Maximum-Lifetime Multi-Channel Routing in Wireless Sensor Networks

Amitangshu Pal and Asis Nasipuri
Electrical & Computer Engineering, The University of North Carolina at Charlotte, Charlotte, NC 28223-0001
E-mail:{apal,anasipur}@uncc.edu

Technical Report
September, 2012

Abstract- **In this paper we consider the joint channel assignment and routing problem in multi-channel wireless sensor networks for maximizing the worst case network lifetime. We assume a data collection traffic pattern where all sensor nodes forward data to a centralized base station (sink). The proposed solutions lead to a tree rooted at the sink, comprising of sub-trees that operate over distinct channels. We prove that this channel assignment and routing problem is *NP-complete*, and present some distributed approaches as well as a centralized solutions. to address this problem. We perform extensive simulation studies that show that our proposed channel assignment and route selection schemes perform significantly better than random channel selection as well as a previously reported solution for the problem.**

Keywords: **Wireless sensor networks, multi-channel routing, distributed algorithms.**

I. INTRODUCTION

Wireless sensor networks (WSNs) are becoming increasingly polular for a wide range of potential applications, ranging from environmental monitoring to a large number of industrial and military applications. Wireless sensor nodes are equipped with an integrated low power processor, memory and a radio and usually operate on batteries. Since batteries are difficult to replace, reducing the energy consumption in the sensor nodes is a key concern for designing sensor networks. A significant amount of power is consumed by the radio for wireless communication and by the sensor boards for sensing physical quantities. Consequently, maximizing the lifetime of WSNs requires that radio transmissions and receptions are minimized. The complexity of this energy optimization problem in data collection sensor networks arises due to the fact that it has to be addressed by *network wide* adaptations as opposed to independent adaptations at the nodes.

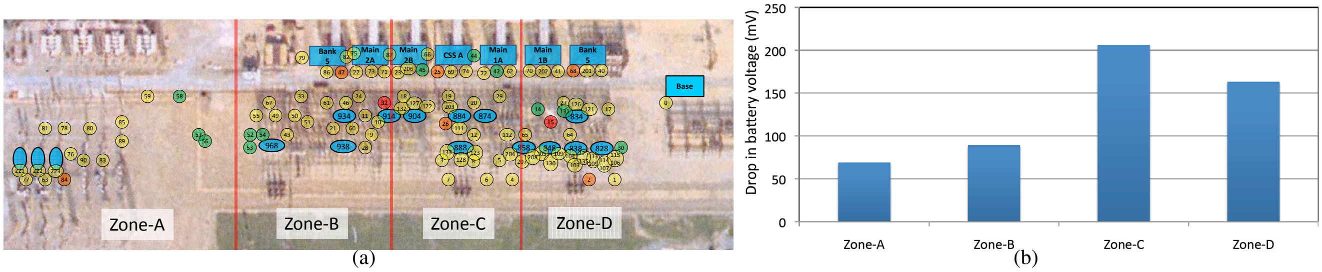
A lot of effort has been directed in the networking community to design routing protocols that address the energy conservation issue. A number of energy aware routing protocols are proposed [1], [2] on single-channel sensor networks. Unfortunately, in single-channel sensor networks, energy wastage due to *overhearing* from other senors is a critical factor. Usage of multiple orthogonal channels can alleviate the overhearing problem. Using multiple channel also helps in reducing interfernce in the network that improves the communication performance. Current WSN hardwares such as MICAz and Telos that use CC2420 radio, provide multiple channels that can help in reducing the overhearing problem. However, designing effective mechanisms to dynamically select channels is a key issue that requires attention. In this paper, we develop energy aware routing for wireless sensor networks in presence of multiple channels. In contrast to the current multi-channel protocols in WSNs that mainly target to reduce interference, our main objective in this paper is to enhance the network lifetime through efficient route and channel selection.

The rest of the paper is organized as follows. In section II, we summerize the related work. Section III describes our motivations behind this work. Section IV describes the problem formulation of our multi-channel routing scheme and our method of calculation network lifetime using some approximations. In section V, we describe a number of distributed multi-channel routing schemes, whereas a centralized scheme is described in section VI. Simulation results of our proposed routing schemes are shown in section VII. We conclude our paper section VIII.

II. RELATED WORK

Multi-channel routing in wireless networks has received a lot of attention in recent times [3], [4], [5], [6], [7], [8]. However, most of the work published in this area either assume a multi-radio transceiver at each node or generate high control overhead for channel negotiation. These schemes are not suitable for WSNs where each sensor is typically equipped with single radio transceiver. In addition, overhead must be minimized since energy resources are at a premium. For the sake of discussion, we classify existing literature on multi-channel MAC protocols into three categories: scheduled multi-channel schemes, contention based multi-channel schemes and hybrid schemes. These are discussed in the following:

Scheduled multi-channel schemes: In scheduled multi-channel scheme, each node is assigned a time slot for data transmission that is unique in it's 2-hop neighborhood. In [9], the authors propose such a scheme name TFMAC where time is partitioned in contention access period and contention



(a) (b)

Fig. 1. (a) Locations of nodes (circles) in ParadiseNet [10], which uses a single-channel link quality based routing protocol. The base station is marked by a blue rectangle in the right.(b) Average battery voltage drops in the four marked zones over five months of oepration.

free period. In the contention access period, nodes exchange control messages in a default channel and then in contention free period, the actual data transmission takes place.

Contention based multi-channel schemes: In [11], the authors propose *Multi-frequency media access control for wireless sensor networks (MMSN)* where time is divided in time slots. Each slot consists of a broadcast contention period and a transmission period. Each node has an assigned receiving frequency. During the broadcast contention period, nodes compete for the same broadcast frequency and during the transmission period, nodes compete for shared unicast frequencies. Authors in [12] propose a *TDMA based multi-channel MAC (TMMAC)* where time is divided into some beacon intervals that consist of an ATIM window and a communication window. In the ATIM window, all nodes listen to the same default channel and the sender and receiver decide on which channel and which slot to use for data transmission. Then in each slot of the communication window, each node adopts the negotiated frequency to transmit and receive packets. In [13], authors propose a *Multi-channel MAC (MMAC)*, where each sensor node notifies it's cluster-head if it wants to transmit. Next the cluster-head distributes the channel assignment information to the sources and destinations.

