Efficient Collection of Sensor Data in Remote Fields Using Mobile Collectors

Yuldi Tirta, Zhiyuan Li, *Yung-Hsiang Lu, and *Saurabh Bagchi
Department of Computer Sciences, * School of Electrical and Computer Engineering
Purdue University, West Lafayette, Indiana 47907

Abstract

This paper proposes using a mobile collector, such as an airplane or a vehicle, to collect sensor data from remote fields. We present three different schedules for the collector: Round-Robin, Rate-Based, and Min Movement. The data are not immediately transmitted to the base station after being sensed but buffered at a cluster head; hence, it is important to ensure the latency is within an acceptable range. We compare the latency and the energy expended of the three schedules. We use the *ns-2* network simulator to study the scenarios and illustrate conditions under which Rate-Based outperforms Round-Robin in latency, and vice-versa. The benefit of Min Movement is in minimizing the energy expended.

1 Introduction

Battery-powered sensor networks comprising many sensor nodes [1] allow continuous data collection in hazardous or remote areas, such as a swamp, a desert, or a volcano, for scientific or environmental studies. The sensed data have to be collected, analyzed, and stored in a "base station". When the sensing field (or "field" for simplicity) is too far away from the base station, transmitting the collected data over long distances to the station becomes a major challenge. For example, the sensing field can be a swamp but the data are analyzed in a university located in a city. To transmit the sensed data to the base station, the current main approach deploys intermediate nodes between the field and the station to conduct multi-hop routing. However, if the field is far away from the station, the number of required intermediate nodes may be prohibitively large. Furthermore, some of the intermediate nodes may become communication bottlenecks, and their batteries may drain much faster than the rest of the sensor nodes.

In this paper, we propose an alternate approach to data collection. Data are collected by mobile *data collectors*, such as an unmanned aerial vehicle (UAV) that flies by the field. Using data collectors in a sensor network is analogous to using the postal service. The postmen visit the residents to collect mail so that the residents (equivalent to sensor nodes) do not have to deposit their mail directly at the post offices (equivalent to the base station). The residents need to travel to the road-side mailboxes in front of their houses. Similarly, the data collectors shorten the transmission distance and reduce the energy consumed by the nodes for wireless communication. We extend the analogy further by using the concept of street-corner mailboxes. These mailboxes present a trade-off between the convenience to residents and the efficiency of the postal service. Each resident needs to walk slightly more to drop mail into the mailboxes. However, the postmen do not have to visit each individual house. For a sensor network, we group nodes into units, called *clusters*. Special nodes are assigned for playing the role of these mailboxes; these nodes are called *cluster heads*. The heads are different from other nodes because the heads have large buffer memory to keep sensed data waiting for the collectors. The data collectors visit cluster heads to collect data so that the collectors do not have to obtain data from each and every sensor node.

Using a collector provides many advantages. First, the collector may cross long distances and hazardous terrain to reach the field. Second, the collector can reduce the necessity of multi-hop routing so that the energy of the intermediate nodes can be conserved. Third, only one or a few collectors are needed and they can be refueled more easily than recharging thousands of sensor nodes. A typical node may collect tens of bytes of data every second. Data gathered by the sensor nodes are stored in the buffers of the cluster heads until collected by the mobile collector which can visit the cluster heads periodically eliminating the need for continuous collection by the collectors.

This paper presents three movement schedules for the motion of the data collector. It also provides analysis and simulation to show the effect of the schedule on the latency, the stable store, and the energy expended. The paper focuses on one crucial question in delayed data collection: Is the stable store required in each node bounded? This question is important in the case that the collector is delayed in a collection schedule. This may lead to an unbounded increase in the buffer requirement. Our theoretical analysis shows that our proposed schedules can lead to stable data collection. Detailed simulation results are shown in [10].

2 Related Work

Sensor nodes use wireless networks for communication. A fundamental challenge is the attenuation of wireless signals. The attenuation rates depend on the environment. In order to maintain the same signal-to-noise ratio at the receiver, the transmission power across distance r is proportional to $c_1 r^{c_2} + c_3$ [7]. The value of c_2 is usually between 2 and 4. The required power grows rapidly as the transmission distance r increases so that short-distance transmission is preferred for conserving energy.

