REAL TIME DATA-GATHERING IN SENSOR NETWORKS

Yoram Revah, Michael Segal and Liron Yedidsion
Department of Communication Systems Engineering and
Department of Industrial Engineering and Management
Ben-Gurion University of the Negev, Beer-Sheva, ISRAEL

Abstract

Wireless sensor networks represent a new generation of real time traffic communications and high data rate sensor applications, such as structural health monitoring and control. We study some problems related to data gathering in sensor networks when the information that the sensors collect about their environment must be delivered to a collecting central Base Station. We prove that scheduling messages through the network to minimize the maximal delivery time with restrictions on the total idle time allowed is $\mathcal{NP}$-hard. We also refer to a special case of linear network topology for which we present two polynomial time optimization algorithms: One is for minimizing the maximal lateness and maximal delay, while the other is for minimizing the number of tardy messages.

Keywords: Scheduling, Sensor Networks, Half-duplex One-port model.

1 Introduction

Wireless Sensor Networks (WSN) have received increased research attention in recent years. Emerging sensor applications include: monitoring fire and flood disasters, weather forecasting, pollution detection, and military surveillance. A typical application for a sensor network could be a deployment of sensors to observe an area and report any intruders entering the protected zone. Sensor networks are very *data-centric*, meaning that the information that they collect about their environment must be delivered to the central processing system referred to as a Base Station (BS) (or the root node in the network graph). This BS has greater computational, storage, and transmission capabilities than the rest of the nodes in the network. The BS typically serves as an entry point to the sensor network, integrating the sensor network with a wired network. In each round of data gathering, all the data from all sensors has to be collected and transmitted towards the BS, where the end-user can access the data. In some sensor network applications, data collection may be needed only from a certain region and therefore, only a subset of sensors will be used. A simple approach to accomplishing this data gathering task is for each node to transmit its data directly to the BS. Since the BS is typically located far away, the energy consumed by transmitting messages from a certain sensor to the BS might be quite high and collision may occur between messages transmitted simultaneously. Therefore, an improved approach is to use multi-hop transmissions, trying to find the shortest multi-hop path for the message to reach the BS. This approach requires scheduling the messages through the network so as to minimize the total energy consumption, minimize the message flow time in the network (the time it takes for a message to reach the BS) and avoid collisions. Some research uses a message routing algorithm in a WSN in order to minimize different scheduling criteria assuming various sensor network models. In our research we analyze three different scheduling criteria: the first is maximal completion time, the second is maximal lateness, and the last is the number of tardy messages.

The maximal completion time criterion (the time it takes the last message to arrive at BS), also referred to as the *makespan* criterion, was studied by Revah and Segal [15, 16], Bermond et al. [3], and Gargano and Rescigno [8]. Revah and Segal [15, 16] Consider linear, two-branch, and star (or multi-branch) network topologies. For each topology they provide a polynomial algorithm to schedule all the messages to the BS, minimizing both the maximal completion time and the average packet delivery time. They present algorithms to minimize the maximal and average completion time for ring and tree network topologies and provide an approximation algorithm with a 1.5 approximation ratio to minimize the maximal completion time for a grid network topology. Bermond et al. [3] transform a network into an undirected graph $G(V, E)$ with V nodes and E edges and model the transmission area and the interference area as balls in the graph by introducing two parameters: d_T, the transmission radius and d_I the interference radius with $d_I \geq d_T$. They [3] deal with gathering information in linear or grid network topologies so as to minimize the maximal completion time. They show that in a general network this problem is $\mathcal{NP}$-hard. Gargano and Rescigno [8] referred to the maximal completion time minimization problem using directional antennas on a general sensor network topology. They considered a special case where all nodes have a single message to transmit to the BS and provided an optimization algorithm for the problem. Gargano [7] reviewed some sensor network gathering problems where the emphasis is on some algorithmic and graph theoretical problems that arise.

