IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 6, 2019, accepted January 9, 2020, date of publication January 23, 2020, date of current version February 4, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2969029

DAG-Coder: Directed Acyclic Graph-Based Network Coding for Reliable Wireless Sensor Networks

EMAN AL-HAWRI, NOÉLIA CORREIA, AND ALVARO BARRADAS
Center for Electronic, Optoelectronic and Telecommunications (CEOT), Faculty of Science and Technology, University of Algarve, 8005-139 Faro, Portugal

Corresponding author: Eman Al-Hawri (esalhawri@ualg.pt)

This work was supported by the Foundation for Science and Technology (FCT) from Portugal through the Center for Electronic, Optoelectronic and Telecommunications (CEOT) under Project UID/MULTI/00631/2019. The work of Eman Al-Hawri was supported by the Thamar University, Yemen.

ABSTRACT Wireless sensor devices are usually powered with limited non-rechargeable batteries. This energy can easily be depleted when no wise consumption procedures are used, leaving the sensor non-functional. As most of the energy is consumed during the transmission of data, developing efficient data manipulation and transmission approaches is crucial and still an open problem that attracts the attention of many research groups. In this article a directed acyclic graph (DAG) based dissemination approach, where clustering and network coding techniques are applied, is proposed. As a main goal, this work aims at improving the network reliability (ensure recovery of lost packets), while minimizing energy consumption and balancing load at gateways. The proposed approach is compared to state-of-art approaches in terms of network reliability and energy saving trade-off. Experimental results demonstrate the strong performance of the proposed work.

INDEX TERMS Wireless sensor network, network coding, directed acyclic graph, clustering, P2P overlay network.

I. INTRODUCTION

By intelligently combining data packets, using binary or linear combinations, network coding is able to improve bandwidth utilization in wired networks for a certain amount of data to be delivered from a source to multiple destinations, allowing an increase in throughput [1]. In Wireless Sensor Networks (WSNs), however, there are multiple sources sending data notifications to gateway/sink nodes, meaning that the traffic flow is mainly many-to-one [2], [3]. For this reason, network coding approaches in WSNs have mainly focused on how to reduce packet loss, avoiding packet retransmissions throughout the network. Although packet loss reduction is achieved at the expense of encoding/decoding overhead, a balance between energy efficiency and packet loss can be achieved because retransmissions are avoided [3]. That is, network lifetime increases when compared with scenarios where retransmission of packets exists, each packet traversing multiple hops. The cost of transceivers is also a prime concern in large-scale deployments and for this reason some authors try to reduce the number of nodes performing encoding [4].

The main arguments behind reducing packet loss, instead of being concerned with bandwidth, are that [3]: *i*) nodes typically produce small volumes of data, and for this reason bandwidth ends up not being that critical in many sensor networks; *ii*) losses are mainly caused by links with high packet-erasure probability (bad link quality) and not due to congestion. For these reasons, multipath communication is explored by many authors. Some authors maintain multiple disjoint paths between communicating end-points, as in [5], while others propose to disseminate coded packets through all available paths (all nodes are encoding nodes), as in [3] and [12]. The approach in [5] does not explore all available paths, and packets may not be relayed even though a viable path exists. The approach is also more adequate for wireless mesh networks having both source and destination nodes at the wireless section. Regarding the approaches in [3] and [12], only the one from [3] is designed for many-to-one data dissemination in WSNs, where traffic is sent towards a gateway/sink. However, too many packets are transmitted

The associate editor coordinating the review of this manuscript and approving it for publication was Cesar Vargas-Rosales.

because a node transmits its own packets, packets from its children (a tree rooted at sink/gateway is assumed), overheard packets and generated coded packets. These packets end up being received by the parent node and heard by all neighbours, meaning that the number of packets traveling in the network will be quite large, in particular near the gateways. Therefore, congestion will be critical in those areas of the network. In [12] all nodes are supposed to receive the generated data, being more suited to disseminate control data among all nodes. This may be applied to many-to-one data dissemination if the dissemination process stops when data reaches the gateway/sink.

In [6], the transmission of too many packets is avoided through the use of multiple gateways and failure scenarios. Sets of critical links, with very high packet-erasure probability, are seen as failure scenarios and encoding nodes are placed so that a hearing path is ensured for linear combinations to reach one of the gateways. A P2P overlay network (distributed storage/retrieval system) is necessary to federate all gateways, ensuring the recovery of lost packets when non-lost related packets are forwarded towards different gateways. When comparing the approaches in [3] and [12] with the one in [6], the latter is suitable for network environments having predictable critical links (failure scenarios), while the approaches in [3] and [12] are suitable for network environments where packet loss location is not predictable. That is, as long as critical/failing links are clearly identified, the approach in [6] presents significant advantages because fewer coded packets are generated, allowing for longer network lifetimes and higher goodputs. The problem, however, is that failure scenarios may be difficult to determine, which is the case of dynamic environments. Note that goodput measures received original data/packets only, while throughput measures all data/packets. The transmission of non innovative coded packets increases the time required to deliver a certain amount of data, because bandwidth is being taken, meaning that goodput (received original data per time unit) decreases.

The approach proposed here in this article explores the best of [3] and [6] approaches. More specifically:

- Pre-defined failure scenarios are avoided, which allows it to be used in dynamic environments.
- A Directed Acyclic Graph (DAG)-based dissemination approach is used, instead of disseminating coded packets through all available paths as in [3]. This allows for a reduction in the number of generated coded packets, when compared to [3].
- Only nodes selected to become Cluster Heads (CHs) participate in the DAG. Since encoding is performed by CHs only, the number of generated coded packets reduces even more.
- Similarly to [6], a P2P overlay storage system is used to federate the multiple gateways/proxies. This allows packet dissemination through shorter paths and ensures the recovery of lost packets when non-lost related packets are forwarded towards different gateways. This also leads to fewer coded packets.

Our main contributions are the following. First, a new DAG-based design framework for the implementation of network coding in WSNs is proposed. Secondly, an optimization model able to design the best DAG-Coder, given a specific scenario as input, is developed. This means choosing CHs and links forming the DAG, a topological order for packet routing, and ensuring flow conservation towards a gateway. Such approach improves network reliability (recovery of lost packets) while avoiding having to transmit too many packets (energy saving) and while balancing load at gateways (congestion reduction), for scenarios where the location of critical links is not predictable. Lastly, the DAG-Coder and two other approaches from the literature are implemented and their performance is compared.

The remainder of this article is organized as follows. Section II presents some network coding principles first, and then relevant related work is discussed. In Section III, the adopted network architecture and data dissemination problem are defined. Section IV presents the mathematical formulation for the DAG-based approach, used to address the data dissemination problem. In Section V the performance analysis of the proposed approach and SenseCode is detailed. Sections VI and VII present the discussion of results and conclusions, respectively.

