CHAPTER 5

PROFICIENT RATE CONTROL WITH DATA AGGREGATION AND FAIR BANDWIDTH ALLOCATION FOR WSN

5.1 INTRODUCTION

In order to solve buffer overflow and congestion in WSNs, a process for PRC-based congestion control was developed.In order to accomplish this, all of the child nodes packets are gathered into a single physical queue, which is subsequently divided into two virtual queues based on the relative significance of the traffic originating from each source. Theproblem of unequal bandwidth allocation is often overlooked by PRC-based congestioncontrol algorithms when dealing with network slowdown in WSNs. This encourages the allocation of available bandwidth fairly and the prioritizing of particular traffic types through the use of a Proficient Rate Control (PRC) approach with Fair bandwidth Allocation (PRC-FBA). Then SINR model,used in the previous studies of the bandwidth allocation problem in WSN, is an attempt tofinda medium between airness andefficiency.Through packets in a WSN network, nodes near sink nodes in the PRC-FBA drop their energy, moving from less-congested to more-congested nodes.The Proficient Rate Control with Data Aggregation and Fair Bandwidth Allocation (PRCDA-FBA) technique, which uses a data aggregation mechanism to maximize the equitable consumption of battery capacity among all participating nodes, is the solution to this problem put out by this study.Data isaggregated using adaptivenetwork coding,which improvesnetwork performancebyloweringlatency andenergy usageduringdata transmission.A commonmethod ofnetwork encoding is Random Linear Network Coding (RLNC). A node transfers datapacketsduringaconversationisquantifiedbyitstransmissionfrequency.

For a sensor node, nine hundred packets for each transmission iteration is the maximum. On the other hand, lower transmission frequency raised network channel capacity, improving network throughput overall.The network coding approach increases channel use and decreases packet redundancy in the network by combining data for transmission to the next hop. An adaptive technique that reduces packet loss rate by having nodes send data by combining network coding is activated when congestion occurs. The Priority based Network Coding approach uses packet priority, node residual energyand delay to determine whether networking coding should be enabled or disabled at theparentnode.

In addition, the bandwidth allocation of PRCDA-FBA was improved by LSTM recurrentneuralnetworks(whichcanlearnlong-termdependencies).It hadatemporaldimension that allows them to discern patterns in data sequences. This task necessitateslearning novel network behaviours while keeping track of historical data, therefore thispropertywasspecialrelevance.The future bandwidth requirements of the path are predicted using the bandwidth consumed in previous events along with factors like packet drop rate, energy, packet priority, and packet latency.

5.2 Proficient Rate Control with Data Aggregation and Fair Bandwidth Allocation Algorithm(PRCDA-FBA) Using Random Linear Network Coding

Proficient Rate Control with Data Aggregation and Fair Bandwidth Allocation (PRCDA-FBA) is a suggested method that makes use of a powerful data aggregationmechanism to maximize the equitable consumption of battery life across all involvednodes, and it is the result of the third phase of study. Through the use of adaptive networkcoding anddata aggregation,network throughputcan beincreased while waitingtimesfordatatransmissionandenergyconsumption,bothare decreased.RandomLinearNetworkCoding (RLNC)isusedtogeneratenetworkcoding.Thetransmissionfrequency of a node is the same to the number of data packets it sends in one communicationround. Sensor nodes should ideally only transmit a single packet every communicationcycle.Conversely, a decrease in transmission frequency led to an increase in network channel capacity and an improvement in overall network throughput.

The network coding approach increases channel use and decreases packet redundancy in the network by combining data for transmission to the next hop. An adaptive technique that reduces packet loss rate by having nodes send data by combining network coding is activated when congestion occurs.When using priority-based network coding, the parent node determines whether to turn on networking coding based on delay, packet priority, and node residual energy.

Bandwidth allocation in PRCDA-FBA is also improved by Long Short-Term Memory(LSTM) recurrent neural networks, which can learn long-term dependencies.

They have atemporal dimension that allows them to discern patterns in data sequences. This tasknecessitateslearning novel networkbehaviors while keeping track of historical data,therefore this property is of special relevance. The future bandwidth requirements of the path are predicted using the bandwidth consumed in previous events along with factors like packet drop rate, energy, packet priority and packet latency.

5.3NETWORK CODING IN WSN

Relay nodes combine packets using mathematical techniques in network coding, atechnique for lowering the overall number of broadcasted packets (Ostovari et al. 2014).Network coding was originally suggested for wired networks as a way to enhance performance and lessen bottlenecks.However, network coding is more appealing in these environments due to wireless networks' broadcast nature and the variety of their links. It is possible to conduct network coding in between and during sessions.

- **Inter-SessionNetworkCoding**

Inter-session network coding allows packets from different sessions (sources) to be merged, removing the bottleneck problem. There is an interference between connectionsand several copies of the same packet are made when they are not needed because wireless networks are broadcast in nature. Yet, the broadcast aspect can be turned into anadvantage bylettingtheintermediarywirelessnodesand codethe packets.The figure 5.1(a) shows the inter session of network coding.