The minimization of the maximal lateness criterion was studied by Richardson and Sieh [14] and Banka, and Jayasumana [2]. The *lateness* of each message is defined as the difference between that message's due-date and its completion time. In real time sensor applications, the data sensor readings reflect the current state of the environment. Since the nature of the environment is almost constantly changing, the sensor date readings have a temporal time interval in which they are relevant. For example, a date collected by a temperature sensor could be irrelevant after a certain time. Since distributed micro-sensing involves direct interaction with a physical environment, data communication in sensor networks often has timing constrains. In other words, any input to the system has a validity interval associated with it, a due-date, and the input must reach the destination ahead of its due-date. Minimizing the maximal lateness is a fairness measure that assures that no input is left behind and that the total picture at the BS is as comprehensive and relevant as possible. This measure aims to prevent data being late within the sensor networks due to network delays or data being dropped, e.g., due to network congestion or wireless link errors in sensor networks (see [4], [17], and [9]). If data is dropped in the network, the prior sample of data available might be used until new input arrives. In many applications, data sensed with significant lateness does not reflect the current state of the environment and may either falsely activate control measures on the remote environment or not activate control measures when needed. Obviously, this affects the system's integrity. Richardson and Sieh [14] present a fault-tolerant LAN architecture for Mobile Mission Critical System (MMCS's). MMCS's are real-time computing platforms that perform continuously in harsh environments. The architecture combines a priority driven, real time, LAN protocol with the adaptive earliest deadline first (AEDF) scheduling approach. Banka, and Jayasumana [2] show that the age of data used by the end application can impact the accuracy of end results, and may produce detrimental consequences for many real time sensing applications. This work uses a tardiness measure for quantitatively capturing the lateness of data due to network dynamics, and presents an analytical model relying on network delay, network packet loss rate and sampling rate to tardiness.

Finally, the minimization of the number of *tardy* messages criterion was studied by Lu et al. [12] and Li and Ramamritham [11]. This criterion is relevant in WSN where each message has its own due-date and any message reaching the BS later than its due-date becomes obsolete and may be discarded. For this kind of WSN it is very important to plan ahead of time and decide which messages to discard and which to transmit to the BS so that no unnecessary energy is wasted. Lu et al. [12] presents the RAP algorithm for real-time communication architecture for large scale sensor network. RAP provides convenient high level query and event services for distributed micro-sensing applications. They confirmed, using simulations, that RAP significantly reduces the end-to-end deadline miss ratio in sensor network. Li and Ramamritham [11] analyze the problem of providing timeliness guarantees for multi-hop message transmissions in robots equipped with sensors that collaborate with one another to achieve a common goal. Their technique schedules messages by carefully exploiting spatial reuse for transmission to avoid collisions, so that deadline misses are minimized.

We chose to focus our analysis on systems equipped with directional antennas based on the results by Florens and McEliece [5] who show that systems with directional antennas outperform systems with omnidirectional antennas by 50% on Linear Networks, where a linear network is a network topology where all the sensors are located on a single line and the BS is located at one end of that line. The idea of using directional antennas in wireless communication is not new. It has already been extensively used in cellular networks for frequency reuse, to reduce interference and to increase the capacity of allowable users within a cell. However, the applications of directional antennas to wireless ad hoc or sensor network to reduce the transmit power of each node to achieve power-efficient scheduling is relatively new (See [10, 13]).

1.1 Network and Problem Definition

A wireless sensor network is modeled as a graph $G(V, E)$ with $n + 1$ nodes $\{v_0, v_1, ..., v_n\}$, where each node v_i is a sensor that can transmit and receive data. There is an edge (v_i, v_j) if and only if v_j can receive v_i's transmissions when v_i points its directional transmission antenna towards v_j.

There are $\mu \leq n$ messages scattered around the WSN, $M = \{1, ..., \mu\}$, and $j = [i]$ indicates that message i is located at node v_j. Several assumptions characterize our model:

- At time 0, each node v_i has at most one message to transmit to the destination. The network has a special node v_0, referred to as the BS, which is the destination of all messages.
- Every node in the network including the BS has the same transmission range r.
- We assume a half-duplex one-port model, i.e., messages between any two antennas can be transferred in both directions, but only one direction at a time (not simultaneously). and any antenna can either send or receive a message at a given time step.
- Unless denoted otherwise, the capacity of each node's buffer is one message. (In some cases we assume an arbitrary buffer capacity.)
- All the information about the input and topology of the network is available at the BS and there are separate, collision-free, control channels between the BS and the other nodes.
- The network is built of directional antennas, i.e., the signal from node v_i to node v_j propagates in a straight line in the direction of node v_j without dispersing to other directions.
- The antennas are directed only toward the BS and therefore messages can not backtrack or circle around the graph.
- Time is slotted and one hop transmission consumes one time slot. A node can either transmit or receive in one time slot. This model of a channel is usually referred to in the literature as the S-TDMA channel model.

