2012 Hawaii University International Conferences
Education, Math & Engineering Technology
July 31st to August 2nd
Waikiki Beach Marriot Resort & Spa
Honolulu, Hawaii

# A Traffic Rate Monitoring Approach to Dynamic Clustered Wireless Sensor Network Topologies

**A.T. Burrell**
*Oklahoma State University*
*Computer Science Department*
*Stillwater, Oklahoma 74078*
*tburrell@okstate.edu*

**P. Papantoni-Kazakos**
*University of Colorado Denver*
*Electrical Engineering Department*
*P.O.Box 173364*
*Denver, Colorado 80217*
*Titsa.papantoni@ucdenver.edu*

# A Traffic Rate Monitoring Approach to Dynamic Clustered Wireless Sensor Network Topologies

**A.T. Burrell**
Oklahoma State University
Computer Science Department
Stillwater, Oklahoma 74078
tburrell@okstate.edu

**P. Papantoni-Kazakos**
University of Colorado Denver
Electrical Engineering Department
P.O.Box 173364
Denver, Colorado 80217
Titsa.papantoni@ucdenver.edu

## Abstract

We consider clustered wireless sensor network topologies, where each cluster contains a large number of distributed sensors whose life-span is limited by the level of their power consumption. In such network topologies, sensor expirations induce uncertainties in the cluster participation of each sensor, as well as traffic rate dynamics across all clusters, where the above uncertainties necessitate the deployment of Random Access transmission algorithms. In this paper, traffic rates are monitored, as the determining quantities for the adaptation of routing parameters and network architectural reconfigurations.

***Keywords***: dynamic sensor topologies; traffic rate monitoring; architectural reconfigurations; random access transmission; sensor expiration; priority data traffic.

## I. Introduction

In wireless sensor networks, data generated from the sensors are transmitted via appropriate multiple access protocols, [12], [22], [24], [26] whose performance is a function of the their input data rates; and are collected and processed by life-limited nodes, whose life-span is a function of the data rates they process, [1], [14], [15], [17]. Required overall data rates, in conjunction with rate-dependent transmission protocol performance and node life-spans, necessitate network-architecture and network-operations adaptations, so that the nodes' survivability limitations do not interfere with the required network overall performance, [4], [30]. Since the network-architecture and network-operations adaptations are functions of the acting data rates, it is eminent that data rates be monitored and that rate changes be detected accurately and rapidly, [23].

The distinguishing feature in wireless sensor networks is limited life-spans of the nodes, induced by energy consumption. Interesting results focusing on energy consumption have been obtained by several researchers: Bounds on energy conservation techniques have been derived in [5], role assignments targeting energy conservation have been developed in [4], energy conservation routing techniques have been proposed in [14], [15], [16] and issues arising due to energy conservation have been discussed in [16]. In addition, topological and node-cooperation issues have been included in [29], [30], while approaches to performance monitoring have been presented in [11]. An interesting rate allocation algorithm has been presented in [17], which is based on a modification of the max-min routing in [3] and the lexicographic linear programming approach in [20]. In [23], a dynamic rate allocation methodology has been presented that is facilitated by a powerful data rate monitoring algorithm. In [2], energy efficient clustering algorithms are proposed, including a discussion on LEACH approaches, where energy consumption is assumed to be strictly a function of geographical distances and where transmission collisions are completely ignored. Considering transmission protocols, those proposed or partially implemented (in Zigbee, IEEE802.15.4 etc.) within the Random Access class are ALOHA based, inducing well-known instabilities which pull the throughput rapidly to zero.

In this paper, we consider wireless clustered sensor network topologies containing many randomly distributed sensors that generate high priority data. We deploy specific stable random access transmission protocols per cluster, that facilitate high priority data. Data rates are time-varying in such a network, mainly due to expiring life-limited nodes. We thus consider dynamic routing adaptations and dynamic architectural reconfigurations, dictated by data rate variations and facilitated by a higher level protocol that monitors such variations. The organization of the paper is as follows: In Section II, we present the system model. In Section III, we describe a data transmission random access protocol that we recommend for deployment in each cluster. In Section IV, we summarize the data rate monitoring protocol used to facilitate the routing adaptation and architectural reconfiguration processes and. In Section V, we discuss overall system issues. In section VI, we draw conclusions.

## II. System Model

We consider a clustered network architecture, as in [17] and [23]. The components in the architecture are micro-sensors, micro-sensor clusters and a backbone network of cluster-heads and a fusion-center. In [17] and [23], the micro-sensors, the cluster-heads and the fusion-center have been respectively termed Micro-Sensor Nodes (MSNs), Aggregation and Forwarding Nodes (AFNs) and Base Station (BS). We will use the same terminology for consistency. Given a pre-determined signal processing objective, the MSNs, AFNs and BS perform the following functions: (a) The MSNs are grouped into distinct clusters, where each cluster contains a single AFN. Each MSN collects local data and transmits them to its local AFN, via some multiple access transmission protocol implemented on a distinct channel associated with the AFN; that is, the overall network encompasses distinct separate channels (such as distinct frequency bands), each associated with a single AFN. ***Since the identity of the sensors contained in each cluster may change due to sensor possible expiring and subsequent architectural reconfigurations, the appropriate class of the deployed multiple access protocols is the Random Access Class, (RAs)*** [11], [12], [22], [24] [26]. The AFNs and the BS are connected via separate channels (or frequencies) comprising the backbone network.

Commands regarding routing adaptations in the latter network and system architectural reconfigurations propagate through the backbone network and do not interfere with the MSN transmissions. Possible reconfiguration broadcasts to MSNs are implemented via their local AFN, either on separate channels (frequencies) or on dedicated minislots of the MSN transmission channel(s) and do not interfere with the MSN transmissions. The MSNs are low-cost and low-energy; thus, short-life devices, and some of them generate high priority data that must be received by the corresponding AFNs in relatively short time. (b) Each AFN collects the data sent by its local MSNs and processes them, using an operation determined by the network signal processing objective [19]; it also receives processed data sent by other neighboring AFNs. The AFN then processes the compounded processed data, utilizing an operation that is determined by the network signal processing objective [19], and transmits the outcome to selected neighboring AFNs or the BS. The AFNs have processing capabilities and are devices with energy and life-spans that are much higher than those of the MSNs; their life-spans and energy are still limited, however. (c) The BS fuses data transmitted to it by neighboring AFNs, utilizing an operation that is determined by the network signal processing objective [19]. The BS has practically unlimited life-span and processing power.