II. NETWORK CODING PRINCIPLES AND RELATED WORK

The store-and-forward technique is used in traditional networks to disseminate data packets without modifications. On the contrary, with network coding the encoding nodes can mix the content of incoming data packets before forwarding them [7], [8]. When using random linear encoding, r packets are combined in the form $\sum_{i=1}^{r} \alpha_i k_i$, where α_i is a coefficient generated over finite field $\mathbb{F}_{2^s}$, of size s ($\mathbb{F}_{2^8}$ is used in this work), and k_i is a specific packet. At the destination, the original packets can be retrieved as long as the destination receives enough linearly independent coded packets [9]. Fig. 1 illustrates an example where k_1 and k_2 are the packets to be encoded and sent to gateways G_1 and G_2. In this example the source S encodes these two packets and sends the resulting coded packets to the relay nodes R_1 and R_2, which forward the coded packets without any processing. When encoding node E receives the coded packets from R_1 and R_2, it encodes them and sends the resulting recoded packet to relay node R_3, which forwards it to gateways G_1 and G_2. Both gateways are able to decode these packets, retrieving the original ones, as long as they receive enough linearly independent coded packets.

In wired networks, network coding is mainly used to increase the throughput in one-to-many multicast transmissions. When traffic flow is many-to-one, as in WSNs, network coding can be used to decrease the packet loss, leading to greater network reliability. Thus, the broadcast nature of wireless transmissions, many times seen as a disadvantage, can help ensure reliability in an elegant way [10]. Any node listening to the packets can work as a next-hop, allowing for an easy tailoring of the flow to the network environment, and

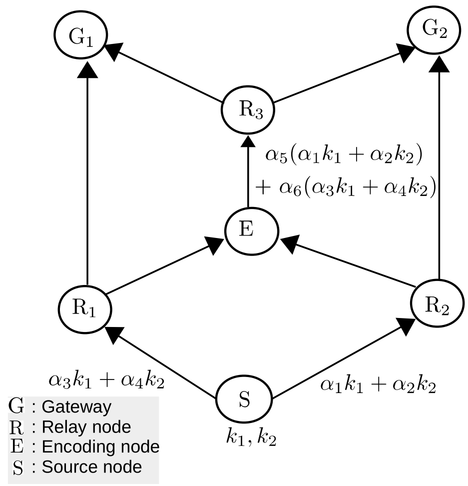


FIGURE 1. Random linear coding example.

TABLE 1. Packets overheard and sent in SenseCode for scenario in Fig. 2.

Node	Overheard	Sent
S_1	k_2	$k_1, k_1 + k_2$
S_2	k_1, k_3	$k_2, k_1 + k_2 + k_3$
S_3	k_2, k_4	$k_3, k_2 + k_3 + k_4$
S_4	k_3	$k_4, k_3 + k_4$
E_1	$k_3, k_2 + k_3 + k_4$	$k_1, k_2, k_3, k_2 + k_3 + k_4$
E_2	$k_2, k_1 + k_2 + k_3$	$k_3, k_4, k_2, k_1 + k_2 + k_3$

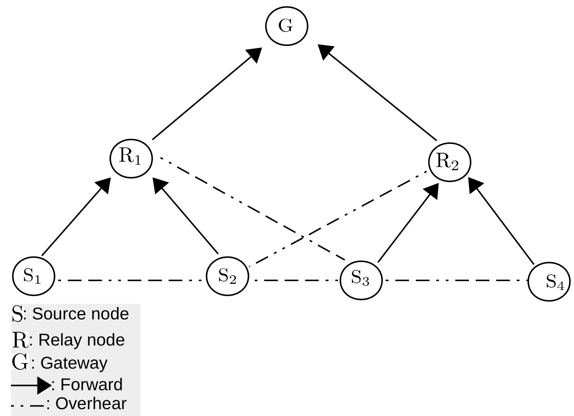


FIGURE 2. Routing tree in SenseCode.

accommodating different traffic patterns. As long as the gateway(s) receive(s) enough linearly independent coded packets, lost packets can be recovered. This decreases the number of required packet retransmissions. WiFi or WiMAX networks can also benefit from listening and binary network coding, as discussed in [11].

A. GENERAL DATA DISSEMINATION RELATED WORK

CodeDrip, discussed in [12], is a data dissemination protocol for WSNs that uses network coding. The aim of using network coding in CodDrip is to enhance the reliability and speed of dissemination while reducing the energy consumed. The authors argue that existing dissemination protocols try to avoid redundant transmissions by selectively transmitting the missing data. Such strategies reduce the energy consumed but cause large delays, an issue that is solved by CodeDrip due to the use of network coding. CodDrip uses the simplest network coding technique, the XOR with Galois field ($\mathbb{F}_2$). Since it has been designed to ensure that all nodes receive the propagated data, it ends up being more suited to disseminate control data in wireless networks. In [13], a network coding scheme is presented that optimizes the amount of overhearing. Since increasing the overhearing leads to more transmissions, and energy consumption, the authors try to solve this trade-off using probabilistic overhearing. That is, a node hears packets from neighbors in its range with a certain probability.

In SenseCode [3], network coding for many-to-one communication in WSNs is introduced. That is, multiple sources forward data packets towards a gateway using tree-based routing. In SenseCode, a node may deal with three kinds of messages: messages received from its children, messages overheard from its neighbors, and its own messages. To perform network coding, the node generates linear combinations of these messages. Applying this technique ensures that the sink will be able to recover the lost packets. Table 1 shows an example of how data is forwarded and overheard in SenseCode, assuming the routing tree depicted in Fig. 2.

In the example, each source S_i, $1 \leq i \leq 4$, sends its k_i packet towards gateway G using R_1 and R_2 as relay nodes. It is assumed that all nodes perform encoding, sources included. As seen in the Table 1, SenseCode ensures that the gateway G will receive the linear combinations of all packets even if one of the relay nodes fails to communicate its data to the gateway.

The NetCoDer in [14] applies linear network coding in an industrial context. A start topology communication model is assumed where multiple sensor devices send their data, in their assigned slot, to a coordinator at the middle of the topology. Nodes can act as relays, depending on the reliability of communications, retransmitting data during retransmission slots. Such proposal can only be applied to local wireless sensor networks having a star topology.

Inter-flow Network Coding based Opportunistic Routing (INCOR), in [15], incorporates both opportunistic routing and inter-flow network coding to increase the performance of WSNs. This approach exploits the broadcast nature of wireless and the spatial diversity of multi-hop wireless networks. The authors present a new metric to define the prioritization of forwarders in the candidate set of nodes. Then, they design the network coding based opportunistic routing using the defined metric. The authors in [16] propose an algorithm that uses network coding as a solution to reduce the energy consumption and to increase the network lifetime in multicast networks.

From all these proposals, SenseCode and CodeDrip are the only network coding based dissemination approaches with reliability concerns that can be applied in many-to-one scenarios.

B. CLUSTERING RELATED WORK

In [4], the authors study a cluster-based WSN model where network coding is applied at nodes located in the overloaded area (near the sink). The network is divided into two areas: cluster and bottleneck/overloaded. At the cluster area, nodes form clusters and every CH receives data from its members. At the bottleneck area, on the other hand, nodes are divided into relay and coding nodes. A relay node is responsible for forwarding data coming from clusters while coding nodes are responsible for coding the data coming from clusters. A similar approach is discussed in [17], but the network is divided into a square grid, and then in each square the optimum CH is selected based on the maximum normalized remaining energy. A different CH is selected at every round, in each square, in order to equally distribute the energy load between sensor nodes. This approach increases the network lifetime when compared with LEACH.

In [18], the authors claim that energy efficient clustering protocols like LEACH are concerned with network lifetime at the expense of reduced stability periods. Therefore, in order to increase the network stability, the authors propose an energy-aware heuristic that balances the load between nodes and subsequently increases the stability period. The main idea is to select the CHs in a deterministic way, based on the remaining energy. The concern is also to provide a full network coverage. In [19], CHs are chosen so that the lifespan of the sensor network is extended.

