

Article

An Artificial Bee Colony-Based COPE Framework for Wireless Sensor Network

Amit Singh and Aitha Nagaraju *

Department of Computer Science, Central University of Rajasthan, Ajmer-305817, India;
amit2013_csphd@curaj.ac.in

* Correspondence: nagaraju@curaj.ac.in; Tel.: +91-82907-27943

Academic Editor: Yevgeniya Kovalchuk

Received: 15 March 2016; Accepted: 27 April 2016; Published: 6 May 2016

Abstract: In wireless communication, network coding is one of the intelligent approaches to process the packets before transmitting for efficient information exchange. The goal of this work is to enhance throughput by using the intelligent technique, which may give comparatively better optimization. This paper introduces a biologically-inspired coding approach called Artificial Bee Colony Network Coding (ABC-NC), a modification in the COPE framework. The existing COPE and its variant are probabilistic approaches, which may not give good results in all of the real-time scenarios. Therefore, it needs some intelligent technique to find better packet combinations at intermediate nodes before forwarding to optimize the energy and maximize the throughput in wireless networks. This paper proposes ABC-NC over the existing COPE framework for the wireless environment.

Keywords: network coding; artificial bee colony-based COPE; wireless sensor network

1. Introduction

Innovative data storage and transmission is a challenging issue for researchers to improve the efficiency and reduce the communication cost in the wireless environment. Cloud applications and Internet usage are increasing day by day due to the rapidly increasing on demand services raised by Internet users. Therefore, data transmission is a crucial parameter of the network in terms of energy efficiency and throughput. Researchers introduced network coding as a concept of intelligent data processing before its transmission to reduce network traffic and make effective utilization of energy during packet transmission.

In the wireless environment, throughput improvement and reduction in energy consumption are the main challenging tasks. Besides this, scalability and dense deployment are further issues for researchers [1]. Network coding provides a way of intelligent data processing before their transmission to reduce network traffic and energy consumption. Earlier network coding was proposed to overcome the bottleneck problem in a wired network, but the diversity of links and the broadcast nature of the wireless medium makes it more popular in a wireless network, such as a sensor and mesh network. The first practical implementation of network coding came in the form of COPE [2] in 2006.

As mentioned by Katti *et al.*, COPE is a new packet-forwarding architecture to raise wireless network throughput by inserting a COPE shim between the IP and MAC layer. COPE provides much less coding gain of approximately 2%–3% in a TCP flow network due to collision-related losses, while an enhancement of 15%–70% in a UDP network, depending on uplink traffic. Liu *et al.* [3] proved that network coding could reduce the number of transmissions in a wireless network. Philipp *et al.* claimed energy savings up to 50% in [4] for a randomly-deployed and scalable sensor network using opportunistic network coding.

In this paper, the implementation is focused on improving the coding gain by applying an intelligent mechanism, which is an artificial bee colony approach. The proposed model would be

applicable to a wireless sensor network while transmitting the packets. The wireless sensor network [5–7] is a collection of tiny devices, distributed in an area for some specific and collaborative purpose. A survey on an intelligent sensor network with its applications is presented in [8]. Another candidate network (not implemented in this paper) for the proposed encoding technique is a wireless mesh network [9–11] in which each node can directly communicate with any other peer node. The sensor nodes may communicate with the other nodes, as well as sink nodes or base stations depending on their range of communication. A vast range of real applications, like battlefield surveillance, animal and environmental monitoring, agriculture improvement, traffic and health monitoring, forest fire monitoring, *etc.*, make sensor networks popular enough in the research community. As an energy consumption concern, it is important for WSNs to limit the number of transmissions/receptions of data and control packets. Network coding is useful for throughput enhancement, robustness and load balancing.

In the literature, several issues of WSNs, like localization, node coverage and data aggregation, are applied in various routing schemes, which are enhanced by nature-inspired artificial intelligence techniques. A compact artificial bee colony optimization for topology control was proposed in [12]. This paper proposed a nature-inspired approach for encoding packets on the COPE framework. The coding technique proposed in COPE was based on a reception report and the probability of having packets at neighbors. Researchers have confirmed that several systems available in nature may better deal with many complex problems in a very efficient manner. The proposed artificial bee colony network coding technique is able to find better packet combinations in a heuristic manner as a natural bee does and gives improved coding gain, as shown in the results. As a population size concern, random or heuristic techniques provide better optimization than the traditional deterministic approaches.

The proposed encoding algorithm enhances the performance of an opportunistic coding phase of the COPE framework using artificial honey bee behavior. The algorithm executes in three steps: employee bee phase, onlooker bee phase and scout bee phase. In this approach, a random wireless sensor network has been considered for the implementation purpose to check the performance of the proposed approach. The proposed research work also enhances the practical implementation of COPE.

The rest of the paper is organized as follows. Section 2 presents some network coding techniques proposed for unicast traffic flow. The proposed approach with an overview of nature-inspired techniques, especially artificial bee colony, is presented in Section 3. Section 4 is dedicated to the implementation of the proposed work and the comparative results' discussion, respectively. Finally, Section 5 is the conclusion and future scope.

2. Related Work

In order to optimize the coding technique, several theoretical concepts have been proposed by in the literature. Inter-session-based network coding COPE was proposed in [2]. In this implementation, Sachin *et al.* distributed a random number of mobile nodes in the sensor field of a fixed region. The nodes in the sensor field perform the network coding concept in a real environment. The COPE basically performs in three steps, which are opportunistic listening, opportunistic coding and learning the neighbor's state.

COPE is an XOR-based local network coding technique. The existing COPE may not work in all real-time applications; the following are a few drawbacks of COPE, which we can overcome in our proposed work:

1. Prolong snooping of the communication medium may quickly deplete node's battery.
2. Periodic broadcast of reception reports and the diversity of links may increase battery consumption and network traffic.
3. COPE is a probabilistic approach. Therefore, the time complexity of the coding algorithm in COPE is less, but it may not be the best encoding solution each time.

Kaikai *et al.* [13] replaced the FIFO queue of COPE with a priority queue. The authors maintain a dedicated FIFO queue called the control queue for special packets and an FIFO queue for each active flow passing through the current node. The author has changed the scheduling algorithm, as well. The simulated result claims an improvement of 16% in coding gain.

