Bandwidth Allocation in Hexagonal Wireless Sensor Networks for Real-Time Communications

K. Shashi Prabh
Real-Time Systems Research Center (CISTER, ISEP)
Polytechnic Institute of Porto, Portugal

Abstract

We present an algorithm for bandwidth allocation for delay-sensitive traffic in multi-hop wireless sensor networks. Our solution considers both periodic as well as aperiodic real-time traffic in an unified manner. We also present a distributed MAC protocol that conforms to the bandwidth allocation and thus satisfies the latency requirements of real-time traffic. Additionally, the protocol provides best-effort service to non real-time traffic. We derive the utilization bounds of our MAC protocol.

1 Introduction

This paper addresses the scheduling of real-time convergecast packets in wireless sensor networks (WSN). Real-time applications require bounded service latency and hence such packets must be delivered to their destination within an specified duration. Since the channel is shared, the delay characteristics of MAC protocols are very important for such applications. Contention-based MAC protocols are not suitable for delay-sensitive as well as critical packets. CSMA is prone to heavy packet losses even if duty cycle is kept very low [2]. In this paper, we focus on contention-free scheduling based on TDMA with spatial reuse (STDMA).

Monitoring sensor readings for a continued period of time can be treated as a time-triggered application where periodic task model fits well. However, data generated by event triggered applications are aperiodic in nature. Providing periodic time slots for such cases is clearly wasteful. A more appropriate alternative is to provide a budgeted amount of time slots on per-group-of-nodes basis. For periodic traffic, time slots may be allocated on per-node basis as well as on per-group-of-nodes basis. In this work, we provide a unified solution for real-time as well as non real-time traffic, where for the real-time traffic, we consider periodic as well as aperiodic transmissions. Non real-time traffic is given best-effort service.

Arikan showed that the problem of deciding whether a TDMA-based packet radio network can support a specified bandwidth requirement is NP-Hard [3]. We introduced *hexagonal wireless sensor networks* in [7, 8]. Hexagonal WSN is a regular topology network characterized by nodes with six neighbors. In the general case, a dominating subset of nodes connected in hexagonal topology constitute the backbone and route multi-hop packets. Xue and Kumar established that for a flat (all nodes are peers) network of n nodes to be fully connected, the number of neighbors scale as $\Theta(\log n)$ [11]. This result indicates that a flat network organization is unsuitable for large-scale WSN.

The feasibility of hexagonal networks in practice has been established in [9]. Figure 1 shows a hexagonal network obtained using simulation of the hexagonal topology formation algorithm of [9] on a random deployment. The nodes forming the hexagonal backbone are connected using solid lines and non-backbone nodes are connected using dashed lines. Some of the major advantages of using hexagonal WSNs are that they admit highly scalable, efficient, and easily optimizable networking protocols, as illustrated by our previous work.

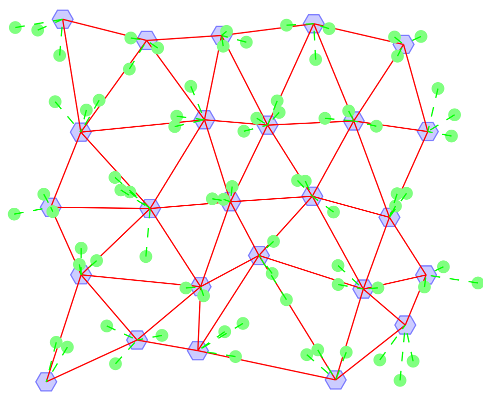


Figure 1. A hexagonal sensor network in a random deployment.

In [7], we presented a bounded end-to-end delay scheduling algorithm for convergecast that gives equal (unit) bandwidth to all hexagonal nodes. We encoded the conflict-free schedule as closed-form expressions. Thus each node computed its transmission slot locally. For the present problem, the per node bandwidth allocation is not a constant. Furthermore, at-least some local coordination is necessary to make scheduling decisions to avoid collisions and to utilize the per-group-of-nodes bandwidth allocations. The contributions of this work are as follows:

- A bandwidth allocation algorithm that satisfies periodic as well as aperiodic real-time traffic requirements.

- A conflict-free STDMA based distributed MAC protocol that conforms with the bandwidth allocations and gives best-effort service to non real-time traffic.

