Telecommun Syst
DOI 10.1007/s11235-017-0356-6

Congestion avoidance in cognitive wireless sensor networks using TOPSIS and response surface methodology

M. Gholipour[1] · A. T. Haghighat[1] · M. R. Meybodi[2]

Abstract Congestion in wireless sensor networks degrades the quality of the channel and network throughput. This leads to packet loss and energy dissipation. To cope with this problem, a two-stage cognitive network congestion control approach is presented in this paper. In the first stage of the proposed strategy, initially downstream nodes calculate their buffer occupancy ratio and estimate congestion degree in the MAC layer. Then, they send the estimated value to both network and transport layers of their upstream nodes. The network layer of the upstream node uses TOPSIS in order to rank all neighbors to select the best one as the next relay node. In the second stage, transport layer of the given node adjusts the transmission rate using an optimized regression analysis by RSM. Extensive simulations demonstrated that the proposed method not only decreases packet loss, but also significantly improves throughput and energy efficiency under different traffic conditions, especially in heavy traffic areas. Also, Tukey test is used to compare performance of algorithms as well as to demonstrate that the proposed method is significantly better than other methods.

✉ M. Gholipour
gholipour@qiau.ac.ir

A. T. Haghighat
haghighat@qiau.ac.ir

M. R. Meybodi
mmeybodi@aut.ac.ir

[1] Computer Engineering and Information Technology Department, Qazvin Branch, Islamic Azad University, Qazvin, Iran

[2] Computer Engineering and Information Technology Department, Amirkabir University of Technology, Tehran, Iran

Keywords Congestion control · Cognitive network · Routing · Transmission rate · TOPSIS model · Response surface methodology

1 Introduction

Wireless sensor networks (WSNs) are comprised of a large number of small and cheap sensor nodes, which suffer from restricted data processing and communication capabilities to sense the environment. In event-driven sensor networks, nodes operate under idle or load states. They are extensively used in several applications such as military surveillance, habitat monitoring, healthcare, and forest fire detection, to name a few [1,2]. WSNs are mainly used to sense the environment, collect data, and process them through sensor nodes cooperation for transferring the packets to sink nodes.

Congestion occurs when the rates of incoming and outgoing packets in the nodes are not equal. When the network congestion occurs, some packets may be lost due to the limited size of the nodes buffer. This reduces the network throughput and leads to energy wasting. Congestion is one of the main difficulties in various computer networks. Wang et al. [3] have proposed an aggressive method to solve this problem in Space System of System (SOS). The proposed mechanism is characterized by a fast start procedure, along with the feedback information for purpose of analyzing network traffic as well as a link terminating processing mechanism. This can be helpful for shedding light on the real reason of packet loss, and keeping the size of the congestion window at a high level. Given the importance of congestion in WSNs, different kinds of methods have been offered to solve this problem [4]. Generally, the congestion control methods are divided into two categories:

Published online: 17 July 2017

- Intially, the source node determines the shortest path in order to send packets to sink. Then, the transmission rate is adjusted in the transport layer of node and/or in each of the relay nodes to eliminate the congestion [5–7]. Obviously, when the heavy congestion happens in the nodes located on the transmission path of the packets, throughput decreases due to a fixed path. This leads to the loss of many packets.
- These congestion control methods are concerned with the transmission of packets to idle or under-load neighbor nodes, using a dynamic routing [8–11]. To put it another way, network layer of node selects the best neighbor as the next destination through accessing the information about the distance to sink as well as the information about their traffic loading of each neighbor. Although, these methods involve bypassing the congestion area and the decreased loss of packets, the amount of delay will increase due to the longer path. Moreover, these methods don't allow bypassing the congestions areas around the sink where the traffic is heavy.

Cognitive network [12] is composed of several cognitive elements. These elements can optimize end-to-end network objectives (e.g. End-to-end delay, resource management, Quality of Service (QoS), security and etc.) dynamically through learning and decision making. Goals of cognitive networks are based on end user's pre-specified requirements. Cognitive elements can interact with each other or act individually. This type of network should be aware of what is happening around and choose appropriate actions to achieve the goals. Cognitive radio [13,14] and cross-layer design [15] are the results of previous wireless studies that have addressed some of the aforementioned issues. These technologies are able to build local cognition in networks. However, they have some shortcomings from the network perspective and therefore they fail to satisfy end-to-end global goals. In [12], the authors defined the notion of cognitive network, which has been inspired by the Knowledge Plane (KP) described in [16] as a "distributed cognitive system that permeates the network". In general, cognitive network is defined as follows: a network with a cognitive process that can perceive current network conditions, plan, decide, act on those conditions, and learn from the consequences of its actions, to meet the end-to-end requirements [12]. Definition of cognition in the cognitive network and definition of cognition in the cognitive radio share some differences and similarities.

Cognition in cognitive radio has some weaknesses from the network perspective. Cognitive radio focuses only on local objectives while cognition in the cognitive network involves end-to-end objectives across all layers and nodes in the entire network. In other words, in cognitive network, multiple layers of the nodes are involved, whereas cognitive radio involves only physical layer. Cognitive network and cross-layer designs are similar in two ways; either, non-adjacent layers can directly communicate with each other, or data required for all layers are hosted in a common repository so that they can directly share them. However, cognitive network has wider objectives than cross-layer design. The cognitive network should support trade-off between multiple objectives, whereas cross-layer is used to solve single objective [12]. In other words, observation in cognitive process spans over all multiple nodes while cross-layer has a node-based observation. In summary, cognitive network shares both the advantages of the cognitive radio and those of the cross-layer design. That is, it spans vertically over layers (thus making use of cross-layer design) and/or horizontally across technologies and nodes (thus making use of cognitive radio).

Since multiple layers are involved in solving the congestion problem, a cross-layer Cognitive Network Congestion Control (CNCC) scheme is proposed. The CNCC uses buffer occupancy ratio (queue length) and congestion degree estimation in MAC layer of the downstream nodes, which are sent to the network and transport layers of their associated upstream nodes. An upstream node uses TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) method to compare all neighbors' data, assessing them as the next hop. Network layer selects the next hop based on the tradeoff between neighbor's depth and neighbor's traffic facts to deal with congestion problems. Then, transport layer adjusts the data transmission rate of the node to avoid congestion and improve end-to-end throughput. Response Surface Methodology (RSM) is used to investigate the effects of the buffer occupancy and the congestion degree in data transmission rate of nodes.

TOPSIS is one of the most widely used methods for Multi Attribute Decision Making (MADM) proposed by Hwang and Yoon [17]. In this method, the ideal alternative solution is selected based on the shortest distance from the ideal position and the farthest distance from the negative positions. Ideal position has maximum benefit and minimum cost criteria, whereas negative position has maximum cost and minimum benefit criteria. RSM [18,19] consists of a series of experimental techniques that forecast a model based on different values of input parameters and evaluate the corresponding responses. The commonest applications of RSM are in situations where several input variables influence some performance measure of the process. Indeed, RSM is an advanced regression method in which the values of the input parameters are not random. Thus, values are selected based on statistical and mathematical techniques that can cover most of the input configurations. By using the proposed method which is based on the RSM, the output would be optimized on the basis of different input values.

The remainder of this paper is organized as follows: Sect. 2 reviews related works. Section 3 describes the problem definition and assumptions generally. Section 4 briefly describes the proposed method and builds up a reasonable system model. In Sect. 5, the implementation of the proposed method is discussed in detail. Section 6 analyzes the performance of the algorithm via extensive simulations. The influence of different parameters on the performance metrics is studied in this section too. Discussion about the result is presented in Sect. 7. This discussion shows that the proposed method is significantly better than other methods, using a Tukey test. Finally, Sect. 8 concludes the paper.

2 Related works

Congestion control is one of the important issues in WSNs and has been investigated by many studies [4–11]. There are several congestion control methods that use transmission rate adjustment [5–7]. The first study on this category named Congestion Detection and Avoidance (CODA) was proposed by Wan et al. [5]. CODA consists of three main strategies: Congestion detection based on the received message, hop-by-hop open-loop feedback to the upstream nodes, and closed-loop adjustment of the transmission rate in the source nodes. CODA guarantees the appropriate network throughput by adjusting the close-loop rate. In this method, continuous monitoring of the channel causes abundant energy consumption in the involved nodes. Differed Reporting Rate (DDR) [6] algorithm uses differed reporting rate to control congestion. This method mainly runs on each source node. It uses buffer occupancy ratio as congestion metric. If the buffer occupancy ratio is greater than the threshold value, then the congestion notification bit of outgoing packets is set and sending rate of nodes should be decreased. Otherwise, the congestion notification bit gets zero and sending rate of nodes should be increased. In this algorithm, differed flow rate component is defined which adjust sensor nodes reporting rate. In the differed reporting rate adjustment, sensor nodes, which are near the sink, have minimum reporting rates and this rate will rise for the nodes that are away from the sink. The output rate of a node is adjusted with newly calculated rate until next congestion notification will be sensed. The weakness of this approach is that each node determines the transmission rate based on the congestion notification of neighbor and relative congestion between two neighboring nodes has not effect on increasing or decreasing the transmission rate. A new transmission rate control method based on Support Vector Machines (SVMs) classification is presented in [7]. The strategy calculates buffer occupancy ratio and estimates the congestion degree of the downstream node. Then, it sends this information to the current node. The current node adjusts the transmission rate to tackle the problem of congestion, improving the network throughput by using multi-classification obtained via Support Vector Machines. SVM parameters are tuned, using genetic algorithm.

The above-mentioned methods focus only on the transmission rate of nodes and therefore they do not work well in congested environments. On the other hand, some studies conducted in this domain have sought to propose dynamic routing protocols by using the idle or under-loaded nodes to solve this problem. Most of these methods use gradient search to figure out dynamic routing approaches to improve the overall network throughput [8–11]. In a gradient search, each node builds its own gradient field and then sends it to all one-hop neighbor nodes, informing them about its situation. Each node compares all neighbors' gradient field and selects the best one as the next relay node. The cost model can be designed in terms of hop count from sink to node, physical distance, energy consumption, or cumulative delay, depending on the objectives of routing such as energy consumption, packet delay, or packet delivery ratio [8]. In what follows, some of these proposed methods will be discussed.