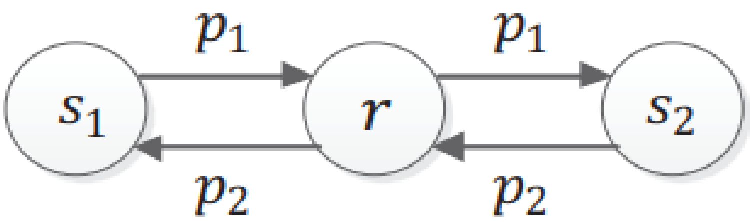

Figure 5.1 (a) Inter-Session Network Coding

In Figure 5.1(a), if the nodes s_1 and $s_2$wish to swap their personal packets, p_1 and $p_2$respectively and the nodes are not in connection with one another, hence there are four transmissions involved in this communication: two transmissions for forwarding the packets to the relay node$r$ and Two packet relay transmissions are made. However, the relay node can simply XOR the packets and send the coded packet$p_1 \oplus p_2$(Katti et al., 2006), which is shown in Figure 5.1(b).

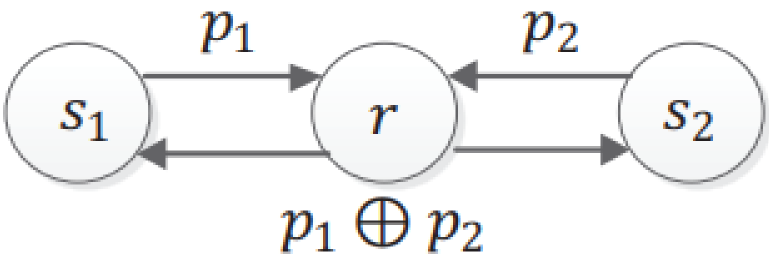

Figure 5.1 (b) Inter-Session Network Coding

The nodes $s_1$and s_2 can retrieve each others packets by XOR-ing$p_1 \oplus p_2$ with their own packets, $p_1$andp_2, respectively. As a result, binary network coding has been used to cut down the number of broadcasts to three. By combining the coding of packets from many sessions (sources), inter-session network coding eliminates bottlenecks and lowers transmission volume. Network coding decreases the number of required broadcasts, increasing speed and decreasing interference between links in wireless networks.

- **Intra-Session Network Coding**

Network coding is also widely used for reliability in wireless networks. Conventionally, dependability for wired and wireless networks is ensured by reporting received (or lost) packets through feedback messages.By using feedback messages, the sender node will be able to identify which packets need to be delivered again. Nevertheless, Bandwidth is used by these feedback messages.Figure 5.2 illustrates an intra-session network coding approach for reliability.

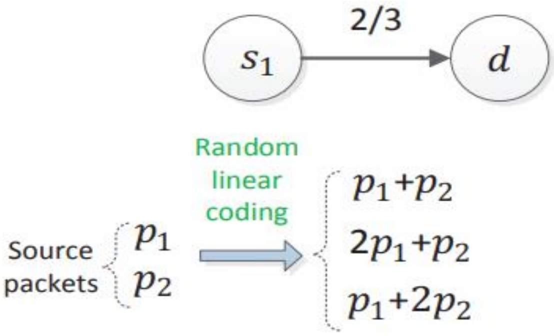

Figure 5.2 Application of intra-Session network coding to provide reliability

In Figure 5.2, the source node desires packet deliveryp_1 and $p_2$to the node d. The dependabilityof the link, $s_1 \rightarrow d$ is equal to 2/3. If three coded packets are sent by the source node,$p_1 + p_2, p_1 + 2p_2$, and $2p_1 + p_2$,Two of the three coded packets will be received by the target node on average. Consequently, the packets will be retrievable by the target nodesp_1 andp_2. If network coding is absent, the source node should broadcast each packet twice using a feedback mechanism.

In this situation, compared to non-coded approaches, network-coded communication strategies can offer reliability with fewer transmissions. Intra-session network coding is the process of coding packets from the same session (source) by utilizing a range of links.The relative values of packets from the same source stay constant in an intra-session network when they are coded together (often linearly). With packet coding, all that is needed for a relay node to do is correctly transfer coded packets from the transmitted coded packets, removing the need for a relay node to precisely identify which packets are received by the destination node.

- **Opportunistic Routing**

According to Biswas and Morris (2005), opportunistic routing techniques are a successful method of mitigating packet loss in wireless networks that do not employ network coding. It is likely that a packet that a node broadcasts would not reach its following hop.The packet is likely to be received and forwarded as the next hop by a neighbor of the sender because of the broadcast nature of the wireless medium and the diversity of links.With opportunistic routing, the packet can be relayed by any node that hears it; there is no predetermined path from the source to the destination.In the Opportunistic routing given in Figure 5.3, the delivery rates of the links are shown next to the nodes, which want to send four (P_1,P_2,$P_3$and P_4) packets to the destination.