This paper is organized as follows: we present some background on hexagonal networks in Sec. 2. We present our bandwidth allocation algorithm in Sec. 3 followed by a presentation of the MAC protocol in Sec. 4. We derive schedulable utilization bounds in Sec. 5, followed by the results of simulation-based evaluations in Sec. 6 and conclusion in Sec. 7.

2 Background on Hexagonal Networks

Hexagonal networks refer to the topology where nodes have six neighbors, except for those that are at the edges of the network. In contrast, *hexagonal tessellation*, which is often used to model cellular networks, refers to a tessellation of the geographical area into hexagonally shaped cells. Hexagonal lattice corresponds to *triangular tessellation* (Figure 2). The lattice representations of hexagonal network are convenient for expositional purposes. Out of the three principal diagonals of hexagonal lattice, we select a pair inclined at 120 degrees to be the principal axes, labeled as X and Y. These three diagonals divide the plane into six regions which we refer to as *hextants*.

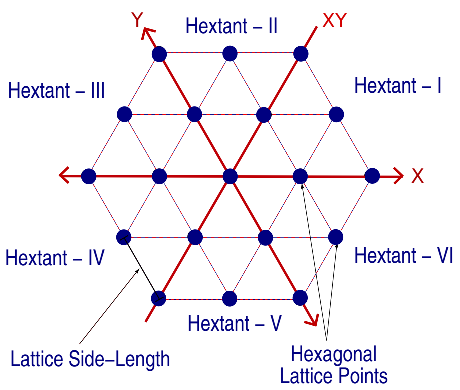


Figure 2. Hexagonal Topology

The euclidean distance between two neighboring lattice points is called the *lattice side-length*, or, side-length for short (Figure 2). A hexagonal lattice is completely specified by either only one pair of neighboring lattice points, or equivalently, by one lattice point and the lattice side-length. Let S denote the side-length of a hexagonal lattice. For a given lattice point (x_0, y_0), the infinite hexagonal lattice is generated by

$$x_0 \pm nS/2, y_0 \pm m\sqrt{3}S/2 \,, \tag{1}$$

where n and m are integers, both even or both odd. In a previous paper [9], we presented a distributed algorithm to construct hexagonal WSN in random deployments. The interested reader is referred to the paper for a detailed treatment. We implemented the algorithm for TinyOS based platforms. In field tests using 50 TelosB and MicaZ motes, our prototype constructed hexagonal networks of 6 hops diameter in less than 2 minutes.

Convergecast: sides and partitions. Due to the symmetry of convergecast, we consider the base station to be at the origin and the rest of the network arranged as concentric hexagons centered at the origin. This view allows us to address the nodes using tuple $[h, i]$ where h is the radius of the concentric hexagon expressed in terms of hop-counts from the center and i is the index of a node on this hexagon. We use the convention of incrementing the index in counter-clockwise direction starting from 0 at the X axis (more details on this and on routing and partitions can be found in [8]). We use Algorithm 1 for routing.

Algorithm 1 Routing Algorithm

Input: $[h, i]$ $\triangleright$ Node address
  $q \leftarrow \lceil i/h \rceil$
Output: $[h, i] \Rightarrow [h-1, i-q]$ $\triangleright$ Route

We chose the set of nodes on a segment of a given concentric hexagonal ring enclosed in one hextant as a basic unit for scheduling, which we refer to as a ***side***. We use the convention of including the node on the first diagonal in the anti-clockwise order in this set. For the purpose of concurrent conflict-free scheduling, we partition the nodes such that if only one node from each side in a given partition transmits concurrently, then no interference occurs. We assume that no interference occurs if the interfering transmitters are at-least two hops away from receivers. This work should be extendible to other graph-based interference models without much difficulty.

Let R be defined as $R([h, i]) \triangleq (h-1) \bmod 3$, and let Q be defined as $Q([h, i]) \triangleq \left\lfloor \frac{i}{h} \right\rfloor$. Then the partition P of $[h, i]$ is given by $P = (Q - 2R) \bmod 6$.

Theorem 1 ([8]) *The expression $P = (Q-2R) \bmod 6$ partitions the network such that transmissions of nodes on any two different sides in the same partition do not interfere.*

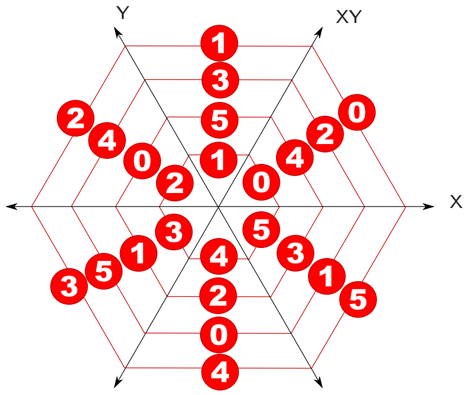


Figure 3. The six partitions.

