
Nabeel I. Sulieman, Student Member, IEEE and Richard D. Gitlin, Life Fellow, IEEE  

Innovations in Wireless Information Networking Laboratory (iWINLAB)  
Department of Electrical Engineering  
University of South Florida  
Tampa, Florida 33620, USA  

Email: nis@mail.usf.edu, richgitlin@usf.edu

Abstract—Diversity and Network Coding (DC-NC), which is the synergistic combination of Diversity Coding and Network Coding, was recently introduced to provide near instant link failure recovery and efficient transmission in a wide variety of network architectures. In this paper, DC-NC is applied to a multi-hop Wireless Sensor Network (WSN) and Diversity Coding (DC) is applied within each link in a multi-hop WSN to enable extremely high reliability with very rapid recovery from wireless link/node failures and the loss of data stream(s) at each link and also provide efficient transmission, which are very important metrics for WSNs. Furthermore, the reliability and rapid fault recovery can be extended to broadcasting applications, along with increased throughput.

Keywords—Diversity Coding, Near-Instant Fault Recovery, Network Coding, reliability, Wireless Sensor Network

I. INTRODUCTION

A Wireless Sensor Network (WSN) contains two kinds of nodes—a gateway node (central controller) and a sensor node [1]-[2]. In a WSN, one or more gateway nodes might be utilized and several sensor nodes are implemented at different locations [1]-[2]. A sensor node has a sensor that can monitor a specific kind of conditions such as motion, pressure, temperature, etc. [1]-[2]. Ultra-reliability and efficient transmission are required for the WSNs [1]. Data loss and link/node failures are some of the principle contributors that decrease system efficiency and reduce network reliability. In this paper, these two challenges are addressed and overcome for multi-hop WSNs. Reliability was improved in [4] by transmitting the same collected information via multiple sensor nodes to multiple gateway nodes. However, this will increase the redundancy by at least 100%. A more efficient approach is Diversity Coding (DC) [5]-[6], which is an open-loop coding technology that utilizes a forward error control technique over diverse paths and can easily and quickly recover from data loss and wireless link failures as soon as the failure is detected with no need to retransmit messages and perform rerouting. This will improve reliability with near-instant recovery time.

Diversity Coding was utilized to improve C-RAN (Cloud Radio Access Network) network performance by enhancing reliability with rapid link/node failure recovery [7]-[8]. Also, the ability of DC to recover from multiple simultaneous link failures is demonstrated in [8]. Although reliability and very rapid fault recovery are very important metrics in WSNs, efficient transmission is another important factor that affects WSN performance. Network Coding (NC) [9] was shown to improve transmission efficiency.

Ultra-reliable networking with extremely fast fault recovery time and efficient transmission are simultaneously achieved for wireless fronthaul networks by DC-NC coding [10], a synergistic combination of Diversity Coding (DC) and Network Coding (NC). Also, DC-NC coding was applied to downlink Coordinated Multi Point (CoMP) within a C-RAN to enhance the wireless fronthaul reliability, enable ultra-low recovery time, and reduce wireless fronthaul resource consumption [11]. In addition, in [12], DC-NC was shown to provide a robust solution for link/node failures in Network Function Virtualization (NFV) for 5G communications. For simplicity, it was generally assumed that there are no multi-hop connections between the source and the destination. However, in actual networks, multi-hop connections are common.

The contribution of this paper is to extend the application of DC-NC coding to multi-hop WSNs and DC within each wireless link in WSNs and enable ultra-reliable networking, very low recovery time, and efficiently broadcast messages to several nodes.

This paper is organized as follows: Section II describes the multi-hop WSN network topology. Section III presents a summary of Diversity Coding and DC-NC coding. The demonstration of the ability of DC-NC coding and DC to provide extremely high reliability, enable efficient transmission and faster recovery from multiple simultaneous data loss and link/node failures in multi-hop WSNs is presented in Section IV. The paper ends with concluding remarks in Section V.