Based on the above assumptions, a transmission from node v_i that is transmitted to node v_j, for $i, j = 1, ..., n$, arrives successfully if for all simultaneous transmissions from v_k,$k \neq i$,$k = 1, ..., n$ using directional antennas pointed in the direction of v_j the following relations hold: $|v_k - v_j| \leq r$, where r is transmission range and $|v_k - v_j|$ is the distance between nodes k and j. A transmission that violates this condition is defined as a collision.

A schedule defines specifically when and towards which direction node v_j should transmit for $j = 1, ..., n$. We denote by d_{ij} the minimal distance, measured in number of hops, between node v_i and node v_j. The minimal arrival time of message i from node $v_{[i]}$ to the BS is denoted by $t_i = d_{[i],0}$.

This paper is organized as follows: In Section 2 we prove that the general scheduling problem of minimizing the Makespan with constraints on the total idle time is $\mathcal{NP}$-hard. In Section 3 we consider the problem of minimizing the maximal lateness assuming Linear Network topology, where all messages are given a feasible time period in which they are expected reach the BS. The problem of maximizing the number of messages that can reach the BS where each message has its own due-date, assuming Linear Network topology, is discussed in section 4. Finally, we conclude at Section 5.

2 Minimizing the Makespan Under Constraint on the Total Idle Time is $\mathcal{NP}$-hard

We aim to find a schedule for every possible input of messages to nodes that minimizes the maximal completion time. We assume that the capacity of each node's buffer is one message. The time it takes for a *message* i to reach the BS (completion time) is denoted by C_i for $i = 1, ..., \mu$. Notice, that $C_i \geq t_i$. The maximal completion time amongst all messages, referred to as the *Makespan*, is denoted by $C_{\max}$ where $C_{\max} = \max\limits_{i=1,...,\mu} C_i$. The *idle time* , idl_i of message i, is the total sum of unit idle times that i suffers starting at t_0 until reaching the BS, where a unit idle time is defined as any time unit during which a message remains at a node without being transmitted. We denote by S the constraint on the total idle times of all messages, i.e., the schedule must obey $\sum_{i=1}^{\mu} idl_i \leq S$. For simplicity, we hereafter refer to the problem of minimizing the makespan under constraint on the total idle time as P1.

We show that P1 for arbitrary graphs is $\mathcal{NP}$-hard by reducing the $\mathcal{NP}$-complete 3*-CNF-SAT problem to P1. The 3*-CNF-SAT is defined as follows: Given a logical expression, ϕ, in a CNF form where each predicate contains exactly 3 variables (literal or its negation) and each variable appears exactly 3 times throughout the expression, is there a feasible assignment to the variables satisfies the expression? This problem is known as $\mathcal{NP}$-hard, see [6].

We proceed as follows. Given a logical expression ϕ in 3*-CNF-SAT form with μ literals and predicates, we aim to construct a certain graph topology with a BS such that ϕ is satisfied if and only if $C_{\max} \leq 4\mu + 2$, where μ stands for the number of variables in ϕ and the total sum of idle times, $S = 0$. The reduction from 3*-CNF-SAT to P1 is as follows: For every variable in ϕ we construct the gadget depicted in Figure 1:

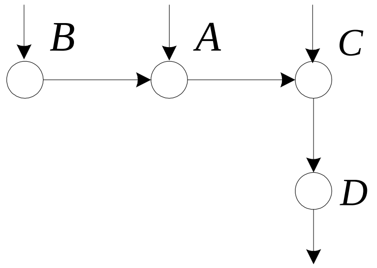

Figure 1: The gadget.

Assume that each *node A, B or C* may contain a message. All the messages will be delivered to D with no idle time if and only if either only node A contains a message or no message is present at A at a given time slot. We associate each variable in ϕ with entries A, B, C of a corresponding gadget such that the literal appearing once is associated with entry A, and the other two identical literals are associated with entries B and C. Notice, that if all the literals appear either as a variable or its negation, we can simply replace these literals by TRUE value in ϕ. We use the box in Figure 2 to represent the gadget.

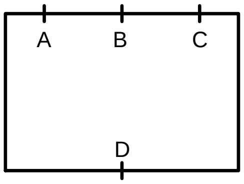

Figure 2: Box describing gadget.

