Optimization Based Hybrid Congestion Alleviation for 6LoWPAN Networks

Hayder A. A. Al-Kashoash, Hayder M. Amer, Lyudmila Mihaylova, *Senior Member, IEEE,* and Andrew H. Kemp, *Senior Member, IEEE*

***Abstract*— The IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) protocol stack is a key part of the Internet of Things (IoT) where the 6LoWPAN motes will account for the majority of the IoT 'things'. In 6LoWPAN networks, heavy network traffic causes congestion which significantly effects the network performance and the quality of service (QoS) metrics. Generally, two main strategies are used to control and alleviate congestion in 6LoWPAN networks: resource control and traffic control. All the existing work of congestion control in 6LoWPAN networks use one of these. In this paper, we propose a novel congestion control algorithm called optimization based hybrid congestion alleviation (OHCA) which combines both strategies into a hybrid solution. OHCA utilizes the positive aspects of each strategy and efficiently uses the network resources. The proposed algorithm uses a multi-attribute optimization methodology called grey relational analysis for resource control by combining three routing metrics (buffer occupancy, expected transmission count and queuing delay) and forwarding packets through non-congested parents. Also, OHCA uses optimization theory and Network Utility Maximization (NUM) framework to achieve traffic control when the non-congested parent is not available where the optimal nodes' sending rate are computed by using Lagrange multipliers and KKT conditions. The proposed algorithm is aware of node priorities and application priorities to support the IoT application requirements where the applications' sending rate allocation is modelled as a constrained optimization problem. OHCA has been tested and evaluated through simulation by using Contiki OS and compared with comparative algorithms. Simulation results show that OHCA improves performance in the presence of congestion by an overall average of 28.36%, 28.02%, 48.07%, 31.97% and 90.35% in terms of throughput, weighted fairness index, end-to-end delay, energy consumption and buffer dropped packets as compared to DCCC6 and QU-RPL.**

***Index Terms*— Congestion alleviation, hybrid solution, multi attribute decision making, optimization theory, 6LoWPAN networks, IoT applications.**

I. Introduction

The Internet of Things (IoT) is considered to be the next big challenge for the Internet research community and it has recently drawn significant research attention [1]. The IoT will comprise billions of intelligent communicating devices which extend the border of the world with physical entities and virtual components [2]. These things, such as wireless sensor nodes, radio frequency identification (RFID) tags and near field communication (NFC) devices, are connected to the Internet with the ability to sense status and condition. Also, they access historical data and developed algorithms, possibly triggering devices. This is leading to very powerful smart environments e.g. building, health care, etc. [1].

Wireless sensor networks (WSNs) are considered as one of the most important elements in the IoT [3]. IPv6 over Low Power Wireless Personal Area Network (6LoWPAN) [4] is used for full integration of WSN with the Internet where sensor nodes implement the Internet Protocol (IP) stack though it was originally designed for wired networks. However, the implementation of the TCP/IP model in WSN and 6LoWPAN networks has many issues and problems due to the limitation of bandwidth, energy and buffer resources. TCP (transmission control protocol) requires extra resources for connection setup and termination before and after the data transmission whilst UDP (user datagram protocol) does not provide a congestion control mechanism. Thus, TCP and UDP are not efficient for WSN and 6LoWPAN networks [1]. Therefore, one of the main issues in WSN and 6LoWPAN networks is congestion that causes packet loss, increased energy consumption and degraded throughput.

In general, two main methods are used to solve and alleviate congestion in WSNs and 6LoWPAN networks: rate adaptation (traffic control) and traffic engineering i.e. selection of an alternate non-congested path (resource control) to forward packets to destination nodes [5], [6]. In traffic control, the sending rate of the source node is reduced to a specific value such that the number of injected packets into the network is reduced and therefore; congestion is alleviated. However, for time critical and delay constrained application (e.g. medical applications and fire detection applications), reducing the data rate is not desirable and impractical. In the resource control method, packets are forwarded to destination node through alternative non-congested paths without adjusting the sending rate. However, sometimes non-congested paths are not available and therefore; congestion can not be avoided. Thus, it is very important to combine the above two strategies into a hybrid scheme and utilizing the positive aspects of using both traffic control and resource control. In such case, the resource control strategy is firstly used for searching non-congested paths. If they are not available, then the sending rate is reduced by applying the traffic control strategy. To the best

H. A. A. Al-Kashoash is with the Electronic and Electrical Engineering School, University of Leeds, Leeds LS2 9JT, U.K, and also with Technical Institute/Qurna, Southern Technical University, Basra, Iraq. (e-mail: ml14haak@leeds.ac.uk; hayderaam@gmail.com).

H. Amer and L. Mihaylova are with Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield S1 3JD, U.K. (e-mail:hmamer1@sheffield.ac.uk; l.s.mihaylova@sheffield.ac.uk).

A. H. Kemp is with Institute of Robotics, Autonomous Systems and Sensing (IRASS), Electronic and Electrical Engineering School, University of Leeds, Leeds LS2 9JT, U.K. (e-mail:a.h.kemp@leeds.ac.uk).

of our knowledge, no existing congestion control mechanism in 6LoWPAN networks combines both strategies to solve the congestion problem.

The RPL (IPv6 routing protocol for low-power and lossy networks) [7] is expected to be the standard routing protocol for 6LoWPAN networks and the IoT. In 6LoWPAN networks, RPL is responsible for constructing the network topology based on an objective function which combines one or more routing metrics into a Rank. Each node selects a neighbor as its parent with the best Rank. In case of congestion, the main challenge is that the node ranks the parents and paths from least to most congested and selects the best one when congestion occurs according to multiple routing metrics. Thus, the selection of a parent can be modelled as a multi-criteria decision problem which can be solved by using a Multi Attribute Decision Making (MADM) technique. MADM presents a suitable approach and promising solution for the parent selection problem within congestion. However, sometimes a non-congested parent is not available and applying the traffic control strategy is important to mitigate and alleviate congestion in the network. When congestion occurs, each node starts to send high data rate packets to its parent without considering the parents forwarding rate, the available bandwidth and other nodes' sending rate. Therefore, adapting and allocating the sending rate to each node subject to congestion alleviation is important. The nodes' sending rate adaptation can be modelled as a constrained optimization problem which can be solved by using optimization theory [8]. Optimization theory provides the necessary tools and techniques that can adjust node sending rate optimally and satisfactorily. However, none of the existing congestion control algorithms in WSNs and 6LoWPAN networks utilizes and uses MADM and optimization theory to mitigate congestion in the network.

This paper is motivated by these considerations to propose a novel congestion control algorithm called "**O**ptimization based **H**ybrid **C**ongestion **A**lleviation" (OHCA) which combines both traffic and resource control strategies into a hybrid solution to utilize the benefits of using both of them. Also, OHCA uses a multi-criteria optimization approach for selecting less congested parent and path to forward packets to the final destination as well as optimization theory for controlling and adapting nodes' sending rate when the non-congested parent is not available. Our main contributions in this paper include:

- Proposal of a new congestion alleviation algorithm called OHCA which provides a hybrid solution to the congestion problem in 6LoWPAN networks to use and utilize the network resources effectively. The proposed algorithm firstly applies the resource control strategy which searches for the non-congested path by utilizing a MADM technique. If the resource control method can not be applied, then the traffic control strategy is executed to reduce the number of injected packets into the network by using optimization theory. Thus, OHCA utilizes the advantages of both strategies by bridging these two methods for congestion control and providing the optimal solution.
- Model the selection of parents within congestion as a multi-criteria decision problem which can be solved by using the Gray Relational Analysis (GRA) method [9]. GRA ranks the parents from least to most congested and selects the best one by combing a set of routing metrics (attributes). In our proposal, we use three routing attributes: expected transmission count (ETX), buffer occupancy (BO) and queue delay (QD). Thus, the GRA approach is integrated with the RPL objective function to make our proposal compatible with the 6LoWPAN protocol stack. The weights of routing metrics are calculated by using the standard deviation method.
- In the IoT applications, sensor nodes host many application types simultaneously with different requirements. Some of them are real time applications where data is important and time critical, while others are non-real time applications. Therefore, it is important that a new proposed algorithm supports awareness of both node priorities and application priorities. Thus, our proposal (OHCA) is aware of node priorities and application priorities to support the IoT application requirements. We model the nodes' sending rate adaptation as a constrained optimization problem which can be solved using Network Utility Maximization (NUM) framework. The NUM was introduced by Kelly et al. [8] in 1998 for wired networks and it has already numerous applications in wired and wireless network optimization [10]. Here, we utilize the NUM framework in 6LoWPAN networks to allocate data rate to each node when congestion occurs where each node has a utility function. The node's utility function is modelled as a constrained nonlinear optimization problem which is solved by using Lagrange multipliers and KKT (Karush-Kuhn-Tucker) conditions such that each node obtains its optimal solution (i.e. sending rate) that satisfies the congestion alleviation.
- Implement and evaluate the performance of the proposed algorithm in the real IoT operating system, Contiki OS [11], through Cooja simulator [12].

