Network Coding-aware Lifetime Maximal routing in Multi-hop Static Wireless Ad-hoc Networks

Rigil Salim, and Rajarshi Roy

Abstract – **In this paper, we address the issue of maximizing the lifetime of a static wireless ad-hoc network wherein the nodes are battery powered and have limited energy. In such scenarios, routing the traffic along shortest paths will lead to over-use of some nodes leading to premature network partition and an eventual end of communication. Network Coding is a promising technique that has been used, of late, by researchers for throughput improvement. We propose an algorithm that exploits network coding to route a set of unicast traffic demands, the objective being network lifetime maximization. The routing algorithm uses a link metric that takes care of the communication power consumption, the residual energy at the nodes and also the potential coding opportunities available at the node. Simulation results show that this algorithm enhances the network lifetime compared to the existing algorithms that do not employ network coding.**

Index Terms – **Network coding, Network lifetime, Network Coding Advantage, routing protocols, wireless multi-hop networks.**

I. INTRODUCTION

A wireless ad-hoc network is a decentralized network, wherein the nodes forward data directly based on the network connectivity, without using any routers or access points. A typical example of such a network is the Wireless Sensor Network (WSN). The nodes are static and are battery-powered, and hence the major limitation of such networks is that their energy resources are non-replenishable. In such scenarios, extending network lifetime becomes a matter of paramount importance. Network lifetime is the time to death of the first node in the network (i.e. node runs out of battery).Hence, developing lifetime-maximal routing protocols for wireless ad-hoc and sensor networks have been an important area of research, of late. Numerous routing techniques for maximizing the network lifetime can be found in literature.

Most works on routing multiple unicast flows in wireless ad-hoc networks deals with finding paths and modifying it according to the varying network conditions. Many proposed routing protocols employ routing based on hop count, called Minimum Hop (MH) routing. Many authors have proposed the Minimum Total energy (MTE) routing scheme [19], [21], considering the limited energy resources available. The MTE routing scheme finds a path between the source and the destination that consumes the minimum amount of energy. But the major problem with MTE is that in course of finding the path that consumes the least energy, some nodes might get overused and exhaust their batteries soon. This may lead to network partition and reduced lifetime is the consequence.

To alleviate the above problem, many routing algorithms have been proposed. The max-min residual energy (MMRE) routing [2] selects a path whose minimum energy fraction will be the maximum after the packet has been routed. This ensures that paths having node with lower energy are discarded. Toh [22] proposed the Conditional Max-min battery capacity routing (CMMBCR) that combines the properties of both MTE and MMRE routing. For a given source-destination pair, it creates a set of paths whose residual energy fraction is above a threshold and chooses the minimum energy path from that. If no such path exists, MMRE routing is utilized. In [15], an approximation algorithm called max – min zP_{min} was proposed to maximize lifetime, which aims to strike a balance between MTE and MMRE routing, but here the incoming traffic sequence is not known *apriori*. Two routing methods were proposed in [12] by Kim et al.,which were named the Minimum Drain Rate (MDR) and the Conditional Minimum Drain Rate (CMDR) protocols, using drain rate as a metric. A realistic model of battery was considered in [18]. The batteries are found to show *Rate Capacity Effect* wherein, capacity of a battery decreases with increase in discharge current. There have been proposals to minimize these effects with proper traffic shaping, pulse shaping and burst shaping at physical layer [4]. The authors in [18] present a routing protocol which minimizes rate capacity effect and exploit the charge recovery effect of a practical battery to give maximum lifetime of the route discovered.

In [3], the authors formulated the problem of routing a set of traffic demands as a linear programming problem, with the objective of maximizing lifetime. They also proposed a routing algorithm called Flow Augmentation (FA) algorithm, where the link costs reflect both the communication energy required as well as the residual energy at the transmitter and receiver. They showed that their proposed algorithm could achieve lifetime values close to the optimal value. Kang et al. [10] addressed the problem of enhancing lifetime for a single broadcast session. They formulated the problem as a min-max optimization problem and solved it using graph-theoretic

---

Manuscript received January 11, 2014; revised March 19, 2014.

Rigil Salim is with Indian Institute of Technology, Kharagpur, India (e-mail: rigilsalim@yahoo.com).

Rajarshi Roy is with Department of Electronics and Electrical Communication, Indian Institute of Technology, Kharagpur, Kharagpur-721302, West Bengal, India (e-mails: rigilsalim@yahoo.com, royr@ece.iitkgp.ernet.in).

approach. They proposed the Directed Minimum Spanning tree (DMST) algorithm, a globally optimal broadcast routing tree solution. An extensive survey on power optimization in routing protocols has been presented in [17].

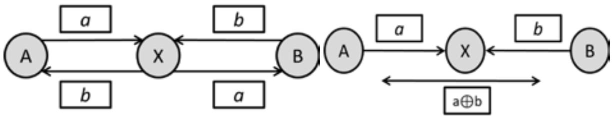
Network Coding has been found to be a potential candidate for network lifetime improvement. Network coding uses the concept of mixing (or coding) packets at the intermediate nodes, instead of just storing and forwarding the packets. Thus, the total number of transmissions required can be reduced. Although network coding was originally introduced as a technique for throughput improvement, it can also be used for lifetime enhancement. This is because, since the number of transmissions gets reduced owing to Network Coding, the energy that could have been expended for those transmissions is saved. This energy can be used to send more packets, hence an improved lifetime.

The basic concept of Network Coding can be explained using the Alice-Bob scenario depicted in Fig.1. Here we have 3 nodes A, X, B sharing a wireless medium and with the constraint that only one node can transmit at a time. Suppose A and B want to exchange information with each other. If coding is not used, then A sends packet *a* to X, then B sends packet *b* to X. In the third transmission, X sends packet *b* to A, and finally X sends packet *a* to B, thus completion the information exchange. Hence four transmissions are required. On the other hand if network coding is used, then the first two transmissions remain the same. Now instead of forwarding the packets individually, node X will XOR the packets to form a single packet $p = a \oplus b$, and broadcasts it over the wireless medium. Since A and B knows what they had sent, they can retrieve the required information. A performs $a \oplus (a \oplus b)$ to get packet *b* and B performs $b \oplus (a \oplus b)$ to get packet *a*. Thus one transmission is saved. Hence the throughput improvement is 33%. Looking from another perspective, 25% of energy has also been saved owing to the saved transmission.

A. Prior Work

The concept of network coding was introduced by R. Ahlswede et al in 2000 [1]. It was introduced as a technique for throughput improvement. Till then, many works have focused on improving network throughput using Network coding. The next substantial work was by Li et. al [14] which showed that linear network coding is sufficient to achieve multicast capacity. Koetter et al. [13] investigated the necessary and sufficient conditions for the feasibility of a given set of connections over a given network that uses linear network coding. They also showed that even if we restrict ourselves to linear codes, determining how to perform inter-session coding is NP-hard. Hence, the focus has been on developing heuristics to reap throughput gains.

It was shown by Li et al. [16] that in wireless multi-hop networks with multiple unicast sessions, network coding can give benefits over traditional approaches. Network coding for multiple unicasts was addressed by Ho et al. in [7] with the restriction that XOR based network coding can be performed only between pairs of flows. Some formidable work towards practical implementation of Network Coding was presented by Katti et.al in 2008 [11].The paper presents COPE architecture for wireless mesh networks. It was shown that significant throughput improvement could be obtained by using network coding. COPE architecture inserted network coding into the protocol stack between network and MAC layer. The problem of minimum energy broadcasting in ad-hoc wireless networks is known to be NP-complete. But if we use network coding, the problem can be formulated as a linear program and thus accepts a polynomial-time solution. The authors in [24] present a linear program formulation for the problem of minimizing the energy per bit when multicasting in a wireless ad-hoc network. Apart from energy-minimization problems, many lifetime maximization problems that incorporate network coding can also be found in literature [8-9].



(a) without coding (b) with network coding

Fig.1. Basic concept of Network coding.

Sengupta et al. [20] analyzed a joint coding-aware and interference-aware routing protocol for multiple unicast sessions in multi-hop wireless networks, with the aim of maximizing network throughput. But the above works assumed that the nodes were not energy-limited, which is really not the case in sensor and ad-hoc networks. The work by Gaddam et al. [6] addressed this issue and network coding was used as an energy-minimization technique. They formulated the lifetime maximization problem as a linear programming (LP) problem, and showed that network coding could improve network lifetime. They also explored the trade-off between selecting paths where network coding is employed and network lifetime, and showed that aggressive coding can hamper lifetime.