II. SYSTEM MODEL

A WSN is a group of sensors that distributed at different locations to observe and collect environmental information and communicate the recorded data to a central node, often called a gateway [1]-[3]. Several topologies can be used to build the WSNs networks such as a star or fully mesh topology and multi-hop partial mesh topology [see Figure 1] [1]-[3]. Also, different wireless techniques may be utilized for WSN communications such as Zigbee [1], [3]. Each sensor node can receive and forward the information to the gateway either directly or via other sensor nodes [1]-[2]. The WSN sends the collected data to

978-1-7281-0137-8/19/$31.00 ©2019 IEEE
the outside network via the gateway, which is considered a bridge between the WSN and the external network(s). The WSN might contain a few to several thousands of nodes based on the nature of the application and a battery is usually utilized as the energy source for these nodes [1]-[3]. Since sensor nodes often suffer from limited resources such as memory, energy, and computational capability [1]-[3], the reliability of the WSN might be affected. Therefore, it is required to design efficient techniques such as DC-NC coding that enhance the reliability of the WSN when data loss or link and/or node failures occur.

In this paper, DC-NC coding and DC are applied to maximize the reliability of a WSN with a multi-hop partial mesh topology where sensor nodes are connecting to each other by wireless links as shown in Fig. 1. The collected information is sent from sensor nodes to the gateway node, and commands and control data is transmitted from the gateway to the sensor nodes. In this study, it is focused on an uplink scenario, where sensor nodes send the collected data to the gateway nodes.

![Diagram of Wireless Sensor Network with multi-hop partial mesh topology.](image)

**III. RELATED WORK**

As high reliability and ultra-low recovery time are required for WSNs, any losing of a data stream and/or link/node failure cause rerouting and/or retransmissions. In addition, efficient transmission is always preferred to overcome the limited resources of WSN. In this section, a brief description of the principles of Diversity Coding is presented. Furthermore, the principle of DC-NC coding is explained to demonstrate its potential of improving the performance of WSNs and providing near-instantaneous recovery from data loss and/or link/node failures at the expense of redundant transmission.

**A. Diversity Coding**

Diversity Coding was introduced in [5]-[6] and its principal idea is depicted in Fig. 2. Here, disjoint paths are used to transmit equal rate digital data streams \(x_1,x_2,...,x_N\) to their destination. To clarify the idea of Diversity Coding, a point-to-point topology is considered where each data stream has been transmitted from the same source and end at the same receiver. Coded data \(c_1\), which is the logical XOR summation of the input data streams,

\[
c_1 = x_1 \oplus x_2 \oplus ... \oplus x_N = \bigoplus_{k=1}^{N} x_k
\]

is transmitted on a disjoint path. If data stream \(x_i\) is lost and the loss is detected, the destination can immediately employ other received data streams and \(c_1\) to form mod 2 addition between them as follows

\[
c_1 \oplus (\bigoplus_{k=1, k \neq i}^{N} x_k) = x_i
\]

Since \(x_1,x_2,...,x_{i-1},x_{i+1},...,x_N\) are available at the receiver and \(x_i \oplus x_k = 0\) (logical XOR operation). Therefore, the lost data stream \(x_i\) is recovered, nearly instantaneously, without retransmission, or rerouting. Other, more general, network topologies can utilize Diversity Coding such that the transmitting and/or receiving nodes are not at common nodes [13]. Note that the time to determine the facility loss, will be a lower bound on the recovery latency.

**B. DC-NC Coding**

DC-NC coding is produced from the synergy of Diversity Coding and Network Coding. The basic idea of DC-NC coding is illustrated in Fig. 3(a) [10], the source nodes 1 and 2 broadcast equal rate data streams \(x_1\) and \(x_2\) respectively to the destination nodes 5 and 6. Data streams \(x_1\) and \(x_2\) are collected in node 3, DC-NC encoding node, and encoded to produce the coded data \(c_1\) and \(c_2\) as follows:

![Diagram of DC-NC coding network.](image)
where \( [\beta_{ij}] \) is the parity generator matrix for \( c_1 \) and \( c_2 \). Note that as multiplication and summation are performed in \( GF(2^m) \), they correspond to the AND and XOR operations respectively. The coded data \( c_1 \) and \( c_2 \) then will be transmitted to nodes 4 and 7 respectively. Nodes 5 and 6 will receive \( c_1 \) from node 4. Node 5 simply decodes \( x_1 \) that is received directly from node 1 and \( c_1 \) that is received from node 4 to get \( x_2 \) as follows:

\[
\hat{c}_1 = c_1 + \beta_{11} x_1,
\]

and applying (3) to (5),

\[
\hat{c}_1 = \beta_{11} x_1 + \beta_{21} x_2 + \beta_{11} x_1 = \beta_{21} x_2,
\]

\[
x_2 = \hat{c}_1 / \beta_{21}.
\]