Opportunistic coding depends on two things: first is the combination of incoming and outgoing links at the node and second is whether the packet received at the node is coded or native. In contrast to COPE, the cross-layer-based network coding-aware routing technique is proposed in [14]. The coding-aware routing approach is based on the linear programming technique. A new inter-session network coding technique, COPR (Coding with Opportunistic Reception) [15], is proposed for multiple unicast flows, and it achieved 25% power savings. In a wired network, COPR decomposes multiple unicast flows into a superposition of multicast and unicast flows with coding in each flow. However, this session-based scheduling method performs coding by searching only the head of the queues, similar to COPE. A distributive approach to network coding-aware queuing management (NCAQM) for TCP flows is discussed in [16]. The NCAQM scheme uses network coding and congestion information for coded packet buffering and dropping at the relay nodes. The author claims that NCAQM better exploits the coding opportunity in TCP flows over wireless network coding, which was not possible in COPE due to the bursty behavior of TCP flows. The simulated result (TCP + NCAQM) shows an improvement of 18%–35% depending on the network topology. CLONE is proposed in [17] for a wireless lossy medium and claimed throughput gain of around 57% with an approximate overhead of 9%–10%. CLONE uses redundancy to increase the probability of delivering packets at the intermediate nodes. It is practically in-feasible due to its complex computation.

In [18], the author considers the flow instead of individual packets and compares simulated results with packet-level network coding. In contrast with the traditional tree-based approach, SenseCode is proposed in [19]. The redundancy and network coding-based SenseCode achieves high reliability up to 90% in highly dynamic network conditions, whereas redundancy alone or opportunistic routing does not. Each node buffers all self-generated messages and the packets it has received from its descendant nodes during the transmission cycle in a queue. Overheard packets may be stored in a separate queue. Whenever a node has a message to send, it creates an un-codable marked packet and transmits to ancestor nodes. If a node receives an un-codable marked packet, it has to relay that packet; otherwise, it has to send the linear combination of the received packet with the queue's packets. Finally, some of coded/uncoded packets are received at the sink node, which provide reliability in a lossy wireless medium. Kasireddy and Wang [20] analyzed the network coding performance on the energy consumption of a grid wireless sensor network.

Intra-session network coding is a more obvious way to deal with the lossy environment problems. It gives equal preference to all packets generated from the same source. Random linear encoding-based MORE [21] was proposed for opportunistic routing. In the opportunistic routing technique [22], each node overhears the medium transmission, and nodes closer to the destination may participate in the relaying of a packet in contrast with traditional routing methods, where a predetermined path exists between the source and destination. In MORE, source node fragments the source file into segments named batches. The batch is a collection of uncoded packets called native packets. A random linear combination of native packets of the batch is created and broadcast in the wireless medium. The destination decodes the encoded packets only if it receives a sufficient number of packets. The optimal rate control for avoiding network congestion and the multi-path routing technique for shortest path exploration-based OMNC were proposed in [23]. This reduces bandwidth usage and avoids congestion in a distributive manner by encoding.

OMNC has less complexity for network coding implementation, but it decreases protocol performance by wasting wireless bandwidth. This problem is addressed, and a new approach, CodeOR, is presented in [24]. CodeOR exchanges the window size (W) with the destination node for more than one simultaneous segment communication. If the source node receives at least one end-to-end acknowledgment from the destination node, it starts transmitting the packets of the next

segment. If an intermediate node receives an acknowledgment from descendant nodes, it starts transmitting packets of the next segment. The problem of loss rate estimation and correlation between the links is emphasized in CCACK [25,26]. Cumulative coded acknowledgment is used for feedback messages. The symbol-level MIXIT protocol is proposed in [27]. Rather than sending the correct packet every time, MIXIT even transmits each correct bit. The MIXIT approach is useful for WSNs in which there are huge traffic flows from sensors to the sink node and for VANETs for content distribution.

3. Proposed Work

In the proposed work, we have made an attempt at an artificial intelligence approach to the existing COPE, which performs the encoding of packets before transmitting them to the next node. The encoding of a packet is nothing but processing before sending it to the next hop. In COPE, the processing consists of three phases, and the proposed work optimizes these steps to overcome the problems in the basic architecture. In the literature, several issues of WSNs, like localization, node coverage, data aggregation and resource optimization in routing schemes, are enhanced by nature-inspired artificial intelligence techniques. The proposed work is an attempt to incorporate the well-known nature-inspired Artificial Bee Colony (ABC) technique to improve coding gain in wireless networks. This section will explain the general features of nature-inspired techniques and the proposed algorithm.

Recently, the major issues in WSNs have been solved more efficiently by meta-heuristic approaches, and swarm intelligence provides the way of doing so. The basic idea behind swarm intelligence is the collective and collaborative behavior of social animals, like ants, birds, fish, bees, bacteria, etc. These social animals solve complex tasks in a distributed manner, which would not be possible for a centralized approach. The intelligent behavior of these insects is analyzed by the research community and it was attempted to develop algorithms for nonlinear, non-convex or combinatorial optimization problems for science and engineering applications. The two important and significant behaviors of swarm intelligence are [28,29] explained in the literature:

3.1. Self-Organization

This is an important feature of social animals that results in global-level response due to interaction with lower level individuals without any central coordinating unit. Self-organization is based on:

1. *Positive feedback* is the information generated by the system and again fed back to the system to create a better structure. It provides diversity and acceleration to the new stable state.
2. *Negative feedback* helps to stabilize the system and compensates the effect of positive feedback.
3. *Fluctuation* helps in finding a new solution by random change and avoids stagnation.
4. *Multiple interactions* provide group intelligent behavior by learning from individuals of the society.

3.2. Division of Labor

Social animals are intelligent enough to distribute small segments of a whole task into specialized individuals and to perform simultaneously for a better and efficient outcome.

Swarms are the collection of interacting individuals that may work together to accomplish a goal. The classical examples are honey bees around their hives, ant colonies, flocks of birds, herding of animals, particle swarm intelligence (PSO), the immune system in a swarm of cells, behavior of termites, etc.

Dervis Karaboga presented an artificial bee colony algorithm [30] inspired by the foraging behavior of honey bees in 2005. The author specifies that the bees in the hive are divided into three categories:

1. *Employee bees* are currently exploiting or employed at some food source. They share information of the food source with other members of the hive with a certain probability.

2. *Onlooker bees* are waiting in the hive for new food sources. They evaluate the information communicated by employed bees to find the best one among those.
3. *Scout bees* are unemployed bees and randomly explore the new food source near the hive.