The previously discussed clustering based approaches do not have packet recovery concerns and/or are not suitable for network coding based many-to-one flows.

III. THE DATA DISSEMINATION PROBLEM

A. MOTIVATION AND ARCHITECTURE

The network coding based data dissemination protocols that can be adopted in many-to-one scenarios, and were designed with packet recovery concerns, are SenseCode and CodeDrip. SenseCode assumes data dissemination through a tree rooted at a sink/gateway. In this case, the failure of a link will affect all traffic coming from the subtree below it. Putting all nodes as encoding nodes is, therefore, a way of increasing the probability of packet recovery when links fail, but a large amount of packets will be generated. Diversity of routing to improve bandwidth utilization, as in Fig. 1, can not be explored because of the tree type routing structure. Regarding CodeDrip, the aim is to enhance the reliability and speed of dissemination, while reducing the energy consumed, but data dissemination stops only when data reaches all nodes, and not gateways in particular. That is, it has not been designed having many-to-one scenarios in mind, although its stopping criteria can change to data reaching one of the gateways, as previously explained.

Here in this article, a DAG-based dissemination approach is proposed that generates fewer packets than SenseCode and CodeDrip, reducing bandwidth requirements and increasing network lifetime, while increasing packet recovery to

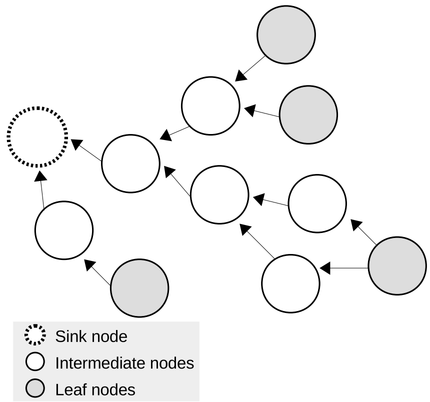


FIGURE 3. DAG with single sink.

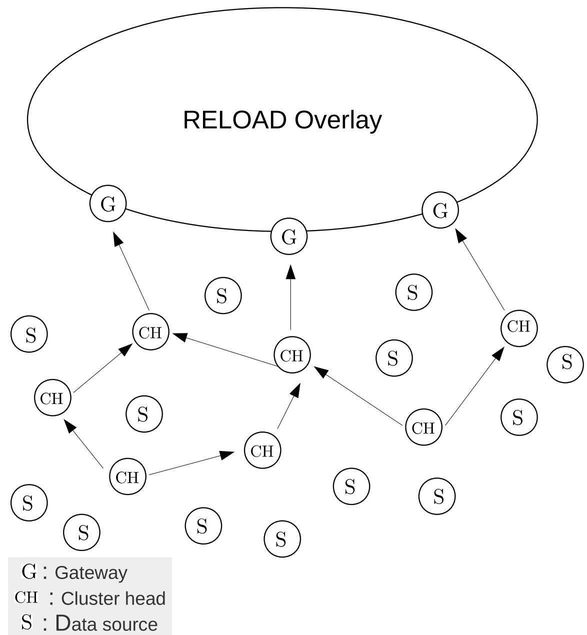


FIGURE 4. DAG with cluster heads.

achieve reliability. Also, and contrarily to CodeDrip, it has been designed having many-to-one flows in mind. A DAG structure is illustrated in Fig. 3. Any DAG has at least one topological ordering, which means that for every directed edge (n_i, n_j), node n_i comes before n_j in the ordering. The proposed approach assumes that gateways are peers in a P2P overlay, which allows diversity in routing towards different gateways to be explored. That is, packets reaching different gateways share a storage system that will allow recovery of lost data packets, even if the required linearly independent combinations have traveled through different routes. These are key features for the proposed approach to perform better than SenseCode and CodeDrip. This comes at the expense of some required planning (CHs and DAG). Such architecture is illustrated in Fig. 4.

The proposed approach works under the following assumptions:

- A wireless node can be selected to work as a collector (CH) or as a non-collector node.
- Non-collector nodes must be associated with a collector node, and send their data to it.
- Collector nodes generate coded packets using: *i*) packets received from lower topological order collector nodes; *ii*) packets from their members (non-collector nodes); *iii*) its own packets.
- A collector node has two or more links to other collector nodes. Original data packets, and generated coded packets, are sent through such links. Routing follows the topological ordering of collector nodes.

B. PROBLEM DEFINITION

Definition 1 (DAG-based Network Coding for WSNs (DNC-WSNs)): Given a constrained sensor network graph $G(\mathcal{N}, \mathcal{L})$, where $\mathcal{N}$ are the nodes and $\mathcal{L}$ are wireless communication channels (links), each with a weight reflecting the required transmission power, determine which nodes of $\mathcal{N}$ should be collector nodes, performing network coding, and which links of $\mathcal{L}$ should be at the DAG, for routing of packets following the topological order of collector nodes, so that energy consumption is minimized while ensuring data flow towards gateways and load balancing at gateways.

Packet loss recovery is possible because all collector nodes will be doing encoding, and sending coded and original packets through multiple paths towards multiple gateways. Energy consumption is minimized because coded packets are transmitted by CHs only. In case of no link failures, all original packets reach one of the gateways, which means that there will be 100% packet recovery.

IV. MATHEMATICAL MODEL

A. NOTATION AND ASSUMPTIONS

Let us assume a network graph $G(\mathcal{N}, \mathcal{L})$, where $w(l)$ denotes the weight (transmission power) of directed link $l \in \mathcal{L}$. Assume also that $\mathcal{G} \subset \mathcal{N}$ denotes the set of gateways.

A topological ordering of nodes in $\mathcal{N}$ is possible if and only if the graph has no directed cycles. In other words, if it is a DAG. Any topological order of $\mathcal{N}$ is any total order τ such that if $(n_i, n_j) \in \mathcal{L}$, then n_i precedes n_j in τ. That is, $\tau_{n_i} \leq \tau_{n_j}$. To incorporate a topological order at an instance of the DNC-WSN problem, it is assumed that τ_n is predefined for gateways: $\tau_n = 1$, if $n \in \mathcal{G}$. For every other $n \in \mathcal{N} \backslash \mathcal{G}$, since no predefined CHs exist (any n can be selected to become CH) and any node can be a data source (there will be no predefined leafs), while forwarding data from others, a topological order must be dynamically found by the optimizer, while taking into account the objective function (goal) and additional constraints. For every $n \in \mathcal{N} \backslash \mathcal{G}$: $0 \leq \tau_n < 1$.

The variables of the problem are the following:

- β_n One if wireless node $n \in \mathcal{N} \backslash \mathcal{G}$ is selected to become a CH, participating in the DAG, zero otherwise.
- σ_l One if link $l \in \mathcal{L}$ is to participate in the DAG (used for data delivery), zero otherwise.
- δ_l^s Percentage of data from source $s \in \mathcal{N} \backslash \mathcal{G}$ that flows through link l.
- τ_n Topological order of $n \in \mathcal{N} \backslash \mathcal{G}$ in the DAG.
- ξ_n Auxiliar variable to avoid having a fixed DAG outdegree at node $n \in \mathcal{N} \backslash \mathcal{G}$.

