, 22]. Thus, we have observed the time to establish a connection vary from 1.2 seconds to as much as 10.3 seconds. Since connection establishment adds considerable latency to a data transfer, PAN-on-Demand maintains Bluetooth connections when it predicts that the network is likely to be used in the near future.

Bluetooth specification [13] recommends the use of sniff and park modes for low-power operations. In practice, we found the sniff mode to be the most flexible and applicable for our tests which was consistent with previous observations [19]. Sniff mode can be configured with a interval, attempt and timeout parameters. Sniff interval specifies how often should the device wake up, i.e., the length of time between subsequent wake ups ($T_{interval}$). When the device wakes up, sniff attempt specifies how long should the device stay awake, i.e., the length of time that workers need to synchronize with the manager ($T_{attempt}$). The value of $T_{attempt}$ is governed by the channel conditions and should be set to a higher value when more retransmissions are required due to interference. When the

device receives a transmission, sniff timeout specifies how long should the device wait for the next transmission, i.e., the length of time that the receiver should continue listening after receiving a transmission ($T_{timeout}$).

For a fair comparison with our PAN-on-Demand algorithms, we hand crafted the Bluetooth sniff mode settings to make it most power-efficient using multiple measurements and offline characterizations. We found that the setting of one RTT or greater is enough for the devices to receive transmissions from each other since our experimental environment has relatively low levels of interference. Thus, we set the value of $T_{attempt}$ to 40 ms (slightly larger than RTT). In order to balance the trade-off between request latency and power consumption for the chosen value of $T_{attempt}$, we adjusted the sleep period, $T_{interval}$, so that the devices would consume only 1% more power than Bluetooth unconnected. This period was 400 ms for our experiments. Hence, the expected latency for sniff mode in our experiments is 200 ms. We set the $T_{timeout}$ value to be slightly larger than 2 times RTT to support request and respond workload, i.e., 80 ms. Our characterization showed that the transition to sniff mode can be achieved at a negligible cost (less than 10 ms and 10 mJ).

9. EVALUATION

Our evaluation answers the following questions:
- How much time and energy is consumed by PAN self-organization and maintenance?
- How much does PAN-on-Demand improve response time and battery lifetime compared to other PAN communication strategies?
- How much benefit is achieved by reorganizing the network structure?

We used the experimental testbed and methodology described in Section 8.1. When evaluating PAN-on-Demand, we set the user-adjustable knob to 0.5 to give weight to performance and battery lifetime.

9.1 Cost of self-organization

We began by measuring the time and energy required to perform a Bluetooth inquiry. In PAN-on-Demand, all managers and isolated devices perform a periodic inquiry to detect if other devices have come within radio range. Our measurements show that an inquiry takes 10.3 seconds and uses 4.3 Joules of energy. These results show that PAN-on-Demand's current strategy of performing an inquiry every five minutes reduces the battery life of an iPAQ by slightly less than 1%.

Next, we measured the time and energy needed for a new member to join the PAN. In this experiment, an isolated device comes within wireless range of an existing PAN with a manager and a single worker. The isolated device joins the PAN as an additional worker. Given an inquiry period of 5 minutes, the expected time for the isolated device to begin a new inquiry after coming within range
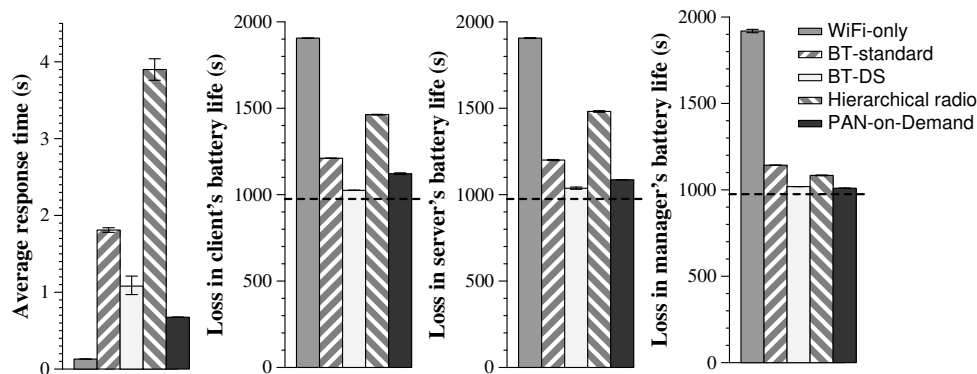
This figure shows results for a workload that emulates a user fetching e-mail from a storage device. Each impact on the battery lifetime value shows the mean of three trials and each average response time value shows the mean of nine trials. The error bars are 90% confidence intervals.