The routine activity of honey bees are artificially presented by Karaboga [30] in the following ABC algorithm:

Algorithm 1 ABC algorithm.

- 1: **procedure** BEST-FOOD-SOURCE
 - 2: Initial food sources generated by scout bees near hive area.
 - 3: **while** (*termination criteria*) **do**
 - 4: Employee bees visit to explore food sources and determine their nectar amounts.
 - 5: Employee bees compute the probability value of the food sources with which food sources are preferred by onlooker bees.
 - 6: Stop exhausted food exploitation by the bees.
 - 7: Scout bees randomly start exploring new food sources.
 - 8: Memorize the best food source so far.
 - 9: **end while**
 - 10: **end procedure**
-

The main concept of the ABC technique is to converge at the best global solution by updating position efficiently. The updating position (Figure 1) performed using the expression:

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \quad (1)$$

where:

- v_{ij} = updated position value
- x_{ij} = current position value
- x_{kj} = neighboring position value
- ϕ_{ij} = random value

Figure 1 is the general pictorial representation of the position updating process of a bit in the packet.

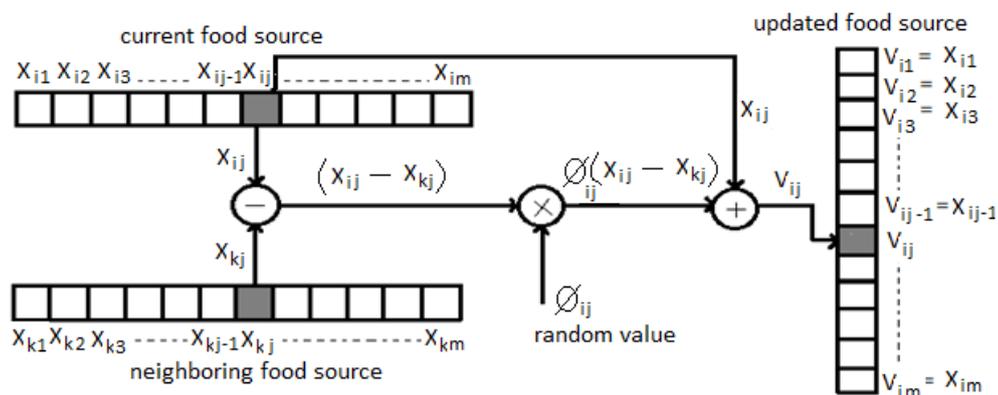


Figure 1. Food source updating process in Artificial Bee Colony (ABC).

As an example, suppose 0111000101110110 is a population member (size 16 bit), representing X_i (the i -th member of the population set) and updating through population member 1110001101010110 representing X_k (the k -th member of the population set). The randomly-selected position updating is 11th, which represents the j -th value in the above equation. ϕ is the random value between -1.0 and $+1.0$ and taken as 0.5596 for this particular example. Figure 2 shows the position updating as follows.

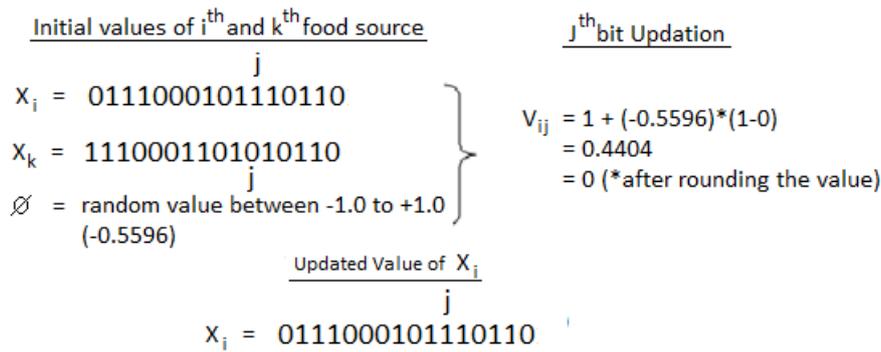


Figure 2. Pictorial representation of the position updating process.

Employee and onlooker bees are responsible for the updating process to approach the global solution of the problem. Exploration of the new food source is carried out by scout bees. The computation of the nectar amount corresponds to the quality of the solution for a particular problem. A similar technique is proposed in the next section to find a better encoding solution of the buffered native packet at intermediate nodes. A conceptual paper on the artificial bee colony-based COPE framework is presented in [31]. The author mentioned that the theoretical concept of the nature-inspired optimization technique in existing probabilistic COPE may enhance the network throughput and efficiency. COPE inserts a coding shim layer between IP and MAC, as shown in Figure 3, Stage (a). To improve network throughput, COPE distributes the work in three stages, as shown in Figure 3, Stage (b). In the proposed work, the intelligent approach is incorporated for better packet combination in opportunistic coding, as shown in Figure 3, Stage (c).

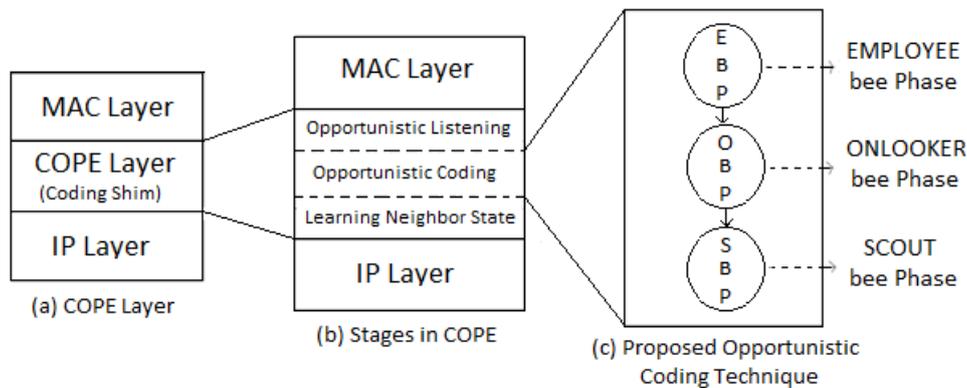


Figure 3. Modifications to the COPE-opportunistic coding stage.