Overall, we have μ gadgets. Next, for every predicate, say $(x \vee \bar{y} \vee z)$, we construct the tree that is shown in Figure 3:

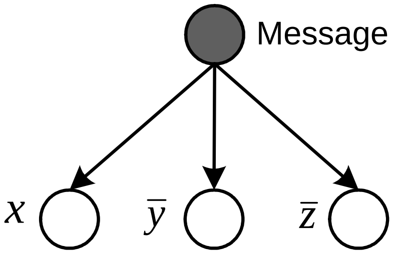

Figure 3: Tree associated with the predicate $(x \vee \bar{y} \vee z)$. The black node contains a message.

We connect the leaves in trees corresponding to the literals to the corresponding entries in associated gadgets (see example in Figure 4). We add additional nodes after entry D in order to overcome a possible collision between messages that are sent to the BS. The maximal time it takes for a message to propagate from entry to exit in any gadget is at most 4 time slots, therefore the i^{th} gadget is added $4(i-1)$ nodes for $i = 1, ..., \mu$.

Claim 1 *ϕ is satisfied if and only if $C_{\max} \leq 4\mu + 2$ and $S = 0$.*

Proof. Suppose that there is an assignment of variables such that ϕ is satisfied. It means that every predicate, $h \in \phi$, has a variable assigned TRUE value. We can route the message of the tree corresponding to h via the node associated with one of the literals that is assigned a TRUE value arbitrarily. This schedule guarantees that no gadget will have a message simultaneously at entry A and at entries B and/or C, since A corresponds to literal that is the negation of entries B and C's associated literal. The maximum time it takes for a message from some tree to pass a gadget (and get to node D) is at most 5 time slots. Since the i^{th} gadget entry D is connected to $4(i-1)$ additional nodes then

$$C_{\max} \leq 5 + 4(\mu - 1) + 1 = 4\mu + 2.$$

Clearly, no idle times are possible during this process unless messages are scheduled simultaneously through entry A and entries B or C, or in other words simultaneously using literal and its negation.

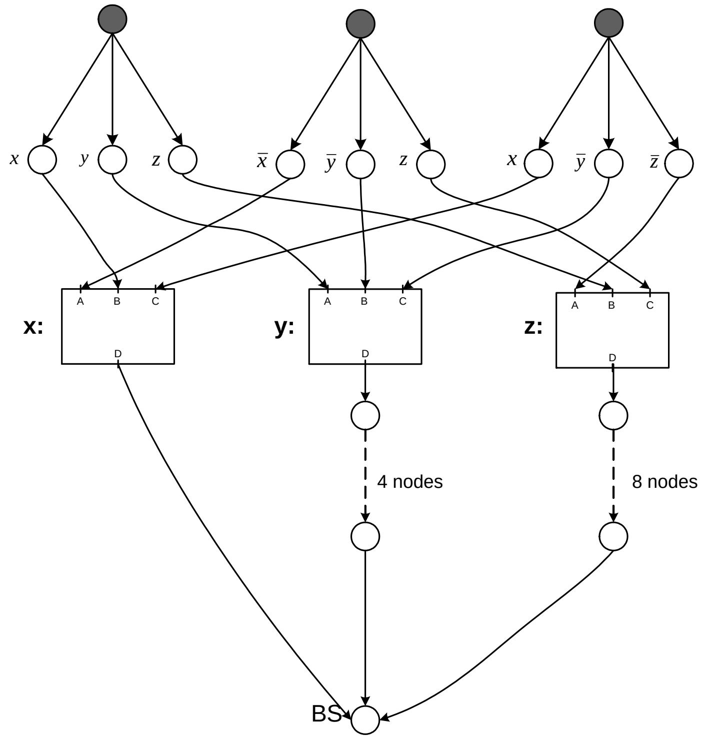

Figure 4: The graph representing ϕ:$(x \vee y \vee z) \wedge (\bar{x} \vee \bar{y} \vee z) \wedge (x \vee \bar{y} \vee \bar{z})$.

Conversely, if we have a schedule for the given instance with $C_{\max} \leq 4\mu + 2$ and $S = 0$, then obviously for every gadget messages enter the gadget either at entry A or at entries either B or C, but not both. Hence, assigning values to each variable according to the entries that were used to transfer messages in the gadget associated with it, will ensure a feasible assignment. Since all messages reach the BS, we conclude that there is at least one literal in each predicate that is assigned a TRUE value. Thus there is an assignment of variables such that ϕ is satisfied. That completes our proof ■