Therefore, broadcasted data streams \( x_1 \) and \( x_2 \) can be received by node 5. Note that the coefficients \( \beta_{ij} \) are fixed and known at all nodes. Similarly, node 6 decodes \( x_2 \) that is received directly from node 2 and \( c_1 \) that is received from node 4 to recover \( x_1 \) as follows:

\[
\hat{c}_1 = c_1 + \beta_{21} x_2,
\]

and applying (3) to (8),

\[
\hat{c}_1 = \beta_{11} x_1 + \beta_{21} x_2 + \beta_{21} x_2 = \beta_{11} x_1,
\]

\[
x_1 = \hat{c}_1 / \beta_{11}.
\]

So that node 6 can also receive \( x_1 \) and \( x_2 \). Note that similar link capacity is assigned for each link in the network, which is equal to the data rate of the broadcasted data [10].

DC-NC throughput gain is enhanced by one-third which is similar to that of Network Coding [10]. However, reliability can be lost with any link failure and the destination nodes 5 and 6 will not get the targeted data streams.

Network reliability can be enhanced by sending \( c_2 \) from node 7 to nodes 5 and 6. The coded data \( c_2 \) will be ignored when there is no link failure. For example, in case the link from node 1 to node 5 fails as shown in Fig. 3(b), node 5 detects the failure then recovers \( x_1 \) and \( x_2 \) by utilizing \( c_1 \) and \( c_2 \) as follows:

Expressing (3) and (4) in a matrix form

\[
\begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} \beta_{11} & \beta_{21} \\ \beta_{12} & \beta_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix},
\]

Using the inverse matrix transform, \( x_1 \) and \( x_2 \) can be easily recovered. The parameters \( \beta_{ij} \)'s should be chosen such that \( \beta_{11}, \beta_{21}, \beta_{12} \) and \( \beta_{22} \) are linearly independent.

Therefore, node 5 obtains \( x_1 \) and \( x_2 \) as follows:

\[
\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \beta_{11} & \beta_{21} \\ \beta_{12} & \beta_{22} \end{bmatrix}^{-1} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}.
\]

Furthermore, if \( c_1 \) lost, node 5 has \( x_1 \) directly and \( c_2 \) then can easily form

\[
\hat{c}_2 = c_2 + \beta_{12} x_1,
\]

and applying (4) to (13)

\[
c_2 = \beta_{12} x_1 + \beta_{22} x_2 + \beta_{11} x_1 = \beta_{22} x_2
\]

\[
x_2 = \hat{c}_2 / \beta_{22}.
\]

Similarly, data streams \( x_1 \) and \( x_2 \) can be recovered at node 6. It is shown in [10] that DC-NC coding has the ability to simultaneously recover from one link failure at each destination node (nodes 5 and 6).

Furthermore, in addition to link failure recovery, DC-NC coding can recover one intermediate node failure such as nodes 4 or node 7 [10]. Moreover, since DC-NC coding uses forward error control technology, ultra-low link failure recovery time can easily be achieved. As a result, there is no need to retransmit messages or perform rerouting. In this way, reliability is enhanced and efficient transmission is achieved by applying DC-NC coding as mathematically proved early in this subsection.

IV. APPLYING DC AND DC-NC CODING TO WSNs

Wireless Sensor Network resource limitations, such as limited energy and transmission bandwidth, in the presence of data loss and/or link/node failures, can cause degradation in throughput and reliability. DC-NC with DC within each link is a promising technology to maximize the reliability of WSNs while enabling low energy cost for data loss and/or link/node failure recovery and providing efficient transmission in broadcast applications.

As shown below, DC-NC coding is applied to a wireless multi-hop sensor network and only DC is applied within each link. Considering that each wireless link has the ability to carry for example (for simplicity) three data streams, two raw data streams and one coded data stream such that the data will be protected at each link. At the same time, DC-NC is applied at the network level as shown in Fig. 4. Here, bi-directional wireless links connect sensor nodes and gateway nodes to each other in a multi-hop partial mesh topology. An uplink multipoint-to-multipoint network topology models the broadcasting of four data streams, \( x_1 \) and \( x_2 \) from sensor node S1 and \( x_3 \) and \( x_4 \) from the sensor node S2 to two gateways G1 and G2. It is assumed that these two gateways are working in active/stand by (ACT/STBY) mode to eliminate single points of failure and to make sure that the required data is collected from sensor nodes i.e. even if one gateway fails for any reason, the collected information will still arrive to the user. In order to broadcast two raw data streams \( x_1 \) and \( x_2 \) from sensor node S1 and another group of two raw data streams \( x_3 \) and \( x_4 \) from sensor node S2 to two gateway nodes G1 and G2, S1 and S2 will form two coded data streams \( c_1 \) and \( c_2 \) respectively to protect the raw data streams as follows:

\[
c_{1} = x_1 + x_2,
\]

\[
c_{2} = x_3 + x_4,
\]

where \( c_{ij} \) represents Diversity Coded data stream within each link. Then S1 will transmit the data streams \( x_1, x_2, \) and \( c_{1} \) to G1 via S4 and S7. Also, S2 will transmit the data streams \( x_3, x_4, \) and \( c_2 \) to G2 via S5 and S8. Note that \( c_{1} \) and \( c_{2} \) are utilized to protect the raw data streams at each hop until arriving to the
destination nodes (G1 and G2 respectively). The sensor node S3, DC-NC encoding node, will receive data streams x₁, x₂, and c₁, from S1 and data streams x₃, x₄, and c₂, from S2 and if there is no error in the raw data streams (by checking via a CRC check), c₁ and c₂ will be ignored then S3 will form new coded data streams as follows:

\[ c_{1N} = \beta_{11}x_1 + \beta_{21}x_2 + \beta_{31}x_3 + \beta_{41}x_4, \quad (18) \]
\[ c_{2N} = \beta_{12}x_1 + \beta_{22}x_2 + \beta_{32}x_3 + \beta_{42}x_4, \quad (19) \]
\[ c_{3N} = \beta_{13}x_1 + \beta_{23}x_2 + \beta_{33}x_3 + \beta_{43}x_4, \quad (20) \]
\[ c_{4N} = \beta_{14}x_1 + \beta_{24}x_2 + \beta_{34}x_3 + \beta_{44}x_4, \quad (21) \]
\[ c_{3i} = c_{1N} + c_{2N}, \quad (22) \]
\[ c_{4i} = c_{3N} + c_{4N}, \quad (23) \]

where \( c_{ki} \) represents the DC-NC coded data stream and \( [\beta_{ij}] \) is the parity generator matrix for \( c_{ki} \). Note that as multiplication and summation are performed in \( GF(2^n) \), they correspond to the AND and XOR operations respectively. To obtain the throughput gains, the coded data streams \( c_{1N}, c_{2N}, \) and \( c_{3i} \) are sent to G1 and G2 via S6 and S9. Note that \( c_{3i} \) is used to protect the DC-NC coded data streams at each hop until arriving to the destination nodes (G1 and G2). At G1, if there is no error in the data streams (by checking via a CRC check), \( c_{1i} \) and \( c_{3i} \) will be ignored and raw data streams \( x_3 \) and \( x_4 \) will be obtained as follows:

\[ \tilde{c}_{1N} = c_{1N} + \beta_{11}x_1 + \beta_{21}x_2, \quad (24) \]
\[ \tilde{c}_{2N} = c_{2N} + \beta_{12}x_1 + \beta_{22}x_2, \quad (25) \]

and applying (18) and (19) to (24) and (25) respectively,

\[ \tilde{c}_{1N} = \beta_{31}x_3 + \beta_{41}x_4, \quad (26) \]
\[ \tilde{c}_{2N} = \beta_{32}x_3 + \beta_{42}x_4, \quad (27) \]

expressing (26) and (27) in a matrix form

\[ \begin{bmatrix} \tilde{c}_{1N} \\ \tilde{c}_{2N} \end{bmatrix} = \begin{bmatrix} \beta_{31} & \beta_{41} \\ \beta_{32} & \beta_{42} \end{bmatrix} \begin{bmatrix} x_3 \\ x_4 \end{bmatrix}, \quad (28) \]

an inverse matrix transformation is used to directly obtain \( x_3 \) and \( x_4 \) as follows:

\[ \begin{bmatrix} x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} \beta_{31} & \beta_{41} \\ \beta_{32} & \beta_{42} \end{bmatrix}^{-1} \begin{bmatrix} \tilde{c}_{1N} \\ \tilde{c}_{2N} \end{bmatrix}. \quad (29) \]

The parameters \( \beta_{ij} \)'s should be chosen such that they are linearly independent. Hence, G1 will get all broadcast data streams. Similarly, raw data streams \( x_1 \) and \( x_2 \) can be obtained at G2 and it will get all broadcast data streams. It is worth noting that the throughput gain of this example network is similar to that of well-known butterfly Network Coding topology [9].