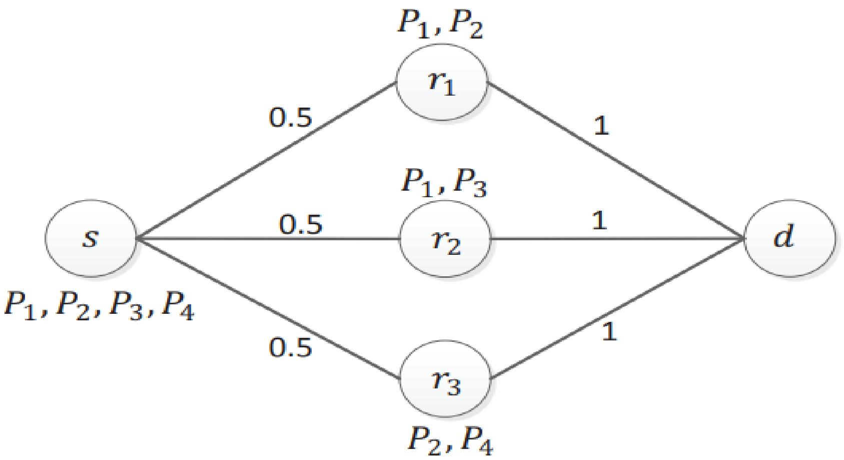

Figure 5.3 Opportunistic Routing

It made it possible for every relay node to receive the packets indicated by the nodes. The route from s to d will be fixed if the conventional shortest path routing method is applied, and the selected route is$s \rightarrow r_1 \rightarrow d$, the packets must be retransmitted by the source node p_3 and p_4. Conversely, if the additional nodes that got the packetsp_3 and $p_4$are the source node that shouldnot have to retransmit any packets if it is permitted to pass them.Organizingthe intermediary nodes is the primary obstacle in opportunistic routing. To avoid sending the same packet again, the intermediate nodes must either listen to what other nodes are sending or give feedback to them to find out if their neighbour has received it. The intermediary nodes must be able to overhear one another in order to accomplish this, which may not be feasible, as Figure 5.3 illustrates.

The source node splits the packets to be delivered into batches of k packets for this reason.The source continues to deliver form-coded packets$\sum_{i=1}^{k} \alpha_i p_i$,where $\alpha_i$is a random coefficient chosen over a finite field. After receiving a coded packet, an intermediary node verifies that it was linearly independent of the packets it had previously received. A packet will be added to the nodes buffer if that is the case. The packets in each intermediary nodes buffer are combined linearly, and the coded packets are then sent out. Once it began receiving linearly independent packets, the destination node was able to decode each packet in the batch. In this instance, the destination node notifies the source to start sending packets.

- **Cross Layer Design**

Wireless protocols present additional difficulties when using network coding techniques.It follows that the previous routing protocols were unaware of network coding. Nonetheless, the coding potential is impacted by the routing protocol. If two flows transit through relay nodes that are separated from one another, coding will not be possible. Conversely, there is more interference when flows are next to one another.Consequently, cross-layer techniques are required to improve the efficacy of the suggested protocols for wireless networks. Cross-layer approaches have independent procedures for each layer. They converse with one another, nevertheless, in order to decide and work more effectively.

In terms of performance indicators, traffic patterns and memory and processing resource availability, sensor networks in WSNs are not the same as regular wireless networks (Keller et.al., 2013). These differences make certain network coding strategies that were created for regular wireless networks inappropriate for WSNs. Nodes must be aware of their surroundings and record any communications they overhear in their buffers according to network coding regulations.The protocols used by WSNs need to be straightforward and simple to use. Furthermore, nodes in the sensor network may fail or disconnect, and the quality of the links between them varies with time. As such, it is critical to consider the dynamic environment and methods for design should be flexible enough to accommodate this dynamic nature. (Hou and associates, 2008).

5.4CLASSIFICATION OF NETWORK CODING APPROCHE

Random Linear Network Coding (RLNC) and XOR (binary) coding are two subfields that fall under the general term network coding. Figure 5.4 illustrates the mathematical process for XOR and RL network coding.

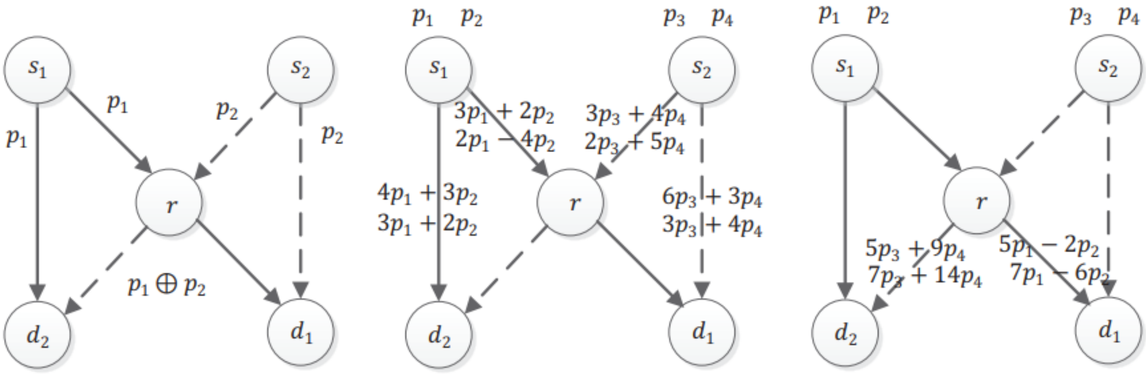

Figure 5.4 (a),(b) and (c) Mathematical operation for XOR and RL network coding

To conduct binary encoding, XOR operations are carried out between each pair of packets. Two distinct flows between two separate two node pairs (s_1 and d_1) and (s_2 and d_2) are shown in Figure 5.4(a).The relay node needs to send out two broadcasts—one for each flow—if network coding is not used. By utilizing the broadcast function of the relay node's output links and the XOR operation of the two packets, the number of broadcasts can be reduced to a single instance. By XORing the packets that were overheardp_2 and p_1 with the coded packet p_2⊕ p_2, nodes d_1 and d_2 may interpret the traffic encoded.