The area of reducing energy consumption for data collection or dissemination in sensor networks has been a very active area of research. A sample of the approaches proposed can be found in [2, 3, 4, 5]. Several studies investigate the advantages of using mobile robots to carry sensor nodes. LaMarca et al. [6] suggested using mobile robots to deploy and calibrate sensors, to detect their failures, and to recharge nodes using radio frequency or infrared signals. They built a prototype of sensor networks for house plants with a mobile robot. Sibley et al. [9] built miniature robots with sensors, called Robomotes. These sensors were equipped with wireless communication, odometer, infrared object sensors, and solar cellars. Each robomote is only 47 cm^3. Rybski et al. [8] presented a system for reconnaissance and surveillance using two types of robots: small-sized scouts and rangers that can carry and launch multiple scouts. These examples demonstrate the practicability of combining sensor networks and robots. On the other hand, the issue of efficient data collection in large scale sensor networks using robots has not been fully explored. Specifically our work presents an architecture for data collection eliminating the need for lengthy multi-hop routing by ordinary sensor nodes using the concept of data collectors.

3 Mobility Algorithms for Collector

In this paper, we assume a single collector and analyze three schedules for the collector to visits the cluster heads. The schedules are (a) Round-Robin schedule: the collector visits each cluster head to collect data in a round-robin manner. (b) Data-Rate Based schedule: the frequency of visiting a cluster head is proportional to the aggregate data rate from all the nodes in the cluster. (c) the Min Movement schedule: the collector visits the cluster heads in the proportion of the aggregate data rate, but also with the goal of reducing the distance traversed.

The cluster heads hold sensed data in the buffer before the collector arrives. Both the buffer size requirement and the average data collection latency are affected by the collector's movement schedule. When there is the possibility of rare event detection by the sensor nodes, it is important to keep a low average data latency. If the time to transmit the data from the sensor node to the cluster head is fixed (as it will be for stationary sensor nodes), the latency is determined by the time difference between the data arriving at the cluster head and it being transmitted to the collector. The *average data latency* for a particular cluster head is the sum of the latency of all the bits collected at the cluster head, divided by the number of bits. Similarly, the average data latency for all cluster heads is the sum of the latency of all the bits collected at all cluster heads, divided by the total number of bits. In the rest of this section, we analyze the buffer requirement and the data latency under different collector movement schedules mentioned above.

3.1 Round Robin Schedule

We consider static sensor nodes and a single collector which moves through the network for collecting data from the cluster heads. There are n cluster heads numbered $0,\ldots,n-1$, such that the collector follows the cycle of $0 \to 1 \to \ldots \to n-1 \to 0$. When the collector arrives at a cluster head, it stays there long enough to collect the data accumulated at the cluster head since the last visit of the collector, emptying the cluster head's buffer.

Let α_i be the sensor data accumulation rate at cluster head i, where $i \in [0, n-1]$. Let β_i be the data collection rate of the mobile collector when its visits the cluster head i. We assume that $\beta_i > \alpha_i$; otherwise, the sensed data will eventually be lost. The time to collect data from each head is divided into two parts: the time for travel to the head and the time for transferring the data. Suppose d_i is the time for the collector to travel from i^{th} head to the $(i+1)^{th}$ head (modulo n). We use t_i to represent the time to collect data from the i^{th} head ($t_i > 0$). The value T is the total time for the collector to visit all heads once collecting the data: $T = \sum_{i=0}^{n-1}(d_i + t_i)$. This is also called the round-trip time. When the collector visits the i^{th} head again after T, totally $\alpha_i T$ bits of data have been accumulated at this cluster head. Since the data are transmitted within time t_i, the following inequality must hold

$$\alpha_i T \leq \beta_i t_i \tag{1}$$

We assign r_i as the ratio between the accumulation rate and collection rate: $r_i = \frac{\alpha_i}{\beta_i}$. This inequality can be rewritten as $t_i \geq r_i T$. Since the head has to keep the data for time $T - t_i$ when the collector is away, the head's buffer must be larger than $\alpha_i(T - t_i)$.

We can determine the condition to achieve the minimum round-trip time T. Since $t_i \geq r_i T$, we can obtain the relationship $\sum_{i=0}^{n-1} t_i \geq \sum_{i=0}^{n-1} r_i T$. Let D be the sum of d_i: $D = \sum_{i=0}^{n-1} d_i$. Then, $\sum_{i=0}^{n-1} t_i = T - D \geq \sum_i r_i T$, i.e. $T(1 - \sum_i r_i) \geq D$. This is possible only if $\sum_{i=0}^{n-1} r_i < 1$. Conversely, if $\sum_{i=0}^{n-1} r_i < 1$ is satisfied, we can derive the solution for t_i to minimize T. We define $\tilde{T}$ as the minimum round-trip time: $\tilde{T} \equiv \frac{D}{1 - \sum_{i=0}^{n-1} r_i}$; we also define $\tilde{t}_i$ as $r_i \tilde{T}$. The minimum buffer requirement for cluster i is $\alpha_i(\tilde{T} - \tilde{t}_i)$.