B. Our Contribution

The above-said works on lifetime maximization using Network coding aimed at formulating it as a linear programming problem and solving for optimal values. But in actual scenario, protocols that can deliver the optimum values or values close to optimum are required. A recent work by Xia [25] tried to explore the idea of Joint Coding to find paths to route a set of unicast traffic demands in a wireless multi-hop network. They developed an analytical framework to maximize throughput and also proposed an iterative algorithm to determine routes, keeping the mutual interference between flows into consideration.

In this paper, given a set of multiple unicast sessions to be routed in a static wireless multi-hop network, our aim is to route the traffic flows such that network lifetime is maximized. For this purpose, we employ network coding as an energy-minimization technique. We aim at proposing an algorithm that can route these traffic demands and incorporate as many coding opportunities as possible that can improve lifetime. The effect of path reconfiguration, update interval and effect of allowing multiple coding groups in a node will be analyzed. The effect of aggressive coding on network performance will also be explored.

The rest of the paper is organized as follows. In section II, we describe the system model and explain the various terms and notations used in the paper. In section III, we describe our proposed algorithm and some enhancements to the same. In section IV we present our simulation results and observations. Finally in section V, we present our conclusions.

II. SYSTEM MODEL

In this section, we explain the meanings of various terms and notations that will be used throughout the paper.

A. Network Model

The network model used here is a static, wireless multi-hop network. The nodes are deployed in a k×k m^2 square area. Every node in the network has identical wireless communication capability and is equipped with an omni-directional antenna for communication. All the links in the wireless network are assumed to be lossless and have the same capacity *C*. Nodes run on batteries and the initial energy levels of node *i* is denoted by E_{i0} and residual energy level after *n* iterations is given by $E_i(n)$ respectively. Power consumption by nodes in wireless networks consists of three modes: transmission, reception and idle. The power consumption in the reception and idle modes has been assumed to be zero.

If d_{ij} is the Euclidean distance between nodes *i* and *j*, then the received power is known to vary as $d_{ij}^{-\propto}$, where α is the path loss factor .It usually varies between 2 and 4. So if the receiver sensitivity is *K (*say), then the power required at the receiver for proper detection of the signal is $d_{ij}^{\propto}$. For notational simplicity we assume *K* to be 1, so the transmit power $P_{ij} = d_{ij}^{\propto}$.

The network has been modeled as a weighted, directed graph G = *(N, A)* where *N* is the set of nodes in the network and *A* is the set of links. A link *(i, j)* is said to exist if $d_{ij} \le R_{max}$, where R_{max} is the maximum possible communication range of the node. Let S_i be the set of all nodes that are directly reachable from *i* with its transmit power level. This model is called unit disk graph model [5].

In the problem under consideration, we have a set of *F* unicast traffic demands to be routed across the network. Each traffic flow has a source node and a destination node. Each of the flows is assumed to have same data rate Q_f. Hence, a traffic flow *f* can be denoted by $\mathcal{F}_f(s_f, d_f)$. The power consumed by node *i* to send a unit of data to node *j* can be denoted by e_{ij}.

Let $q_{ij}^{(f)}$ be the transmission rate of flow *f* from node *i* to node *j*. Then the lifetime of node *i* under the given flow $\boldsymbol{q} = \left\{q_{ij}^{(f)}\right\}$ is given as defined in [3] :

$$T_i(\boldsymbol{q}) = \frac{E_{i0}}{\sum_{j\in S_i} e_{ij} \cdot \sum_{f\in\mathcal{F}} q_{ij}^{(f)}} \qquad (1)$$

The *network lifetime* under the given flow is the minimum lifetime among all the nodes. It can be defined as follows:

$$T_{sys}(\boldsymbol{q}) = \min_{i \in N} T_i(\boldsymbol{q}) \qquad (2)$$

The above two equations for lifetime hold when network coding is not used. The network lifetime formulation with network coding included, will be defined in the subsequent section. At this point it is important to define the notion of network lifetime used in the paper. It is defined as the time to death of first node in the network due to battery depletion. Looked at from another perspective, we can say that the total number of packets that have been delivered across the network is indicative of the network lifetime. If we assume the data rates of the flows to be 1 packet/time slot, then Eq. (2) reduces to the above notion of lifetime. The list of notations and definitions used in this paper is given in Table I.

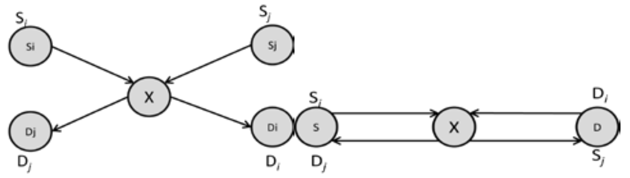
TABLE I
NOTATIONS AND DEFINITIONS

Notation	Meaning
(i,j)	a directed edge in the graph *G = (N, A).*
E_{i0}	initial energy level of node *i*
$E_i(n)$	residual energy level of node *i* after *n* iterations/updates.
α	path loss factor, ranges between 2 and 4.
W_{ij}	weight of an edge *(i,j)* in the graph.
d_{ij}	Euclidean distance between nodes *i* and node *j*
$T_{sys}(\boldsymbol{q})$	lifetime of the network under the given flow **q.**
R_{ij}	Maximum communication range of node *i* for flow *j*.
$\boldsymbol{\aleph_j(i)}$	Physical neighbor of node *i* for flow *j*. $\aleph_j(i) := \{ k \mid 0 < d_{ik} \le R_{ij} \}$ (3)
$\boldsymbol{\mathfrak{R}_j(i)}$	Logical neighbor (next- hop) of node *i* in flow *j*. $\mathfrak{R}_j(i) := \{ k \mid (i,k) \in F_j \}$ (4)
$\boldsymbol{\pi_j(i)}$	Predecessor of node *i* in flow *j*. $\pi_j(i) := \{k \mid (k,i) \in F_j \}$ (5)
λ	Update interval
$\boldsymbol{M_i}$	Set of coded-flow groups at node *i*.

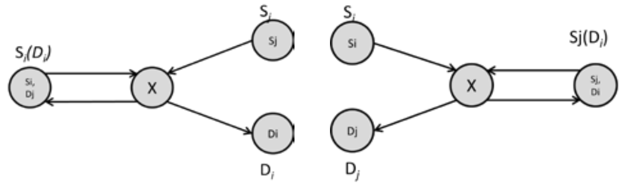
Fig. 2. Network Coding Opportunities

B. Network Coding Model and assumptions

The network coding model used here is a simple XOR-based coding and one-hop decoding. This means that the coded packets will be decoded at its next immediate hop. The network coding model relies on a very important property of wireless networks, namely, Wireless Broadcast Advantage (WBA), whereby all the physical neighbors of a node receive the packets sent by it, irrespective of the node being the intended receiver or not. If network coding is introduced, nodes have to store the packets it overhears for some period of



(a)X-structure (b) Alice-Bob structure



(c) Y-structure - I (d) Y-structure - II

time, even if the packets are not for them. The concept of next-hop decoding reduces the number of such extra packets that need to be stored.

Consider k packets p_1, p_2 ….. p_k having distinct next-hops $n_1, n_2, \ldots, n_k$,being coded at a node. XOR-based coding implies that XOR operation is performed on all the k packets on a bit-by-bit basis. Thus we get a coded packet $p = p_1 \oplus p_2 \ldots\ldots \oplus p_k$, of the same size as p_i. This coded packet will be broadcast over the wireless medium to all the next-hop nodes. The above coding is valid if all the next-hop nodes have the information about all packets except its own destined packet. In other words, node n_i has information of all packets p_j $(j \neq i)$. Node n_i could have received packet p_j in two ways:

- node n_i is the predecessor node for packet p_j, or
- node n_i overhears the transmission of packet p_j from the transmission of its predecessor. This is called opportunistic listening.

The first case requires that all three participating nodes be a part of both the flows. In other words, we have two flows in opposite directions. The second case is called opportunistic listening wherein a node overhears a packet not destined for it, for decoding purposes. The above discussion gives an insight to the existence of potential coding structures that can be found in a network. The basic coding structures are the X-structure, Y-structure and Alice-Bob structure. Alice-Bob structure does not use opportunistic listening, while the other three structures utilize the advantage of opportunistic listening to harness coding advantages. Fig.2 (a)-(d) shows these structures where there are two unicast flows $\mathcal{F}_1(S_1, D_1)$ and $\mathcal{F}_2(S_2, D_2)$.

