Fair Scheduling in IEEE 802.11ah Networks for Internet of Things Applications

L. Rajya Lakshmi and Biplab Sikdar
Department of Electrical and Computer Engineering
National University of Singapore, Singapore

***Abstract*—The IEEE 802.11ah standard has been developed to provide Internet access to a large number of devices in the Internet of Things (IoT) and machine-to-machine (M2M) networks. To handle contention from a large number of devices and reduce the collision probability, IEEE 802.11ah partitions nodes into groups by adopting a group-based MAC protocol. The formed groups may consist of nodes with different traffic patterns and hence, the data rate requirements of nodes in a group (and consequently the groups themselves) may not be uniform. To maximize the throughput while minimizing unfairness across groups, this paper formulates fair scheduling in IEEE 802.11ah networks as a multi-objective optimization problem. To maintain fairness among the nodes in a group, contention window size selection of nodes is formulated as an integer programming problem. Since it is difficult to solve these problems in real time, heuristic methods are also proposed. Performance of the proposed methods is evaluated in a dense IoT network and compared with the existing methods. As the number of nodes and groups increase, the proposed method consistently shows a superior performance in terms of fairness, throughput, delay, and power consumption, compared to the existing methods.**

I. Introduction

Internet-of-Things (IoT) has brought revolutionary changes to information and communication technology by facilitating connectivity and accessability to anything, anywhere, and anytime. Realization of IoT involves deployment of a large number of battery operated devices and connecting these devices to the Internet. On the other hand, machine-to-machine (M2M) networks that consist of smart meters and various sensors, are emerging as a potential solution to realize smart grids and smart cities. Cellular networks are not a cost-effective option to provide connectivity between Internet and IoT and M2M devices. Hence, other communication technologies such wireless personal area networks (WPANs), and low power wide area networks (LPWANs), etc. are under consideration. However, these access technologies can support only some of the IoT scenarios. On the other hand, IEEE 802.11 standard targets to achieve high throughput in wireless local area networks (WLANs), and not optimized for handling high contention from a large number of devices in IoT and M2M networks.

The IEEE 802.11ah Wi-Fi standard [1] has been developed to realize large-scale networks such as IoT and M2M networks, in the unlicensed sub-1 GHz band. With this standard, each access point (AP) can manage upto 8192 nodes, and can provide much higher range (upto 1 km) and throughput (150 kbps to 8 Mbps) than WPANs and LPWANs. To handle the high collision probability that is expected to cause severe performance degradation in IoT and M2M networks, IEEE 802.11ah standard proposes a restricted access window (RAW) mechanism, to restrict contention to a smaller number of devices. Nodes are partitioned into groups and the network time is divided into RAW slots. Each RAW slot is allocated to a group exclusively, and only the group members are allowed to access the channel in that RAW slot.

Groups are combinations of nodes (or sensors) and these nodes differ from each other in terms of their traffic generation characteristics. Hence, it is often the case that the data rate requirements of one group are quite different from that of others. To provide fair service to various applications coexisting in an IoT or a M2M network, it is important to allocate the available bandwidth to various groups in proportion to their data rate requirements. However, the existing literature on IEEE 802.11ah mainly handles issues such as grouping [2], [3], selection of optimal number of groups [4], etc, but not fair resource allocation to groups and nodes.

Fair scheduling is an important issue in all types of networks. In the literature, fair scheduling has been studied in both wireline and wireless networks [5], [6]. Most of the existing fair scheduling methods target per-flow fairness while assuming either perfect or partial per-flow information. However, lack of per-flow information at the scheduler prevents the direct applicability of the existing fair scheduling methods in IEEE 802.11ah networks. Hence, this paper aims to address fair scheduling in IEEE 802.11ah networks.

The main contributions of this paper are four fold:

- To maximize the network throughput while maintaining fairness across the groups, scheduling of RAW slots is formulated as a multi-objective optimization problem.
- Owing to the hardness of the optimal fair scheduling problem, a heuristic scheduling method is also proposed.
- To maintain fairness among the nodes within a group, optimal contention window size selection of nodes is formulated as an optimization problem, and a heuristic method is also proposed.
- Performance of the proposed methods is evaluated in a dense IoT network to simulate the effectiveness of the proposed methods in highly congested scenarios.