We now analyze the average data latency. First, we consider $t_i = \tilde{t}_i$. Each time the collector revisits the i^{th} cluster head, the oldest bit in the buffer has stayed in the buffer for the amount of time $\tilde{T} - \tilde{t}_i$; this is the longest data latency. Since the collector keeps collecting the data until the buffer becomes empty, the last bit collected has the shortest data latency 0. The data arrival rate and the transmission rate are both constant. Therefore, the average data latency for the i^{th} head equals $(\tilde{T} - \tilde{t}_i)/2$. The number of bits collected from this head during each visit by the collector equals $\alpha_i T$. Therefore, for the entire sensor network, the average data latency equals the total delay divided by the total amount of data:

$$\frac{\sum_{i=0}^{n-1} \alpha_i \tilde{T}(\tilde{T} - \tilde{t}_i)/2}{\sum_{i=0}^{n-1} \alpha_i \tilde{T}} = \frac{\sum_{i=0}^{n-1} \alpha_i(\tilde{T} - \tilde{t}_i)}{2\sum_{i=0}^{n-1} \alpha_i} = \frac{\tilde{T}\sum_{i=0}^{n-1} \alpha_i(1 - r_i)}{2\sum_{i=0}^{n-1} \alpha_i} \tag{2}$$

For $t_i > \tilde{t}_i$, the analysis is divided into two parts dependent on whether the buffer is empty. Suppose the buffer becomes empty at time x. Before it becomes empty, the head has accumulated data for $T - t_i + x$ so totally $\alpha_i(T - t_i + x)$ bits have to be transmitted. Since the transmission rate is β_i, it takes $x = \frac{\alpha_i(T-t_i+x)}{\beta_i}$ to deplete the buffer. Thus, the value of x is $\frac{\alpha_i(T-t_i)}{\beta_i-\alpha_i}$. Up until x, the average latency is $(T-t_i)/2$. For the remaining time, the head transmits sensed data to the collector immediately so the latency is zero. The total number of bits collected from the cluster head i during each round trip equals $\alpha_i T$. Therefore, the average data latency for cluster head i equals

$$\frac{(T-t_i)\beta_i x}{2\alpha_i T} = \frac{(T-t_i)^2\beta_i\alpha_i}{2\alpha_i T(\beta_i-\alpha_i)} = \frac{(T-t_i)^2}{2T(1-r_i)}. \tag{3}$$

For the entire network, the average latency equals

$$\frac{\Sigma_{i=0}^{n-1}\frac{\alpha_i}{1-r_i}\cdot(T-t_i)^2}{(\Sigma_{i=0}^{n-1}\alpha_i)2T} \tag{4}$$

The above formula leads to the following theorem.

Theorem 1: The solution $t_i = \tilde{t}_i$ minimizes the average data latency of the entire sensor network.

Proof: We first fix the value of T and decide the value of t_i that minimizes the average latency. According to formula (4), we need to minimize

$$\sum_{i=0}^{n-1}\frac{\alpha_i}{1-r_i}(T-t_i)^2, \tag{5}$$

Because $T = D + \Sigma_{i=0}^{n-1} t_i$, we can substitute $t_{n-1} = T - D - \Sigma_{i=0}^{n-2} t_i$ into formula (5) to minimize

$$\sum_{i=0}^{n-2}\frac{\alpha_i}{1-r_i}(T-t_i)^2 + \frac{\alpha_{n-1}}{1-r_{n-1}}(D+\Sigma_{i=0}^{n-2}t_i)^2. \tag{6}$$

Taking the partial differential for each t_i, $0 \le i \le n-2$, and equating it to zero, we obtain the following equation

$$\frac{2\alpha_i}{1-r_i}(T-t_i) = \frac{2\alpha_{n-1}}{1-r_{n-1}}(D+\Sigma_{i=0}^{n-2}t_i). \tag{7}$$