Figure 4: Benefit of PAN-on-Demand for e-mail viewing

is 150 seconds. Once the inquiry begins, the new device takes, on average, an additional 12.2 seconds to join the PAN. During this period, the isolated device uses 22.7 Joules of energy and the PAN manager uses 6.0 Joules.

Finally, we measured the time and energy needed to reorganize the PAN. This experiment starts with a PAN consisting of three iPAQs in the BT-connected state. The existing PAN manager selects one of its workers to act as the new manager, signals the reorganization to its workers, and disconnects. The newly selected manager reestablishes Bluetooth connections to the other two computers. The original manager joins the new PAN after 4.5 seconds, and the other worker joins 3 seconds later. The original manager uses 3.6 Joules of energy during reorganization, the newly selected manager uses 3.8 Joules, and other PAN members use 3.1 Joules.

9.2 Impact of PAN-on-Demand

Next, we compared PAN-on-Demand to four current PAN communication strategies. The first comparison is a WiFi-only strategy where the members communicate directly with one another using only their Wi-Fi radios. PAN members keep their Wi-Fi interfaces continuously active, but disable their Bluetooth interfaces. The second and third strategies are Bluetooth-only strategies where members communicate with one another using only their Bluetooth interface. Bluetooth-standard strategy consists of all members maintaining a continuous Bluetooth connection with the manager and utilizing that link to transfer data with one another. Bluetooth-DS strategy is a distributed implementation of Bluetooth standard with sniff mode where the members can use the scatternet capability to directly communicate with one another. Note that scatternet is not supported by all implementations of the Bluetooth stack but the results of Bluetooth-DS helps us better understand the impact of sniff mode with this capability when present. The fourth is a hierarchical multiple radio strategy, similar to Wake-on-Wireless [23, 25], that uses the Bluetooth to signal Wi-Fi radios to switch-on. PAN members exchange their intent to share data on the Bluetooth channel and perform the data transfer on Wi-Fi channel. Since, this strategy does not implement any proactive technique to stay awake, Wi-Fi radios are immediately switched off after every data transfer.

Since PANs are an emerging technology, there do not yet exist traces of actual PAN usage. We therefore chose three applications (MP3 playing, e-mail viewing, and photo sharing) that we

felt would be likely candidates for this emerging environment. We evaluated the impact of PAN communication strategies on these applications using the iPAQs from our testbed as two WPAN workers and a manager.

For our evaluation, we compared the PAN communication strategies across two user-observable properties. First, we measured the average response time for all requests in the application trace. Second, we determined the change in battery lifetime of each PAN member by measuring the energy expended by each PAN member during the execution of the application trace. In order to calculate the impact on battery lifetime, we assume that the device would continue operating at its base power after the trace execution. Thus, the minimal impact on battery lifetime is the amount of idle time in each trace — this is shown by the dashed lines in Figures 4–6.

9.2.1 E-mail workload

We first explored a scenario in which a user views e-mail on a PDA. We assume that mail has been previously downloaded from a server and is currently stored on a storage device within the PAN. Each message is fetched from the storage device when it is viewed by the user. The workload consists of 975 seconds of e-mail client activity previously collected by our research group [1]. The messages vary in size from 1 KB to 1.35 MB.