COPE encodes packets based on the head packet of the output queue and virtual queues. In contrast, our approach finds the random partial packet encoding through an intelligent technique called artificial bee colony and improves the encoded packet size. The proposed approach is divided into three phases: Employee Bee Phase (EBP), Onlooker Bee Phase (OBP) and Scout Bee Phase (SBP). All of these phases runs repeatedly in a sequential manner, as explained in the opportunistic coding section of the proposed algorithm.

3.3. Opportunistic Listening

This phase gives the best packet combination to ensure the overall high throughput of wireless networks. The existing COPE follows a probabilistic approach along with the available reception report information received by the neighbors for packet combination. However, in our approach, we have

made an attempt to find the packet combination by artificial intelligence honey bees. As discussed earlier, opportunistic listening is snooping the wireless medium to overhear packets by the nodes. In COPE, overheard packets are buffered by each node for 0.5 s for the future coding opportunity and decoding purposes. The proposed intelligent optimization will find better packet encoding and take more processing time in comparison with probabilistic COPE. The limiting time for overheard packets in the buffer is assumed as 1.0 s for delay-tolerant applications.

3.4. Opportunistic Coding

Packet encoding and decoding are the two important phases of the COPE technique. Whatever is encoded at the source node should be decodable at the destination node; this is the basic principle of network coding. Decoding is only possible if the destination node has all of the packets except the interested one. The next subsections describe our approach of packet coding and decoding.

3.4.1. Packet Encoding

This proposed work illustrates an artificial-based encoding process based on nature-inspired honey bee behavior. The proposed algorithm is an iterative-based approach, which is a meta-heuristic in nature. The designed algorithm is basically composed of three sub-algorithms: employee bee phase, onlooker bee phase and scout bee phase. All of these phases are artificially implemented on the honey bee life cycle. The working principle of the algorithm is shown in Figure 4. The Partially-Encoded Packet (PEP) is an initial food source generated by scout bees for the EMPLOYEE bee phase. The partially-encoded packet improves itself in iterations, and if not able to improve up to a maximum number of trials, then the SCOUT bee phase abandons that PEP and replaces it with another randomly-generated PEP.

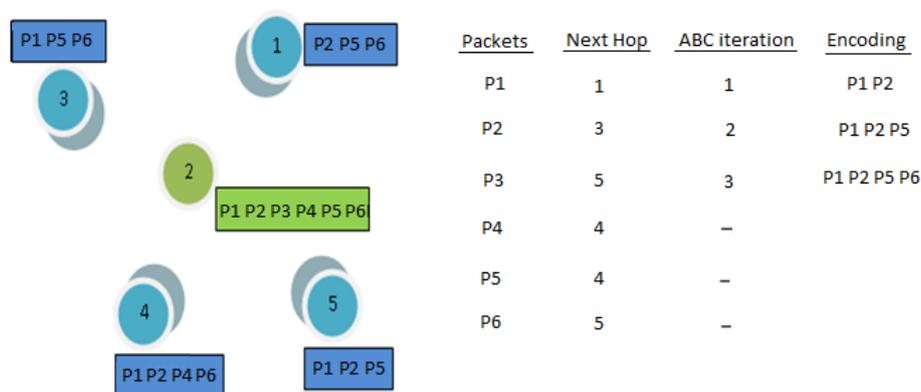


Figure 4. Diagrammatic representation of ABC-based network coding.

The proposed encoding algorithm is given as:

Algorithm 2 ABC-COPE.

- 1: **procedure** ABC-BASED-COPE
 - 2: Scout Bees initialize random food sources/Partially-Encoded Packets (PEP)
 - 3: Compute objective and fitness value of PEPs
 - 4: **while** (*termination criteria*) **do**
 - 5: EMPLOYEE bee phase
 - 6: ONLOOKER bee phase
 - 7: SCOUT bee phase
 - 8: Memorize the best PEP found so far
 - 9: **end while**
 - 10: **end procedure**
-

The initial population is randomly generated by scout bees based on the packets at the node's output queue. In the proposed implementation, the initial partially-encoded packet size varies from 2–11, *i.e.*, the encoded packet contains 2–11 native packets (uncoded packets). The range may vary depending on the processing and memory capacity. Based on the initial encoded packets, all three phases of the algorithm have repeatedly found the best possible final encoded packet of maximum objective value. The objective function is the number of neighboring nodes able to decode the current encoded packet correctly. We have chosen a large number of iterations ($n = 1000$) for better packet combination. The existing COPE is probabilistic in nature, whereas our proposed one would give a better combination after a certain number of iterations. Our proposed algorithm accuracy depends on the number of iterations specified by the algorithm; the proposed algorithm performance will be evaluated by varying the number of iterations in the given algorithm. The proposed algorithm's practical implementation depends on the availability of existing physical resources.

The evaluated objective value computation from the Employee Bee Phase (EBP) sub-algorithm executes to improve population members.

Algorithm 3 EMPLOYEE bee phase.

- 1: **procedure** EMPLOYEE-BEE-PHASE
- 2: **for** each population food source/PEP **do**
- 3: find a random position in the PEP for mutation process
- 4: randomly select neighboring PEP from the population set
- 5: mutate the current PEP with neighboring as follows:

$$new_{PEP(ij)} = current_{PEP(ij)} + rand * (current_{PEP(ij)} - neighbor_{PEP(positionj)})$$

- 6: calculate the objective value of the current PEP as:

$$objVal(i) = \text{number of benefited neighbors in a single transmission cycle}$$

- 7: compute the fitness value of $new_{PEP(ij)}$ as:
- 8:

$$fit(i) = \begin{cases} 1/ObjVal(i) + 1, & \text{if } ObjVal(i) \geq 0 \\ 1 + abs(ObjVal(i)), & \text{otherwise} \end{cases}$$

- 9: perform greedy selection between $new_{PEP(ij)}$ and $current_{PEP(ij)}$
- 10: calculate the probability for ONLOOKER bee phase as:

$$prob(i) = 0.9 * fit(i) / max(fit) + 0.1$$

- 11: **end for**
 - 12: **end procedure**
-

The purpose of this phase is to improve the PEP as much as possible. If any population set as the member/PEP is not able to improve itself in 100 trials (assumption), then scout bees will abandon that PEP and replace that population member with another randomly-generated PEP by the scout bee phase. For the updating process of PEP, the algorithm mutates the current food source with random bits of randomly-selected PEP using Equation (2).

$$new_{PEP(ij)} = current_{PEP(ij)} + rand * (current_{PEP(ij)} - neighbor_{PEP(positionj)}) \quad (2)$$