Coded packets in random linear coding have the form $\sum_{i=1}^{k} \alpha_i p_i$, where i is a random coefficient picked from a finite field and p_i could be a coded or uncoded byte. For the sake of demonstration, then the figure 5.4(b) linkages have a 0.5% delivery rate. There are a total of four packets that are generated and broadcast by each source node, and each of these packets carries a unique random linear code. Each session sends two packets to the relay node r, which are linearly independent of one another. Four coded packets will be generated by the relay node at the end of the session. Two packets that differ by a linear amount are sent to each recipient. An outstanding analogy for the decoding procedure is the resolution of a linear system of equations.

In Figure 5.5, the Network coding can alternatively be categorized as local or global, according on the situation. In a network, packets are sent from a relay node to the next hop nodes, which are prepared to interpret the data that is being delivered. After encoding, the packets are sent over the network, where they are decoded by nodes on the next hop using the same encoding scheme. Therefore, coding and decoding are carried out at each intermediate hop in a multi-hop transmission. Instead of decoding, intermediate nodes in a global network simply recode the packets that have already been encoded. Nodes will be able to decode packets once they have been sent to their final destinations. In contrast to global protocols se of random linear encoding, local network coding protocols' favored method of doing so is XOR coding.

Network coding, as was previously demonstrated, can occur both before and during a session. Relay nodes can reduce the number of transmissions and ease the bottleneck issue in Figure 5.4(a) by encoding packets that originate through inter-session network coding from the same session (source). Relay nodes can reduce the number of transmissions and ease the bottleneck issue in Figure 5.5(b) and (c) by encoding packets that originate from the same session (source) using inter-session network coding.The Figure 5.5 showed the classification of network coding methods.

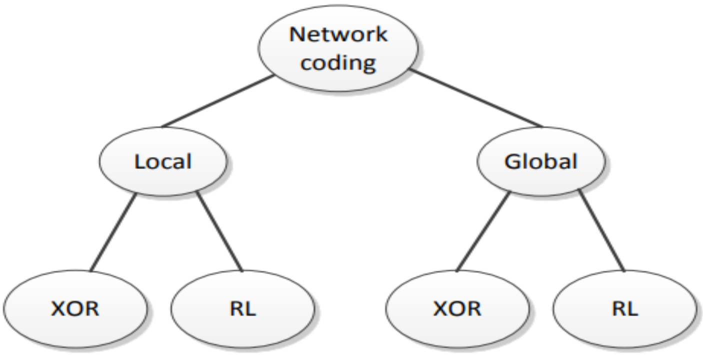

Figure 5.5 Classification of network coding methods

Nodes close to sink nodes in a WSN network have their power reduced using the previously proposed PRC-FBA model, in which packets travel from less-congested to more-congested nodes. As a result, this chapter uses the efficient data aggregate approach to maximize the battery power usage across all participating nodes. Data aggregation using adaptive network coding is used to reduce delays and energy utilization in data transmission, leading to higher total network throughput. Network coding is performed using the RLNC. The data packets sent by a node during a single exchange of information is the transmission frequency. This network coding enhances the security architecture by redistributing and restoring data in a way that is different from the original form.

A sensor nodes transmission frequency should typically not exceed one packet every transmission round. Conversely, a decrease in transmission frequency led to an increase in network channel capacity and an improvement in overall network throughput. The network coding approach increases channel use and decreases packet redundancy in the network by combining data for transmission to the next hop.When congestion happens, an adaptive technique kicks in, allowing nodes to send data by mixing network coding to reduce the rate at which packets are dropped.The parent node uses the adaptive network coding technique to determine whether or not to activate networking coding based on packet priority, node residual energy and delay.

In some cases, networking can result in decreased bandwidth because of the addition of a header or the completion of the reading process. A method of machine learning The utilization of Long Short Term Memory (LSTM) allows for accurate prediction of PRCDA-FBA's bandwidth consumptionwhich helps to prevent packet transmission along the path when capacity is already in use.

One kind of RNN that can potentially develop order dependence during training is the Long Short-Term Memory Network. In an RNN, the stage that is now being processed uses the output from the stage before it as input. This fixed the issue of RNN long-term reliance, where the RNN has trouble making correct predictions from its long-term memory yet performs admirably when working with its most recent input. Inefficient performance is delivered by RNN as the gap length increases. The LSTM was created with the ability to keep information for a very long time in mind. This technique can be used to analyze, predict, and categorize time series data. When compared to standard feed-forward neural networks, LSTM has the unique ability to recognize feedback connections.

Since, LSTM are capable of learning long-term dependencies, that allows them to discern patterns in data sequences by temporal dimension. This task necessitates learning novel network behaviours while keeping track of historical data, therefore this property is of special relevance. The bandwidth used in prior events together with variables like packet loss rate, energy, packet priority and packet latency are used to predict the paths future bandwidth needs.PRCDA-FBA is the name of the overall model that ensures all nodes in a WSN get an equitable share of the battery life.