Hybrid schemes: These protocols combine the principles of scheduled and contention based approaches. In [14], the authors propose a TDMA-based multi-channel MAC protocol. The scheme allocates a time slot to each receiving node, where each slot consists of a contention window and a window for data transmission. A sender first contends for getting access to the channel in the contention window and then the winner transmits in the remaining slot. The scheme uses channel-hopping to take advantage of multiple channels.

All these MAC protocols have the following disadvantages: Firstly, they need precise time synchronization which is hard to obtain in WSNs. Secondly, for multi-hop networks, all these MAC protocols require nodes to switch channels to receive and forward packets that can cause very frequent channel switching. Recently, some channel assignment strategies are proposed in [15], [16], [17] for multi-hop routing in WSNs. In [15], the authors propose a Tree-based multichannel protocol (TMCP) where the whole network is statically divided into mutually exclusive single-channel subtrees to reduce interference. Authors in [16] propose a control theory approach that selects channel dynamically to achieve load balancing among channels, whereas in [17] authors propose a channel assignment scheme for WSNs based on game theory to reduce interference. All of the above schemes mainly consider reducing network interference. Interference is proportional to packet size as well as packet interval. Generally in WSNs the packet size as well as packet interval is small, thus interference is of secondary importance for WSNs. The primary concern in WSNs is increasing battery life of the network that is the main contribution of this paper. Also, all the above approaches are either centralized or need the topology information that is not always possible to obtain in WSNs. In this paper we address the problem of channel assignment and routing together for improving battery lifetime in WSNs. We develop a number of distributed as well as centralized routing and channel assignment schemes and evaluated their performance using simulations.

III. MOTIVATION BEHIND THIS WORK

Radio transmissions as well as receptions are the critical energy-consuming tasks in typical low-powered wireless sensor nodes. For instance, the MICAz nodes draw about $20mA$ of current while transmitting and receiving, whereas it draws about 20 μA in idle mode and $1\mu A$ in sleep mode. Hence, a key aspect of designing energy-efficient wireless sensor nodes is to minimize the radio active periods, allowing the node to sleep as long as possible. Popular energy efficient wireless sensor networking protocols such as $XMesh$ [10] employs low-power (LP) operation by letting nodes duty cycle in their sleep modes for brief periods of time to detect possible radio activity and wake up when needed. While this principle extends the battery life (lifetime) of the nodes considerably, a key factor that leads to energy wastage is *overhearing*, i.e. receiving packets that are intended for other nodes in the neighbourhood. The traditional mechanism to avoid overhearing is scheduling, which requires time synchronization that we assume is absent in the WSNs.

To depict the effect of overhearing in a WSN that does not use scheduling, we present experimental observations from a WSN testbed that was developed by the authors for health monitoring of high-power equipment in a power substation in Figure 1. The WSN, called *PradiseNet* [18], consists of 122 wireless sensor nodes that were deployed in 1000×400 feet area, and uses a link-quality based routing protocol.

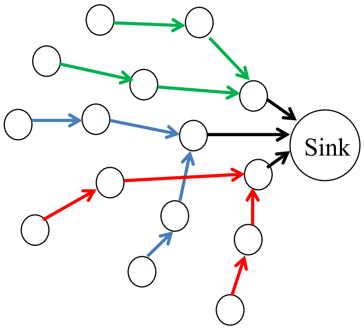


Fig. 2. A multi-channel tree for WSNs

Figure 1(a) depicts the location of nodes in *ParadiseNet* and Figure 1(b) depicts the average drops in the battery levels in the four regions of the network over a period of five months of operation. It can be observed that while nodes closer to the base station generally have higher voltage drops, Zone-C has the highest drop, conceivably due to higher overhearing effects, since it has the highest node density among all zones. Consequently, a mechanism to optimally distribute the network traffic over multiple channels would lead to significant improvement in the lifetime of the network.

IV. MULTI-CHANNEL ROUTING IN WSNS

We consider *data collecting* wireless sensor networks where all nodes sense some paramemers and forward them to the sink. This forwarding scheme follows a tree structure connecting the nodes to the sink. With a single channel, a node overhears all nodes that are in the receiving range of that node. To cope with this, in this paper we propose the idea of multi-channel tree. Let us define the nodes that are immediate neighbors of the sink as first-level nodes. Thus, for F first-level nodes, the multi-channel tree partitions the whole network in F vertex-disjoint subtrees all rooted at any of the F first-level nodes. All first-level nodes choose any of the K available channels and all their children transmit on the same channel. The sink is tuned to a default channel and all the first-level nodes switch to that channel only when they want to transmit to the sink, otherwise they stay on their chosen channel. Figure 2 depicts a multi-channel tree where different channels are shown in different colors. This multi-channel scheme reduces overhearing and thus increases the battery life of the whole network. At the same time using multiple-channel reduces the interference as well as contention delay of the network.

A. Lifetime Calculation

We assume that each sensor node has finite electrical energy, which is determined by the capacity of the onboard battery. Based on the experimentally validated model developed in [18], we represent the estimate the average current consumption in a node by

$$\begin{aligned}\mathcal{I} &= \frac{I_{Rt}T_{Rt}}{T_{rui}} + \frac{I_{Dt}T_{Dt}}{T_D} + O\left(\frac{I_{Rr}T_{Rr}}{T_{rui}} + \frac{I_{Dr}T_{Dr}}{T_D}\right) \\ &+ F\frac{I_{Dt}T_{Dt}}{T_D} + \frac{I_sT_s}{T_D} + 8I_PT_P \qquad (1)\end{aligned}$$

where I_x and T_x represent the current drawn and the duration, respectively, of the event x; O is the number of neighbors of the mote; and T_{RUI} and T_D represent the route update and data intervals, respectively. Transmission/reception of route uptate packets is denoted by R_t/R_r, data transmit/receive is denoted by D_t/D_r and processing and sensing are denoted as P and S, respectively. F is the number of nodes whose packets are forwarded by the test node.