B. FORMALIZATION

In this section the mathematical model of the DNC-WSN problem is formalized. This requires choosing CHs and links forming the DAG, a topological order for packet routing, and ensuring flow conservation towards a gateway. Among all possible solutions, the one minimizing energy consumption and balancing load at gateways, while ensuring the recovery of lost packets in case of link failure, should be found. Since some diversity in routing is required when using network coding for packet recovery, and since this can be achieved with two outgoing links from CHs, the mathematical model should promote solutions with no more than two outgoing links from CHs, for energy saving purposes.

After solving such mathematical model to generate the DAG, and find CHs, a second step follows where CHs perform network coding operations on packets (see Section V-A).

1) OBJECTIVE FUNCTION

$$\text{Minimize} \sum_{\{n \in \mathcal{N}\}} \xi_n + \frac{\sum_{\{n \in \mathcal{N}\}} \beta_n + \sum_{\{l \in \mathcal{L}\}} w(l) \times \sigma_l}{\Delta}, \quad (1)$$

where $\Delta = |\mathcal{N}| \times |\mathcal{L}|$ so that the second component of the objective function does not compete with the first. The first component of the objective function minimizes ξ_n variables, which encourages not using a single outgoing link per CH (see Eq. 7 and its explanation), for diversity in routing. The second component is used to minimize energy consumption, which also leads to load balancing at gateways because CHs will use nearer gateways, in order to save energy. The number of gateways and their distribution is pre-planned to serve well a population of nodes. In other words, CHs are selected in a way that energy consumption is minimized and balanced.

2) CLUSTER HEADS

$$\sum_{\{l \in \mathcal{L}: src(l)=n\}} \beta_{dst(l)} \geq 1 - \beta_n, \quad \forall n \in \mathcal{N} \backslash \mathcal{G} \quad (2)$$

where $src(l)$ and $dst(l)$ are the source and destination endpoints of directed link l, respectively. Constraints (2) state that if a node is not CH (collector node) then it must be associated with a CH, for its data to be delivered.

3) DATA DELIVERY

$$\sum_{\{l \in \mathcal{L}: src(l)=n\}} \delta_l^s - \sum_{\{l \in \mathcal{L}: dst(l)=n\}} \delta_l^s = \begin{cases} \beta_s, & \textit{if } n = s \\ 0, & \textit{otherwise} \end{cases}, \forall s, n \in \mathcal{N} \backslash \mathcal{G} \quad (3)$$

$$\sigma_l \geq \delta_l^s, \quad \forall l \in \mathcal{L}, \forall s \in \mathcal{N} \backslash \mathcal{G} \tag{4}$$

$$\sigma_l \leq \frac{1}{2}(\beta_{src(l)} + \beta_{dst(l)}), \quad \forall l \in \mathcal{L} \tag{5}$$

Constraints (3) ensure flow conservation from any source node s towards any gateway, using CHs as intermediate nodes, and thus avoiding disconnected paths at the DAG. Constraints (4) activate links used by the flows in (3), while constraints (5) ensure that the endpoints of any DAG link are CHs. That is, data flow towards the gateways occurs using CHs only.

4) ACYCLINESS

$$\tau_{dst(l)} - \tau_{src(l)} > -\Delta + \Delta \times \sigma_l, \quad \forall l \in \mathcal{L} \tag{6}$$

Constraints (6) define the topological order at the end points of used links, establishing acycliness. This is done for links participating in the DAG, information given by σ_l variables.

5) NODE DEGREE

$$\xi_n \geq 2 \times \beta_n - \sum_{\{l \in \mathcal{L}: l = src(n)\}} \sigma_l, \quad \forall n \in \mathcal{N} \backslash \mathcal{G} \tag{7}$$

In these constraints, the auxiliar variables ξ_n are used to avoid having a fixed DAG outdegree at CHs. Although an outdegree of 2 should be promoted (see Section III), this might not be possible in certain physical wireless topologies. This impossibility is not only related with the physical topology, but also with the topology order that is imposed to ensure acycliness. Note that, since the goal includes minimizing all ξ_n, and following constraints (7), the ξ_n variables become: 0, if n is not chosen to become CH; 1, if there is a single outgoing link from n; and 0, if there are two or more outgoing links from n. Thus, it is of interest to have 2 or more outgoing links (if CH), whenever possible, but the approach is flexible to have a single one, if more outgoing links are not possible. Since the objective function also includes minimizing the number of CHs and energy consumption, through link weights, the solutions tend to use 2 outgoing links at CHs, which avoids increasing packet transmissions more than needed.

6) BOUNDS AND BINARY ASSIGNMENTS

$$0 \leq \delta_l^s, \ \tau_n, \ \xi_n \leq 1; \ \sigma_l, \ \beta_n \in \{0, 1\}. \tag{8}$$

Expression (8) states the type of each decision variable, and bounds. Note that, although ξ_n variables have been defined as continuous variables, these will take 0 or 1 value because of expression (7) . Stating these as continuous variables, instead of binary, reduces problem complexity and in this particular case does not affect the solution.

CPLEX[1] optimizer is used to solve this problem. The solution found will be the optimal solution for the DNC-WSN problem instance under consideration. Note, however, that this is a Mixed Integer Linear Programming (MILP) problem, which will be difficult to solve for large network instances.

V. PERFORMANCE ANALYSIS

A. SCENARIO SETUP

Randomly generated physical topologies of 20 and 30 nodes were used to evaluate the performance of the DNC-WSN approach, SenseCode and CodeDrip. As previously mentioned, these are the network coding based data dissemination protocols that can be adopted in many-to-one scenarios, which is the case of WSNs. Both dense and sparse topologies were evaluated, and the distance between any two nodes is calculated using the Euclidean distance. A dense topology is assumed to have a connectivity degree of 0.3, while for a sparse topology this will be 0.2. The connectivity degree is calculated using $\frac{|\mathcal{L}|}{|\mathcal{N}| \times (|\mathcal{N}|-1)}$, where $\mathcal{L}$ is the set of available directed links and $\mathcal{N}$ is the set of network nodes. The connectivity degree is a consequence of the maximum coverage area specified for the nodes.

The evaluation follows two steps:

1) Solving the mathematical optimization model to select CHs and generate the DAG for the DNC-WSN approach. This is implemented in CPLEX.
2) Implementing random linear network coding for the solutions obtained in Step 1 (DNC-WSN approach). This step also includes the implementation of SenseCode and CodeDrip methods for comparison with DNC-WSN. Such step is implemented in Matlab[2]. In DNC-WSN, the encoding nodes will be the CHs, while in SenseCode and CodeDrip all nodes are encoding nodes. In CodeDrip, however, there will be encoding depending on a certain probability. At this step two versions of DNC-WSN are created for testing:
    a) Non-CHs perform overhearing, and forward any data heard from neighbours (besides their own data) towards the CH to which they are associated.
    b) Non-CHs perform no overhearing, and forward just their data towards the CH to which they are associated.