5.5PROPOSED METHODOLOGY

It can be represented as a graph $G(V,E)$, where N is the sum of nodes and E is the sum of connections between them.$e(a,b) \in E$ defines the different nodes communication relationship between them$a \in V$ and $b \in V$, then theideal receiving node is the sink node.The relationship$e(a,b) \in E$ also symbolises the nodes a and b at the transmitter (Tr),reception (Rr) ends, respectively. When nodes are closer to one another than the transmission range, a communication link is formed. The sensor node transmitted the data it had collected to the subsequent node as soon as it exited the application field.

Then, the suggested technique increases network performance while reducing power consumption and latencies in data transmission by using network coding for data aggregation. Once per transmission round, on average, a given node will send out a certain number of packets at a certain frequency. Sensor nodes, in general, should not broadcast more than a single packet during any given transmission interval. Higher throughput was achieved, though, by increasing the network channel capacity by lowering the transmission frequency.In the network, the network coding approach reduces packet redundancy and increases channel usage by combining data for transmission to the next hop (Saranya et al., 2019).An adaptive technique is provided when congestion happens, wherein the node sends packets by increasing the packet dropping rate and aggregating them using network coding.

Network coding provides a few speed and throughput benefits in a dynamic setting. On the other hand, network coding requires a great deal of computing complexity on both transmission link ends.Consequently, the need to create an algorithm that maximizes performance while minimizing operational expenses.In order to allow the source node to alternate between archiving actual packets and sending actual packets into networks and attempting to transmit data packets into systems by coding them using Random Linear Network, a novel adaptive network coding technique was created (Yoshida et al. 2018).

Under the adaptive RLNC approach, the origin node determines whether to turn on or off networking coding based on various factors such as packet size, expected time of disconnection between connected hubs, data flow in networks, packet rate, and others.Network Coding should be turned off if nodes sent material is smaller than the link maximum sending capacity and its anticipated lifespan. After the data rate reaches a certain threshold, network coding is implemented (Xie et al., 2018). By means of encoding, packets are formed and transmitted as a concatenation of their constituent components. The original data is recovered when a receiving node receives and decodes an encoded packet.The followings are the calculations of parameter to establish Network coding from Eq. (5.1) to Eq. (5.4):

$$Packet(sizeinbits) = 8 * Packet(size) \tag{5.1}$$

$$D = Data\ rates\ in\ bps \tag{5.2}$$

$$ET = Estimated\ link\ expiration\ Time \tag{5.3}$$

$$MDT = Max\ Data\ rate\ Transmit = D * ET \tag{5.4}$$

It is both essential and sufficient for nodes to meet specific requirements in order to create ideal paths with prospective coding nodes for network coding to occur.Then the certain notations first before looking at the network coding scenario a∈d_f denotes node a beside the data flow d_f, wherein both the origin and destination nodes participate. Ns is shorthand for the set of a node's nearest neighbors, which consists of only one node Ns(a). Nodes whose arrows point Forward(a,d_f) and nodes whose arrows point Backward(a,d_f) are, On the d〗 _fdata flow path and at its source, respectively. Consequently, in the event that the network requirement was satisfied, the sensor node e at the point where the incoming flows converge encrypts the data it has collected and transmits it.

The packet flow of network is represented by the letters like O_1 and O_2. It is important to articulate the necessary and sufficient criteria for system coding in order to identify potential avenues for coding opportunities. Unless a condition has been fulfilledwhen the flows d_f1 and d_f2 overlay at node e was network coding possible (Xie et al., 2017). The problem of network coding collision has emerged because different flows may interfere with one another.

Conditions :

1.Existing node $n_1\ \epsilon Backward(a, d_{f1})$ while $n_1 \in\ N_s(m_2)\ \Lambda\ m_2Forward(e, d_{f2})$ or $n1\epsilon\ Forword(e, d_{f2})$.

2. Existing node $n_2\ \epsilon Backward(a, d_{f2})$ while $n_2 \in\ N_s(m_1)\Lambda\ m_1Forward$ (e,d_{f1}) or $n_2\epsilon Forward$ (e,d_{f1})

Here, n_1, $n_2, m1$ and $m2$are neighbours of node a and e respectively.When there are many possible paths through a network, it will select the one that offers the most ways to code. However, extensive coding at many conflicting nodes en routemay prevent a native packet from arriving at its destination decrypted.Yet, there is a point when d_{f3}flow connects to the network. Both d_{f1} and d_{f3} and Node C_2 from met the network coding condition.Node C_1 gets $O_1⛢O_3$when it encodes packets $O_1⛢O_3$ and delivered them along route df3.Furthermore, node $C_2$was the coding node, and packets will be encoded once more$O_1⛢O_3$ and O_2, i.e., $O_1⛢O_2⛢O_3$, and send them to L3 and N2, correspondingly, using the routes d_{f3} and d_{f2}. Given that it listens for packets O_1 and O_2 from nodes S1 and S2, it might observe the target node, L3 decodes the packets O_3 from $O_1⛢O_2⛢O_3$. Ought topackets reach node E2 as the destination, It cannot decipher the initial packetsO_2, but it can decode packet O_3. As can be observed, nodeC_2 cannot be used as a coding node. because of the extensive coding along the path, d_{f3}has an impact on the origin of the coding collision problem in this instance. To prevent code clashes, more limitations ought to be implemented.