The MSNs transmit to their assigned AFN via a RA protocol. Operations are performed at all nodes of the backbone network: at the AFNs and the BS. The nature of the operations is determined by the network signal processing objective, the environment that generates the data and the data rates [19]. At the same time, the energy consumption of the AFNs is a function of the data rates they receive and produce, and the complexity of the operations they perform. Each AFN performs operations on its input data rates, to produce the data rates it outputs to neighboring AFNs and/or the BS. Let $\lambda_{Ci}$ be the data rate from the MSNs in the cluster of the $i^{th}$ AFN to the AFN; let T be the time constraint imposed on the network for completing the designated signal processing operation; let $\mu$ be the data rate collected by the BS. We will assume that the MSNs per cluster are spatially randomly and independently distributed, and their number is large. In addition, their life-span is short: Thus, each MSN basically transmits a short duration data message/or signal and expires. This, gives rise to a limit Poisson data packet user model per cluster [22]; that is, the cluster user model reflects infinitely many independent Bernoulli users, each generating a single data packet, where an overall Poisson data packet traffic is induced.

We initially consider the static problem stated in [23] and [17]: The network signal processing objective is specified, the network architecture is fixed, the data environments and the RA transmission protocols are known and unchanged, the operations performed by the AFNs are determined and the data rates $\{\lambda_{Ci}\}$ generated in the clusters are fixed. Given, in addition, the time constraint T, for the completion of the signal processing objective, the performance of the network will be maximized when the maximum number of data are fused in time length T [17]. Since the fusion operation is performed by the BS and since the maximum number of data in T corresponds to the maximum attainable data rate, maximum network performance is then attained when the data rate $\mu$ is maximized. A generally nonlinear dynamic programming problem is then formalized to determine the optimal routing protocol in the backbone network comprised of the AFNs and the BS. This is known as the *Static Rate Allocation* problem, [23], [21].

The network is required to complete its signal processing objective within T time units. During this time period, some MSNs generally expire, since their life-spans are only fractions of the time period T. This causes changes in the cluster data rates $\{\lambda_{Ci}\}$, and thus induces dynamics, first in the rate allocation problem, and then in network architectural configurations. Specifically, the rates $\{\lambda_{Ci}\}$ may generally change during the T time period, when the network operates towards the satisfaction of its objective. For rate changes that are relatively modest, a dynamic rate allocation problem arises: if such changes of the rates $\{\lambda_{Ci}\}$ can be detected, then the constants of the static rate allocation problem will be adjusted accordingly, and the new data rate allocations will be then dictated by the solution of the adjusted problem. For more dramatic detected rate changes, the satisfaction of the overall system performance requirements may necessitate network architectural reconfigurations. To detect changes in cluster data rates, an algorithm must be developed, that detects changes accurately and rapidly. Such algorithm will be deployed at the AFNs, as an upper level protocol, it will detect changes and will communicate them across the AFNs. Then, either a static rate allocation algorithm on the unchanged network architecture will be initiated that uses the newly detected cluster rates as constants, or an architectural network reconfiguration will be implemented with a matching rate allocation technique.

The allowable values of the cluster data rates $\{\lambda_{Ci}\}$ in the network are determined by the throughput/delay characteristics of the transmission Random Access (RA) algorithms deployed in the clusters. Specifically, given the cluster transmission RA algorithm, the highest allowable cluster data rate should be such that the induced delays and retransmissions are non-detrimental to the network mission, in the sense of causing the expiration of an unacceptable percentage of MSNs. We will elaborate upon this concept in Section IV. We note that algorithmic delays induced by RAs are dependent on their algorithmic throughput. Thus, the cluster data rates $\{\lambda_{Ci}\}$ are all bounded from above by bounds determined by the characteristics of the per cluster deployed transmission protocols from the MSNs to the corresponding AFN. In addition, as induced by the deployed transmission algorithm, when the data rate in a cluster drops below a certain level, the existence of the cluster becomes wasteful, due to the then unnecessary induced increase in the size of the backbone network. Assuming that the transmission RA protocol per cluster is fixed, identical for all clusters and known, let us denote by $\nu_l$ and $\nu_u$ the common determined lower and upper bounds to each cluster data rate, respectively. If the aggregate data rate from all MSNs in the overall network is denoted $\lambda_C$, the smallest possible number of clusters in the network is then $\lceil \lambda_C/\nu_u \rceil$, while the largest such number is then $\lceil \lambda_C/\nu_u \rceil$.

We will assume that a rate monitoring algorithm is deployed at the AFNs. When the data rates in all clusters remain within the $(\nu_l, \nu_u)$ range, then, detected data rate changes dictate only rate reallocations in the backbone network of the overall structure, without any imposed architectural reconfigurations, as in [23]. When, instead, some cluster data rates fall outside the $(\nu_l, \nu_u)$ range, then, architectural reconfigurations are first imposed, followed by data rate reallocations on the new backbone network topology, as dictated by the deployed routing algorithm.

## III. The Per Cluster Random Access Transmission Protocol

Due to the dynamically changing architectural reconfigurations in the considered sensor topologies, the identities of all sensors are not known to the AFNs at all times. Thus, the unknown user population model arises, where the sensors are the users in this case: the identity of each sensor becomes known to the system, only after the sensor accesses successfully some AFN. The only possible class of transmission protocols for the unknown user population model is the Random Access (RA) class, [25]. In addition, within the RA class of transmission protocols, the only implementable sub-class is the stable class of Limited Sensing Random Access Algorithms (LSRAAs), where it is required that upon generation of a message, each user starts monitoring the channel feedbacks continuously, until this message is successfully accepted for transmission, [12], [22]. [24], [26]. We emphasize that the ALOHA based protocols currently proposed (e.g. Zigbee, IEEE802.15.4, etc.) are unstable, leading to rapidly decreasing throughputs with increasing MSN rates.

As stated in Section II, we assume a limit Poisson data packet user model per cluster. We will also assume a synchronous system, where all AFN channels are slotted synchronously across all AFNs, and where a slot is the time corresponding to the transmission of a single packet. In this model, each packet represents a separate independent user (separate MSN here); thus, any number of packets may simultaneously attempt transmission, unless organized by some protocol; an LSRAA, in this case.