Here, $new_{PEP(ij)}$ represents the modified PEP using the value of under-considered PEP, *i.e.*, $current_{PEP(ij)}$, and a random bit of another randomly-selected member of the population set

$neighbor_{PEP(position_j)}$. After updating PEP, the algorithm again evaluates the objective and fitness value of the updated PEP using Equations (3) and (4).

$$objVal(i) = \text{number of benefited neighbors in a single transmission cycle} \quad (3)$$

$$fit(i) = \begin{cases} 1/ObjVal(i) + 1, & \text{if } ObjVal(i) \geq 0 \\ 1 + abs(ObjVal(i)), & \text{otherwise} \end{cases} \quad (4)$$

Fitness ($fit(i)$) depends on the objective value computed in Equation (3). The objective function ($objVal(i)$) is defined as a function of the neighbor count benefited in a single transmission cycle. Here, we have introduced only one parameter for optimization, but we may also include battery power remaining with the node, packet latency and many more in the future. If we consider other parameters, as well, then the objective value function of Equation (3) may look like:

$$objVal(i) = fun(x, y, z, \dots)$$

where x denotes the number of benefited neighboring nodes in a single transmission cycle. y represents the remaining battery power with the node. z may be considered as the packet latency, and so on.

Greedy choice is executed for the best PEP selection between current and modified PEP. The last step of this phase is to calculate the probability by using the PEP fitness value as in Equation (5).

$$prob(i) = 0.9 \times fit(i) / max(fit) + 0.1 \quad (5)$$

The above probability value range is modified as:

$$0.1 \leq prob(i) \leq 1.0$$

This probability is helpful for the onlooker bee phase in the selection process just like in a real honey bee hive; onlookers select that food source/PEP for which the nectar amount (information shared by employee bees) is higher.

The onlooker bee phase only updates those PEPs whose probability value is greater than α . α are the random value taken between zero and one. The selected PEP for which α -criteria is fulfilled is considered for updating the same as the employee bee phase using the same Equations (2)–(4), as shown in the algorithm:

Algorithm 4 ONLOOKER bee phase.

```

1: procedure ONLOOKER–BEE–PHASE
2:   for each population food source/PEP do
3:     if prob >  $\alpha$ 
4:     find a random position in the PEP for mutation process
5:     randomly select neighboring PEP from the population set
6:     mutate the current PEP with neighboring as follows:

```

$$new_{PEP(ij)} = current_{PEP(ij)} + rand * (current_{PEP(ij)} - neighbor_{PEP(positionj)})$$

```

7:   calculate the objective value of the current PEP as:

```

$$objVal(i) = \text{number of benefited neighbors in a single transmission cycle}$$

```

8:   compute the fitness value of  $new_{PEP(ij)}$  as:
9:

```

$$fit(i) = \begin{cases} 1/ObjVal(i) + 1, & \text{if } ObjVal(i) \geq 0 \\ 1 + abs(ObjVal(i)), & \text{otherwise} \end{cases}$$

```

10:  perform greedy selection between  $new_{PEP(ij)}$  and  $current_{PEP(ij)}$ 
11:  end
12: end for
13: end procedure

```

The scout bee phase explores new food sources/PEPs, as well as abandons those PEPs that are not able to improve itself in the iterations. The corresponding algorithm is given as follows:

Algorithm 5 SCOUT bee phase.

```

1: procedure SCOUT–BEE–PHASE
2:   for each population food source/PEP do
3:     if PEPcounter  $\geq$   $max\_trail$ 
4:     then
5:     replace PEP with another randomly-generated PEP
6:     update the objective and fitness value of PEP
7:     end
8:   end for
9: end procedure

```

Here, max_trail is taken as 100. If any partially-encoded packet is not able to update itself in 100 trails, it is replaced by another random PEP. After this replacement, the objective and fitness values are updated.

The last step of the main algorithm is to memorize the best PEP so far. The algorithm returns the best PEP heuristically. All of these three phases will be repeated 1000 times for a better opportunity for the packet combination.

3.4.2. Pocket Decoding

The packet decoding process is the same as the COPE framework. A node can decode the desired packet only if it has $n - 1$ packets out of n encoded packets.

3.5. Learning Neighbor State

Each node transmits the reception report to the neighboring nodes to circulate information about the packet it has stored in the buffer pool. In COPE, each node, along with finding the probability of

having any packet at the neighboring node, also receives a reception report from neighboring nodes. However, sometimes, the information collected by the reception report may be lost. The computed probability may also be erroneous. Due to insufficient and erroneous information, the coded packet may not be decodable at the next hop, which may cause inefficient encoding. The proposed ABC-COPE finds a better coding combination using the iteration of three phases.

Similar to the COPE protocol, the ABC-COPE header is attached along with the MAC and IP header.

The ABC-COPE header is divided into three parts as shown in Figure 5:

- i. ENCODED_NUM contains the information regarding the packets XORed together
- ii. REPORT_NUM carries the reception report information
- iii. ACK_NUM contains the cumulative acknowledgment information

Bit Seq is the combination of bits, representing the cumulative feedback and reception report information. The structure of the reception report and acknowledgment are $(SRC_IP, LAST_PKT, Bit\ Map)$ and $(NEIGHBOR, LAST_ACK, Bit\ Map)$, respectively. The reception report bit sequence (134.10.23.19, 50, 10000001) represents that the last packet heard by the current node is 50th from source 134.10.23.19, and also, it has heard the 42nd and 49th packet from the same source. The acknowledgment bit sequence (134.10.23.17, 50, 00111111) represents that the destination received the 50th packet, as well as a sequence of the 44th–49th packets from neighbor 134.10.23.17.