As expected, the WiFi-only strategy offers the best performance since it never disables the high-speed Wi-Fi interface. The cost of good performance is a substantial reduction in battery lifetime for all members. In comparison, the Bluetooth-only strategy increases battery lifetime on the three iPAQs by using 36–40% less energy to complete the trace. Yet, Bluetooth-only also increases the average response time by more than an order of magnitude (1.7 seconds), because it is much quicker to transfer large mail messages via Wi-Fi.

In general, the hierarchical strategy performs worse than Bluetooth only strategies. It more than doubles average response time and requires substantially more energy from the client and server. Most data transfers for this workload are small. The hierarchical strategy performs poorly for short transfers because the time and energy required to enable and disable the Wi-Fi interface is not recouped during the short transfer. In contrast, if all data transfers are large (as is the case in the forthcoming MP3 workload), a hierarchical strategy can perform well.
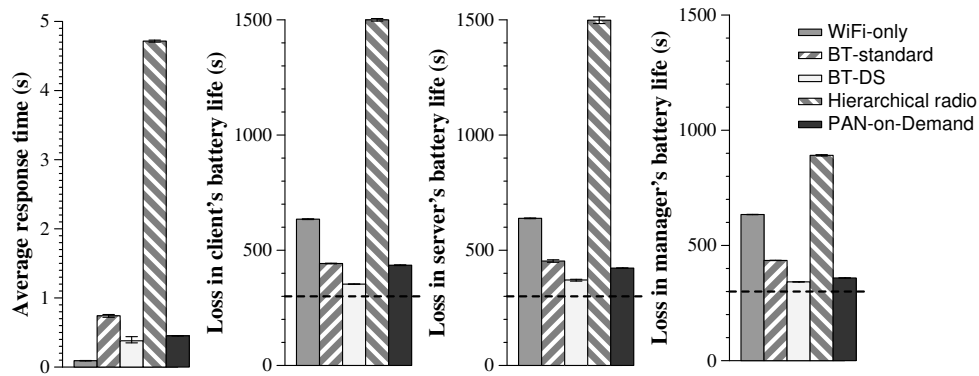
This figure shows results for a workload that emulates a user fetching photos from a storage device and viewing them. Each impact on the battery lifetime value shows the mean of three trials and each average response time value shows the mean of nine trials. The error bars are 90% confidence intervals.

Figure 5: Benefit of PAN-on-Demand for sharing photos

PAN-on-Demand yields the best battery lifetime of any strategy. PAN-on-Demand garners additional energy savings during idle periods by tearing down connections. It also saves both time and energy for large transfers by performing them using Wi-Fi instead of using Bluetooth. Compared to the WiFi-only strategy, the three devices use 41–47% less energy. The energy savings compared to the Bluetooth-only strategy are smaller (7–12%), but PAN-on-Demand yields an average response time that is 2.5 times faster.

In presence of sniff mode with scatternet capability, shown as BT-DS in the figure, the response time of Bluetooth-only strategies are cut in half as the communication between members has to traverse only 1-hop instead of 2. Bluetooth-DS is only 37% slower and outperforms PAN-on-Demand by 5–9% in extending battery life for client and server. PAN-on-Demand is able to match the energy-savings of Bluetooth-DS by employing sniff mode also.

Compared to the hierarchical strategy, PAN-on-Demand save time and energy by only enabling a Wi-Fi interface when the forthcoming transfer is large enough to justify the transition cost. Once the interface has been enabled, its proactive strategy for keeping the interface active allows transfers that follow soon after to benefit from the enabled Wi-Fi interface — this sometimes amortizes interface transition costs across multiple transfers. These adaptive strategies enable PAN-on-Demand to provide over 5 times faster response than the hierarchical strategy, while at the same time reducing energy usage by 7–26%.

While the WiFi-only strategy produces better response time than PAN-on-Demand in these experiments, the choice of 0.5 for the user-adjustable knob indicates that the user is willing to sacrifice some performance to improve battery lifetime. We confirmed that if the knob is set to maximize performance, PAN-on-Demand produces behavior identical to that of WiFi-only within experimental error.

