University of Nebraska - Lincoln

DigitalCommons@University of Nebraska - Lincoln

CSE Journal Articles

Computer Science and Engineering, Department of

2010

XLP: A Cross-Layer Protocol for Efficient Communication in Wireless Sensor Networks

Mehmet C. Vuran
IEEE, mcvuran@cse.unl.edu

Ian F. Akyildiz
IEEE

Follow this and additional works at: https://digitalcommons.unl.edu/csearticles

Part of the Computer Sciences Commons

XLP: A Cross-Layer Protocol for Efficient Communication in Wireless Sensor Networks

Mehmet C. Vuran, *Member, IEEE,* and Ian F. Akyildiz *Fellow, IEEE*

Abstract—Severe energy constraints of battery-powered sensor nodes necessitate energy-efficient communication in Wireless Sensor Networks (WSNs). However, the vast majority of the existing solutions is based on classical layered protocols approach, which leads to significant overhead. It is much more efficient to have a unified scheme which blends common protocol layer functionalities into a cross-layer module. In this paper, a cross-layer protocol (XLP) is introduced, which achieves congestion control, routing, and medium access control in a cross-layer fashion. The design principle of XLP is based on the cross-layer concept of *initiative determination*, which enables receiver-based contention, initiative-based forwarding, local congestion control, and distributed duty cycle operation to realize efficient and reliable communication in WSNs. The initiative determination requires simple comparisons against thresholds, and thus is very simple to implement, even on computationally impaired devices. To the best of our knowledge, XLP is the first protocol that integrates functionalities of all layers from PHY to transport into a cross-layer protocol. A cross-layer analytical framework is developed to investigate the performance of the XLP. Moreover, in a cross-layer simulation platform, the state-of-the- art layered and cross-layer protocols have been implemented along with XLP for performance evaluations. XLP significantly improves the communication performance and outperforms the traditional layered protocol architectures in terms of both network performance and implementation complexity.

Index Terms—Cross-Layer Protocol, Congestion Control, Routing, Medium Access Control, Wireless Sensor Networks.

I. INTRODUCTION

WIRELESS sensor networks (WSNs) are event-based systems that exploit the collective effort of densely deployed sensor nodes and continuously observe a certain physical phenomenon. The main goal is to reliably detect/estimate event features from the collective information provided by sensor nodes respecting their limited energy, storage, and processing capabilities. To this end, there has been significant amount of research efforts that aim to develop collaborative networking protocols to achieve communication with maximum energy efficiency.

The majority of the communication protocols are individually developed and optimized for different networking layers, i.e., transport, network, medium access control (MAC), and physical layers. While these protocols achieve very high performance in terms of the metrics related to each of these individual layers, they are not jointly designed and optimized to maximize the overall network performance while minimizing the energy expenditure. Considering the scarce energy and

processing resources of WSNs, joint design of networking layers, i.e., *cross-layer design*, stands as the most promising alternative that has gained interest recently.

Recent results [13], [15] reveal that cross-layer integration and design techniques result in significant improvement in terms of energy efficiency in WSNs. These results have, recently, led to several solutions on the cross-layer interaction and design [2], [3], [5], [7], [8], [9], [11], [21], [22], [27], [28], [29], [38], [39]. A detailed discussion of these solutions can be found in [25]. However, as discussed in Section II, these studies either provide analytical results without any communication protocol design, or perform cross-layer design within a limited scope, e.g., routing and MAC layers.

Clearly, there is still much to be gained by rethinking the functionalities of protocol layers in a unified way so as to provide a single communication module for efficient communication in WSNs. To this end, this paper introduces a novel concept, i.e., *initiative determination* and illustrates how certain traditional networking functionalities can be jointly designed based on this concept to implement a cross-layer operation of medium access, distributed routing, and local congestion control functionalities. The *initiative determination* procedure is used for each node to decide on participating in communication based on its current state related to link quality, location, current traffic load, buffer level, and remaining energy level. These fundamental operation states are incorporated into a unified decision incentive to define a node's level of willingness in participating in the communication. Accordingly, a cross-layer protocol (XLP) is developed to achieve efficient and reliable event communication in WSNs with minimum energy expenditure.[1] In a cross-layer simulation platform, the state-of-the-art layered and cross-layer protocol configurations have been implemented along with XLP to provide a complete performance evaluation. Analytical performance evaluation and simulation experiment results show that XLP significantly improves the communication performance and outperforms the traditional layered and recent cross-layer protocol architectures in terms of both network performance and implementation complexity. These results highlight the advantages of the initiative concept, which is a novel perspective for networking in WSNs.

The remainder of the paper is organized as follows. In Section II, we provide a review of existing work on cross-layer design in WSNs. The XLP basics, overview, and protocol description are introduced in Section III as well as the theoretical analysis framework. In Section IV, we provide performance evaluations of the XLP solution and provide a comparative analysis with five layered suites. Finally, the paper

M. C. Vuran is with the Department of Computer Science and Engineering, University of Nebraska-Lincoln, Lincoln, NE 68588, Tel: (402) 472-5019, Fax: (402) 472-7767, Email:mcvuran@cse.unl.edu

I. F. Akyildiz is with the Broadband Wireless Networking Laboratory, School of Electrical & Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, Tel: (404) 894-5141, Fax: (404) 894-7883, Email:ian@ece.gatech.edu

[1]A preliminary version of this work has appeared in [2].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

IEEE TRANSACTIONS ON MOBILE COMPUTING

2

is concluded in Section V.

II. RELATED WORK

A cross-layer MAC/routing protocol has been independently proposed by [29] and [39]. In [38] and [39], the performance analysis of the geographical random forwarding (GeRaF) algorithm is presented. This algorithm introduces receiver-based routing for cross-layer interaction between MAC and routing layers. However, the GeRaF algorithm requires a sensor node with two radios for signaling, which may not be feasible in some scenarios. In [37], the MAC protocol is modified for a single radio node. However, the solutions in [37], [38], [39] consider a perfect channel model and are based purely on geographical relations. In [29], a receiver-based routing protocol is proposed, where the performance of the protocol is analyzed based on a simple channel model and lossless links. Moreover, the latency performance of the protocol is presented based on different delay functions and collision rates. Also, the effect of the physical layer are not considered in the protocol operation.

An integrated MAC/routing solution is proposed in [26] for geographical routing in WSNs, which considers a realistic channel model including fading channel statistics. However, this work focuses on the interaction between MAC and routing layers and omits the transport layer and physical layer issues. Furthermore, the integrated MAC/routing scheme proposed in this work does not explicitly address the energy-efficiency requirement of energy-constrained sensor networks.

In [2], we have developed the cross-layer protocol (XLP), which integrates physical, MAC, routing, as well as transport layer functionalities into a unified communication framework. In this work, we extend this protocol to distributively account for the local minima experienced by geographical routing protocols, provide an analytical analysis of the XLP protocol, and perform extensive comparative evaluations including state-of-the-art layered and cross-layer protocols.

Recently, receiver-based contention techniques have also been adopted in several cross-layer MAC and routing protocols. The adaptive load balanced algorithm (ALBA) is described in [3], [4], and [5], which is based on the GeRaF framework in [38]. In addition to the location of the nodes, the traffic load on each node is considered for route establishment. More specifically, each potential node computes two values: geographic priority index (GPI) and queue priority index (QPI), which indicate the progress of the node towards the destination and its traffic load, respectively. Accordingly, if a node has a packet to send, it sends several RTS packets to scan QPI and GPI values of its neighbors. Each neighbor responds to this packet by CTS packets if their values match the requested values. The source node then selects one of the neighbors if the requested value is found. Moreover, in [5], ALBA-R is proposed, which enhances the original protocol to avoid local minima in routing through a coloring scheme called Rainbow. Based on its previous success in finding relays, each node assigns itself a different color, which is used to participate in communication. While ALBA-R employs a cross-layer MAC/routing technique, the route selection is performed at the sender node, which incurs high overhead due to the QPI and GPI scanning. Moreover, the transport layer issues such as congestion control are not completely addressed in ALBA-R as we will discuss in Section IV.

In [11], an integrated MAC/routing (MACRO) protocol is developed, where the next hops are selected according to a weighted progress factor, which considers the progress towards destination per transmitted power. MACRO also employs the receiver-based contention scheme but considers only energy efficiency and geographical locations for communication. The convergent MAC (CMAC) protocol is developed in [21], where next hop nodes are selected according to a normalized latency metric. The CTS replies are prioritized according to this metric and the routes are selected based on this prioritization mechanism.

In [22], an integrated routing and interest dissemination system (IRIS) is proposed, where several communication techniques including node density estimation, hop count-based routing protocol, and an interest dissemination system are integrated into a cross-layer framework. Although this cross-layer framework is shown to provide fast dissemination and balanced operation in an heterogeneous setting, transport layer issues such as congestion control have not been addressed. In [27], a routing protocol is proposed, where a weighted cost function including position cost, queuing cost, and remaining energy cost is used for relay selection. The position cost is associated with the angular offset of the potential relay node from the base station as seen by the transmitting node.

While there exist several cross-layer protocols that exploit the receiver-based contention and geographical routing principles, these solutions are limited to MAC and routing cross-layer operation. However, network congestion and link reliability that are related to transport and physical layer functionalities are yet to be considered in a unified cross-layer protocol.

In [7] and [8], a cross-layer optimization solution for power control at the physical layer and congestion control at the transport layer is considered. Moreover, a cross-layer analysis of the impact of physical layer constraints on link-level and network-level performance of CDMA sensor networks is presented in [10]. This work underlines important trade-offs between topology control and receiver design principles. However, these studies apply only to CDMA-based wireless multihop networks which may not apply to WSNs where CDMA technology may not be the most efficient scheme.

In [28], new forwarding strategies for geographic routing are proposed based on the analytical work in [40]. Expressions for the optimal forwarding distance for networks with and without ARQ are also provided. However, the forwarding protocol does not consider the impact of medium access and results in a high overhead. Moreover, the analysis for the distribution of optimal hop distance is based on a linear network structure, which may not be practical for WSNs, where a 2-dimensional terrain exists.

In [9], a joint routing, MAC, and link layer optimization framework is proposed. Although the optimization problems are insightful, no communication protocol for practical implementation is proposed. Moreover, the transport layer issues such as congestion and flow control are not considered. Furthermore, in [19] and [20], a thorough investigation of optimization techniques for cross-layer design in wireless networks are performed. It has been stated that scheduling constitutes the bottleneck in optimization due to the nonconvex nature of the scheduling problem. Consequently, in [20], a distributed cross-layer congestion control and scheduling al-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

IEEE TRANSACTIONS ON MOBILE COMPUTING

3

gorithm is developed. However, the provided solution focuses only on two layers for cross-layer design including transport and link layers.