As stated in Section II, the MSNs in a single cluster transmit through a single channel (to their AFN) and overall system connectivity is attained via the backbone network. Due to the possible MSN mobility or the dynamics of the architectural topology, some MSNs may move across clusters, while they have a packet to transmit. At cluster boundaries, such MSNs are exposed to feedbacks from more than one cluster channels and have the capability to transmit through anyone of those channels. The latter phenomenon can be exploited to avoid temporary isolation of users moving across clusters, and especially to reduce their transmission delays while in transit across clusters. In addition, some MSNs may generate high priority data, requiring accelerated transmission. We encompass boundary MSNs into the high priority category and name them all *high priority MSNs*. We name non-high priority MSNs, *regular MSNs*. The following approach can be then taken: To each regular MSN assign as a single channel for its transmissions that corresponding to its local AFN. To each MSN that is high priority, provide the capability to monitor and possibly transmit through a group of channels, instead, associated with the corresponding group of AFNs. The MSNs can be then partitioned into two categories: the regular MSNs transmitting through a single local channel, and the high priority MSNs having access to several cluster- channels. If we consider the environment where high priority MSNs have access to M AFN channels, then incorporating the regular MSNs, M+1 classes of MSNs arise: (a) M classes of regular MSNs, each class transmitting through one of the distinct M AFN channels, and (b) the (M+1)th class of high priority MSNs that have access to all the M AFN channels. The objective is the design of Limited Sensing Random Access Algorithms (LSRAAs) for the system that require no knowledge of the system state by the users and present the high priority MSNs with a delay advantage (compared to the non high priority MSNs), even when the traffic rates in the system change dynamically, while keeping the delays of the regular MSNs as low as possible. When architectural reconfigurations are in place, each regular MSN selects one of the AFNs equiprobably, while each high priority MSN selects one M-size group of AFNs, equiprobably among all possible such groups, for their transmissions.

For the multi-channel system considered in the above paragraph, let the M channels be indexed from 1 to M. Then, the regular MSNs assigned to channel i deploy a limited sensing random access algorithm, named LSRAAi, for their transmissions. For each of these LSRAAs, we adopt the class of limited sensing random access algorithms in [12] due to their simple operational properties and their performance characteristics: the algorithms are easily implemented in the limited sensing environment, they operate with binary (collision versus non-collision) feedback, they have superior resistance to feedback errors and in the presence of the limit Poisson user model they attain a throughput of 0.43. ***Throughput is here defined as the highest traffic rate maintained by the algorithm with finite delays.*** The high priority MSNs in the system deploy the same LSRAA as that the local MSNs do, in conjunction with a ***selection policy*** via which they choose their transmission channel. A selection policy may be either dynamic or probabilistic. The operations of a dynamic selection policy are dictated by the feedback sequences that the boundary MSNs observe. In contrast, a probabilistic selection policy is represented by a priori assigned probabilities. A probabilistic selection policy is simple, but it does not generally present the high priority MSNs with significant advantage, especially in the presence of changing rates of the regular MSN traffics. In this paper, we propose a dynamic selection policy which does not require any a priori system state knowledge, and which indeed presents the high priority MSNs with a significant delay advantage. The policy, which was first introduced in [24], induces light delay penalization for the regular MSNs, at the expense of a reduction in overall system throughput.

Considering an M-channel system as above, we assume that some synchronous LSRAA is deployed by all the MSNs in the system. In particular: (1) Time is divided into slots of length equal to the duration of a packet, and the starting instants of the slots are identical in all channels. (2) Per channel, feedback per slot exists and corresponds to the outcomes induced by the local LSRAA. This feedback is binary, collision versus non-collision (NC). (3) Each regular MSN is required to monitor the feedback from its assigned channel continuously from the time it generates a new packet to the time that this packet is successfully transmitted. We initially assume that no propagation delays and no forward or feedback channel errors exist in the system.

We also assume that each high priority MSN receives the feedbacks from all channels correctly and without propagation delays. At the time when a high priority MSN generates a new packet, it starts monitoring the feedbacks from all channels continuously until it decides to join the operations of one of the LSRAAs for the transmission of his packet. Upon this decision, it maintains the continuous monitoring of only those feedbacks that correspond to the LSRAA it chose, until its packet is successfully transmitted.

The regular MSN populations for each channel and the high priority MSN population are all modeled as limit Poisson. That is, it is assumed that the regular data packet traffic i, i = l, 2, … , M, is a limit Poisson process with intensity $\lambda_i$ , i = l, 2,..., M, and that the data packet traffic generated by the high priority MSNs is another independent limit

Poisson process with intensity $\lambda_{M+1}$. As an interesting remark, for a large class of LSRAAs, including that considered in this paper, the limit Poisson user model provides a lower bound in performance within the class of identical and independent users whose packet generating process is independent and identically distributed (i.i.d.).

We assume that the M LSRAAs in the M-channel system are identical and belong in the class of window algorithms in [12], all of which induce throughput 0.43 and operate with binary feedback. Upon generation of a new packet, a high priority MSN imagines itself belonging to the systems of all M LSRAAs and follows their algorithmic steps until the first time that they enter a collision resolution event in one of them: the rules of this first entry are stated by the channel selection policy. Then, the MSN remains with the latter LSRAA system, until its packet is successfully transmitted.

Let time be measured in slot units, where slot t occupies the time interval [t, t+1). Let $x_t(j)$ denote the feedback that corresponds to slot t for channel j, j=1,2,..,M, where $x_t(j)=C$ and $x_t(j)=NC$ represent collision and non-collision slot t for channel j, respectively. The LSRAA for channel j is implemented independently by each MSN in the system; its steps are dictated by the feedback sequence $\{x_t(j)\}_t$. Each regular MSN is assigned one of the channels a priori and thus observes only the feedback sequence corresponding to the latter channel. Each high priority MSN observes, instead, the feedback sequences of both channels until it selects the channel which it will transmit its packet through. Below, we first explain the algorithmic steps implemented by each local MSN, dropping the channel index j, for simplicity in notion.

Each algorithm in the class in [12] utilizes a window of size $\Delta$ as a operational parameter and induces a sequence of consecutive Collision Resolution Intervals (CRIs). The window length $\Delta$ is subject to optimal selection for throughput maximization. Each CRI corresponds to the successful transmission of all packet arrivals within an arrival interval of length $\Delta$. The length of the CRI is determined by the number of MSNs in the window $\Delta$ and the algorithmic steps of the collision resolution process. The placement of the $\Delta$-size window on the arrival access is determined asynchronously by the MSNs. We will first describe the collision resolution process induced by the algorithm. Then, we will explain the process which determines the placement of the $\Delta$-size window per CRI.