The remainder of the paper is organized as follows: in section II, we provide a review of related work on congestion control in 6LoWPAN networks. Section III introduces the network setup and formulates the problem. Section IV introduces resource control strategy based on MADM. The traffic control strategy based on optimization theory and NUM framework is given in section V. The implementation of the hybrid congestion control algorithm in 6LoWPAN networks is provided in section VI. In section VII, simulation scenarios and results are given. Finally, section VIII draws conclusions.

II. Related Work

Many algorithms have been proposed in the congestion control literature for mitigating congestion in WSNs (see [5], [6], [13] and references therein). However, the majority of the existing literature do not take into account the unique characteristics of the IEEE 802.15.4 standard, IPv6 and 6LoWPAN protocol stack (i.e. RPL routing protocol, the adaptation layer and IEEE 802.15.4 MAC and PHY layers). Recently, a number of papers suggest new congestion control algorithms for 6LoWPAN networks. A short review of these mechanisms is given below. However, according to the best of our knowledge, none of the proposed congestion control algorithms in

6LoWPAN networks combines and utilizes both traffic and resource control strategies to solve the congestion problem. Also, none of the existing algorithms in congestion control literature for WSNs and 6LoWPAN networks uses MADM and optimization theory to alleviate congestion. Moreover, our proposal (OHCA) is aware of both node priorities and application priorities to support the IoT application requirements where each node is assigned a priority based on its importance and hosted application types as well as each application is given a priority according to its type (i.e. real-time application or not, time-critical application or not, etc.).

In [14], Michopoulos et al. proposed a new congestion control algorithm called Duty Cycle-Aware Congestion Control for 6LoWPAN networks (DCCC6). The proposed algorithm detects the presence of radio duty cycle and adjusts its operation accordingly. The proposed protocol uses a dynamic buffer occupancy as a congestion detection method as well as a modified AIMD (Additive-Increase Multiplicative-Decrease) to reduce the congestion in the network. In [15], Castellani et al. proposed three different congestion control schemes called Griping, Deaf and Fuse for controlling unidirectional and bidirectional data flows in (Constrained Application Protocol) CoAP/6LoWPAN networks. The proposed algorithms are based on a distributed back pressure concept. The proposed algorithms use a buffer occupancy strategy (in Griping) and missing acknowledgement packet (in Deaf and Fuse) to detect the congestion as well as AIMD scheme to mitigate the congestion by adjusting the transmission rate to reduce the injected packets into the network.

In [16], Hellaoui and Koudil proposed a congestion control solution for CoAP/6LoWPAN networks. The proposed algorithm is based on a bird flocking concept to pass packets through uncongested areas and avoid congested ones. The proposed mechanism uses the buffer occupancy strategy to detect congested nodes in the network as well as the resource control method to mitigate the congestion by selecting the least congested routes to deliver packets to the destination (sink node). In [17], [18], Kim et al. proposed an effective queue utilization based RPL algorithm called (QU-RPL). QU-RPL uses the queue utilization factor in the parent selection process to satisfy the traffic load balancing. When a node experiences a certain number of consecutive buffer overflows, it broadcasts a DIO (DODAG Information Object) message which contains the congestion information. The node changes its parent on experiencing congestion with one that has less buffer occupancy and lower hop distance to the sink node. Otherwise, without congestion, the node chooses its best parent based on the same parent selection mechanism of the default RPL.

In [19] and [20], the authors proposed a congestion control mechanism called Game Theory Congestion Control (GTCC) for 6LoWPAN networks. The proposed protocol detects congestion by using the network packet flow rate which is packet generation rate subtracted by packet service rate. When a parent node detects congestion, it sends a congestion message to its children through a DIO control packet. When the children nodes receive the DIO packet, they start the parent-change procedure. In this procedure, the node uses the potential game theory method to decide whether to change its parent or not. When the node changes its parent, it broadcasts a new DIO message to notify other nodes and update their information. In [21], Tang et al. proposed a congestion avoidance multipath routing algorithm based on RPL called CA-RPL. Also, the authors propose a routing metric for RPL called DELAY_ROOT which minimizes the average delay toward the root node. CA-RPL mitigates network congestion by distributing a large amount of traffic to different paths. The proposed algorithm uses the DELAY_ROOT and three other metrics: ETX (expected transmission count), rank and number of received packets for parent selection process.

In [22], Al-Kashoash et al. proposed a new RPL based objective function called congestion-aware objective function (CA-OF) that works efficiently when congestion occurs. The proposed objective function combines two metrics (buffer occupancy and ETX) and forwards packets to sink node through less congested nodes. CA-OF reflects how much the nodes are congested by using buffer occupancy metric and how much the wireless link is congested by using the ETX metric. Recently, in [23], Al-Kashoash et al. formulated the congestion problem in 6LoWPAN networks as a noncooperative game framework where the nodes (players) behave uncooperatively and demand high data rate in a selfish way. Based on this framework, we proposed a simple congestion control mechanism called Game Theory based Congestion Control Framework (GTCCF). The proposed algorithm adapts the nodes' sending rate using Nash Equilibrium solution concept such that congestion is mitigated. GTCCF is aware of node priorities and application priorities to support the IoT application requirements.

III. Network Setup and Problem Formulation

In 6LoWPAN networks, the RPL routing protocol [7] is responsible for constructing the network topology. Three types of nodes are defined: sink (root) nodes which provide connectivity to other networks, intermediate nodes which forward packets to the sink and leaf nodes. The construction of network topology is based on the DAG (Directed Acyclic Graph) concept where every node selects a neighbour as its parent based on an objective function which combines one or more routing metrics into a Rank. RPL organises nodes as Destination Oriented DAGs (DODAG) where a sink node works as the root of the DAG which is responsible to start forming the network topology. The DAG root broadcasts a DIO control message to other nodes in the network. When an intermediate node receives the DIO message, it replies to the sink node with DAO (Destination Advertisement Object) for joining the DODAG. Then, the intermediate node sends a DIO message to all neighbours. This process continues until the DIO message reaches the leaf nodes. When a node receives a DIO message from more than one neighbour, it selects its parent with a best Rank. Also, when a node does not receive a DIO message within a specific time, it sends a DIS (DODAG Information Solicitation) message to solicit DIO messages from its neighbours. The formed network topology is shown in Fig. 1.

Consider a part of the formed network (dashed-line rectangle [A] in Fig. 1) where 5, 2 and 1 leaf nodes select node

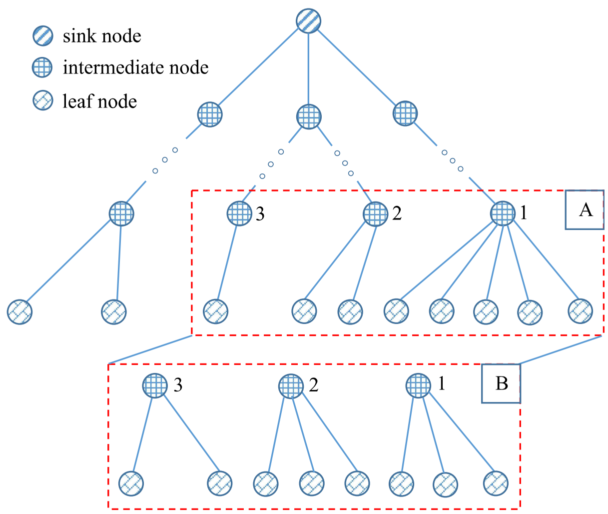


Fig. 1. Network topology based on RPL

1, node 2 and node 3 respectively as their parents at the network topology construction stage. Under low data rate, the leaf nodes send packets to the sink through their parents successfully. However, when congestion does occur, the leaf nodes start to send heavy traffic packets to their parents. In this situation, node 1 forwards packets from 5 leaf nodes, whereas node 2 and node 3 forward packets from 2 and 1 leaf nodes respectively. According to congestion analysis in [24], the majority of packets are lost due to buffer overflow when congestion occurs in 6LoWPAN network. Thus, a large number of packets are lost at node 1's buffer as its receiving rate from 5 leaf nodes is much higher than its forwarding rate. The default routing metrics specified in RFC 6551 [25] and de facto objective functions (ETX-OF [26] and OF0 [27]) do not reflect or are aware of congestion occurring. Hence, they do not distribute and balance the traffic load among parent nodes to reduce packet loss due to parents' buffer overflow (i.e. the leaf nodes do not change their current parent and select another less or non-congested one) as shown in dashed-line rectangle [B] in Fig. 1. The authors in [17], [22], [28], [29] also have demonstrated the problem of "load balancing" or "parent selection" within congestion in the RPL routing protocol. However, even with congestion aware routing metrics and objective functions, sometimes a leaf node can not find a less or non-congested parent and the incoming rate to the parent is higher than its outgoing rate. Therefore, according to Queuing Theory [30], the parent's buffer starts overflowing the incoming packets and congestion still exists. Thus, it is very important to have a rate adaptation policy to reduce the number of sent packets and therefore congestion can be controlled in the network. In this paper, we address both "parent selection" and "rate adaptation" problems and develop a hybrid solution to alleviate congestion in 6LoWPAN networks as shown in the next sections.