These studies either provide analytical results without any communication protocol design, or perform cross-layer design within limited scope, e.g., only routing and MAC. In this paper, we argue that a new networking paradigm is required to design a cross-layer protocol that addresses medium access, routing, and congestion issues in WSNs.

III. XLP: Cross-Layer Protocol for WSNs

The design principle of XLP is a unified cross-layering such that both the information and the functionalities of three fundamental communication paradigm (medium access, routing, and congestion control) are considered in a single protocol operation. Consequently, XLP incorporates the required functionalities by considering the channel effects.[2] The details of these functionalities are explained in the following sections. Before explaining the specifics of the XLP operation, we first introduce the *initiative determination* concept, which constitutes the core of the XLP.

A. Initiative Determination

The *initiative determination* concept coupled with the receiver-based contention mechanism provides freedom to each node in participating in communication. In WSNs, the major goal of a communication suite is to successfully transport event information by constructing (possibly) multi-hop paths to the sink. To this end, the cross-layer initiative determination concept constitutes the core of the XLP and implicitly incorporates the intrinsic communication functionalities required for successful communication in WSNs.

A node, i, initiates transmission by informing its neighbors that it has a packet to send. This is achieved by broadcasting a request to send (RTS) packet. Upon receiving this packet, each neighbor of node i decides to participate in the communication or not. This decision is made through the *initiative determination* based on the current state of the node. The initiative determination is a binary operation where a node decides to participate in communication if its initiative is 1. Denoting the initiative as $\mathcal{I}$, it is determined as follows:

$$\mathcal{I} = \begin{cases} 1, & \text{if} \begin{cases} \xi_{RTS} \geq \xi_{Th} \\ \lambda_{relay} \leq \lambda_{relay}^{Th} \\ \beta \leq \beta^{max} \\ E_{rem} \geq E_{rem}^{min} \end{cases} \\ \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

The initiative is set to 1 if all four conditions in (1) are satisfied, where each condition constitutes a certain communication functionality in XLP. The first condition, i.e., $\xi_{RTS} \geq \xi_{Th}$, ensures reliable links to be constructed for communication based on the current channel conditions. For this purpose, it is required that the received signal to noise ratio (SNR) of an RTS packet, ξ_{RTS}, is above some threshold ξ_{Th} for a node to participate in communication. The effect of this threshold on

routing and energy consumption performance will be analyzed and the most efficient value of this threshold will be chosen in Section III . The second, i.e., $\lambda_{relay} \leq \lambda_{relay}^{Th}$, and the third, i.e., $\beta \leq \beta^{max}$, conditions are used for local congestion control in XLP. As will be explained in Section III-F, the second condition prevents congestion by limiting the traffic a node can relay. More specifically, a node participates in the communication if its relay input rate, λ_{relay}, is below some threshold λ_{relay}^{Th}. The third condition ensures that the buffer occupancy level of a node, β, does not exceed a specific threshold, β^{max}, so that the node does not experience buffer overflow and the congestion is prevented. The last condition, i.e., $E_{rem} \geq E_{rem}^{min}$, ensures that the remaining energy of a node E_{rem} stays above a minimum value, E_{rem}^{min}. This constraint helps preserve uniform distribution of energy consumption throughout the network.

The cross-layer functionalities of XLP lie in these constraints that define the *initiative* of a node to participate in communication. Using the initiative concept, XLP performs receiver-based contention, initiative-based forwarding, local congestion control, hop-by-hop reliability, and distributed operation. The details of XLP operation are explained next. More specifically, we first define the basic parameters and the network model considered for the operation of XLP in Section III-B. For a successful communication, a node first initiates transmission using a broadcast message as explained in Section III-C. Then, the neighbors that receive this message perform initiative determination. The neighbors that decide to participate in communication perform receiver-based contention or angle-based routing as described in Section III-D and Section III-E, respectively. Moreover, the local congestion control component ensures energy efficient as well as reliable communication by two-step congestion control as explained in Section III-F. Finally, based on this protocol operation description, the operation of XLP is analytically investigated in Section III-G.

B. Basics, Definitions, and Network Model

We assume the following network model for the operation of XLP: Each node performs a distributed duty cycle operation such that the transceiver circuit of the node is *on* for a certain fraction of the time and is switched *off* for the remaining fraction of the time, during which the sensors can still sample data. The on-off periods are managed through a *duty cycle parameter*, δ, which defines the fraction of the time a node is active. More specifically, each node is implemented with a sleep frame with length T_S sec. A node is active for $\delta \times T_S$ sec and is asleep state for $(1 - \delta) \times T_S$ sec. Note that the start and end times of each node's sleep cycle are not synchronized. Consequently, a distributed duty cycle operation is employed. Furthermore, we assume that each node is aware of its location. This assumption is motivated by the fact that WSN applications inherently require location information to associate the observed information by each node to a physical location. Hence, each node is required to be aware of its location, which can be provided through either an on-board GPS or a localization algorithm [23]. Thus, it is only natural to leverage this information for communication. The network model is also geared towards event-based information flow, where nodes send information to a single stationary sink if an

---

[2]Note that the sensor nodes equipped with XLP will still have RF transceiver which has all necessary physical layer functionalities, e.g., modulation/demodulation, channel coding, RF power control, specified according to the specific deployment and application requirements.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

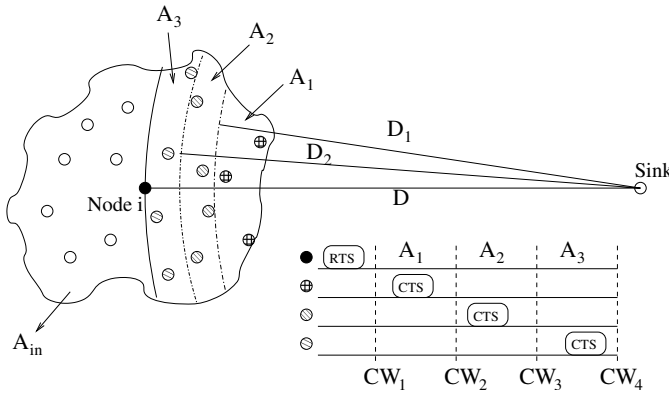IEEE TRANSACTIONS ON MOBILE COMPUTING

4

Fig. 1.   Priority regions and the prioritization mechanism.

event occurs in their vicinity. The area that an event occurs is denoted as the event area and the nodes in this area generate event information. Based on this network model, the protocol operation details are explained in the following sections.

C. Transmission Initiation

When a node i has a packet to transmit, it first listens to the channel for a specific period of time. If the channel is occupied, the node performs backoff based on its contention window size, CW_{RTS}. When the channel is idle, the node broadcasts an RTS packet, which contains the location information of the sensor node i and the sink. This packet also serves as a link quality indicator and helps the neighbors to perform receiver-contention, which is explained in Section III-D. When a neighbor of node i receives an RTS packet, it first checks the source and destination locations. We refer to the region, where the neighbors of a node that are closer to the sink reside, as the *feasible region* and the remaining neighborhood as the *infeasible region*. A node, which receives a packet, first checks if it is inside the feasible region. To save energy, nodes inside the infeasible region switch to sleep for the duration of the communication. The nodes inside the feasible region perform initiative determination as explained in Section III. If a node decides to participate in communication, it performs *receiver contention* as explained next.

D. Receiver Contention

The receiver contention operation of XLP leverages the initiative determination concept with the receiver-based routing approach [29], [38]. After an RTS packet is received, if a node has an initiative to participate in the communication, i.e., $\mathcal{I} = 1$, it performs receiver contention to forward the packet. The receiver contention is based on the routing level of each node, which is determined based on the progress a packet would make if the node forwards the packet. The feasible region is divided into N_p priority regions, i.e., $A_i, \ i = 1, ..., N_p$. Nodes with longer progress have higher priority over other nodes. According to the location information, each node determines its priority region and performs contention for medium access as explained next.

Each priority region, A_i, corresponds to a backoff window size, CW_i. Based on its location, a node backs off for $\sum_{j=1}^{i-1} CW_j + cw_i$, where cw_i is randomly chosen such that

$cw_i \in [0, CW_i]$. This backoff scheme helps differentiate nodes of different progress into different prioritization groups. Only nodes inside the same group contend with each other. The winner of the contention sends a CTS packet to node i indicating that it will forward the packet. On the other hand, if during backoff, a potential receiver k receives a CTS packet, it determines that another potential receiver j with a longer progress has accepted to forward the packet and node k switches to sleep for the duration of the communication.

The case for $N_p = 3$ priority regions is shown in Fig. 1. Based on their potential advancement, each feasible node corresponds to one of the three priority regions A_1, A_2, or A_3. The backoff scheme is also illustrated in Fig. 1, where the possible times when a CTS packet can be sent are shown. As an example, if a node in A_2 satisfies the initiative function, it first waits for CW_2 in addition to a random cw_2 value. Consequently, the node in A_2 can transmit CTS packet only if no node in A_1 transmits a CTS packet.

When node i receives a CTS packet from a potential receiver, it determines that the receiver contention has ended and sends a DATA packet with the position of the winner node in the header. The CTS and DATA packets both inform the other contending nodes about the transmitter-receiver pair. Hence, other nodes stop contending and switch to sleep. In the case of two nodes sending CTS packets without hearing each other, the DATA packet sent by node i can resolve the contention. It may happen that multiple CTS packets from the same priority region can collide and a node from a lower priority region can be selected. XLP does not try to resolve this problem as this probability is very low since the contention region is already divided into multiple regions and the cost of trying to resolve this outweighs the gains.

Note that node i may not receive a CTS packet because of three reasons: (1) CTS packets collide, (2) There exists no potential neighbors with $\mathcal{I} = 1$, or (3) There exists no nodes in the feasible region. However, node i cannot differentiate these three cases by the lack of a CTS packet. Hence, the neighbors of node i send a keep alive packet after $\sum_{j=1}^{\mathcal{N}_p} CW_j + cw$ if no communication is overheard. In this case, cw is a random number, where $cw \in [0, CW]$ and N_p is the number of priority regions as explained before. The existence of a keep alive packet notifies the sender that there exist nodes closer to the sink, but the initiative in (1) is not met for any of these nodes. With the reception of this packet, node performs retransmission. However, if a keep alive packet is not received, the node continues retransmission in case there is a CTS packet collision. If no response is received after k retries, node i determines that a local minimum is reached and switches to angle-based routing mode as explained next.