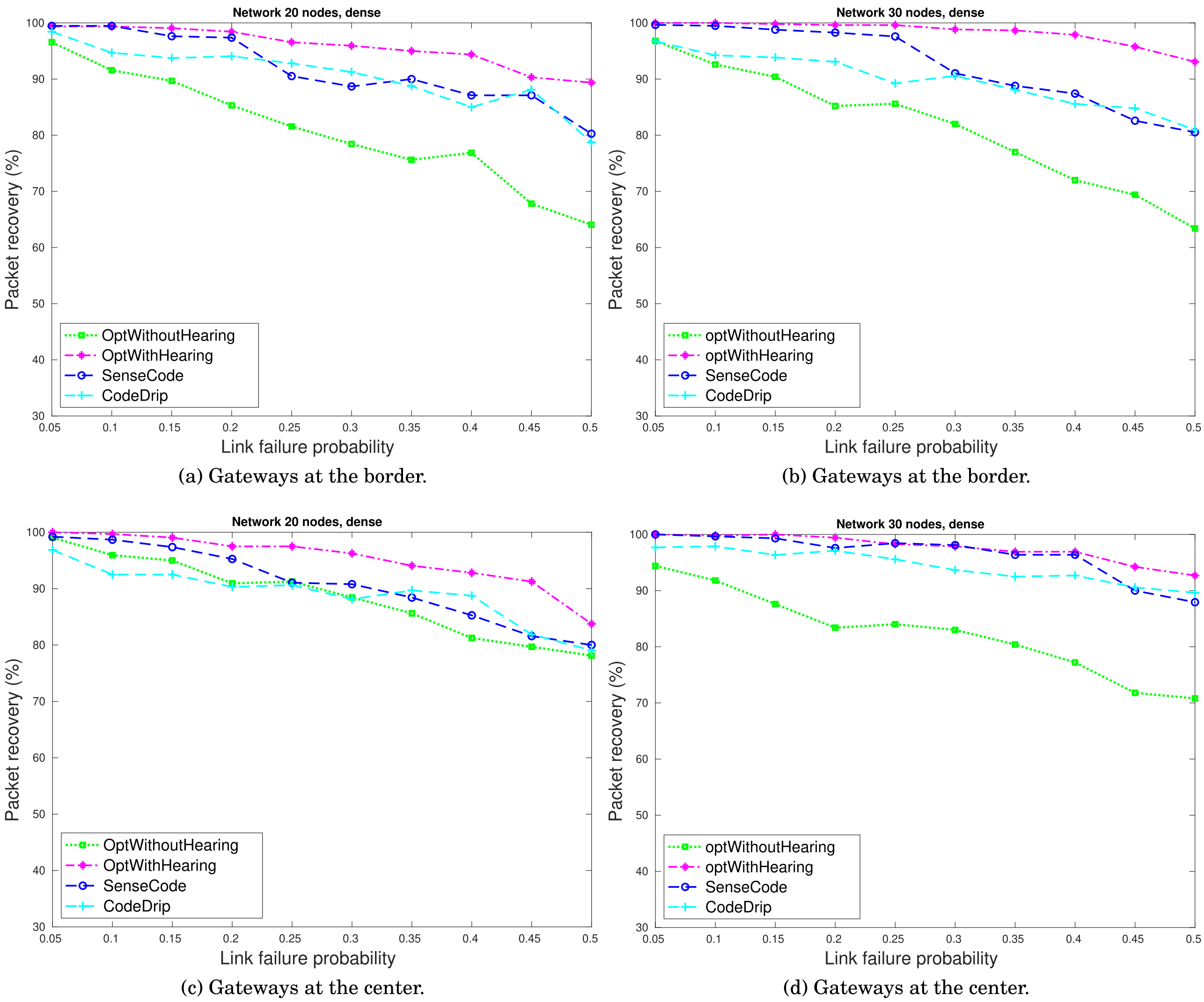
Two buffers are required at CHs. One will be used to store original packets received from associated sources, while the other will be used to store coded packets received from other CHs.

The tests performed, in order to compare DNC-WSN, SenseCode and CodeDrip, assume the following parameters:

- Number of gateways: DNC-WSN and CodeDrip consider 4 gateways, for both 20-node and 30-node topologies, while for SenseCode a single gateway is assumed, as in [3].
- Gateway locations: These are either at the center or at the border of the network.
- Link failure probability: Ranges from 0.05 to 0.5.

[1] IBM ILOG CPLEX Optimizer version 12.8.

[2] MathWorks, Inc.



(a) Gateways at the border. (b) Gateways at the border.

(c) Gateways at the center. (d) Gateways at the center.

FIGURE 5. Reliability for dense topologies.

- Network connectivity degree: 0.2 for sparse and 0.3 for dense.

In each scenario a link failure probability is assumed, so one or more links will be inoperable. For a specific link failure probability, 20 runs are performed (failing links change randomly at each run) and the average is used for the plotting of results.

B. PERFORMANCE METRICS

In the following plots two performance metrics are used:

- Reliability: Amount of original packets that are successfully stored at the P2P overlay (not lost or were recovered). A method achieves 100% reliability if all data packets sent from sources successfully arrive at the gateway(s).
- Packet transmissions: Total number of packet transmissions throughout all the wireless network section. The higher the number of packet transmissions, the greater the amount of energy consumed in each round.

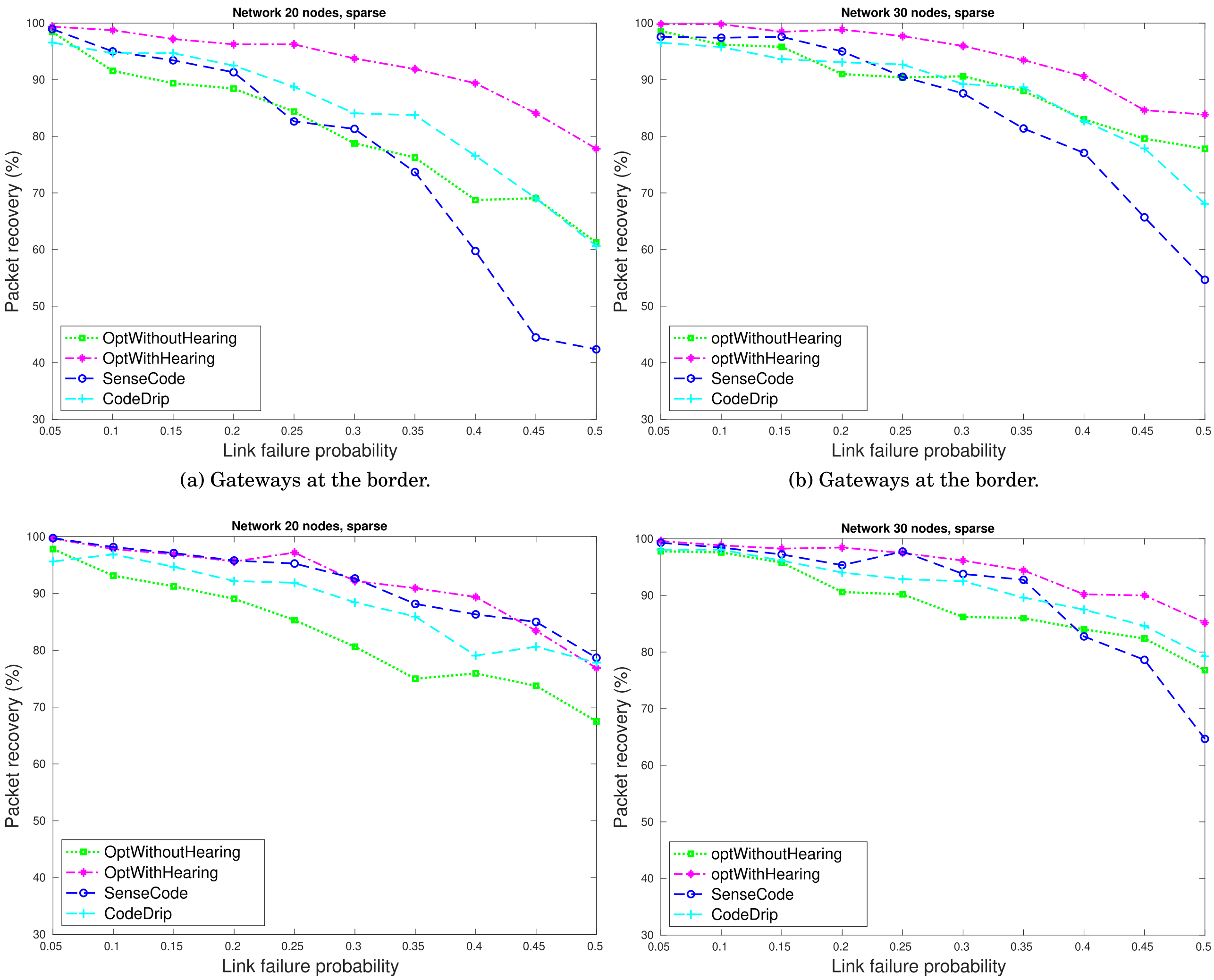
Nodes generate a single packet in each round, and a round ends when no more packets are traversing the network. Network coding operations are applied separately to each round. The nodes selected to become CHs, by the mathematical optimization model, will be the ones performing such network coding operations. Table 2 shows the overall number of CHs for the different types of topology.