One of the first studies that applies the concept of gradient to route the packets toward sinks is presented in [11]. The proposed method in this work uses long term information field and data rate to increase the WSN life time. In Gradient-based Routing Protocol for Load-balancing (GLOBAL) [8] every node estimates the cumulative traffic of the least loaded path designated for the specific sink using the gradient field and then forwards the packets to the next hop. Gradient based Traffic-Aware routing (GRATA) [9] utilizes the expected packet delay and the number of hops at one-hop neighbor to make routing decisions. The gradient field in this model contains two types of information: geographic distance and forwarding delay. Indeed, forwarding delay is chosen to reflect the traffic situation at each node. However, because of imprecise information, the routing scheme might forward a packet to another congested area. In [10], the authors presented a distributed traffic-aware routing scheme with a capability of adjusting the data transmission rate in a multi-sink WSN. The algorithm constructs a hybrid virtual gradient field using shortest path and traffic load metrics for routing. The disadvantage of this method is the modelling of the objective function which uses a very simple linear weighted function. As a result, the weights of these variables are not accurate and in many cases, the algorithm could not solve the problem of congestion.

3 Problem definition

When the rate of input packet into a sensor node is higher than its output, the buffer queue of the same node will become full gradually. This leads to the destruction of new input packets, resulting in congestion in wireless sensor network. Given the destructive effect of congestions on quality of ser-

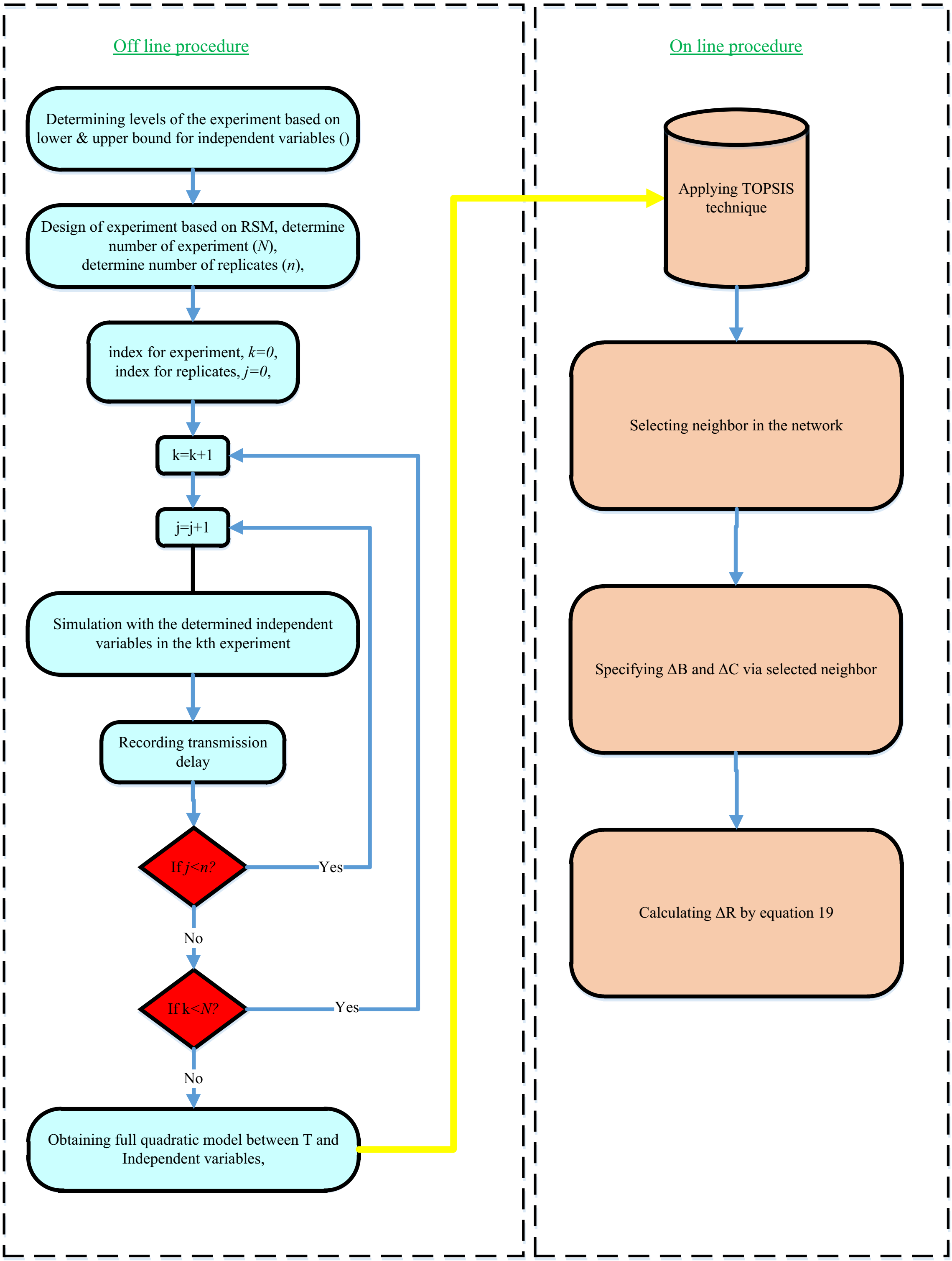

Off line procedure
Determining levels of the experiment based on lower & upper bound for independent variables ()
Design of experiment based on RSM, determine number of experiment (N), determine number of replicates (n),
index for experiment, k=0, index for replicates, j=0,
k=k+1
j=j+1
Simulation with the determined independent variables in the kth experiment
Recording transmission delay
If j<n?
Yes
No
If k<N?
Yes
No
Obtaining full quadratic model between T and Independent variables,
On line procedure
Applying TOPSIS technique
Selecting neighbor in the network
Specifying ΔB and ΔC via selected neighbor
Calculating ΔR by equation 19

Fig. 1 The network topology

vice (QOS) in sensor networks, it is necessary to control this effect. In fact, when source nodes or the relay nodes intend to transmit the packets to the sink node, they need to choose the best neighbor through the shortest path with the lowest traffic load. The nearest neighbor to the sink is not necessarily the best neighbor to receive the packets. This is because the nearest neighbor to the sink may have the highest traffic load. Consequently, the node has to transmit the packet to the nodes that have equal or more distance from the sink. This allows the node to bypass the congestion, delivering the packets through a more distant path.

Moreover, the transmission rate of packet is enormous importance when a neighbor is chosen as the receiver of the packet. Clearly, when the traffic load of the neighbor node is higher than that of the current node, the transmission rate should be reduced and the traffic load of the neighbor node is lower than that of the current node, the transmission rate should be increased.

Given the multi-layered nature of congestion as well as the advantages mentioned above for each category of congestion control methods, in this paper a two-phase algorithm based on the cognitive network concept is proposed to meet the fidelity requirements. In the first phase, the algorithm specifies the most appropriate path to forward the packets across non-congested areas. Also, in the second phase, the data transmission rate is adjusted based on congestion conditions of nodes.

Besides, it is important to figure out the characteristics of traffic in each node, which is addressed by this paper. In addition the following assumptions are made.

- Nodes are assumed to be fixed in the network.
- Communication radius of all nodes is assumed to be equal.
- The data is assumed to be deterministic in this method.
- Simulation is performed in a given time.
- The initial energy of all nodes is assumed to be equal.

Figure 1 summarizes the network topology to solve the congestion problem.

4 CNCC: cognitive network for congestion control

The cognitive network framework works on different network layers to achieve an end-to-end goal [12]. In this section, a new multilayer cognitive network is introduced to solve the congestion problem. The cognitive process in each node consists of two cognitive elements: Routing control and Congestion rate control. Routing control's objective is to use the idle or under-loaded nodes to avoid congestion. That is, a node will attempt to select the best neighbor as the next relay node considering the trade-off between shortest path and surrounding neighbors traffic. Having a packet to send toward the sink node, node (s) calculates and compares the routing metrics of its neighbors $x(x\epsilon nbr(s))$, using TOPSIS method. TOPSIS ranks all neighbors and selects the suitable neighbor as the next relay node. After selecting the next relay node, the *Congestion rate control* dynamically adjusts the data transmission rate in the transport layer of the sender node based on the MAC layer channel information, including buffer occupancy ratio and congestion degree of receiver node. In order to predict the data transmission rate in each node, an optimized regression analysis is performed by RSM. Analysis of the results shows that RSM can be successfully used to predict the data transmission rate, using buffer occupation and congestion degree.

4.1 Routing control

The proposed scheme aims at figuring out the utilization function using two factors: (1) depth of the node and (2) traffic loading which consists of normalized queue length, congestion degree of each node, and average cumulative queue length in each node. The depth of a node (i.e., Distance cost) finds the shortest path toward the specific sink. The second factor bypasses the congested areas. Congestion happens when traffic factors exceed the threshold in each node. The congested node is not suitable for receiving data and hence, packets should be re-routed across other paths. As a result, this method should meet a trade-off between shortest path and traffic factors at overloaded nodes. Assuming a packet (p) is in the buffer of node v, the next hop should be determined using the following metrics.

- **Depth of node field**

The depth of a node is the basis of the proposed routing method and is defined as the minimum hop counts from the given node to its corresponding sink i.

$$V_d^i(v) = \min(hop - count). \tag{1}$$

It is clear that the depth difference between node v and its neighboring node x, $x\epsilon nbr(v)$, is -1, 0, or 1. Therefore, the depth field encourages the packet to be forwarded directly across the shortest path to the sink.

- **Normalized queue length field**

The queue length field at node v is defined as the number of packets in the buffer of node v:

$$V_q(v) = Q(v). \tag{2}$$

The normalized buffer size at node v, Q (v), is defined as the buffer occupancy ratio of node v as below:

$$Q(v) = \frac{number\ of\ packet\ in\ the\ queue\ buffer}{buffer\ size\ at\ nodev} \tag{3}$$

The value of Q(v) is in the range of [0, 1], which specifies nodes' traffic information. Based on this field, packets will be forwarded toward the idle or under-loaded area.

- **Congestion degree field**

Using the normalized queue length field, the packets will be routed across the non-congested areas. However, sometimes congestion happens because of the burst traffic. When a burst occurs around node x with low queue length, obviously x is not a suitable candidate to be the next relay node, since a lot of packets will enter the queue of x at the same time. Therefore, queue length is not solely a reasonable metric for determining the next relay node. To cope with this issue, another metric named 'congestion degree' field is defined. Congestion degree indicates the changing tendency of the queue buffer in a period of time. The value of V_c is defined as follows:

$$V_c = \frac{T_s}{T_a} \tag{4}$$

In Eq. 4, T_a denotes the time interval between the arrivals of two adjacent data packets in MAC layer and T_s represents the average processing time of data packets in the node. If $V_c > 1$, the arrival rate is greater than the departure rate of data packets. In other words, congestion may occur in this node in the near future. As a result, this node is not a suitable choice for the next relay node.