9.2.2 Photo-sharing workload

For photo sharing, we created a trace to emulate a scenario where a PDA is first used to view 100 6 KB thumbnails of pictures stored on a camera phone, and then later used to view 10 100 KB full-sized images. These images are similar in size to those taken using the camera in the Motorola E680 cell phone. The workload consists of 300 seconds of user activity with 30 seconds of user think time between downloading two full size images.

The results in Figure 5 are similar to those for the e-mail scenario. PAN-on-Demand realizes better battery lifetime than any other strategy, while reducing response time compared to all but the WiFi-only strategy. Compared to the Bluetooth-only strategy, PAN-on-Demand improves response time by over 39%, primarily due to network reorganization. These benefits are explored in more detail in Section 9.3. Bluetooth-DS outperforms PAN-on-Demand in both performance and energy-savings for this workload. It is 19% faster and extends battery life by 5–23%. Without the overhead of network reorganization, PAN-on-Demand is 11% faster than Bluetooth-DS but even after employing sniff mode PAN-on-Demand is still 13% worse than Bluetooth-DS for its impact on client's battery lifetime as the energy cost of proactively spinning up the Wi-Fi interface is not recovered for this workload.

The hierarchical radio strategy performs especially poorly for this workload during thumbnail fetching since it enables and disables the Wi-Fi between each transfer. Thus, its average response time is 4.7 seconds more than that of PAN-on-Demand, while it reduces battery lifetime by 60–72% across the 3 devices.

9.2.3 MP3 workload

For MP3 playing, we consider a scenario in which music files are stored on a high-capacity mobile storage server such as Intel's Personal Server [26]. When the user selects a song from a mobile MP3 client, the file is fetched from the storage server via the PAN and played. When the song finishes, a new song is selected, fetched, and played on the MP3 client. Figure 6 shows results for a workload consisting of 6 songs ranging in duration from 139 to 331 seconds. The size of these files ranges from 2.5 to 5.6 MB.

Surprisingly, for this workload, the Bluetooth-only strategy reduces battery life by almost as much as the WiFi-only strategy. Even though Bluetooth uses less power, its relative throughput is also smaller. Due to transmission delays, the trace takes much longer to finish. Thus, each PAN member sees only meager energy savings by using Bluetooth rather than Wi-Fi. Compared to the two static strategies, PAN-on-Demand performs very well. PAN-on-Demand extends battery lifetime by 39–47% compared to the WiFi-only strategy and by 35-45% compared to the Bluetooth-only strategy. It reduces response time by more than an order of magnitude compared to the Bluetooth-only strategy, while adding only 7 seconds to the (best possible) response time that is achieved with
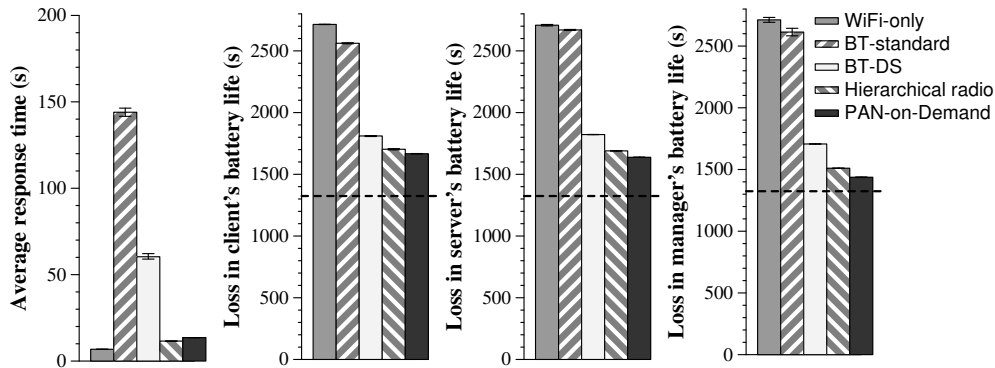
This figure shows results for a workload that emulates a user selecting and playing MP3 files. Each impact on the battery lifetime value shows the mean of three trials and each average response time value shows the mean of nine trials. The error bars are 90% confidence intervals.