This can be rewritten as

$$\frac{\alpha_i}{1-r_i}(T-t_i) = \frac{\alpha_{n-1}}{1-r_{n-1}}(T-t_{n-1}) = K \tag{8}$$

for some K. Since $t_i = T - \frac{K(1-r_i)}{\alpha_i}$, we substitute it into $T = D + \Sigma_{i=0}^{n-1} t_i$ and obtain $K = \frac{D+(n-1)T}{\Sigma_{i=0}^{n-1}\frac{1-r_i}{\alpha_i}}$. Since the second order partial differential of formula (5) over t_i is positive in the entire domain of formula (6), it is easy to verify that $t_i = T - \frac{K(1-r_i)}{\alpha_i}$ is the minimum point. Substituting this minimum point into formula (4), the minimum average latency for a given T equals

$$\frac{((n-1)T+D)^2}{(\Sigma_{i=0}^{n-1}\frac{1-r_i}{\alpha_i})(\Sigma_{i=0}^{n-1}\alpha_i)2T}. \tag{9}$$

To minimize formula (9), we only need to minimize $((n-1)T + D)^2/T$, whose derivative equals $(n-1)^2 - D^2/T^2 > 1$. Since T is bounded from below by $\tilde{T}$, $T = \tilde{T}$ minimizes the average latency. The theorem is proved.

3.2 Rate-Based Schedule

In this schedule, the frequency of visiting a head is proportional to the aggregate data rate from all the nodes in the cluster. We call the period for which the collector stays at a cluster head a *slot*. A slot is defined as the minimum time it takes the collector to drain the data at the cluster head since the last visit of the collector. The consecutive number of slots over which scheduling decisions are made is called a *round*. The data rate based schedule tries to make the time between visits to the same cluster head evenly spaced during each round. Even spacing of the visits makes the latency and buffer requirements comparatively smooth. There exist many ways to make the visits evenly spaced. Our scheme is as follows.

Given m slots in each round and n cluster heads, such that $m_0 + m_1 + ... + m_{n-1} = m$, where m_i is the number of slots assigned to cluster head i according to the data rate. Without the loss of generality, we assume $m_0 \le m_1 \le \ldots \le m_{n-1}$ (we can reorder the heads to satisfy this requirement). Since the $(n-1)^{th}$ cluster has the highest data rate, we assign slots to this cluster first. If the schedule for cluster head $n-1$ is performed later, it can be difficult to make its visits evenly spaced, because there is a good chance that the remaining m_{n-1} slots are not evenly distributed. The m_{n-1} slots for the $(n-1)^{th}$ cluster are determined according to the following recursive formula:

$$s_1 = 1, s_{j+1} = s_j + \frac{m}{m_{n-1}}, \text{ for } 1 \le j < m_{n-1} \tag{10}$$

where the symbol s_i represents the slot number of the i^{th} slot assigned to the particular cluster head. If it is not an exact division, we use $\lfloor\frac{m}{m_{n-1}}\rfloor$ so that no slot is wasted. If at any stage, the computed slot has already been assigned to a cluster head, the next higher slot is assigned. After assigning slots to cluster $n-1$, we then assign the slots to cluster $n-2$ according to the following recursive formula:

$$s_1 = 2, s_{j+1} = s_j + \frac{m}{m_{n-2}}, \text{ for } 1 \le j < m_{n-2}. \tag{11}$$

In the above, if s_{j+1} is already occupied, then we let s_{j+1} be the number that is the closest slot above the computed value. The rest of the time slots are determined by

$$s_{j+1} = s_j + \frac{m}{m_i}, \text{ for } 1 \le j < m_i. \tag{12}$$

Consider four clusters having aggregate data rate in the ratio 1:2:3:4. Using the Rate-Based schedule, in a round of 10 slots, the collector visits the cluster heads in the order 4, 3, 4, 2, 3, 4, 1, 4, 3, 2. In this schedule, cluster head 4 is visited every 2.5 slots apart.

The time between visits to the same cluster head may vary for different visits, because m_i does not always divide m. Hence, the time taken to empty the buffer may also vary from visit to visit. Nonetheless, for the purpose of the analysis in this section, we assume the duration of slots at a particular cluster head are equal.

