01 Jan 2005

# Predictive Congestion Control MAC Protocol for Wireless Sensor Networks

Maciej Jan Zawodniok
*Missouri University of Science and Technology*, mjzx9c@mst.edu

Jagannathan Sarangapani
*Missouri University of Science and Technology*, sarangap@mst.edu



## Recommended Citation

M. J. Zawodniok and J. Sarangapani, "Predictive Congestion Control MAC Protocol for Wireless Sensor Networks," *Proceedings of the International Conference on Control and Automation, 2005. ICCA '05*, Institute of Electrical and Electronics Engineers (IEEE), Jan 2005.
The definitive version is available at https://doi.org/10.1109/ICCA.2005.1528114

2005 International Conference on Control and Automation (ICCA2005)
June 27-29, 2005, Budapest, Hungary



# Predictive Congestion Control MAC Protocol for Wireless Sensor Networks[1]

*Maciej Zawodniok, IEEE Member and Sarangapani Jagannathan, IEEE Senior Member

[1]*Abstract*— **Available congestion control schemes, for example transport control protocol (TCP), when applied to wireless networks results in a large number of packet drops, unfairness with a significant amount of wasted energy due to retransmissions. To fully utilize the hop by hop feedback information, a suite of novel, decentralized, predictive congestion control schemes are proposed for wireless sensor networks in concert with distributed power control (DPC). Besides providing energy efficient solution, embedded channel estimator in DPC predicts the channel quality. By using the channel quality and node queue utilizations, the onset of network congestion is predicted and congestion control is initiated.**

**Stability of the hop by hop congestion control is demonstrated by using a Lyapunov-based approach. Simulation results show that the proposed schemes result in fewer dropped packets than a network without the hop-by-hop congestion control, better fairness index and network efficiency, higher aggregate throughput, and smaller end-to-end delays over the other available schemes like IEEE 802.11 protocol.**

## I. Introduction

DUE to constraints imposed on energy, memory and bandwidth in wireless sensor networks (WSN), energy-efficient data transmission protocols are being developed. Congestion is quite common in wireless networks. Network congestion occurs either when offered load exceeds available capacity or the link bandwidth is reduced due to fading channels. Network congestion causes channel quality to degrade and loss rates rise, leads to packets drops at the buffers with increased delays, energy wastage, and requires retransmissions. Moreover, traffic flow will be unfair for nodes whose data has to traverse a larger number of hops.

Normally data from a cluster head in the sensor networks are forwarded on a multi-hop basis through the network towards a single point of destination. At some intermediate nodes and the base station, a large amount of traffic from several sensor cluster heads aggregate causing heavy congestion. As a result, a significant number of packets are dropped causing retransmissions. This considerably reduces the performance and lifetime of the network. A congestion control mechanism is needed in order to balance the load, to prevent packet drops and to avoid network deadlock.

Rigorous work has been done in wired networks in terms of end-to-end congestion control [5]. Though there are several advantages to an end-to-end congestion control scheme this approach will slowly reacts to congestion since the onset of congestion has to be propagated first towards the destination and then back towards the source nodes where the transmission rate is adjusted. This is unacceptable in WSN due to increased energy consumption and reduced lifetime of the network besides dropped packets. Hence, hop-by-hop congestion control, which reacts to congestion faster than an end-to-end scheme, is preferred to minimize packet losses.

The CODA protocol proposed in [3] uses both hop-by-hop and end-to-end congestion control schemes. However, CODA allows packet drops at nodes toward sources from the congested node to minimize the effects of congestion at the expense of retransmissions and energy wastage. On the other hand, the Fusion scheme from [2] uses both channel and queue utilizations to assess congestion. Similar to CODA, Fusion uses a threshold for detecting the onset of congestion. However, thresholding is not a quick strategy due to the difficulty in determining the thresholds. Additionally in CODA and Fusion protocols, a node uses a broadcast message to indicate the occurrence of congestion to its neighbors. Though this is quite interesting, this method does not guarantee that the source nodes will receive the onset of congestion occurring inside the network.