Figure 6: Benefit of PAN-on-Demand for MP3 playing

PAN-on-Demand strategies	Response Time (s)	Loss in Battery Life (s)		
		Client	Server	Manager
MP3 playing				
no reorganization	14.5	1663.93	1643.38	1442.92
with reorganization	13.56	1667.35	1637.67	1437.44
after reorganization	12.35	1633.79	1580.82	1384.93
E-mail viewing				
no reorganization	0.74	1107.08	1108.45	1028.08
with reorganization	0.67	1121.46	1086.3	1009.6
after reorganization	0.46	1098.86	1068.49	987.44
Photo sharing				
no reorganization	0.624	463.47	471	412.56
with reorganization	0.452	435.16	422.6	358.9
after reorganization	0.338	423.06	399.77	337.15

Figure 7: Impact of reorganization on PAN workloads

WiFi-only. Bluetooth-DS extends the battery life by 29–35% compared to Bluetooth-standard. With the use of Wi-Fi radios for data transfer, PAN-on-Demand improves the average response time by 78% while extending the battery life by 8–15% over Bluetooth-DS for this workload.

This workload is ideal for the hierarchical radio strategy since it consists only of intermittent transfers, each of which is large enough that enabling the Wi-Fi radio to transfer the data becomes the right choice. Since the static policy embedded in the hierarchical strategy is always right, an adaptive strategy should realize little benefit in comparison. PAN-on-Demand is able to approximate the (correct) behavior of the hierarchical radio strategy by adapting to the offered network load. It incurs a 1–2 second performance penalty at the beginning of each transfer because it tears down Bluetooth connections between transfers to save power. However, connection tear-down allows PAN-on-Demand to increase battery life by 2–5% for the three devices compared to the hierarchical strategy. These savings comes despite the fact that PAN-on-Demand proactively leaves the Wi-Fi interface active for the break-even period (14 seconds) in case another transfer begins shortly after the previous one finishes.

9.3 Impact of reorganization

We studied the impact of network reorganization by comparing the default behavior of PAN-on-Demand with its behavior when re-

organization is disabled. Figure 7 shows results for three scenarios. The rows labeled "no reorganization" show results when reorganization is disabled. The rows labeled "with reorganization" show the results from the previous section where PAN-on-Demand changes its topology during the experiment based on the offered workload. The rows labeled "after reorganization" show the results of running the experiment a second time — in these cases, PAN-on-Demand begins in the correct topology and does not need to reorganize.

The results show that the main benefit of reorganization is improved response time for user requests. When PAN-on-Demand reorganizes during the experiment, it improves response time by 7–38%. The subsequent runs (with the correct initial topology) show even greater benefit; response time decreases 16–46%. Reorganization also improves battery lifetime, though not by as much. Reorganization during the experiment barely affects lifetime in the case of the MP3 workload, but improves lifetime during the photo experiment from 6–15%. On subsequent runs after reorganization, energy-efficiency is improved for all scenarios. The improvement in battery lifetime ranges from 2% in the MP3 workload to up to 18% in the photo workload.

10. LIMITATIONS AND FUTURE WORK

In this work, we do not leverage the presence of infrastructure support, such as base stations, in the mobile environment. Fixed infrastructure can reduce the energy expended by PAN devices since the devices can employ power savings mode currently unavailable in ad-hoc environments. An approach to overcome this limitation would be to allow the PAN manager to monitor wireless traffic to recognize the presence of base stations, it can then trigger a switch to infrastructure mode so that all members can communicate via that base station.