E. Angle-based Routing

Since the routing decisions depend, in part, on the locations of the receivers, there may be cases where the packets reach local minima. In other words, a node may not find any feasible nodes that are closer to the sink than itself. This problem is known as communications void in geographical routing-based approaches and is generally resolved through face routing techniques [6], [12], [17], [18], [31]. Although localized, face routing necessitates a node to communicate with its neighbors to establish a planarized graph and construct routes to traverse
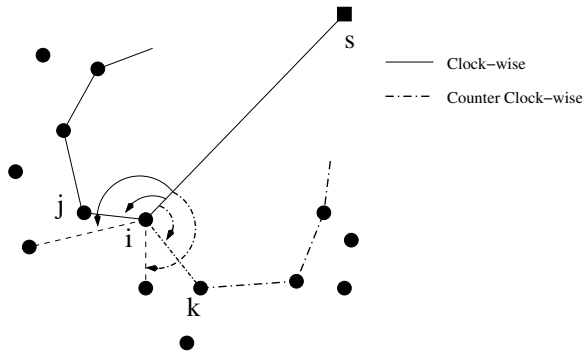
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

IEEE TRANSACTIONS ON MOBILE COMPUTING

5

Fig. 2.   Illustration of angle-based routing.

Fig. 3.   Sample route created by angle-based routing.

around the void. This requires information exchange between the neighbors of a node. Since this communication increases the protocol overhead, we introduce a stateless solution to face routing, i.e., *angle-based routing* technique.

The main principle of the angle-based routing can be seen in Fig. 2. When a packet reaches node i, which is a local minimum towards the sink, the packet has to be routed around the void either in clock-wise direction (through node j) or in counter clock-wise direction (through node k). Assume that lines are drawn between the node i and the sink, s, as well as between node i and its neighbors. If we compare the angles between the line i, s and the other lines, the angle $\angle sij$ (angle $\angle sik$) has the smallest angle in the counter clock-wise (clock-wise) routing direction. Using this geometric property, routes can be constructed around the void. Once a direction is set (clock-wise or counter clock-wise), the packet traverses around the void using the same direction. Hence, for angle-based routing, we introduce the term *traversal direction* to indicate this direction. Note that clockwise (counter clock-wise) traversal direction refers to the traversal direction of the packets rather than the way the angles are measured.

When a node switches to angle-based routing mode, it also sets the traversal direction to clock-wise and sends an RTS packet, which indicates both the routing mode and the traversal direction. The nodes that receive this packet calculate their angle relative to the source-sink direction. Denoting the angle by θ_{ij}, node j sets its contention window to $c\theta_{ij} + cw_i$, where cw_i is a random number, and c is a constant.[3] The node with the smallest angle (hence, the smallest contention window) sends a CTS packet and the data communication takes place. This procedure is repeated until the packet reaches a local minimum. In this case, the traversal direction is set to counter clock-wise and the procedure is repeated. Angle-based routing is terminated and the basic XLP is performed when the packet reaches a node that is closer to the sink than the node that initiated the angle-based routing. A sample route found by this algorithm is shown in Fig. 3, where the sink is denoted by s. Since node a is a local minimum, XLP switches to angle-based routing mode in clock-wise direction. The packet is routed towards node c, where the traversal direction is changed and the packet reaches node d. Since node d is closer to the sink than node a, the angle-based routing mode is terminated and packet is forwarded until node e using basic XLP. At node e,
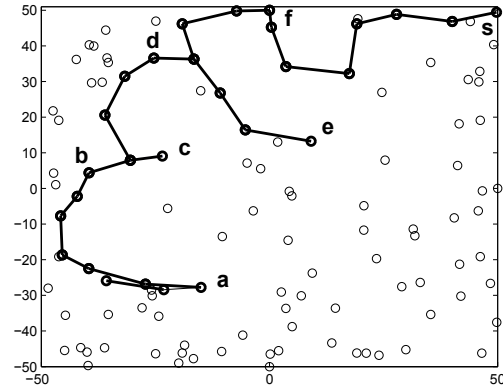
a local minimum is reached and angle-based routing mode is used again. Finally, at node f, this mode is terminated since node f is closer to the sink than node e. Packet is routed from node f to the sink s using basic XLP.

The correctness of the angle-based routing protocol can be proved by proving that no loops are generated. It is well known that according to the right-hand rule, traversing along a face prevents loops as long as the next face is correctly chosen [6]. In angle-based routing, nodes select their next hops according to the minimum angle between themselves and the previous nodes. This ensures that the packet always traverses on the edge of the face. Consequently, loops can only be created if the packet traverses through the initial face back to the initial sender. However, for this to occur, the packet should coincide with the direct line between the initial sender and the sink, which indicates that the packet reaches a node that is closer to the sink. This is a contradiction since in this case, the protocol switches to geographical routing and the loops are prevented.

F. Local Cross-Layer Congestion Control

XLP incorporates a new hop-by-hop local cross-layer congestion control component, which is devised based on the buffer occupancy analysis presented here. The objective of this component is to perform hop-by-hop congestion control by exploiting the local information in the receiver-contention and avoid the need for end-to-end congestion control. It also exploits the local reliability measures taken by the channel access functionality and hence, does not necessitate traditional end-to-end reliability mechanisms.

A sensor node has two duties in WSNs, i.e., source duty and router duty. Accordingly, we consider two sources of traffic as an input to the buffer of each node: (1) *Generated packets:* The first source is the application layer, i.e., the sensing unit of a node, which senses the event and generates the data packets to be transmitted. The rate of the generated packets is denoted by λ_{ii}. (2) *Relay packets:* In addition to generated packets, as a part of its router duty, a node also receives packets from its neighbors to be forwarded to the sink due to multi-hop nature of sensor networks. The rate at which node i receives relay packets from node j is denoted as λ_{ji}.

The input rate to the buffer of node i is, hence, the combination of the input rates of these two types of packets.

---

[3]The constant can be selected according to the latency requirements and the density of the network.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

IEEE TRANSACTIONS ON MOBILE COMPUTING

6

Since the sensor nodes utilize a duty cycle operation, the buffer occupancy of the nodes build up while they sleep because of the generated packets unless appropriate actions are taken. The local cross-layer congestion control component of XLP has two main measures to regulate the congestion in router duty, by providing the sensor nodes with the freedom of deciding whether or not participating in the forwarding of the relay packets based on the current load on the node, and in source duty, by explicitly controlling the rate of the generated packets.

We first analyze the upper bound for the total relay packet rate that will prevent congestion. Accordingly, a decision bound is derived for local congestion at each node. More specifically, this bound, denoted by λ_{relay}^{Th} is used in the XLP initiative determination as presented in (1) in Section III.

The overall input packet rate at the node i, λ_i, can be represented as

$$\lambda_i = \lambda_{ii} + \lambda_{i,relay} = \lambda_{ii} + \sum_{j \in \mathcal{N}_i^{in}} \lambda_{ji} \qquad (2)$$

where λ_{ii} is the generated packet rate, λ_{ji} is the relay packet rate from node j to node i, $\mathcal{N}_i^{in}$ is the set of nodes from which node i receives relay packets, and $\lambda_{i,relay}$ is the overall relay packet rate of node i. Node i aims to transmit all the packets in its buffer and hence, the overall output rate of node i is given by

$$\mu_i = (1 + e_i)(\lambda_{ii} + \lambda_{i,relay}) \qquad (3)$$

where e_i is the packet error rate and $1 + e_i$ is used to approximate the retransmission rate since the routes are selected by considering a high SNR value through the initiative determination process. Note that since the node retransmits the packets that are not successfully sent, the output rate is higher than input rate.

According to (2) and (3), in a long enough interval, T_∞, the average time the node i spends in transmitting and receiving are given by

$$\begin{aligned} T_{rx} &= \lambda_{i,relay} T_\infty T_{PKT} \ , \\ T_{tx} &= (1 + e_i)(\lambda_{ii} + \lambda_{i,relay}) T_\infty T_{PKT} \end{aligned} \qquad (4)$$

respectively, where T_{PKT} is the average duration to transmit a packet to another node including the medium access overhead.

To prevent congestion at a node, the generated and received packets should be transmitted during the time the node is active. Because of the duty cycle operation, on the average, a node is active δT_∞ sec. Therefore,

$$\delta T_\infty \geq \left[(1 + e_i)\lambda_{ii} + (2 + e_i)\lambda_{i,relay} \right] T_\infty T_{PKT} \ . \qquad (5)$$

Consequently, the input relay packet rate, $\lambda_{i,relay}$ is bounded by

$$\lambda_{i,relay} \leq \lambda_{i,relay}^{Th} \ , \qquad (6)$$

where the relay rate threshold, $\lambda_{i,relay}^{Th}$, is given by

$$\lambda_{i,relay}^{Th} = \frac{\delta}{(2 + e_i)T_{PKT}} - \frac{1 + e_i}{2 + e_i}\lambda_{ii} \ . \qquad (7)$$

The above analysis shows that by throttling the input relay rate, congestion at a node can be prevented. This result is incorporated into XLP through a hop-by-hop congestion control mechanism, where nodes participate in routing packets as long as (6) is satisfied. The implementation of the inequality

in (6) necessitates a node to calculate the parameters e_i, T_{PKT}, and λ_{ii}. To this end, the generated packet rate, λ_{ii}, is easily extracted from the rate of injected packets from the sensing boards to the communication module. The packet error rate, e_i, is stored as a moving average of the packet loss rate encountered by the node. Similarly, T_{PKT} is determined by using the delay encountered in sending the previous packet by the node. Consequently, each node updates these values after a successful or unsuccessful transmission of a packet.

According to (7), the relay rate threshold, $\lambda_{i,relay}^{Th}$, is directly proportional to the duty cycle parameter, δ. This suggests that the capacity of the network will decrease as δ is reduced. Moreover, the inequality in (6) ensures that the input relay rate of source nodes, i.e., nodes with $\lambda_{ii} > 0$, is lower than that of the nodes that are only relays, i.e., $\lambda_{ii} = 0$. This provides homogenous distribution of traffic load in the network, where source nodes relay less traffic.

The inequality given in (6) controls the congestion in the long term. However, in some cases, the buffer of a node can still be full due to short term changes in the traffic. To prevent buffer overflow in these cases, the third inequality in (1) is used by the nodes to determine their initiative. More specifically, the inequality $\beta \leq \beta^{max}$ ensures that the buffer level, β, is lower than the threshold, β^{max}, which is the maximum buffer length of a node. Consequently, if a node's buffer is full, it does not participate in communication.