## II. PROPOSED METHODOLOGY

The network congestion, as shown in Fig.1, occurs when either the incoming traffic (received and generated) exceeds the capacity of the outgoing link or link bandwidth drops due to channel fading caused by path loss, shadowing and Rayleigh fading. The latter one is common to wireless networks. In this paper, an efficient control scheme will be designed to proactively react to onset of congestion by monitoring and predicting queue utilizations along with channel state information. Additionally, the schemes ensure weighted fairness.

---

[1] - Research Supported by NFS ECS#0296191.

* Maciej Zawodniok and Sarangapani Jagannathan are with the Department of Electrical and Computer Engineering, University of Missouri-Rolla Rolla, Missouri 65409. (corresponding author phone (573)-202-0378; e-mail: mjzx9c@umr.edu; sarangap@umr.edu).

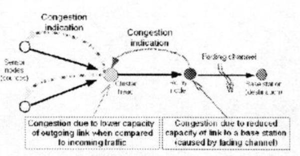

Fig. 1. Congestions in wireless sensor networks.

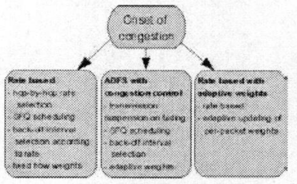

Fig. 2. Congestions in wireless sensor networks

### A. Performance Metrics

Onset of congestion causes packets to be dropped due to buffer overflows. Packets dropped at the intermediate nodes will cause low network throughput and a decrease in energy-efficiency. Consequently, the total number of packets dropped at the intermediate nodes will be considered as a metric for the designed protocol. Energy efficiency measured as a number of bits transmitted per joule will be used as the second metric. The network efficiency will be measured as the user throughput as a percentage of a channel bandwidth at the base station. Finally, weighted fairness will be used an additional metric, since congestion can cause unfair handling of flows. Formally, the weighted fairness is defined in terms of fair allocation of resources as

$$\left| \frac{W_f(t_1,t_2)}{\varphi_f} - \frac{W_m(t_1,t_2)}{\varphi_m} \right| = 0 \tag{1}$$

where $f$ and $m$ are considered flows, $\varphi_f$ is the weight of flow $f$ and $W_f(t_1,t_2)$ is the aggregate service (in bits) received by it in the interval $[t_1,t_2]$.

### B. Predictive Congestion Strategy

To predict the onset of congestion, the proposed schemes use queue utilization at each node along with the power required to transmit the packet under the current channel conditions that is provided by the distributed transmitter power control (DPC) algorithm. The queue utilization indicates that there is significant amount of incoming traffic flowing in when compared to the outgoing rate.

On the other hand, the wireless network faces fading problems. During fading, the available bandwidth is reduced and the outgoing rate will be lowered. Consequently, input and output buffers will accumulate the incoming traffic indicating the occurrence of congestion. The channel fading is estimated by using the feedback information from the DPC protocol [1] through the power value required for the next packet transmission. The DPC algorithm predicts the channel state for the next transmission and calculates the power required. If this power exceeds the maximum threshold, then channel is considered bad and congestion control scheme can initiate back-off. Hence, power information can be utilized to predict the onset of congestion due to fading channels.

### C. Congestion Mitigation

When the onset of congestion is detected different strategies can be applied to avoid it. We propose three schemes, with a goal to prevent and to minimize the effect of congestion while ensuring weighted fairness. Fig. 2 briefly characterizes these schemes. In the next few sections, we present details about each of these solutions.

### D. Distributed Power Control (DPC)

The goal of the DPC is to maintain a target signal to noise ratio (SNR) threshold for each network link while the transmitter power is adjusted autonomously at each link so that the least possible power is consumed in the presence of channel uncertainties. The DPC algorithm used in this paper is capable of predicting changes in the channel state. Moreover, the faster convergence of the hop by hop DPC algorithm [1] allows its applicability in congestion control. The information about the channel state is utilized by the congestion control scheme.