An area of further improvement for PAN communication would be to incorporate the ability to recognize and respond to changing channel conditions. Wireless throughputs are severely affected by contention for the physical medium (both Bluetooth and Wi-Fi operate in 2.4 GHz frequency band), by the physical surroundings due to multi-path fading, and by the interference from sources other than data radios (such as microwaves and cordless phones). PAN-on-Demand would need to adjust its behavior based on observed throughput and transfer delays.

Another limitation of our work is the use of a tunable knob. We use the knob to describe the utility of saving 1 second of battery

lifetime versus the utility of improving performance by 1 second. In our experiments, we found that the relationship between knob values and performance and battery life characteristics is non-linear. Further, only a few discrete ranges of knob settings result in different power management decisions. The impact of knob values are discussed in detail in our previous work [1, 2].

Finally, our PAN model makes an assumption that all devices are altruistic to each other with respect to resource consumption. Hence, our system design cannot be applied directly to peer-to-peer environments that have selfish devices with competing goals for energy-savings.

11. CONCLUSION

PAN-on-Demand has three goals: maximize performance, extend battery lifetime, and minimize user distraction. Our experimental results show that it does a good job of meeting these goals for MP3, e-mail and photo sharing workloads. The strategies PAN-on-Demand uses to achieve these results are:

- **self-organization.** Nodes discover each other without user involvement. Users do not initiate communication on each mobile device.
- **adapting the network topology.** Nodes that are actively communicating tend to be moved to the hub of the network. This improves performance and saves energy by shortening routes.
- **choosing the right route and set of interfaces.** For each data transfer, PAN-on-Demand adapts its communication strategy to match the size of the expected transfer and the current state of network interfaces throughout the PAN.
- **proactive mode transitions.** Interfaces are disabled and connections are dropped to save power during idle periods. Interfaces are re-enabled when many transfers are anticipated in the near future.

PAN-on-Demand is step towards realizing personal environments that are smarter, more responsive and accommodating to the needs of the user.

Acknowledgments

12. REFERENCES

[1] ANAND, M., NIGHTINGALE, E. B., AND FLINN, J. Ghosts in the machine: Interfaces for better power management. In *Proceedings of the 2nd International Conference on Mobile Systems, Applications and Services* (Boston, MA, June 2004), pp. 23–35.

[2] ANAND, M., NIGHTINGALE, E. B., AND FLINN, J. Self-tuning wireless network power management. *Wireless Networks 11*, 4 (July 2005), 451–469.

[3] BAHL, P., ADYA, A., PADHYE, J., AND WOLMAN, A. Reconsidering wireless systems with multiple radios. *Computer Communication Review 34*, 5 (2004), 39–46.

[4] Bluez - official Linux Bluetooth protocol stack. http://www.bluez.org/.

[5] CARTER, C., AND KRAVETS, R. User devices cooperating to support resource aggregation. In *Proceedings of the Fourth IEEE Workshop on Mobile Computing Systems and Applications* (Callicoon, NY, June 2002), pp. 59–69.

[6] CARTER, C., KRAVETS, R., AND TOURRILHES, J. Contact networking: a localized mobility system. In *Proceedings of the 1st International Conference on Mobile Systems, Applications and Services* (San Francisco, CA, May 2003), pp. 145–158.

[7] CERPA, A., AND ESTRIN, D. ASCENT: Adaptive Self-Configuring sEnsor Networks Topologies. *IEEE Transactions on Mobile Computing 3*, 3 (July–September 2004), 272–285.

[8] CHIASSERINI, C.-F., AND RAO, R. R. Coexistence mechanisms for interference mitigation between IEEE 802.11 WLANs and Bluetooth. In *Proceedings of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies* (New York, NY, June 2002).

[9] DOUGLIS, F., KRISHNAN, P., AND BERSHAD, B. Adaptive disk spin-down policies for mobile computers. In *Proceedings of the 2nd USENIX Symposium on Mobile and Location-Independent Computing* (Ann Arbor, MI, April 1995), pp. 121–137.