IV. MADM Based Resource Control

In RPL, the objective function, which is completely responsible for constructing the network topology, is separated from the core protocol specifications. This allows easy design and implementation of a new objective function that satisfies the application and network requirements. The objective function combines one or more routing metrics to produce a Rank value which is advertised by a DIO control message. Here, we use and utilize a multi-criteria optimization approach to combine three routing metrics and develop a new objective function called MADM-OF. The proposed objective function addresses and solves the "parent selection" problem within congestion by selecting a less or non-congested parent node from the existing "alternatives" or "parents" by considering multiple "attributes" or "routing metrics". There are many common methodologies for MADM such as simple additive weighting (SAW), the technique for order preference by similarity to ideal solution (TOPSIS), analytical hierarchy process (AHP), grey relational analysis (GRA), etc. [9]. In our proposal, we use GRA approach which is part of grey theory developed by Deng [31] and it has been successfully applied for solving different problems in various fields [32]. Before we describe the procedures of GRA methodology, we list and explain the routing metrics (attributes) used to find the best parent (alternative) in term of congestion. We use three routing metrics which reflect how much the nodes and network are congested as follows:

- Buffer Occupancy (BO): is defined as the number of packets stored at the node's buffer waiting to be transmitted. As the majority of packets are lost due to buffer overflow when congestion occurs, BO is a good indicator of congestion and reflects the current congestion level at each node (i.e. it reflects how much the node is congested) [24]. Also, Michopoulos et al. [33] have demonstrated that BO can successfully detect and confront congestion under the 6LoWPAN protocol stack with a RDC mechanism.
- Expected Transmission Count (ETX): is defined as the expected number of transmissions to successfully transmit a packet on a wireless link. ETX metric finds a high throughput path on a multi-hop wireless network [34]. Also, ETX is used to distinguish (identify) lossy and/or congested wireless links [35].
- Queuing Delay (QD): is the amount of time since a packet is enqueued until it is dequeued (i.e. the amount of time a packet spent in the node's buffer). When congestion occurs, the amount of traffic load injected on a wireless link exceeds its capacity. As a result, this will cause queuing delay to increase rapidly as buffer fills up [36]. Thus, queuing delay is also a good indication of congestion occurrence.

One can use more routing metrics such as channel load (channel busyness ratio), packet loss and energy consumption [37]. But, as a sensor node has limited computation capability and to keep the calculation simple and straightforward; we use the above three metrics which are appropriate and reflect how much the node and wireless link are congested.

A. Grey Relational Analysis Procedure

Suppose a node (decision maker) has a set of m candidate parents (alternatives) $A = \{a_i, i = 1, 2, \ldots, m\}$ with a set of 3 routing metrics (attributes) $R = \{r_j, j = 1, 2, 3\}$ for each parent and a weight vector $W = \{w_j, j = 1,2,3\}$ which represents the importance (weight) of the attributes. Then, the MADM parent selection problem can be represented by a decision matrix D as follows:

$$D = \begin{bmatrix} r_1(a_1) & r_2(a_1) & r_3(a_1) \\ r_1(a_2) & r_2(a_2) & r_3(a_2) \\ \vdots & \vdots & \vdots \\ r_1(a_m) & r_2(a_m) & r_3(a_m) \end{bmatrix} \quad (1)$$

where $r_j(a_i)$ represents the value of j^{th} routing metric (attribute) for the i^{th} parent (alternative) for all $i = 1, 2, \ldots, m$ and $j = 1,2,3$. For our proposal, we have three routing metrics: BO, ETX and QD. Thus, $r_1 =$ BO, $r_2 =$ ETX and $r_3 =$ QD.

The procedure of GRA consists of four steps to generate the global comparison among the candidate parents as follows [9]:

1) *Grey Relational Generating (Normalization)*: as the unit of routing metrics are different (e.g. BO is measured in packets, while QD is measured in seconds), processing all values for every routing metric into a comparability sequence is necessary as follows:
$$x_{ij} = \frac{\max_{\forall i}\{r_j(a_i)\} - r_j(a_i)}{\max_{\forall i}\{r_j(a_i)\} - \min_{\forall i}\{r_j(a_i)\}} \quad (2)$$
where $x_{ij} \in [0,1]$ is the normalized value of j^{th} routing metric for the i^{th} parent for all $i = 1, 2, \ldots, m$ and $j = 1,2,3$. In our MADM, all attributes (BO, ETX and QD) are cost. Equation (2) is used for cost attributes, while for benefit attributes; there is another equation (see equation (2) in [9]).
2) *Reference Sequence Definition*: the reference sequence is used to find the alternative (parent) whose comparability sequence is closet to the reference (preferred) sequence. In our MADM, if the value of x_{ij} is equal to 1 or nearer to 1, this means the performance of parent i is the best one for routing metric j. Thus, we define the reference sequence $x_{0j} = 1$ for all $j = 1,2,3$.
3) *Grey Relational Coefficient Calculation*: grey relational coefficient is used to determine how x_{ij} is close to x_{0j} and it can be calculated as follows:
$$\gamma(x_{ij}, x_{0j}) = \frac{\min_{\forall i,\forall j}\{\Delta_{ij}\} + \zeta \max_{\forall i,\forall j}\{\Delta_{ij}\}}{\Delta_{ij} + \zeta \max_{\forall i,\forall j}\{\Delta_{ij}\}} \quad (3)$$
where $\Delta_{ij} = |x_{0j} - x_{ij}|$ and $\zeta \in [0,1]$ is the distinguishing coefficient for all $i = 1, 2, \ldots, m$ and $j = 1,2,3$.
4) *Grey Relational Grade Calculation*: after the grey relational coefficients $\gamma(x_{ij}, x_{0j})$ $\forall i, \forall j$ are calculated, finally; the grey relational grade of parent (alternative) a_i for all $i = 1, 2, \ldots, m$ can be calculated as follows:
$$\Gamma(a_i) = \sum_{j=1}^{3} w_j \gamma(x_{ij}, x_{0j}) \quad (4)$$
where w_j is the weight of routing metric (attribute) j for all $j = 1,2,3$ such that $\sum_{j=1}^{3} w_j = 1$.

The grey relational grade is equivalent to the RPL objective function Rank where a node selects a parent with largest grey relational grade which represents the best Rank. The procedures to calculate the Rank value is similar to the default RPL but with different methodology (Here we use GRA method). The advantages of GRA methodology are: (i) the results are based on the original data and (ii) the calculations are simple and straightforward where the 6LoWPAN mote has limited processing capability [38].

B. Routing Metric Weights Calculation

The weights w_1, w_2 and w_3 represent the importance of attributes (routing metrics) BO, ETX and QD respectively. The weight of attributes plays an important role in the process of decision making where many methods have been proposed to determine the weights [39]. Here, we use the standard deviation (SD) method due to its simple calculations as 6LoWPAN motes have constrained computational power. The SD method determines the weights in terms of their standard deviations as follows [39]:

$$w_j = \frac{\sigma_j}{\sum_{u=1}^{3} \sigma_u} \quad (5)$$

$$\sigma_j = \sqrt{\frac{1}{m} \sum_{i=1}^{m} (x_{ij} - \bar{x}_j)^2} \quad (6)$$

$$\bar{x}_j = \frac{1}{m} \sum_{i=1}^{m} x_{ij} \quad (7)$$

for all $j = 1,2,3$.