**Remark 1:** Though DPC scheme improves throughput and energy efficiency when compared to 802.11, it addresses neither fairness nor network congestion mitigation, which is the central focus of this paper.

## III. RATE-BASED CONGESTION CONTROL USING BACK-OFF

The first congestion control scheme is summarized as follows:

1) The buffer occupancy along with the power signal required to overcome the channel condition from the DPC at a link's receiver will be used to detect an onset of congestion. The rate selection algorithm is executed to determine the appropriate rate.
2) The bandwidth (or rate) is allocated for the flows according to the flow weights. This ensures weighted fairness in terms of bandwidth allocation among the neighboring nodes.
3) Packets at each node are scheduled by using the start time fair queuing (SFQ) algorithm via flow assigned weights to ensure the fair handling of the packets.
4) The DPC and rate information is communicated by the receiver to the transmitter on every link.
5) At the transmitter, a back-off interval is selected based on the assigned outgoing rate.

The adaptive rate selection scheme will control congestion on a hop-by-hop basis. The back-pressure mechanism will propagate information about congestion to the sources sending traffic. Consequently the congestion is alleviated by designing suitable back off intervals and by controlling the flow rates to prevent buffer overflowing at the congested nodes. Next, we describe the rate and back-off selection algorithms in detail.

*A. Rate selection*

The rate selection takes into account the buffer occupancy and a target outgoing rate. The target rate indicates what the incoming rate should be. Next, the selection of the incoming rate is described.

**Buffer occupancy and Rate Selection**

Consider buffer occupancy at a particular node. The following equation describes the change in buffer occupancy in terms of incoming and outgoing traffic

$$q_i(k+1) = q_i(k) + [u_i(k) - f_i(k)]T \tag{2}$$

where $T$ is the measurement interval, $q_i(k)$ is the buffer occupancy of node 'i' at time instant $k$, $u_i(k)$ is a regulated (incoming) traffic rate, and $f_i(k)$ is an outgoing traffic rate dictated by a node downstream on the path to the destination. The regulated incoming traffic rate $u_i(k)$ have to be calculated and propagated as a feedback to a node *(i-1)* located on the path to the source, which is nothing but the outgoing rate for this upstream node $f_{i-1}(k)$.

Select the desired buffer occupancy at node $i$ to be $q_{id}$. Then, buffer occupancy error defined as $e_i(k)=q_i(k)-q_{id}$ can be expressed as

$$e_i(k+1) = q_i(k) + T \cdot u_i(k) - T \cdot f_i(k) - q_{id} \tag{3}$$

Now, define the traffic rate input, $u_i(k)$ as

$$u_i(k) = T^{-1}\left(q_{id} - q_i(k) + T \cdot f_i(k) + k_v \cdot e_i(k)\right) \tag{4}$$

where $k_v$ is a gain parameter. In this case, an error in the time instant $k+1$ becomes

$$e_i(k+1) = k_v e_i(k) \tag{5}$$

The error will become zero as $k \to \infty$, provided $0 < k_v < 1$.

The rate selected by the above algorithm is subject to fading channels. To mitigate congestion due to channel fading, the selected rate has to be reduced when the transmission power calculated by the DPC scheme exceeds the transmitter node's capability (greater than maximum transmission power).

**Rate Propagation**

This total incoming rate is then divided among the upstream nodes proportionally to the sum of flow weights passing through a given node $j$ as

$$u_{ij}(k) = u_i(k) \cdot \sum_{n}^{\text{flows at } j^{th} \text{ node}} \varphi_n \Big/ \sum_{m}^{\text{flows at } i^{th} \text{ node}} \varphi_m \tag{6}$$

where $u_{ij}(k)$ is the rate allocated for a transmitting node $j$ at receiving node $i$, $u_i(k)$ is the rate selected for all incoming flows at *i-th* node, and $\varphi_n$, $\varphi_m$ are weights of the *n-th* and *m-th* flows. Next, the selected rate $u_{ij}(k)$ is communicated to the upstream node *j to mitigate congestion.*

This feedback continues recursively to the nodes upstream from the congested link so that they will also reduce transmission rates, and thus prevent overflowing buffers. This rate can be achieved in practice by either selecting the modulation scheme appropriate to the rate or by selecting the back off interval of the nodes. Here we select the back off intervals dynamically using local information.