C. ANALYSIS OF RESULTS

1) RELIABILITY

The results on reliability for dense and sparse network topologies are shown in Fig. 5 and Fig. 6, respectively. These results include the 20 and 30 node topology cases, for gateways located at the border and center of the network.

Regarding the impact of gateway location, SenseCode and CodeDrip seem not to perform well, when gateways are



(a) Gateways at the border.

(b) Gateways at the border.

(c) Gateways at the center.

(d) Gateways at the center.

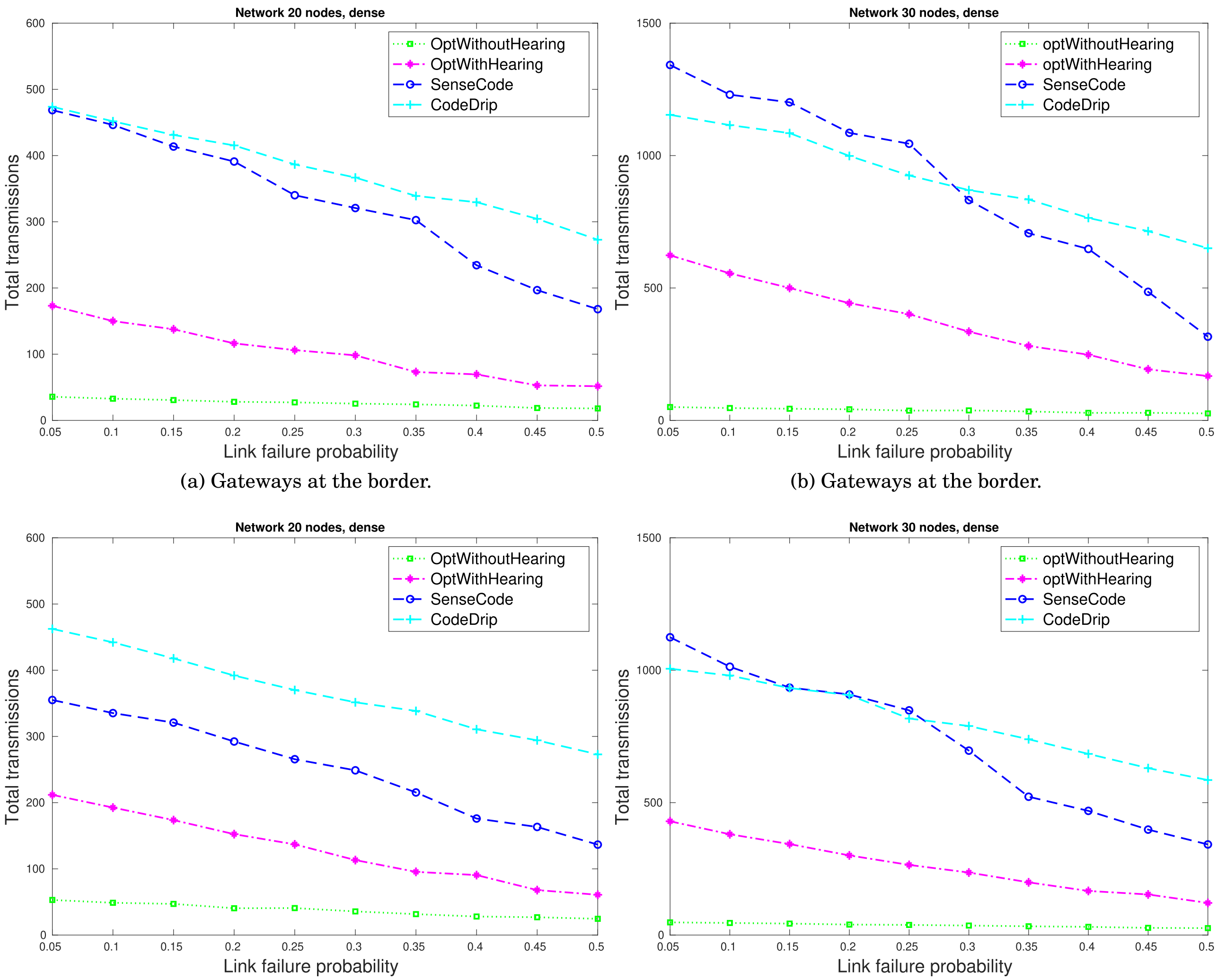
FIGURE 6. Reliability for sparse topologies.

TABLE 2. CHs selected by the optimization model.

# Nodes	Connectivity	Gw location	# CHs
20	Sparse	Border	6
20	Sparse	Center	9
20	Dense	Border	6
20	Dense	Center	10
30	Sparse	Border	12
30	Sparse	Center	10
30	Dense	Border	8
30	Dense	Center	6

located at the border, as link failure probability increases. This is more visible when network topologies are sparse. In addition, while larger networks generally show some improvement compared to their counterparts of smaller size, SenseCode seems to degrade for larger sparse networks when gateways are in the center. The other methods show some stability and improve or keep their performance when the network size increases. SenseCode has, therefore, scalability problems in these scenarios. It has also stability problems, like CodeDrip, because its performance is affected by gateway location and sparseness. Its poor performance in the mentioned scenarios is related to the fact that SenseCode is using a single tree rooted at a single gateway, having no routing diversity. Packets require more hops to get to the gateway, which increases the probability of packet loss, particularly in sparse topologies and when gateways are in the border. That is, packets must successfully traverse more links to reach the gateway. This leads to higher delays and waste of resources because successfully transmitted packets (at the first hops) may still be dropped further ahead, and for their transmission to happen other packets had to stay in queue.