With this the lifetime of a mote can be calculated as $L = \frac{B}{\mathcal{I}}$ where B is the initial capacity of the battery. In this paper we define the lifetime of a network as the time until the first node depletes it's energy, i.e. the *worst case network lifetime (WNL)*. Thus, the WNL can be expressed as $L = \min(L_1, L_2, ..., L_V)$ where $L_1, L_2, ..., L_V$ are the lifetime of the sensor nodes respectively.

B. Problem Formulation

Our objective is to maximize the worst case network lifetime L, subject to the following constraints:

One parent constraint: This constraint states that each node has one parent. If N_i is the set of neighbors of i, then

$$\sum_{j=1}^{|N_i|} x_i^j = 1 \qquad (2)$$

where x_i^j is a binary variable that is 1 when j is the parent of i and 0 otherwise.

Directionality constraint: A child sends packets to only its parent, thus

$$x_i^j + x_j^i \leq 1 \qquad (3)$$

Connectivity constraint: Note that j is a parent of i if there is a connection between i and j in the connectivity graph, i.e.,

$$x_i^j \leq W_i^j \qquad (4)$$

where W_i^j is 1 if j can be a parent of i and 0 otherwise.

Flow constraint: The rate of flow at i, denoted by F_i is given by the rate at which node i generates packets plus the rate at which it's children send packets, i.e.,

$$F_i = c + \sum_{j=1}^{|N_i|} x_j^i F_j \qquad (5)$$

where c is the rate at which each node sends packets.

Overhearing constraint: The amount of overhearing traffic at node i, denoted by O_i is the total amount of traffic from all the nodes that are in the overhearing range of i and in the same channel of i, i.e.,

$$O_i = \sum_{j=1, j\neq i}^{|N_i|} Y_i^j F_j U_i^j \qquad (6)$$

where Y_i^j is a binary variable that is 1 if i is in overhearing range of j and 0 otherwise and U_i^j is 1 if i and j are in the same channel and 0 otherwise. If C_i is the channel chosen by i, then U_i^j can be written as:

$$\begin{aligned} |C_i - C_j| &\leq M(1 - U_i^j) \\ 1 - U_i^j &\leq |C_i - C_j| \end{aligned} \tag{7}$$

where M is a very large number.

Parent-child constraint: If j is the parent of i, then the channel of i is same as the channel of j, i.e.,

$$C_i = \sum_{j=1}^{|N_i|} C_j x_i^j \tag{8}$$

Energy constraint: Also the total energy spent by node i cannot be more than the residual energy (B_i) of that node, i.e.,

$$\begin{aligned} L & \left(\frac{I_{Rt}T_{Rt}}{T_{rui}} + \frac{I_{Dt}T_{Dt}}{T_D} + O_i \left(\frac{I_{Rr}T_{Rr}}{T_{rui}} + \frac{I_{Dr}T_{Dr}}{T_D} \right) \right. \\ + & \left. F_i \frac{I_{Dt}T_{Dt}}{T_D} + \frac{I_s T_s}{T_D} + 8 I_P T_P \right) \leq B_i \end{aligned} \tag{9}$$

Critical node constraint: If we assume that there are some nodes named *critical nodes* that can support a maximum overhearing traffic and C_r is the set of these nodes then

$$O_i \leq O_i^M \ \forall i \in C_r \tag{10}$$

where O_i^M is the maximum number of allowed overhearing rate for i.

First-level node channel constraint: The first-level nodes choose any of the K available channels and Z_k^l is a binary variable that is 1 if first-level node k chooses channel l and 0 otherwise, than

$$\sum_{l=1}^{K} Z_k^l = 1 \ \forall k \in G \tag{11}$$

where G is the set of first-level nodes. Thus the channels chosen by the first-level nodes are given by

$$C_k = \sum_{l=1}^{K} l Z_k^l \ \forall k \in G \tag{12}$$

From constraints (2)-(12), we can observe that the problem is nonlinear. Next we calculate the complexity of this problem.

C. Complexity of Maximum-Lifetime Multi-Channel Routing Problem

We show that the maximum-lifetime routing problem is NP-complete using reduction from the *Degree constrained spanning tree problem* even if for single channel. Degree constrained spanning tree is a spanning tree where the maximum vertex degree is limited to a certain constant k. One instance of our problem is when all the nodes overhear each other. In that case, from equation (9) the worst case lifetime L is given by:

$$L = min \frac{B_i}{F_i + c} \forall i \tag{13}$$

as all the other terms are constants. In equation (13), c is a constant. If D_i is the degree of node i, then $D_i = F_i + 1$, thus $L = min \frac{B_i}{D_i + \mathcal{C}} \forall i$ for some constant $\mathcal{C}$.

Proof: First, it is clear that the maximum-lifetime multi-channel routing belongs to NP, since given a tree, we can calculate the worst case lifetime of the network in polynomial time.

To show that the problem is NP-hard, we show that for a graph G has a spanning tree of maximum vertex degree of k if and only if G has a tree whose lifetime is greater than or equal to $\frac{1}{k+\mathcal{C}}$. We set $B_i = 1$, $\forall i \in G$.

Suppose G has a spanning tree T with a maximum vertex degree of k. Then it is straightforward that the lifetime of T is

$$L(T) = min \frac{B_i}{D_i(T) + \mathcal{C}} \geq \frac{1}{k + \mathcal{C}} \tag{14}$$

Similarly, if G has a spanning tree T with $L(T) \geq \frac{1}{k+\mathcal{C}}$, then we have $D_i(T) \leq k$, $i = 1...N$. Otherwise, if $D_j(T) > k + 1$ for some $j \in [1, N]$, then

$$L(T) \leq \frac{B_i}{D_j(T) + \mathcal{C}} \leq \frac{1}{k + 1 + \mathcal{C}} \tag{15}$$

which is contradictory.

Thus, we can reduce an instance of the degree constrained spanning tree problem to an instance of our maximum-lifetime routing problem. As the degree constrained spanning tree problem is NP-complete, the maximum-lifetime routing problem is NP-hard even for single channel.