3 Bandwidth Allocation

In this section, we consider allocation of bandwidth to hexagonal nodes in order to meet real-time traffic demand, i.e., to satisfy the traffic rate and delay requirements. We refer to the packets generated by a node as *local* packets. Local packets could also contain aggregated messages from the neighboring non-backbone nodes. In the following, the end-to-end latency refers to the delay incurred on a multi-hop route from a hexagonal node to the base station. The allocated bandwidth is consumed to transmit local packets and to route packets. We assume that a time slot is long enough to transmit one data packet and coordination messages associated with the transmission (details on the latter appear in the next section). In the following, we use the slot size as the unit of time. Let a node $n(h,i)$ send n^p delay sensitive real-time periodic packets every P_n units of time. To avoid the notations getting too cumbersome, we shall often drop the arguments whenever no ambiguity arises (e.g., $n(h,i)$ is abbreviated as n). Let D_n be the relative deadline of these packets. We impose no constraint on the deadline ($D_n \lesseqgtr P_n$). We use T to denote the size of one TDMA cycle.

Periodic traffic. The following treatment is for the case of bandwidth allocation for periodic real-time traffic on a per-node basis. Let l_n denote the bandwidth per T units of time allocated to node $n[h,i]$ for local periodic traffic. The packets scheduled at the beginning of a cycle, say at time t, must reach the base station by time $t + D_n$. Since nodes must transmit at-least once every D_n units of time, T can't exceed the minimum D_n for all n:

$$\mathsf{T} \leq D_{\text{Min}} \tag{2}$$

The number of cycles contained in time D_n is $\lfloor D_n/\mathsf{T} \rfloor$. Applying the equilibrium condition on the the rate of packet generation and the rate of packet consumption at the base station, we get

$$l_n \lfloor D_n/\mathsf{T} \rfloor \geq \lceil D_n/P_n \rceil n^p.$$

Thus the minimum l_n is given by

$$l_n = \frac{\lceil D_n/P_n \rceil}{\lfloor D_n/\mathsf{T} \rfloor} n^p. \tag{3}$$

Let $\mathcal{M}$ be the set of nodes whose packets are routed (using Algorithm 1) to n. Let f_i be the bandwidth allocated to node i for forwarding packets. Then,

$$f_n = \sum_{m \in \mathcal{M}} l_m + f_m. \tag{4}$$

Let $\mathcal{S}(h,k)$ (or, $\mathcal{S}$ when no ambiguity arises) denote a side, as well as, the hexagonal nodes located on the side, where h is the radius of the concentric hexagon on which the side is located and k is its hextant. Let $\mathcal{S}^p (\equiv \mathcal{S}^p(h,k))$ denote the total periodic bandwidth allocated to the side $\mathcal{S}$. Then,

$$\mathcal{S}^p = \sum_{n \in \mathcal{S}} l_n + f_n. \tag{5}$$

Observe that $\mathcal{S}^p$ is monotonically non-increasing in h, i.e., $\mathcal{S}^p(h,k) \geq \mathcal{S}^p(h+l,k) \;\; \forall\, l \leq H - h$. Since one packet from every side in any given partition may be scheduled concurrently without conflict, the sufficient bandwidth allocation to partition j for the periodic traffic, B_j^p, is given by:

$$B_j^p = \text{Max}(\mathcal{S}^p | \mathcal{S} \in \text{Partition}(j)). \tag{6}$$

We note that due to the monotonicity of $\mathcal{S}^p$, we need to consider only the first three hops to determine all six B_j^ps (Fig. 3).