- **Average cumulative queue length field**

The field of average cumulative queue length determines the estimated congestion of the best path toward the sink. Each node inserts its normalized buffer size into the awareness packet (AP) and then broadcasts it to neighbors in order to update their TOPSIS decision matrix. Upon receiving APs, every node ranks all neighbors based on their associated queue length field and selects the best one as the next relay node. Then, it adds its queue length to its best neighbor's cumulative queue length based on Eq. 5.

$$V_a^i = \frac{\sum_0^n Q(v)}{n} \tag{5}$$

where $\sum_0^n Q(v)$ is the cumulative queue length from node v towards sink i and n is the number of hops across the best path toward the sink.

4.1.1 TOPSIS method

TOPSIS is one of the most widely used methods for Multi Attribute Decision Making (MADM) proposed by Hwang and Yoon [17]. According to this technique, the best alternative is selected based on the shortest distance from the ideal position and the farthest distance from the negative position. Ideal position has two criteria: maximum benefit (for positive criteria) and minimum cost (for negative criteria), whereas negative position has a maximum cost (for positive criteria) and minimum benefit (for negative criteria). In short, ideal solutions consist of all the best values available based on the criteria, while the negative ideal solutions contain all the worst values available according to the criteria. Numerous investigations use TOPSIS as the solution for MADM problems [17]. To use TOPSIS method, the following steps should be considered:

Step 1 At first, decision matrix for ranking is constructed as follows:

$$D = \begin{array}{c} \\ A_1 \\ A_2 \\ \vdots \\ A_i \\ \vdots \\ A_n \end{array} \begin{array}{c} \begin{array}{cccccc} F_1 & F_2 & \cdots & F_j & \cdots & F_J \end{array} \\ \begin{bmatrix} f_{11} & f_{12} & \cdots & f_{1j} & \cdots & f_{1J} \\ f_{21} & f_{22} & \cdots & f_{2j} & \cdots & f_{2J} \\ \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ f_{i1} & f_{i2} & \cdots & f_{ij} & \cdots & f_{iJ} \\ \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ f_{n1} & f_{n2} & \cdots & f_{nj} & \cdots & f_{nJ} \end{bmatrix} \end{array} \tag{6}$$

where, F_j denotes jth desired criterion ($j = 1, 2, \ldots,$ J), A_i represents the desired options i ($i = 1, 2, \ldots,$ n), and f_{ij} is the value of option A_i according to criterion F_j.

Step 2 Calculate the normalized decision matrix $\mathbf{R}(= |r_{ij}|)$ to scale weight options. The normalized value r_{ij} is calculated as follows:

$$r_{ij} = \frac{f_{ij}}{\sqrt{\sum_{i=1}^n f_{ij}^2}} j = 1, 2, ..J \quad i = 1, 2, \ldots, n; \tag{7}$$

Step 3 The final weight of each criterion is calculated using the following method:

$$V_{ij} = w_j {}^* r_{ij} j = 1, 2, \ldots, J; i = 1, 2, \ldots, n; \tag{8}$$

where, w_j denotes the weight of the jth criterion.

Step 4 Positive and negative ideal vectors are computed by using the following two relations:

$$A^* = \{v_1^*, v_2^*, ..., v_j^*\} = \{(\min_i v_{ij} \left| j \in I' \right.\}, \tag{9}$$

$$A^- = \{v_1^-, v_2^-, ..., v_j^-\} = \{(\max_i v_{ij} \left| j \in I'' \right.\}, \tag{10}$$

where, I' and I" are associated with the benefit criterion and the cost criterion, respectively.

Step 5 The separation measures from positive-ideal solution and negative-ideal solution are determined as follows:

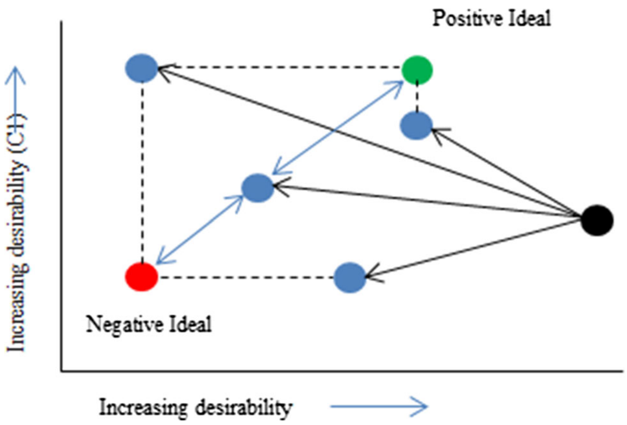


Fig. 2 Graphical representation of TOPSIS

$$D_i^* = \sqrt{\sum_{j=1}^{J}\left(v_{ij} - v_j^*\right)^2} \quad i = 1, 2, \ldots, n \qquad (11)$$

$$D_i^- = \sqrt{\sum_{j=1}^{J}\left(v_{ij} - v_j^-\right)^2} \quad i = 1, 2, \ldots, n \qquad (12)$$

Step 6 The relative closeness to the ideal solution is determined, using the following formula. The relative closeness of the alternative A_i can be expressed as follows:

$$C_i^* = \frac{D_i^-}{D_i^* + D_i^-} \quad i = 1, 2, \ldots, n \qquad (13)$$

where, the C_i^* index value lies between 0 and 1. The larger the index value, the better the performance of the alternatives.

Step 7 Finally, we rank the alternatives based on C_i^* values.

Figure 2 shows how the next destination of packets is selected between the candidate nodes, using TOPSIS which measures the distance of each node from the ideal points D_i^+ and negative ideal D_i^- (The ideal positive and ideal negative are the best and the worst criteria between the available criteria in candidate nodes, respectively). This method defines the best neighbor as the alternative that has the longest distance from negative ideal and the shortest distance from the ideal positive. In Fig. 1, the black node is a current one and should transfer the message to one of its neighbors (candidates). Neighboring nodes are in blue. As mentioned, the current node calculates the distance of each neighbor from the positive ideal and negative ideal. TOPSIS methodology will describe how these distances are calculated. Then, the value of relative closeness will be calculated for each candidate node Ci so that the node with the least relative closeness will be selected as the next destination. The green and red points represent the positive and negative ideals which are selected based on the features of candidate nodes so that for each criterion the best and the worst states for each neighbor are considered as the positive and negative ideals, respectively.

In the proposed algorithm, the decision matrix consists of 4 columns (V_d, V_q, V_c, V_i^a). The number of matrix rows is variable and equal to the number of neighboring nodes. For instance, assuming nodes x, y, z are neighbors of node v, the decision matrix of node v is represented as below:

$$D_v = \begin{array}{c} \\ x \\ y \\ z \end{array} \begin{array}{c} \begin{array}{cccc} V_d & V_q & V_c & V_a^i \end{array} \\ \begin{bmatrix} f_{xd} & f_{xq} & f_{xc} & f_{xa} \\ f_{yd} & f_{yq} & f_{yc} & f_{ya} \\ f_{zd} & f_{zq} & f_{zc} & f_{za} \end{bmatrix} \end{array}$$

As an example, F_{xq} indicates the normalized queue size of node x. Each node calculates its closeness to the ideal solution and ranks the performance order and then selects the node with larger relative closeness value (C_i) as the next hop. Weighted factors have an effect on choosing between depth of node and traffic cost information before forwarding a packet depending on the requirements. It is obvious that w is a factor that affects the routing decision. Weighted factors consider the depth of the node as the basis of the proposed scheme.

Using TOPSIS to select the next destination in each node is considered as an appropriate approach since the complexity of this method is an O (m*n) where n represents the number of each node's neighbors and m represents the number of selection criteria [19]. In the same veins, given that calculation complexity is linear and compatible with n and m, it follows that the calculation costs are not high. As a result, there will be no problem for online-calculation in sensing nodes with limited energy.

4.2 Congestion rate control

Bypassing the intermediate local hotspots cannot solely omit the congestion. In order to solve this problem completely, we propose a hop-by-hop congestion control scheme in this sub-section. Here, each node receives the MAC layer channel information of upstream node and then adjusts the data transmission rate. This strategy affects some metrics of network performance such as energy saving, transmission fairness, and network throughput. When the current node processes the AP signal received from a downstream node, it should also consider its own congestion condition and adjust the data transmission rate based on this information. In our local congestion processing method, R is the given node's data transmission rate and $Rmax$ is the maximum data transmission rate at each node. Also, ΔR determines the amount of increasing or decreasing data transmission rate in each node. Let Br and $B'r$ be the buffer occupancy ratio of the current and downstream nodes respectively, which can be attached in AP signal. If $Br > B'r(\Delta B > 0)$, then the current node is more congested than the downstream node. Clearly, the local node should increase its data transmission rate. However, in this case, the congestion degree of downstream node may be greater than that of current node ($C'_\mathrm{d} > \mathrm{C_d}$). In other words, changing tendency of downstream node is greater than the changing tendency of the current node. In this situation, according to the proposed algorithm the transmission rate is increased or decreased, using the trade-off between buffer occupancy and the congestion degree. We used RSM (Response Surface Methodology) in order to estimate the proper transmission rate.

RSM consists of a series of experimental techniques that forecast a model based on different values of input parameters and evaluate the corresponding responses. The commonest applications of RSM are in the situations where several input variables influence some performance measure of the process. RSM is an advanced regression method, in which the values of the input parameters are not random; instead they are selected based on statistic and mathematical techniques that can cover most input possibilities. By using the proposed RSM model, the output will be optimized based on different input values. In other words, RSM uses the central composite design (CCD) as an experimental model, which has the correct forecast in all directions from the center; the total number of combinations is $4\mathrm{k} + \mathrm{n}_0(2\mathrm{k} + 2\mathrm{k} + \mathrm{n}_0)$ where k represents the number of independent variables and n_0 denotes the number of repetitions of the experiments at the center point [19].