In the presence of a link failure, WSN reliability can be improved by transmitting \( c_{3N}, c_{4N} \) and \( c_{4i} \) from S3 to G1 and G2 via S10 and S11. The coded data streams \( c_{3N}, c_{4N} \) and \( c_{4i} \) will be ignored when there is no link failure. In the presence of a link failure, for example if the link from S7 to G1 fails, G1 detects the failure and then recovers \( x_1, x_2, x_3 \), and \( x_4 \) by utilizing \( c_{1N}, c_{2N}, c_{3N} \) and \( c_{4N} \) (assuming that the DC-NC coded data streams arrive at G1 with no error (checking via CRC) and \( c_{3i} \) and \( c_{4i} \) are ignored) as follows:

expressing (18)-(21) in a matrix form

\[ \begin{bmatrix} c_{1N} \\ c_{2N} \\ c_{3N} \\ c_{4N} \end{bmatrix} = \begin{bmatrix} \beta_{11} & \beta_{21} & \beta_{31} & \beta_{41} \\ \beta_{12} & \beta_{22} & \beta_{32} & \beta_{42} \\ \beta_{13} & \beta_{23} & \beta_{33} & \beta_{43} \\ \beta_{14} & \beta_{24} & \beta_{34} & \beta_{44} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}, \quad (30) \]

It is mentioned earlier that \( \beta_{ij} \) are known and fixed at all nodes and they should be chosen such that they are linearly independent. Hence, the required data streams can be easily recovered at G1 via inverse matrix transformation as follows:

\[ \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} \beta_{11} & \beta_{12} & \beta_{13} & \beta_{14} \\ \beta_{21} & \beta_{22} & \beta_{23} & \beta_{24} \\ \beta_{31} & \beta_{32} & \beta_{33} & \beta_{34} \\ \beta_{41} & \beta_{42} & \beta_{43} & \beta_{44} \end{bmatrix}^{-1} \begin{bmatrix} c_{1N} \\ c_{2N} \\ c_{3N} \\ c_{4N} \end{bmatrix}. \quad (31) \]

Also, if the link between S9 and G1 fails, \( c_{1N} \) and \( c_{2N} \) are lost, G1 has \( x_1, x_2, c_{3N}, \) and \( c_{4N} \) then can quickly and easily recover \( x_3 \) and \( x_4 \) as shown below:

\[ \tilde{c}_{3N} = c_{3N} + \beta_{13}x_1 + \beta_{23}x_2, \quad (32) \]
\[ \tilde{c}_{4N} = c_{4N} + \beta_{14}x_1 + \beta_{24}x_2, \quad (33) \]
\[ \tilde{c}_{3N} = \beta_{33}x_3 + \beta_{43}x_4, \quad (34) \]
In the first hop, the data streams follow: the network, the reliability of path (S1-S4-S7-G1) is studied as (S1-S7), (S2-S6), (S3-S8), and (S4-S10), nodes S7, S6, S8, and S10 will do the link between the following pair of nodes (S4 - S7), (S3 - S6), (S2 - S5), and (S1 - S4), nodes S4 and S5 will form the XOR operation between the corrected received data streams.

Similarly, G2 can obtain the required data streams in the presence of any link failure.

In case of losing a data stream or a high level of errors (by a CRC check), the XOR operation between the corrected received data streams achieves (34) and (35) in a matrix form

\[ \begin{bmatrix} \hat{e}_{3W} \\ \hat{e}_{4W} \end{bmatrix} = \begin{bmatrix} \beta_{33} & \beta_{34} \\ \beta_{43} & \beta_{44} \end{bmatrix} \begin{bmatrix} x_3 \\ x_4 \end{bmatrix}, \]

the data streams \( x_3 \) and \( x_4 \) are easily recovered using the inverse matrix transformation as follows:

\[ \begin{bmatrix} x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} \beta_{33} & \beta_{34} \\ \beta_{43} & \beta_{44} \end{bmatrix}^{-1} \begin{bmatrix} \hat{e}_{3W} \\ \hat{e}_{4W} \end{bmatrix}. \]

Similarly, G2 can obtain the required data streams in the presence of any link failure.

Note that since there are three wireless links at each path in this example network, the DC-NC coding has the ability to recover the required data streams even if all links at the path fail simultaneously because a path is considered a failure as long as there is a link failure regardless of the number of failed links.