For all the above network coding structures, the first and foremost condition to be satisfied is that the relay node X must be part of both the unicast flows being coded. Let i and j be the two flows that will be coded at node X. The path selected for flow i is given by $\mathcal{F}_i = \{n_{i1}, n_{i2}, n_{i3}, \ldots, n_{ik}\}$ and comprises of k nodes. Then for network coding to happen at X, we require,

$$X \in \mathcal{F}_i \qquad (6a)$$

$$X \in \mathcal{F}_j \qquad (6b)$$

The additional conditions to be satisfied for the coding structures are also given below, for flows $\mathcal{F}_i$ and $\mathcal{F}_j$, $j \neq i$.

X-structure:

$$\Re_i(X) \in \aleph_j\left(\pi_j(X)\right) \qquad (7a)$$

$$\Re_j(X) \in \aleph_i\left(\pi_i(X)\right) \qquad (7b)$$

Y-structure:

$$\Re_i(X) \in \aleph_j\left(\pi_j(X)\right) \qquad (8a)$$

$$\Re_j(X) = \pi_i(X) \qquad (8b)$$

Alice-Bob structure:

$$\Re_i(X) = \pi_j(X) \qquad (9a)$$

$$\Re_j(X) = \pi_i(X) \qquad (9b)$$

Fig. 2 shows coding of two flows at a node. We can also have more than two flows being simultaneously coded at a node. Let the set of flows being form a coding group. Assume that there are $\mathcal{M}_i$ such coded groups in node i. The coding groups can be denoted by $M_i = \left\{M_i^{(1)}, M_i^{(2)} \ldots\ldots M_i^{(\mathcal{M}_i)}\right\}$ .Each coded group k at node i, $M_i^{(k)}$ consists of the set of flows that are being coded at node i in group k. When network coding is incorporated, there will be only one transmission corresponding to a coded group. The transmit power level is chosen such that the packet reaches the next-hop of all the flows in the coded group.

Incorporating network coding into the routing algorithm necessitates a modification in the equation for network lifetime. Hence, the lifetime of node i under a given flow $\boldsymbol{q} = \left\{q_{ij}^{(f)}\right\}$ with network coding incorporated is given as:

$$T_i(\boldsymbol{q}) = \frac{E_{i0}}{\sum_{j \in S_i} e_{ij} \,.\, \sum_{f \in \mathcal{F}} q_{ij}^{(f)} - K_i} \qquad (10)$$

where,

$$K_i = \begin{cases} \sum_{m=1}^{\mathcal{M}_i} \left[\sum_{j \in S_i} e_{ij} \,.\, \sum_{f \in \mathcal{F}} I_i^{(m)} . q_{ij}^{(f)} - \max_{j \in S_i} \left(e_{ij} . \sum_{f \in \mathcal{F}} I_i^{(m)} \right) . C \right] & , \ \mathcal{M}_i > 0 \\ 0 & , \ \mathcal{M}_i = 0 \end{cases} \qquad (11)$$

$$I_i^{(m)} = I\{f \in M_i^{(m)}\} = \begin{cases} 1, & \textit{if flow f is an element of } m^{th} \textit{ coding group at node i} \\ 0, & \textit{otherwise} \end{cases} \qquad (12)$$

The network lifetime under the given flow will be the minimum among all the nodes and is given by Eqn. (2). As can be seen, K_i is either zero or positive, indicating that lifetime of all nodes have increased, consequently leading to an increased network lifetime.

III. Lifetime Maximal Network Coding-Aware Routing Algorithm

Network coding helps in reducing number of transmissions, and hence energy, by exploiting the mutual influence between flows. The problem addressed in this paper is to incorporate network coding into the routing algorithm such that the paths generated can exploit as many coding opportunities as possible and also ensure maximum lifetime.

It is known that using MTE algorithm tends to create hot-spots in the network, thus reducing lifetime. In [3], the authors proposed a metric that included the transmit power, the initial and residual energy levels at each node. The link metric proposed in [3] is given as:

$$W[i][j] = (P_{ij}^{t})^{x_1}.E_i^{-x_2}.E_{i0}^{x_3} + (P_{ij}^{r})^{x_1}.E_j^{-x_2}.E_{j0}^{x_3} \qquad (13)$$

where, P_{ij}^{t} and P_{ij}^{r} are transmit power and reception power respectively for communication from node i to node j. E_{i0} is the initial energy level and $E_i(n)$ is the residual energy level at the node after n iterations. x_1, x_2 and x_3 are non-negative weighting factors for each quantity. x_1 can be either 1 or 0. The reception power has been assumed to be zero in our work, thus leaving only a single term in the metric. To incorporate Network coding into this framework, a new term called "Network Coding Advantage" (NCA) has been introduced by us. It is defined as the energy expended by the node if it is sending a coded packet, to the energy expended in sending them, if they were un-coded. Mathematically it can be expressed as:

$$NCA_i = \frac{\max_{f \in M_i^{(m)}} E_f}{\sum_{f \in M_i^{(m)}} E_f} \qquad (14)$$

where, E_f is the energy expended in transmitting a data unit(packet) from node *i* to its next-hop in flow *f*.

The lifetime of a link *(i, j)*, also called residual link longevity [3], is the maximum number of packets that can be sent through the link before the node runs out of battery. The residual link longevity after n iterations $L_{ij}(n)$, is given by $L_{ij}(n) := E_i(n)/\, P_{ij}$. Now consider the Alice-Bob structure in Fig. 2(b). Let the central node be denoted by i and the other nodes be denoted by j and k. Without network coding, the link longevity is given by$L_{ij}(n) := E_i(n)/(P_{ij} + P_{ik})$. If network coding is used, the link longevity increases to $L_{ij}(n) := E_i(n)/max(P_{ij}, P_{ik})$. Thus we can see that the inclusion of network coding increases the link longevity by a factor $(P_{ij} + P_{ik})/\max(P_{ij}, P_{ik})$ .It can be easily observed that this factor is, in fact, the inverse of NCA_i . If we assume same initial energy for all nodes and all x_i values to be 1, then the metric given in Eqn.(13) is the inverse of residual ink longevity multiplied by the node's initial energy. Hence the modified link metric after inclusion of network coding can be expressed as follows:

$$W'[i][j] = (P_{ij}^{t})^{x_1}.E_i^{-x_2}.E_{i0}^{x_3}.NCA_j^{x_4} \qquad (15)$$

where, x_4 is also a non-negative weighting factor. If $x_4 \neq 0$, then it indicates usage of network coding, and 0 indicates no network coding. The meanings of these parameters are given in Table II. An important point to be observed is that even though NCA is calculated for the coding node, the index value used is that of the receiving node. This signifies the advantage we get, by taking this node as the successor of the current node in the flow being constructed. If a coding opportunity emerges at the current node on selecting node j as successor of node i in the current flow, then NCA_j is computed, else it is set to 1.

TABLE II
MEANINGS OF PARAMETERS USED IN CA_LMR ALGORITHM

CA_LMR (x_1, x_2, x_3, x_4)	Meanings
CA_LMR(0, 0, 0, 0)	Hop-based Routing(MH)
CA_LMR(1, 0, 0, 0)	Minimum Total-Energy routing(MTE)
CA_LMR (-, x, 0, 0)	Metric incorporates absolute residual energy
CA_LMR (-, x, x, 0)	Metric incorporates normalized residual energy.
CA_LMR (-, x, 0, 1)	Coding-aware routing using absolute residual energy metric
CA_LMR (-, x, x, 1)	Coding-aware routing using normalized residual energy metric

A. The Algorithm

In this section, we present a heuristic called Network-Coding aware Lifetime-Maximal routing algorithm (CA_LMR). In this algorithm, a modified version of Dijkstra's algorithm has been used that takes care of finding paths with coding opportunities. In every iteration, the algorithm finds the shortest path for flow $f \in \mathcal{F}$. Packets are sent through this path until the next routing information update. The update interval is denoted as λ, which means that λ packets from every flow will be routed before the next update. At every update, the residual node energy is calculated, the link costs are recomputed and the shortest path algorithm is run again. The algorithm stops when any of the nodes run out of energy. The total number of packets routed till then is indicative of network lifetime.

The CA_LMR algorithm uses the link metric as given in Eqn. (15). When the algorithm selects the successor node of the current node in the current flow, it checks *whether selecting this node as the successor of the current node in the current flow can create coding opportunities at the current node with the existing flows".* Hence the relaxation criterion as used in Dijlstra's algorithm has been modified to incorporate network coding advantage. This modified relaxation criteria forms the crux of the algorithm. The modified relaxation criterion is given below:

$$d[j].length = \min(d[j].length\,, d[i].length + w[i][j] * NCA[j] \qquad (16)$$

A similar work on incorporating network coding into a routing framework can be found in [25]. Here they use an iterative algorithm, which has been termed NCA_Dijkstra algorithm. However our proposal differs from theirs in many aspects. Firstly, their aim is minimization of total number of transmissions, while our algorithm aims at enhancing lifetime. Our algorithm formally incorporates network coding into the routing metric and avoids an iterative approach. This aspect is absent in the work in [25] and is the novelty of this work.