In each round, the traveled path of the collector can be represented by a cycle consisting of m nodes, each representing a visit to one of the cluster heads. The cycle has exactly m edges, each representing the collector's travel from one cluster head to the next. Let d_j denote the travel time of the collector to make the j^{th} visit in the round. Let $\hat{D}$ denote the sum of $d_j, 1 \leq j \leq m$. Let $\hat{t}_j$ be the data transmission time in the j^{th} time slot. Let T be the time it takes to finish each round, including the collector's travel time and the data transmission time. The average time between two consecutive visits to the same cluster head i equals $\frac{T}{m_i}$. If the cluster head visited in the j^{th} time slot head is i, then we let $\hat{\beta}_j$ equal β_i, $\hat{\alpha}_j$ equal α_i, $\hat{r}_j$ equal r_i, $\hat{t}_j$ equal the time spent collecting data at the i^{th} cluster head, and $\hat{m}_j$ equal m_i. Obviously, $\hat{t}_j\hat{\beta}_j = \hat{\alpha}_j\frac{T}{\hat{m}_j}$.

From $T = \hat{D} + \Sigma_{j=0}^{n-1}\hat{t}_j$ and $\hat{t}_j = \frac{\hat{r}_j}{\hat{m}_j}T$, we have $T - \hat{D} = \Sigma_{j=0}^{n-1}\frac{\hat{r}_j}{\hat{m}_j}T$, which is possible if and only if $\Sigma_{j=0}^{n-1}\frac{\hat{r}_j}{\hat{m}_j} < 1$. When this condition is satisfied, we have $T = \frac{\hat{D}}{(1-\Sigma_{j=0}^{n-1}\frac{\hat{r}_j}{\hat{m}_j})}$. Thus, the buffer size required for cluster head i equals $\frac{\alpha_i}{m_i}T$. Since the collector always move to the next cluster head as soon as it empties the buffer of the current head, the average data latency for cluster head i equals $\frac{1}{2}(\frac{T}{m_i} - \hat{t}_i) = \frac{T}{2m_i}(1-r_i)$, half the the time for the i^{th} cluster to accumulate data until the next visit of the collector. The average latency for the entire sensor network equals

$$\frac{\Sigma_{i=0}^{n-1}(\alpha_i T\frac{T}{2m_i}(1-r_i))}{\Sigma_{i=0}^{n-1}\alpha_i T} = \frac{T}{2\Sigma_{i=0}^{n-1}\alpha_i}\Sigma_{i=0}^{n-1}\frac{\alpha_i(1-r_i)}{m_i} \quad (13)$$

Since $T = \frac{\hat{D}}{1-\Sigma_{j=1}^{m}\frac{\hat{r}_j}{\hat{m}_j}} = \frac{\hat{D}}{1-\Sigma_{i=0}^{n-1}r_i}$, the average data latency can be rewritten as

$$\frac{\hat{D}\Sigma_{i=0}^{n-1}\frac{\alpha_i}{m_i}(1-r_i)}{2(\Sigma_{i=0}^{n-1}\alpha_i)(1-\Sigma_{i=0}^{n-1}r_i)} \quad (14)$$

We compare this average latency to that of Round-Robin schedule. According to formula (2), the latency of Round-Robin equals

$$\frac{D\cdot\Sigma_{i=0}^{n-1}\alpha_i(1-r_i)}{(2\Sigma_{i=0}^{n-1}\alpha_i)(1-\Sigma_{i=0}^{n-1}r_i)} \quad (15)$$

Thus, the Rate-Based schedule has a lower average latency than the Round-Robin schedule if and only if

$$\hat{D}\cdot\Sigma_{i=0}^{n-1}\frac{\alpha_i}{m_i}(1-r_i) < D\cdot\Sigma_{i=0}^{n-1}\alpha_i(1-r_i) \quad (16)$$

Since $\frac{\alpha_i}{m_i}$ equals a certain constant K for all i, the inequality above can be rewritten as $\hat{D}\cdot\Sigma_{i=0}^{n-1}(1-r_i) < D\cdot\Sigma_{i=0}^{n-1}m_i(1-r_i)$. Recall that $\Sigma_{i=0}^{n-1}r_i < 1$. Without loss of generality, we can find ε_1 and ε_2 such that $\varepsilon_1 < r_i < \varepsilon_2$ for all i. The following two theorems are easily derived.

Theorem 2: The Rate-Based schedule has a lower average latency than the Round-Robin schedule if $\frac{\hat{D}}{m} < \frac{D}{n}(1-\varepsilon_2)$.

Theorem 3: The Rate-Based schedule has a lower average latency than the Round-Robin schedule if and only if $\frac{\hat{D}}{m}(1-\varepsilon_1) \leq \frac{D}{n}$.

