01 Jan 2007

Route Aware Predictive Congestion Control Protocol for Wireless Sensor Networks

Carl Larsen

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

C. Larsen et al., "Route Aware Predictive Congestion Control Protocol for Wireless Sensor Networks," *Proceedings of the IEEE 22nd International Symposium on Intelligent Control (2007, Singapore)*, Institute of Electrical and Electronics Engineers (IEEE), Jan 2007.
The definitive version is available at https://doi.org/10.1109/ISIC.2007.4450853

22nd IEEE International Symposium on Intelligent Control
Part of IEEE Multi-conference on Systems and Control
Singapore, 1-3 October 2007

Route Aware Predictive Congestion Control Protocol for Wireless Sensor Networks

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

***Abstract*—Congestion in wireless sensor networks (WSN) may lead to packet losses or delayed delivery of important information rendering the WSN-based monitoring or control system useless. In this paper a routing-aware predictive congestion control (RPCC) yet decentralized scheme for WSN is presented that uses a combination of a hop by hop congestion control mechanism to maintain desired level of buffer occupancy, and a dynamic routing scheme that works in concert with the congestion control mechanism to forward the packets through less congested nodes. The proposed adaptive approach restricts the incoming traffic thus preventing buffer overflow while maintaining the rate through an adaptive back-off interval selection scheme. In addition, the optimal routing scheme diverts traffic from congested nodes through alternative paths in order to balance the load in the network, alleviating congestion. This load balancing of the routes will even out the congestion level throughout the network thus increasing throughput and reducing end to end delay. Closed-loop stability of the proposed hop-by-hop congestion control is demonstrated by using the Lyapunov-based approach. Simulation results show that the proposed scheme results in reduced end-to-end delays.**

I. Introduction

Network congestion in wireless networks occurs when offered load exceeds available capacity or the link bandwidth is reduced due to fading channels. This causes channel quality to degrade and loss rates rise. It leads to packets drops at the buffers, increased delays, wasted energy, and required retransmissions. Additionally, in wireless sensor networks (WSN), bandwidth limitations due to channel fading and hardware limitations in terms of memory, processing, energy, and communication capacity of each node have a significant impact on scalability for network protocols, especially in a congested network. In the case of a routing protocol, the memory requirements for a wireless sensor node depend upon the number of traffic flows and buffer length for each flow. The former depends on how well the routing load is distributed among the nodes. For the latter, network congestion is the main source of increased buffer occupancy. Besides increased occupancy, network congestion will result in unfair handling of traffic flows which traverse either through congested area or through a significant number of hops thus reducing the performance and lifetime of the network.

Therefore, adaptive data transmission protocols are required to mitigate congestion resulting from fading channels and to balance the load preventing packet drops and network deadlock. Finally, an adaptive routing protocol that will balance a load between several nodes will further lower the memory and processing requirements of sensor nodes minimizing buffer overflows.

Rigorous work was reported in wired networks on end-to-end congestion control techniques [5]. In spite of several advantages in end-to-end control schemes, a hop-by-hop congestion control scheme reacts to congestion faster and is normally preferred to minimize packet losses in wireless networks. Other protocols [2][3][4] do not predict the onset of congestion in dynamic environments resulting from fading channels. Additionally, very few analytical results are presented in the literature in terms of guaranteeing the performance of available congestion control protocols.

On the other hand, certain researchers [6][7] have focused on the performance analysis of the back-off selection schemes for static environments since in CSMA/CA-based wireless networks, a back-off selection mechanism is used to provide simultaneous access to a common transmission medium and to vary transmission rates. Scheduling schemes [6][7][8][9] tend to vary the rate based on traffic type, However, these schemes lack the ability to adapt to the channel state, congestion level, and network size. By contrast, the proposed algorithm dynamically alters back-off intervals according to current network conditions, for instance the varying number of neighbor nodes and fading.

By combining the routing and hop by hop congestion control, the proposed scheme overcomes network congestion by selecting alternative route that avoids the congested nodes. In contrast, typical congestion control scheme assumes that the routes are fixed by routing scheme. While the autonomous routing scheme ignores congestion levels thus not taking advantage of alternative paths that may bypass the congested area. Additionally, the performance of this scheme is ensured through Lyapunov-based analysis.