To find whether coding opportunity exists at a node, we make use of the knowledge of potential coding structures depicted in Fig. (2). If opportunistic listening is not allowed,

Algorithm. 2 .*NCA_DIJKSTRA(G, s_f, d_f)*

```
Require: Network G = (N,A)
Require : traffic demand f, $\mathcal{F}_f(s_f, d_f)$
Ensure : flow f is laid as per the metric & coding opportunities are exploited
Initialize dist, predecessor, coding_group & coded_flow values in all nodes.
dist[$s_f$] := 0
$Q := N$        //set of all nodes in the graph
while ($Q \neq \emptyset$) do
    $u$ := vertex in $Q$ with smallest dist[ ]
    if dist[$u$] == INFINITY || $u == d_f$ then
        break        //either no path exists or destination reached
    end if
    remove vertex $u$ from $Q$.
    for each neigbour $v$ of $u$, where $v \in Q$ do
        function NET_COD ($\mathcal{F}$,v, u, f) is called & NCA[v] computed
        value := dist[$u$] + w[$u$][$v$] * NCA[v]
        if value < dist[$u$] then
            dist[$v$] := value
            predecessor[$v$] := $u$
            if coding is possible at node $u$ then
                coding_group[$v$] := $g'$
                coded_flow[$v$] := $f'$
            end if
        end if
    end for
end while

//path traceback
$P_f$ := empty sequence
$u := d_f$
while predecessor[u] ≠ NIL do
    insert $u$ at the beginning of list $P_f$
    $u$ := predecessor[$u$]
    copy back the network coding details into the respective nodes
end while
```

then only Alice-Bob structure is the possible coding structure. If opportunistic coding is allowed, then we need to explore the existence of any of the 4 potential coding structures at a node. The CA_LMR algorithm in its simplest form employs opportunistic listening and allows only a single coded group per node. The CA_LMR algorithm is given in Algorithm 1.

Algorithm. 1 . *CA_LMR(x_1, x_2, x_3, x_4) algorithm*

```
Require: Network G = (N,A)
Require : F traffic demands, $\mathcal{F}_f(s_f, d_f)$ ,$\forall f \in \mathcal{F}$
Ensure : all paths are laid and data transfer occurs till network death
while ($E_i > 0, \forall i \in N$)
    for f = 1 to F do
        Run NCA_ DIJKSTRA (G, $s_f$, $d_f$) algorithm for the given flow $\mathcal{F}_f(s_f, d_f)$
        Update the path $\mathcal{F}_f$ and coding opportunities at each node.
    end for
    for p = 1 to λ do
        Send a packet through each of the shortest paths.
        if $E_i \leq 0$ for at least one node then
            break
        end if
    end for
end while
```

Algorithm 1 shows the proposed CA_LMR(x_1, x_2, x_3 , x_4). The NCA_DIJKSTRA algorithm, as given in Algorithm 2, is called for every flow and it generates paths keeping into account, the existing flows. The NCA_DIJKSTRA algorithm uses 2 new fields when compared to the traditional Dijkstra's algorithm. Also, for every neighbor of the current node, a function named NET_COD, given as Function1 is called, which takes care of finding the coding opportunities and computing NCA.

C. Reconfiguring the paths and multi-group coding

The CA_LMR algorithm lays the paths in a sequential manner. While finding paths, the algorithm detects whether some form of coding is possible with the existing flows, that can improve lifetime. Evidently, the first flow has no existing flows to check with, and the final flow being laid will have all other paths at its disposal. So, we propose a path reconfiguration scheme so that more coding chances can be discovered.

After the first iteration, all paths have been generated. We now remove the first flow and lay it once again taking into account the existence of other flows. This is continued sequentially for other flows as well. Thus we iterate the algorithm a few times (*NUM_ITERATIONS*) or till all the paths converge, whichever is earlier. The modified algorithm is termed as CA_LMR_CONF(x_1, x_2, x_3, x_4), and is given below as Algorithm 3.

In section II, it was mentioned that there are $\mathcal{M}_i$ coded groups in each node *i*. Each coded group can have two or more flows. If there is no coding at node *i,* then its value is zero. At this point, two alternatives are possible: either we allow only a single coding group at a node or we allow multiple coding groups. Allowing multiple coded groups in a node would result

Algorithm. 3 . *CA_LMR_CONF(x_1, x_2, x_3, x_4) algorithm*

```
Require: network G = (N,A)
Require : F traffic demands, $\mathcal{F}_f(s_f, d_f)$ ,$\forall f \in \mathcal{F}$
Ensure : all paths are laid and data transfer occurs till network death
while ($E_i > 0, \forall i \in N$)
    for k = 1 to NUM_ITERATIONS do
        for f = 1 to F do
            Remove flow f and the dependent coding opportunities.
            Run NCA_ DIJKSTRA (G, $s_f$, $d_f$) algorithm for the given flow $\mathcal{F}_f(s_f, d_f)$ pair.
            Update the path $\mathcal{F}_f$ and coding opportunities at each node.
        end for
        if all paths converge then
            break
        end if
    end for
    for p = 1 to λ do
        Send a packet through each of the shortest paths.
        if $E_i \leq 0$ for at least one node then
            break
        end if
    end for
end while
```

Function 1 . *NET_COD* ($\mathcal{F}$,*v, u, f*)

```
Require: set of flows $\mathcal{F}$ and all related coding opportunities
Require: traffic demand $f$, $\mathcal{F}_f(s_f, d_f)$
Require : current node $u$ and probable successor $v$
Ensure : compute the best $NCA[v]$ exploiting coding
if ( $\mathcal{M}_u \neq 0$) then
    for $i$ = 1 to $\mathcal{M}_u$ do
        if flow $f$ can be coded with flows in group $i$ then
            Compute $NCA[i]$
        else
            $NCA[i]$ := 1
        end if
    end for
    $NCA[v] \leftarrow$ minimum of all $NCA[i]$
    if ($NCA[v] < 1$) then
        add flow $f$ to coding_group $i$
    end if
end if

if ($\mathcal{M}_u$ = 0) || flow $f$ couldn't be added to existing groups then
    for $f'$ = 1 to $F$ do
        if flow $f'$ is uncoded & can be coded with $f$ then
            Compute $NCA[f']$
        else
            $NCA[f']$ := 1
        end if
    end for
    $NCA[v] \leftarrow$ minimum of all $NCA[f']$
    if ($NCA[v] < 1$) then
        make a coding_group $g' = \mathcal{M}_u + 1$ ,with flows $f$ &$f'$.
    end if
end if
if $NCA[v]$ == 1 then
    Add the flow $f$ to un-coded flows list
end if
```

in increased complexity in terms of the buffer space required and the number of comparisons required. In this paper, CA_LMR_SCG denotes single coded group and CA_LMR_MCG denotes multiple coded groups.

The pseudo-code presented in Algorithm 1 and 2 is applicable for multiple-coded groups case also. It reduces to single-coded group case when $\mathcal{M}_u$, as given in algorithm 2, is limited to one.

C. Complexity Analysis

As explained before, the CA_LMR algorithm is a modified version of Dijksta's algorithm. The complexity of Dijkstra's algorithm is known to be $O(N^2)$ if the labeled vertices are stored in an array or list. A quick analysis of the pseudo-code reveals that the algorithm differs from Dijkstra's algorithm at the point where we check all the neighbours of the current node. At this point, we have to compare the current flow with all the existing flows in the network. In the best case, there are no flows to be compared with, and in the worst case, all the *F-1* flows are there to be compared.

In our proposed algorithm, the labeled vertices are stored in a list. The computational complexity of adding a vertex or decreasing the distance of a vertex is $O(1)$. To extract the minimum vertex we examine each labeled vertex for a cost of $O(N)$. There are $N-1$ calls to find the node with minimum cost. At any point in the shortest-path computation, there are at most N labeled vertices. Hence the above operations add a computational cost of $O(N^2)$. There |A| labeling operations and less than |A| calls to reduce the cost which together add a cost of $O(|A|)$, where A is the set of edges.. Thus the computational complexity of Dijkstra's algorithm using an array or list to store the labeled vertices is $O(N^2 + A) = O(N^2)$. In this modifed algorithm, for every neighbor of the current node, there are a maximum of F comparisons. Once a working node is fixed, these comparisons are made for all possible neighbours, and then the minima is extracted from the list. Hence, the complexity of the algorithm is given by $O(|F|N^2)$.