An adaptive bandwidth allocation system informed by machine learning can adjust for high-bandwidth traffic patterns and cut down on latency. LSTM was trained and tested by Babu et al., (2021) to handle high-bandwidth traffic with different burstiness levels.It denotes duration(p,q) as the duration of the packet flow, k(p,q) denotes packet transmit, number of packets is represents n(p,q), and BWreq(p,q)is requested bandwidth and DataRate is denoted as DR.

Algorithm 5.1 PRCDA-FBA

Input: Set of path

Output: Selected path

Step 1: Set the following values at the outset: service time $\left(\mathrm{ST}_{\mathrm{n}}^{\mathrm{sink}}\right)$, β, δ, μ traffic class priorities and sink priority.

Step 2: Assuming that the sink node is handling all of the virtual queues, the average service time for the nth queue may be calculated as:

$$\overline{\mathrm{ST}}_{\mathrm{n}}^{\mathrm{sink}}(\mathrm{t}+1) = (1-\alpha)\overline{\mathrm{ST}}_{\mathrm{n}}^{\mathrm{sink}}(\mathrm{t}) + \alpha \cdot ST_{\mathrm{n}}^{\mathrm{sink}}$$

Step 3: Calculated the variance of raten^{th}“virtual queue”by applying the formula in the sink node

$$\Delta r^{sink} = \beta . r_{out}^{sink} - r_{in}^{sink}$$

where r_{in}^{sink} represents the sink node's input throughputandr_{out}^{sink}represents the output throughput of the sink.

β is a positive constant between 0 and 1.

Step 4: Use the below formula to find the k^{th} set of the parent nodes.

$$\Delta r^{k} = \beta . r_{out}^{k} - r_{in}^{k}$$

where r_{out}^{k} is the throughput of ak^{th}connected sibling node of the sink. the k^{th}parent node's input rate is denoted by r_{in}^{k}.

Step 4: Use the k^{th} parent node to calculate the current throughput of the n^{th} virtual queue.

Step 5: Analyze the frequency with which the n^{th} virtual queue in the k^{th} parent node is updated and forwarded to the l^{th} child node.

Step 6: Continue Step 2 to Step 5 until completion of the specified simulation period.

Step 7: Node has information to share.

Step 8: if {

Step 9: Check for active neighbouring nodes then

Step 10: Check if the data rate is sufficient by comparing it to the maximum data rate.

If $Packet(sizeinbits) \geq \mathrm{MDT}$

{

The sequences O_1, $O_{2,}$ O_3,…., O_n indicate all of the data packets included in a single coded Block or Frame.

$B(k) = O_1 \eth O_2 \eth O_3 \eth O_n$ ...

$$B(k) = \sum_{k=1}^{n} A_k \times O_k$$

```
                Else
                Compute Traffic Load Intensity $TLI_{(i)} = \frac{TL_{(i)}}{q_{max}^{(i)}}$
                End
              }
   Step 11: When flow df1 and df2 intersect in a node e, Network coding is feasible only if {
   Previous node $n_1 \epsilon$ Backward $(a, df_1)$ while $n_1 \epsilon$ $N_s(m_2)$ $\Lambda m_2 For(e,f_2)$ or $n_1 \epsilon$ $For(e,f_2)$.
   Existing node $n_2 \epsilon$ Backward(a, $df_2$) while $n_2 \epsilon$ $N_s(m_1)$ $\Lambda m_1$ For(e,$f_1$) or $n_2 \epsilon$ For(e,$f_1$).
   }
   Step 12: Eliminate Coding collision
           // Training using LSTM
   Step 13: $x_{p,q} = \{k(p,q), n(p,q), a(p,q), BWreq(p,q), duration(p,q)\}$
   Step 14: $duration(p+1,q)$
   Step 15: $y = ANN(x_{p,q})$
   Step16:BW $pred(p+1,q) =$
   $yDRtstart(p+1,q) - yDRtstart(p,q)\ tstart(p,q)$;
       $pred(p+1,q) = yDRtstart(p+1,q) - yDRtstart(p,q)\ tstart(p,q)$;
   Step 17: update $\delta(p+1,q)$
   Step 18: Computes LinkCost$LC_{m,n} = \frac{O_i}{G_{m,n}}$
   Step 19: EvaluatePacketForwardingProbability(PFP)
             If $Oi \geq 0.2$ and $R \geq 2$ then
             Send data packets according to the probability of forwarding node.
             GOTO step 21;
   Else
   Step 20: Otherwise A Request to Rebroadcast (RR)
   GOTO step 18;
```

```
End
Step 21: Else
Step 22: Broadcast Request
Step 23: WaitForReply RREP
Step 24: Received RREP
Step 25: GOTO step 10
// Testing with LSTM
Step 26: GOTO step 13 and do step 17
Step 27: End
```

Therefore, the PRCDA-FBA approach was employed as an effective data aggregation mechanism to increase the equitable battery power utilization across all participating nodes. A considerable quantity of energy was lost during the use of battery power and energy consumption, necessitating power management in the PRCDA-FBA. The suggested PRCDA-FBA model flowchart is displayed in Figure 5.6.