II. Proposed Methodology

In this paper, we will minimize congestion by varying the back off intervals and finding alternate routes using

Research Supported by GAANN Fellowship through Department of Education, NSF I/UCRC on Intelligent Maintenance Systems, NSF ECCS#0633769, NSF IIP#0639182, and Intelligent Systems Center.

C. Larsen, Maciej Zawodniok and S. Jagannathan are with the Electrical and Computer Engineering Department, University of Missouri-Rolla, Rolla, MO 65409 USA. (e-mail: cvlgd8@umr.edu; maciej_zawodniok@ieee.org; sarangap@umr.edu).

knowledge of the channel state. Next, an overview of the proposed scheme is presented.

A. Overview of the Proposed Scheme

The scheme can be summarized in the following steps:

1) The *buffer occupancies* at the transmitter and receiver nodes along with the *transmitter power* required to overcome the channel state at the subsequent time interval will be used to detect an *onset of congestion*. The *rate selection algorithm* is then executed *at the receiver* to determine the appropriate rate (or available bandwidth) for the *predicted channel state*.
2) The available 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. The weights can be selected initially and held subsequently or updated over time.
3) The distributed power control (DPC) and rate information is communicated by the receiver to the transmitter for every link.
4) At the transmitter node, a back-off interval is selected by using the proposed scheme based on the *assigned outgoing rate*.

Remark 1: The feedback information is piggybacked to the ACK frame of the MAC protocol thus ensuring receipt. By contrast, the broadcast message used in CODA does not guarantee the receipt due to the lack of acknowledgement.

Remark 2: In this paper a single MAC data rate is considered without addressing the interlayer coordination and routing protocols. However, the mathematical analysis suggests that changes in routes and MAC data rates (bandwidth) will be accommodated by the outgoing traffic estimation algorithm. Embedded channel estimator in DPC [1] indicates the state of the wireless channel which is utilized to assess the onset of congestion.

B. OEDSR based Routing Scheme

In optimal energy delay subnetwork routing (OEDSR)[10][11], sub-networks are formed around an event/ fault and nodes wake up in the sub-networks while the nodes elsewhere in the network are in sleep mode. An appropriate percentage of nodes in the sub-network are elected as cluster heads (CHs) based on a cost metric composed of available energy and relative location to an event [10] in each sub-network. Once the CHs are identified and the nodes are clustered relative to the distance from the CHs, the routing towards the BS is initiated. Our scheme extends this protocol by adding a term, based on the measured queue utilization and outgoing flow rate, to the link cost factor. Since these are good predictors of congestion and queuing delay, the protocol will select less congested relay nodes which can provide lower delay. Next the proposed scheme is discussed.

III. Adaptive Congestion Control

The congestion is alleviated by a) designing suitable back off intervals for each node based on channel state and current traffic; and b) by controlling the flow rates of all nodes including the source nodes to prevent buffer overflowing. The rate and back-off selection algorithms are described next.

A. Rate Selection Based on Buffer Occupancy

The rate selection scheme takes into account the buffer occupancy and a target outgoing rate. The target rate at the next hop node indicates what the incoming rate should be. The selection of the incoming rate is described next.

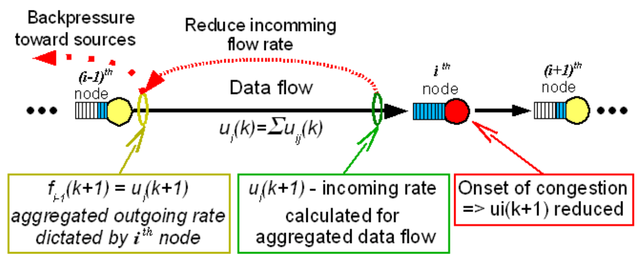

Fig. 1. Rate selection overview.

Consider buffer occupancy at a particular node i, as shown in Fig. 1. The change in buffer occupancy in terms of incoming and outgoing traffic at this node is given as

$$q_i(k+1)=Sat_p[q_i(k)+T\cdot u_i(k)-f_i(u_{i+1}(k))+d(k)] \qquad (1)$$

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, $d(k)$ is an unknown disturbance in traffic, $f_i(\bullet)$ represents an outgoing traffic which is dictated by the next hop node i+1 and is disturbed by changes in channel state, and Sat_p is the saturation function that represents the finite-size queue behavior. The regulated incoming traffic rates $u_i(k)$ have to be calculated and propagated as a feedback to the node i-1 located on the path to the source, which is then used to estimate the outgoing traffic for this upstream node $f_{i-1}(\bullet)$.