*B. Data dissemination*

Packets at the receiver are first scheduled using SFQ algorithm. Weights that correspond to the packets are used to build a transmission schedule. This algorithm ensures weighted fairness among the flows passing a given node.

Then, the packet to be transmitted is sent to the MAC layer, where the back-off interval is applied. The back-off interval is selected to meet the target rate $f_j(k)=u_{ij}(k)$ that was received from the next-hop node. The selection algorithm is presented next.

*C. Back-off interval selection algorithm*

Since multiple nodes in a wireless sensor network compete to access the shared channel, back-off interval selection plays a critical role in deciding which node gains access to the channel. Thus, the proposed rate selection is implemented by suitably modifying the back-off intervals of the nodes around the congested node to achieve the desired rate control.

In the case of contention based protocols, due to multiple nodes are competing for the channel it is difficult to select an appropriate back-off interval. For a node, a relationship between the transmission rate and its back-off interval depends upon the back-off intervals of all nodes within a sensing range of a transmitting node. To exactly calculate this relationship, a node needs to know the back-off intervals of all its neighbors. Such a scenario would require a large traffic overhead to communicate the intervals. Moreover, the communication delay will render the calculations ineffective making the congestion control scheme development difficult. Therefore, we propose using a distributed and predictive algorithm to estimate back-off intervals, such that a target rate is achieved.

**Goal**: Select back-off interval $BO_i$ at *i-th* transmitting node such that the actual throughput meets the desired outgoing rate $f_i(k)$.

To obtain the back-off intervals, we consider inverse of the back-off interval, which we call a virtual rate $VR_i$

$$VR_i = 1 / BO_i \tag{7}$$

where $VR_i$ is the virtual rate at *i-th* node, and $BO_i$ is the corresponding back-off interval. However, the virtual rate is not equal to the actual rate; instead, the virtual rate is proportional to the actual rate.

The back-off scheduling algorithm schedules the packet transmissions according to a back-off interval. The interval is counted-down when a node does not detect any transmission, and pauses otherwise. Consequently, node will gain access to the channel proportional to its virtual rate (7) and inversely proportional to the sum of virtual rates of its neighbors. The actual rate of an *i-th* node is a fraction of the channel bandwidth $B(t)$ defined as

$$R_i(t)=\frac{B(t)\cdot VR_i(t)}{\sum_{l\in S_i} VR_l(t)}=\frac{B(t)\cdot VR_i(t)}{TVR_i(t)} \tag{8}$$

where $TVR_i$ is the sum of all virtual rates for all neighbor nodes.

Since the scheme considers only a single modulation scheme, bandwidth ($B$) is assumed time-invariant until the back-off interval is selected. We assume that the total bandwidth is constant as long as communication is possible on a link (above predefined SNR threshold). However, when the severe fading occurs, the bandwidth will drop to zero. In such a case, selecting any back-off interval is pointless since successful transmission is impossible.

Differentiating equation (8) to get

$$\dot{R}_i(t)=\frac{B}{TVR_i^2(t)}\begin{bmatrix}\dot{VR}_i(t)\,TVR_i(t)\\ -VR_i(t)\,T\dot{VR}_i(t)\end{bmatrix} \tag{9}$$

To transform the differential equation into the discrete domain, Euler's formula is used as follows

$$R_i(k+1)-R_i(k)=\frac{B}{TVR_i^2(t)}\begin{bmatrix}(VR_i(k+1)-VR_i(k))TVR_i(k)\\ -VR_i(k)(TVR_i(k+1)-TVR_i(k))\end{bmatrix} \tag{10}$$