IV. PERFORMANCE EVALUATION

In this section, we compare the performance of the proposed algorithm with the existing algorithms. We compare our algorithm with minimum-hop based routing (MH), Minimum total energy routing (MTE) and Flow Augmentation (FA) algorithm [3]. The proposed algorithm has been simulated in C programming language.

The simulation setup consists of N nodes randomly distributed in 50m*50m area, constituting a static, wireless multi-hop network. The nodes are equipped with omni-directional antenna. The maximum transmission range of each node is set to 25m, i.e. $j \in S_i$ if $d_{ij} < 25m$. The path loss factor α has been set to 2. The reception energy and idle mode energy has been assumed to be negligible compared to the transmit power. The energy spent in sending control packets and such other overheads are ignored in the simulation. All the simulations have been averaged over 50 randomly generated networks. The paths are computed using a modified Dijkstra's algorithm in a centralized manner. A distributed implementation of the same algorithm is also possible.

In the simulation, we have assumed lossless wireless links. Though practically not feasible, it suffices for our simulation, because we are comparing the performance of several algorithms under the same conditions. If lossy links are assumed, then there will be packet losses and consequently, packet retransmissions. Since network coding is being employed, it will help reduce the total number of re-transmissions also, thus enhancing the lifetime. The network model used in the simulation assumes the lower layers in the stack to be ideal. The proposed algorithm computes the shortest paths for the F randomly generated unicast flows, all having the same data-rate. If in a static multi-hop wireless network we have varying degree of unreliability among the wireless links the routing problem takes a very different shape [26] even in a situation where network coding is not used. Then an energy optimal routing under delay constraint may strongly tend to prefer certain counter intuitive longer hop count paths containing low energy nodes in order to avoid some highly unreliable ("blacklisted") links to protect itself from the overhead of repeated retransmission which effectively modifies both the energy and delay cost of a path for the worse. This may obscure out the real tricks and methods that one may be able to reveal regarding finding and utilizing network coding opportunities. As this problem takes quite a different shape we believe it deserves a thorough different type of investigation in future.

The performance metrics that will be used for comparison are network lifetime (L), transmissions per flow (TPF), Energy per packet (EPP), and Standard Deviation (STDEV). Network lifetime is the time to death of the first node in the network. In our work, the total number of packets that has been reliably delivered across the network is considered as network lifetime. TPF [25] is the ratio of total number of transmissions to the total number of successfully delivered packets. EPP [23] is the average energy expended in transmitting a packet from its source to destination. Standard Deviation (STDEV) of residual energy at the nodes is a measure of how well the traffic has been balanced across the nodes in the network. In all the figures network lifetime is measured in terms of number of packets the network could route before its "death".

The proposed algorithm CA_LMR, in its simplest form uses network coding with opportunistic listening and allows only a single coded group per node (CA_LMR_OPP_SCG). The advantage obtained by allowing multiple coded groups in a node and path reconfiguration will be dealt with later. In our simulation, CA_LMR(1,x,y,0) is similar to FA(1,x,y) except for the fact that reception energy has not been accounted for in our algorithm. Also CA_LMR(0,0,0,0) and CA_LMR(1,0,0,0) denote MH and MTE routing respectively.

A. Performance of CA_LMR(x_1,x_2,x_3,x_4) algorithm

Fig. 3 compares the lifetime obtained with the proposed algorithm with MH, MTE and FA(x_1, x_2, x_3) algorithms for different number of nodes and a fixed number of traffic demands. There are 40 flows to be routed across the network and the update interval λ has been set to 1. All nodes have the same initial energy of 10^5 units.

Fig. 3 shows that the proposed algorithm gives an improved lifetime over the existing algorithms that do not use network coding. It was shown in [3] that FA(1,10,10) works best for multiple flow cases. It can be seen that CA_LMR(1,10,10,1) gives an enhanced lifetime over CA_LMR(1,1,0,1). It can also be seen that as *N* increases, the relative increase in lifetime over the other schemes is more. This emphasizes the usefulness of network coding in larger networks.

The plot of transmissions per flow is depicted in Fig. 4. It can be seen that minimum hop routing gives the lowest TPF, which is very obvious since it takes the path with lowest number of hops, hence lowest number of transmissions. An important point to be observed is that inclusion of network coding helps reduce the number of transmissions required to deliver a set of packets, and hence gives a lower TPF compared to no-coding case for both CA_LMR(1,1,0,1) and CA_LMR(1,10,10,1).

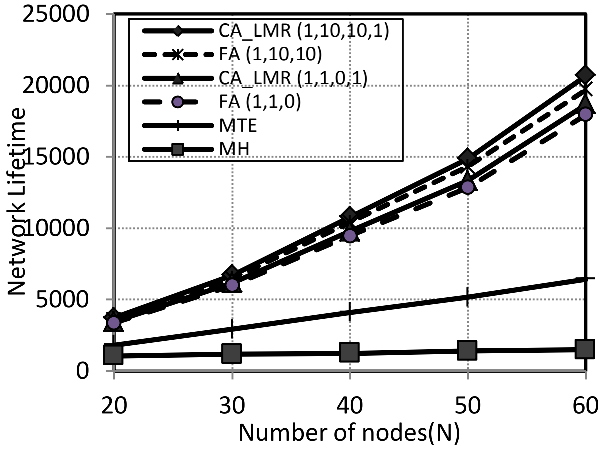


Fig.3– Network lifetime with different routing schemes for fixed number of flows.

A look at the energy expended per packet in different schemes in Fig. 5 reveals that MH routing spends the most energy and MTE expends the least. This is because hop-based routing tries to minimize the hop count, and hence to attain that, it expends large power to reach their next-hops. Among CA_LMR(1,1,0,0), ie. FA(1,1,0) and CA_LMR(1,10,10,0) ie. FA(1,10,10), the former one spends lesser energy. It can be observed that inclusion of network coding reduces the energy spent per packet in both cases.

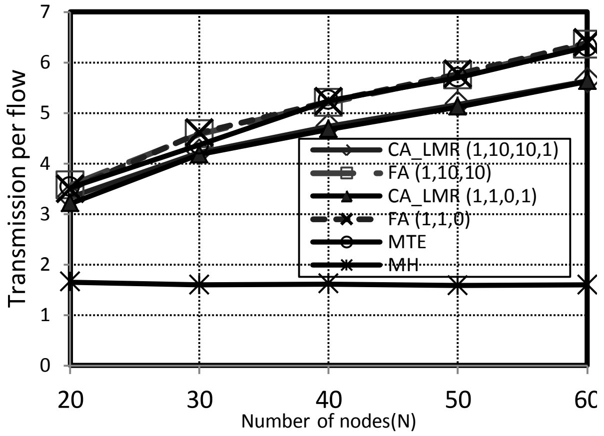


Fig.4 Transmissions/flow (TPF) with different routing schemes for fixed number of flows.

Fig 6 shows the standard deviation of residual energy at the nodes after the network's death. It is a measure of the traffic balancing capability of the routing scheme among the nodes. It can be seen that CA_LMR(1,10,10,0) or equivalently FA(1,10,10) performs much better compared to all the other schemes. Contrary to the previous cases, the standard deviation increased when network coding was used. Despite this load balancing, much improvement in lifetime could not be obtained because energy expended per packet (EPP) for CA_LMR(1,10,10,0) is higher.

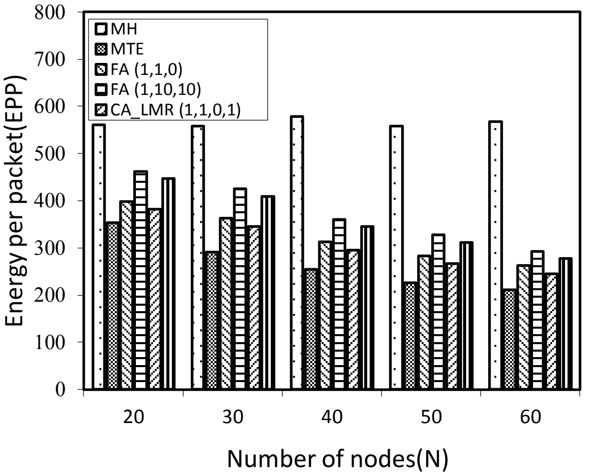


Fig. 5 - Energy per packet (EPP) with different routing schemes for fixed number of flows.