Select the desired buffer occupancy at node i to be q_{id}. Then, buffer occupancy error defined as $e_{bi}(k)=q_i(k)-q_{id}$ can be expressed using (1) as $e_{bi}(k+1)=q_i(k)+Tu_i(k)-f_i(u_{i+1}(k))+d(k)-q_{id}$. Next, the controller is introduced and its stability analysis is presented by using two different scenarios.

The outgoing traffic $f_i(\bullet)$ is unknown and has to be estimated. In such a case, we define the traffic rate input, $u_i(k)$ as

$$u_i(k)=Sat_p\left[T^{-1}\left(\hat{f}_i(u_{i+1}(k))+(k_{bv}-1)e_{bi}(k)\right)\right] \qquad (2)$$

where $\hat{f}_i(u_{i+1}(k))$ is an estimate of the unknown outgoing traffic $f_i(u_{i+1}(k))$. In this case, the buffer occupancy error at the time instant k+1 becomes

$$e_{bi}(k+1)=Sat_p\left(k_{bv}e_{bi}(k)+\tilde{f}_i(u_{i+1}(k))+d(k)\right) \qquad (3)$$

where $\tilde{f}_i(u_{i+1}(k)) = f_i(u_{i+1}(k)) - \hat{f}_i(u_{i+1}(k))$ represents the estimation error of the outgoing traffic.

The outgoing traffic function is estimated, using a traffic parameter vector θ, by $f_i(u_{i+1}(k)) = \theta \cdot f_i(k-1) + \varepsilon(k)$, where $f_i(k-1)$ is the past value of the outgoing traffic and the approximation error $\varepsilon(k)$ is assumed bounded by a known constant ε_N. Now, define traffic estimate in the controller as $\hat{f}_i(u_{i+1}(k)) = \hat{\theta}(k) f_i(k-1)$, where $\hat{\theta}_i(k)$ is the actual vector of traffic parameters, $\hat{f}_i(u_{i+1}(k))$ is an estimate of the unknown outgoing traffic $f_i(u_{i+1}(k))$, and $f_i(k-1)$ is the past value of the outgoing traffic.

Given the incoming rate selection scheme above with the back-off interval updated as (2) with the parameter θ_i updated as

$$\hat{\theta}_i(k+1) = \hat{\theta}_i(k) + \lambda \cdot f_i(k) e_{bi}^T(k+1) - \Gamma \left\| I - f_i^2(k) \right\| \hat{\theta}_i(k) \tag{4}$$

where $\Gamma > 0$ is a design parameter. Then the mean error $e_{bi}(k)$ in queue utilization and the mean estimated parameters $\hat{\theta}_i(k)$ can be shown to be bounded.

The rate selected by the above algorithm in equation (2) does not take into account the fading channels. To mitigate congestion due to channel fading, the selected rate from (4) has to be reduced when the transmission power calculated by the DPC scheme exceeds the transmitter node's capability (greater than maximum transmission power). This is accomplished by using virtual rates and back-off interval selection. Selecting the back-off interval for a given node is a difficult task since it depends upon the back-off intervals of all neighboring nodes which are normally unknown. Therefore an adaptive scheme is proposed to estimate the back-off interval.

B. Back-off Interval Selection

For a given node, a relationship between transmission rate and back-off interval depends upon the back-off intervals of all nodes within a sensing range of a transmitting node in the case of CSMA/CA paradigm. To calculate this relationship, a node needs to know the back-off intervals of all its neighbors, which is not feasible.

Therefore, we propose using a distributed and *predictive* algorithm to estimate back-off intervals, such that a target rate is achieved. The main goal is to select back-off interval, BO_i, at the i^{th} transmitting node such that the actual throughput meets the desired outgoing rate $f_i(k)$. To simplify calculations, we consider the inverse of the back-off interval, which is denoted as $VR_i = 1/BO_i$, where VR_i is the virtual rate at i^{th} node, and BO_i is the corresponding back-off interval. The fair scheduling algorithm schedules the packet transmissions according to the calculated node's back-off interval. The fair scheduling scheme is discussed in the next subsection. The interval is counted-down when a node does not detect any transmission, and pauses otherwise. Consequently, a node will gain access to the channel proportional to its virtual rate and inversely proportional to the sum of virtual rates of its neighbors. The actual rate of the i^{th} node is a fraction of the channel bandwidth, $B(t)$, defined as