Aperiodic delay-sensitive traffic. We put a budget on the aperiodic real-time traffic demand. The per T aperiodic bandwidth budget $\mathcal{S}^a$, shared by all nodes of a side $\mathcal{S}$ is $n_{\mathcal{S}} / \lfloor D_s/\mathsf{T} \rfloor$, where $n_{\mathcal{S}}$ is the number of packets to be scheduled for transmission in one deadline period. The aperiodic bandwidth allocation required for partition j, $\hat{B}_j^a$, is given by

$$\hat{B}_j^a \triangleq \text{Max}(\mathcal{S}^a | \mathcal{S} \in \text{Partition}(j)). \tag{7}$$

From (6), usually the bandwidth allocated for periodic traffic will exceed the bandwidth required by a side. Since these bandwidth slacks can be utilized to schedule aperiodic packets, the extra bandwidth in excess of B_j^p allocated to partition j for aperiodic traffic, denoted by B_j^a, satisfies $B_j^a \leq \hat{B}_j^a$. We define B_j^a to be the minimum bandwidth allocation in excess of B_j^p such that the aperiodic bandwidth requirements of all sides are satisfied. The total bandwidth allocation of partition j for real-time traffic, denoted by B_j is $B_j = B_j^p + B_j^a$.

For illustration, consider the example is presented in Figure 4. Suppose that the numbers on the sides of the hexagon represent the total periodic bandwidth requirements of the nodes on that side. Thus, the periodic bandwidths required

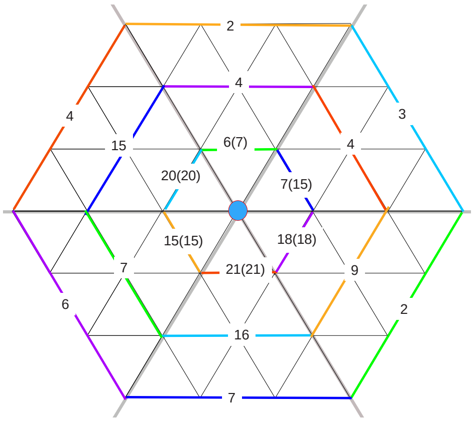


Figure 4. An example of bandwidth allocation. Only the first three hops are shown.

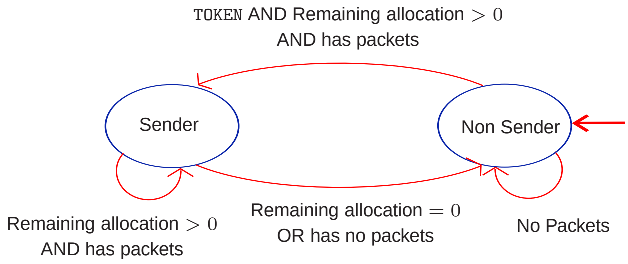


Figure 5. State diagram of nodes

by the first three sides of the first partition are $\mathcal{S}^p(1,1) = 7$, $\mathcal{S}^p(2,3) = 15$ and $\mathcal{S}^p(3,5) = 7$. From (6), the periodic bandwidth allocated to the first partition, B_1^p, is 15 (shown inside the parentheses). Similarly, $B_2^p = 7$, $B_3^p = 20$ and so on. Although the periodic bandwidth allocated to the first partition is 15, the maximum of the periodic bandwidth demands of all sides in the first hextant is only 7. Therefore, to allocate an aperiodic budget of up-to 8 time units to, say, the side at the first hop of the first hextant, no increment of B_1^p is needed. Therefore, in this case, $B_1 = B_1^p$. However, to allocate a budget of x time units to the side at the first hop of the third hextant, the bandwidth allocation of the third partition must be increased to $B_3 = B_3^p + x$ since no slack exists in this case.

4 MAC Protocol

In this section, we present a distributed MAC protocol that schedules transmissions in accordance with the bandwidth allocations presented earlier. This MAC protocol is inspired by the timed token protocol [4, 10, 6, 5, 1]. Our protocol and analysis are fundamentally different from these since we consider end-to-end scheduling and multi-hop routes.

The periodic bandwidth allocations are stored at the respective nodes. The nodes on the diagonals (of the concentric hexagons) coordinate the scheduling on their sides. They keep track of the side's aperiodic and total bandwidth allocations and their usage. The diagonal nodes determine which partition may transmit in a given time slot distributedly as follows: an integer array R of six elements is initialized to $B_j + H - 1$ where $B_j, j = 1 \ldots 6$ are the bandwidth allocations and H is the radius of the network. If partition j is transmitting in time slot t, then during time slot $t + 1$ partition k transmits where $R[k]$ is the first non-zero element after $R[j]$ (the next to the sixth element of the array is taken to be the its first element as in circular traversal.) All diagonal nodes decrement $R[k]$ by one at the beginning of the time slot $t + 1$. This is done till every element of $R[]$ is zero, which implies that the bandwidth allocated to real-time packets has been used up. After that the partitions may transmit the best effort traffic till the end of T time slots (round-robin or CSMA). The fractional bandwidths are supported by keeping a history.