After transformation equation (10) can be expressed as

$$R_i(k+1)=\left[\frac{B\cdot VR_i(k+1)}{TVR_i(k)}\right]+R_i(k)\left[1-\frac{TVR_i(k+1)}{TVR_i(k)}\right] \tag{11}$$

Applying (8) to equation (11) to get

$$R_i(k+1)=\left[\frac{R_i(k)VR_i(k+1)}{VR_i(k)}\right]+R_i(k)\left[1-\frac{TVR_i(k+1)}{TVR_i(k)}\right] \tag{12}$$

Now, define

$$\alpha_i(k)=1-TVR_i(k+1)/TVR_i(k), \tag{13}$$

$$\beta_i(k)=R_i(k)/VR_i(k), \tag{14}$$

and

$$v_i(k)=VR_i(k+1)=1/BO_i(k+1)\,. \tag{15}$$

The variable $\alpha$ describes a variation of back-off intervals of flows at the neighbor nodes, from time instant $k$ to $k+1$. This variation is due to congestion caused by traffic and fading channels. Since this information is not available locally, it is considered an unknown parameter, and thus estimated by the algorithm. The $\beta$ parameter is the ratio between actual rate and the used virtual rate at time instant k, and can be easily calculated. The term $v_i$ is the back off interval that has to be calculated.

Now, equation (12) can be written as

$$R_i(k+1)=R_i(k)\alpha_i(k)+\beta_i(k)v_i(k) \tag{16}$$

Equation (16) indicates that the achieved rate at the instant, $k+1$, depends on the variations of back-off intervals in the neighboring nodes. Since $\alpha$ is unknown, it has to be estimated. Select the back-off interval as

$$v_i(k)=(\beta_i(k))^{-1}\left[f_i(k)-R_{ij}(k)\alpha_i(k)+K_v e_{Ri}(k)\right] \tag{17}$$

where $\hat{a}_i(k)$ is estimate of $a_i(k)$, and $e_{Ri}(k)=R_i(k)-f_i(k)$ is defined as an throughput error. In this case, the throughput errors are expressed as

$$\begin{aligned}e_{Ri}(k+1)&=K_v e_{Ri}(k)+\alpha_i(k)R_i(k)-\hat{\alpha}_i(k)R_i(k)\\&=K_v e_{Ri}(k)+\tilde{\alpha}_i(k)R_i(k)\end{aligned} \tag{18}$$

where $\tilde{\alpha}_i(k)=\alpha_i(k)-\hat{\alpha}_i(k)$ is the error in estimation.

The throughput error of the closed-loop system is driven by the error in back-off interval selection of the neighbors. If these uncertainties are properly estimated a suitable back-off interval is selected. If the error in back-off interval selection tends to zero, equation (18) reduces to $e_{Ri}(k+1)=K_v\cdot e_{Ri}(k)$. In the presence of error in estimation, only a bound on the error in back-off interval selection can be shown. In other words, the congestion control scheme will ensure that the actual throughput is close to its target value.

***Theorem 1***: Given the back-off selection scheme above with uncertainties that are estimated accurately (no estimation error), if the back-off interval is updated as (17), then the mean estimation error along with the mean error in back-off interval converges to zero asymptotically, if the parameters updates are taken as

$$\hat{a}_i(k+1)=\hat{\alpha}_i(k)+\sigma\cdot R_i(k)\cdot e_{Ri}(k+1) \tag{19}$$

provided

$$\sigma\|R_i(k)\|^2<1 \tag{20}$$

$$K_{vmax}<1/\sqrt{\delta} \tag{21}$$

where $\delta=1/\left[1-\sigma*\|R_i(k)\|^2\right]$, $K_{vmax}$ is the maximum singular values of $K_{vmax}$, and $\sigma$ is the adaptation gain.