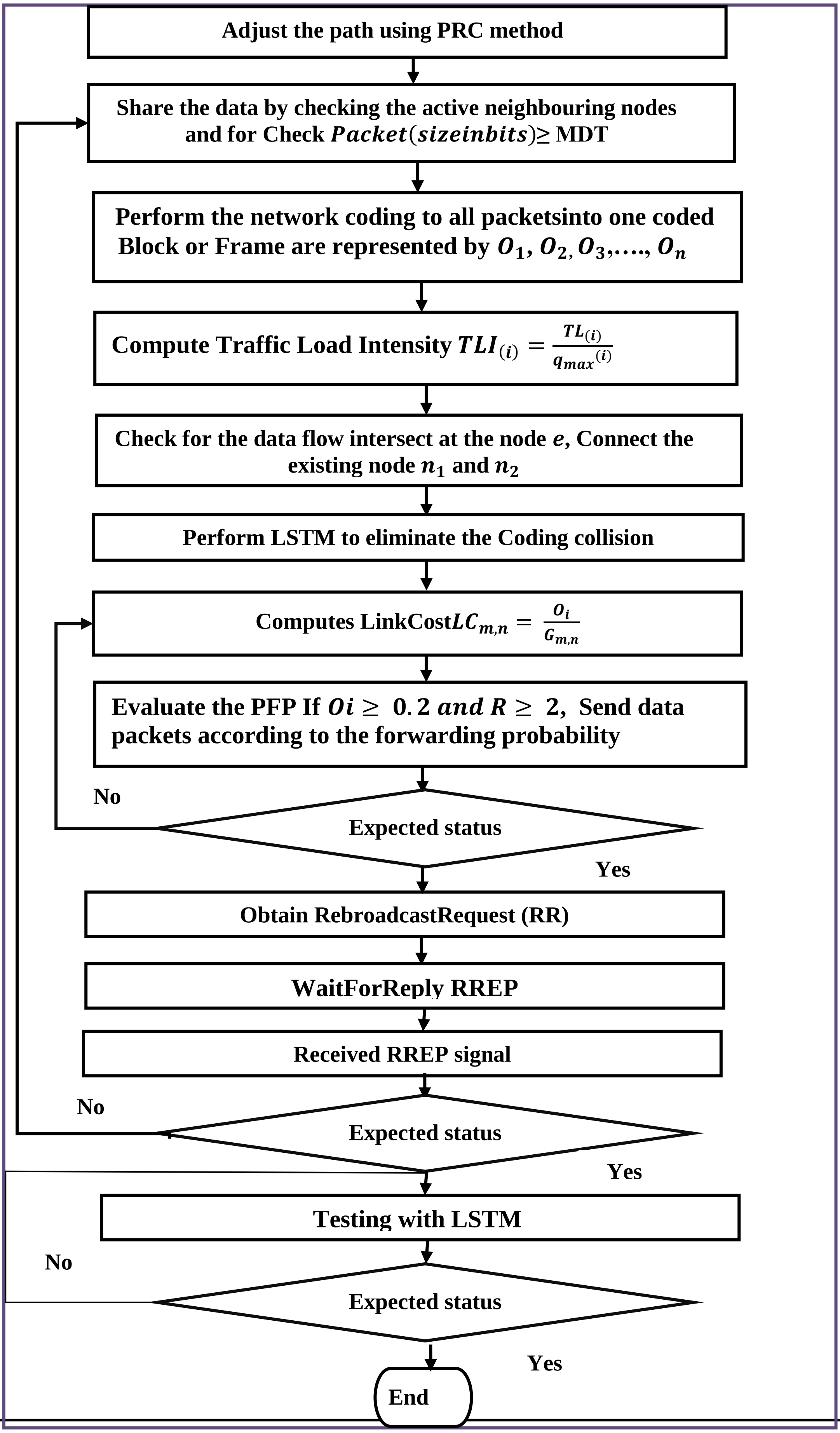
Adjust the path using PRC method
Share the data by checking the active neighbouring nodes and for Check $Packet(sizeinbits) \geq$ MDT
Perform the network coding to all packetsinto one coded Block or Frame are represented by $O_1, O_2, O_3, \ldots, O_n$
Compute Traffic Load Intensity $TLI_{(i)} = \frac{TL_{(i)}}{q_{max}{}^{(i)}}$
Check for the data flow intersect at the node e, Connect the existing node n_1 and n_2
Perform LSTM to eliminate the Coding collision
Computes LinkCost$LC_{m,n} = \frac{O_i}{G_{m,n}}$
Evaluate the PFP If $Oi \geq 0.2$ and $R \geq 2$, Send data packets according to the forwarding probability
No
Expected status
Yes
Obtain RebroadcastRequest (RR)
WaitForReply RREP
Received RREP signal
No
Expected status
Yes
Testing with LSTM
No
Expected status
Yes
End

Figure 5.6 Flow Diagram of proposed PRCDA-FBA method

5.6 SIMULATION RESULTS

The performance of PRCDA-FBA method is examined and compared with the PRC and PRC-FBA methods. The explanations ofthe simulation parameters used in this study are provided here. Based on E2E delay, packet loss, throughput, queue size, and source data transfer rate adjustment, the results are analyzed.