If at the beginning of a time slot a diagonal node determines that the nodes on its side can transmit during the current slot, then it coordinates the right to transmit using token packets. Token packets originate at the diagonal nodes and get propagated towards the farthest node on a side. Upon reaching the farthest node, the tokens are propagated backwards to the diagonal node.

The tokens can be in two states, namely, `FWD-TOKEN` and `BKWD-TOKEN`. The tokens contain two bits of information. The first is set by the diagonal node if aperiodic allocation has not been used-up. Upon transmitting an aperiodic packet, a node resets the first bit. The second bit is set after a packet (of any kind) has been transmitted. If a node gets the token in the `FWD-TOKEN` state, it can transmit one aperiodic packet if the first bit of the token is set or one periodic packet if the node has not used-up its allocated periodic bandwidth during the current cycle (Figure 5). If a node gets the token in the `BKWD-TOKEN` state with no packet transmission so far, it can transmit one non real-time packet. The diagonal nodes update bandwidth use upon the receipt of the `BKWD-TOKEN`.

Token loss and other faults. Tokens are generated by the diagonal nodes at the beginning of *every* time slot. Therefore, a token loss amounts to at-most one packet missing a deadline. Observe that tokens play no role at the first and second hop transmissions. The significance of node failure depends upon the proximity of the node to the base station; the nearer the node to the base station, the more severe the packet loss. We note that a complete hexagonal structure is not necessary for the functioning of our protocol, which functions even with partially formed hextants, provided routes to the base-station exist.

5 Schedulability Analysis

In a given partition j, at-most B_j transmissions of real-time packets from the first hop node of the partition to the base station take place in one TDMA cycle, assuming that these packets arrive at the beginning of the cycles. Hence, if the transmissions are interleaved among the six partitions as described previously, then all real-time packets can be transmitted to the base station in $\sum_j B_j + K$ time units, where K is the initial "warm-up time," that is, the time needed for one packet from every hextant to reach the six first hop nodes. The exact value of K depends on a particular load distribution, but it is upper bounded by $6(H-1)$. Therefore, using (2), the feasibility conditions are:

$$6(H-1) + \sum_j B_j \leq \mathsf{T} \leq D_{\text{Min}}. \quad (8)$$

The remaining bandwidth $\beta = \mathsf{T} - 6(H-1) - \sum_j B_j$ can be used to transmit non real-time packets. Thus a TDMA cycle consists of an initial warm-up period of $6(H-1)$ time slots, followed by $\sum_j B_j$ time slots allocated for real-time traffic and β time slots for the best-effort traffic.

We now express the schedulable utilization as a function of deadlines and periods. From (3),

$$l_n = \frac{\lceil D_n/P_n \rceil}{\lfloor D_n/\mathsf{T} \rfloor} n^p = \frac{\lceil D_n/P_n \rceil P_n}{\lfloor D_n/\mathsf{T} \rfloor \mathsf{T}} \frac{n^p}{P_n} \mathsf{T} \ \geq \ \gamma \rho_n \mathsf{T},$$

where

$$\gamma = \text{Min}_n \left(\frac{\lceil D_n/P_n \rceil P_n}{\lfloor D_n/\mathsf{T} \rfloor \mathsf{T}} \right); \text{ and } \rho_n = \frac{n^p}{P_n}. \quad (9)$$

From (6) and the monotonicity of $\mathcal{S}^p$,

$$B_0^p = Max(\mathcal{S}^p(1,1), \mathcal{S}^p(2,3), \mathcal{S}^p(3,5)) \geq \mathcal{S}^p(1,1).$$

Similarly, $B_j^p \geq \mathcal{S}^p(j+1,1)\ \forall j$ and $B_j^a \geq \mathcal{S}^a(j+1,1)\ \forall j$. Therefore,

$$\begin{aligned} \sum_j B_j &\geq \textstyle\sum_j (\mathcal{S}^p(j+1,1) + \mathcal{S}^a(j+1,1)) \quad (10) \\ &\geq \textstyle\sum_n l_n + \sum_{h,j} \mathcal{S}^a(h,j)., \end{aligned}$$