As it is known, buffer occupancy, congestion degree, and data transmission rate (ΔB, ΔC and ΔR) are independent variables which alleviate congestion in each node. The effect of these variables on congestion control is studied by RSM. On the basis of conducted experiments, the different values of buffer occupancy, congestion degree, and data transmission rate between two neighbor nodes (ΔB, ΔC, and ΔR) are normalized between −1 and 1. To evaluate the performance of these metrics in congestion control, we determined transmission delay using different combinations of these three parameters. It is clear that the optimal delay is zero. The behavior of the system is expressed by the following second-degree polynomial equation [19]:

$$Y = B_0 + \sum\nolimits_{i=1}^{n} B_i x_i + \sum\nolimits_{ij} B_{ij} x_i x_j + \sum\nolimits_{j=1}^{n} B_{jj} x_i^2. \tag{14}$$

where, Y represents the predicted response and x_i denotes the value of the *ith* independent variable. Thus, by substituting the value 3 for n, Eq. 14 becomes:

$$\begin{aligned} Y = {} & B_0 + B_1x_1 + B_2x_2 + B_3x_3 + B_{12}x_1x_2 + B_{13}x_1x_3 \\ & + B_{23}x_2x_3 + B_{11}x_1^2 + B_{22}x_2^2 + B_{33}x_3^3. \end{aligned} \tag{15}$$

where $x_1, x_{2,} x_3$ represent input variables; B_0 is a constant; and B_1, B_2 and B_3 denote linear coefficients. In this equation, B_{13} and B_{23} are quadratic coefficients. In this study, $x_1 = \Delta B$, $x_2 = \Delta C, x_3 = \Delta R$ and T is also the transmission delay. Therefore:

$$\begin{aligned} T = {} & B_0 + B_1\Delta B + B_2\Delta C + B_3\Delta R + B_{12}\Delta B\Delta C \\ & + B_{13}\Delta B\Delta R + B_{23}\Delta C\Delta R + B_{11}\Delta B^2 \\ & + B_{22}\Delta C^2 + B_{33}\Delta R^2 \end{aligned} \tag{16}$$

In our RSM method, there are 2^3 factorial central composite designs for three independent variables. Additionally, our method has 6 axial points and 6 replicates at the center points applied to fit a second-order polynomial model, indicating that 20 experiments are needed for this procedure. To obtain more accurate results, each experiment is repeated 5 times. Therefore, RSM has100 experiments with different combinations of parameters to evaluate the effect of 3 independent variables on transmission delay so that it can obtain a suitable second-degree polynomial equation (Table 1).

Design Expert software [21] is used for regression analysis of the data. The fitness of regression models is checked by the adjusted coefficient of determination (R^2_{Adj}) Also, the statistical significance of the model is determined by the application of Fischer's F test.

Based on preliminary experiments, the effect of buffer occupancy, congestion degree, and data transmission rate on transmission delay are evaluated using RSM. Application of RSM yielded the following regression equation, which is an empirical relationship between three aforementioned factors. The following formula is the second-order polynomial equation showing the fitness response surface:

$$\begin{aligned} T = {} & 0.6 + 0.04\Delta B - 0.04\Delta C - 0.08\Delta R \\ & + 0.4\Delta B\Delta C - 1.25\Delta B\Delta R - 0.7\Delta C\Delta R \\ & + 0.6\Delta B^2 - 0.2\Delta C^2 + 0.6\Delta R^2. \\ S.T & \\ -1 \le & \Delta B \le 1 \\ -1 \le & \Delta C \le 1 \\ -1 \le & \Delta R \le 1 \end{aligned} \tag{17}$$

Table 2 depicts the analysis of variance (ANOVA) used to obtain regression equation. In the test phase, 40 experiments with different combinations of parameters using the Eq. 18 are applied to the nodes. The results show that the equation predicts the exact amount of the transmission delay in 92% of cases.

Table 1 RSM 100 experiments with different combinations of parameters

R.O	ΔB	ΔC	ΔR	T	R.O	ΔB	ΔC	ΔR	T
1	−1	−1	−1	0	51	0	−1	0	1
2	1	−1	−1	3	52	0	1	0	1
3	−1	1	−1	2	53	0	0	−1	1
4	1	1	−1	4	54	0	0	1	1
5	−1	−1	1	4	55	0	0	0	0
6	1	−1	1	1	56	0	0	0	0
7	−1	1	1	2	57	0	0	0	0
8	1	1	1	0	58	0	0	0	1
9	−1	0	0	2	59	0	0	0	0
10	1	0	0	1	60	0	0	0	1
11	0	−1	0	1	61	−1	−1	−1	0
12	0	1	0	1	62	1	−1	−1	2
13	0	0	−1	1	63	−1	1	−1	0
14	0	0	1	2	64	1	1	−1	4
15	0	0	0	0	65	−1	−1	1	4
16	0	0	0	0	66	1	−1	1	0
17	0	0	0	1	67	−1	1	1	1
18	0	0	0	0	68	1	1	1	0
19	0	0	0	0	69	−1	0	0	2
20	0	0	0	2	70	1	0	0	1
21	−1	−1	−1	0	71	0	−1	0	0
22	1	−1	−1	2	72	0	1	0	0
23	−1	1	−1	1	73	0	0	−1	2
24	1	1	−1	4	74	0	0	1	1
25	−1	−1	1	4	75	0	0	0	0
26	1	−1	1	2	76	0	0	0	1
27	−1	1	1	1	77	0	0	0	0
28	1	1	1	0	78	0	0	0	2
29	−1	0	0	2	79	0	0	0	0
30	1	0	0	1	80	0	0	0	0
31	0	−1	0	1	81	−1	−1	−1	0
32	0	1	0	1	82	1	−1	−1	2
33	0	0	−1	2	83	−1	1	−1	0
34	0	0	1	2	84	1	1	−1	4
35	0	0	0	0	85	−1	−1	1	4
36	0	0	0	0	86	1	−1	1	0
37	0	0	0	0	87	−1	1	1	2
38	0	0	0	1	88	1	1	1	0
39	0	0	0	2	89	−1	0	0	1
40	0	0	0	0	90	1	0	0	1
41	−1	−1	−1	0	91	0	−1	0	0
42	1	−1	−1	1	92	0	1	0	0
43	−1	1	−1	0	93	0	0	−1	1
44	1	1	−1	4	94	0	0	1	1
45	−1	−1	1	3	95	0	0	0	1
46	1	−1	1	0	96	0	0	0	0
47	−1	1	1	1	97	0	0	0	2
48	1	1	1	0	98	0	0	0	0
49	−1	0	0	1	99	0	0	0	0
50	1	0	0	2	100	0	0	0	0

Table 2 Analysis of variance for delay

Source	Df	Seq SS	Adj SS	Adj MS	F	*P* value
Regression	9	111.580	111.580	12.3978	31.50	0.000
Linear	3	0.480	0.480	0.1600	0.41	0.749
ΔB	1	0.080	0.080	0.0800	0.20	0.653
ΔC	1	0.080	0.080	0.0800	0.20	0.653
ΔR	1	0.320	0.320	0.3200	0.81	0.370
Square	3	22.600	22.600	7.5333	19.14	0.000
ΔB × ΔB	1	17.640	4.950	4.9500	12.58	0.001
ΔC × ΔC	1	0.010	0.550	0.5500	1.40	0.240
ΔR × ΔR	1	4.950	4.950	4.9500	12.58	0.001
Interaction	3	88.500	88.500	29.5000	74.96	0.000
ΔB × ΔC	1	6.400	6.400	6.4000	16.26	0.000
ΔB × ΔR	1	62.500	62.500	62.5000	158.81	0.000
ΔC × ΔR	1	19.600	19.600	19.6000	49.80	0.000
Residual	Error	90	35.420	35.420	0.3936	
Lack-of-Fit	5	2.753	2.753	0.5507	1.43	0.221
Pure	Error	85	32.667	32.667	0.3843	
Total	99	147.000				

Since the values of ΔB and ΔC are determined at each node as parameters with known values, T would be a single variable function based on ΔR. The desired value of the function T is the minimum value (i.e., 0). Therefore:

$$\frac{dT}{d\Delta R} = 1.2\Delta R - 0.8 - 1.25\Delta B - 0.7\Delta C = 0 \tag{18}$$

Thus:

$$\Delta R = \frac{0.8 + 1.25\Delta B + 0.7\Delta C}{1.2} \tag{19}$$

Evidently, this operation runs offline on each node with no effect on computing overhead in WSNs.

The data transmission rate of two neighbor nodes, which is determined by Eq. 19 is calculated so as to increase the accuracy of the estimated data transmission rate between each two neighbor nodes.

From Eq. 20, it can be observed that the second derivative of the equation is positive. As a result, T is a convex function and has an absolute minimum. Thus, delay gets zero in some points.

$$\frac{d^2T}{d\Delta R} = 1.2 \succ 0. \tag{20}$$

5 Creating the routing algorithm

This section discusses the implementation of our proposed CNCC algorithm and outlines the procedures for achieving the utilization field in detail. To select the best neighbor as the next destination, CCNC needs the depth value and traffic information from neighbor nodes. However, the excessive update messages impose overhead on the network. In order to minimize this overhead, a novel signaling mechanism which significantly reduces overhead is presented. Each node broadcasts an awareness packet (AP) to all neighbors to inform them about its depth and traffic. We take it for granted that all nodes are homogeneous in the network and have the same buffer size.

At first, the depth of the sink is initialized at 0. The sink sends the AP packet containing the hop count to all its neighbors. Each node compares AP messages received from all neighbors and calculates its own depth by adding 1 to the minimum depth value in the AP packets of neighbors. Obviously, if a node discovers the topological changes, it will recalculate the depth field. The simplest manner is to add 1 to the minimum depth value being in the routing table. Each node constructs its distance and traffic cost fields and then broadcasts this information to give advice to neighbors about its distance and traffic situations. The hop count field of the packet also increases by 1 since it is relayed by a node to show the number of hops over which the packet flows from the source to the sink.

Clearly, the depth field does not need fast updating. However, due to extensive changes in the traffic field, it should be updated in a triggered method. In our implementation, to avoid sending additional AP messages, CNCC defines two factors between two successive updated messages: *maximum interval time* (MIT) and *least interval time* (LIT). Each node should deliver updated messages (APs) between the MIT and LIT. MIT is used to preserve the connectivity of the network while LIT avoids sending excessive update messages. When there are more than two MITs elapsed since the reception of the last AP from a neighbor, this neighbor is considered dead and will be removed from the local decision matrix. If the elapsed time since the transmission of the last update message exceeds the MIT, the node will immediately send a new one, regardless of whether the depth or the traffic information factors have changed or not. MIT is a relatively large time span, used to preserve the connectivity of the network. When the depth of a node alters or traffic fields value changes exceed a specified threshold and meanwhile the elapsed time exceeds an LIT since the last successful AP message, the node will broadcast a new AP message to inform neighbors about its situation. In summary, as for a large MIT, CCNC algorithm can efficiently reduce the overhead. Additionally, each node has only its direct neighbor's information which further decreases the cost. The pseudo code used to make CCNC algorithm is displayed in Fig. 3. In this paper, following a series of experiments, the optimized values of periods MIT and LIT are taken as 10 and 0.2 s, respectively.