The algorithmic class contains algorithms whose collision resolution process can be depicted by a stack with finite number of cells. Let us consider this algorithm in the class which can be described by a K-cell stack. Then, in the implementation of the collision resolution process, each MSN utilizes a counter whose values lie in the set of integers, [1,2,…K]. We denote by $r_t$ the counter value of some MSN at time t. The K different possible values of the counter place the user in one of the K cells of a K-cell stack. When its counter value is 1, the MSN transmits; it withholds at K-1 different stages otherwise. When a CRI begins, all MSNs in the $\Delta$-size window set their counters at 1; thus, they all transmit within the first slot of the CRI. If the window contains at most one packet, the first slot of the CRI is a non-collision slot and the CRI lasts one slot. If the window contains at least two packets, instead, the CRI starts with a collision which is resolved within the duration of the CRI via the following rules:

The MSN transmits in slot t if and only if $r_t = 1$. A packet is successfully transmitted in t if and only if $r_t = 1$ and $x_t = NC$.

The counter values transition in time as follows:

If $x_{t-1}$ = NC and $r_{t-1}$ = j ; j=2,3,..,K, then $r_t$ = j-1
If $x_{t-1}$ = C and $r_{t-1}$ = j; j=2,3,…K, then $r_t$ = j
If $x_{t-1}$ = C and $r_{t-1}$ = 1, then, $r_t$ =

$$r_t = \begin{cases} 1 & ;\ \text{w.p. } 1/K \\ 2 & ;\ \text{w.p. } 1/K \\ 3 & ;\ \text{w.p. } 1/K \\ \ldots & \\ K & ;\ \text{w.p. } 1/K \end{cases}.$$

For any given K, the throughput of the algorithm is 0.43. This throughput is attained for different optimal window sizes $\Delta^*$, as K varies. For K=2, $\Delta^* = 2.33$. For K=3, $\Delta^* = 2.56$.

From the above rules, it can be seen that a CRI that starts with a collision slot ends with K consecutive non-collision slots, an event which can not occur at any other instant during the CRI. Thus, the observation of K consecutive non-collision slots signals the certain end of a CRI to all users in the system; it either signifies the end of a CRI that started with a collision or the end of a sequence of K consecutive length-one CRIs. Therefore, a MSN packet that arrives in the system without any knowledge of the channel feedback history can synchronize with the system upon the observation of the first K-tuple of consecutive non-collision slots. This observation leads to the asynchronous by the MSNs generating of the size-Δ window placement on the arrival axis. Specifically, if a CRI ends with slot t, the window of the next CRI is selected with its right most edge K-1 slots to the left of slot t and it contains those packets whose *updates* fall in the interval (t- K+1-Δ , t-K+1). The *updates* $\{t^k\}$ of a packet are generated as follows: Let $t_0$ be the slot within which a packet is generated. Then define $t^0$ to be equal to $t_0$. Starting with slot $t^0$, the corresponding regular MSN senses continuously the channel feedbacks. It does so passively, until it observes the first K-tuple of consecutive NC slots, ending with slot $t_1$. If $t^0 \in (t_1–K+1 - \Delta,\ t_1–K+1)$, the MSN participates in the CRI that starts with slot $t_1 + 1$. Otherwise, it updates its arrival instant to $t^1 = t^0 + \Delta$ and waits passively until the end of the latter CRI, ending with slot $t_2$. If $t^1 \in (t_2–K+1-\Delta,\ t_2–K+ 1)$, the MSN packet participates in the CRI which starts with slot $t_2$; otherwise, the MSN updates its arrival instant by Δ again and repeats the above process. In general, if $\{t_n\}$ $n \geq 1$ denotes the sequence of consecutive CRI endings since the first K-tuple of consecutive NC slots, the MSN packet participates in the $k^{th}$ CRI if $t^{k-1} \in (t_k –K+1-\Delta,\ t_k –K+1)$ and $t^n \notin (t_{n+1} –K+ 1 - \Delta,\ t_n –K+1)$ ; for all $n \leq k-2$.

Each high priority MSN observes all M feedback sequences $\{x(j)\}_{t\geq t1}$; j=1,…,M and follows the evolution of the M time sequences $\{t_i(j)\}_{i\geq 2}$ ; J=1,…,M. If $t_k(m) < t_k(j)$; for all j = 1,…,M that are different than m, then, in slot $t_k(m)$+l its packet enters a CRI within the LSRAA of channel m, and the packet is successfully transmitted during the process of this CRI. When a high priority MSN with a packet to transmit observes simultaneous beginnings of N CRls and its updates are within the Δ window of all of them, it selects each one of the K LSRAAs with probability 1/N. The evolution of the updates for each LSRAA and each high priority MSN is exactly as those of the regular MSNs. That is, the arrival sequences are always updated by $\Delta$ , and the $\Delta$-size window is equally divided whenever the updates of packets fall simultaneously within the window of N CRIs.

*Expiration of MSNs during the Transmission Process*

The MSNs that have a packet to transmit consume energy due to channel monitoring and due to retransmissions within the CRI of that accommodates the transmission of the packet. If the energy consumed this way exceeds their limit, the MSNs expire and their packet is lost.
We will assume that the energy consumed per single observed channel slot is a constant, and so is the energy consumed per retransmission. Then, we form a linear expiration expression, as explained below. Let us first define:

$\beta$: The amount of energy consumed by the monitoring of a single channel slot by a MSN

$\eta$: The amount of energy consumed by a single packet (re) transmission by a MSN

$\tau_1$: The number of slots during which a MSN monitors both channels in the system

$\tau_2$: The number of slots during which a MSN monitors a single channel in the system

m: The number of retransmissions during the CRI

$\xi$: The total energy stored in a single MSN for packet transmission

Using the above notation, we assume that the MSN expires, with subsequent loss of its generated packet, if:

$$\beta\,(2\tau_1 + \tau_2) + \eta m > \xi \qquad (1)$$

As first stated in Section II, the MSN expiration, as given by expression (1), causes packet rejections and subsequent reduction of cluster traffic rates. Other causes of MSN expiration may be environmental hazards or random electronic failures. Regardless of the causes, MSN expirations may result in traffic rate reductions that dictate architectural reconfigurations.