$$R_i(t) = B(t) \cdot VR_i(t) \Big/ \sum_{l \in S_i} VR_l(t) = B(t) \cdot VR_i(t) / TVR_i(t) \tag{5}$$

where TVR_i is the sum of 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. It is assumed that the total bandwidth is constant as long as communication is possible on a link (when the received power is above a certain threshold). However, when the severe fading occurs, the bandwidth will drop to zero. In such a case, back-off intervals are set at a large value, *lar*, to prevent unnecessary transmissions when a suitable signal to noise ratio (SNR) cannot be achieved at a destination node due to power constraints. Additionally, under normal circumstances, the algorithm presented below is used to calculate the back-off interval BO_i, which is then randomized in order to minimize probability of collision on access between nodes. Consequently, the MAC layer back-off timer BT_i value is defined as

$$BT_i = \begin{cases} \rho * BO_i(k), & for\ B(k) = 1 \\ lar, & for\ B(k) = 0 \end{cases} \tag{6}$$

where ρ 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 channel fading.

Equation (5) represents the relationship between the back-off intervals and the outgoing flow rate. In order to design a controller which will track the target value of that rate, the system equation is differentiated and then transformed into discrete-time domain. This allows the feedback controller design for the selection of the appropriate back-off interval.

1) Adaptive Back-off Interval Selection

A discrete-time state equation is obtained using [12] as

$$R_i(k+1) = R_i(k)\alpha_i(k) + \beta_i(k)v_i(k) \tag{7}$$

where $\alpha_i(k)$=1-$TVR(k+1)/TVR_i(k)$, $\beta_i(k)=R_i(k)/VR_i(k)$, and $v_i(k)=VR_i(k+1)=1/BO_i(k+1)$. The variable α_i describes a variation of back-off intervals for flows at the neighboring nodes from the time instant k to $k+1$. This variation is caused due to congestion resulting from traffic and fading channels. Since this information is not available locally, it is considered an unknown parameter, and thus estimated by the algorithm. The parameter β_i is the ratio between actual and the used virtual rate at time instant k, and can be easily calculated. The term v_i is the back off interval that needs to be calculated for the node under consideration.

Equation (7) indicates that the achieved rate at the instant, k+1, depends on the variations of back-off intervals in the neighboring nodes. Now, select the back-off interval as

$$v_i(k) = (\beta_i(k))^{-1}\left[f_i(k) - R_{ij}(k)\hat{\alpha}_i(k) + \kappa_v e_i(k)\right] = 1/BO_i(k+1) \quad (8)$$

where $\hat{\alpha}_i(k)$ is estimate of $\alpha_i(k)$, $e_i(k)=R_i(k)-f_i(k)$ is defined as throughput error, and K_v is the feedback gain parameter. In this case, the throughput error is expressed as

$$e_i(k+1) = K_v e_i(k) + \tilde{\alpha}_i(k)R_i(k) + f_i(k) - f_i(k+1) \quad (9)$$

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

Given the back-off selection scheme above with the back-off interval updated as (8), consider that the parameter α_i is updated as

$$\hat{\alpha}_i(k+1) = \hat{\alpha}_i(k) + \sigma \cdot R_i(k) \cdot e_i(k+1) - \Gamma_i \left\| I - R_i^2(k) \right\| \hat{\alpha}_i(k) \quad (10)$$

where $\Gamma_i > 0$ is a design parameter. Then the mean error $e_i(k)$ in queue utilization and the mean estimated of the variable α_i can be shown to be bounded

2) Overall Convergence of the Proposed Scheme

In this section, the generalized theorem on convergence without the need for the persistency of excitation condition is presented. This shows that the overall system that combines the queue utilization control and back-off interval selection scheme is stable with a specific bound.