Consider the closed loop throughput error system with estimation error, $\varepsilon(k)$, as

$$e_{Ri}(k+1)=K_v e_{Ri}(k)+\alpha_i(k)R_i(k)+\varepsilon(k) \tag{22}$$

***Theorem 2***: Assume the hypothesis as given in Theorem 1, with the uncertainties estimated by

$$\hat{a}_i(k+1)=\hat{\alpha}_i(k)+\sigma\cdot R_i(k)\cdot e_{Ri}(k+1) \tag{23}$$

with $\varepsilon(k)$ is the error in estimation which is considered bounded above $\|\varepsilon(k)\|\le\varepsilon_N$, with $\varepsilon_N$ a known constant. Then the mean error in throughput and the estimated parameters are bounded provided (20) and (21) hold.

### D. Implementation

The proposed scheme has been implemented in NS-2 simulator and evaluated against the DPC [1] and the 802.11 protocols. Since, all the information for back-off calculation is available locally, only the rate information that a transmitter node has to send has to be feedback to upstream nodes. This feedback is incorporated into the MAC frames. The calculations of the rate and back-off interval are performed periodically at every 0.5 second.

### E. Tree topology results

The tree topology, which is typical for a sensor network, is used for the simulations where the sources represent cluster heads, and the sink is a base station collecting the sensor data. Traffic accumulates near the destination node causing congestion at the intermediate nodes. In the simulations, the traffic consists of five flows, which had been simulated for two cases: with the same weights equal to 0.2, and with weights equal to 0.4, 0.1, 0.2, 0.2, and 0.1 respectively. All the sources generate equal traffic that exceeds channel capacity. The initial rates for each flow have been assigned proportional to the weight such that they saturate the radio channel. The other parameters that were used include 2Mbps channel with path loss, shadowing and Rayleigh fading with AODV routing protocol. The queue limit is set to 50 with the packet size taken as 512 bytes. The SFQ scheduling algorithm was used to ensure fairness among the flows passing at a given node. It uses the assigned weights to schedule packets for transmission and the weights are not updated. The proposed scheme is compared with others.

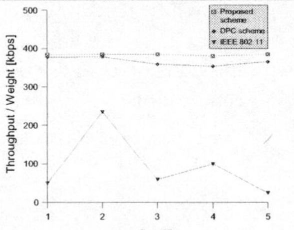


Fig. 3 Performance for tree topology

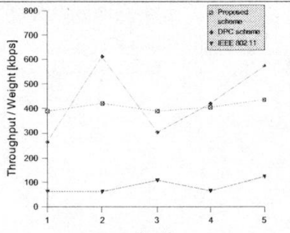


Fig. 4 Performance for tree topology with varying weights

**Remark 2:** Since the DPC protocol does not address fairness or congestion, the proposed scheme is compared with that of DPC to observe its impact.

Fig. 3 and 4 depict the throughput/weight (normalized weights) ratio for each flow. Ideally, the throughput over the initial weight ratio plot should be a straight line parallel to the x-axis for fair scheduling schemes. It is visible that the proposed protocol results in a fair allocation of bandwidth compared to the DPC and to the 802.11 MAC protocol. The fairness of the proposed protocol is maintained also for the case of variable weights assigned to the flows, as observed in Fig. 4. The DPC protocol achieves very good fairness in the case of identical weights. However, the DPC fails when the weights are different, since there is no mechanism to vary allocation of the channel resources to flows depending on their weights. In all, the proposed scheme achieves better fairness compared to DPC and 802.11 protocols.

In terms of throughput, the 802.11 performs the least since it cannot handle increased number of collisions that occurs in a congested network. It is important to note that due to CSMA/CA nature the packets collide and channel becomes idle. Therefore, throughput for 802.11 is less. On the other hand, the DPC detects the idle channel after collisions and resumes transmission sooner than the 802.11 protocol. In the case of the proposed scheme, the throughput is further increased since it mitigates congestion by preventing packet losses. Consequently, the proposed algorithm outperforms other protocols.