In addition to regulating the relay functionality as discussed above, the XLP local congestion control component also takes an active control measure by directly regulating the amount of traffic generated and injected into the network. During the receiver-contention mechanism described in Section III-D, node i may not receive any CTS packets but receive keep alive packets. In this case, node i decides that there is a congestion in the network. Then, it reduces its transmission rate by decreasing the amount of traffic generated by itself. In other words, since the traffic injected by any node due to its router duty is controlled based on (6), the active congestion control is performed by controlling the rate of generated packets λ_{ii} at the node i.

In case of congestion, XLP node reduces the rate of generated packets λ_{ii} multiplicatively, i.e.,. $\lambda_{ii} = \lambda_{ii} \cdot 1/\nu$, where ν is defined to be the transmission rate throttle factor. If there is no congestion detected, then the packet generation rate can be increased conservatively to prevent oscillation in the local traffic load. Therefore, XLP node increases its generated packet rate linearly for each ACK packet received, i.e., $\lambda_{ii} = \lambda_{ii} + \alpha$. XLP adopts a rather conservative rate control approach mainly because it has two functionalities to control the congestion for both the source and the router duties of a sensor node. As the node decides to take part in the forwarding based on its buffer occupancy level and relay rate, it already performs congestion control as a part of the XLP's forwarding mechanism. Hence, XLP node does not apply its active congestion control measures, i.e., linear increase and multiplicative decrease, to the overall transmission rate. Instead, only the generated packet rate, λ_{ii}, is updated.

Since the local congestion control is specific to certain regions and may not apply to the entire event area, nodes inside a congested region may reduce their transmission rates and the overall event reliability may still be met at the sink from the data from other nodes due to the sheer amount of correlated

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

IEEE TRANSACTIONS ON MOBILE COMPUTING

7

data flows as in [1]. Thus, instead of an inefficient end-to-end reliability mechanism, the local cross-layer congestion control exploits the local congestion control and reliability to maintain high network utilization and overall reliability in a distributed manner. In fact, this is also clearly observed in the performance evaluation results as presented in Section IV.

G. XLP Duty Cycle Analysis

XLP employs a distributed duty cycle operation as described in Section III-B. Hence, the choice of the duty cycle value, δ, is important for the performance of XLP. Accordingly, we investigate the effect of duty cycle on the network performance using an energy consumption analysis. In this respect, the energy consumed by the network for a packet sent to the sink as a function of the distance of its source to the sink is investigated.

The total energy consumed from a source node at distance D from the sink can be found as

$$E_{flow}(D) = E_{per\_hop} \, E[n_{hops}(D)] \qquad (8)$$

where E_{per_hop} is the average energy consumed in one hop for transmitting a packet, and $E[n_{hops}(D)]$ is the expected hop count from a source at distance D to the sink. An accurate approximation for the expected hop count is given in [38] as

$$E[n_{hops}(D)] \simeq \frac{D - R_{inf}}{E[d_{next\_hop}]} + 1 \qquad (9)$$

where $E[d_{next\_hop}]$ is the expected hop distance, which we have previously analyzed and derived in [34], [35] and R_{inf} is the approximated transmission range.

The energy consumed in one hop has three components as given by

$$E_{per\_hop} = E_{TX} + E_{RX} + E_{neigh} \qquad (10)$$

where E_{TX} is the energy consumed by the node transmitting the packet, E_{RX} is the energy consumed by the node receiving the packet, and E_{neigh} is the energy consumed by the neighbors of both transmitter and the receiver nodes. Similar energy consumption analysis has also been performed in the literature in a node-centric manner, which require assumptions for the generated traffic, e.g., [29], [38]. On the other hand, the effect of neighbor nodes has not been considered [14]. In our analysis, we investigate the energy consumption to transmit a single packet to the sink with the effect of neighbor nodes, which provides a clearer insight into the energy consumption.

To successfully transmit the packet, a pair of nodes need to complete the four-way handshaking. Assume that the distance between the pair of nodes is $d_h = E[d_{next\_hop}]$. Moreover, the probability to successfully receive a data packet and a control packet at this distance are $p_s^D(d_h)$ and $p_s^C(d_h)$, respectively.[4] When a transmitter node sends an RTS packet, it is received by the receiver node with probability $p_s^C(d_h)$ and the node replies with a CTS packet. If the CTS packet is received (also with probability $p_s^C(d_h)$), the transmitter node sends a DATA packet, and the communication is concluded with an ACK packet. In every failure event, the node begins retransmission.

[4]We reasonably assume that the length of RTS, CTS and ACK packets are the same.

Therefore, the expected energy consumed by the transmitting node, E_{TX}, is

$$E_{TX} = \frac{K}{(p_s^C)^3 p_s^D} \quad, \qquad (11)$$

where

$$
\begin{aligned}
K \;=\; & E_{sense} + (p_s^C)^2 \left[E_{tx}^R + E_{wait}^C + E_{rx}^C \right] \\
& + \left(1 - \left(p_s^C \right)^2 \right) E_{t/o}^C + \left(p_s^C \right)^3 p_s^D \left[E_{tx}^D + E_{rx}^A \right] \\
& + \left(p_s^C \right)^2 (1 - p_s^C p_s^D) E_{t/o}^A
\end{aligned}
$$

and E_{sense} is the energy consumption spent for sensing the region, E_{tx}^x and E_{rx}^x are the packet transmission and reception energies spent for packets, where the superscripts R, C, D, A refer to RTS, CTS, DATA and ACK packets, respectively. E_{wait}^{CTS} is the expected energy consumption for waiting for a receiver CTS, and $E_{t/o}$ is the energy consumed before the transmitter node times out, deciding that a suitable relay node does not exist. The two terms in (11), E_{wait}^C and $E_{t/o}^C$, are the only system dependent terms. The expected waiting time for the next hop E_{wait}^C is calculated next.

According to the discussion in Section III-D, on the average, each node in priority region, A_i, waits for $CW_i/2$ in its priority slot in addition to waiting for the previous priority slots. Denoting the probability that the next hop for node i, $\mathcal{N}_i$, exists in A_k by $P_i = P\{N_i = j, \; s.t. \; j \in A_k\}$, the average waiting time for the next hop is given by

$$E_{wait}^C = e_{rx} \left\{ \sum_{i=1}^{N_P} \left[\left(\sum_{k=1}^{i-1} CW_k \right) + \frac{CW_i}{2} \right] P_i \right\} \quad, \qquad (12)$$

where

$$P_i = \left(1 - p_{[A(\gamma_{i-1}),\xi_{Th}]} \right) p_{[A(\gamma_i),\xi_{Th}]} \quad, \qquad (13)$$

$p_{[A(\gamma_i),\xi_{Th}]} = 1 - p_i$, p_i is given in (12) in [34], [35], e_{rx} is the energy consumption for receiving and γ_k is maximum distance from the sink for nodes in A_k. Using the same approach, the energy consumption of the receiver node can be found as shown in (14).

The last term in (10), E_{neigh}, is the energy consumed by the neighbors of the transmitter and the receiver nodes, which is expressed in (15). The first term in the parenthesis in (15) is the energy consumption of the neighbors of the transmitter node, which consume energy for RTS packet reception. The second term models the remaining neighbors of the receiver nodes that listen only to the CTS message sent by the receiver node.

Finally, the probability that a packet is received is given by [40]

$$p_s = \left(1 - \frac{1}{2} e^{-\frac{\xi}{1.28}} \right)^{16l} \qquad (16)$$

where Mica2 architecture is assumed with Manchester encoding, ξ is the received SNR, and l is the packet length in bits, which is l_C and l_D for p_s^C and p_s^D, respectively.

Using (11), (14), (15), and (16) in (10), the overall energy consumption of a flow can be found. Using numerical integration methods, the effect of distance, D, on the energy consumption of a flow is found and shown in Fig. 4. Considering Fig. 4, energy consumption of a flow is minimal for $\delta \sim 0.002$. However, in relatively small sized networks of < 1000 nodes, this operating point may not provide

$$E_{RX} = \frac{1}{(p_s^C)^3 p_s^D} \left\{ E_{rx}^R + E_{wait}^C + E_{tx}^C + E_{rx}^D + E_{rx}^A \right\} \tag{14}$$

$$E_{Neigh} = \frac{1}{(p_s^C)^2 p_s^D} \left\{ \rho\delta(\pi R_{inf}^2 - 2)p_s^C E_{rx}^R + \left(\rho\delta A(D, R_{inf}, D) - 2 \right)\left(E_{wait}^C + E_{rx}^C + \frac{E_{rx}^D}{2} \right) \right\} \tag{15}$$
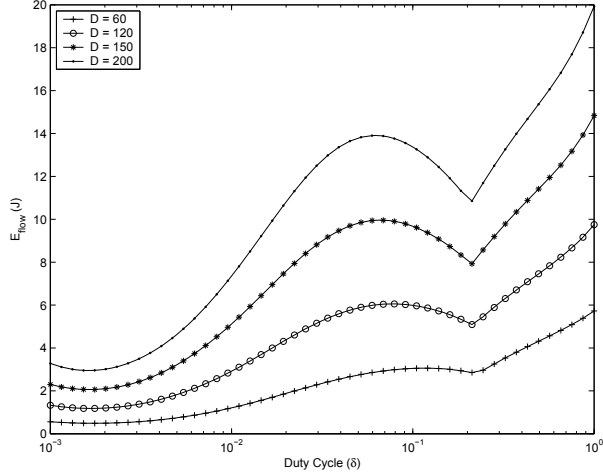

Fig. 4. Avg. energy consumption vs. duty cycle for different values of D.
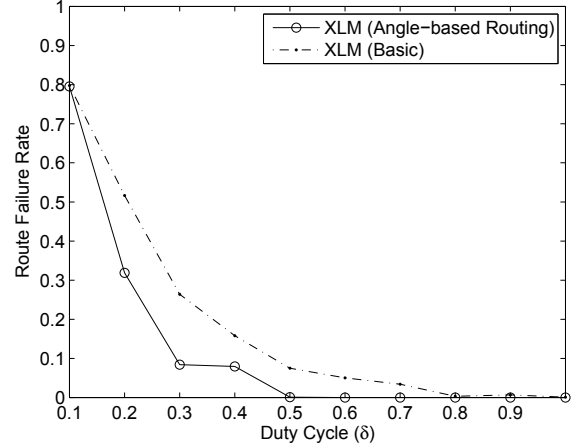

Fig. 5. Route Failure Rate for XLP with and without angle-based routing.

connectivity in the network. On the other hand, note also that the energy consumption has a local minima around $\delta = 0.2$. We will also show by comprehensive performance evaluations in Section IV that, this value is a suitable operating point for XLP in terms of energy efficiency.