Let's rewrite error equations (3) and (9) as

$$\xi_i(k+1) = P_i(k)\tilde{\psi}_i^T(k) + K_v \cdot \xi_i(k) + \varepsilon(k) \quad (11)$$

where $\xi_i^T(k) = [e_{bi}(k) \quad e_i(k)]$ is a vector of estimation errors, $\tilde{\psi}_i^T(k) = [\tilde{\theta}_i(k) \quad \tilde{\alpha}_i(k)]$ is a vector of errors in estimation of system parameters, $P_i(k) = diag(f_i(k), R_i(k))$ is a regression matrix, $K_v = diag(k_{bv}, k_v)$ is gain matrix, and $\varepsilon(k)$ is vector of disturbance errors. Now, the parameters updates (4) and (10) can be rewritten as

$$\hat{\psi}_i(k+1) = \hat{\psi}_i(k) + \Lambda \cdot P_i(k) \cdot \xi_i(k+1) - \mathrm{H}\left\| I - P_i^2(k) \right\| \hat{\psi}_i(k) \quad (12)$$

where Λ and H are adaptation gains.

Theorem 1 presents that the proposed congestion control scheme ensures convergence of the actual value to its target outgoing rate and buffer utilization level where the dynamics are estimated by an adaptive scheme. The estimation error is bounded by known value ε_N. The proof guarantees stability in the sense of Lyapunov [10][11] as the buffer utilization control and back-off interval selection algorithms are posed a feedback dynamic system. Moreover, Theorem 1 relaxes requirement of the persistency of excitation condition present in [13].

Theorem 1 (Convergence of system with queue utilization and back-off interval selection algorithms): Given the incoming rate selection scheme and the back-off selection scheme above, and if the back-off intervals are selected as (2) and back-off interval selected as (8) with algorithm parameters update from (12). Then the mean estimation errors for queue utilization and outgoing rate, $\xi_i(k)$, and the mean estimated parameters $\hat{\psi}_i(k)$ are bounded without the need for the persistency of excitation condition, with the bounds specifically given by (22) or (24) provided the following conditions hold:

$$\Lambda\left\|P_i(k)\right\|^2 < 1 \quad (13)$$

$$0 < \mathrm{H} < 1 \quad (14)$$

$$K_{vmax} < 1/\sqrt{\delta} \quad (15)$$

where K_{vmax} is the maximum singular value of K_v, Λ is the adaptation gain, and

$$\delta = \eta + \frac{1}{1-\Lambda\left\|P_i(k)\right\|^2}\left[\mathrm{H}^2\left(1-\Lambda\left\|P_i(k)\right\|^2\right)^2 + 2\Lambda\mathrm{H}\left\|P_i(k)\right\|^2\left(1-\Lambda\left\|P_i(k)\right\|^2\right)\right] \quad (16)$$

Note: The parameters Λ, η, δ are dependent upon the desired outgoing rate value with time.

Proof: The Lyapunov candidate function is selected as

$$J_i = \xi_i^T(k)\xi_i(k) + \Lambda^{-1}\kappa\left[\tilde{\psi}_i^T(k)\tilde{\psi}_i(k)\right] \quad (17)$$

Then, the first difference is equal to

$$\begin{aligned}\Delta J_i &= \xi_i^T(k+1)\xi_i(k+1) - \xi_i^T(k)\xi_i(k) \\ &\quad + \Lambda^{-1}\kappa\left[\tilde{\psi}_i^T(k+1)\tilde{\psi}_i(k+1) - \tilde{\psi}_i^T(k)\tilde{\psi}_i(k)\right]\end{aligned} \quad (18)$$

Use the estimation error (11) and parameter tuning mechanism (12) to obtain

$$\begin{aligned}\Delta J_i \le &-\left[1-\Lambda k_{vmax}^2\right]\left\|\xi_i(k)\right\|^2 - \left[1-\Lambda\left\|P_i^2(k)\right\|\right]\cdot\| \; \tilde{\psi}_i^T(k)f_i(k) \\ &- \frac{1}{\left(1-\Lambda\left\|P_i(k)\right\|^2\right)}\left(\Lambda\left\|P_i(k)\right\|^2 + 2\mathrm{H}\left\|I-\Lambda\left\|P_i(k)\right\|^2\right\|\right) \\ &\cdot\left(K_v\left\|\xi_i(k)\right\| + \varepsilon(k)\right)\|^2 + 2\gamma k_{vmax}\left\|\xi_i(k)\right\| \\ &+ \rho - \frac{1}{\Lambda}\left\|I-\Lambda\left\|P_i(k)\right\|^2\right\|^2\left[\mathrm{H}(2-\mathrm{H})\left\|\hat{\psi}_i(k)\right\|\psi_{max} - \mathrm{H}^2\psi_{max}^2\right]\end{aligned} \quad (19)$$

where

$$\gamma = \eta(\varepsilon_N) + \mathrm{H}\left(1-\Lambda\left\|P_i(k)\right\|^2\right)\left\|P_i(k)\right\|\psi_{max}, \text{ and} \quad (20)$$

$$\rho = \left[\; \eta(\varepsilon_N)^2 + 2\mathrm{H}\left(1-\Lambda\left\|P_i(k)\right\|^2\right)\left\|P_i(k)\right\|\psi_{max}(\varepsilon_N) \;\right] \quad (21)$$