Therefore, each node needs to maintain the following data structures:

1. An output FIFO queue is used to store packets that need to relay to the next-hop
2. The packet pool stores all packets heard in the last T unit of time
3. Two virtual queues to store per neighbor information shared/computed by the reception report/deterministic probability

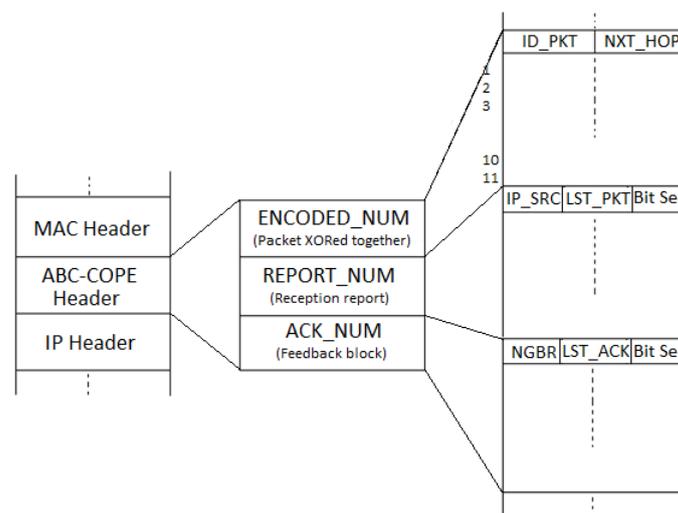


Figure 5. Description of the ABC-COPE header.

Acknowledgment and re-transmissions: each sender expects acknowledgment from the next hop, and if not acknowledged within a certain time interval, the sender needs to re-transmit the native packet again with the next potentially-encoded packet.

4. Results and Discussion

This section shows the results and comparison of our nature-inspired encoding algorithm with COPE Framework. The implementation is done in MATLAB R2008a Version 7.6.0.324. The proposed

algorithm was executed on a WSN of Nodes 10–40. Nodes are randomly distributed over an area of $100 \times 100 \text{ unit}^2$. In the result, the coding gains of COPE and ABC-COPE are evaluated for medium delivering probabilities of 100%, 80%, 70% and 60%. The delivering probability represents the information about storing packets at the neighboring node's buffer, *i.e.*, successful delivery of reception reports. The proposed algorithm is a heuristic iterative approach, whereas COPE is probabilistic. This paper shows the above differences with respect to the network coding gain. Some examples (Figure 6) of randomly-distributed WSNs are shown below. The dots represent the sensor nodes, and dashed lines shows wireless communication links among the sensors. The following WSN shows the number of nodes varying with the reliability factor. The reliability factor is increasing 60%, 70%, 80% and 100%. The number of nodes in the network is varying from 10–40 nodes.

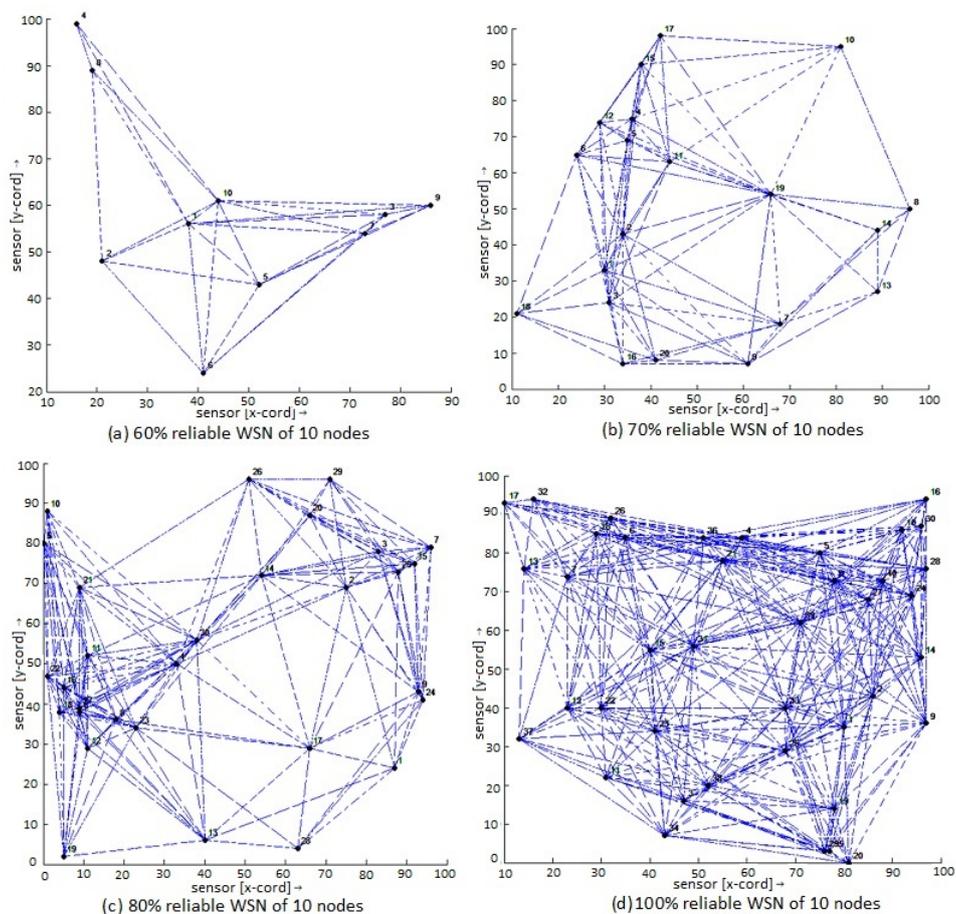


Figure 6. Random wireless sensor network.

Coding gain comparisons are given below. In the bar chart, X-axis values represent the number of nodes in the sensor network, and Y-axis values represent a coding gain. It is clearly seen that the coding gain improvement is good for 80%, 70% and 60% reliable networks, whereas for a 100% reliable network, it is as good as COPE. In real life, most of the wireless networks are less error prone due to limited resources. Therefore, the proposed mathematical model concludes that ABC-COPE could give better network coding gain in error prone WSNs with limited resources.

All of the results are shown (Figure 7) for unicast traffic. The sensor node range is assumed as 50 units in the network area of $100 \times 100 \text{ unit}^2$ to maximize the neighboring node, *i.e.*, to create a dense network.