IV. PERFORMANCE EVALUATION

To gain more insight into the protocol operation, we first investigate the effects of XLP parameters on the overall network performance. Then, we present a comparative study between XLP and five different layered protocol suites consisting of state-of-the-art protocols and a cross-layer protocol. Finally, we discuss the overall communication complexity of these solutions. We evaluate XLP and various protocol suites in a cross-layer simulator (XLS) developed at our laboratory in C++. XLS consists of a realistic channel model based on [40] and ns-2 and an event-driven simulation engine. The channel errors, packet collisions, and the energy consumption at the transceiver are accurately modeled based on ns-2. We present simulation results for a sensor topology of 300 nodes randomly deployed in a 100x100 m^2 sensor field. The sink is located at the coordinates (80,80). The simulation parameters for both sensor nodes and the communication suites are given in Table I. In each simulation, an event occurs in an *event area* located at coordinates (20,20) with an event radius of 20m. Each source node reports its event information to the sink. To investigate the effect of duty cycle, each simulation is performed for duty cycle values of $\delta \in [0.1, 1]$. Each simulation lasts for 300s and the average of ten trials for each of ten different random topologies are shown along with their 95% confidence intervals.

In the evaluations, we investigate the following performance metrics:

TABLE I
SIMULATION PARAMETERS

Parameter	Value	Parameter	Value
Re-tx. Limit	7	P_t	5 dBm
ν	2	PL(d_0)	55 dB
α	0.125	P_n	-105 dBm
Buffer Length	30	d_0	1m
$l_{control}$	20 bytes	n	3
l_{data}	100 bytes	σ	3.8
Frame Length	5s	$T_{coherence}$	16 ms
Energy Threshold	100 μJ	E_{rx}	13.5 mW
ξ_{Th}	10 dB	E_{tx}	24.75 mW
T_{sss}	5s	E_{sleep}	15 μW

- *Throughput* is the number of bits per second received at the sink. In calculating this metric, only unique packets are considered since multiple copies of a packet can be received at the sink for certain protocols.
- *Goodput* is the ratio between the total number of unique packets received at the sink and the total number of packets sent by all the source nodes. As a result, the overall communication reliability of the suites is investigated.
- *Energy Efficiency* is the most important metric in WSNs. We consider the average energy consumption per unique packet that is received at the sink, which can be considered as the inverse of energy efficiency. Hence, a lower value refers to a more energy efficient communication.
- *Number of Hops* is the number of hops each received packet traverses to reach the sink. This metric is used to evaluate the routing performance of each suite.
- *Latency* is the time it passes between the time a packet is generated at a source node and the time it is received at the sink. This delay accounts for the queuing delay and the contention delay at the nodes as well as specific protocol operation overhead.
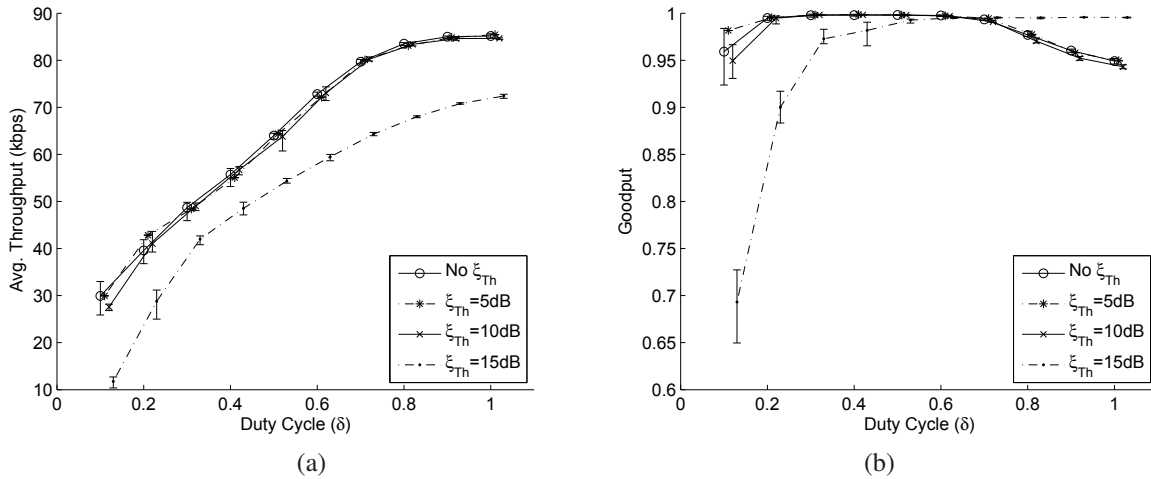
Fig. 6. (a) Average throughput and (b) average goodput vs. duty cycle for different values of ξ_{Th}.

A. XLP Parameters

The parameters that affect the XLP operation are the angle-based routing, SNR threshold, ξ_{Th}, and duty cycle, δ. We present the effects of these parameters on the XLP performance in this section.

The effect of angle-based routing is shown in Fig. 5, where the route failure rate vs. duty cycle parameter δ is shown for XLP with and without angle-based routing. In these experiments, a snapshot of the network is considered and the routes are found considering this topology. The route failure is the ratio of the number of unsuccessful routes between each nodes in the network and the number of all possible routes. The results show that route failure rate increases as the duty cycle parameter δ is decreased. On the other hand, angle-based routing limits the route failure rate to less than 10% for $\delta \geq 0.3$. This leads to up to 70% decrease in failure rate. Note that the failure rate of XLP with angle-based routing also increases as δ is further decreased since the probability that at any given time the network is partitioned increases.

In Fig. 6 (a), the total throughput received at the sink is shown. The x-axis shows the duty cycle, δ, and the throughput is shown for different SNR threshold, ξ_{Th}, values. The network throughput increases as the duty cycle, δ, is increased. An increase in the duty cycle results in an increase in the number of nodes that are active at a given time. Consequently, the capacity of the network increases. This fact is also evident from our buffer occupancy analysis in Section III-F. The effect of the SNR threshold, ξ_{th}, is also shown in Fig. 6 (a). The first curve in the figure, i.e., No ξ_{Th}, is the case where the first condition in (1) is not implemented. In other words, nodes contend for participating in routing irrespective of the received SNR value. It can be observed that, increasing the SNR threshold, ξ_{Th}, improves the network throughput upto a certain ξ_{Th}. Above this value, the network throughput degrades. This shows that a very conservative operation of XLP leads to performance degradation.

In Fig. 6 (b), the goodput performance is shown. Accordingly, XLP provides reliability above 90% for $\delta \geq 0.2$ and $\xi_{Th} \leq 10dB$. The decrease in goodput at $\delta = 0.1$, is due to the fact that the connectivity of the network cannot be maintained at all times. Moreover, for $\xi_{Th} = 15dB$, the goodput decreases
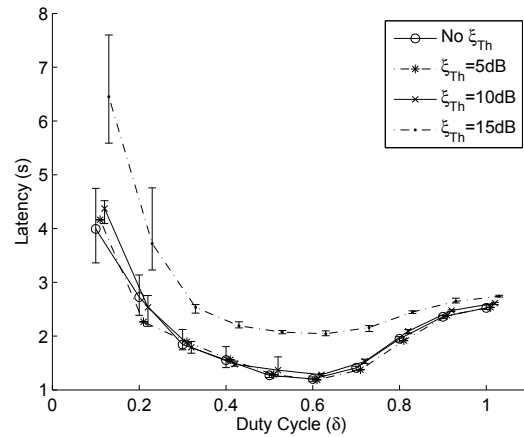

Fig. 7. Average latency vs. duty cycle for different values of ξ_{Th}.

to 0.7 as the duty cycle is decreased. This is due to the fact that potential receivers with the desired channel quality cannot be found and the reliability of the XLP degrades. For high duty cycle ($\delta > 0.7$), a slight decrease in goodput is observed for $\xi_{Th} < 15dB$. This accounts for the increased contention in the network since higher number of nodes are active for participation in routing at a given time. In the case for $\xi_{Th} = 15dB$, since fewer number of nodes are selected for contention in participation, collisions are limited and the goodput is not affected.

In Fig. 7, the end-to-end latency is shown, which reveals that increasing SNR threshold, ξ_{Th}, improves the end-to-end latency performance up to a certain ξ_{Th} value. $\xi_{Th} = 10dB$ results in the lowest latency. It is also interesting to note that there is a suitable operating point for duty cycle, δ, considering end-to-end latency ($\delta \sim 0.6$). Above this value, end-to-end delay starts to increase because of the increase in receiver based contention. Since for all above performance metrics, $\xi_{Th} = 10dB$ results in the most efficient performance, we use this value in the following comparative evaluations.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

IEEE TRANSACTIONS ON MOBILE COMPUTING

10

B. Comparative Evaluation

In this section, we compare the performance of the XLP with five different layered protocol architectures and a cross-layer protocol. We first identify the protocol configurations implemented in our study along with the implementation issues. Then, we present the results of our comparative evaluation for networks with and without duty cycle operation. The complexity issues are also discussed.

1) Protocol Configurations: The protocol configurations implemented for the comparative evaluation are as follows:

Flooding: This configuration serves as the baseline for the other configurations. Each node broadcasts its packet and the nodes that are closer to the sink rebroadcast this packet until it reaches the sink. At the MAC layer, CSMA type broadcast mechanism is used. No retransmission mechanism is used. At the transport layer, packets are injected at a constant rate and no rate control is used. The results shown include the unique packets received at the sink.

[GEO]: Geographical Routing + CC-MAC + ESRT: This protocol configuration is composed of ESRT [1], geographical routing [28], and CC-MAC [33] at transport, routing, and MAC layers, respectively. The CC-MAC protocol is implemented using $r_{corr} = 7m$, and $T_{SSS} = 5s$. For routing, *distance-based blacklisting* [28] is used such that the nodes in the farthest 20% of the radio range are blacklisted and the next closest node to the sink is selected as the next hop.

[PRR]: PRR-based Geographical Routing + CC-MAC + ESRT: This protocol configuration is similar to *GEO* with the exception that the routing decisions are based on the channel quality of each node with its neighbors. The channel quality is measured in terms of the packet reception rate (PRR) as discussed in [28]. The node that maximizes the *PRR×geographical advancement* product is selected as the next hop.

[PRR-SMAC]: PRR-based Geographical Routing + SMAC + ESRT: This protocol configuration is similar to *PRR* with the MAC layer replaced by the SMAC protocol [36]. In this configuration, the duty cycle operation proposed in [36] is implemented instead of the distributed duty cycle operation.