Completing the squares for $\tilde{\psi}_i(k)$ in (19) and taking expectations on both sides results in $E(J) > 0$ and $E(\Delta J) \le 0$, this shows the stability in the mean via sense of Lyapunov provided the conditions (13) and (15) hold. This demonstrates that $E(\Delta J)$ is negative outside a compact set U. According to a standard Lyapunov extension, the rate estimation error $E[\xi_i(k)]$ is bounded for all $k \ge 0$ and the upper bound on the mean rate error is given by

$$E\left(\left\|\xi_i(k)\right\|\right) > 1/\left(1-\Lambda k_{vmax}^2\right)\left[\gamma k_{vmax} + \sqrt{\rho_1\left(l-\Lambda k_{vmax}^2\right)}\right] \quad (22)$$

where $\rho_1 = \rho + \Lambda^{-1}\mathrm{H}/(2-\mathrm{H})\left(1-\Lambda\left\|\psi_i(k)\right\|^2\right)^2\psi_{max}^2$ (23)

On the other hand, completing the squares for $\left\|\xi_i(k)\right\|$ in (19) results in $E(\Delta J) \le 0$ as long as the conditions (13)-(15) are satisfied and

$$E\left(\left\|\tilde{\psi}_i(k)\right\|\right) > \left(\mathrm{H}(1-\mathrm{H})\psi_{max} + \sqrt{\mathrm{H}^2(1-\mathrm{H})^2\psi_{max}^2 + \mathrm{H}(2-\mathrm{H})\Theta}\right)\Big/\left(\mathrm{H}(2-\mathrm{H})\right) \quad (24)$$

where

$$\Theta = \left[\mathrm{H}^2\psi_{max}^2 + \Lambda\cdot\rho_2\Big/\left(1-\Lambda\left\|P_i(k)\right\|^2\right)^2\right] \text{ and} \quad (25)$$

$$\rho_2 = \rho + \gamma^2 k_{vmax}^2\Big/\left(1-\delta k_{vmax}^2\right) \quad (26)$$

In general $E(\Delta J) \le 0$ in a compact set as long as (13) and (15) are satisfied and either (22) or (24) holds. According to the standard Lyapunov extension theorem [12], this demonstrates that the tracking error and the error in parameter estimates are bounded without the need for any PE condition on the inputs.

Remarks:

a) Note that for practical purposes, (22) or (24) can be considered as bounds for $\|\xi_i(k)\|$, and $\|\tilde{\psi}_i(k)\|$. ■

3) 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 $u_{ij}(k) = u_i(k) \cdot \sum_m^{flows\ at\ j^{th}\ node}[\varphi_n] \big/ \sum_m^{flows\ at\ i^{th}\ node}[\varphi_m]$, 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 given by (3), and φ_n, φ_m are pre-assigned weights of the n^{th} and m^{th} flows respectively. Next, the selected rate $u_{ij}(k)$ is communicated to the upstream node *j to mitigate congestion*. This feedback continues recursively.

IV. Routing Scheme With Congestion Control

In this section we present the derivation of the delay cost function used to select less congested nodes in routing.

A. Route Selection with Delay Optimization

Consider a new flow m with weight φ_m for which a new route is being established. The goal of the proposed scheme is to optimize the end-to-end delay for the new flow. Hence, the cost function, V, is selected as a sum of delays on each hop toward the destination node

$$V(P) = \sum_{i \in P} D_i \tag{27}$$

where P is the list of hops on the route of flow m; D_i is the delay at hop i^{th}. This cost can be recursively expressed as

$$V(i) = D_{i+1} + V(i+1) \tag{28}$$

where $V(i)$ is a cost of route from i^{th} hop. The routing scheme has to select the node for the i^{th} hop such that the cost is minimized. The value of D_i is derived from the outgoing flow controlled by the congestion control scheme from Section III. The cost "to-go" $V(i+1)$ have to be estimated since it is not known at the time the routing decision is being made. The node with the lowest cost (28) is selected to be the next hop node. Next, the details of route selection are presented.