3 Minimizing the Maximal Lateness ($L_{\max}$) on a Linear Network

In this section we aim to minimize the maximal lateness amongst all messages, that is, we are looking for a minimum of

$$L_{\max} = \max_{i=1,\ldots,\mu} L_i$$

where

$$L_i = C_i - dd_i \quad \text{for } i = 1, ..., \mu. \tag{1}$$

In this section we assume a WSN that allows each message an identical time window of size slk in which the message can arrive to the BS and not be considered late. That is

$$dd_i = t_i + slk \quad \text{for } i = 1, ..., \mu. \tag{2}$$

This due-date assignment method is referred to as the slack due-date assignment method and was introduced by Adamopoulos and Pappis [1] for a single-machine scheduling system. We adopt their due-date assignment method for a WSN and assume a Linear Network model [15] where the capacity of each sensor's buffer is arbitrary and, thus, messages may bypass one another. In this model each sensor is considered to be a node in the graph *G(V,E)*, the BS is always at the right end of the network and node v_j is the j^{th} node from the BS (without loss of generality). In fact, in this topology each node receives transmissions only from its left-side neighbor and thus node v_j is at a distance of exactly j hops from the BS.

Our goal is to find an optimal algorithm that schedules all the messages to the BS and minimizes $L_{\max}$. In the following we prove that applying the *Linear Network Algorithm* [16] to the Linear Network with legal input will minimize $L_{\max}$. This algorithm is as follows

Linear Algorithm Network (LNA): At any given time slot, if v_j carries a message and v_{j-1} has no message, then transmit the message from v_j to v_{j-1} for $j = 1, ..., n$.

Theorem 1 *LNA minimizes $L_{\max}$ for the slack due-date assignment method.*

Proof. To prove Theorem 1 we use a simple interchange proof method as follows: We assume two schedules, π_j and π_i, for a given input. The schedules are similar except for two adjacent messages, k and l. In both schedules these messages are scheduled to arrive at the BS one immediately after the other, such that in schedule π_j, message k is scheduled to arrive at the BS before message l and in schedule π_i message l is scheduled to arrive at the BS before message k. Without loss of generality we assume that $t_k < t_l$. Therefore, from eq. (2) we get $dd_k < dd_l$. According to π_j message k arrives at the BS before message l, so $C_k(\pi_j) < C_l(\pi_j)$ and thus

$$\max\left(L_k(\pi_j), L_l(\pi_j)\right) = \max\left(C_k(\pi_j) - dd_k, C_l(\pi_j) - dd_l\right). \tag{3}$$

In π_i message k arrives after message l so $C_k(\pi_i) > C_l(\pi_i)$ and since $dd_k < dd_l$ we get:

$$\max\left(L_k(\pi_i), L_l(\pi_i)\right) = \max\left(C_k(\pi_i) - dd_k, C_l(\pi_i) - dd_l\right) = C_k(\pi_i) - dd_{k,}.$$

We denote the first available time slot after the completion of all the messages that precede messages l and k in which a message can reach the BS by T (note that T is identical for π_j and π_i). Trying to minimize the value of $\max(L_k, L_l)$ we would like to schedule both messages l and k as early as possible. Therefore in π_j we get

$$C_k(\pi_j) = \max(T, t_k) \tag{4}$$

and according to the network limitation the BS can receive a message once every two time slots, hence

$$C_l(\pi_j) = \max(C_k(\pi_j) + 2, t_l) = \max(T + 2, t_k + 2, t_l). \tag{5}$$

Similarly in π_i we get

$$C_l(\pi_i) = \max(T, t_l) \tag{6}$$

and

$$C_k(\pi_i) = \max(C_l(\pi_i) + 2, t_k) == \max(T + 2, t_l + 2, t_k) = \max(T + 2, t_l + 2). \tag{7}$$

Since $t_k < t_l$ then substituting for eqs. (4) and (7) into eq. (1) we get

$$L_k(\pi_i) = \max(T + 2, t_l + 2) - t_k - slk > \max(T, t_k) - t_k - slk = L_k(\pi_j),$$

and substituting for eqs. (5) and (7) into eq. (1) we get

$$L_k(\pi_i) = \max(T + 2, t_l + 2) - t_k - slk > \max(T + 2, t_k + 2, t_l) - t_l - slk = L_l(\pi_j),$$

and, thus, $\max(L_k(\pi_j), L_l(\pi_j)) < \max(L_k(\pi_i), L_l(\pi_i))$. The lateness of any message h that is scheduled after messages k and l depends on the completion of messages k in schedule π_i and l in schedule π_j. Since $C_k(\pi_i) \geq C_l(\pi_j)$ we have $L_h(\pi_i) \geq L_h(\pi_j)$ and, therefore $L_{\max}(\pi_i) \geq L_{\max}(\pi_j)$. Obviously, it is preferable to schedule any couple of messages ordered by their distance from the BS in ascending order. By minimizing the maximal lateness for each pair of messages we ensure the minimization of $L_{\max}$. Therefore, scheduling all messages ordered by their distance from the BS in ascending order using LNA will minimize $L_{\max}$. ■