Notice that β_i is usually much greater than α_i. If $r_i = \frac{\alpha_i}{\beta_i}$ is so small that $1 - r_i$ is almost equal to 1 for all i, then the two theorems above imply that the Rate-Based schedule has a lower average latency than the Round-Robin schedule if and only if $\frac{\hat{D}}{m} \leq \frac{D}{n}$, i.e. if and only if the Rate-Based schedule has a shorter average travel time between two cluster heads visited consecutively than the Round-Robin schedule. This can be illustrated by two opposite examples, assuming r_i to be very small. Suppose there are three clusters with aggregate data rates in the ratio $C_0 : C_1 : C_2 = 3 : 2 : 1$. In the Round-Robin schedule for the three cluster heads, the collector takes the route of $C_0 \rightarrow C_1 \rightarrow C_2 \rightarrow C_0$ with the travel time 4, 4, 1. The route in the Rate-Based schedule is $C_0 \rightarrow C_1 \rightarrow C_0 \rightarrow C_1 \rightarrow C_2 \rightarrow C_0$, with the travel time 4, 4, 4, 4, 1. The Round-Robin schedule has the average travel time 3 per leg, but the Rate-Based schedule has $17/5 = 3.4$ travel time per leg. A "leg" is the distance between two heads which the collector visits consecutively. The Round-Robin schedule wins. If, instead, the route in the Round-Robin schedule takes time 1, 1, 7. Then the average travel time is still 3 per leg for Round-Robin. But the round of the Rate-Based schedule takes time 1, 1, 1, 1, 7. The average travel time is 2.2 per leg. The Rate-Based scheme wins.

3.3 Min Movement Schedule

Intuitively, in the Rate-Based schedule, the spacing between the visits to the same cluster head in each round may increase the amount of movement of the collector. A variation is called the *Min Movement Schedule* which it tries to minimize the movement of the collector. This schedule may also be viewed as a Round-Robin schedule, where the collector spends different amounts of time at a cluster head. The time spent is proportional to the aggregate data rate of the corresponding cluster. The collector collects the data from the cluster head till it empties the buffer. Then the collector sleeps for a while (equal to the time it took to empty the buffer in the earlier step), and resumes the data gathering on wake up till it empties the buffer again, and so on. The number of these sleep-wake up periods is dependent on the data rate.

4 Results

We build a simulation model using the *ns*-2 network simulator and simulate two different scenarios with the three different movement schedules. The scenarios correspond to different topologies with different placements of the base station and the cluster heads. We use three clusters in the simulations. The properties of the cluster heads, the collector, and the schedules are shown in Table 1. Note that the collector parameters are representative ones, and the absolute values will not affect the relative results of the movement schedules. According to the schedule policies outlined in Section 3, the schedules for a round are derived and shown in Table 1.

Scenario 1 is shown in Figure 1. Scenario 2 is identical to scenario 1 except that the positions of cluster heads 1 and 3 are interchanged.

The output parameters from the simulation are the average latency for the sensed data and the time between successive recharges of the collector at the base station. The second param-

Data rate (kb/s)	CH1:CH2:CH3 = 2:1:1
Collector parameters	Speed: 10 m/s; Power consumption: 0.5 J/m; Initial power: 10 KJ

	Round-Robin	Rate-Based	Min Movement
Schedule	1 2 3	1 2 1 3	1 1 2 3

Table 1: Movement Schedules

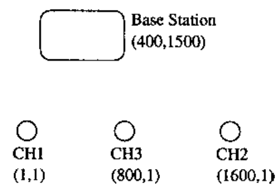

Figure 1: Topology for Scenario 1 (not drawn to scale)

eter is a direct indication of how energy conserving the movement schedule is. The results are shown in Table 2.

The Min Movement schedule is more energy-efficient than the Round-Robin schedule and the evenly spaced Rate-Based Schedule discussed in previous subsection. However, it is easy to see that the average data latency is longer in the Min Movement schedule than in the Round-Robin schedule. This is because the average data latency is shortened if the idle time periods are eliminated. The schedule then becomes a special case of Round-Robin. Similarly, it is easy to see that the buffer requirement is higher in the Min Movement schedule than in the Round-Robin schedule.