[DD-RMST]: Directed Diffusion + RMST: This case consists of RMST [30], directed diffusion [16], and CSMA scheme. The RMST protocol is implemented with hop-by-hop recovery and caching, and no link-layer ARQ is used at the link layer as presented in [30]. *DD-RMST* is used in the comparative evaluations for operation without duty cycle, i.e., $\delta = 1$.

[ALBA-R]: The cross-layer ALBA-R protocol [5] is also included in the analysis since it represents many of the state-of-the-art concepts in cross-layer communication in WSNs and has been shown to outperform the MACRO protocol [11]. Since ALBA-R does not consider rate control, three different traffic rates are investigated: $\lambda = \{3, 4, 6.25\}$ pkts/s with a Poisson traffic according to [5]. Maximum values for QPI and GPI are selected as 20 and 3, respectively, since 20 is the buffer length and 3 is also employed in [5]. The rainbow mechanism uses 4 colors and the results after this mechanism converges are shown.

XLP: Our proposed cross layer protocol (XLP) is implemented according to the protocol description in Section III with SNR threshold $\xi_{Th} = 10dB$.

It is important to note that the existing protocols that we have implemented in the layered suites are usually proposed considering only their related layers with reasonable assumptions about the other layers. As an example, in the geographical routing protocol [28], each node is assumed to know the locations of its neighbors. However, the actual implementation and operation of such an information exchange procedure is important especially when comparing these solutions to the proposed XLP solution. In that sense, XLP does not require that a node has location knowledge about each of its neighbors. Instead, each node participates in communication according to *its* own location. Consequently, such an explicit information exchange between each neighbor is not required in XLP. In conventional geographical routing protocols, however, this constitutes a major overhead in terms of both communication and storage. Moreover, since duty cycle is deployed in our solution, each neighbor of a node may not always be active. Hence, in order for each protocol to work together in the protocol suites, we have made the following implementation modifications.

Accordingly, in *GEO, PRR*, and *PRR-SMAC*, each node broadcasts a beacon to inform its position and the remaining time to sleep. This beacon is sent at the beginning of each sleep frame when a node wakes up. Each neighbor that receives this beacon determines that the specific node will be active for the duration specified in the beacon. In the case of *PRR* and *PRR-SMAC*, this beacon also serves as a channel quality indicator. To optimize the network performance, in *GEO* and *PRR*, the beacons are piggybacked if there is a packet in the queue. In *PRR-SMAC*, a pairwise cross-layering is used and the routing beacons are sent with the SYNC packets. Similarly, SYNC packets are piggybacked if there is a packet in the queue.

We have indicated that *DD-RMST* is used only for operation without duty cycle, i.e., $\delta = 1$, since neither directed diffusion nor RMST considers duty cycle operation [16], [30]. Therefore, the *DD-RMST* protocol configuration is evaluated only for $\delta = 1$ and shown as a single point in the figures for fairness and completeness of the evaluations.

We next present the results for operation with duty cycle, by changing the duty cycle δ from 0.1 to 1 in Section IV-B2. The lines for each protocol is shifted slightly from each other with respect to the original duty cycle values (x-axis) to clearly show the confidence intervals. We represent the results for the evaluated protocols selectively to improve clarity in the figures.

2) Results: In Fig. 8 (a), the throughput comparison is presented. The throughput achieved by XLP is up to 55% higher than that of the layered protocol suites. This shows the clear advantage of using a cross-layer approach that governs the functionalities of three traditional layers. In the layered protocol suites, the cross-layer information is not efficiently exploited for each functionality. For example, in *PRR* and *PRR-SMAC*, route selection is only performed based on location information and link quality, whereas the congestion level at a specific node is not considered. Another important result emerges in the comparison between XLP, *PRR*, and *PRR-SMAC*. *PRR* uses CC-MAC at the MAC layer. CC-MAC results in smaller number of nodes to send information as a representative of all the nodes in that area [33]. However, SMAC [36] does not exploit this property and all the nodes inside an event area send information to the sink. This results in almost 3-times increase in the number of source nodes.
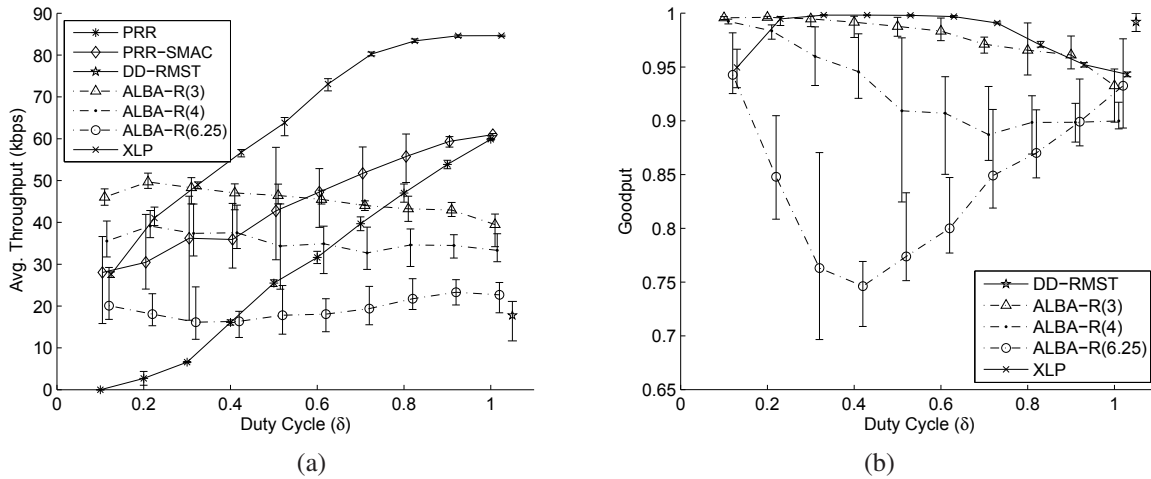
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

IEEE TRANSACTIONS ON MOBILE COMPUTING

11

Fig. 8. (a) Average throughput and (b) average goodput vs. duty cycle for different values of ξ_{Th}.

The higher throughput value of *PRR-SMAC* compared to *PRR* can be explained with this fact. However, XLP still outperforms *PRR-SMAC* in terms of total throughput although fewer number of nodes send information, which shows that the network capacity is exploited in a more efficient manner.

ALBA-R provides up to 40% higher throughput compared to XLP, when the duty cycle value and the data rate are low: $\delta \leq 0.2$ and $\lambda = 3$ pkts/s, respectively. When the duty cycle is low, the possibility of finding an active next hop at a given time interval decreases. Since ALBA-R is a more persistent protocol because of the longer QPI and GPI scanning mechanism, this protocol can establish routes when the traffic rate is low. However, since the network cannot accommodate higher rates at low duty cycle values and ALBA-R does not provide a rate control mechanism, throughput is lower for higher traffic rates. For duty cycle values $\delta > 0.3$, however, XLP more than doubles the througput achieved by ALBA-R. This is mainly because of the significant delay incurred at each hop because of the scanning mechanism. XLP, on the other hand, has a fixed delay for relay contention. Moreover, as the traffic rate increases from 3 to 6.25 pkts/s, the throughput of ALBA-R is halved. As the traffic load increases, the average QPI values of the nodes in the network increases. Consequently, a node spends larger amount of time for scanning the QPI values until a suitable neighbor is found. On the other hand, XLP employs an hop-by-hop congestion control mechanism to select nodes with lower queue occupancy in a shorter time. If the network becomes congested, traffic load is relieved through the source rate control mechanism.

Note that the total throughput achieved by the *DD-RMST* is significantly lower than XLP (~ 3.7 times), *PRR-SMAC* (~ 2.4 times), and *PRR* (~ 2.3 times), and comparable to *Flooding* (not shown [2]). This is due to two main reasons. First, the additional traffic created for recovering the lost packets increases the contention and decreases the capacity of the network. Second, the control packets of directed diffusion, i.e., the interest and exploratory packets, constitute a significant amount of traffic due to their broadcast nature.

The goodput is shown in Fig. 8 (b), where *DD-RMST*, *ALBA-R*, and XLP are shown since they consistently provide higher goodput than other protocols. Irrespective of the duty
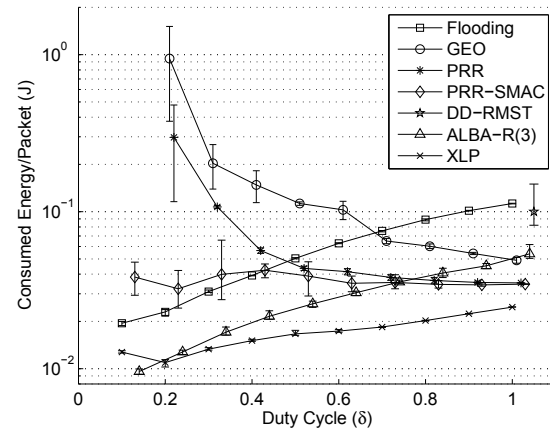

Fig. 9. Average energy consumption per packet vs. duty cycle for different values of ξ_{Th}.

cycle value, δ, both cross-layer protocols, XLP and ALBA-R ($\lambda = 3$) provide goodput higher than 90%. The cross-layer communication paradigm of the XLP that is adaptive to the network topology enables such high performance even when the network operates at low duty cycle. This can also be observed for the ALBA-R protocol. Coupled with the high throughput of XLP as shown in Fig. 8 (a), our cross-layer approach enables highly efficient communication. Moreover, *DD-RMST* provides 100% reliability while XLP and ALBA-R results in a reliability of 94% and 93%, respectively, for operation without duty cycle, i.e., $\delta = 1$. Note that RMST protocol uses hop-by-hop recovery with negative acknowledgments to request missing packets. On the other hand, XLP aims to first prevent link losses by constructing non-congested, high quality paths and then ensures high reliability by hop-by-hop ARQ technique. ALBA-R also employs a similar approach by selecting nodes with lower traffic load as next hop. This approach results in reliability comparable to RMST at a significantly lower cost as we will discuss next. ALBA-R also results in 5% higher goodput for $\delta = 0.1$ and $\lambda \leq 4$ pkts/s. This is related to the higher persistency of the protocol than
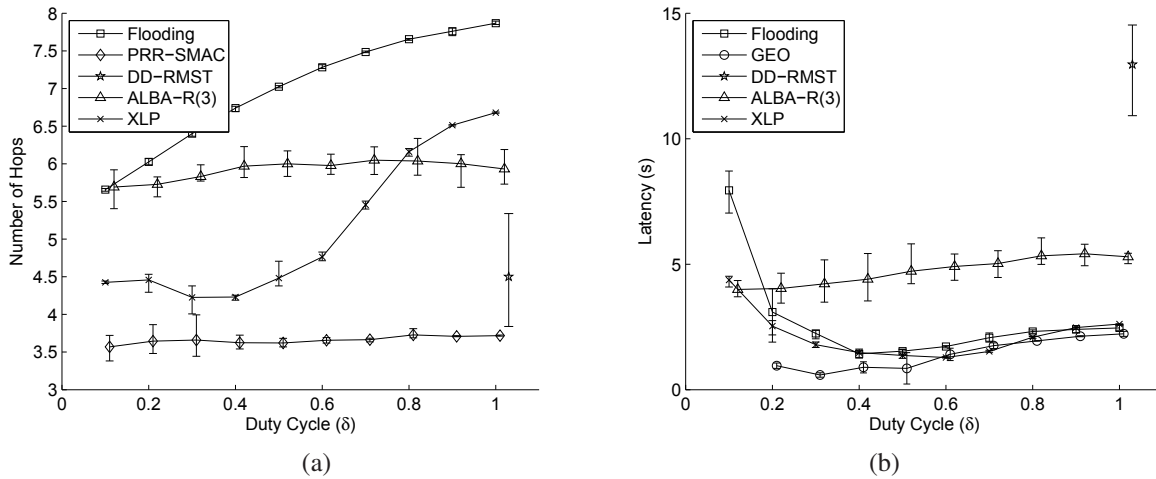
Fig. 10.   (a) Average hop count, and (b) average latency vs. duty cycle for layered protocol suites and XLP.