Another problem related to the minimization of $L_{\max}$ problem is the minimization of the maximal delay problem. We define the delay of a message i, Δ_i, as the difference between the distance of message i from the BS and the actual time of delivery, that is $\Delta_i = C_i - t_i$ for $i = 1, \ldots, \mu$. We denote by $\Delta_{\max}$ the maximal delay time amongst all messages where

$$\Delta_{\max} = \max_{i=1,\ldots,\mu} \Delta_i. \tag{8}$$

Obviously the minimization of $\Delta_{\max}$ problem is a special case of the minimization of $L\max$ problem with $slk = 0$ and thus the following corollary holds:

Corollary 1 *LNA minimizes* $\Delta_{\max}$.

4 Minimizing the Number of Tardy Messages (N_T) in a Linear Network

In this section we deal with the following scheduling problem. Given an input M of messages in a Linear Network where each message has its own due-date and any message arriving late is obsolete and can be discarded. We would like to find a schedule that minimizes the number of messages not reaching the BS before their due-date expires , i.e., our goal is to minimize:

$$N_T = \sum_{i=1}^{\mu} Z_i \tag{9}$$

where, $Z_i = \begin{cases} 0 & \text{if } C_i \leq dd_i \\ 1 & \text{if } C_i \leq dd_i \end{cases}$. We assume that the capacity of each sensor's buffer is enough to keep one message. In the following we present a minimization algorithm for N_T.

```
Algorithm min$N_T$

  1. Set $Q = M$.
  2. Use LNA on set $Q$ and calculate $C_i$ and $Z_i$ for $i \in Q$.
  3. If $\sum_{i \in Q} Z_i \geq 0$ then find the message $i \in$ closest to BS for which $Z_i = 1$, set $Q = Q \backslash i$
     and go back to 2. Otherwise, schedule group $Q$ according to LNA and set $M = M \backslash Q$.
  4. $N_T = |M| \setminus |Q|$.
```

Theorem 2 *Algorithm minN_T minimizes the number of tardy messages.*

Proof. Without loss of generality we assume that the messages are numbered in ascending order where message 1 is the closest to the BS. Since each antenna has a one message buffer, the messages are queued according to their distance from the BS and bypasses are impossible. In that case it is clear that pushing the queue as fast as possible towards the BS, as done in LNA, will minimize the C_i for $i = 1, ..., \mu$ if no messages are being discarded. Hence we can conclude that for any given input without discarding any messages, the LNA provides an optimal schedule for minimizing N_T. Therefore the problem is reduced to simply discarding the minimal number of messages. If, by using LNA, we get $\sum_{i \in M} Z_i$ =0, then the schedule is obviously optimal; otherwise we have to discard at least one message.

Before we continue, we would like to make use of few definitions from [15]. Message i is said to be *dependent* on message j if in a schedule that minimizes C_i for $i = 1, ..., \mu$, i is idle because we need to transmit message j. Similarly, we can define a set of independent groups.

Definition 1 *Two groups, m_i and m_j, of maximal length are said to be independent if the messages belonging to group m_i can be transmitted at maximum rate without being delayed by any message of group m_j and vise versa.*

Maximal length means that we cannot increase the size of either group keeping them independent. Revah and Segal [15] showed the construction of group u_i in interval $|I|$. They introduced δ_i values as follows:

$$\delta_i = m - h - 1 \tag{10}$$

where h is the number of nodes without an input message in interval $|I|$ (depicted by spaces between messages in figure 5), and m is the number of nodes with an input message in a interval $|I| = m+h$.