V. Optimization Based Traffic Control

The MADM-OF searches for non-congested parents to mitigate congestion by achieving traffic load balancing and distribution. On the other hand, sometimes; the non-congested parent is not available and congestion still exists. Thus, applying the traffic control strategy is important to reduce the number of injected packets and therefore congestion can be controlled and solved. Here, we utilize optimization theory to propose a new **T**raffic **C**ontrol mechanism called NUM-TC which adapts the source nodes' sending rate by using the NUM framework when the resource control strategy can not be applied. Consider a parent node has a set of z children nodes, $L = \{N_l, l = 1, 2, \ldots, z\}$ which are competing to send data packets to sink through their parent. Also, we assume that: (i) Each node in the network has a buffer size of B packets, (ii) The children nodes have different priorities $P = \{p_1, p_2, ..., p_z\}$ where p_l is the priority of node N_l such that $p_l > 0$ for all $l = 1, 2, \ldots, z$. The priorities of children nodes are specified by user, based on the importance of node and the importance of the hosted applications, (iii) Each child node hosts a set of y applications $K = \{app^k; k = 1, 2, \ldots, y\}$ with different priorities; denoted by p_l^k to the priority of application app^k hosted in child node N_l such that $p_l^k > 0$ for all $l = 1, 2, \ldots, z$ and $k = 1, 2, \ldots, y$. The priorities of hosted applications are specified by user based on importance and type of application (i.e. real-time application, reliable application, etc.).

According to Queuing Theory, congestion and buffer overflow occur when the incoming rate to a parent node (λ_{in}) from its children nodes is higher than its forwarding rate (λ_{out}). So, the problem is how to allocate the available parent's forwarding rate (λ_{out}) among the children nodes in an efficient manner such that congestion can be alleviated. The NUM framework can be used to model the "sending rate allocation" problem as a constrained optimization problem where a node N_l has a utility function $U_l(\lambda_l)$ and λ_l is the sending rate allocated to node N_l for all $l = 1, 2, \ldots, z$. Formally, the NUM problem can be expressed as follows [40]:

$$\begin{aligned} & \underset{\boldsymbol{\lambda}}{\text{maximize}} && \sum_{l=1}^{z} U_l(\lambda_l) \\ & \text{subject to} && \sum_{l=1}^{z} \lambda_l \leq \lambda_{out} \\ &&& \lambda_l \geq 0, \qquad \forall l = 1, 2, \ldots, z \end{aligned} \tag{8}$$

where $\boldsymbol{\lambda}$ is a vector consisting of $\lambda_1, \lambda_2, \ldots, \lambda_z$ and $\lambda_{out} > 0$.

Many types of utility function are commonly used such as exponential, logarithmic, linear and sigmoidal [41]. In our framework, we use the logarithmic utility function as it has strict concavity property. Also, different utility functions exist in term of fairness such as proportional fairness, weighted proportional fairness and max-min fairness [42]. We select the weighted proportional fairness to satisfy that each node obtains sending rate according to its priority. Thus, the utility function of node N_l can be expressed as follows:

$$U_l(\lambda_l) = \omega_l \log(\lambda_l) \tag{9}$$

where ω_l is the weight of node N_l's utility function such that $\omega_l > 0$ for all $l = 1, 2, \ldots, z$.

A. Optimal Sending Rate Computation

The proposed utility function $U_l(\lambda_l)$ is an increasing, strictly concave and continuously differentiable function of λ_l over $\lambda_l \geq 0$ for all $l = 1, 2, \ldots, z$. Therefore, the problem in equation (8) has a unique global maximum solution (point) (*Proof*: See Proposition 2.1.1 in [43]). The problem in equation (8) can be solved through decentralized distributed algorithms by decomposing the original problem into sub-problems (solved locally) and a master problem (e.g. primal decomposition and dual decomposition) to reduce information exchanged among nodes in the network [10], [40]. However, by using these algorithms, convergence to an optimal solution may require a long time and the solution in 6LoWPAN networks has to be fast and quick. Also, in our framework; the parent node can send congestion information in a simple way by sending a broadcast message. Now, since $\log(\lambda_l) \longrightarrow -\infty$ as $\lambda_l \longrightarrow 0$, the optimal sending rate (solution) will assign a strictly positive rate to each node, and so the last constraint can be ignored [40]. Thus, in order to solve the problem in equation (8) without decomposing, we introduce the Lagrange

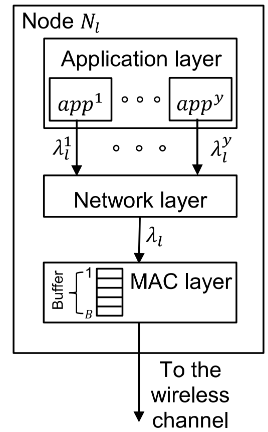


Fig. 2. Node model

multiplier v and define the Lagrangian function $\mathcal{L}(\boldsymbol{\lambda}, v)$ for all $l = 1, 2, \ldots, z$ as follows:

$$\mathcal{L}(\boldsymbol{\lambda}, v) = \sum_{l=1}^{z} U_l(\lambda_l) + v(\lambda_{out} - \sum_{l=1}^{z} \lambda_l) \tag{10}$$

where the KKT conditions for optimality are as follows:

$$\begin{gathered} v \geq 0 \\ \lambda_{out} - \sum_{l=1}^{z} \lambda_l \geq 0 \\ \nabla \sum_{l=1}^{z} U_l(\lambda_l) + v \nabla(\lambda_{out} - \sum_{l=1}^{z} \lambda_l) = 0 \\ v(\lambda_{out} - \sum_{l=1}^{z} \lambda_l) = 0 \end{gathered} \tag{11}$$

Then, the optimal sending rate of node N_l after solving the problem in equation 8 is as follows:

$$\lambda_l = \frac{\omega_l \lambda_{out}}{\sum\limits_{c=1}^{z} \omega_c} \tag{12}$$

B. Allocation of Node's Sending Rate among Its Applications

In the IoT applications, a sensor node does not host a single application as in the traditional WSNs. However, it hosts many applications with different requirements. Some of them are real time applications where data is time critical, while others are non-real time applications. Therefore, it is important for each node to be aware of the priorities of the hosted applications. Consider a node hosts a set of y applications $K = \{app^k; k = 1, 2, \ldots, y\}$ with different priorities competing to send data packets through the node as shown in Fig. 2. We denote by p_l^k to the priority of application app^k hosted in node N_l for all $l = 1, 2, \ldots, z$ and $k = 1, 2, \ldots, y$. To allocate the node's sending rate (λ_l) fairly among its applications according to their priorities and prevent buffer overflow to occur inside the node (i.e. internal congestion), we can model the "application sending rate allocation" problem as a constrained optimization problem by using the NUM framework. In the NUM framework, an application, app^k, has a utility function $U^k(\lambda_l^k)$ where λ_l^k is the sending rate allocated to application app^k hosted in node N_l for all $k = 1, 2, \ldots, y$ and $l = 1, 2, \ldots, z$. It can be expressed as follows:

$$\begin{aligned} \underset{\lambda_l^1, \lambda_l^2, \ldots, \lambda_l^k}{\text{maximize}} \quad & \sum_{k=1}^{y} U^k(\lambda_l^k) \\ \text{subject to} \quad & \sum_{k=1}^{y} \lambda_l^k \leq \lambda_l \\ & \lambda_l^k \geq 0, \qquad \forall k = 1, 2, \ldots, y \end{aligned} \tag{13}$$

We use the logarithmic, weighted proportional fairness utility function such that each application obtains a sending rate according to its priority as follows:

$$U^k(\lambda_l^k) = \omega^k \log(\lambda_l^k) \tag{14}$$

where ω^k is the weight of application app^k's utility function such that $\omega^k > 0$ for all $k = 1, 2, \ldots, y$.
To solve the problem in equation (13), following the same procedures used to solve the problem in equation (8) and the optimal sending rate of application, app_l^k, is as follows:

$$\lambda_l^k = \frac{\omega^k \lambda_l}{\sum_{d=1}^{y} \omega^d} \tag{15}$$

With regards to the values of ω_l and ω^k, if a node (an application) with higher p_l (p_l^k) value has high priority (e.g. if $p_i = 1$ and $p_j = 2$, this means that node N_j has higher priority than node N_i), then $\omega_l = p_l$ ($\omega^k = p_l^k$). On the other hand, if a node (an application) with a lower p_l (p_l^k) value has high priority, then $\omega_l = 1/p_l$ ($\omega^k = 1/p_l^k$).