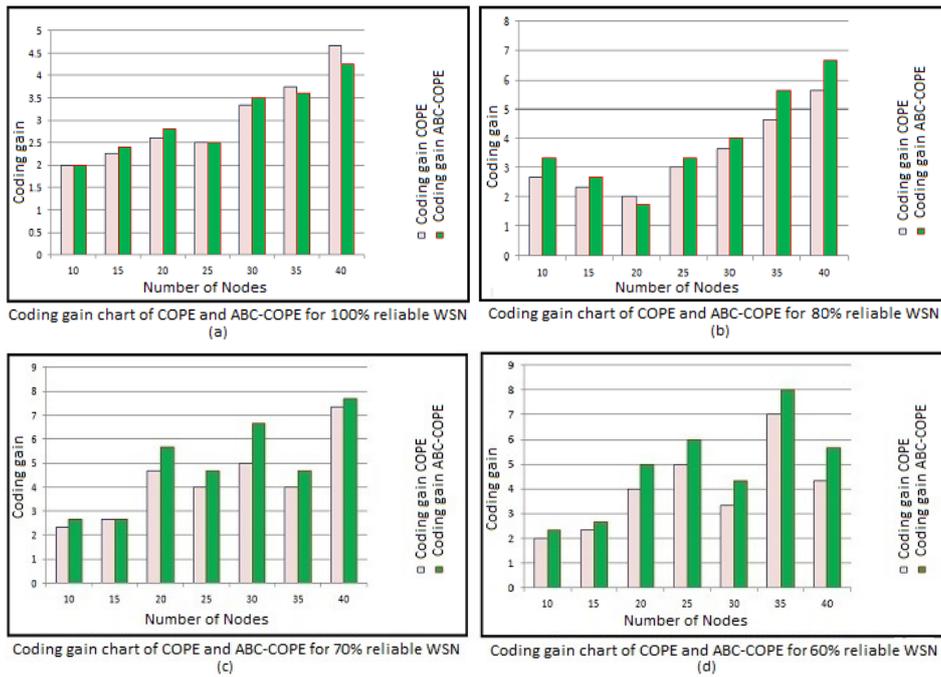


Figure 7. Coding gain chart of COPE and ABC-COPE for dense WSN.

Some random sparse WSNs are also considered for a 200×200 unit² area. The communication range of the sensor nodes is unchanged *i.e.*, 50 units of radius. The proposed coding technique executed and the results are presented in Figure 8.

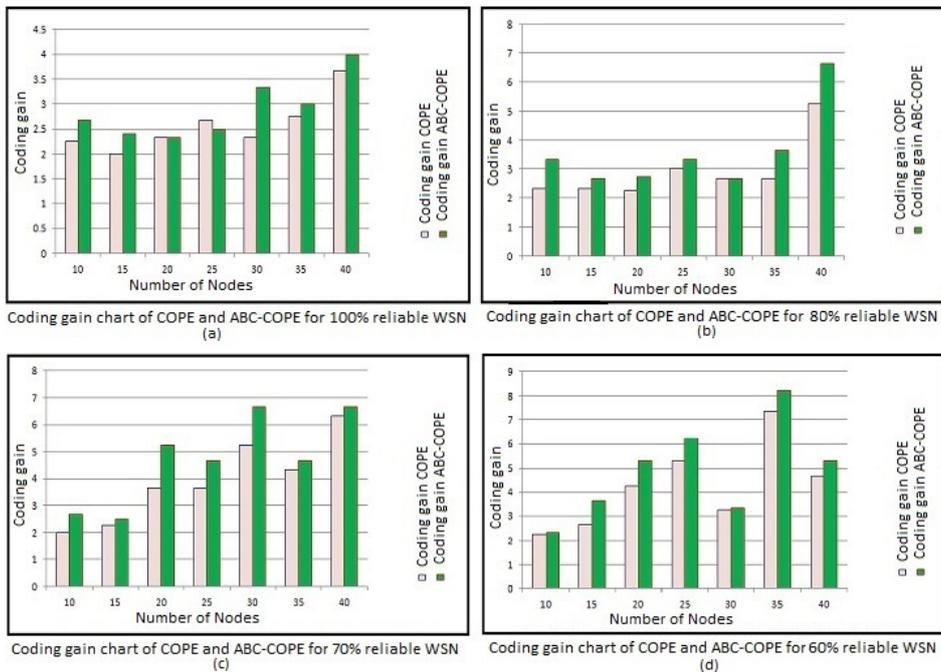


Figure 8. Coding gain chart of COPE and ABC-COPE for sparse WSN.

5. Conclusions and Future Work

The proposed research work designed and presented an updated COPE framework with a meta-heuristic approach, which is based on a nature-inspired optimization technique to enhance the

coding gain of a wireless sensor network. The proposed work performance can be improved based on the number of iterations and existing resources. The comparative results of the proposed work with existing COPE have been presented in the form of bar graphs. The proposed work performs well when it is compared to the existing COPE framework. The assumption of the proposed work is that the nature-inspired techniques need much computing power as compared to other traditional approaches, but simultaneously, it is also true that they may save much energy. Therefore, the proposed algorithm is a trade-off between the energy and processing power available. This technique considers a random initial population set and improves it at the best level. As a future scope, some initial selection procedures may be applied for population set initialization, which may enhance the coding gain and throughput of the network. We have presented comparative results for wireless sensor networks, but this will also be suitable for wireless mesh networks due to its structural deployment. The mesh network's element has high processing power and enough energy sources. Therefore, ABC-COPE will be more suitable for a wireless mesh network. In the future, the proposed framework could be applicable in WSNs and WMNs along with routing schemes to improve throughput performance. The proposed work may be simulated using wireless sensor network simulators, such as TinyOS, NS2, Qualnet, etc.

Acknowledgments: The authors would like to thank the reviewers for their constructive comments which helped to improve the quality of paper. We thank to department of computer science and Central University of Rajasthan.