1) Estimation of the Cost "to-go"

The cost "to-go" can either calculated based on reports from the nodes on a potential path to BS, or estimated using expected values of per-hop delays and number of hops. The former method, though very accurate, is inefficient in wireless networks due to high communication and computational overheads. Hence, the latter method is proposed next.

The value $V(i)$ can be estimated as a sum of average delays, $\overline{D}$, over the $n(i)$ hops to the BS

$$V(i) = n(i) \cdot \overline{D} \tag{29}$$

Remark: The average per-hop delay, $\overline{D}$, proportional to ideal queue utilization level from Section III.A, since the proposed congestion control scheme will maintain queue level close to ideal value [13].

The number of hops can be calculated by dividing distance from the current node to BS, x_i, by the average distance gain toward BS per hop, $\overline{\delta}$

$$n(i) = x_i / \overline{\delta} \tag{30}$$

Remark: The number of hops, $n(i)$, is a real number since it represents the expected (average) number of hops.

Remark: The average distance gain, $\overline{\delta}$, can be calculated as an average difference in distance to BS between the current node and a randomly located next-hop node that is within the communication range, r. Moreover, the difference will be positive since the routing protocol requires that a relay node candidate is closer to BS than the current one.

Now, the cost "to-go" (29) can be rewritten as a function of a distance between i^{th} node and BS $V(x_i) = x_i \cdot \overline{D} / \overline{\delta}$.

2) Next Hop Delay

The next hop delay can be calculated using parameters from the proposed back-off selection algorithm from Section III.B. A packet delay for a given hop is expressed as

$$D_i = q_i / f_i \tag{31}$$

where q_i is a queue length at node i, and f_i is the outgoing flow rate. The queue level is equal to desired queue utilization level since the proposed congestion control algorithm is used to maintain the queue utilization at the desired level. The outgoing flow is provided by the congestion control algorithm.

Overall the cost function for a next hop candidate node (28) can be rewritten as

$$V(x_i) = q_{i+1} / f_{i+1} + x_{i+1} \cdot \overline{D} / \overline{\delta} \tag{32}$$

The selection of a next hop node on the route is based on the cost function (32). The node with the lowest cost is added to a new route $\min_{x_i}[V(x_i)] = \min_{x_i}[q_{i+1} / f_{i+1} + x_{i+1} \cdot \overline{D} / \overline{\delta}]$.

B. Protocol Design

The calculation of the cost value (32) is performed by the relay candidate node when it receives a request for a route. As a result, the calculation of the delay cost is distributed such that each node has to calculate one value. Next, the acknowledgement (ACK) message is transmitted back with the calculated delay cost value.

The relay node requesting a route will receive the ACK responses for a predefined interval and select node with the lowest cost value. The node will temporarily store information for the current best relay candidate thus minimizing memory requirements for the routing protocol.

V. SIMULATION RESULTS

Simulations of the proposed routing scheme were performed using MATLAB to provide an initial evaluation of RPCC routing performance. Simulations are performed on a 500m by 500m grid topology of 121 sensor nodes with each node 50m from its closest neighbors. The base station is located at position (50,250). The communication range for transmissions is 115m, the sensing range is 230m, each source generates 0.1 Mbps, channel bandwidth is 1 Mbps, and packets are 64 bytes. The ideal queue level is set to 10 packets. Since the RPCC scheme is used the average queue utilization is assumed to be equal to the ideal one. In cases where the channel is insufficient to carry all traffic RPCC is assumed to give each flow a proportional share of the bandwidth. New routes are established one after another. RPCC performance in terms of end-to-end delay has been compared to the scheme using a shortest path algorithm and congestion control scheme similar to the RPCC.