CodeDrip presents no scalability problems (change in network size does not affect its behaviour) because it has routing diversity and explores the multiple gateways, but presents stability problems (performance is affected by the location of gateways and sparseness). Its poor performance in sparse topologies with gateways at the border is related to



(a) Gateways at the border. (b) Gateways at the border.

(c) Gateways at the center. (d) Gateways at the center.

FIGURE 7. Total transmissions for dense topologies.

CodeDrip's policy, which seems not to favor packet recovery in these scenarios, when compared with DNC-WSN. When a packet arrives, CodeDrip uses a probability to decide whether to send the packet or to combine it with other packet (randomly selected) for sending. XOR is used to combine packets. Although this could save some energy, some packets may not go through the coding process and some lost packets will not be recovered. This also explains the non recovered packets in CodeDrip when the link failure rate is low, which does not happen in SenseCode and DNC-WSN. The fact that SenseCode and CodeDrip are less adequate for gateways located at the border turns out to be a critical issue because such kind of network deployment is very common.

The DNC-WSN with hearing presents the best performance and, contrarily to SenseCode and CodeDrip, high stability since performance is not dependent on gateway location and network size. It is also less affected by network sparseness. This is related to routing diversity towards multiple gateways, explored by both DNC-WSN and CodeDrip, but DNC-WSN's criteria of performing linear encoding using all packets received from lower topological order collector nodes, packets from their members (non-collector nodes) and its own packets, seems to ensure the recovery of more packets than using the probabilistic approach, and XOR, of CodeDrip. The no hearing version of DNC-WSN ends up being ineffective.

2) PACKET TRANSMISSIONS

The number of packet transmissions for dense and sparse network topologies, with impact on energy and delay, are shown in Fig. 7 and Fig. 8, respectively. These include the 20 and 30 node topology cases, for gateways located at the border and center of the network.

From plots it is possible to observe that in SenseCode and CodeDrip there are too many packet transmissions, when

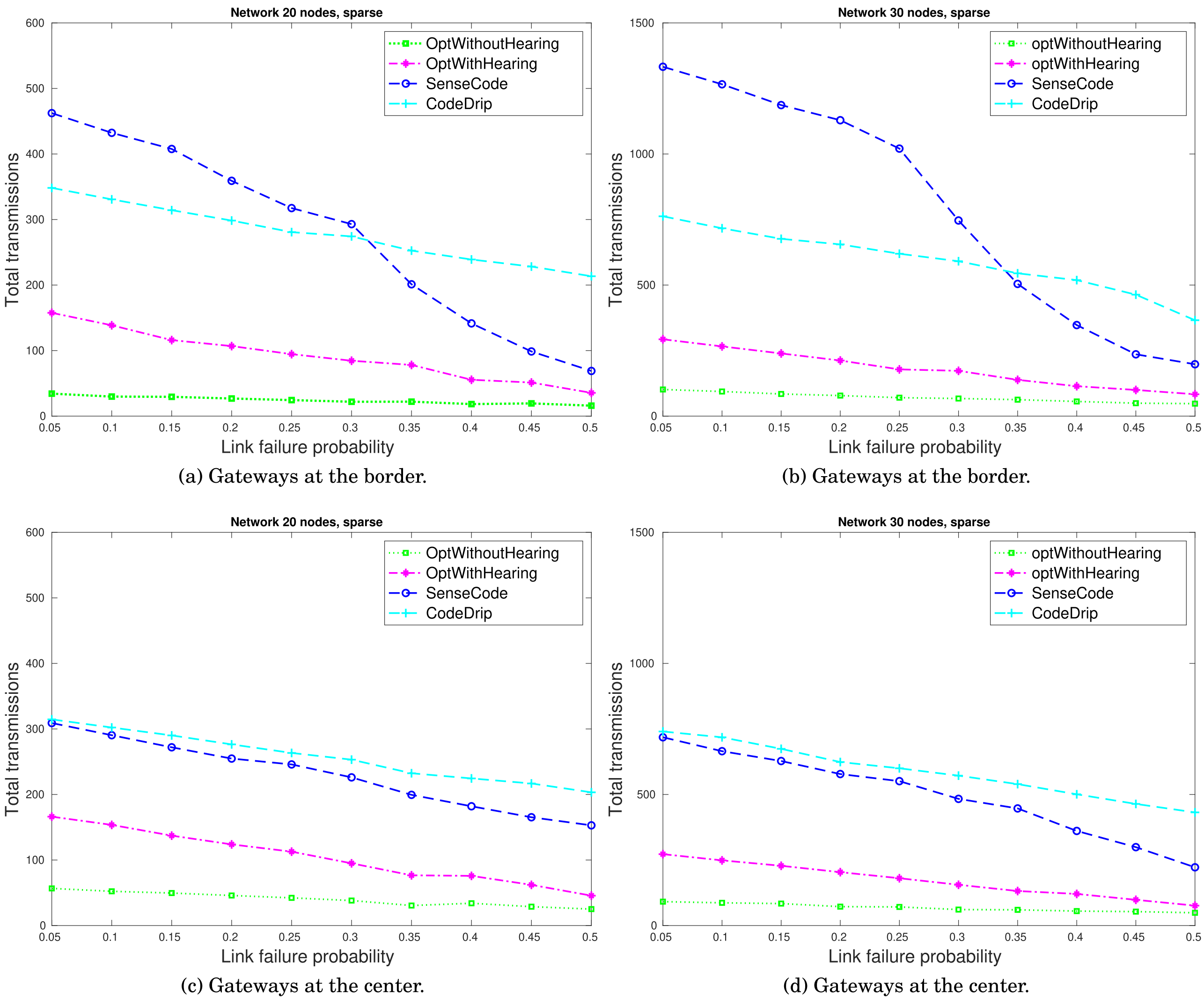


FIGURE 8. Total transmissions for sparse topologies.

compared with DNC-WSN with hearing and no hearing. This is because all nodes are encoding nodes. Since such transmissions do not translate into more packet recoveries than DNC-WSN, these approaches seem not to provide the best tradeoff between packet recovery and energy saving. In SenseCode, the number of packet transmissions is lower when gateways are at the center, due to fewer hops, and packet transmissions reduce significantly when the link failure probability increases, leading to few packet recoveries. This is more evident in sparse topologies, and is basically related with the tree based routing towards a single gateway.

CodeDrip and DNC-WSN show a linear behaviour, due to diversity in routing. Link failures affect less traffic flows, meaning that their impact is not as drastic as in SenseCode. Still, CodeDrip performs much more packet transmissions than DNC-WSN because all nodes are encoding nodes, while in DNC-WSN only CHs perform encoding. The XOR operations in CodeDrip also involve just two packets, which means that there will be more coded packets when compared with linear encoding. Although the DNC-WSN with hearing shows more packet transmissions than its no hearing version, these are required for packet recovery, meaning that DNC-WSN with hearing can be seen as the approach having the best packet recovery to energy saving tradeoff.

VI. DISCUSSION

In the previous section a comparison between the proposed DNC-WSN optimization model, SenseCode from [3], and CodeDrip from [12], is performed. In the proposed model, the data from sources can be protected against link failures by using overhearing (each source can hear its neighbors and send data to its CH). Furthermore, the data in CHs can be protected against link failures by using network coding (each CH encodes data from its members and sends both original and coded packets to CHs of higher topological order, or to the nearest gateway).