As the single channel routing is a special case of multi-channel routing, thus the maximum-lifetime multi-channel routing is also NP-complete. In the light of NP-completeness, we propose some heuristics to solve this problem. We develop three distributed schemes as well as one centralized scheme CRCS for route and channel selection which are explained in the following sections.

V. DISTRIBUTED ROUTE AND CHANNEL ASSIGNMENT SCHEMES FOR SENSOR NETWORKS

We now present the proposed distributed route and channel assignments schemes named DRCS-1, DRCS-2 and DRCS-3 as described below:

A. DRCS-1

We define the nodes that are immediate neighbors of sink as first-level nodes. Nodes that are neighbors of first-level nodes are termed as second-level nodes and so on. For all the distributed schemes, we assume that all nodes know the battery life of their neighbors, i.e. if there is any change in battery life, nodes broadcast *update* messages. Our DRCS-1 scheme can be explained by the following set of actions.

Battery broadcast phase: At first all the nodes are on the same channel (say C). The sink first broadcasts the route request to all first-level nodes through C. All the first-level nodes go on random backoff based on their battery life and then choose the least used channel (out of K orthogonal channel) around their neighbors and broadcast L_{P_i} (L_{P_i} is the estimated battery lifetime of P_i) and chosen channel through C. We call these packets *battery broadcast (BB)* packets. BB packets have a field named *full* that is 0 if a node still can afford children, otherwise the full bit is set to 1. All first-level nodes choose sink as their parent.

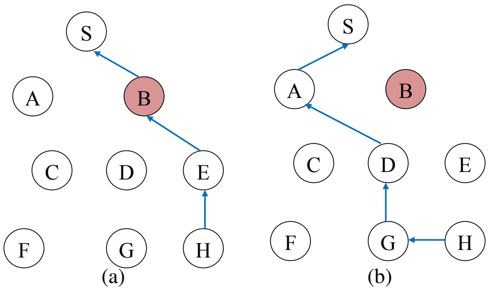
Parent broadcast phase: All the second-level nodes, upon receiving the BB packet, check their own battery power and based on the battery status, they wait for a random backoff that is proportional to their battery life. This is espected to give preference to the nodes to select channel that have lower power. In the backoff period, all nodes overhear the channel and calculate the usage of each channel in their neighborhood. When the backoff timer expires, each second-level node chooses it's parent as follows. For any channel c, each node calculates $\mathcal{L}_c = \min\{L_i\}\ \forall\ i \in S_c$ where S_c is the set of neighbors that are in channel c. Then a node chooses the channel j such that $\mathcal{L}_j = \max\{\mathcal{L}_c\}\ \forall\ c$. After choosing the channel j, a node chooses a parent P_i with maximum $L_{P_i}, \forall P_i \in \mathcal{P}_j$ where $\mathcal{P}_j$ is the set of parents of that node with channel j. This avoids making a less powered node their parent. Also the channels used by the less powered neighboring nodes are avoided. After choosing their parent, nodes broadcast *parent broadcast (PB)* packets that consist of the parent ID.

Parent confirmation phase: After receiving the parent broadcast packet from a child, the parent confirms by sending *parent confirmation packet*. The parent P_i calculates a new L_{P_i} and sends this in the parent confirmation packet. If some nodes have a strict constraint on maximum overhearing traffic (say a maximum of $\mathcal{N}$ packets/second) and it has n nodes that are overheard by it, then it informs all it's neighbors not to send more than $\frac{\mathcal{N}}{n}$ packets/sec in the parent confirmation packet. All its neighbors in the next parent confirmation phase do the same to their children. This process goes on until and unless one node is reached that cannot afford more children. Thus this node broadcasts with a BB packet with full bit set to 1, implying that it cannot take any more children. All the children avoid using that node as their parent if they have other options. If they do not have any other parent, then they connect to that node. If may happen that a node can afford few children (say 2) in their parent broadcast phase. Thus they broadcast BB packets with full = 0, but after getting 2 children, they immediately broadcast a BB packet with full = 1.

This process goes on until the last-level nodes are reached. The last-level nodes choose their parent, send the BB packets and after sometime all nodes switch to their chosen channels and start sending packets to their parents.

Overhead analysis: Let us assume that there are $\mathbb{L}$ labels and number of nodes in level i is l_i, $\forall\ i \in (1, \mathbb{L})$. At first the sink sends a route request packet to all the first level nodes. This is followed by l_1 BB packets from the first level nodes, followed by l_2 parent broadcast packets from the second level nodes, followed by l_2 parent confirmation packets from the first level nodes. Thus the total overhead for the parent discovery of the second level nodes is given by $l_1 + 2l_2$. This process goes on until the last level nodes, where each of the $l_{\mathbb{L}}$ nodes broadcasts one BB packets. Thus the total overhead of DRCS-2 is given by $1 + \sum_{i=1}^{\mathbb{L}-1}(l_i + 2l_{i+1}) + l_{\mathbb{L}}$. This calculation ignores the case when the maximum overhearing constraint of any node is violated, in this case some extra overhead should be taken into account. Also the overheads of the update messages are not considered in overhead calculation.

B. DRCS-2



(a) (b)

Fig. 3. Route chosen by (a) DRCS-1, (b) A better route.

The advantage of DRCS-1 lies in it's simplicity and low overhead. But the problem with DRCS-1 is that a node at level i always chooses parent from level $i-1$, where there may be a better route via other nodes of level i or level $i+1$. As an example as shown in Fig. 3, if node B is very low powered, route H→D→G→A→S may be a better choice rather than H→E→B→S. Keeping this in mind, we present another distributed scheme DRCS-2. The idea of DRCS-2 stems from the idea of how water flows from a point to another, avoiding the high altitude areas. The notion of this scheme is that there is some altitude associated with the critical nodes and the nodes that are overheard by the critical nodes. Thus all nodes try to avoid these set of nodes to reach the sink. With this, the scheme of DRCS-2 can be described as follows:

At first the sink sends a broadcast packet with hop-count = 0. Any node that receives the packet increments the hop-count and rebroadcasts it. In this way, all nodes are able to get the hop-count from the sink. Then each node goes on random backoff that is proportional to it's battery life. When this backoff timer expires, it starts discovering the routes. Each node i calculates a metric named priority as $p_i = c_3\left(c_1 b_i + \frac{c_2}{l_i}\right)$ and notifies it's neighbors, where c_1, c_2 and c_3 are constants. In the expression of priority, b_i is the battery life of node i and l_i is the hop-count from node i to the sink. The variable l_i ensures that the routes should not be too long. $c_3 < 1$ for all critical nodes and their neighbors and equal to 1 for all other nodes. This makes sure that the critical nodes and their neighbors (nodes that are overheard by the critical nodes) gets less priority (more altitude) than others in relaying others traffics.