A notable difference between CA_LMR(1,10,10,1) and CA_LMR(1,1,0,1) is the percentage of coded transmissions achieved by both. It can be observed that CA_LMR(1,1,0,1) routing exploits more coding opportunities than CA_LMR(1,10,10,1). There is a constant 2% difference between both the schemes. But as it was shown, the lifetime of CA_LMR(1,10,10,1) is higher. This is attributed to the excellent traffic balancing property of CA_LMR(1,10,10,1) as depicted in Fig.6. Though CA_LMR(1,10,10,1) utilizes lesser coding opportunities and selects paths with slightly higher energy consumption, it utilizes the energy of the nodes in the best possible way. Fig.7 shows the plot of lifetime against number of flows employing CA_LMR(1,1,0,1) algorithm. It was observed that for a fixed number of nodes (i.e. a fixed energy pool), network lifetime increases with the number of traffic flows. The relative increase in lifetime with increase in node count is also evident from the figure. Another point to be observed is that the improvement obtained in lifetime, is much more pronounced in a network with larger number of nodes. This is because the numbers of coding chances rise due to presence of more number of flows, which account for the improved lifetime.

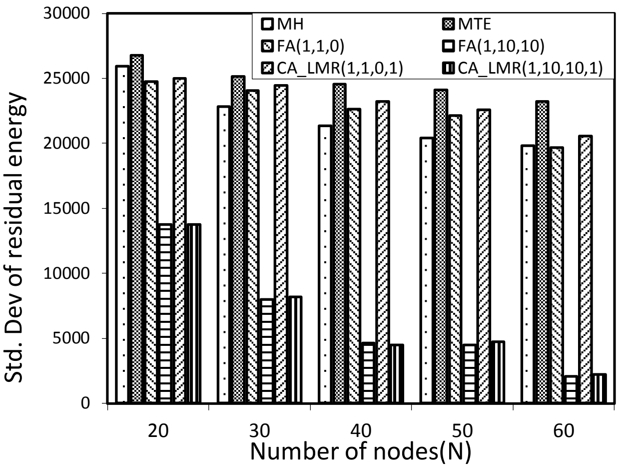


Fig.6. Std. Dev. of residual energy with different routing schemes for fixed number of flows.

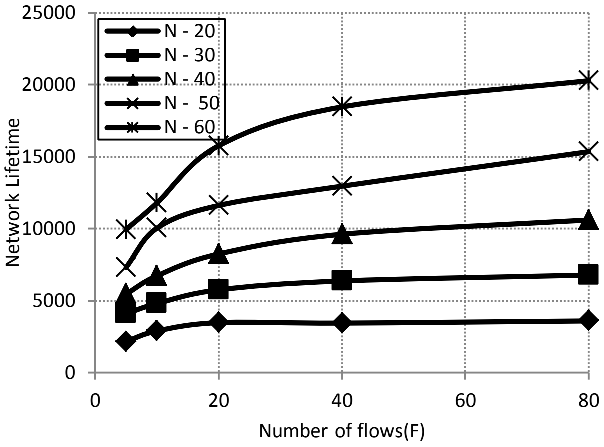


Fig.7 - Network lifetime with different routing schemes for fixed number of nodes.

B. Effect of inclusion of reconfiguration and multiple coded-groups

We compare the lifetime obtained by using CA_LMR(1,1,0,1), and then including path reconfiguration and multiple coded-groups, for a scenario with 40 flows, a path loss factor of 2, all nodes with initial energy of 10^4 units, and varying number of nodes. CA_LMR_CONF_MCG denotes the proposed algorithm with path reconfiguration and multiple coded-groups.

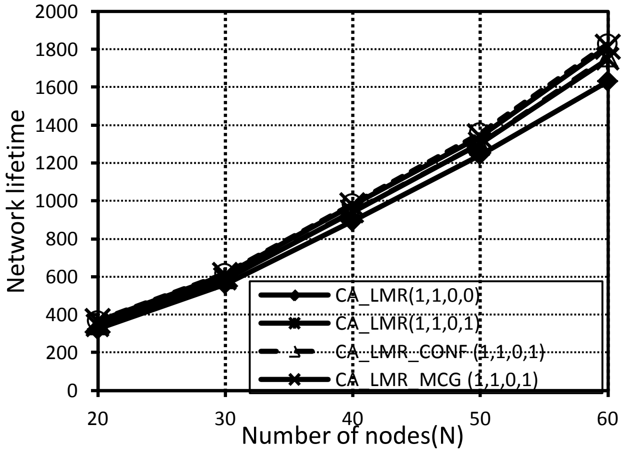


Fig. 8 Effect of inclusion of multiple coded-groups and reconfiguration on lifetime with CA_LMR(1,1,0,1) .

Fig. 8 shows the network lifetime using CA_LMR(1,1,0,0) (I,e. FA(1,1,0)) and all variants of CA_LMR(1,1,0,1). The maximum number of iterations for path reconfiguration was fixed as 10. It was seen that most times the paths converged in 2 to 3 iterations. It can be observed that CA_LMR_CONF and CA_LMR gave similar lifetime values. This shows that the algorithm, in its simple form itself is sufficient to find lifetime-maximal paths. However, allowing multiple coded-groups in the node was found to improve network lifetime by around 7-8% compared to single-coded groups.

The effect of inclusion of path reconfiguration and multiple-coded groups on transmissions per flow (TPF) is depicted in Fig. 9. It can be seen that reconfiguration doesn't give a significant improvement either in lifetime or in TPF. This is due to the fact that the algorithm, in its first iteration itself helps find most coding opportunities. However, allowing multiple coding groups per node helps in reducing the number of transmissions, by discovering more coding opportunities.

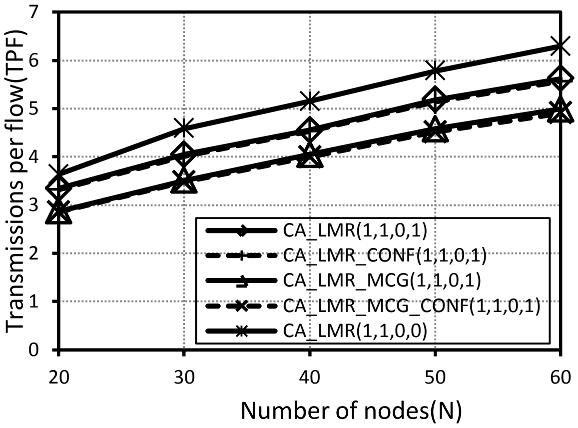


Fig. 9. TPF for CA_LMR(1,1,0,1) with inclusion of multiple coded-groups and reconfiguration

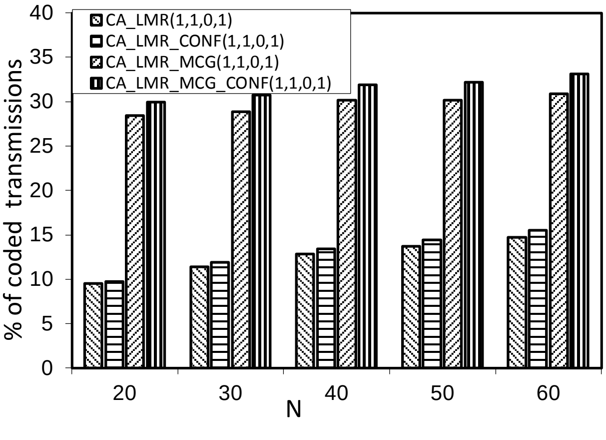


Fig.10 Percentage of coding opportunities with inclusion of multiple coded-groups and reconfiguration .

The increase in coding opportunities by allowing multiple coded groups is evident from Fig. 10. The percentage of coded transmissions increases from 14% to 34% on inclusion of multiple coded groups for a 60-node network. The reason for this increase is that in single coded-group case, the maximum number of coding opportunities possible is *N*, hence many coding spots have to be dropped in that case. This problem is rectified by allowing as many coding groups as possible.

For a scenario with fixed number of nodes and varying number of traffic demands, network lifetime was found to increase with inclusion of multiple-coded groups and also with increase in number of flows. This is due to availability of more number of coding opportunities. It was observed that in single coded-group case, the percentage of coded transmissions increase up to a certain number of flows, and then reduces as in Fig. 11. This is due to the fact that the total coding chances are restricted by the number of nodes in the network. This explains the necessity of allowing multiple coded-groups per node.