XLP. However, as the traffic load increases, goodput of ALBA-R decreases by 5%-25% because of the lack of congestion control and the corresponding buffer overflows. Since source nodes do not throttle the injected traffic, intermediate nodes start to drop these packets if they cannot find next hop with $QPI < QPI_{max}$. Since it results in the best performance, we show the results for $\lambda = 3$ pkts/s for ALBA-R in the following figures but note that it is an important challenge for the ALBA-R protocol to find the suitable traffic rate.

The decrease in reliability for the other layered protocol suites is mainly because of the significant number of packet drops due to retransmission timeouts [2].[5] This suggests that nodes cannot find their intended next hops due to either low channel quality or because the intended nodes switch to sleep state before receiving any packets. This is exacerbated especially in the case of low duty cycle. As a result, the reliability of the network is significantly hampered.

In Fig. 9, the energy consumption per packet is shown, where the values for *GEO* and *PRR* at $\delta = 0.1$ are not shown since no packets are received at the sink. XLP consumes 28%-66% less energy per packet compared to the most energy efficient layered protocol suite, *PRR-SMAC*. Similarly, XLP is 14%-54% more energy efficient compared to ALBA-R for duty cycle values ($\delta \geq 0.2$). This difference is mainly because of the periodic broadcast of beacon packets in *GEO* and *PRR*, and SYNC packets in *PRR-SMAC*. Furthermore, the significant percentage of retransmission timeouts indicate significant energy wastage due to packets that cannot be transmitted to the sink. Since the network and MAC layers operate independently, a node chosen by the routing layer cannot be reached by the MAC protocol immediately due to the duty cycle operation. This delay leads to increased energy consumption.

An interesting result is the 3-fold higher energy consumption of *DD-RMST* compared to XLP. Although this configuration provides 100% reliability as shown in Fig. 8 (b), the layered structure of the routing, transport, and MAC functionalities results in a high penalty. The routing layer, i.e., directed diffusion, incurs significant amount of overhead in

order to maintain end-to-end paths between sources and the destination. On the contrary, XLP employs an adaptive routing technique that provides an energy efficient path in terms of both link quality and energy consumption distribution.

ALBA-R results in 32% lower energy consumption for $\delta = 0.1$ compared to XLP. However, for increasing δ, energy consumption is up to 54% higher than XLP. As the duty cycle increases, the number of active nodes increases. As a result, due to continuous polling of QPI values, the neighbors of a node consume higher amount of energy. Furthermore, since the probability of finding nodes with the same QPI increases, these nodes also go through GPI scanning. Although ALBA-R aims to transmit multiple packets after a next hop is found, the cost of finding the next hop is higher. Instead, XLP resolves contention by the integration of next hop determination mechanism to the backoff mechanism of the contending nodes, which results in higher energy efficiency. Another important observation from Fig. 9 is that the energy consumption per packet for XLP has a minimum at $\delta = 0.2$. This is consistent with the mathematical analysis provided in Section III-G.

The advantages of using a separate routing layer in the layered protocol suites can be seen from Fig. 10 (a), where the average hop count is shown. *GEO, PRR, PRR-SMAC*, and *DD-RMST* result in fewer number of hops than XLP. This is due to the fact that the routing algorithms in these layered protocol suites aim to find the smallest number of hops. Moreover, XLP results in 2%-12% more number of hops than ALBA-R when the duty cycle is high ($\delta \geq 0.8$). This result may be incorporated as a disadvantage of XLP when only routing layer is taken into account. However, the overall performance of XLP reveals that maximizing routing layer performance alone does not provide efficient communication in WSNs. In other words, while smaller number of hops might be optimal in terms of *routing* efficiency, other effects such as link quality, contention level, congestion level, and overall energy consumption necessitate a cross-layer approach in route selection for *overall* efficiency.

Furthermore, as shown in Fig. 10 (b), XLP provides end-to-end latency comparable to *Flooding* and *GEO*. Furthermore, ALBA-R results in up to ~ 2.8 times higher latency, which is also reported in [5]. Fig. 10 (b) also clearly shows the tradeoff

---

[5]Not shown here for space considerations.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

IEEE TRANSACTIONS ON MOBILE COMPUTING

13

of *DD-RMST* in achieving high reliability. This configuration results in significantly high latency values due to end-to-end reliability mechanism. Note that the end-to-end latency of XLP more than doubles for low δ. This is due to the fact that sender nodes cannot find any neighbors that satisfy the constraints in (1) discussed in Section III. Nevertheless, this is still within 8% of the latency provided by ALBA-R.

3) Implementation Complexity: In addition to the performance of our XLP module in terms of network metrics, the complexity and implementation issues of cross-layer design are also important. In this section, we provide a qualitative comparison of cross-layer design and layered protocol architectures implemented in our simulation environment.

One of the major advantages of cross-layer design for communication protocols is the implementation efficiency. In a traditional layered protocol architecture, each layer has clear boundaries. This layered structure leads to computation delays due to the sequential handling of a packet. For example, in TinyOS [41], each layer has to wait for the lower layers to process the packet since a single buffer is used for a packet for all layers. XLP, however, blends the functionalities of traditional medium access, routing, and congestion control into a unified cross-layer communication module by considering physical layer and channel effects. Hence, these functionalities are performed as a whole and overall protocol efficiency can be improved using XLP.

In addition to the simulation performance, the implementation issues are also important for a complete comparison. As explained in Section III, XLP does not require any tables or extra buffer space for routing and congestion control functionalities. The routing is performed based on receiver initiatives, which eliminates the need for a routing table at each node. The implementation of XLP is both simple and compact. On the other hand, in *PRR-SMAC*, *SMAC* protocol maintains schedule table for each of one-hop neighbors to provide synchronized sleeping cycles. Similarly, in *DD-RMST*, at the routing layer, each node has to implement reinforcement table for each source indicating the next hop in the reinforced path. In case a node is a source node, it also has to keep track of multiple neighbors which has a path to the sink for exploratory messages. At the transport layer, RMST [30] requires a separate queue to cache data locally to support loss recovery at all hops. These requirements, due to either layered operation of the protocol stack or the internal protocol structure at each layer, places burden in memory space for communication in sensor nodes. Compared to the layered protocol stacks, ALBA-R requires significantly smaller code space because of the cross-layer MAC/routing operation. However, compared to XLP, ALBA-R necessitates large state information to be stored in each node because of the QPI and GPI scanning process.

The extra space required by the communication stacks limits the available space to develop new applications for sensor networks. On the other hand, the careful use of code space and cross-layer implementation of communication functionalities in XLP provides a more efficient operation in WSNs. When coupled with the noticeably better communication performance as discussed in IV-B2, XLP becomes a successful candidate for communication protocols in WSNs.

V. CONCLUSION

Recently, cross-layering in designing a communication stack such that state information flows throughout the stack has been investigated. Recent work on WSNs [13], [15] also reveals that cross-layer integration techniques result in significant energy gains. In this paper, we propose a novel *initiative determination* concept that allows many communication and networking functionalities be implemented in a single protocol. Accordingly, the cross-layer protocol (XLP) is proposed to provide the functionalities of medium access, routing, and congestion control. Based on the initiative determination concept, XLP serves as a proof of concept and performs receiver-based contention, initiative-based forwarding, local congestion control, and distributed duty cycle operation to realize efficient and reliable communication in WSNs. Analytical performance evaluation and simulation experiment results show that XLP significantly improves the communication performance and outperforms the traditional layered protocol architectures in terms of both network performance and implementation complexity.

The ultimate goal in the cross-layer design techniques is to develop a single communication module that is responsible for the functionalities of each networking layer. The initiative determination concept developed in this work is the first step in this approach to replace the entire traditional layered protocol architecture that has been used so far in WSNs so that both the information and the functionalities of traditional communication layers are blended in a single module. Consequently, the future work for our research include the investigation of various networking functionalities such as adaptive modulation, error control, and topology control in a cross-layer fashion to develop a unified cross-layer communication module.

REFERENCES

[1] Ö. B. Akan and I. F. Akyildiz, "Event-to-Sink Reliable Transport in Wireless Sensor Networks," *IEEE/ACM Transactions on Networking*, vol. 13, no. 5, pp. 1003-1016, October 2005.
[2] I. F. Akyildiz, M. C. Vuran, and O. B. Akan, "A Cross-layer Protocol for Wireless Sensor Networks," in *Proc. Conference on Information Science and Systems (CISS '06)*, Princeton, NJ, March 22-24, 2006.
[3] P. Casari, M. Nati, C. Petrioli and M.Zorzi, "ALBA: an Adaptive Load-Balanced Algorithm for Geographic Forwarding in Wireless Sensor Networks," in *Proc. IEEE Military Communications Conference (MILCOM'06)*, Washington DC, October 2006.
[4] P. Casari, M. Nati, C. Petrioli, and M. Zorzi, "Geographic Forwarding and Adaptive Load Balancing in Wireless Sensor Networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, p. 53-54 (extended abstracts of the best ACM MobiCom06 posters), vol. 11, issue 2, April 2007.
[5] P. Casari, M. Nati, C. Petrioli, and M.Zorzi, "Efficient Non Planar Routing around Dead Ends in Sparse Topologies using Random Forwarding," in *Proc. IEEE International Conference on Communications (ICC'07)*, Scotland, UK, June 2007.
[6] D. Chen and P.K. Varshney, "A Survey of Void Handling Techniques for Geographic Routing in Wireless Networks," *IEEE Communications Surveys and Tutorials*, vol. 9, no. 1, pp. 50-67, 2007.
[7] M. Chiang, "To Layer or Not To Layer: Balancing Transport and Physical Layers in Wireless Multihop Networks," in *Proc. IEEE INFOCOM '04*, vol. 4, pp. 2525 - 2536, March 2004.
[8] M. Chiang, "Balancing transport and physical Layers in wireless multihop networks: jointly optimal congestion control and power control," *IEEE JSAC*, vol. 23, no. 1, pp. 104-116, Jan. 2005.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

IEEE TRANSACTIONS ON MOBILE COMPUTING

14

[9] S. Cui, R. Madan, A. Goldsmith, S. Lall, "Joint routing, MAC, and link layer optimization in sensor networks with energy constraints," in *Proc. IEEE ICC '05*, vol. 2, pp. 725 - 729, May 2005.