VI. Hybrid Congestion Alleviation Algorithm Implementation

The OHCA algorithm is designed to use the network resources effectively and utilize positive aspects of using both resource and traffic control strategies. According to Queuing Theory, if the arrival rate (λ_{in}) at a parent's buffer is higher than the service rate (λ_{out}), the parent's buffer will overflow and congestion will occur. Thus, the parent node periodically checks the congestion condition ($\lambda_{in} > \lambda_{out}$) every interval time '$I_{check}$'. If the parent node encounters congestion, it broadcasts a DIO message, which contains congestion information, to its children. When a child node receives the DIO message, it firstly applies the resource control strategy by using MADM-OF to select a non-congested parent and subsequently forwards packets through it. MADM-OF combines three metrics (BO, ETX and QD) to produce a Rank value such that a candidate parent with the best Rank becomes selected as the current parent. To compute and accurately estimate the value of these metrics, we use Brown's simple exponential smoothing model [44] as follows:

$$r_j(t+1) = \psi_j r_j(t) + (1 - \psi_j) r_j(t-1) \tag{16}$$

where $r_j(t+1)$, $r_j(t)$ and $r_j(t-1)$ are the expected, current and historic values of metric j respectively for $j = 1, 2, 3$ and ψ_j is smoothing factor of metric j such that $0 < \psi_j < 1$. A large value of ψ_j reduces the level of smoothing and gives high weight to current measurement of r_j, while a value of ψ_j close to zero gives greater smoothing effect and less responsive to recent changes in r_j value. Similarity, the forwarding rate of parent λ_{out} is not constant with time. It is increased or decreased due to the operation of the CSMA algorithm (i.e. backoff time), MAC parameters (i.e. channel check rate) and number of active nodes. Thus, to avoid sending high overhead DIO packets, we use Brown's simple exponential smoothing model to estimate the actual maximum service rate as follows:

$$\lambda_{out}(t+1) = \psi \lambda_{out}(t) + (1 - \psi) \lambda_{out}(t-1) \tag{17}$$

where $\lambda_{out}(t+1)$, $\lambda_{out}(t)$ and $\lambda_{out}(t-1)$ are the expected, current and historic forwarding rate of the parent respectively and ψ is smoothing factor such that $0 < \psi < 1$. The equations (16) and (17) are updated on a per incoming packet basis.

On the other hand, if the child node can not find a non-congested parent node, it applies the traffic control strategy by using the NUM-TC mechanism. Firstly, the child node selects the less congested parent from the candidate parents. Then, it adjusts its sending rate based on equation (12) and congestion information received from the selected parent. After that, the child node allocates its updated sending rate among the hosted applications according to their priorities as in equation (15). Lastly, the network topology is governed by RPL through transmission of DIO, DAO and DIS control messages. The DIO transmission strategy is controlled by the "Trickle Algorithm" [45] where the Trickle timer is set to the minimum interval size, I_{min}, and it is doubled after the timer expires until it reaches the maximum interval size, I_{max}. Thus, the Trickle algorithm is not aware of the occurrence of congestion. Therefore, the operation of the algorithm is modified such that when congestion occurs, the timer is reset to I_{min}.

VII. Performance Evaluation

The proposed algorithm has been tested and evaluated on different network scenarios through simulation using Contiki 3.0 OS and Cooja simulator. In the first scenario, we use a network topology of one sink node, 5 intermediate nodes and 4 leaf (source) nodes with node ID of N_4, N_5, N_6 and N_7 (as illustrated in Fig. 3). In the second scenario, we use a network of one sink node, 18 intermediate nodes and 6 source nodes with node ID of N_{20}, N_{21}, N_{22}, N_{23}, N_{24} and N_{25}. Also, our proposal is compared with a traffic control based algorithm (DCCC6 [14]) and a resource control based algorithm (QU-RPL [17], [18]). In the simulation, the source nodes start sending packets at high data rate (6 packets/s) to create a congested situation. During the simulation, the source nodes start sending packets after 60s so the network topology construction is completed, the simulation time is set to 600s. Cooja simulates the hardware of a set of real sensor nodes, such as Tmote Sky, which is used in the simulation. Also, Cooja simulator implements a number of wireless channel models such as Unit Disk Graph Medium (UDGM) - Distance Loss, which is used in the simulation. We use Powertrace [46] to measure the energy consumption of each node where it is a

TABLE I
PROTOCOL STACK AND SIMULATION PARAMETERS

Layer	Protocol	Parameter value
Application	Every leaf node sends high data rate packets to sink	Application payload = 30 bytes
Transport	UDP	
Network	uIPv6 + RPL	OF = MADM-OF (OHCA) OF = OF0 (DCCC6) OF = QU-OF (QU-RPL)
Adaptation	SICSlowpan layer	Compression method = HC06
Data Link	CSMA (MAC layer) Contikimac (RDC layer) 802.15.4 (framer)	Buffer size = 8 packets MAC reliability (ACK) = enabled MAC max. retransmission = 3 Channel check rate = 8 Hz Max. frame size = 127 bytes
Physical	CC2420 RF transceiver	

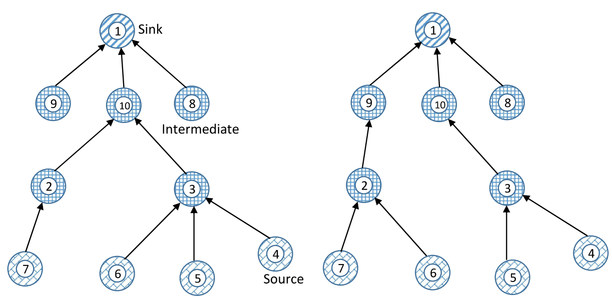


Fig. 3. Network topology in scen. 1 (left) DCCC6 (right) OHCA and QU-RPL

run-time network-level power profiling system that uses state tracking to estimate the energy consumption and it is accurate up to 94%. The protocol stack and simulation parameters used in the simulation are shown in Table I. We assume that a node (an application) with a higher value of priority $(p_l(p_l^k))$ has high priority. In the first scenario, we have set priorities of N_4, N_5, N_6 and N_7 to 2, 1, 1 and 2 respectively where they host two, one, two and three applications respectively with priorities $p_4^1 = p_6^2 = p_7^1 = 1$, $p_4^2 = p_6^1 = p_7^2 = 2$ and $p_7^3 = 3$. In the second scenario, we have set priorities of N_{20}, N_{21}, N_{22}, N_{23}, N_{24} and N_{25} to 2, 1, 2, 2, 1, and 2 respectively where they host two, one, two, two, one and two applications respectively with priorities $p_{20}^1 = p_{22}^2 = p_{23}^1 = p_{25}^2 = 1$ and $p_{20}^2 = p_{22}^1 = p_{23}^2 = p_{25}^1 = 2$. For our proposal, we have set $I_{check} = 384$ clock ticks and $\psi = \psi_j = 0.4$; $\forall j = 1, 2, 3$ where 128 clock ticks = 1 second.

Next, we compare OHCA, DCCC6 and QU-RPL in terms of network topology layout, overall throughput, average throughput per node, applications' sending rate, weighted fairness index, end-to-end delay, energy consumption and lost packets due to buffer overflow. We have computed the average value of results obtained from scenario 1 and scenario 2 as follows:

A. *Network topology*

Fig. 3 shows the routing topology for OHCA, DCCC6 and QU-RPL algorithms in scenario 1. At the topology construction stage, nodes 2 and 3 select node 10 as their parent and

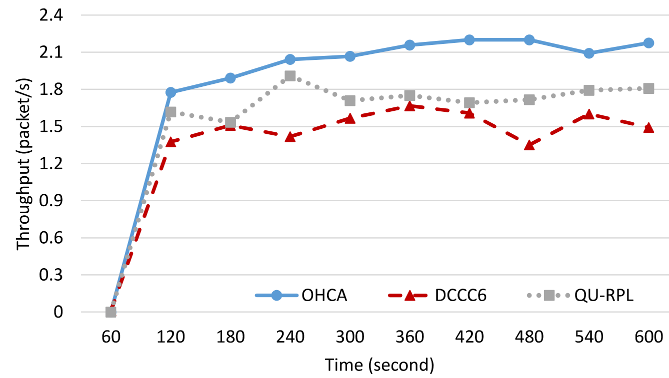


Fig. 4. Throughput

nodes 4, 5 and 6 select node 3 as their parent, while node 2 is selected as parent by node 7. When congestion occurs, many packets overflow buffers of nodes 3 and 10. As DCCC6 does not consider the load balancing problem with RPL and is not aware of buffer overflow, thus; nodes do not change their parents and select less congested ones. In contrast, with OHCA and QU-RPL algorithms, node 2 changes its current congested parent, node 10, and selects less congested parent which is node 9. Also, node 6 changes its forwarding parent from node 3 to node 2. The reason is that OHCA and QU-RPL are aware of buffer overflow and congestion at nodes and they consider the load balancing problem in the routing protocol by using MADM-OF and QU-OF respectively. Similarly, in scenario 2; nodes forward packets through less congested parents in OHCA and QU-RPL, while DCCC6 does not consider the parent selection problem within congestion in RPL.