TABLE 1
DELAY, THOUGHPUT AND ENERGY-EFFICIENCY

| Protocol | Average delay [s] | Network efficiency | Throughput [kbps] | Energy Effciency *[kbit / joule]* |
|---|---|---|---|---|
| ***Proposed*** | 0.8 | 20.05% | 400.99 | 13.05 |
| 802.11 | ----- | 3.89% | 77.86 | 3.23 |
| DPC | 1.06 | 18.43% | 368.55 | 11.79 |

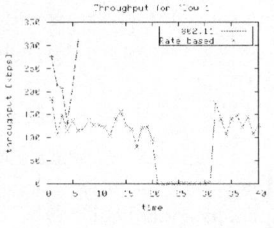


Fig. 5 Throughput for flow 1

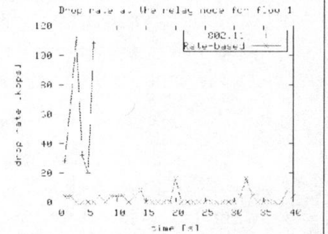


Fig. 6 Drop rate at the relay node

Table 1 summarizes overall performance of the considered protocols. The 802.11 protocol achieves low throughput and low energy efficiency since it is susceptible to congestions. The network becomes deadlocked since frequent collisions prevent the 802.11 protocol from successfully transmitting packets from the source node, as observed in Fig. 6. As a result, the average delay for 802.11 is significantly high (not shown) compared to other schemes. The other protocols, DPC and rate–based with DPC algorithm provides a better performance in terms of throughput and energy-efficiency. However, the congestion control used in the proposed scheme further increases the throughput and energy-efficiency of the proposed scheme when compared with the DPC alone without congestion control. This improvement is due to lower drop rate and minimal energy wastage because of flow control. Fig. 5 and 6 present throughput and drop rate for the flow one. It can be noticed, during the interval of 21st to 31st seconds, that the link from relay node to destination experiences severe fading and no transmission is possible. The proposed protocol prevents overflowing buffer at the relay node by sending backpressure indication to the sources and preventing them from overflowing the relay node. Thus, no increase in drop rate is observed for the rate-based protocol.

## IV. FAIR SCHEDULING BASED CONGESTION CONTROL

The second control scheme is based on the adaptive distributed fair scheduling (ADFS). The ADFS follows the weighted fairness criterion defined in [6]. The main contribution in this scheme is the dynamic adaptation of weights with network state defined, as a function of delay experienced, number of packets in the queue and the previous weight of the packet. However, the ADFS does not address congestion issues in a proactive manner. The buffer occupancy and channel estimation from DPC, as mentioned in the previous section, is propagated by hop-by-hop basis towards the sources, allowing these nodes to reduce the incoming rate and prevent buffer overflows. The fair scheduling ensures proportional allocation of the resources.

The initial weights are selected by using the user-defined quality of service (QoS). In fact, analytical results are included in [7] to demonstrate the throughput and end-to-end delay bounds in contrast with the existing literature. The NS simulation results indicate that the proposed scheme renders a fair protocol for wireless sensor networks even in the presence of congestion due to fading channels. The weights are updated as given next.

### A. Dynamic Weight Adaptation

To account for the changing traffic and channel conditions that affect the fairness and end-to-end delay, the weights for the flows are updated dynamically. The actual weight for the $i^{th}$ flow, $j^{th}$ packet denoted by $\hat{\varphi}_{ij}$ , is updated as

$$\hat{\varphi}_{ij}(k+1) = \alpha.\hat{\varphi}_{ij}(k) + \beta.E_{ij} , \tag{24}$$

where $\hat{\varphi}_{ij}(k)$ is the previous weight of the packet, $\alpha$ and $\beta$ are design constants, $\{\alpha , \beta\} \in [-1,1]$, and $E_{ij}$ is defined as

$$E_{ij} = e_{ij,queue} + 1/e_{ij,delay} , \tag{25}$$

where $e_{ij,queue}$ is the error between the expected length of the queue and the actual size of the queue and $e_{ij,delay}$ is the error between the expected delay and the delay experienced by the packet so far. According to (25), when queues buildup, the packet weights will be increased to clear the backlog. Similarly, with large end-to-end delays, the packet weights will be assigned high so that the nodes can service these packets sooner. Note that the value of $E_{ij}$ is bounded due to finite queue length and delay, as packets experiencing delay greater than the delay error limit will be dropped.