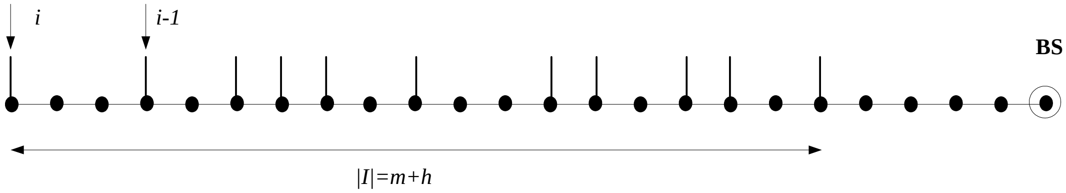

Figure 5: Messages dependency

The following recurrence relation between δ_{i-1} and δ_i for dependent messages $i-1$ and i holds:

$$\delta_i = \delta_{i-1} + 1 - \Gamma_i$$

where Γ_i is the number of nodes without an input message that lie between nodes $[i]$ and $[i-1]$. Since the total delay time Δ_i of message i is a non-negative value we get:

$$\Delta_i = \max(0, \delta_i) = \max(0, \Delta_{i-1} + 1 - \Gamma_i) \tag{11}$$

where $\Delta_i > 0$ indicates that message i is dependent on message $i-1$. Since $C_i = t_i + \Delta_i$, minimizing Δ_i will help to minimize Z_i. As we have shown above the scheduling of all messages that have not been discarded is done using LNA. Therefore, minimizing N_T for each independent group will minimize N_T for the entire network. Hence, it is enough to analyze only one maximal length dependent group m_i.

We denote the subgroup of messages in m_i that have not been discarded during the algorithm by Q_i. Without loss of generality we can renumber the messages in group Q_i in ascending order starting at 1 according to their distance from the BS. Since bypass is impossible, the completion time of message i is influenced only by the messages positioned ahead of it. Suppose that by using LNA we get a schedule where message k is the first message with $Z_k = 1$. Thus, we would have to discard at least one message of the subgroup $\{1, 2, ..., k\}$, otherwise we will still have at least one message in group Q_i that will not arrive on time.

We will now show that out of the subgroup $\{1, 2, ..., k\}$, discarding message k is an optimal decision. From eq. (10) we can see that discarding any message i will decrease the value of δ_j for any message $j > i$, by no more than 2 (it will increase h by 1 and decrease m by 1). Therefore, any discarded message i will decrease the values of Δ_j for any message $j > i$ by no more than 2 as well. In fact, Δ_j will decrease by only 1 for messages that have $\Delta_j = 1$ (prior to the removal of message i). Moreover, according to eq. (11), if Δ_j decreases by only 1 then the Δ_l values for any message $l \in \{j, j+1, ..., |Q_i|\}$ will also decrease by no more than 1. Hence, in order to decrease Δ_l by as much as possible (and thus increase the chance of message l not being late) we would like to discard a message i for which there is no other message j such that $\Delta_j = 1$ (prior to the removal of message i) where $i < j < l$. In general, choosing to discard a message i with higher index, decreases the possibility of having a message j for which $\Delta_j = 1$ where $i < j < l$. Since k is the highest index in the group $\{1, 2, ..., k\}$, discarding message k is optimal. Moreover, if $C_k > dd_k + 2$, discarding message $i < k$ will leave us with $Z_i = 1$ and $Z_k = 1$ whereas discarding message k will leave us with just $\sum_{i=1}^{k} Z_i = 1$. Therefore, in this case discarding message k is the *only* optimal decision. ■

Algorithm $\min N_T$ runs in $O\left(n^3\right)$ time, since it discards at most $O\left(n\right)$ messages and for every iteration it uses LNA which runs in $O\left(n^2\right)$ time.

A special case of the N_T minimization problem with reduced computational time is where all messages have the same due-date.

Theorem 3 *For the special case where all messages have the same due-date, i.e., $dd_i = dd$ for $i = 1, ..., \mu$, running LNA for dd time slots minimizes N_T.*

Proof. Since LNA minimizes C_i for $i = 1, ..., \mu$ for all messages and since all the messages have to arrive at the BS no later than dd time slots, minimizing the completion time for all messages maximizes the number of messages arriving at the BS before time dd. ■

Using LNA for only dd periods of time we reduce the computational time for this special case to $O\left(\min\left(dd^2, n^2\right)\right)$ time. For every dependent group m_i with μ_i messages, Revah and Segal [15] have shown that using LNA, after the arrival of the first message in group m_i all other messages in that group arrive in a periodic sequence time of 2 time slots apart. Therefore the maximal number of messages arriving at the BS before their due-date expires (dd time slots) for group m_i is: $\min\left(\mu_i, \left\lceil \frac{dd - \min\limits_{j \in m_i}(t_j)}{2} \right\rceil\right)$. Given that the input consists of independent groups $\{m_i\}_{i=1}^{q}$ we conclude that the number of messages arriving at the BS before their due-date expires for the entire network is: $\sum_{i=1}^{q} \min\left(\mu_i, \left\lceil \frac{dd - \min\limits_{j \in m_i}(t_j)}{2} \right\rceil\right)$.