*Algorithmic Analysis*

In the absence of MSN expirations, the 2-channel system induces regenerative points within its stability region. Then, the regenerative theory can be applied to derive the latter stability region. In the case of the M-channel system, with M > 2, interactions among the M LSRAAs is induced, and the evolving recursions for their evaluation give rise to multidimensional coupling. It is possible to devise a simplified recursive methodology which will basically first decompose the M-channel system into lower dimensionality subsystems, and then use these subsystems as units towards an overall system evaluation. Results on the 2-channel system, as well as specifics on the M-channel decoupling methodology and the resulting throughput/delay results can be found in [24].

We point out that the stability results in [24] represent the case where the traffic is ***maintained***, meaning that no MSNs expire. We also point out that when traffic is maintained, the collision resolution process induced by the deployed LSRAAs changes the traffic statistics: that is, when the input traffic is limit Poisson, the output traffic is not Poisson then, while it maintains the input rate and can be closely approximated by a Poisson process.

Considering the 2-channel, let $\lambda_1$ and $\lambda_2$ denote the Poisson rates of the regular MSN traffics assigned to channels 1 and 2, respectively; let then $\lambda_3$ denote the rate of the Poisson traffic induced by the high priority MSNs. From the throughput/delay results in [24], we decide that: (a) below the ($\lambda_1$=$\lambda_2$=0.1, $\lambda_3$=0.1) rate region, the system is wasteful, and (b) above the ($\lambda_1$=$\lambda_2$=0.3, $\lambda_3$=0.05) rate region, the delays of the regular MSNs are unsatisfactory. Rate region ($\lambda_1$=$\lambda_2$=0.1, $\lambda_3$=0.2) reflects 0.15 rate per cluster and 33% of the total MSN population being high priority MSNs. Rate region ($\lambda_1$=$\lambda_2$=0.3, $\lambda_3$=0.05) reflects 0.325 rate per cluster and 7.6% of all MSNs being high priority MSNs. Referring to the architectural reconfiguration implementation in Section II and the notation there, we may initially subsequently select:

$$\nu_l=0.15, \ \nu_u=0.325, \text{ between 0\% and 10\% high priority MSNs}$$

We note that the selection of the $\nu_l$ and $\nu_u$ values should be finally tuned depending on the MSN expiration constants in (1).

## IV. The Data Rate Monitoring Protocol

As stated in Section II, routing adaptations and architectural reconfigurations are facilitated by a high level data rate monitoring protocol. In this section, we present the highlights of the algorithm that implements the latter protocol, as presented and fully analyzed in [23]. Isolating a single cluster in the system, we first ***denote by $\lambda$ the data rate accessing the AFN of the cluster: this data rate encompasses MSN expirations***. We subsequently decide on the $\nu_l$ and $\nu_u$ rate values to be monitored, where 0.15 and 0.325 has been respectfully selected in Section III. The objective is then to decide rate shifts from $\nu_u$ to $\nu_l$. Below, we will summarize the generalized algorithm from [23], which monitors shifts among a set $\{\lambda_i\}$ of rates.

The proposed algorithm is sequential, with its operational values updated at discrete, equally spaced, time instants $\{T_i\}_{i\geq 0}$ , where the time interval between consecutive $T_i$'s is a fraction of the life expectancy of MSNs. We name the time interval between two consecutive $T_i$'s, *frame*. Let us denote by $\{S_i\}_{i\geq 0}$ the subsequence of the sequence $\{T_i\}_{i\geq 0}$, such that , $T_0 = S_0$ is the beginning of time when the system starts operating, and $S_i$ is the ith after $S_0$ time instant (corresponding to the beginning of some frame) at which the monitoring algorithm decides that a shift in the acting rate region has just occurred. Let us assume that, given any fixed rate $\lambda_k$, the cluster data arrival process is stationary, and let $f_k$ ( $n_1,\ldots,n_q$ ) denote its q-dimensional distribution in frame lengths; that is, $f_k$ (

$n_1,\ldots,n_q$ ) is the probability that, in a sequence of q consecutive frames, $n_1$ data arrivals occur in the first frame, and so on, with $n_q$ data arrivals finally occurring in the qth frame, given that the active rate is $\lambda_k$ throughout. Let us assume that the distributions { $f_k$ ( $n_1,\ldots,n_q$ ) } are well-known for all k's (or all $\lambda_k$'s ). Then, we propose the following sequential data rate monitoring algorithm, from [23], which traces consecutive data rate shifts.

### *A. The Algorithmic System*

1. The monitoring algorithmic system is designed at the central rates $\{\lambda_k\}_{1\leq k\leq n}$.
2. Let at some $S_i$ the rate $\lambda_k$ be decided as just starting, by the monitoring algorithmic system. Then, immediately (at $S_i$), a set of n-1 parallel algorithms starts operating. The $j^{th}$ such algorithm is monitoring a possible shift from the rate $\lambda_k$ to the rate $\lambda_j$, where $\lambda_k \neq \lambda_j$ , sequentially, with adaptations occurring at beginnings of frames. The n-1 parallel algorithms use a reflective threshold at zero and a common decision threshold, $\eta_k$ . Let $V_{kj}$ (0) denote the operating value of the algorithm monitoring a $\{\lambda_k \rightarrow \lambda_j\}$ shift at $S_i$ , when it starts operating; let $V_{kj}(r)$ denote its operating value at its $r^{th}$ adaptation step. Then, the operating values of the algorithm are sequentially updated as follows:

$$\begin{aligned} &V_{kj}(0) \triangleq 0 \\ &V_{kj}(r+1) = \max\left(0, V_{kj}(r) + \log \frac{f_j(n_{r+1} \mid n_1,\ldots,n_r)}{f_k(n_{r+1} \mid n_1,\ldots n_r)}\right) \end{aligned} \quad (2)$$

$S_{i+1}$ is then the first time after $S_i$ that one of the above n-1 algorithms *first* crosses the common decision threshold $\eta_k$. If the latter algorithm is the one which monitors a $\lambda_k \rightarrow \lambda_\rho$ shift, a set of n-1 parallel algorithms, each monitoring a shift from $\lambda_\rho$ to one of the remaining rates, becomes initialized. The latter algorithmic system has operational characteristics as those stated above, utilizing a common decision threshold $\eta_\rho$.