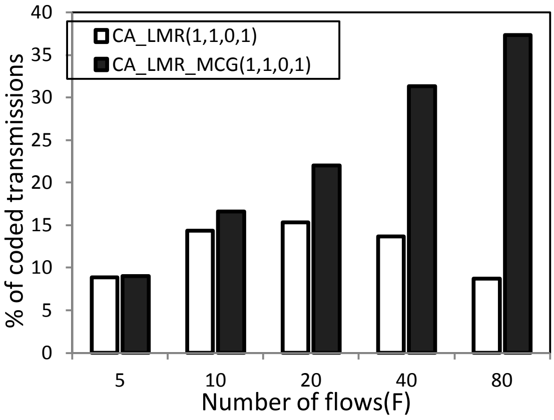


Fig.11 - Percentage of coded transmissions with CA_LMR(1,1,0,1) with single & multiple coded-groups.

C. Effect of update interval on lifetime

In practical networking scenario, the frequency of updates is a very important issue. More frequent updates require exchange of many control packets and eventually higher overhead. In this section, analyze the effect of update interval on lifetime. There are 40 flows to be routed and all nodes have same initial energy of 10^5 units. The effect of update interval on lifetime for CA_LMR(1,10,10,1) is depicted in Fig. 12. It can be seen that if the updates are more frequent, the lifetime obtained is more. Lifetime of the network is least if the network is not updated at all.

An interesting point to be observed is the effect of update interval on lifetime for CA_LMR(1,1,0,1) algorithm (i.e, FA(1,1,0)), shown in Fig. 13. When no update is done, the lifetime is the least as in the above case. But here, the lifetime values are less sensitive to update intervals.

It can be seen that update intervals of 1 and 5 give almost the same lifetime values, while the decrease in lifetime is less steeper compared to CA_LMR(1,10,10,1) algorithm.

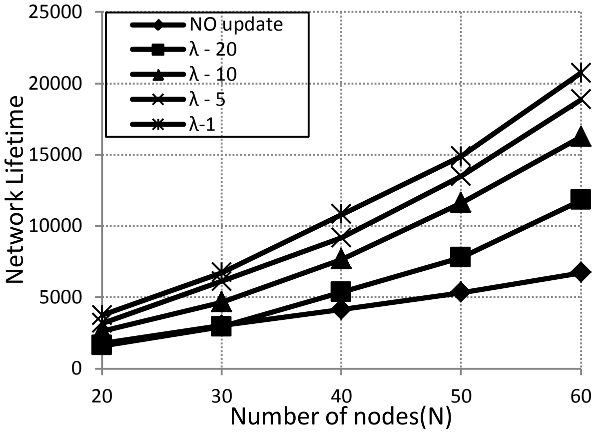


Fig.12 - Effect of update interval on network lifetime for CA_LMR(1,10,10,1).

The energy consumed in exchange of control packets for routing updates has been ignored in this work. But if it is included, it also takes up a share of the node's energy. The CA_LMR(1,1,0,1) algorithm can then be employed keeping a large update interval. This will reduce the lifetime marginally, but will save a lot of energy that would have been spent in routing updates. Keeping these things into consideration, it can be said that CA_LMR(1,1,0,1) will perform better .

D. Effect of aggressive coding

It can be observed that increasing the value of x_4 in the routing metric paves way for exploring and utilizing many more network coding opportunities in the network. Thus, we can say that the value of x_4 is indicative of the aggressiveness of network coding. In this subsection, we analyze the effect of employing aggressive coding in our proposed algorithm, on network performance. We analyze the effects on CA_LMR_MCG $(1,1,0,x_4)$ and CA_LMR_MCG$(1,10,10,x_4)$. There are 40 flows to be routed in a 40-node network and all nodes have initial energy of 10^4 units.

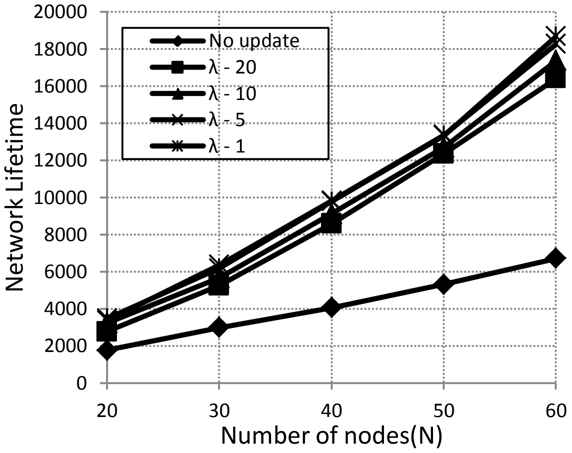


Fig.13- Effect of update interval on network lifetime for CA_LMR(1,1,0,1).

Fig.14 shows the effect of aggressive coding on network lifetime using CA_LMR_MCG$(1,1,0,x_4)$. It can be seen that lifetime is highest when x_4 is set to 1, and the lifetime *decreases* as x_4 increases. This shows that, even though coding can improve lifetime, excessive network coding can hamper lifetime. This fact was also demonstrated in [6]. The reason for this is that as we make the algorithm find more opportunities, the routing schemes tends to select longer and complex paths and this results in greater power usage per node and the network dies off relatively early.

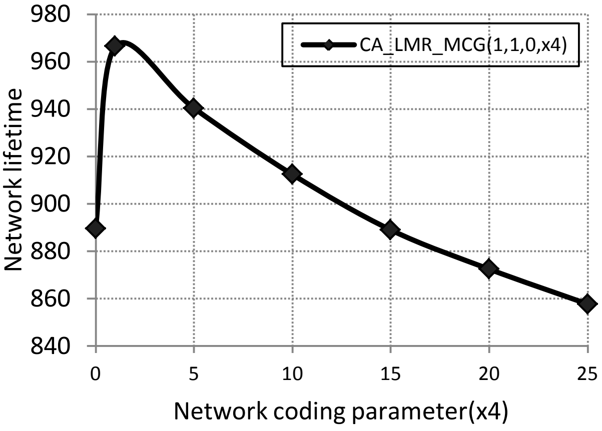


Fig.14- Variation of lifetime with network coding parameter(x_4)

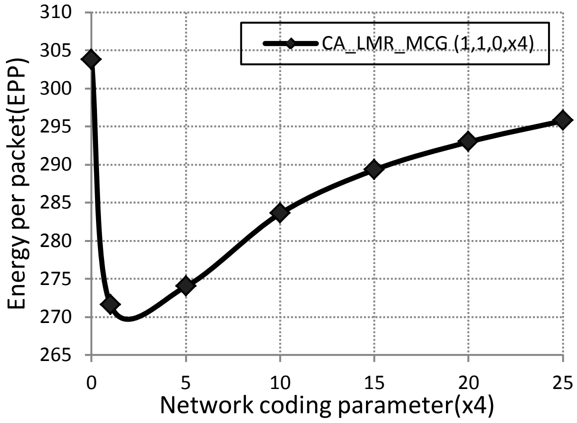


Fig.15- Variation of EPP with network coding parameter(x_4)

Fig. 15 shows the effect of aggressive coding on the energy expended per packet (EPP) for CA_LMR_MCG$(1,1,0,x_4)$ algorithm. It was already shown that energy per packet is higher in the absence of network coding. However EPP dips to the minimum value at around $x_4 = 1$ or 2, and then again it starts increasing. This shows that from an energy efficiency point of view, network coding should be done but not in excess. The reason for the increase is that, the algorithm starts taking longer and complex paths in pursuit of coding chances, and eventually ends up spending more energy.

V. CONCLUSION

In wireless ad-hoc networks with limited energy resources coding-aware routing is an important issue [27-29]. It has also been seen that MTE routing is not a good solution in this context from lifetime perspective.. In our work, we propose a routing algorithm that exploits the network coding opportunities to enhance network lifetime. The algorithm aims to maximize the information transferred between the source-destination pairs which is indicative of network lifetime. The CA_LMR algorithm uses network coding with opportunistic listening and has been found to enhance network lifetime over existing algorithms. The effect of inclusion of multiple coded-groups per node and path reconfiguration was also explored. It was found that multiple coded-groups per node increased the number of coded transmissions to a significant extent. The effect of routing update intervals was also studied. In future work, we aim to study the effects of overhead on the performance of the algorithm and also aim to implement a distributed version of the algorithm. The concept of co-operative communication [30] can also be incorporated into the routing framework.

REFERENCES

[1] R. Ahlswede, N. Cai, S-Y. R. Li, "Network Information Flow", *IEEE Trans. on Information Theory* ,Vol. 46, Issue no. 4, pp 1204-1216, July 2000.