At first, all the first level nodes choose any channel silimar to DRCS-1 and then all the nodes go on random backoff based on their battery status. This channel selection of the first level nodes is same for all the schemes. When a node's turn comes, it chooses a node among it's neighbors with the highest priority and sends a *parent notify (PN)* packet to that neighbor that consists of the neighbor ID. The neighbor does the same and this process goes on until the PN packet reaches the second level nodes. For any channel c, each second level node calculates $\mathcal{L}_c = \min\{L_i\} \; \forall \; i \in S_c$ where S_c is the set of neighbors that are in channel c. Then the node chooses the channel with $\max\{\mathcal{L}_c\} \; \forall \; c$ and then chooses a parent P_i with maximum $L_{P_i}, \forall P_i \in \mathcal{P}$ where $\mathcal{P}$ is the set of parents of that node with the same channel. The PN packet carries the IDs of the nodes that it visits. To ensure that the PN packet does not circulate in a loop, an intermediate node upon receiving the PN packet, chooses a parent that is not visited by the packet. This process goes on for all the nodes until and unless all the nodes get a route to the sink. When the PN packets traverse in the network, nodes that can overhear the packet, update their battery life with the new information.

Overhead analysis: At first the sink sends a broadcast packet for determining the hop-count of all nodes from the sink. For a n node network, this requires an overhead of n. After that all the nodes except the first level nodes select their parents and send PN packets, which incurs a total overhead of $n - l_1$ (assuming that there are l_1 nodes in the first level) packets. Thus, totally DRCS-2 needs $2n - l_1$ packets as overhead of route and channel selection. We assume that all nodes broadcast their recent priority metric by sending the update messasges, the overhead due to these updates are not taken in overhead calculation.

C. DRCS-3

In the above two schemes, each node chooses its route based on the informations from it's neighbors. The information from all the intermediate nodes in the route is not used in these two cases. DRCS-3 is a scheme that exploits the information from the intermediate nodes of a route at the cost of more overhead. This scheme is described using the following stages:

Route Discovery: At first all the nodes notify the sink about their battery condition. The critical nodes also notify the sink of their neighbor's ID so that the sink knows the critical nodes and their neighbors. The sink first sorts the nodes according to their battery life and sends route discovery packets in increasing order of battery life. In the discovery packet, the sink includes the IDs of the critical nodes as well as the nodes that overhears the critical nodes. When the discovery packet travels through the network, it carries the sequence of node IDs that it traverses. Any intermediate node i calculates $a_i = \min(b_i, b_j), \forall j$ in its neighborhood, where b_i is the battery life of node i. Discovery packet also has a field that carries $\mathcal{T} = \min_{i \in \text{route}} a_i$.

Route Reply: The destined node waits for the first N packets and stores the routes in it's cache as well as their corresponding $\mathcal{T}$ values. Let us define $\mathcal{T}_i$, $i \in (1, N)$ as the minimum battery life of the i-th discovery packet. Then it chooses the route with highest $\mathcal{T}_i$, $i \in (1, N)$ and sends reply through that route.

Route Accept: After getting the reply packet, the sink checks whether this route fulfills the overhearing constraint or not. If the overhearing constraint is fulfilled then the sink sends an acknowledgement message with the accept bit set to 1, otherwise it sends accept message with accept bit set to 0. All the intermediate nodes update their route cache if the accept bit is 1. All nodes that overhears this packet get informed about the number of active nodes and their amount of traffic and recalculate their battery life. If the accept bit is 0, the destined node again sends reply packet through the next best route. Note that when the accept bit is 1, all the intermediate nodes are termed as explored nodes as they can get their path towards the sink as well. Next the sink sends the discovery packets form the list of unexplored nodes based on their battery life and this process goes on until all the nodes are explored.

It should be noted that this process incurs a large overhead. Thus in our scheme we consider that the sink sends route discovery for K destinations at a time. When K is small, the route and channel selection is very good but the route overhead is high and for large K, the route and channel selection is poor where the route overhead is low. Next, we derive the average number of route discovery phases that the sink has to go through before exploring the whole tree. Number of overheads is also calculated analytically.

Overhead analysis: We assume that there are $\mathbb{L}$ labels and number of nodes in level i is l_i, $\forall \; i \in (1, \mathbb{L})$. At first all the nodes are unexplored. Let us denote $P^j_{l_i}$, V^j, $V^j_{l_i}$, NV^j and $NV^j_{l_i}$ are the probability of choosing any unexplored node in level i at phase j, the number of nodes explored at phase j, the number of i-th level nodes explored at phase j, the number of unexplored nodes of level i at phase j respectively. Now, at first all the nodes are unexplored, i.e. $V^0 = 0$, $V^0_{l_i} = 0$, $\forall i \in (1, \mathbb{L})$, $NV^0 = n$ and $NV^0_{l_i} = l_i$, $\forall i \in (1, \mathbb{L})$, thus, $P^0_{l_i} = \frac{l_i}{\sum_{i=1}^{\mathbb{L}} l_i} = \frac{l_i}{n}$, $\forall i \in (1, \mathbb{L})$.