B. *Throughput*

Fig. 4 shows the overall throughput which is the total number of received packets every second at the sink node. It is clear that OHCA has higher throughput ($\approx$ 2 packet/s) than DCCC6 ($\approx$ 1.5 packet/s) and QU-RPL ($\approx$ 1.7 packet/s). The reason is that OHCA forwards packets through less congested nodes by using MADM-OF as well as adapting the sending rate of nodes by using NUM-CC framework when buffer drops still occur. Therefore, the number of forwarded packets to the sink node increases by exploiting the available network resources in an effective manner. Also, from this figure; QU-RPL is seen to have better performance in term of throughput as compared to DCCC6. The reason is that QU-RPL utilizes the available non-congested nodes and therefore; packets forwarded to the sink node increase. While DCCC6 does not utilize the available network resources (non-congested nodes) and it only adapts the nodes' sending rate by using a modified AIMD policy and therefore throughput decreases.

C. *Throughput per node*

Fig. 5 and Fig. 6 show the average number of received packets every second from the source nodes at sink in scenario 1 and scenario 2 respectively. From these figures, it is clear that nodes in OHCA obtain throughput according to their priorities. For instance, with OHCA in scenario 1, N_4 and N_7

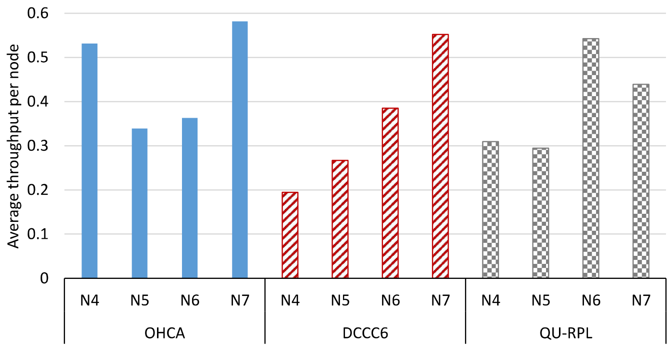


Fig. 5. Received packets/s from nodes in scen. 1

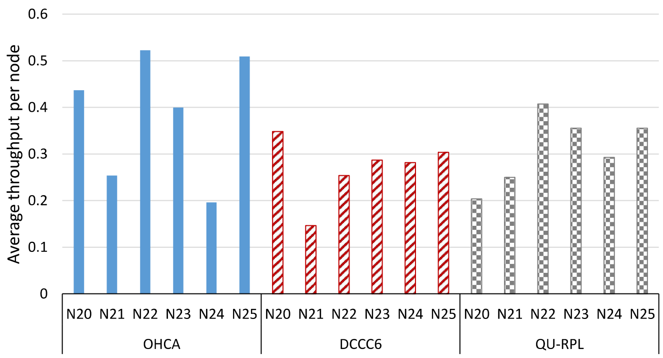


Fig. 6. Received packets/s from nodes in scen. 2

have the highest number of received packets (≈ 0.53 and ≈ 0.58 packet/s) respectively. While, nodes N_5 and N_6 have the lowest throughput (≈ 0.34 and ≈ 0.36 packet/s) respectively as they have low priorities as compared to other nodes. The reason is that OHCA is aware of node priorities where each node gets sending rate according to its priority. On the other hand, the nodes in DCCC6 and QU-RPL do not obtain a sending rate based on their priorities as these algorithms do not support awareness of node priorities. For example, with DCCC6 in scenario 1; node N_4 with higher priority has a lower number of received packets at sink (≈ 0.19 packet/s) as compared to node N_5 (≈ 0.26 packet/s) which has low priority. Similarity, in QU-RPL; node (N_7) with higher priority has lower throughput (≈ 0.43 packet/s) than node N_6 (≈ 0.54 packet/s) which has low priority.

D. Applications' sending rate

Fig. 7 and Fig. 8 show the average sending rate of applications (packet/s) hosted in the source nodes for OHCA in scenario 1 and scenario 2 respectively. Each application obtains the sending rate according to its priority. For example, in scenario 1, application app^1 in node N_4 obtains low sending rate (≈ 0.17 packet/s) as compared to application app^2 (≈ 0.35 packet/s) which has higher priority, similarity for nodes N_6 and N_7. While, in scenario 1, the application hosted in node N_5 gets sending rate equal to N_5's sending rate as it is hosted alone. In contrast, other algorithms do not support multiple applications hosted in each sensor node and they are not aware of application priorities.

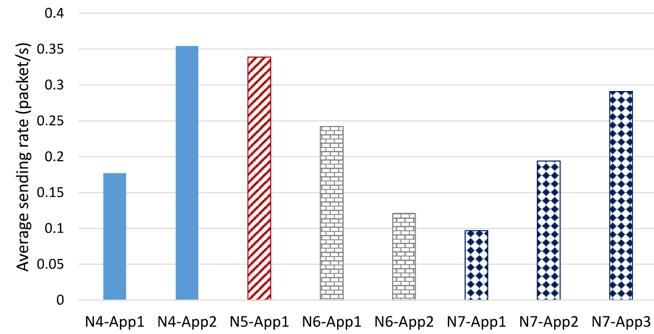


Fig. 7. Applications' rate of OHCA in scen. 1

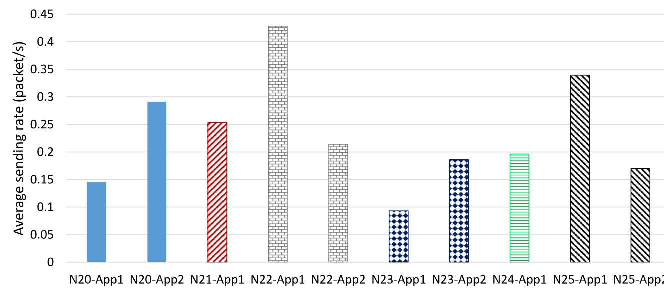


Fig. 8. Applications' rate of OHCA in scen. 2

E. Weighted fairness index

Fig. 9 shows the weighted fairness index (WFI) which is an indication of how much the nodes associated with a parent are treated fairly according to their priorities. We have calculated this metric similar to that used in [47] as follows:

$$WFI = \frac{\left[\sum_{l=1}^{z}(\frac{th_l}{\omega_l})\right]^2}{z\sum_{l=1}^{z}(\frac{th_l}{\omega_l})^2} \tag{18}$$

where th_l is throughput of node N_l.
From this figure, it is clear that OHCA achieves fairness index close to 1 (≈ 0.97) which indicates a high fairness allocation of overall throughput among the source nodes based on their priorities. On the other hand, DCCC6 and QU-RPL have lower WFI (≈ 0.89 and ≈ 0.66 respectively) than OHCA as they do not support awareness of node priorities.

F. End-to-end delay

Fig. 10 shows end-to-end delay which is the time between a packet being generated at the application of the source until its successful reception at the application of the final destination. OHCA has lower end-to-end delay as compared to DCCC6 and QU-RPL. The reason is that OHCA firstly searches for a non-congested parent to forward packets and if congestion still exists, then the number of injected packets into the network is reduced by reducing the nodes' sending rates. Therefore, buffer overflow is removed and packets do not a wait long time in the buffer. On the other hand, DCCC6 has high delay because of the modified AIMD mechanism used where the nodes' sending rates are increased periodically and decreased