6 Performance evaluation

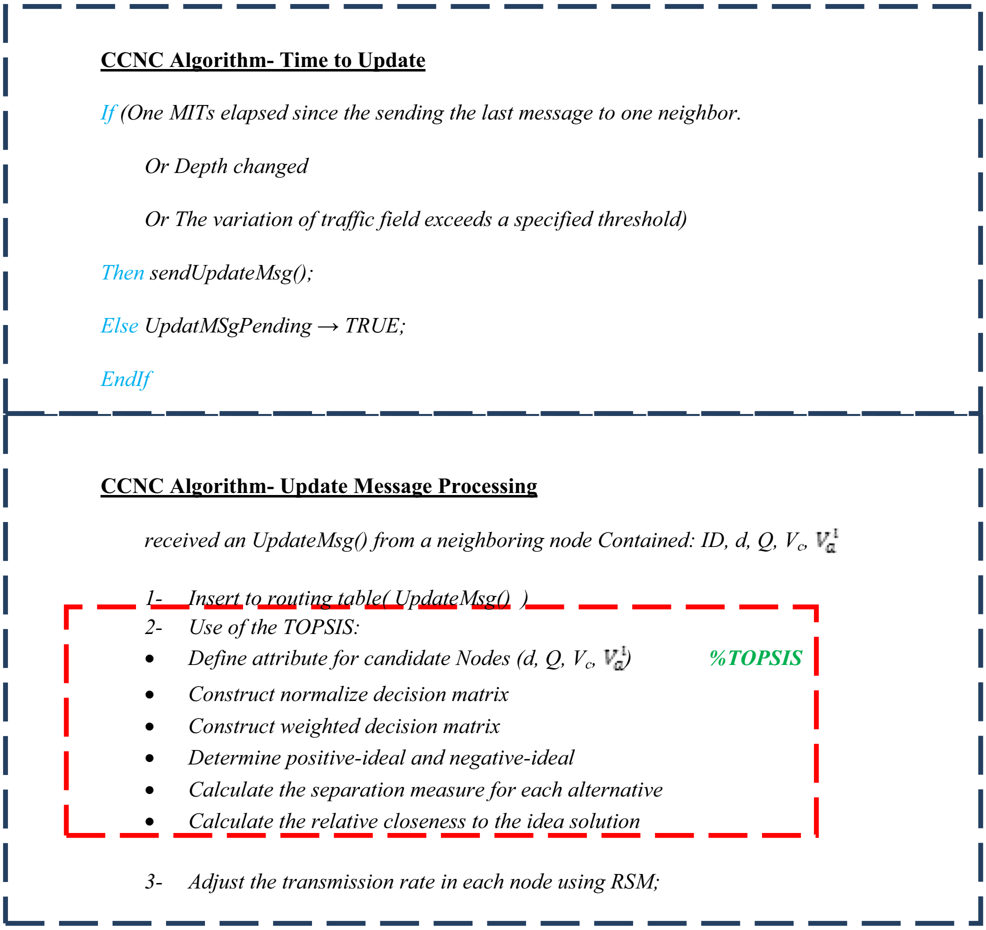
To evaluate the performance of the proposed method, we implemented 5 congestion control schemes, including the CODA [5], DRR [6], GRATA [9], GLOBAL [8], and hop by hop dynamic method [10] in terms of receiving packet rate, energy efficiency, and network throughput using NS2 Simulator [22]. In addition, the influence of various values of traffic factors is also considered. This section first describes the simulator configuration used in the study followed by a discussion of the simulation results based on the hypotheses.

6.1 Setting and configuration

It is assumed that the simulation is carried out on 100 homogeneous sensor nodes (including source nodes and sink node) with the sink at the upper right place. These nodes are randomly distributed in a square region of 100 m × 100 m. The source nodes and sink are taken to be unchangeable. IEEE 802.15.4 Standard protocol is used in physical and MAC layers. It should be mentioned that there are other protocols such as TSMA and MACA for simulation of physical layers, yet, IEEE 802.15.4 protocol is selected as it deals effectively with hidden node and exposed terminal problem [23]. Furthermore, IEEE 802.15.4 is used to investigate a low data rate solution with multi-month or multi-year battery life and very low complexity.

The communication mode for the protocol is assumed to be beacon-less. This is because the proposed method uses MIT and LIT packets to update the neighbors regarding its own status quo (routing table). Since an overload has been imposed on the network due to the use of beacon frame which leads to an increase in energy consumption of the nodes, the communications are taken as beaconless.

Each burst occurs between 120 and 160 s. There are two bursts for each event which have been shown in blue in Fig. 3. Each node maintains traffic information at its neighbors by awareness packets ($\mathrm{AP_s}$) periodically and using proposed algorithm to forward packets over optimal paths with an optimum rate toward the sink. Through several simulation rounds under high traffic condition as well as the use of Genetic



```
CCNC Algorithm- Time to Update

If (One MITs elapsed since the sending the last message to one neighbor.
    Or Depth changed
    Or The variation of traffic field exceeds a specified threshold)
Then sendUpdateMsg();
Else UpdatMSgPending → TRUE;
EndIf

CCNC Algorithm- Update Message Processing

    received an UpdateMsg() from a neighboring node Contained: ID, d, Q, $V_c$, $V_\alpha^1$
    1-  Insert to routing table( UpdateMsg() )
    2-  Use of the TOPSIS:
    •   Define attribute for candidate Nodes (d, Q, $V_c$, $V_\alpha^1$)        %TOPSIS
    •   Construct normalize decision matrix
    •   Construct weighted decision matrix
    •   Determine positive-ideal and negative-ideal
    •   Calculate the separation measure for each alternative
    •   Calculate the relative closeness to the idea solution

    3-  Adjust the transmission rate in each node using RSM;
```

Fig. 3 The pseudo code for creating the routing algorithm

algorithm, the TOPSIS weighted factors are calculated as $w_1 = 0.3$, $w_2 = 0.4$, $w_3 = 0.2$, and $w_4 = 0.1$. We used uniform mutation and uniform crossover [23] with 0.6 probabilities of cross over and 0.25 probabilities of mutation as well as the population of 50 chromosomes. A summary of the simulation parameters is displayed in Table 3. The simulation results are compared to CODA, DRR, GRATA, GLOBAL, and hop by hop dynamic method in order to demonstrate the improvement in the network performance with varied traffic rates. The amount of energy consumption in three states, namely, transmission, receive, idle are 1.3, 0.9, and 0.74 J, respectively

6.2 Network throughput

Network problems such as channel congestion and overflowed buffers can cause packet loss. The network throughput is a ratio between the number of packets received by the sink and the number of packets sent by the source nodes. Packet loss leads to the reduction in the network throughput. It can be caused by congestion and overflowed buffers in the nodes due to routing schemes.

CNCC has the ability to successfully smooth the bursts and adjust the transmission rate. Therefore, this method improves the receiving packet rate. Figure 3 also depicts the rate at which the sink received packets in the proposed scheme compared with the 5 other congestion control methods. The graph indicates that in most cases, CNCC scheme outperforms the others because of its ability to predict network conditions at the next hops to preempt potential congestions, select the best neighbor as the next destination, and prevent packets from entering into the congested area.

Practically, CODA and DDR method dropped most of the burst packets, while dynamic routing methods (GRATA, GLOBAL and hop-by-hop dynamic method) performed much better because they could pass packets via idle or unloaded nodes. Obviously, when traffic load increases, a larger number of packets enter into the network. As a result, there is a possibility of congestion, which drops the packets. The proposed method improves the network throughput by spreading traffic over under-loaded paths as much as possible and it adjusts the transmission rate, especially in cases of high traffic. In non- bursting time, transmission rate methods received few packets at the sink, but dynamic routing

Table 3 Simulation parameters

Simulator name	NS-2 Version
Physical layer protocol	IEEE.802.15.4
Beacon mode	Disable
Radio type	Radio-accnoise (standard radio model)
Node placement	Random
Application type	Event driven
Frequency band	2.4 Ghz
Link layer transmission rate	250 kbps
Range of channel connection window size	[1,63]
Maximum CSMA/CA attempts	4
Simulation time	400 s
Area size	100 m × 100 m
Network architecture	Homogeneous, flat
Number of nodes	100
Largest depth	20
Average node degree	6
Sink coordinate	(15,15) m
Communication radius	5 m
Packet size	25 byte
Buffer size	10 packets
MUI period	10 s
LUI period	0.2 s
Transmission energy	1.3 J
Receive energy	0.9 J
Idle energy	0.74 J

methods continued to receive more packets. The superiority of proposed method compared to hop by hop dynamic method is due to the former's objective function modeled with suitable methods (TOPSIS and RSM).

As Fig. 4 shows, the better throughput of the proposed method is more highlighted in terms of receiving packet rate at the time of two bursts (intervals 120—160 s, 240–280 s). This superior performance can be attributed to bypassing the traffic and the adjusted transmission rate at the time of congestion presented by the new method. Besides, as the proposed method involves the transmission of some packets through the longer paths on their way to the sink (see Fig. 4), the receiving transmission rate will be high in the sink for a while after the burst. For example, the receiving rate decreases at 200th second while the burst ends at 160th second.

Previous experiments demonstrated the ability of our method to receive maximum packets in the sink in burst and heavy traffic environment. Now, we configure the light load environment to investigate the compatibility of our method to various loading environments. Figure 5 shows the number of received packets in the light loading network. As the figure shows, the receiving rate of the packets using CNCC is displayed in blue and the receiving rate of packets in other methods are displayed in different colors. As Fig. 5 shows, the packet receiving rate is appropriate in all methods, with the rate being near the rate of sending. That is, when the traffic load is low the rate of lost packets is low in the case of all methods as there is no congestion hence no loss of packets. Yet, the proposed method has advantages over the other methods. This is because the traffic around the sink is relatively heavy as all nodes transmit the packets to the sink. As already discussed, the proposed method can cope with the traffic more effectively than other methods.

6.3 Energy efficiency evaluation

The average amount of energy balance in the nodes, fairness, and life time are three important factors used to evaluate the energy efficiency in the network. Table 4 represents the average amount of energy balance in different methods. The average energy balance is the ratio of remained energy in all nodes after the completed simulation to the total amount of primary energy of nodes. As you can see, the proposed method has a few advantages compared to other methods. The resending of the packets is reduced as the packets bypass the traffic. This leads to the decreased loss of packets as well as the reduced amount of energy used for resending the packets. However, bypassing congested nodes will have the packets go through longer paths with an increase in the number of relay nodes, leading to an increase in energy consumption. In addition, the use of TOPSIS and RSM methods in each node leads to increased energy consumption in the nodes. Because of this tradeoff, the average energy efficiency in the nodes is slightly higher in the proposed method compared to other methods.

It is worth mentioning that the average energy balance is not a thorough indicator of the level of energy efficiency. Due to the unbalanced energy consumption, the total amount of energy in a node may have been used while the energy in another node may have not been used at all. Obviously, a better balanced consumption of energy will lead to longer lifetime. Fairness parameter can be calculated, using Eq. 21 which calculates the variance of normalized energy balance in network nodes to the energy balance throughout the network (at the worst, the half of the nodes have used up energy while the other half have not used energy at all). In Eq. 21, energy represents the energy balance in node i at the end of the simulation. It can be concluded from this formula that the higher value of fairness parameter in network leads to more balanced energy consumption in nodes, with the energy balance in all nodes being near to each other. Obviously, if fairness parameter equals 1, the network has the best balanced energy consumption.