At the first phase a random node is chosen from the list of unexplored nodes. For simplicity, let us assume that all the nodes choose any of their previous level nodes to reach the sink. If any unexplored node of the i-th level is chosen, then the number of nodes explored at the first phase is i (at each level 1 node is explored). Thus, the number of explored nodes in first phase is $V^1 = \sum_{i=1}^{\mathbb{L}} i \times P^0_{l_i} = \sum_{i=1}^{\mathbb{L}} \frac{i \times l_i}{n}$ and $V^1_{l_i} = \frac{\sum_{j=i}^{\mathbb{L}} l_j}{n}$.

In general, at any phase k, if any unexplored node at i is chosen, then that node is explored with probability of 1. But for any previous level j ($j < i$), an unexplored node is explored with a probability of $\frac{NV^{k-1}_{l_j}}{l_j}$. Thus the number of nodes explored at phase k is $V^k = \sum_{i=1}^{\mathbb{L}} P_{l_i} \times \left(1 + \sum_{j=i-1}^{0} \frac{NV^{k-1}_{l_j}}{l_j}\right)$. The number of unexplored vertices at k-th phase is the difference between the number of unexplored nodes at $(k-1)$-th phase and the number of vertices explored at k-th phase, i.e. $NV^k = NV^{k-1} - V^{k-1}$. For the same reason, at each level $i \in (1, \mathbb{L})$,

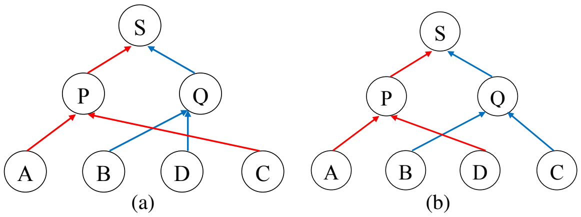
$NV_{l_i}^k = NV_{l_i}^{k-1} - V_{l_i}^{k-1}$. This process goes on until the number of explored nodes is less than n. As an example, if $l_1 = 20$, $l_2 = 30$, $l_3 = 40$, $l_4 = 50$ and $l_5 = 60$, we get the number of phases required is 114. Thus, if the sink sends $K = 10$ discovery packets at a time, then the number of times the sink has to send discovery packets is 12.

At first all nodes need to send their battery state to the sink, we assume that it takes an overhead of a broadcast i.e. n packets. Next in the i-th phase, it needs a broadcast and one route reply and route accept packet. If all nodes are explored in $\mathcal{K}$ phases, the total overhead is given by $(\mathcal{K}+1)n+2\sum_{i=1}^{\mathcal{K}} V^i$.

VI. CENTRALIZED ROUTE AND CHANNEL ASSIGNMENT FOR SENSOR NETWORKS (CRCS)

In the distributed approach, nodes do not have the picture of the whole network, thus this distributed solution can be further made better if this solution is passed from to the sink or base station where it can refine the route and channel selection and send this information to the nodes. For this, the sink needs the neighboring informations of all the nodes as well as their battery states.

We use a *simulated annealing* based approach to solve this problem. Let us assume that there are n nodes $\{v_1, v_2, ..., v_n\}$ and $\mathcal{S}_{v_i}$ is the set of neighbors of v_i. We use simulated annealing so that the solution does not get stuck into the local optima. As an example in Fig. 4, let us assume that *(A, B)*, *(B, D)* and *(D, C)* can overhear each other. If nodes *A, B, C, D* choose their parents sequentially, and *A, C* choose *P* as their parent and *B* chooses *Q*, then *D* would choose *Q* as parent. Then the system is in a local optimum. In this solution, *B* and *D* will overhear. It should be observed that a better solution is to assign *A* and *D* to *P* and *B* and *C* to *Q*, which is also another optimum and gives better performance than the previous one. Next, we introduce a centralized route and channel selection scheme that comes out of this local optimum with some probability.



(a)
(b)

Fig. 4. An example of (a) local optimal solution, (b) global optimal solution.

Our centralizaed route and channel selection (CRCS) scheme is shown in Fig. 5. It takes the solution given by *DRCS* (or a random initial solution) and then tries to make it better iteratively. Each of the n nodes has a set of neighbors. In each iteration the leaf level nodes first choose their parents one by one and then the upper level nodes and so on. When a node's turn comes, it runs simulated annealing as shown in Fig. 5. In simulated annealing there is a control parameter T that starts with a high temperature and then gradually reduces to a low temperature. For each T a node chooses a different parent and checks whether it reduces the cost or not. Here cost is defined as $\frac{1}{min(L_1,L_2,...,L_V)}$, thus our objective is to minimize the cost. If there is an improvement, the solution is accepted, otherwise the solution is accepted with a probability equal to $e^{-\frac{\Delta}{T}}$, where Δ is the difference between the previous cost and the new cost. This probailistic acceptance avoids stucking into local optimum. Because of this probability, there is a possibility that *C* chooses *Q* and then *D* choses *P*, i.e. the optimal solution. Also if the new solution does not satisfy equation (10), it is rejected. This process is iterated until a maximum number of iteration is reached or the nodes do not change parents for a predefined number of iterations.