We have $\sum_n \rho_n = \sum_n n^p/P_n$, and hence, is equal to the periodic traffic utilization, U^p. From(8) and (11),

$$\begin{aligned} \mathsf{T} &\geq \sum_n l_n + \sum_{h,j} \mathcal{S}^a(h,j) + 6(H-1) \\ &\geq \gamma \mathsf{T} \sum_n \rho_n + \sum_{h,j} \mathcal{S}^a(h,j) + 6(H-1) \\ \Rightarrow U^p + \frac{U^a}{\gamma} &\leq \frac{1}{\gamma} - \frac{6(H-1)}{\gamma \mathsf{T}} \quad (11) \end{aligned}$$

where U^a is the utilization of aperiodic real-time packets. Real-time traffic utilization $U_{RT} = U^p + U^a$. Therefore,

$$U_{RT} \leq \frac{1}{\gamma} - \frac{6(H-1)}{\gamma \mathsf{T}} - \frac{(1-\gamma)}{\gamma} U^a. \quad (12)$$

Hence, the schedulable utilization is an increasing function of cycle size T. Thus, when selecting the cycle size, the largest feasible value should be chosen.

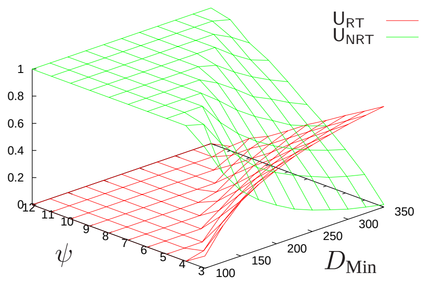
Similarly, it can be shown that U_{NRT}, the guaranteed utilization of best effort traffic is given by:

$$U_{NRT} \geq 1 - \Gamma U_{RT} + (\Gamma - 1) U^a, \quad (13)$$

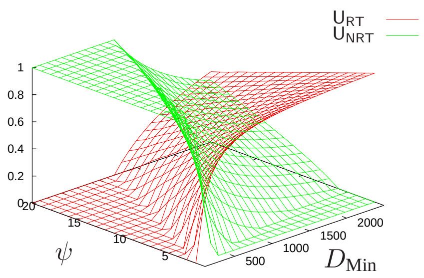
where,

$$\Gamma = \text{Max}_n \left(\frac{\lceil D_n/P_n \rceil P_n}{\lfloor D_n/\mathsf{T} \rfloor \mathsf{T}} \right). \quad (14)$$

6 Evaluations



(a) $N = 90$ $(H = 5)$



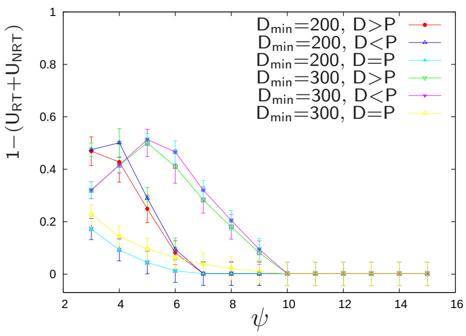
(b) $N = 330$ $(H = 10)$

Figure 6. Feasibility regions, $D_i = P_i$

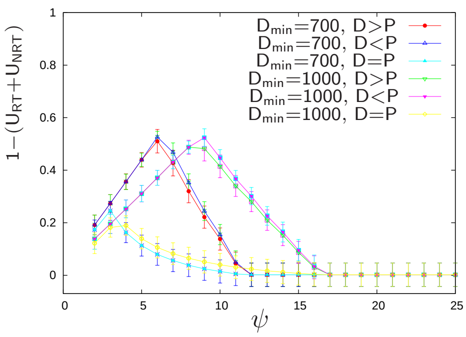
We implemented a simulator in C++ to evaluate the MAC protocol. The simulator takes the network size, minimum deadline, D_{Min}, and the ratio of D_{Min} and the TDMA cycle time T, denoted by ψ, as input parameters. It generates periodic packets with deadlines, the deviations of which from the D_{Min} are exponentially distributed. We generated three kinds of loads corresponding to the three cases of $P_n \geq, =, \leq D_n$. In the plots shown here, each data point shown is averaged over 1000 runs, where loads were generated afresh in each run. We performed the evaluations on networks of 90 and 300 hexagonal nodes, corresponding to hexagonal networks of radius 5 and 10 respectively.