### *The Algorithm for the Poisson*

As explained in Section II, the data generated by the MSN's in cluster i may be modeled as cumulatively comprising a homogeneous limit Poisson process, with rate $\lambda_{Ci}$ bits/time unit. As stated in Section III, the traffic accessing the AFN of the cluster, while not Poisson then, it can be closely approximated by a Poisson process. The Poisson arrival process is memoryless: the conditioning in the log-ratio updating step of the algorithm is (2) then drops. If *d* denotes the length of a frame in time units, the $\{\lambda_k \rightarrow \lambda_j\}$ monitoring algorithm in (2) takes the following form:

$$V_{kj}(0) \overset{\Delta}{=} 0$$

$$V_{kj}(r+1) = \max\left(0, V_{kj}(r) + d\left[\lambda_k - \lambda_j + n_{r+1}\log\left(\frac{\lambda_j}{\lambda_k}\right)\right]\right) \quad (3)$$

where $n_{r+1}$ denotes the number of transmitted data within the $(r+1)^{th}$ frame from the beginning of the $\{\lambda_k \rightarrow \lambda_j\}$ monitoring algorithm.

Let us define

$$\zeta(\lambda_k, \lambda_j) \triangleq [\lambda_k - \lambda_j][\log(\frac{\lambda_k}{\lambda_j})]^{-1} \quad (4)$$

where $\min(\lambda_k, \lambda_j) < \zeta(\lambda_k, \lambda_j) < \max(\lambda_k, \lambda_j)$. We may select $\{\lambda_i\}$ rates such that $\zeta(\lambda_k, \lambda_j)$ are rational numbers for all k and j, and we then define the integers $t_{kj}$ and $s_{kj}$ , $t_{kj} < s_{kj}$, as follows:

$$\zeta(\lambda_k, \lambda_j) = \frac{t_{kj}}{s_{kj}} \quad (5)$$

The algorithmic thresholds can then all be selected as positive integers, and without lack in generality, the algorithmic adaptation in (3) can be transformed as follows:

$$V_{kj}(r+1) = \max\{0, V_{kj}(r) + (-1)^{ime(k,j)}[s_{kj}n_{r+1} - dt_{kj}]\} \quad (6)$$

where,

$$ime(k,j) = \begin{cases} 0; \text{ if } \lambda_j > \lambda_k \\ 1; \text{ if } \lambda_j < \lambda_k \end{cases}$$

## V. Overall System Issues

When a $\nu_u \rightarrow \nu_l$ shift is detected by at least one of the cluster data rate monitoring high level protocols, an architectural reconfiguration algorithm may instruct the overall system to reconfigures itself, where all MSNs reselect then the AFN(s) that they will be connected to for their transmissions. This reselection halts the transmission process, until reconfiguration of the clusters is completed, causing interruptions of those transmissions that are in progress at the time of the halting. The latter transmissions need to be restarted by the corresponding MSNs, resulting in additional delays and retransmissions that may exceed the limit imposed by the low energy MSN –characteristic. Thus, architectural reconfigurations may cause increased MSN expiration rates, where such increase may be reduced if reconfiguration is not imposed in the case where the $\nu_u \rightarrow \nu_l$ rate shift in one or a few clusters does not change the number of clusters in the system.

The false alarms induced by the data rate monitoring algorithm in Section IV may occasionally dictate erroneous cluster reconfigurations and subsequent backbone network reductions, imposing elimination of some clusters, while the actual overall network data rate $\lambda_C$ has not been sufficiently reduced to necessitate such reduction. Implementation of cluster elimination will then result in traffic overloading of some clusters, with subsequent increase of the MSN expiration rate in them. The latter expiration rate increase must be evaluated against such increase due to the excessive routing and propagation delays in the non-reduced backbone network, for the subsequent adjustment of the data rate monitoring decision threshold, in harmony with the per cluster boundary rates $(\nu_l, \nu_u)$.

Given a member of the LSRAA class in Section IV, given the upper bound $\nu_u$ of the maintained data rate per cluster, given the MSN-expiration formula in (1), a specific value of the MSN expiration rate is induced. When the latter value is subtracted from $\nu_u$, the result should be bounded from below by the data rate lower bound $\nu_l$. In addition, if it is desirable that the MSN expiration rate, solely due to (1), be upper bounded, then, an upper bound to the value of $\nu_u$ is induced. As a conclusion, the pair $(\nu_l, \nu_u)$ of rate values are dependent on the MSN-expiration formula in (1).

We first simulated a two-cluster network, where the high priority MSNs may transmit via either one of the two cluster channels. We assumed limit Poisson traffics generated by both the regular and the high priority MSN populations, and we considered high priority traffics from 0% to 10% of the regular traffics. We used the expression in (1) to model MSN expiration, with various values of the $\beta$, $\eta$ and $\xi$ constants, while we adopted the LSRAA transmission protocol for two channels in Section IV. We simulated the two-cluster system and computed the resulting MSN expiration rates as well as the delay distributions of the successfully transmitted packets. In accordance with the results and the subsequent discussion in Section IV, we selected $\nu_l$=0.15 and $\nu_u$=0.325, with 0% to 10% high priority MSNs. We plot some indicative results in Figures 1 to 3. We observed that, as expected, for fixed constant values $\xi$, $\beta$ and $\eta$, the expiration rates of all three regular and high priority traffics converge to identical values, while the non-expired high priority MSNs are presented with a substantial delay advantage. When the constants $\beta$ and $\eta$ increased, the rejection rates increased uniformly, at the benefit of substantial decrease in the delays of the non-expired nodes.

## VI. Conclusions

We presented a dynamic approach to clustered wireless sensor networks, based on data rate monitoring. The approach is based on per cluster cumulative rate control, in contrast to other existing cluster formation schemes (LEACH-based) that ignore the composite rate factor; it is facilitated by a distributed and highly efficient rate monitoring algorithm. We modeled the expiration of the low power sensors and evaluated the approach for a system deploying a powerful stable random access transmission protocol that accommodates high priority data transmissions. The latter protocol is limited sensing and stable, accelerating effectively the high priority delays, while simultaneously accommodating low priority delays satisfactorily. In contrast, the existing alternative proposals for transmission protocols (e.g. Zigbee, IEEE802.15.4) are unstable with rapidly decreasing throughput, while they also do not accommodate high priority data.