Furthermore, not only link failures can be recovered by DC-NC coding in this example multi-hop WSN network. Intermediate node failures such as S4 and/or S7, S5 and/or S8, or S6 and/or S9 can be tolerated since this corresponds to one path failure at each destination node. This means two intermediate nodes at each path in this example network can be tolerated via DC-NC coding for the same reason that explained above. Also, when failures occur at S10 and/or S11, \( c_{3W} \) and \( c_{4W} \) will be lost, that is protection of the network, but, if these are the only failures, successful data communication can still be achieved.

In case of losing a data stream or a high level of errors is determined (via a CRC check) within each wireless link between S1 and S4 and between S2 and S5, nodes S4 and S5 will form the XOR operation between the corrected received data streams and then transmit the result, which is the lost data stream with other received two data streams to nodes S7 and S8 respectively. Similarly, in case of losing a data stream within each wireless link between the following pair of nodes (S4 - S7), (S3 - S6), (S5 - S8), and (S3 - S10), nodes S7, S6, S8, and S10 will do the same process that explained above. Furthermore, the same procedure will be applied in case of any link in a data stream between the following pair of nodes (S7 - G1), (S6 - S9), (S9 - G1), (S9 - G2), (S8 - G2), (S10 - S11), (S11 - G1), and (S11 - G2).

In order to show how losing data streams are recovered in the network, the reliability of path (S1-S4-S7-G1) is studied as follows:

In the first hop, the data streams \( x_1, x_2, \) and \( c_{1t} \) are sent to S4. In case of losing a data stream or a high level of errors (by a CRC check), e.g. losing the data stream \( x_1, \) S4 forms the following:

\[ x_2 \oplus c_{1t} = x_1, \]

(38)

In the second hop, the data streams \( x_2, c_{1t}, \) and \( x_1 \) are transmitted to S7. In case of losing a data stream, e.g. losing the data stream \( x_2, \) S7 forms the following:

\[ c_{1t} \oplus x_1 = x_2, \]

(39)

In the third (last, in this example network) hop, the data streams \( c_{1t}, x_1, \) and \( x_2 \) are sent to G1. In case of a data stream in error, e.g. the data stream \( c_{1t} \) is in error, there will be no need to recover \( c_{1t} \) since G1 received \( x_1 \) and \( x_2 \) with no error. Assuming that G1 receives \( c_{1W} \) and \( c_{2W} \) with no errors it will have \( x_1, x_2, c_{1W}, \) and \( c_{2W} \), so, G1 can easily obtain \( x_2 \) and \( x_4 \) as shown in (24)-(29). Similarly, other failure scenarios can be handled in a similar manner to recover the data.

Here, data streams will be protected at each hop via DC as shown in the path (S1 – S4 – S7 – G1). In this way, each path is protected from the loss of one data stream such that in this example network each path can tolerate losing three data streams (one for each hop). In addition, for this example network, the loss of ten simultaneous data streams can be tolerated for the entire network (one for each hop). This means the total number of lost data streams that can be tolerated equals the number of hops in the entire WSN. Moreover, by applying DC-NC, one path failure (i.e. three wireless links and two intermediate nodes for each path in this example network) can be tolerated for each destination node. Furthermore, the throughput gain of this example network is similar to that of well-known Network Coding butterfly topology.

There is no need to simulate the results in this paper because the link failure and/or data loss is considered independently of the failure mode and it is mathematically proven how the multi-hop WSN network can be enhanced and protected by DC at each hop and DC-NC for the entire network. Of course, the recovery time is lower bounded by the time to detect a failure.

V. CONCLUSIONS

This paper presented the application of the recently introduced coding technique, DC-NC, that synergistically combines Diversity Coding and Network Coding, to improve the performance of multi-hop WSNs. In addition, Diversity Coding is applied to each link in the WSN to protect the data streams from being lost or being in error. It has been demonstrated that DC-NC can simultaneously recover from multiple link/node failures at each path nearly-instantaneously. Furthermore, using DC, one lost data stream can be tolerated at each hop i.e. the total number of tolerated data streams equals the number of hops in the entire WSN. Applying DC-NC network coding and DC in the links improves the WSNs reliability and enables near-instant recovery from multiple wireless link/node failures and multiple lost data streams (one for each hop), while simultaneously improving the throughput in the WSN network.

ACKNOWLEDGMENT

Nabeel I. Sulaiman is supported by the Higher Committee for Education Development in Iraq (HCED-IRAQ).

REFERENCES