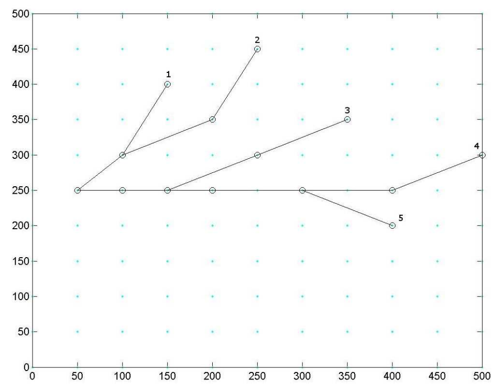

Fig. 2. OEDSR Routing without RPCC

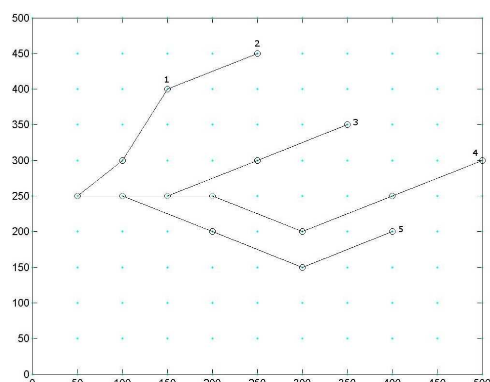

Fig. 3. OEDSR Routing with RPCC

Fig. 2 illustrates the paths selected by a shortest path routing. Five new sources, labeled 1 to 5, are created and the routes are established toward the BS one at a time. In Fig. 3, a routes selected by the RPCC scheme are shown. Table I summarizes the average end-to-end delay and throughput results of 20 simulations. In these simulations routes from sources 1 through 5 were established sequentially in random order. The paths for sources 2, 4, and 5 are different than in shortest path case since the route selection algorithm chooses less congested paths. This reduces the delay on these routes since the nodes along the RPCC path have fewer nodes that share the same channel bandwidth than for the shortest path scheme, as shown in Table I. Furthermore, by diverting these routes, RPCC also reduces the overall network congestion since it reduces number of interfering nodes. However, for route 3, delay and throughput does not change since the source is in the middle of transmitting network and there exist no alternative route with lower congestion.

TABLE I. SIMULATION RESULTS

	End-to-End Delay [s]					Throughput [kbps]				
Route	1	2	3	4	5	1	2	3	4	5
Shortest Path	.23	0.34	0.35	0.51	0.43	87	83	80	74	74
RPCC	.22	0.32	0.35	0.48	0.38	91	90	80	79	81

VI. CONCLUSIONS

This paper presents a novel routing-aware, predictive congestion control scheme whereby the congestion is mitigated by suitably predicting the back-off interval of all the nodes and providing alternate routes based on the current network conditions. The network conditions include the traffic flow through a given region and channel state. The convergence analysis is demonstrated by using a Lyapunov-based analysis. The simulation results show that the performance of RPCC improves over scheme with autonomous routing and congestion control.

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] B. Hull, K. Jamieson and H. Balakrishnan, "Mitigating congestion in wireless sensor networks", ACM SenSys 2004, pp. 134-147, November 2004

[3] C.-Y. Wan, S. Eisenman and A. Campbell, "CODA: Congestion detection and avoidance in sensor networks", Proc. 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 multi-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", Proc. of the IEEE LCN, pp. 547-556, 2002.

[6] J.C.R. Bennett, and H. Zhang, "WF2Q: worst-case fair weighted fair queuing", IEEE INFOCOM, Vol. 1, pp. 120–128, March 1996.

[7] N.H. Vaidya, P. Bahl, and S. Gupta, "Distributed fair scheduling in a wireless LAN", 6th Annual International Conference on Mobile Computing & Networking, pp. 167-178, August 2000.

[8] S.-C., Wang, , Y.-M., Chen, , T. H. Lee, A. Helmy, "Performance evaluations for hybrid IEEE 802.11b and 802.11g wireless networks", 24th IEEE IPCCC 2005, pp.111–118, April 2005.

[9] Wen-Kuang Kuo and C.-C. Jay Kuo, "Enhanced backoff scheme in CSMA/CA for IEEE 802.11", Vehicular Technology Conference, Fall 2003, IEEE 58th, vol. 5, pp.2809-2813, Oct. 2003.

[10] K. J. Astrom and B. Wittenmark, "Adaptive Control," Reading, MA: Addison-Wesley, 1989.

[11] S. Jagannathan and F.L. Lewis, "Robust implicit self tuning regulator/MRAC convergence and stability," Automatica, vol. 32, December 1996.

[12] F.L. Lewis, S. Jagannathan and A. Yesildirek, "Neural Network Control of Nonlinear Discrete-time Systems", Taylor&Francis, UK, 2006.