### B. MAC Layer - Dynamic Back-off Intervals

In order to achieve global fairness, the nodes must access the channel in a fair manner. This algorithm calculates the back-off interval by using the packet weight from (25). The back-off interval is updated at each node due to weight adaptation (25). Back-off interval, $BI_{ij}$ , for $i^{th}$ flow $j^{th}$ packet with length $L_{ij}$ and weight $\varphi_{ij}$ is defined as

$$BI_{ij} = \rho * SF * L_{ij} / \varphi_{ij} , \quad for\ B(k) = 1,$$
$$BI_{ij} = lar , \quad for\ B(k) = 0 \tag{26}$$

where $SF$ is the scaling factor, $\rho$ is a random variable with mean one, *lar* is a large value of the back-off interval and B(k) is the variable that is used to identify whether there is an onset of congestion or not. Equation (26) indicates that when the onset of congestion is detected, back-off intervals are set at a large value (*lar*) to free the channel whereas under normal circumstances, the packet weights are utilized to select the back-off interval so that a fair allocation of the bandwidth is achieved.

## V. RATE-BASED CONGESTION CONTROL WITH WEIGHT UPDATING

The final solution is a fusion of the previous schemes. It uses rate-based algorithms to select rate and back-off intervals, and it will dynamically update weights for each packet as the adaptive and distributed fair scheduling scheme does. The weights are dynamically updated using (25). Consequently, this solution will adapt to a changing channel state. The weight update will ensure fair handling of all flows: the ones passing through the congested nodes, and the ones not experiencing congestion.

## VI. CONCLUSIONS

This paper presents a suite of predictive congestion control schemes. Simulation results show that the proposed schemes increase throughput, network efficiency and energy conservation. With addition of fair scheduling algorithm, we can guarantee desired quality of service (QoS) and weighted fairness for all flows even during congestion and fading channels. Final proposed scheme provides a hop by hop mechanism for throttling packet flow rate, which will help in mitigating congestion. The convergence analysis is demonstrated by using a Lyapunov-based analysis.

## REFERENCES


[1] M. Zawodniok and S. Jagannathan, "*A distributed power control MAC protocol for wireless ad hoc networks*", 2004, IEEE WCNC'04, Vol. 3, pp. 1915–1920, March 2004.

[2] Bret Hull, Kyle Jamieson and Hari Balakrishnan, "Mitigating congestion in wireless sensor networks", *ACM SenSys'04*, Baltimore (MD), USA, November 2004

[3] C.-Y. Wan, S. Eisenman and A. Campbell, "CODA: Congestion detection and avoidance in sensor networks", Proceedings of the ACM SenSys'03, pp. 266-279, Los Angeles (CA), USA, November 2003.

[4] Yung Yi and Sanjay Shakkottai, "Hop-by-hop congestion control over wireless mulit-hop networks, *Proceedings of IEEE INFOCOM '04*, Vol. 4, pp. 2548-2558, Hong-Kong, March 2004.

[5] S. Jagannathan, "End to end congestion control in high-speed networks", *Proceedings of the 27th IEEE LCN'02*, pp. 547-556, USA, 2002.

[6] S.J. Golestani, "A self-clocked fair queueing scheme for broadband applications", *IEEE INFOCOM'94 Networking for Global Communications*, June 1994, vol. 2, pp. 636 – 646.

[7] N. Regatte and S. Jagannathan, "Adaptive and distributed fair scheduling scheme for wireless adhoc networks", Proc. of the Word Wireless Congress, pp. 101-106, May 2004.