Author Contributions: Aitha Nagaraju conceptually formulated the problem and guided the work to make it a worthy contribution to the technical community. Amit Singh designed the overall framework and implemented the same. Finally, both reviewed the paper content many times to improve the manuscript and to make it easier for readers to understand the methodology.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Aguayo, D.; Bicket, J.; Biswas, S.; Judd, G.; Morris, R. Link-level measurements from an 802.11b mesh network. In Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, Portland, OR, USA, 30 August–3 September 2004; Volume 34, pp. 121–132.
2. Katti, S.; Rahul, H.; Hu, W.; Katabi, D.; Médard, M.; Crowcroft, J. XORs in the air: Practical wireless network coding. *IEEE/ACM Trans. Netw.* **2008**, *16*, 497–510.
3. Liu, J.; Goeckel, D.; Towsley, D. Bounds on the gain of network coding and broadcasting in wireless networks. In Proceedings of the IEEE 26th International Conference on Computer Communications, INFOCOM, Anchorage, AK, USA, 6–12 May 2007; pp. 724–732.
4. Glatz, P.M.; Hormann, L.B.; Steger, C.; Weiss, R. Opportunistic network coding for energy conservation in wireless sensor networks. In Proceedings of the IEEE Ninth Annual Communication Networks and Services Research Conference (CNSR), Ottawa, ON, USA, 2–5 May 2011; pp. 239–246.
5. Akyildiz, I.F.; Su, W. Sankarasubramaniam, Y.; Cayirci, E. Wireless sensor networks: A survey. *Comput. Netw.* **2002**, *38*, 393–422.
6. Culler, D.; Estrin, D.; Srivastava, M. Guest editors' introduction: Overview of sensor networks. *Computer* **2004**, *37*, 41–49.
7. Yick, J.; Mukherjee, B.; Ghosal, D. Wireless sensor network survey. *Comput. Netw.* **2008**, *52*, 2292–2330.
8. Chang, F.-C.; Huang, H.-C. A survey on intelligent sensor network and its application. *J. Netw. Intell.* **2016**, *1*, 1–15.
9. Akyildiz, I.F.; Xie, J.; Mohanty, S. A survey of mobility management in next-generation all-IP-based wireless systems. *IEEE Wirel. Commun.* **2004**, *11*, 16–28.
10. Jun, J.; Sichertiu, M.L. The nominal capacity of wireless mesh networks. *IEEE Wirel. Commun.* **2003**, *10*, 8–14.
11. Akyildiz, I.F.; Wang, X.; Wang, W. Wireless mesh networks: A survey. *Comput. Netw.* **2005**, *47*, 445–487.
12. Dao, T.-K.; Pan, T.-S.; Nguyen, T.-T.; Chu, S.-C. A compact Artificial Bee Colony optimization for topology control scheme in Wireless Sensor Networks. *J. Inf. Hiding and Multimed. Signal Process.* **2015**, *6*, 297–310.
13. Kaikai, C.H.; Jiang, X.; Horiguchi, S. A more efficient cope architecture for network coding in multihop wireless networks. *IEICE Trans. Commun.* **2009**, *92*, 766–775.

14. Sengupta, S.; Rayanchu, S.; Banerjee, S. An analysis of wireless network coding for unicast sessions: The case for coding-aware routing. In Proceedings of the IEEE 26th International Conference on Computer Communications, INFOCOM, Anchorage, AK, USA, 6–12 May 2007; pp. 1028–1036.
15. Cui, T.; Chen, L.; Ho, T. Energy efficient opportunistic network coding for wireless networks. In Proceedings of the IEEE 27th Conference on Computer Communications, INFOCOM, Phoenix, AZ, USA, 15–17 April 2008.
16. Seferoglu, H.; Markopoulou, A. Network coding-aware queue management for unicast flows over coded wireless networks. In Proceedings of the IEEE International Symposium Network Coding (NetCod), Toronto, ON, Canada, 9–11 June 2010; pp. 1–6.
17. Rayanchu, S.; Sen, S.; Wu, J.; Banerjee, S.; Sengupta, S. Loss-aware network coding for unicast wireless sessions: Design, implementation, and performance evaluation. In Proceedings of the ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems, Annapolis, MD, USA, 2–6 June 2008; Volume 36, pp. 85–96.
18. Khreishah, A.; Khalil, I.M.; Ostovari, P.; Wu, J. Flow-based xor network coding for lossy wireless networks. *IEEE Trans. Wirel. Commun.* **2012**, *11*, 2321–2329.
19. Keller, L.; Atsan, E.; Argyraki, K.; Fragouli, C. SenseCode: Network coding for reliable sensor networks. *ACM Trans. Sens. Netw.* **2013**, *9*, doi:10.1145/2422966.2422982.
20. Kasireddy, B.; Wang, Y.; Liu, J. Energy conservation using network coding in grid wireless sensor networks. In Proceedings of the 3rd IEEE International Workshop on Intelligent Systems and Applications (ISA), Wuhan, China, 28–29 May 2011; pp. 1–4.
21. Biswas, S.; Morris, R. ExOR: Opportunistic multi-hop routing for wireless networks. In Proceedings of the ACM SIGCOMM 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, Portland, OR, USA, 30 August–3 September 2004; Volume 35, pp. 133–144.
22. Chachulski, S.; Jennings, M.; Katti, S.; Katabi, D. Trading structure for randomness in wireless opportunistic routing. In Proceedings of the ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, Kyoto, Japan, 27–31 August 2007.
23. Zhang, X.; Li, B. Optimized multipath network coding in lossy wireless networks. *IEEE J. Sel. Areas Commun.* **2009**, *27*, 622–634.
24. Lin, Y.; Li, B.; Liang, B. CodeOR: Opportunistic routing in wireless mesh networks with segmented network coding. In Proceedings of the IEEE International Conference on Network Protocols (ICNP), Orlando, FL, USA, 19–22 October 2008; pp. 13–22.
25. Koutsonikolas, D.; Wang, C.C.; Hu, Y.C. Efficient network-coding-based opportunistic routing through cumulative coded acknowledgments. *IEEE/ACM Trans. Netw.* **2011**, *19*, 1368–1381.
26. Khreishah, A.; Khalil, I.M.; Wu, J. Universal opportunistic routing scheme using network coding. In Proceedings of the 9th IEEE Annual Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON), Seoul, Korea, 18–21 June 2012; pp. 353–361.
27. Katti, S.; Katabi, D.; Balakrishnan, H.; Medard, M. Symbol-level network coding for wireless mesh networks. In Proceedings of the ACM SIGCOMM Computer Communication Review, Seattle, WA, USA, 17–22 August 2008; Volume 38, pp. 401–412.
28. Bonabeau, E.; Dorigo, M.; Theraulaz, G. *Swarm Intelligence: From Natural to Artificial Systems*; Oxford University Press: Oxford, UK, 1999.
29. Jeanne, R.L. The evolution of the organization of work in social insects. *Monit. Zool. Ital. Ital. J. Zool.* **1986**, *20*, 119–133.
30. Karaboga, D. *An Idea Based on Honey Bee Swarm for Numerical Optimization*; Technical Report-tr06; Engineering Faculty, Computer Engineering Department, Erciyes University: Kayseri, Turkey, 2005.
31. Singh, A.; Nagaraju, A. Network coding: ABC based COPE in wireless sensor and Mesh network. In Proceedings of the IEEE Contemporary Computing and Informatics (IC3I), Mysore, India, 27–29 November 2014; pp. 320–325.