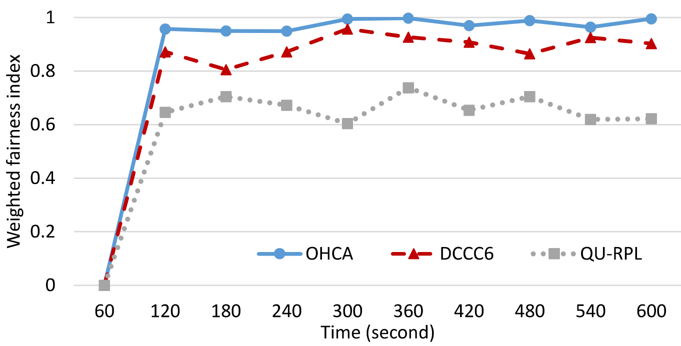


Fig. 9. Weighted fairness index

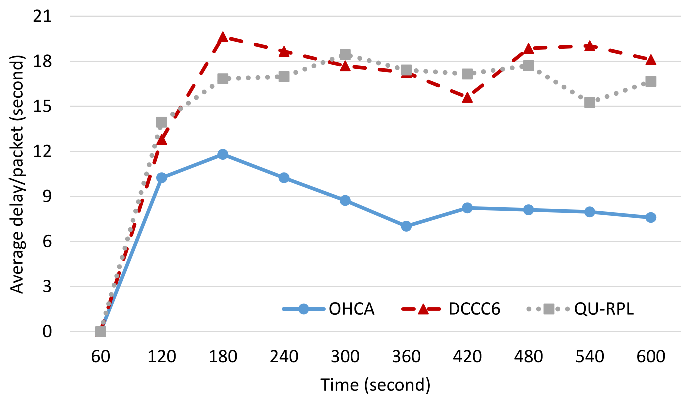


Fig. 10. End-to-end delay

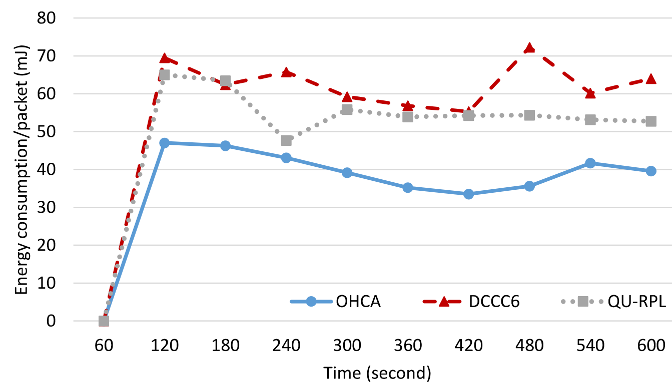


Fig. 11. Energy consumption per successful packet

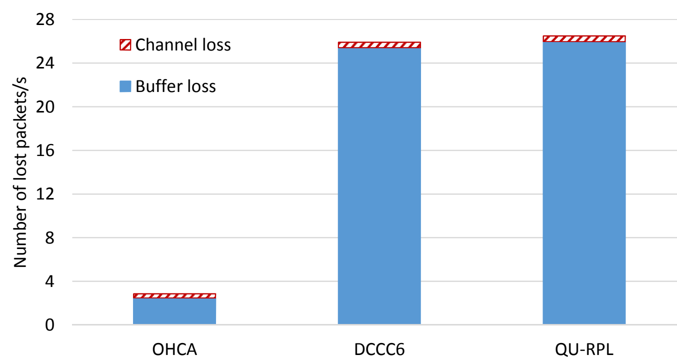


Fig. 12. Number of lost packets

when congestion occurs and then this process continues. As a result, the packets wait a long time in the nodes' buffers. Although QU-RPL forwards packets through less congested paths, it does not have a policy to reduce the nodes' sending rates when buffer drops still occur. Consequently, packets experience a long end-to-end delay if buffers are full most the time.

G. Energy consumption

Fig. 11 shows the energy consumption due to transmission and reception in the source and intermediate nodes per successfully delivered packet. We note that with OHCA, the energy consumption in the network is less than others as DCCC6 and QU-RPL waste energy by transmitting and receiving packets which are then lost due to buffer overflow on the path without successful delivery.

H. Lost packets

Fig. 12 shows the total number of lost packets every second in the network due to buffer overflow and due to wireless channel loss. It is obvious that OHCA loses less packets at the buffer than others for reasons stated above. However, the number of lost packets in DCCC6 and QU-RPL is higher than OHCA algorithm as DCCC6 uses the modified AIMD policy and QU-RPL does not have a sending rate adaptation mechanism. From this figure, the number of buffer overflowed packets per second for OHCA, DCCC6 and QU-RPL are 2.47, 25.41 and 25.91 respectively. Also, the number of lost packets due to channel loss per second for OHCA, DCCC6 and QU-RPL are 0.37, 0.49 and 0.54 respectively.

Overall, based on the simulation results, it is clear that OHCA has superior performance than DCCC6 and QU-RPL algorithms. Also, it is clear that OHCA improves performance in terms of overall throughput, average weighted fairness index, end-to-end delay, energy consumption and number of lost packets due to buffer overflow by an overall average of more than 28.36%, 28.02%, 48.07%, 31.97% and 90.35% respectively compared to DCCC6 and QU-RPL schemes.

VIII. Conclusion

In this paper, the congestion problem in 6LoWPAN networks is addressed by using a hybrid solution which combines traffic control and resource control strategies. We have modelled the "parent selection" problem as a MADM problem which is solved by using grey relational analysis methodology for achieving traffic load balancing and distribution in the presence of congestion and forwarding packets through non-congested parents. Also, we have modelled the "nodes' sending rate adaptation" and "applications' sending rate allocation" as constrained optimization problems by using optimization theory and the NUM framework. The optimal sending rates of nodes and applications are computed by using Lagrange multipliers and KTT conditions. Based on the MADM and NUM frameworks, we propose a new congestion control algorithm called optimization based hybrid congestion alleviation (OHCA) which utilizes the advantages of using both traffic and resource control strategies and uses the network resources effectively. To support the IoT application requirements, OHCA is aware of node priorities and application priorities as well as being designed for the unique characteristics of IEEE 802.15.4, IPv6 and 6LoWPAN. The proposed algorithm has been evaluated in Contiki 3.0 OS and compared with other algorithms. Simulation results show that OHCA improves the QoS parameters i.e. throughput, weighted fairness index, end-to-end delay, energy consumption and lost packets due to buffer overflow as compared to existing algorithms.

REFERENCES

[1] Z. Shelby and C. Bormann, *6LoWPAN: The Wireless Embedded Internet*. John Wiley & Sons, 2009.

[2] S. Li, L. Da Xu, and S. Zhao, "The internet of things: a survey," *Information Systems Frontiers*, vol. 17, no. 2, pp. 243–259, 2015.

[3] N. Khalil, M. R. Abid, D. Benhaddou, and M. Gerndt, "Wireless Sensors Networks for Internet of Things," in *Proceeding of IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*. IEEE, 2014, pp. 1–6.

[4] N. Kushalnagar, G. Montenegro, and C. Schumacher, "IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals," *Internet Engineering Task Force (IETF), RFC 4919*, 2007.

[5] A. Ghaffari, "Congestion Control Mechanisms in Wireless Sensor Networks: A Survey," *Journal of Network and Computer Applications*, vol. 52, pp. 101–115, 2015.

[6] M. A. Kafi, D. Djenouri, J. Ben-Othman, and N. Badache, "Congestion Control Protocols in Wireless Sensor Networks: A Survey," *Communications Surveys & Tutorials, IEEE*, vol. 16, no. 3, pp. 1369–1390, 2014.

[7] T. Winter, P. Thubert, A. Brandt, J. Hui, and R. Kelsey, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks," *Internet Engineering Task Force (IETF), RFC 6550*, 2012.

[8] F. P. Kelly, A. K. Maulloo, and D. K. Tan, "Rate Control for Communication Networks: Shadow Prices, Proportional Fairness and Stability," *Journal of the Operational Research society*, vol. 49, no. 3, pp. 237–252, 1998.

[9] Y. Kuo, T. Yang, and G.-W. Huang, "The use of grey relational analysis in solving multiple attribute decision-making problems," *Computers & Industrial Engineering*, vol. 55, no. 1, pp. 80–93, 2008.

[10] G. Tychogiorgos and K. K. Leung, "Optimization-based resource allocation in communication networks," *Computer Networks*, vol. 66, pp. 32–45, 2014.

[11] A. Dunkels, B. Grönvall, and T. Voigt, "Contiki - a Lightweight and Flexible Operating System for Tiny Networked Sensors," in *Proceeding of 29th Annual IEEE International Conference on Local Computer Networks*. IEEE, 2004, pp. 455–462.

[12] F. Osterlind, A. Dunkels, J. Eriksson, N. Finne, and T. Voigt, "Cross-Level Sensor Network Simulation with COOJA," in *Proceedings of 31st IEEE Conference on Local Computer Networks*. IEEE, 2006, pp. 641–648.

[13] C. Sergiou, P. Antoniou, and V. Vassiliou, "A Comprehensive Survey of Congestion Control Protocols in Wireless Sensor Networks," *Communications Surveys & Tutorials, IEEE*, vol. 16, no. 4, pp. 1839–1859, 2014.

[14] V. Michopoulos, L. Guan, G. Oikonomou, and I. Phillips, "DCCC6: Duty Cycle-aware congestion control for 6LoWPAN networks," in *Proceeding of IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*. IEEE, 2012, pp. 278–283.