The overall system proposed in this paper is robust and highly effective, controlling expiration rates and transmission delays with high efficiency.

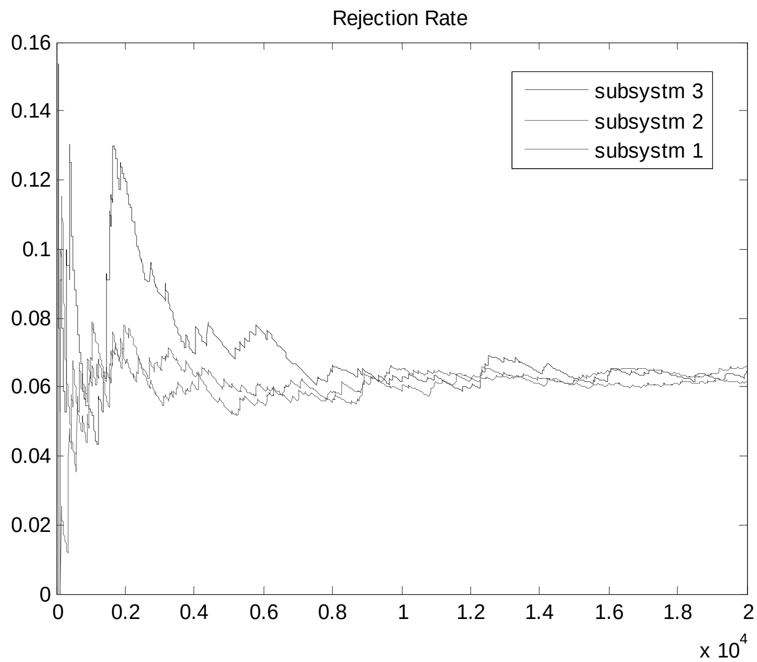


Figure1

Rejection rate for ξ =100, β=5, and η=10.

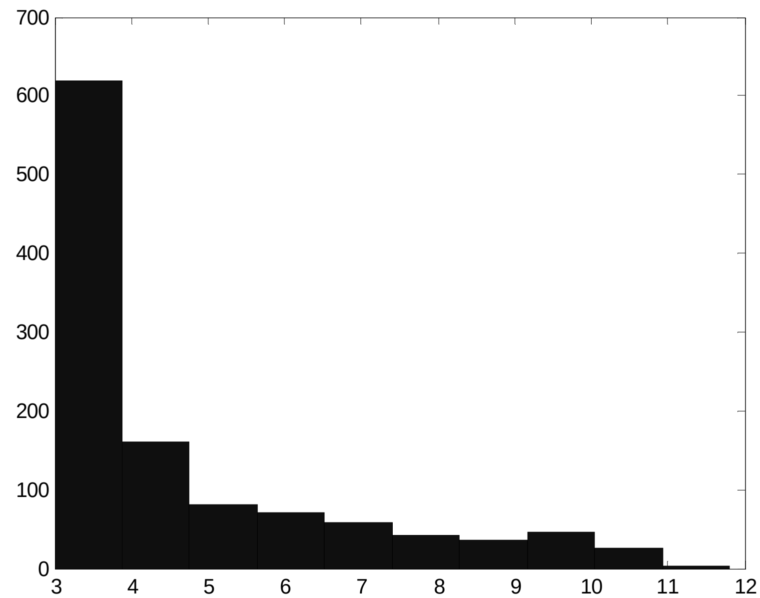


Figure 2

The delay distribution of non-expired nodes for the Subsystem 3 for $\xi$ =100, $\beta$=5, and $\eta$=10.

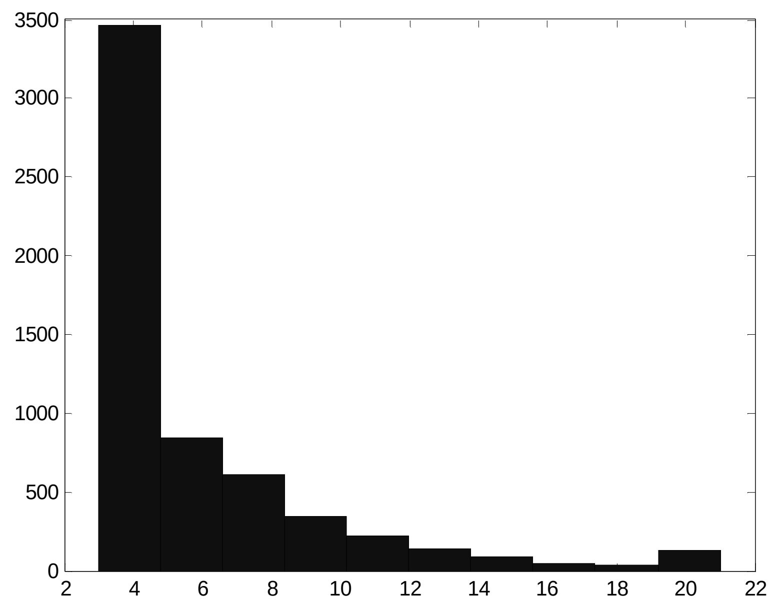


Figure 3

The delay distribution of non-expired nodes for the Subsystems 1and 2 for ξ =100, β=5, and η=10.

## References

[1] I. F. Akyildiz, W.Su, Y. Sankkarasubramaniam and E Cayirci, “Wireless Sensor Networks: A Survey”, *Computer Networks (Elsevier),* vol. 38, no. 4, pp. 393-422, 2002.

[2] S. Bandyopadhyay and E.J. Coyle, “An Energy Efficient Hierarchical Clustering Algorithm for Wireless Sensor Networks”, *IEEE INFOCOM 2003.*

[3] D. Bertsekas and R. Gallager, *Data Networks*, Englewood Cliffs, NJ, Prentice Hall, 1992.

[4] M. Bhardwaj and A.P. Chandrakasan, “Bounding the Lifetime of Sensor Networks via Optimal Role Assignments”, in Proc. *IEEE INFOCOM,* New York, Jun. 23-27, 2002, pp. 1587-1596.

[5] D. Blough and S. Paolo, “Investigating Upper Bounds on Network Life-Time Extension for Cell-Based Energy Conservation Techniques in Stationary Ad Hoc Networks”, in Proc. *ACM MobiCom,* Atlanta, GA, Sep. 23-28, 2002, pp. 183-192.