[10] S. De, et.al., "An integrated cross-layer study of wireless CDMA sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 7, pp. 1271 - 1285, Sep 2004.

[11] D. Ferrara, L. Galluccio, A. Laonardi, G. Morabito, and S. Palazzo, "MACRO: An integrated MAC/routing protocol for geographical forwarding in wireless sensor networks," in *Proc. IEEE INFOCOM '05*, Miami, FL, March 2005.

[12] M. Fraser, E. Kranakis, J. Urrutia, "Memory Requirements for Local Geometric Routing and Traversal in Digraphs," in *Proc. CCCG '08*, Montreal, Canada, Aug. 2008.

[13] V. C. Gungor, M. C. Vuran, and O. B. Akan, "On the cross-layer Interactions between Congestion and Contention in Wireless Sensor and Actor Networks," *Ad Hoc Networks Journal (Elsevier)*, vol. 5, no. 6, pp. 897-909, August 2007.

[14] J. Haapola, Z. Shelby, C.A. Pomalaza-Ráez, and P. Mähönen, "Cross-layer energy analysis of multi-hop wireless sensor networks," in *Proc. European Workshop on Wireless Sensor Networks*, February 2, 2005.

[15] L. van Hoesel, T. Nieberg, J. Wu, and P. J. M. Havinga, "Prolonging the lifetime of wireless sensor networks by cross-layer interaction," *IEEE Wireless Communications*, vol. 11, no. 6, pp. 78 - 86, Dec. 2004.

[16] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, F. Silva, "Directed diffusion for wireless sensor networking," *IEEE/ACM Trans. on Networking*, vol. 11, no. 1, pp. 2 - 16, February 2003.

[17] B. Karp and H. T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *Proc. ACM/IEEE MobiCom '00*, pp. 243 - 254, Boston, MA, Aug. 2000.

[18] B. Leong, S. Mitra, B. Liskov, "Path vector face routing: geographic routing with local face information," in *Proc. IEEE ICNP '05*, Nov. 2005.

[19] X. Lin, N. B. Shroff, and R. Srikant, "A Tutorial on Cross-Layer Optimization in Wireless Networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, pp. 1452- 1463, June 2006.

[20] X. Lin and N. B. Shroff, "The Impact of Imperfect Scheduling on Cross-Layer Congestion Control in Wireless Networks," *IEEE/ACM Trans. on Networking*, vol. 14, no. 2, pp. 302 - 315, April 2006.

[21] S. Liu, K.-W. Fan, and P. Sinha, "CMAC: An energy efficient MAC layer protocol using convergent packet forwarding for wireless sensor networks," in *Proc. SECON '07*, San Diego, CA, June 2007.

[22] M. Mastrogiovanni, C. Petrioli, M. Rossi, A. Vitaletti, and M. Zorzi, "Integrated data delivery and interest dissemination techniques for wireless sensor networks," in *Proc. IEEE Globecom 2006*, pp. 1-6, Nov. 2006.

[23] D. Moore, J. Leonard, D. Rus, and S. Teller, "Robust Distributed Network Localization with Noisy Range Measurements," in *Proc. ACM Sensys '04*, Baltimore, MD, Nov. 2004.

[24] P. R. Morin, "Online Routing in Geometric Paths," *Ph.D. Thesis*, School of Computer Science, Carleton University, 2001.

[25] D. Pompili, M. C. Vuran, and T. Melodia, "Cross-layer Design in Wireless Sensor Networks," in *Sensor Network and Configuration: Fundamentals, Techniques, Platforms, and Experiments, Edited by M. P. Mahalik*, Springer-Verlag, August 2006.

[26] M. Rossi, M. Zorzi, "Cost Efficient Localized Geographical Forwarding Strategies for Wireless Sensor Networks," in *Proc. TIWDC 2005*, Sorrento, Italy, 2005.

[27] L. Savidge, H. Lee, H. Aghajan, and A. Goldsmith, "Event-driven geographic routing for wireless image sensor networks," in *Proc. COGIS '06*, Paris, March 2006.

[28] K. Seada, M. Zuniga, A. Helmy, B. Krishnamachari, "Energy-efficient forwarding strategies for geographic routing in lossy wireless sensor networks,", in *Proc. ACM Sensys '04*, November 2004.

[29] P. Skraba, H. Aghajan, A. Bahai, "Cross-layer optimization for high density sensor networks: Distributed passive routing Decisions," in *Proc. Ad-Hoc Now'04*, Vancouver, July 2004.

[30] F. Stann and J. Heidemann, "RMST: Reliable data transport in sensor networks," in *Proc. IEEE SNPA '03*, pp. 102-112, Anchorage, Alaska, USA, April, 2003.

[31] J. Urrutia, "Local solutions for global problems in wireless networks," *J. of Discrete Algorithms*, no. 5, vol. 3, pp. 395-407, Sep. 2007.

[32] M. C. Vuran, Ö. B. Akan, and I. F. Akyildiz, "Spatio-Temporal Correlation: Theory and Applications for Wireless Sensor Networks," Computer Networks Journal (Elsevier), Vol. 45, No. 3, pp. 245-261, June 2004.

[33] M. C. Vuran and I. F. Akyildiz, "Spatial Correlation-based Collaborative Medium Access Control in Wireless Sensor Networks," *IEEE/ACM Transactions on Networking*, vol. 14, no. 2, pp. 316 -329, April 2006.

[34] M. C. Vuran and I. F. Akyildiz, "Cross-Layer Analysis of Error Control in Wireless Sensor Networks," in *Proc. IEEE SECON '06,* Reston, VA, September 25-28, 2006.

[35] M. C. Vuran and I. F. Akyildiz, "Error Control in Wireless Sensor Networks: A Cross Layer Analysis," *IEEE/ACM Transactions on Networking*, vol. 17, no. 4, pp. 1186-1199, Aug. 2009.

[36] W. Ye, J. Heidemann, and D. Estrin, "Medium Access Control with Co-ordinated Adaptive Sleeping for Wireless Sensor Networks," *IEEE/ACM Transactions on Networking*, vol. 12, no. 3, pp. 493-506, June 2004.

[37] M. Zorzi, "A new contention-based MAC protocol for geographic forwarding in ad hoc and sensor networks," in *Proc. IEEE ICC '04*, vol. 6, pp. 3481 - 3485, June 2004.

[38] M. Zorzi, R. Rao, "Geographic random forwarding (GeRaF) for ad hoc and sensor networks: multihop performance," *IEEE Trans. Mobile Computing*, vol. 2, no. 4, pp. 337-348, Oct.-Dec. 2003.

[39] M. Zorzi, R. Rao, "Geographic random forwarding (GeRaF) for ad hoc and sensor networks: energy and latency performance," *IEEE Trans. Mobile Computing*, vol. 2, no. 4, pp. 349-365, Oct.-Dec. 2003.

[40] M. Zuniga, B. Krishnamachari, "Analyzing the transitional region in low power wireless links," in *Proc. IEEE SECON '04*, pp. 517 - 526, Oct. 2004.

[41] TinyOS, [online], *http://www.tinyos.net/*.

Mehmet C. Vuran (M'98) received his B.Sc. degree in electrical and electronics engineering from Bilkent University, Ankara, Turkey, in 2002. He received his M.S. and Ph.D. degrees in electrical and computer engineering from School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, under the supervision of Prof. Ian F. Akyildiz in 2004 and 2007, respectively.

Currently, he is an Assistant Professor in the Department of Computer Science and Engineering at the University of Nebraska-Lincoln. His current research interests include cross-layer design, wireless sensor networks, underground sensor networks, cognitive radio networks, and deep space communication networks.

Ian F. Akyildiz (M'86-SM'89-F'96) received the B.S., M.S., and Ph.D. degrees in Computer Engineering from the University of Erlangen-Nuernberg, Germany, in 1978, 1981 and 1984, respectively.

Currently, he is the Ken Byers Distinguished Chair Professor with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, and Director of Broadband and Wireless Networking Laboratory. He is an Editor-in-Chief of *Computer Networks Journal (Elsevier)* as well as the founding Editor-in-Chief of the *AdHoc Network Journal (Elsevier)*. His current research interests are in next generation wireless networks, sensor networks, IPN Internet, wireless networks, and satellite networks.

He received the "Don Federico Santa Maria Medal" for his services to the Universidad of Federico Santa Maria, in 1986. From 1989 to 1998, he served as a National Lecturer for ACM and received the ACM Outstanding Distinguished Lecturer Award in 1994. He received the 1997 IEEE Leonard G. Abraham Prize Award (IEEE Communications Society) for his paper entitled "Multimedia Group Synchronization Protocols for Integrated Services Architectures" published in the IEEE Journal of Selected Areas in Communications (JSAC) in January 1996. He received the 2002 IEEE Harry M. Goode Memorial Award (IEEE Computer Society) with the citation "for significant and pioneering contributions to advanced architectures and protocols for wireless and satellite networking". He received the 2003 IEEE Best Tutorial Award (IEEE Communication Society) for his paper entitled "A Survey on Sensor Networks," published in IEEE Communications Magazine, in August 2002. He also received the 2003 ACM Sigmobile Outstanding Contribution Award with the citation "for pioneering contributions in the area of mobility and resource management for wireless communication networks". He received the 2004 Georgia Tech Faculty Research Author Award for his "outstanding record of publications of papers between 1999-2003". He also received the 2005 Distinguished Faculty Achievement Award from School of ECE, Georgia Tech. He has been a Fellow of the Association for Computing Machinery (ACM) since 1996.