[15] A. P. Castellani, M. Rossi, and M. Zorzi, "Back pressure congestion control for CoAP/6LoWPAN networks," *Ad Hoc Networks*, vol. 18, pp. 71–84, 2014.

[16] H. Hellaoui and M. Koudil, "Bird Flocking Congestion Control for CoAP/RPL/6LoWPAN Networks," in *Proceedings of the 2015 Workshop on IoT challenges in Mobile and Industrial Systems*. ACM, 2015, pp. 25–30.

[17] H.-S. Kim, J. Paek, and S. Bahk, "QU-RPL: Queue Utilization Based RPL for Load Balancing in Large Scale Industrial Applications," in *Proceeding of 12th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, 2015, pp. 265–273.

[18] H.-S. Kim, H. Kim, J. Paek, and S. Bahk, "Load balancing under heavy traffic in rpl routing protocol for low power and lossy networks," *IEEE Transactions on Mobile Computing*, 2016.

[19] J. P. Sheu, C. X. Hsu, and C. Ma, "A Game Theory Based Congestion Control Protocol for Wireless Personal Area Networks," in *Proceeding of 2015 IEEE 39th Annual Computer Software and Applications Conference (COMPSAC)*, vol. 2, July 2015.

[20] C. Ma, J.-P. Sheu, and C.-X. Hsu, "A Game Theory Based Congestion Control Protocol for Wireless Personal Area Networks," *Journal of Sensors*, vol. 2016, 2015.

[21] W. Tang, X. Ma, J. Huang, and J. Wei, "Toward Improved RPL: A Congestion Avoidance Multipath Routing Protocol with Time Factor for Wireless Sensor Networks," *Journal of Sensors*, vol. 2016, 2015.

[22] H. A. A. Al-Kashoash, Y. Al-Nidawi, and A. H. Kemp, "Congestion-Aware RPL for 6L0WPAN Networks," in *Proceeding of Wireless Telecommunications Symposium (WTS), 2016*. IEEE, 2016, pp. 1–6.

[23] H. Al-Kashoash, M. Hafeez, and A. Kemp, "Congestion Control for 6LoWPAN Networks: A Game Theoretic Framework," *IEEE Internet of Things Journal*, vol. 4, no. 3, pp. 760–771, 2017.

[24] H. A. A. Al-Kashoash, Y. Al-Nidawi, and A. H. Kemp, "Congestion Analysis for Low Power and Lossy Networks," in *Proceeding of Wireless Telecommunications Symposium (WTS), 2016*. IEEE, 2016, pp. 1–6.

[25] J.-P. Vasseur, M. Kim, K. Pister, N. Dejean, and D. Barthel, "Routing Metrics Used for Path Calculation in Low-Power and Lossy Networks," *RFC 6551*, 2012.

[26] O. Gnawali and P. Levis, "The ETX Objective Function for RPL," *Internet Draft: draft-gnawali-roll-etxof-00*, 2010.

[27] P. Thubert, "Objective Function Zero for The Routing Protocol for Low-Power and Lossy Networks (RPL)," *RFC 6552*, 2012.

[28] X. Liu, J. Guo, G. Bhatti, P. Orlik, and K. Parsons, "Load Balanced Routing for Low Power and Lossy Networks," in *Proceeding of 2013 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2013, pp. 2238–2243.

[29] J. Guo, X. Liu, G. Bhatti, P. Orlik, and K. Parsons, "Load Balanced Routing for Low Power and Lossy Networks," Jan. 21 2013, US Patent App. 13/746,173.

[30] J. Sztrik, *Basic Queueing Theory*. University of Debrecen, Faculty of Informatics, 2012.

[31] D. Ju-Long, "Control problems of grey systems," *Systems & Control Letters*, vol. 1, no. 5, pp. 288–294, 1982.

[32] S. Liu and J. Y. L. Forrest, *Grey Systems: Theory and Applications*. Springer, 2010.

[33] V. Michopoulos, L. Guan, G. Oikonomou, and I. Phillips, "A Comparative Study of Congestion Control Algorithms in IPv6 Wireless Sensor Networks," in *Proceeding of 2011 International Conference on Distributed Computing in Sensor Systems and Workshops (DCOSS)*. IEEE, 2011, pp. 1–6.

[34] D. S. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A High-Throughput Path Metric for Multi-Hop Wireless Routing," *Wireless Networks*, vol. 11, no. 4, pp. 419–434, 2005.

[35] P. Karkazis, H. C. Leligou, L. Sarakis, T. Zahariadis, P. Trakadas, T. H. Velivassaki, and C. Capsalis, "Design of primary and composite routing metrics for RPL-compliant Wireless Sensor Networks," in *Proceeding of 2012 International Conference on Telecommunications and Multimedia (TEMU)*. IEEE, 2012, pp. 13–18.

[36] J. Koo, S. Ahn, and J. Chung, "A Comparative Study of Queue, Delay, and Loss Characteristics of AQM Schemes in QoS-Enabled Networks," *Computing and Informatics*, vol. 23, no. 4, pp. 317–335, 2012.

[37] M. A. Kafi, D. Djenouri, J. B. Othman, A. Ouadjaout, and N. Badache, "Congestion Detection Strategies in Wireless Sensor Networks: A Comparative Study with Testbed Experiments," *Procedia Computer Science*, vol. 37, pp. 168–175, 2014.

[38] R. Verma and N. P. Singh, "GRA Based Network Selection in Heterogeneous Wireless Networks," *Wireless personal communications*, vol. 72, no. 2, pp. 1437–1452, 2013.

[39] Y.-M. Wang and Y. Luo, "Integration of correlations with standard deviations for determining attribute weights in multiple attribute decision making," *Mathematical and Computer Modelling*, vol. 51, no. 1, pp. 1–12, 2010.

[40] R. Srikant and L. Ying, *Communication Networks: An Optimization, Control, and Stochastic Networks Perspective*. Cambridge University Press, 2013.

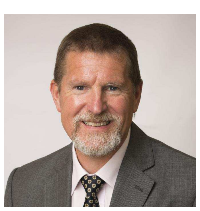
[41] L. Wang and G.-S. Kuo, "Mathematical Modeling for Network Selection in Heterogeneous Wireless Networks—A Tutorial," *Communications Surveys & Tutorials, IEEE*, vol. 15, no. 1, pp. 271–292, 2013.

[42] S. Huaizhou, R. V. Prasad, E. Onur, and I. Niemegeers, "Fairness in Wireless Networks: Issues, Measures and Challenges," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 5–24, 2014.
[43] D. P. Bertsekas, *Nonlinear Programming*. Athena scientific Belmont, 1999.
[44] R. G. Brown, *Smoothing, Forecasting and Prediction of Discrete Time Series*. Courier Corporation, 2004.
[45] P. Levis, T. Clausen, J. Hui, O. Gnawali, and J. Ko, "The trickle algorithm," *Internet Engineering Task Force, RFC6206*, 2011.
[46] A. Dunkels, J. Eriksson, N. Finne, and N. Tsiftes, "Powertrace: Network-level Power Profiling for Low-power Wireless Networks," Swedish Institute of Computer Science, Tech. Rep., 2011.
[47] M. Zawodniok and S. Jagannathan, "Predictive Congestion Control Protocol for Wireless Sensor Networks," *IEEE Transactions on Wireless Communications*, vol. 6, no. 11, pp. 3955–3963, 2007.

Hayder A. A. Al-Kashoash is Lecturer with Technical Institute/Qurna, Southern Technical University, Basra, Iraq. He recently got his PhD degree from School of Electronic and Electrical Engineering, University of Leeds, UK. His research Interests include congestion control and resource management in WSNs, 6LoWPAN and LPWAN by utilizing game theory and optimization theory.

Hayder Amer is a PhD student in Department of Automatic Control and Systems Engineering, University of Sheffield, UK. His research Interests include traffic control and vehicle re-routing in Vehicular Networks and Internet of Vehicles by using game theory and multi-objective optimization.

Lyudmila Mihaylova is Professor of signal processing and control at the Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield, U.K. Her research interests include the areas of machine learning and autonomous systems with various applications, such as navigation, surveillance, and sensor network systems. She is currently an Associate Editor of the IEEE TRANSACTIONS ON AEROSPACE AND ELECTRONIC SYSTEMS and of the Elsevier Signal Processing Journal.



Andrew H. Kemp is Senior Lecturer and Joint Programme Director with the Electronic and Electrical Engineering School, University of Leeds, UK. His research interests are in aspects of WSNs, in particular, related to localization, routing, and also multipath propagation studies to assist system development, cross-layer optimization, and the Internet of Things.