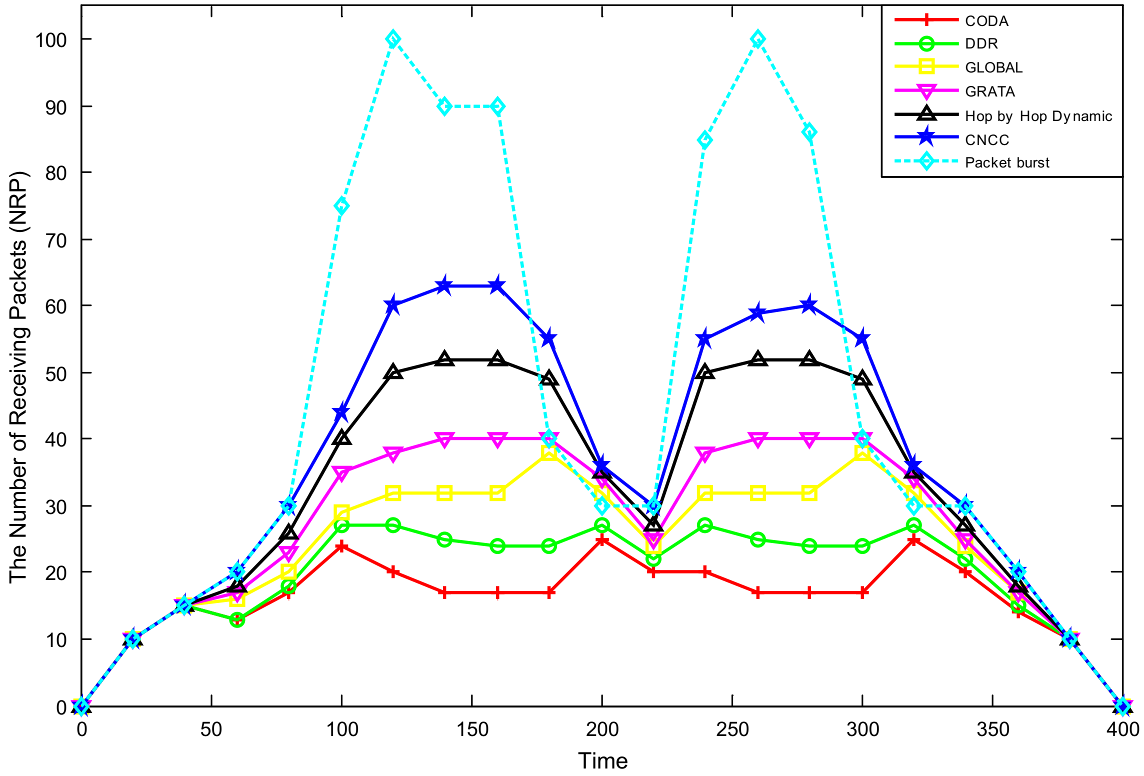


Fig. 4 Received packets to all sent packets ratio in different time intervals

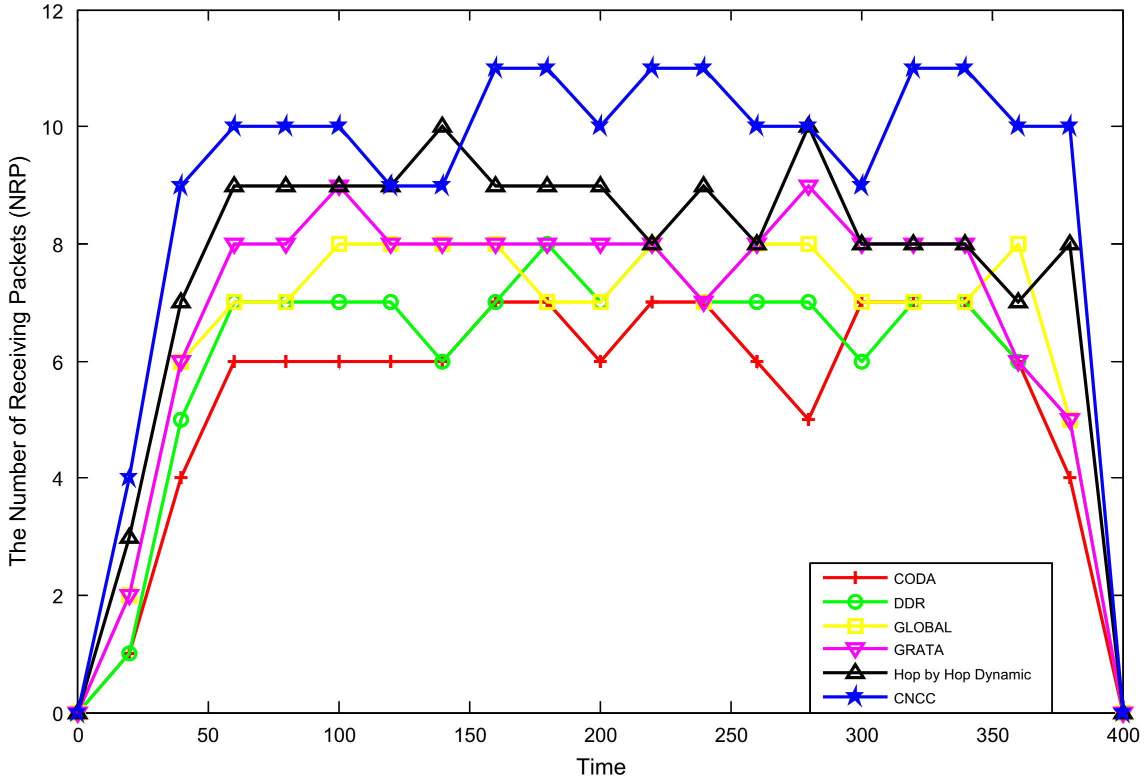


Fig. 5 Average packet receiving to send packets ratio over time in the light congested area

Table 4 The average amount of energy balance in the nodes

Method	CODA	DDR	GRATA	GLOBAL	Hop by Hop	CCNC
Percentage	37.7	37.2	33.5	36.4	37.3	42.5

Table 5 The average values of fairness parameter in different methods

Method	CODA	DDR	GRATA	GLOBAL	Hop by Hop	CCNC
Percentage	49.7	52.1	71.3	73.6	81.6	90.8

$$DEV = \sum_{i=1}^{n} (energy_i - Ave)^2$$
$$Fairness = \frac{1 - DEV}{DEV_{worst}} \tag{21}$$

Table 5 displays the average values of fairness parameter using different methods. The fairness parameter in the proposed method is considerably better than others. In DRR and CODA methods, limited numbers of nodes are involved in the transmission of packets as the packet paths are fixed. This makes some nodes idle when it comes to packet transmission, resulting in a decreased value of fairness parameter in these nodes. In the case of dynamic methods such as GRATA, GLOBAL and hop by hop method, a larger number of nodes are involved in the packets transmission, leading to an increase in value of fairness parameter. However, balanced energy consumption is archived and the value of fairness parameter is significantly higher, using the proposed method compared to other methods. This superiority is due to dynamic routing, using appropriate criteria to evaluate the traffic load of the neighbors as well as the appropriate selection of receiving node.

Network lifetime represents the duration in which the algorithm is run until the first node dies. Clearly, the higher the value of the fairness parameter, the longer the network lifetime. As Table 6 shows, CNCC involves the longest network lifetime compared to other methods. This is because in the proposed method the packets are sent using multiple paths. In fact, this method ensures that the packets are transmitted toward the sink in fair distribution of traffic, while CODA and DRR use single path to transmit packets. Moreover, the proposed method has a longer lifetime compared to GRATA, GLOBAL and hop by hop method due to the appropriate distribution of packets in the nodes.

Given the above results, it can be concluded that although in the proposed method, the average energy balance in nodes is not considerably better than other methods; it yields significantly better results when it comes to fairness parameter, lifetime and the number of lost packets. These features make the method distinct from other methods.

6.4 Normalized buffer size and end to end delay

Figure 6 shows the normalized buffer size $Q(v)$ over time for the nodes engaged in the packets transfer. The mean size of normalized buffer is a tool used to measure the delay. The higher normalized buffer leads to an increase in delay. This is because when the normalized buffer size increases, the collision and overflow of packets are more likely to happen, leading to resending and the delay in the transfer of the packets. As Fig. 6 shows, all methods involve a bigger normalized buffer size at burst time (interval of 120–160 and 240–280 s) compared to other times. In the case of adjusted transmission rate methods, the amount of delay is not appropriate, in particular, at burst time. This is because these methods use fixed routes to send the packets, with the normalized buffer size increasing at the time of traffic due to the arrival of too many packets at congested nodes. The mean amount of normalized buffer size in the proposed methods is in blue, showing better results compared to other methods. This good performance is mainly due to the following factors: bypassing the traffic, the participation of low load and idle nodes in packet transmission, and the appropriate adjustment of transfer rate.

The average end to end delay is another essential parameter evaluated in CNCC. End to end delay is defined as the time needed to send a packet from the source to the sink, including the transfer duration, the processing in relay nodes, and the resending of the packets as they may be lost. Delay parameter is considered as one of the essential issues in real-time applications. Figure 7 shows that the amount of delay using CNCC is the least compared to other methods. Although, in rate based methods the packets may be sent to the sink through the shorter path compared to CNCC, the latter bypasses the traffic, leading to a decrease in lost packets and hence decreased resending of packets. These results lead to less delay compared to rate methods. Compared to dynamic routing-based methods, CNCC results in the less amount of delay as the latter performs more accurate evaluation to determine the next destination of the packets. Moreover, CNCC has the capability to adjust the rate of transfer, leading to the selection of the best path as well as a considerable decrease in lost packets. This, in turn, leads to decreased delay.

6.5 Effectiveness of data transmission rate

Figure 8 compares the effect of RSM rate adjusted strategy and that of RSM non-adjusted strategy on the rate of reception of packets. The cyan line indicates the sending rate of the packets from the origin in various simulation times. Blue and gray lines show the rate of packet reception in the sink using the proposed method with regulated reception rate and unregulated rate, respectively.