The first set of figures (Figure 6(a)–6(b)) show the feasibility regions obtained by the MAC protocol. The corresponding plots for the $P_n \geq D_n$ and $P_n \leq D_n$ cases are qualitatively similar, and hence are omitted due to space limitations. Since $\mathsf{T} = D_{\text{Min}}/\psi$, larger values of ψ result in small cycle times. At small cycle times, the warm-up and token-passing overheads become significant and hence the real-time utilization decreases.

The cycle time depends on D_{Min} only. The periods enter into the bandwidth allocation expressions only as the ratio $\lceil D_i/P_i \rceil$, and since D_i are exponentially distributed, the real-time utilization, U_{RT} does not change noticeably for the cases of the periods being larger or smaller than the deadlines. Finally, figures 7(a)– 7(b) show the values of $1-(U_{RT}+U_{NRT})$ for a few cases. At smaller ψ, U_{RT} dominates and at larger ψ, U_{NRT} dominates the feasibility region. In the feasibility regions where both of these are non-zero, the sum of utilizations was less than 1. We found that this utilization gaps were the least when deadlines were the same as periods.



(a) $N = 90$ $(H = 5)$



(b) $N = 330$ $(H = 10)$

Figure 7. Plots of $1-(U_{RT}+U_{NRT})$

7 Conclusion

We presented an algorithm for bandwidth allocation to meet both real-time and non real-time traffic requirements in multi-hop wireless sensor networks. We presented an STDMA-based distributed MAC protocol that that satisfies the latency requirements of real-time traffic as well as supports best-effort service for non real-time traffic. This work illustrates that the topological regularity, such as that of hexagonal networks, facilitates highly scalable networking protocols, which is crucial for large-scale deployments.

Acknowledgments

We thank Tarek Abdelzaher for useful discussions. This work was partially supported by the Foundation for Science and Technology of Portugal (FCT) grant PTDC/EIA-CCO/109027/2008.

References

[1] G. Agrawal, B. Chen, W. Zhao, and S. Davari. Guaranteeing synchronous message deadlines with the timed token medium access control protocol. *IEEE Transactions on Computers*, 43(3):327–339, 1994.

[2] G.-S. Ahn, S. G. Hong, E. Miluzzo, A. T. Campbell, and F. Cuomo. Funneling-MAC: a localized, sink-oriented MAC for boosting fidelity in sensor networks. In *SenSys '06: Proc. of the 4th international conference on Embedded networked sensor systems*, pages 293–306, New York, NY, USA, 2006. ACM Press.

[3] E. Arikan. Some complexity results about packet radio networks. *IEEE Transactions on Information Theory*, 30(4):681–685, Jul 1984.

[4] R. M. Grow. A timed-token protocol for local area networks. In *Proc. of the Electro '82*, 1982.

[5] M. Hamdaoui and P. Ramanathan. Selection of timed token protocol parameters to guarantee message deadlines. *IEEE/ACM Transactions on Networks*, 3(3):340–351, 1995.

[6] R. Jain. Performance analysis of FDDI token ring networks: effect of parameters and guidelines for setting TTRT. In *SIGCOMM '90: Proc. of the ACM symposium on Communications architectures & protocols*, pages 264–274, New York, NY, USA, 1990. ACM Press.

[7] K. S. Prabh. *Real-Time Wireless Sensor Networks*. PhD thesis, University of Virginia, Charlottesville, VA, USA, 2007.

[8] K. S. Prabh and T. Abdelzaher. On scheduling and real-time capacity of hexagonal wireless sensor networks. In *ECRTS '07: Proc. of the 19th Euromicro Conference on Real-Time Systems*, pages 136–145. IEEE Press, Los Alamitos, CA, 2007.

[9] K. S. Prabh, C. Deshmukh, and S. Sachan. A distributed algorithm for hexagonal topology formation in wireless sensor networks. In *ETFA '09: Proc. of the 14th IEEE Intl. Conf. on Emerging Technologies and Factory Automation*. IEEE Press, Los Alamitos, CA, 2009.

[10] K. Sevcik and M. J. Johnson. Cycle time properties of the FDDI token ring protocol. In *SIGMETRICS/PERFORMANCE '86: Proc. of the 1986 ACM SIGMETRICS*, pages 109–110, New York, NY, USA, 1986. ACM Press, New York.

[11] F. Xue and P. Kumar. The number of neighbors needed for connectivity of wireless networks. *Wireless Networks*, 10:169–181, 2004.