[10] FORD, B., STRAUSS, J., LESNIEWSKI-LAAS, C., RHEA, S., KAASHOEK, F., AND MORRIS, R. Persistent personal names for globally connected mobile devices. In *Proceedings of the 7th Symposium on Operating Systems Design and Implementation* (Seattle, WA, November 2006), pp. 233–248.

[11] GELLERSEN, H. Smart-Its: Computers for artifacts in the physical world. *Communications of the ACM 48*, 3 (2005), 66.

[12] GOLMIE, N., DYCK, R. E. V., SOLTANIAN, A., TONNERRE, A., AND RÉBALA, O. Interference evaluation of Bluetooth and IEEE 802.11b systems. *Wireless Networks 9*, 3 (2003), 201–211.

[13] GROUP, T. B. S. I. Bluetooth specification, 2001. http://www.bluetooth.com/techn/index.asp.

[14] KRASHINSKY, R., AND BALAKRISHNAN, H. Minimizing energy for wireless web access with bounded slowdown. In *Proceedings of the 8th International Conference on Mobile Computing and Networking* (Atlanta, GA, July 2002), pp. 119–130.

[15] NIKON. *Nikon Coolpix S50c.* http://www.imaging-resource.com/PRODS/CPS50C/CPS50CA.HTM.

[16] NOKIA. *Nokia N810 Internet Tablet.* http://www.nseries.com/products/n810/.

[17] NOKIA. *Nokia N95.* http://www.nseries.com/products/n95/.

[18] PEEK, D., AND FLINN, J. EnsemBlue: Integrating consumer electronics and distributed storage. In *Proceedings of the 7th Symposium on Operating Systems Design and Implementation* (Seattle, WA, November 2006), pp. 219–232.

[19] PERING, T., AGARWAL, Y., GUPTA, R., AND WANT, R. CoolSpots: Reducing the power consumption of wireless mobile devices with multiple radio interfaces. In *Proceedings of the 4th International Conference on Mobile Systems, Applications and Services* (Uppsala, Sweden, June 2006), pp. 220–232.

[20] PERING, T., RAGHUNATHAN, V., AND WANT, R. Exploiting radio hierarchies for power-efficient wireless device discovery and connection setup. In *Proceedings of the 18th International Conference on VLSI Design* (Kolkata, India, January 2005), pp. 774–779.

[21] QADEER, W., ROSING, T. S., ANKCORN, J., KRISHNAN, V., AND MICHELI, G. D. Heterogeneous wireless network management. In *Third International Workshop on Power-Aware Computer Systems* (San Diego, CA, December 2003), pp. 86–100.

[22] SALONIDIS, T., BHAGWAT, P., TASSIULAS, L., AND LAMAIRE, R. O. Distributed topology construction of Bluetooth personal area networks. In *Proceedings of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies* (Anchorage, AL, April 2001), pp. 1577–1586.

[23] SHIH, E., BAHL, P., AND SINCLAIR, M. J. Wake on wireless: An event-driven energy saving strategy for battery operated devices. In *Proceedings of the 8th International Conference on Mobile Computing and Networking* (Atlanta, GA, September 2002), pp. 160–171.

[24] SINGH, S., WOO, M., AND RAGHAVENDRA, C. S. Power-aware routing in mobile ad hoc networks. In *Proceedings of the 4th International Conference on Mobile Computing and Networking* (Dallas, TX, October 1998), pp. 181–190.

[25] SORBER, J., BANERJEE, N., AND CORNER, M. D. Turducken: Hierarchical power management for mobile devices. In *Proceedings of the 3rd International Conference on Mobile Systems, Applications and Services* (Seattle, WA, June 2005), pp. 261–274.

[26] WANT, R., PERING, T., DANNEELS, G., KUMAR, M., SUNDAR, M., AND LIGHT, J. The personal server: Changing the way we think about ubiquitous computing. In *4th International Conference on Ubiquitous Computing* (Goteborg, Sweden, September–October 2002), pp. 194–209.