[2] J.-H. Chang and L. Tassiulas, “Maximum lifetime routing in wireless sensor networks,” ATIRP Conf., College Park, MD, Mar. 2000.

[3] J.-H. Chang and L. Tassiulas, “Maximum lifetime routing for wireless sensor network”, *IEEE/ACM Trans. on Networking*, vol. 12, Issue. 4, pp. 609 – 618, August 2004.

[4] C. F. Chiasserini, R.R. Rao, "Energy Efficient Battery Management," *IEEE JSAC Wireless Series*, Vol. 19, No. 7, pp. 1235 -1245, July 2001.

[5] B. N. Clark, C. J. Colbourn, and D. S. Johnson, “Unit Disk Graphs”, *Discrete Mathematics*, Vol.86, Issues. 1-3, pp. 165-177, December 1990.

[6] N. Gaddam, S. A. Gathala, D. Lastine, A. Somani, “Energy Minimization through Network Coding for Lifetime Constrained Wireless Networks” in *Proc. of 17th International Conference on computer Communications and Networks(ICCCN ’08)*, pp.1-6, Aug. 2008.

[7] Tracey C. Ho, Yu-Han Chang, and Keesook J. Han, “On Constructive Network Coding for Multiple Unicasts”, *44th Annual Allerton Conference on Communication, Control, and Computing*, September 2006.

[8] Yongfa Hong ,Juan Xu and Changjun Jiang, “Lifetime Maximization in Wireless Sensor Networks with Network Coding”, in *International Conference on Wireless Commns., Networking and Mobile Computing, 2007. WiCom 2007.* ,pp-2527-2530 ,September 2007.

[9] H.Hosseinmardi, F. Lahouti, “Multicast lifetime maximization using network coding: A cross-layer approach”, in *24th Biennial Symposium on Communications,* pp-1-4, June 2008.

[10] I. Kang and R. Poovendran, “Maximizing network lifetime of broadcasting over wireless stationary ad hoc networks”, in *ACM journal of Mobile Networks and Applications (MONET),* Vol.10, Issue.6, pp.879-896, December 2005.

[11] S. Katti, H. Rahul, D. Katabi, M. Medard, J Crowcroft, "XORs in the Air: Practical Wireless Network Coding", in *IEEE/ACM Trans. on Networking*, Vol. 16, Issue no. 3, pp.497-510, June 2008.

[12] Dongkyun Kim, J. J. Garcia-Luna-Aceves, Katia Obraczka, Juan-Carlos Cano & Pietro Manzoni, “Routing Mechanisms for Mobile Ad Hoc Networks Based on the Energy Drain Rate” *IEEE Transaction on Mobile Computing*, Vol.2, Issue.2, pp 161-173,Apr-Jun 2003.

[13] R. Koetter, M. Medard, "An Algebraic Approach to Network Coding", in *IEEE/ACM Trans. on Networking*, Vol. 11, Issue no. 5, pp 782-795, Oct. 2003.

[14] S-Y. Li, R. W. Yeung, N. Cai, "Linear Network Coding", in *IEEE Trans. on Information Theory*, Vol. 49, Issue no. 2, pp 371-381, Feb 2003.

[15] Q. Li, J. Aslam, and D. Rus, “Online power-aware routing in wireless ad hoc networks,” presented *at the IEEE/ACM Int. Conf. Mobile Computing and Networking (MobiCom 2001)*, Rome, Italy, July 2001.

[16] Z. Li and B. Li, “Network coding: The case for multiple unicast sessions”, Allerton Conference on Communications, 2004.

[17] S. Lindsey, K. Sivalingam, and C. S. Raghavendra, “Power Optimization in Routing Protocols for Wireless and Mobile Networks,” in *Handbook of Wireless Networks and Mobile Computing.* New York: Wiley, 2002.

[18] Kumar Padmanabh and Rajarshi Roy, "Maximum Lifetime Routing in Wireless Sensor Network by Minimizing Various Chemical limitations of Practical Batteries", in *IEEE International Conference on Consumer Network and Communication (IEEE CCNC 2007),* pp. 525-529 ,Las Vegas, USA, January 2007

[19] V. Rodoplu and T. H. Meng, “Minimum energy mobile wireless networks,” in *Proc. IEEE Int. Conf. Communications (ICC’98)*, vol. 3, Atlanta, GA, June 1998, pp. 1633–1639.

[20] S. Sengupta, S. Rayanchu, and S. Banerjee, ”An Analysis of Wireless Network Coding for Unicast Sessions: The Case for Coding-Aware Routing” in *Proc. of IEEE INFOCOM, 26th IEEE International Conference on Computer Communications*, pp.1028-1036, May 2007.

[21] S. Singh, M.Woo, and C. S. Raghavendra, “Power-aware routing in mobile ad hoc networks,” in *Proc. 4th Annu. IEEE/ACM Int. Conf. Mobile Computing and Networking*, Dallas, TX, Oct. 1998, pp. 181–190.

[22] C.-K. Toh, “Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks,” *IEEE Commun. Mag.*, pp. 138–147, June 2001.

[23] T. Tran, D. Nguyen and T. Nguyen, "A Case for Joint Network Coding and Power Control for Wireless Linear Networks", in *6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks Workshops*, *SECON Workshops '09*, pp-1-6, June 2009.

[24] Y. Wu, P. A. Chou, and S.-Y. Kung, “Minimum-energy multicast in mobile ad hoc networks using network coding,” in *IEEE Transactions on Communications*, Vol. 53, No. 11, November 2005.

[25] Z.Xia, Yu Hong-Yi, “Network coding aware joint routing for multiple unicast flows in Wireless Multihop Networks” in *4th International Conf. on Wireless Communications, Networking and Mobile Computing, 2008. (WiCOM '08.)*, pp.1-5, October 2008.

[26] Zhenhua Zou, Pablo Soldati, Haibo Zhang, Mikael Johansson: Energy-Efficient Deadline-Constrained Maximum Reliability Forwarding in Lossy Networks. IEEE Transactions on Wireless Communications 11(10): 3474-3483 (2012)

[27] Tahir Mehmood, Lavy Libman, Hooman Reisi Dehkordi, Sanjay K. Jha, Optimal opportunistic routing and network coding for bidirectional wireless flows, Computer Networks: The International Journal of Computer and Telecommunications Networking, v.57 n.18, p.4030-4046, December, 2013

[28] Ronghui Hou , Sikai Qu , King-Shan Lui , Jiandong Li, Coding- and interference-aware routing protocol in wireless networks, Computer Communications, v.36 n.17-18, p.1745-1753, November, 2013

[29] Muhammad Azhar Iqbal , Bin Dai , Benxiong Huang , A. Hassan , Shui Yu, Review: Survey of network coding-aware routing protocols in wireless networks, Journal of Network and Computer Applications, v.34 n.6, p.1956-1970, November, 2011.

[30] Tamaghna Acharya, Samiran Chattopadhyay, Rajarshi Roy, “Maximum Lifetime Broadcasting in cooperative multi-hop wireless adhoc networks’ , in *Int. J. Ad Hoc and Ubiquitous Computing, Vol. 6, No. 1, 2010.*

Mr. Rigil S. now serves Cisco systems, Bangalore, India as an engineer. He did his M.Tech. from Dept. of E & ECE, Indian Institute of Technology, Kharagpur with a specialization in Telecommunication system Engineering (2010). He did his B.Tech in Electronics and Communication from TKMCE (Kerala University) (2007). He did study in prestigious Trinity Lyceum, Kollam, Kerala, India. His area of interest is Communication networking, switching and routing, networking standards, protocols, management, software, and hardware.

Dr. Rajarshi Roy now serves Dept. of E & ECE, Indian Institute of Technology, Kharagpur, India as an associate professor. He did work for ISI, Kolkata and HUT, Helsinki, Finland as academic visitor and Bell Labs, Holmdel, NJ, USA as summer intern. He worked for Lucent, Bangalore and Comverse, MA, USA. He got his PhD in Electrical Engineering with a specialization in networks from Polytechnic University, Brooklyn, NY, USA (Now, known as Polytechnic Engineering faculty of New York University) (2001). He got his M.Sc. (Engg.) in Electrical Communication Engineering from Indian Institute of Science, Bangalore, India (1995) and his Bachelor of Engineering in Electronics and Telecommunication Engineering with 1st Class (Hons.) from Jadavpur University, Kolkata, India (1992). His area of interest is Communication networks, Communication theory, network science and control, optimization, resource allocation, and performance evaluation of networked complex systems.