5 Summary

This paper analyzes data gathering problems in sensor networks. First we have shown that the problem of minimizing the maximal completion time with a constraint on the total idle time is $\mathcal{NP}$-hard under a half duplex one port model equipped with directional antennas. Next, we referred to two linear network topology related problems: One is the problem of minimizing the maximal lateness assuming the slack due-date assignment method, for which we presented an optimal $O\left(n^2\right)$ time algorithm. The other is the problem of minimizing the number of tardy messages where we presented an optimal $O\left(n^3\right)$ time algorithm. For that problem we also analyzed a special case where all messages have a common due-date and gave a reduced computational time algorithm.

References

[1] Adamopoulos, G.I., and Pappis, C.P., 1996, Single Machine Scheduling with Flow Allowances, *Journal of the Operational Research Society*, **47**, 1280-1285.

[2] Banka T., and Jayasumana A. P., 2007, Impact of Network Dynamics on Tardiness Of Data in Sensor Networks, *Proc. IEEE/ACM 2nd. Intl. Conf. on Communication System Software and Middleware, Bangalore, India.*

[3] Bermond J. C., Galtier J., Klasing R., N. Morales and S. Perennes, 2006, Hardness and Approximation of Gathering in Static Radio Networks,*Parallel Processing Letters*, 16(**2**):165-183.

[4] Chiasserini C. and Garetto M., 2004, Modeling the Performance of Wireless Sensor Networks, *Proc. IEEE INFOCOM.*

[5] Florens C., and McEliece R., 2004, Lower bounds on data collection time in sensory networks, *IEEE Journal on Selected Areas in Communications*, 22(6), pp. 1110-1120.

[6] Garey M. and Johnson D., 1979, Computers and Intractability: A Guide to the Theory of NP-Completeness. *W.H. Freeman and Company, New York, NY*

[7] Gargano L., 2007, Time Optimal Gathering in Sensor Networks, *SIROCCO , LNCS 4474*, pp. 7-10.

[8] Gargano L., and Rescigno A. A. 2006,, Optimally Fast Data Gathering in Sensor Networks, *MFCS LNCS 4162*, pp. 399-411.

[9] Hull B., Jamieson K. and Balakrishnan H., 2004, Mitigating Congestion in Wireless Sensor Networks, *Proc. ACM SenSys, Baltimore, MD*, pp.134–147.

[10] Ko Y. B ., Shankarkumar V., and Vaidya N.H., 2000, Medium access control protocols using directional antennas in ad hoc networks, *Proc. IEEE conference on Computer Communications(INFOCOM), Tel Aviv, Isreal,* vol. 1(**3**), pages 13-21.

[11] Li H. and Ramamritham K., 2005, Scheduling Messages with Deadlines in Multi-hop Real-time Sensor Networks, *Proc. IEEE international Real Time Systems Symposium.*

[12] Lu C., Blum B. M., Abdelzarher T. F., Stankovitc J., and He T., RAP, 2002, A Real Time Communication Architecture for Large Scale Wireless Sensor Networks, In Real Time Technology and Applications, *Symposium, San Jose, CA.*

[13] Nasipuri A., Mandava J., Manachala H., and Hiromoto R. E., On demand routing using directional antennas in mobile ad hoc networks, Proc. IEEE Computer Communications and Networks, 2000.

[14] Richardson P., and Sieh L, 2000, A Fault Tolerant Local Area Network Architecture for Mobile Mission Critical System, *Proc.Ninth international conference on Computer Communications and Networks, Las Vegas* pp.170-175.

[15] Revah Y. and Segal M., 2007, Improved Algorithm for Data-Gathering Time in Sensor Networks II: Rings, Tree and Grid Topologies, *Proc. IEEE, The Third International Conference on Networking and Services, ICNS Athens, Greece.*

[16] Revah Y. and Segal M., 2007, Improved Algorithm for Data-Gathering Time in Sensor Networks, IEEE, *Proc. IEEE, The Third International Conference on Networking and Services, ICNS Athens, Greece.*

[17] Zhao J., and Govindan R., 2003, Understanding Packet Delivery Performance in Dance Wireless Network Sensor Networks, *Proc.of Intel. Conf. on embedded Networked Sensor systems, Los Angeles.*