The results show that with respect to latency, the Rate-Based schedule and the Round-Robin schedule can outperform each other depending on the scenario, while Min Movement is always inferior to Round-Robin. Intuitively, if the high data rate cluster heads are close together, the amount of movement per leg is smaller in Rate-Based than in Round-Robin. Min movement, on the other hand, has a sleep time for the collector at any cluster head that the collector visits twice or more in a row. This increases the latency. However, this property of staying longer at a cluster head reduces the energy consumption of Min Movement and therefore its time between recharges is the highest among the three schedules. In scenario 1, the latency of the Round-Robin schedule is better than that of the Rate-Based schedule by 16.5%. The average distance traveled in a leg in the Round-Robin schedule is (160+80+80)/3 = 106.7 m, while for the Rate-Based schedule is (160+160+80+80)/4 = 120 m. Since ε, the data collection rate at cluster head/rate of draining data by the collector, is much smaller than 1 (0.026 and 0.052), the theoretical analysis also predicts that Round-Robin is better than Rate-Based in this case. The energy performance can be predicted by the latency result. If the latency is high, it means the collector has a larger time interval between successive visits to a cluster head. Since in all cases the same aggregate amount of data is collected by the collector, this means the collector is spending longer at a cluster head, which implies less frequent movement. Since movement energy consumption is much higher than transmission energy consumption, this leads to a more energy conserving schedule. In scenario 2, the distance per leg is higher in the Round-Robin schedule than in the Rate-Based schedule, and therefore the latency is also higher by 30.2%.

	Round Robin	Rate Based	Min Movement
Scenario 1			
Latency (sec)	196.56	228.93	198.68
Recharge Interval (min)	41.12	41.15	43.64
Scenario 2			
Latency (sec)	199.95	153.61	201.67
Recharge Interval (min)	41.88	41.83	44.35

Table 2: Simulation Results

5 Conclusion

In this paper we have proposed an efficient model for sensor data collection from nodes that are in inaccessible locations. Cluster heads temporarily buffer sensed data and mobile data collectors visit the cluster heads and collect the data which they send to the base station. Three movement schedules are compared with respect to the latency and the energy expended.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A Survey on Sensor Networks. *IEEE Communications Magazine*, 40(8):102–114, August 2002.

[2] M. Bhardwaj, T. Garnett, and A. P. Chandrakasan. Upper Bounds on The Lifetime of Sensor Networks. In *International Conference on Communications*, pages 785–790, 2001.

[3] L. M. Feeney. An Energy Consumption Model for Performance Analysis of Routing Protocols for Mobile Ad Hoc Networks. *Mobile Networks and Applications*, 6(3):239–249, June 2001.

[4] D. Ganesan, R. Govindan, S. Shenker, and D. Estrin. Highly-Resilient, Energy-Efficient Multipath Routing in Wireless Sensor Networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(4):11–25, October 2001.

[5] J. Kulik, W. Heinzelman, and H. Balakrishnan. Negotiation-Based Protocols for Disseminating Information in Wireless Sensor Networks. *Wireless Networks*, 8(2):169–185, March 2002.

[6] A. LaMarca, D. Koizumi, M. Lease, S. Sigurdsson, G. Barriello, W. Brunette, K. Sikorski, and D. Fox. Making Sensor Networks Practical With Robots. Technical report, Intel Research, IRS-TR-02-004, 2002.

[7] J. W. Mark and W. Zhuang. *Wireless Communications and Networking*. Prentice Hall, 2003.

[8] P. E. Rybski, N. P. Papanikolopoulos, S. A. Stoeter, D. G. Krantz, K. B. Yesin, M. Gini, R. Voyles, D. F. Hougen, B. Nelson, and M. D. Erickson. Enlisting Rangers and Scouts for Reconnaissance and Surveillance. *IEEE Robotics and Automation Magazine*, 7(4):14–24, December 2000.

[9] G. T. Sibley, M. H. Rahimi, and G. S. Sukhatme. Robomote: A Tiny Mobile Robot Platform for Large-scale Ad-hoc Sensor Networks. In *International Conference on Robotics and Automation*, pages 1143–1148, 2002.

[10] Yuldi Tirta, Bennett Lau, Nipoon Malhotra, Saurabh Bagchi, Zhiyuan Li, and Yung-Hsiang Lu. Controlled Mobility for Efficient Data Gathering in Sensor Networks with Passively Mobile Nodes. To Appear in *Sensor Network Operations*, Shashi Phoha (ed.), IEEE Press, Wiley Publications, 2004.