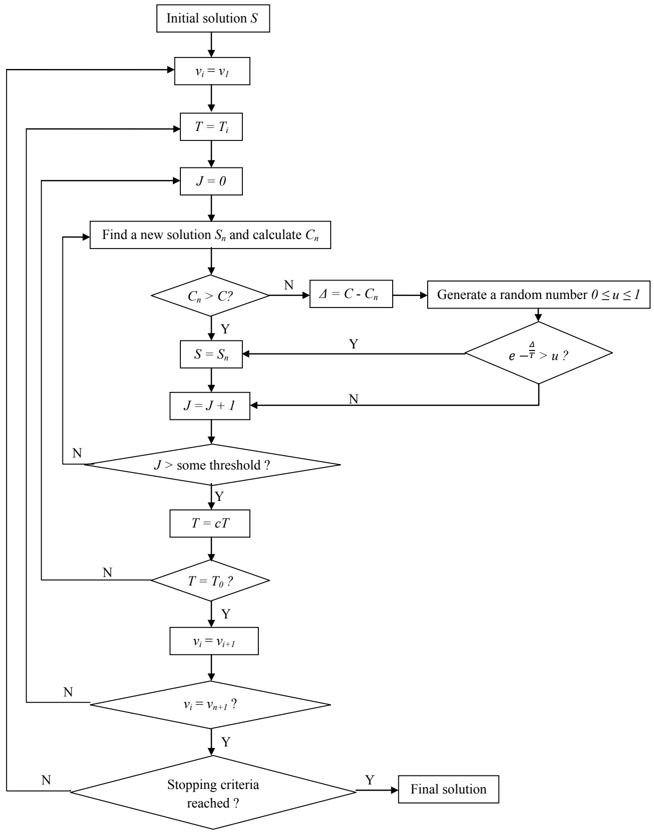
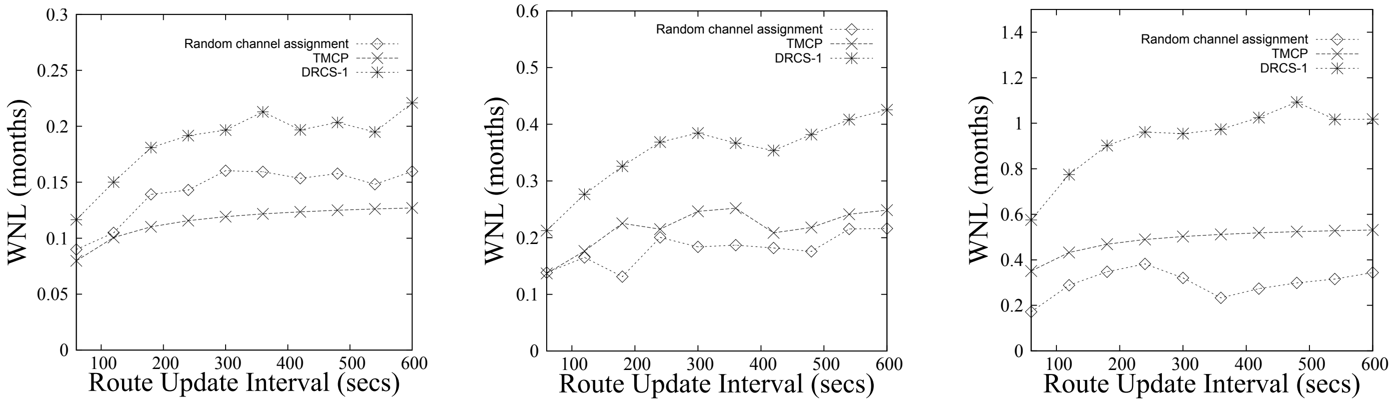


Fig. 5. Our route and channel selection scheme using simulated annealing

VII. PERFORMANCE EVALUATION

In this section, we present the performance of our proposed schemes as obtained from simulations. We consider one sink and 50 nodes in a grid topology of 450 meters × 500 meters as shown in Fig 6. The transmission as well as overhearing distance is 160 meters and interference range is 250 meters. The data interval (T_D) is assumed to be 60 seconds. The parameters used in the simulations are listed in Table I.

Compariosn of lifetime: We choose two critical nodes with an initial battery current of 500 mAhr and all other nodes have an initial battery current uniformly distributed between 1000-5000 mAhr. We vary the route update intervals and plot



(a) (b) (c)

Fig. 7. Comparison of lifetime when initial battery capacities are uniformly distributed for (a) 1 channel (b) 2 channels (c) 8 channels

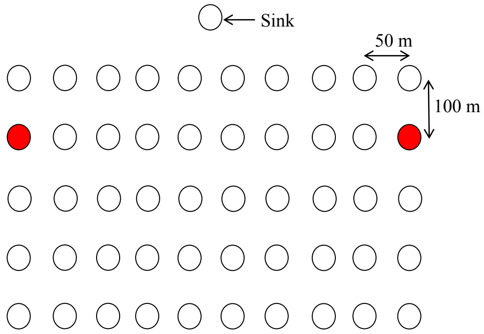


Fig. 6. The simulation environment, the red nodes are two critical nodes

TABLE I

SIMULATION ENVIRONMENT

Var	Values	Var	Values	Var	Values	Var	Values
I_{Rt}	20 mA	T_{Rt}	140 ms	I_{Rr}	20 mA	T_{Rr}	140 ms
I_{Dt}	20 mA	T_{Dt}	140 ms	I_{Dr}	20 mA	T_{Dr}	140 ms
I_P	8 mA	T_P	3 ms	I_S	7.5 mA	T_S	112 ms

the variations of worst-case battery lifetime of the networks for all the proposed schemes in Fig 7. Besides that, we also compare our schemes with the random channel and route selection scheme and TMCP [15]. From Fig 7 we can observe that all our proposed schemes outperform the random channel and route selection scheme as well as TMCP. Among the proposed distributed approaches, DRCS-2 performs very similar to DRCS-1. Also we can observe that DRCS-3 (with K = 1) performs better than the other two. The drawback of DRCS-3 is that it incurs more overhead in terms of exchanging route discovery, reply and accept packets. While comparing DRCS-3 and CRCS we can observe that CRCS gives higher lifetime as the sink acts as a central agent to choose the routes with the global information of the networks.

Comparison with number of channels: Fig 8 shows the comparison of lifetime with the variation of number of channels for different schemes when the route update interval is 600 seconds. Similar to Fig 7, we can observe that DRCS-3 performs better than DRCS-1 and DRCS-2 over different number of channels and CRCS performs the best. Also we can observe that after 6 channels, the performance start getting saturated.