[6] A. T. Burrell, P. Papantoni-Kazakos, et. al, “Dynamic Capacity Allocation and Hybrid Multiplexing Techniques For ATM Wireless Networks”, *ACM/Springler Wireless Networks (WINET)*, 1998, Vol. 3, pp. 307-316.

[7] A. T. Burrell and P. Papantoni-Kazakos, “Wireless Networks in Hybrid Topologies: Signaling, Transmission, and Dynamic Capacity Allocation Viewed in an Integrated Fashion,” *ITC Mini-Seminar Performance Modeling and Design of Wireless/PCS Networks*, Cambridge, MA, October 3, 1996.

[8] A. T. Burrell and P. Papantoni-Kazakos, “Extended Sequential Algorithms for Detecting Changes in Acting Stochastic Processes”, *IEEE Trans. System, Man and Cybernetics,* Vol. 28, no. 5, Sept. 1998, pp. 703-710.

[9] A. T. Burrell, D. Makrakis, and P. Papantoni-Kazakos, "Traffic Monitoring for Capacity Allocation of Multimedia Traffic in ATM Broadband Networks", *Telecommunication Systems*, Vol. 9, July 1998, pp. 173-206.

[10] A.T. Burrell and P. Papantoni-Kazakos, "On-Line Learning and Dynamic Capacity Allocation in the Traffic Management of Integrated Services Networks”, *European Transactions on Telecommunications,* Special Issue on Architectures, Protocols, and Quality of Service for the Internet of the Future, Vol. 10, No. 5, March/April 1999, pp. 202-214.

[11] A. T. Burrell and P. Papantoni Kazakos, "Performance Monitoring in Sensor Networks". *ITNG 2009*, April 27-29, 2009, Las Vegas, Nevada.

[12] A.T.Burrell and T. Papantoni, "A Class of Limited Sensing Random Access Algorithms with Resistance to Feedback Errors and Effective Delay Control", *Journal of Communications and Networks,* 2006, Vol. 8, No. 1.

[13] J.H. Chang and L. Tassiulas, "Routing for Maximum System Lifetime in Wireless Ad Hoc Networks", in Proc. *37$^{th}$ Annual Allerton Conference on Communications, Control and Computing,* Monticello, IL, Sep. 1999, vol. 1, pp. 22-31.

[14] J.H. Chang and L. Tassiulas, "Energy Conserving Routing in Wireless Ad Hoc Networks", in Proceedings of *IEEE INFOCOM*, Tel Aviv, Israel, Mar. 26-30, 2000, pp. 22-31.

[15] Y. T. Hou, Y. Shi and H.D. Sherali, "On Node Lifetime Problem for Energy-Constrained Wireless Sensor Networks", *ACM/Springer Mobile Netw.Applicat.,* vol. 10, no. 6, pp. 865-878, Dec. 2005.

[16] Y.T. Hou, Y. Shi, J.H. Reed and K. Sohraby, "Flow Routing for Variable Bit Rate Source Nodes in Energy-Constrained Wireless Sensor Networks", in Proc. *IEEE International Conference on Communications,* Seoul Korea, May 16-20, 2005, pp. 3057-3062.

[17] Y. T. Hou, Y. Shi and H.D. Sherali, "Rate Allocation and Network Lifetime Problems for Wireless Sensor Networks", *IEEE Trans. Networking,* vol. 16, no. 2, April 2008, pp. 321-334.

[18] M. Kaplan, "A Sufficient Condition for Nonergodicity of a Markov Chain", *IEEE Transactions on Information Theory*, vol. 25, 1979, pp. 470-471.

[19] D. Kazakos and P. Papantoni-Kazakos, *Detection and Estimation,* Computer Science Press, New York, NY, 1990.

[20] H. Luss and D.R. Smith, "Resource Allocation among Competing Activities: A Lexicigraphic Minimax Approach". *Operations Research Letters,* vol. 5, no. 5, Nov. 1986, pp. 227-231.

[21] S. P. Meyn and R.L. Tweedie, *Markov Chains and Stochastic Stability,* Springer-Verlag, 1993.

[22] P. Papantoni-Kazakos, "Multiple Access Algorithms for a System with Mixed Traffic: High and Low Priority," *IEEE Trans. Comm*., March 1992, Vol. 40, pp. 541-555.

[23] P. Papantoni Kazakos and A.T. Burrell "The Implementation of Dynamic Rate Allocation in Sensor Networks" *Journal of Intelligent and Robotic Systems,* 2010, Vol. 58, pp. 211-238.

[24] P. Papantoni-Kazakos, H. Delic, M. Paterakis, and M. Liu, "Transmission Algorithms for a Multi-Channel Packet Radio System with Priority Users," *International Journal of Digital and Analog Communication Systems (IJDACS) John Wiley & Sons*, October-December 1993, Vol. 6, No. 4, pp. 193-212.

[25] A. Papoulis and S.U. Pillai, *Random Variables and Stochastic Processes,* 4th Edition, McGraw-Hill, New York, NY, 2002.

[26] M. Paterakis and P. Papantoni-Kazakos, "A Simple Window Random Access Algorithm with Advantageous Properties," *IEEE Trans. Inform. Th.,* September 1989, Vol. 35, pp. 1124-1130.

[27] V. Rodoplu and T.H. Meng, "Minimum Energy Mobile Wireless Networks", *IEEE J. Sel. Areas Commun.,* vol. 17, no. 8, Aug. 1999, pp. 1333-1344.

[28] K. Sohrabi, J. Gao, V. Ailawadhi and G. Pottie, "Protocols for Self-Organizing of a Wireless Sensor Netrwork", *IEEE Pers. Communications Magazine,* vol. 7, Oct. 2000, pp. 16-27.

[29] V. Srinivasan, P. Nuggehalli, C.F. Chiasserini and R. Rao, "Cooperation in Wireless Ad Hoc Networks", in Proc. *IEEE INFOCOM,* San Francisco, CA, Mar. 30-Apr. 3 2003, pp. 808-817.

[30] R. Wattenhofer, L. Li, P. Bahl and Y.M. Wang, "Distributed Topology Control for Power Efficient Operation in Multihop Wireless Ad Hoc Networks", in Proc. *IEEE INFOCOM,* Snchorage, AK, Apr. 22-26, 2001, pp. 1388-1397.