The graph indicates that congestion control with data transmission rate outperforms the others because of its ability

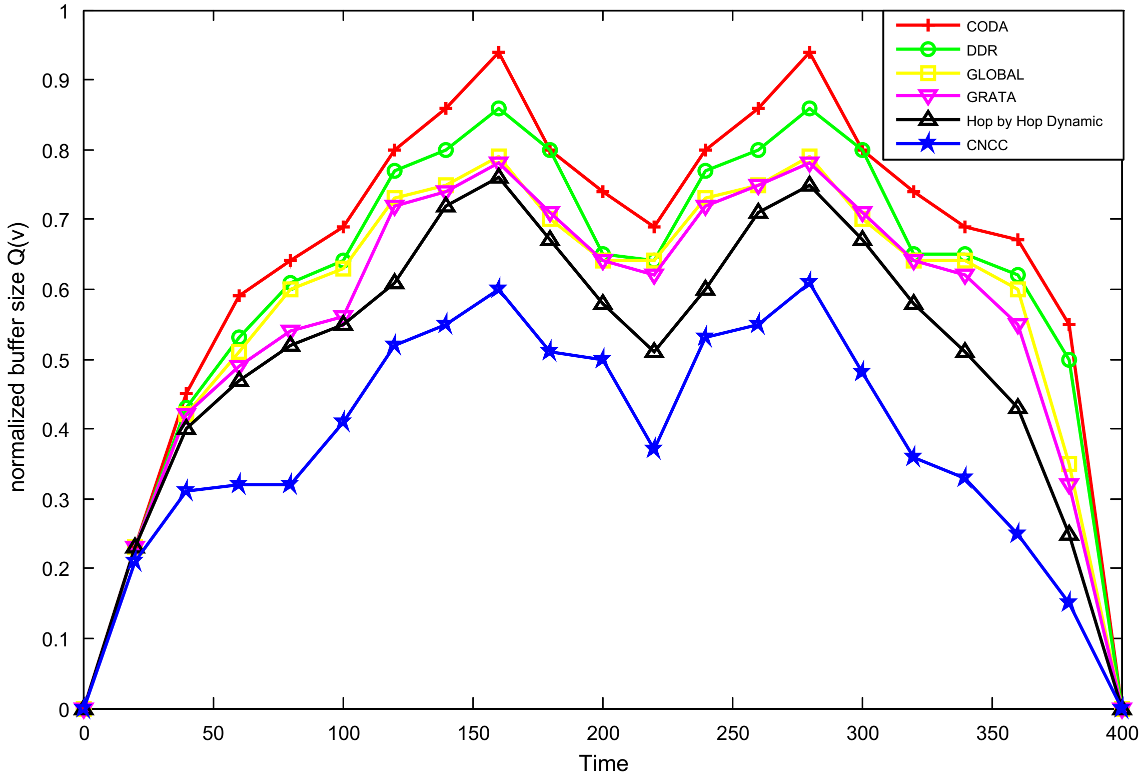


Fig. 6 The normalized buffer size $Q(v)$ over time

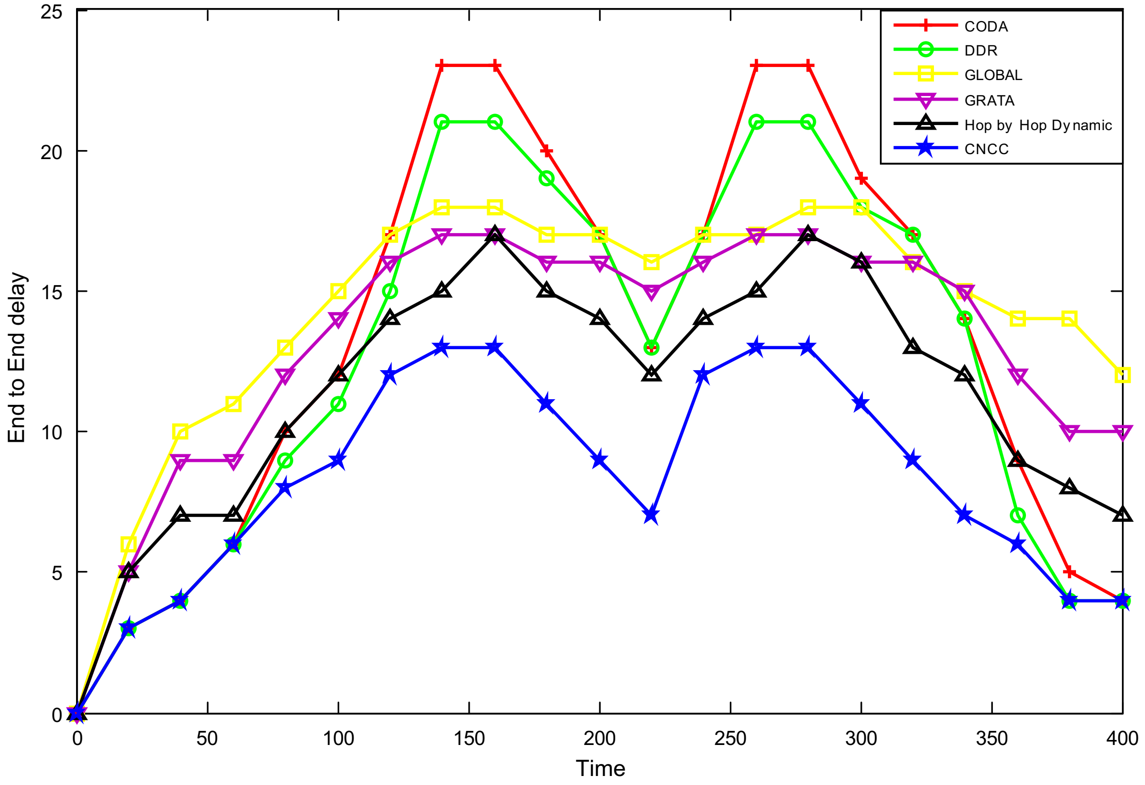


Fig. 7 The average end to end delay

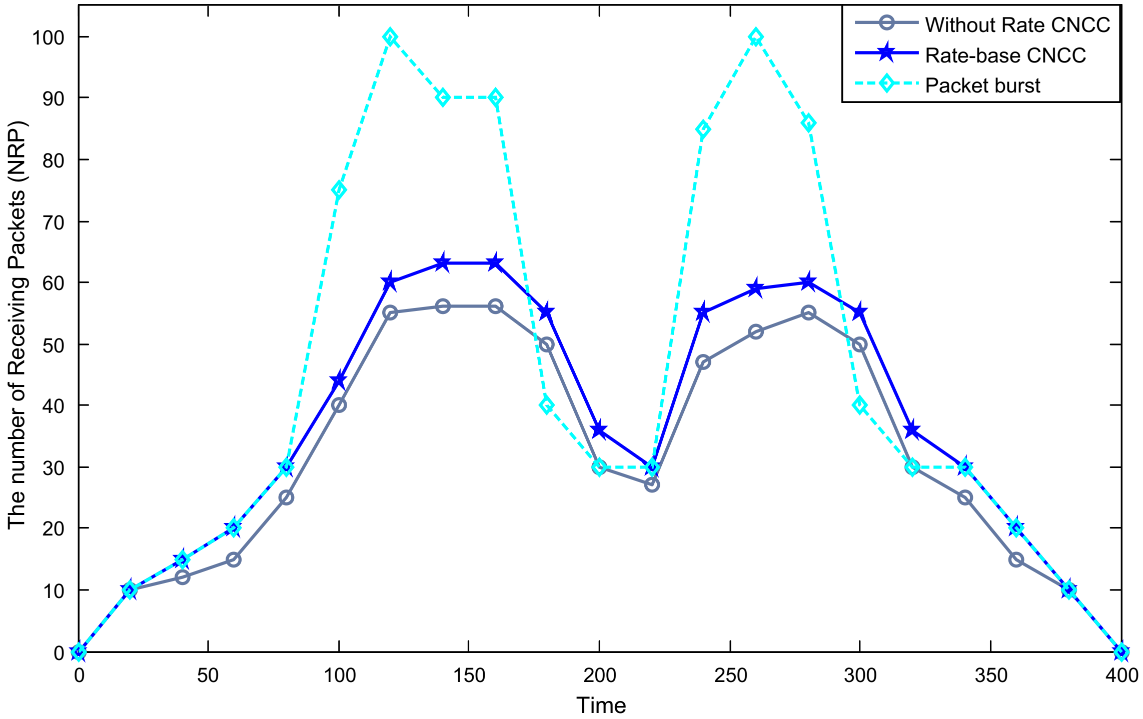


Fig. 8 The influence of transmission rate adjustment on the average end-to-end delay

to decrease or increase data transmission rate in congested or unloaded places. Put it another way, regulating the sending rate in the nodes leads to better balanced load distribution in the network nodes. This may decrease the likelihood of buffer engagement, resulting in an increase in the rate of reception of packets in the sink.

7 Discussion

Comparing the averages of the same factor in different groups is one of the most important concerns in research. In this context, the averages are analyses, using ANOVA. After determining whether or not there is difference among the groups (the investigation of significance based on ANOVA table), it is necessary to determine which two groups are significantly different. Tukey [25] proposed a multiple comparison test based on t-student. Based on this test, when the absolute value of the test statistic is greater than the critical value, the difference is significant.

A comparison was made between the results of CNCC and the results obtained by five other methods, using a statistical test. The Null Hypothesis proposes the equal averages of the results obtained by different methods, whereas Alterative Hypothesis proposes the significant difference between the results obtained by different methods. Given that in this study the comparison was made between more than two methods, Tukey was used to compare the results of methods in pairs.

Table 6 The average values of network lifetime in different methods

Method	CODA	DDR	GRATA	GLOBAL	Hop by Hop	CCNC
Second	115	125	164	171	192	241

We used Tukey test to compare algorithms performance. Tukey test is a more general form of *t* test used to compare the rate of reception of packets in various algorithms. In variance analysis test, the independent variable or grouping variable have more than two levels, with dependent (output) variable being a quantitative one. This test is used to examine the difference between means of multiple groups (more than 2 groups or inputs).

Hypothesis test presented in Eq. 22. is used to compare these algorithms in terms of the number of packets received by the sink. If the ANOVA test is significant, then the Tukey test is used for algorithms grouping. ANOVA test only shows the difference between the means. It doesn't indicate which group has a mean different from other groups. Using the Tukey test, these differences can be identified.

$$\begin{cases} H_0 : \mu_{CNCC} = \mu_{Hop-by-hop} = \mu_{CODA} = \mu_{GLOBAL} = \mu_{GRATA} \\ H_1 : Otherwise \end{cases} \tag{22}$$

As ANOVA Table 7 shows, there is a significant difference among the methods. As a result, the H_0 hypothesis is rejected.

Table 7 One-way ANOVA to compare methods

Source	DF	SS	MS	F	*P* value	Result
Factor	5	53,186.4	10,637.3	8479.49	0.000	Rejected
Error	240	301.1	1.3			
Total	245	53,487.5				

Table 8 Grouping information using Tukey test

Algorithm	N	Mean	SD	Grouping	Rank
CODA	41	19.024	1.107	A	1
DDR	41	25.829	0.863	B	2
GLOBAL	41	33.122	1.327	C	3
GRATA	41	39.732	1.025	D	4
Hop-by-hop Dynamic	41	50.951	0.773	E	5
CNCC	41	62.415	1.466	F	6

To have a better analysis, Tukey test is used in the burst times (simulation 120 and 160 s).The Tukey-test results are shown in Table 8.