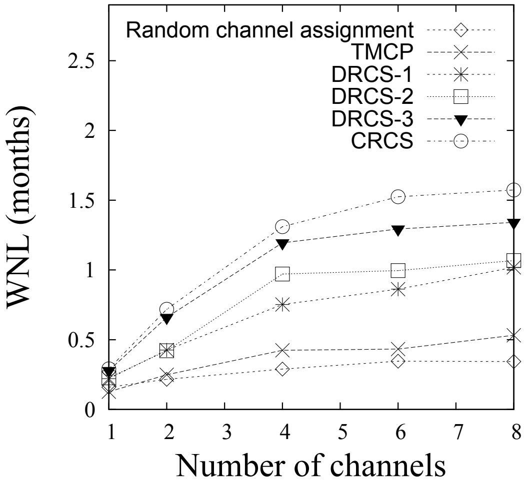


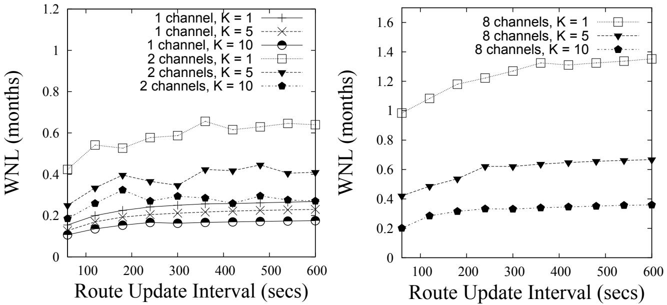
Fig. 8. Comparison of lifetime when different number of channels

TABLE II

COMPARISON OF OVERHEAD

Initial Distribution of Battery	DRCS-1	DRCS-2	DRCS-3 (K=1)	DRCS-3 (K=5)	DRCS-3 (K=10)
Uniform	131	91	1416	406	289

Comparison of overhead: Table II shows the comparison of the routing overhead for different distributed schemes. These overheads are only the control messages that are to be exchanged throughout the network only at the time of routes and channel assignment. Thus the periodic route updates and data exchanges are not considered in these overhead calculations. From Table II, we can observe that DRCS-3 has a much higher overhead compared to DRCS-1 and DRCS-2, but the performance of DRCS-3 is better compared to the other two. Thus DRCS-3 achieves better performance at the cost of high overheads.



(a) (b)

Fig. 9. Comparison of lifetime when initial battery capacities are uniformly distributed for (a) DRCS-3, 1 and 2 channels (b) 8 channels

Comparison of DRCS-3 for different K: Fig 9 shows the comparison of DRCS-3 with different values of K. From this figure, we can observe that the lifetime decreases with increase in K. As K increases, the route and channel updates are less frequent, which results in poor channel and route selection.

VIII. CONCLUSIONS

In this paper, we demonstrate the construction of data gathering tree in multi-channel wireless sensor networks. The problem turns out to be an NP-complete problem, whcih motivates the investigation of some distributed and centralized approximation schemes to solve this problem. Through simulations, we demonstrate the effectiveness of our proposed channel assignment and routing schemes compared to random channel and route selection and TMCP [15].

REFERENCES

[1] O. Chipara, Z. He, G. Xing, Q. Chen, X. Wang, C. Lu, J. A. Stankovic, and T. F. Abdelzaher, "Real-time power-aware routing in sensor networks," in *IWQoS*, 2006, pp. 83–92.

[2] L. Lin, N. B. Shroff, and R. Srikant, "Energy-aware routing in sensor networks: A large system approach," *Ad Hoc Networks*, vol. 5, no. 6, pp. 818–831, 2007.

[3] M. K. Marina and S. R. Das, "A topology control approach for utilizing multiple channels in multi-radio wireless mesh networks," in *BROADNETS*, 2005, pp. 412–421.

[4] A. P. Subramanian, H. Gupta, S. R. Das, and J. Cao, "Minimum interference channel assignment in multiradio wireless mesh networks," *IEEE Transactions on Mobile Computing*, vol. 7, no. 12, pp. 1459–1473, 2008.

[5] A. Raniwala, K. Gopalan, and T. cker Chiueh, "Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks," *ACM Mobile Computing and Communications Review*, vol. 8, pp. 50–65, 2004.

[6] A. Raniwala and T. cker Chiueh, "Architecture and algorithms for an ieee 802.11-based multi-channel wireless mesh network," in *INFOCOM*, 2005, pp. 2223–2234.

[7] P. Kyasanur and N. H. Vaidya, "Routing and link-layer protocols for multi-channel multi-interface ad hoc wireless networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 10, no. 1, pp. 31–43, 2006.

[8] A. Pal and A. Nasipuri, "JRCA: A joint routing and channel assignment scheme for wireless mesh networks," in *IEEE IPCCC*, 2011.

[9] M. D. Jovanovic and G. L. Djordjevic, "TFMAC: Multi-channel mac protocol for wireless sensor networks," in *TELSIKS*, 2007.

[10] "Xmesh user manual, Crossbow technology," `http://www.xbow.com/Products`, 2007.

[11] G. Zhou, C. Huang, T. Yan, T. He, J. A. Stankovic, and T. F. Abdelzaher, "MMSN: Multi-frequency media access control for wireless sensor networks," in *INFOCOM*, 2006.

[12] J. Zhang, G. Zhou, C. Huang, S. H. Son, and J. A. Stankovic, "TMMAC: An energy efficient multi-channel mac protocol for ad hoc networks," in *ICC*, 2007, pp. 3554–3561.

[13] X. Chen, P. Han, Q.-S. He, S. liang Tu, and Z. long Chen, "A multi-channel mac protocol for wireless sensor networks," in *CIT*, 2006.

[14] Y. Kim, H. Shin, and H. Cha, "Y-MAC: An energy-efficient multi-channel mac protocol for dense wireless sensor networks," in *IPSN*, 2008, pp. 53–63.

[15] Y. Wu, J. A. Stankovic, T. He, and S. Lin, "Realistic and efficient multi-channel communications in wireless sensor networks," in *INFOCOM*, 2008, pp. 1193–1201.

[16] H. K. Le, D. Henriksson, and T. F. Abdelzaher, "A control theory approach to throughput optimization in multi-channel collection sensor networks," in *IPSN*, 2007, pp. 31–40.

[17] Q. Yu, J. Chen, Y. Fan, X. Shen, and Y. Sun, "Multi-channel assignment in wireless sensor networks: A game theoretic approach," in *INFOCOM*, 2010, pp. 1127–1135.

[18] A. Nasipuri, R. Cox, J. Conrad, L. V. der Zel, B. Rodriguez, and R. McKosky, "Design considerations for a large-scale wireless sensor network for substation monitoring," in *LCN*, 2010, pp. 866–873.