The version of DNC-WSN with better performance is the one with hearing since besides stability it shows the best tradeoff between energy saving and reliability. Its performance results from the fact that gateways act as peers in a P2P overlay network, allowing the recovery of packets even if their related coded packets have traveled towards different gateways. This lowers the required number of encoding nodes, for a certain recovery rate. In networks where failure probability is low, the no hearing variant may be more practical since there is less delay and more energy saving, leading to an increase of network lifetime.

VII. CONCLUSION

In this work, a DAG-based dissemination approach, using both clustering and network coding techniques, to achieve a balancing between reliability and energy efficiency is discussed. Also, an architecture for the federation of network coding based WSNs is proposed for storage purposes. To solve the DAG-based network coding problem, a mathematical model is developed to select CHs and generate the DAG. These CHs, forming the DAG, are the only nodes performing encoding operations in the wireless sensor section, while the other Non-CHs nodes perform just hearing. The performance evaluation shows that the DNC-WSN optimization model improves the network reliability, while reducing significantly the packet redundancy when compared to SenseCode and CodeDrip. The proposed DNC-WSN optimization model shows better results in both performance metrics: packet recovery and total number of transmissions.

REFERENCES

[1] L. Wang, Z. Yang, L. Xu, and Y. Yang, "NCVCS: Network-coding-based video conference system for mobile devices in multicast networks," *Ad Hoc Netw.*, vol. 45, pp. 13–21, Jul. 2016.

[2] C. Cirstea, M. Cernaianu, and A. Gontean, "Packet loss analysis in wireless sensor networks routing protocols," in *Proc. 35th Int. Conf. Telecommun. Signal Process. (TSP)*, Jul. 2012, pp. 37–41.

[3] L. Keller, E. Atsan, K. Argyraki, and C. Fragouli, "SenseCode: Network coding for reliable sensor networks," *Trans. Sensor Netw. (TOSN)*, vol. 9, no. 2, p. 25, 2013.

[4] Y. Hong and X. Liu, "Extension of lifetime with network coding in cluster based wireless sensor networks," in *Proc. 2nd Adv. Inf. Technol., Electron. Automat. Control Conf. (IAEAC)*, Mar. 2017, pp. 1219–1223.

[5] C. Gui, H. Chen, B. Sun, and Y. Song, "Energy efficient with network coding multipath routing algorithm in wireless sensor networks," *Int. J. Future Gener. Commun. Netw.*, vol. 7, no. 6, pp. 205–216, 2014.

[6] E. Al-Hawri, N. Correia, and A. Barradas, "Design of network coding based reliable sensor networks," *Ad Hoc Networks*, vol. 91, Aug. 2019, Art. no. 101870.

[7] R. Ahlswede, N. Cai, S.-Y. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.

[8] M. Médard and A. Sprintson, *Network Coding: Fundamentals and Applications*. New York, NY, USA: Academic, 2011.

[9] C. Fragouli, J.-Y. Le Boudec, and J. Widmer, "Network coding: An instant primer," *SIGCOMM Comput. Commun. Rev.*, vol. 36, no. 1, pp. 63–68, 2006.

[10] P. Ostovari, J. Wu, and A. Khreishah, "Network coding techniques for wireless and sensor networks," in *The Art of Wireless Sensor Networks*. Berlin, Germany: Springer, 2014, pp. 129–162.

[11] D. Nguyen, T. Tran, T. Nguyen, and B. Bose, "Wireless broadcast using network coding," *IEEE Trans. Veh. Technol.*, vol. 58, no. 2, pp. 914–925, Feb. 2009.

[12] N. S. R. Júnior, R. C. Tavares, M. A. Vieira, L. F. Vieira, and O. Gnawali, "CodeDrip: Improving data dissemination for wireless sensor networks with network coding," *Ad Hoc Networks*, vol. 54, pp. 42–52, Jan. 2017.

[13] S. Pfletschinger, M. Navarro, and C. Ibars, "Energy-efficient data collection in WSN with network coding," in *Proc. IEEE GLOBECOM Workshops (GC Wkshps)*, Dec. 2011, pp. 394–398.

[14] O. T. Valle, C. Montez, G. Medeiros de Araujo, F. Vasques, and R. Moraes, "NetCoDer: A retransmission mechanism for WSNs based on cooperative relays and network coding," *Sensors*, vol. 16, no. 6, p. 799, 2016.

[15] D. Zhu, X. Yang, W. Yu, C. Lu, and X. Fu, "INCOR: Inter-flow network coding based opportunistic routing in wireless mesh networks," in *Proc. Int. Conf. Commun. (ICC)*, Jun. 2015, pp. 3666–3671.

[16] B. Khodabakhshi and M. Khalily, "An energy efficient network coding model for wireless sensor networks," *Procedia Comput. Sci.*, vol. 98, pp. 157–162, 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050916321548

[17] R. Kaur and G. Kaur, "Load balanced clustering protocol for enhancing the lifetime of wireless sensor network," *Int. J. Adv. Technol. Eng. Explor.*, vol. 5, no. 46, pp. 326–334, 2018.

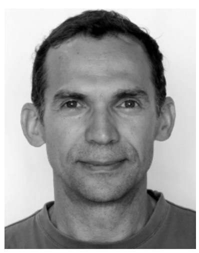
[18] N. Mittal, U. Singh, and B. S. Sohi, "A stable energy efficient clustering protocol for wireless sensor networks," *Wireless Netw.*, vol. 23, no. 6, pp. 1809–1821, 2017.

[19] J. A. Martins, A. Mazayev, N. Correia, G. Schütz, and A. Barradas, "GACN: Self-clustering genetic algorithm for constrained networks," *Commun. Lett.*, vol. 21, no. 3, pp. 628–631, 2017.

EMAN AL-HAWRI received the bachelor's degree in computer science from Thamar University, Thamar, Yemen, and the master's degree in informatics engineering from the University of Algarve, Faro, Portugal, where she is currently pursuing the Ph.D. degree in optimization and wireless sensor network.

NOÉLIA CORREIA received the B.Sc. and M.Sc. degrees in computer science from the University of Algarve, Faro, Portugal, in 1995 and 1998, respectively, and the Ph.D. degree in optical networks (computer science) from the University of Algarve, in 2005, and in collaboration with University College London, U.K.

She is currently a Lecturer with the Science and Technology Faculty, University of Algarve. She is also a Founding Member of the Center for Electronics, Optoelectronics and Telecommunications, University of Algarve, a research center supported by the Portuguese Foundation for Science and Technology. She is also the Networks and Systems Group Coordinator. Her research interests include the application of optimization techniques to several network design problems in the optical, wireless, and sensor networks fields, and development of algorithms.



ALVARO BARRADAS received the bachelor's degree in computer science and the Ph.D. degree in electronics engineering and computing from the University of Algarve, Faro, Portugal, in 1995 and 2009, respectively.

His major field of study has been on load balancing for optical networks. He is currently an Auxiliary Professor with the Faculty of Science and Technology, University of Algarve. His previous professional experience includes two years working on system test for computer science and telecommunications consulting companies in Lisbon and Dusseldorf. He is a member of the Center of Electronics Optoelectronics and Telecommunications (CEOT), University of Algarve, a research center supported by the Portuguese Foundation for Science and Technology. His current research interest includes the study and simulation of hybrid networks where QoS provisioning is challenging.