5.6.1 Throughput

The throughput of PRC, PRC-FBA andPRCDA-FBA was comparedwith definitenumber of iterations.Table 5.1provided thecomparison values of Throughput for proposed PRCDA-FBA with existing methods like PRC and PRC-FBA with the different number of iterations.

Table 5.1 Throughput Comparison

Simulation Time(sec)	PRC (Kbps)	PRC-FBA(Kbps)	PRCDA-FBA (Kbps)
20	375	396	406
40	408	421	431
60	429	444	454
80	450	473	483
100	472	497	507
120	490	515	525

The PRCDA-FBA was shown to produce the maximum throughput of any method examined.Throughput is higher for PRCDA-FBA by 1.94% over PRC-FBA and 7.14% over PRC techniques at the 120 second simulation time setting. Due to the equitable bandwidth allocation for each node in the network, traffic class priorities were set for each virtual queue.

5.6.2 Packet Loss

ThepacketlossofPRC,PRC-FBAandPRCDA-FBA methods was comparedwith definitenumber of iterations and is given in Table 5.2.

Table 5.2 Packet Loss Comparsion

Simulation time(secc)	PRC (%)	PRC-FBA(%)	PRCDA-FBA(%)
20	4.4	3.2	2.9
40	6.4	5.5	4
60	10.6	9.1	7
80	15.1	13.8	10
100	19.5	17.6	16
120	25.2	23.7	20

According to data, PRCDA-Fair Bandwidth Allocation, or FBA, outperforms other techniques in terms of packet loss. PRCDA-FBA reduces packet loss by 15.61% compared to PRC-FBA and by 20.6% percentage compared toPRC techniques. As a result, PRCDA-FBA had the lowest packet loss because it distributes fair bandwidth among all nodes and uses virtual queues to handle congestion over the WSN.

5.6.3. End-to-End Delay

The end-to-end (E2E) delay (in ms) for the PRC-FBA, PRC, andPRCDA-FBA methods across a range of simulation times (in sec) in shown in Table 5.3.

Table 5.3 E2E Delay Comparison

Simulation Time(sec)	PRC(ms)	PRC-FBA(ms)	PRCDA-FBA(ms)
20	83	71	60
40	101	90	79
60	130	119	108
80	142	132	121
100	170	157	146
120	183	172	161

It was shown the E2E delay of PRCDA-FBA is 6.4% lower than that ofPRC-FBA and 12.02% lower than that of PRC techniques, assuming a simulation time of120sec.Thus, the highest throughput and the least amount of packet loss was correlated with the lowest End to End latency.

5.6.4 Queue Size

The Queue Size of PRC, PRC-FBA and PRCDA-FBA was compared with definite number of iterations. Table 5.4 provided the comparison values of queue size for proposed EPRCDA-FBAwith existing algorithm like PRC, PRC-FBA and PRCDA-FBA with the number of iterations.

Table 5.4 Queue SizeComparison

Simulation time(secc)	PRC (Pkts)	PRC-FBA(Pkts)	PRCDA-FBA(Pkts)
20	5	3	2.5
40	8	6	5.3
60	11	9	8.2
80	14	11	9.4
100	18	15	14.2
120	22	18	16

Meanqueuesize(innumberofpackets)forPRC,PRC-FBA andPRCDA-FBA methodsat different simulationtimesisshownin Table 5.4. According to the findings, compared to the other ways, the PRCDA-FBA approach produces a mean queue size that was shorter on average.At 120 seconds in the simulation, PRCDA-FBA shown a meanqueue size that is 11.11% smaller than PRC-FBA and 27.27% smaller than PRC methods.Packet loss and E2E latency will consequently go down if the minimum queue length was increased.When it comes to maintaining an ideal queue length and providing a more consistent average queue size, the PRCDA-FBA clearly outperforms other options.

5.6.5 Data Transfer Rate

Table 5.5 provided thecomparsionvaluesofdatatransferrateadjustmentforthe proposedPRCDA-FBAalgorithm withthe existingalgorithms PRCandPRC-FBAfor different numberofiterations.

Table 5.5 Data Transfer Rate Comparison

Simulation time(sec)	**PRC (Pkts/s)**	**PRC-FBA (Pkts/s)**	**PRCDA-FBA(Pkts/s)**
20	68	72	74
40	64	67	69
60	60	63	65
80	57	60	63
100	54	57	59
120	51	54	57

The data transfer rate (in packets/sec) for the PRC, PRC-FBA, and PRCDA-FBA techniques was shown in Table 5.5 for various simulation times (in seconds). TheEPRCDA-FBAtechniquedeliversthehighestdatatransmission rate due to its effective rate adjustment and bandwidth distribution.The data rate of PRCDA-FBA was 5.26% higher than PRC-FBA and 10.53% higher than PRC techniques if the simulation lasts for 120 seconds.It was discovered that the PRCDA-FBA was progressively lower the data transmission rate in contrast to the nodes' initial transfer rate.Thus, the highest priority traffic classes was appropriately distributed without any congestion before lowering the transfer rate.

.