Table 8 represents algorithms grouping where CNCC belongs to group F. Hop-by-hop Dynamic, GRATA, GLOBAL, DDR and CODA belong to group E, D,C,B and A, respectively. It means that, difference between means of the number of packets received by the sink in all methods is significant.

As the Fig. 9 shows, there is a significant difference between the number of packets received by the sink at bursting time (120–160 s) obtained by the proposed method and those obtained by other methods. These results prove that CNCC algorithm is preferred at a confidence level of 95%. Now by checking the Box-Plots of algorithms, presented in Fig. 10. It is clear that CNCC works better than other algorithms.

8 Conclusions

To sum up, we presented a novel distributed congestion control algorithm in wireless sensor networks that avoids congestion while increases the throughput, energy efficiency and the number of packets delivered to the sink node. Additionally, this approach decreases the number of collisions at

```
                                  Individual 95% CIs For Mean Based on Pooled StDev
Level                                   ----+---------+---------+---------+-----
CODA                                    *
DDR                                          (*
GLOBAL                                             (*
GRATA                                                   *
Hop-by-hop Dynamic                                               *)
CNCC                                                                      *
                                        ----+---------+---------+---------+-----
                                           24        36        48        60
```

Fig. 9 Individual 95% CIs for mean based on pooled SD

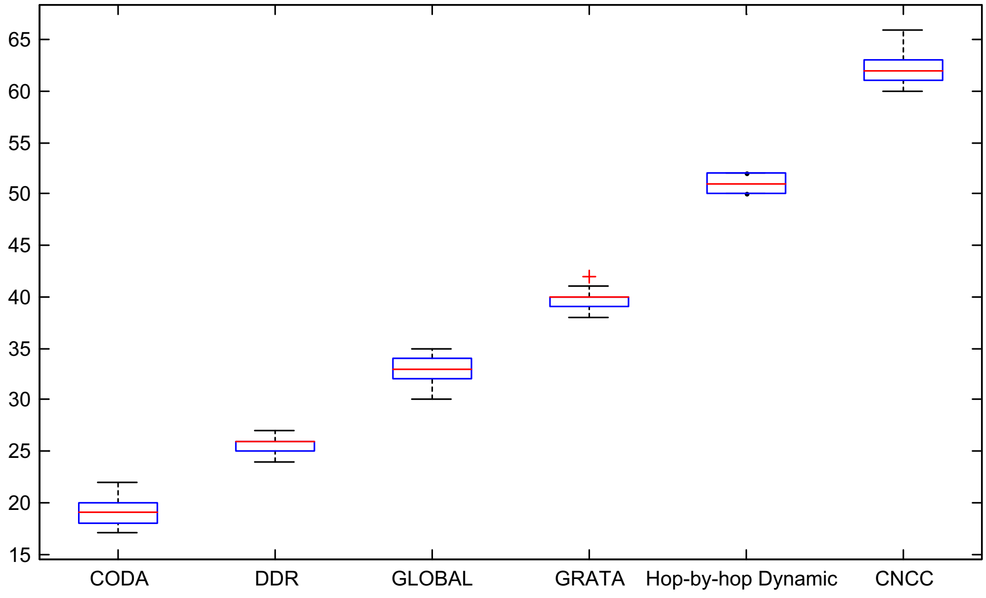


Fig. 10 *Box-plot* for evaluating statistical results graphically

intermediate nodes. At first, we used the depth field and traffic information such as the queue length field, congestion degree, and cumulative queue length to make routing decisions. The depth field provides the basic routing backbone which routes the packets directly to the sink along the shortest path. The traffic information fields make our algorithm traffic-aware. TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) method was used to perform competitive benchmarking of all neighbors and select the optimal neighbor as the next destination. TOPSIS considers all our routing metrics and selects the nodes with the shortest distance from the positive ideal solution and also the farthest from the negative ideal solution as the next relay node. Meanwhile, we adopt the data transmission rates of sensor nodes to adjust the transmission rate among sensor nodes. We presented a new equation by using Response Surface Methodology (RSM). This equation can achieve data transmission rates by using the buffer occupancy and the congestion degree at each node.

References

1. Abdul-Salaam, G., et al. (2016). A comparative analysis of energy conservation approaches in hybrid wireless sensor networks data collection protocols. *Telecommunication Systems*, *16.1*, 159–179.
2. Gholipour, M., & Meybodi, M. R. (2008). LA-mobicast: A learning automata based mobicast routing protocol for wireless sensor networks. *Sensor Letters*, *6*(2), 305–311.
3. Jingjing, W., et al. (2016). Aggressive congestion control mechanism for space systems. *IEEE Aerospace and Electronic Systems Magazine*, *31.3*, 28–33.
4. Kaur, J., Grewal, R., & Saini, K. S. (2015). A survey on recent congestion control schemes in wireless sensor network. *Advance computing conference (IACC), 2015 IEEE international*. IEEE, 2015.
5. Wan, C.-Y., Eisenman, S. B., & Campbell, A. T. (2003). CODA: congestion detection and avoidance in sensor networks. *Proceedings of the 1st international conference on embedded networked sensor systems* (pp. 266–279). ACM.
6. Deshpande, V. S. et al. (2012). Control, congestion, in wireless sensor networks by using differed reporting rate. *Information and communication technologies (WICT), World Congress on IEEE* (pp. 209–213).
7. Gholipour, M., Haghighat, A. T., & Meybodi, M. R. (2017). Hop-by-Hop congestion avoidance in wireless sensor networks based on genetic support vector machine. *Neurocomputing*, *223*, 63–76.
8. Yoo, H., et al. (2010). GLOBAL: A gradient-based routing protocol for load-balancing in large-scale wireless sensor networks with multiple sinks. *Computers and communications (ISCC), 2010 symposium on IEEE* (pp. 556–562). IEEE.
9. Tan, D. D., Dinh, N. Q., & Kim, D.-S. (2013). GRATA: Gradient-based traffic-aware routing for wireless sensor networks. *Wireless Sensor Systems, IET*, *3.2*, 104–111.
10. Majid, G., Haghighat, A. T., & Meybodi, M. R. (2015). Hop-by-hop traffic-aware routing to congestion control in wireless sensor networks. *EURASIP Journal on Wireless Communications and Networking*, *2015*(1), 1.
11. Pussente, R. M., & Barbosa, V. C. (2009). An algorithm for clock synchronization with the gradient property in sensor networks. *Journal of Parallel and Distributed Computing*, *69*(3), 261–265.
12. Thomas, R. W., DaSilva, L, A., MacKenzie, Allen B. (2005). Cognitive networks. New frontiers in dynamic spectrum access networks. DySPAN 2005. *2005 First IEEE international symposium on IEEE* (pp. 5–12).
13. Mitola III, J. (2006). Cognitive radio architecture. *Cooperation in wireless networks: Principles and applications* (pp. 243–311). Dordrecht: Springer.
14. Shankar, P. M. (2016). Performance of cognitive radio in N* Nakagami cascaded channels. *Wireless Personal Communications*, *88*, 657–667.
15. Srivastava, V., & Motani, M. (2005). Cross-layer design: a survey and the road ahead. *Communications Magazine, IEEE*, *43*(12), 112–119.
16. Clark, D. D., et al. (2003). A knowledge plane for the internet. In *Proceedings of the 2003 conference on applications, technologies, architectures, and protocols for computer communications* (pp. 3–10). ACM.
17. Zavadskas, E. K., Turskis, Z., & Kildienė, S. (2014). State of art surveys of overviews on MCDM/MADM methods. *Technological and Economic Development of Economy*, *20*(1), 165–179.
18. Kleijnen, J. P. C. (2015). *Response surface methodology. Handbook of simulation optimization*. New York: Springer.
19. Georgiou, S. D., Stylianou, S., & Aggarwal, M. (2014). A class of composite designs for response surface methodology. *Computational Statistics & Data Analysis*, *71*, 1124–1133.
20. Vibha, G., & Anuja, S. (2011). *Identifying an appropriate requirements prioritization methodology using fuzzy decision-making. computer networks and intelligent computing*. Berlin, Heidelberg: Springer.
21. Stat-Ease, Inc. Design Expert software, Educational Version 7.0. 3. (2007).
22. Issariyakul, T., & Hossain, E. (2011). *Introduction to network simulator NS2*. New York: Springer.
23. Yazdi, E. T., Willig, A., & Pawlikowski, K. (2014). Frequency adaptation for interference mitigation in IEEE 802.15. 4-based mobile body sensor networks. *Computer Communications*, *53*, 102–119.
24. Spector, L., & Helmuth, T. (2014). *Uniform linear transformation with repair and alteration in genetic programming. Genetic Programming Theory and Practice XI*. New York: Springer.
25. Tukey, J. W. (1949). Comparing individual means in the analysis of variance. *Biometrics*, *9*, 99–114.

M. Gholipour completed his B.S. and M.S. studies in computer engineering in Iran, dated 2002 and 2005 respectively. He also achieved a Ph.D. degree in computer engineering from Islamic Azad University (Tehran Science and Research branch) in 2016. He joined the faculty of computer engineering department at Islamic Azad University (Qazvin Branch) in 2006. His research interests include Wireless Networks, Cognitive Networks, Internet of things (IOT), Learning Systems, Performance Evaluation and Fuzzy Systems.

A. T. Haghighat received his B.S. and M.S. degrees in Electrical Engineering from Sharif University of Technology, Tehran, Iran, in 1993 and 1996, respectively. His Ph.D. degree from Amirkabir University of Technology (AUT), Tehran, Iran, in 2003. He is an Assistant Professor of Computer Engineering and Information Technology at Islamic Azad University of Qazvin. His research interests are in wireless networks, pattern recognition, fault-tolerant computing, and distributed systems.

M. R. Meybodi received B.S. and M.S. degrees in Economics from Shahid Beheshti University in Iran in 1973 and 1977, respectively. He also received M.S. and Ph.D. degrees in Computer Science from Oklahoma University, USA, in 1980 and 1983, respectively. Currently, he is Full Professor in the Computer Engineering Department of Amirkabir University of Technology, in Tehran, Iran. Prior to his current position, he worked from 1983 to 1985 as Assistant Professor at Western Michigan University, and from 1985 to 1991 as Associate Professor at Ohio University, USA. His research interests include Channel Management in Cellular Networks, Learning Systems, Parallel Algorithms, Soft Computing and Software Development.