The rest of the paper is organized as follows. Section II gives an overview of the related research. Section III presents the system model of IEEE 802.11ah. Section IV-A formulates fair scheduling of groups as a multi-objective optimization problem and presents the proposed heuristic scheduling method and Section IV-B handles fair scheduling of nodes within a group. Simulation settings and the results are discussed in Section V. Section VI concludes the paper.

II. Related Work

The related research on IEEE 802.11ah networks covers various aspects such as grouping, RAW size selection, throughput analysis, power saving, etc. In [3], load balancing grouping in IEEE 802.11ah networks is formulated as an optimization problem, and then, to obtain a solution in real-time, the authors proposed a heuristic method that assigns sensors to groups in such a way that the channel utilization is maximized. To improve the network throughput, a random arbitration inter frame space number (AIFSN) based virtual station grouping method is proposed in [7]. Based on the selected AIFSN, the stations are divided into contending and non-contending stations and only the contending stations participate in contention in each cycle. In [4], based on the observed success probability, the AP estimates the number stations contending for uplink access. Then, by using the relationship between the number of contending devices and the RAW size, it determines the optimal RAW size.

In [2], two grouping mechanisms are proposed for IEEE 802.11ah networks. In the centralized grouping mechanism, the AP uniformly divides n sensors into K groups such that each group has (n/K) sensors. In the decentralized grouping mechanism that results in a small throughput loss, each sensor randomly chooses one among the K available RAW slots with probability $(1/K)$. Unlike the other studies that suggest the same size for all RAW slots, in [8], it is proposed to use different RAW slot sizes for groups with different sizes.

In contrast to all methods discussed above, this paper addresses fairness across the groups and among the sensors within each group. To maximize the throughput while minimizing the unfairness across the groups, RAW allocation is formulated as a multi-objective optimization problem. To maintain fairness among the nodes within a group, their success probabilities should be in proportion to their data requirements. Since the contention window of a node decides its success probability, optimal contention window size selection of nodes is formulated as an integer programming model. Since the formulated optimization problems are hard to solve in real-time, we also develop heuristic methods.

III. System Model

In an IEEE 802.11ah network, we have a number of sensors that want to access the Internet through an AP. With RAW based access mechanism, these sensors are partitioned into groups. The network time is divided into beacon periods, and each beacon period is further subdivided into a number of RAW slots, as shown in Figure 1. Each RAW slot b is allocated to a group g, and only the members of group g can contend for channel access in slot b. Hence, the collision probability decreases significantly. At the beginning of each beacon period, the AP transmits a beacon frame that contains information about the number of RAW slots allocated to each group, and the start time of allocation. These beacons can also carry information about which nodes belong to which group.

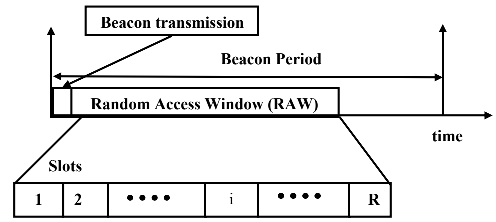

Fig. 1. Beacon period structure.

In a RAW slot, the sensor winning the last transmission opportunity may not complete its transmission by the end of the slot. Hence, beacon frame carries information about whether "cross slot boundary" is allowed or not. If it is not allowed, then each sensor accesses the medium, if and only if, the remaining duration in the allocated RAW slot is enough to complete the data transmission as well as the corresponding acknowledgment. Otherwise, a sensor can access the channel any time within the allocated RAW slot, even if its transmission opportunity crosses the allocated RAW slot boundary.

Inside their allocated RAW slots, sensors use enhanced distributed channel access (EDCA) of IEEE 802.11 to contend for channel access. When the medium is sensed to be idle for DIFS, each sensor selects a backoff counter uniformly from its contention window $[0, CW_\kappa]$, where CW_κ is the contention window size in the κ-th backoff stage. Then, the sensor backoffs and counts down as long as the channel remains idle. When the counter reaches 0, if the channel is still idle, then the sensor initiates a transmission. In the backoff state, if a sensor senses a busy channel, then it freezes the backoff counter and continues the countdown when the channel becomes free again. If the backoff counter of a sensor is more than 0 at the end of the allocated RAW slot, then, that sensor continues the countdown in the next beacon period.

We assume that the AP knows the packet generation rates and payload sizes of various sensors in the network [3]. The sampling rate and payload size of sensor (or node) s_i are represented by r_i and ρ_i, respectively. Assume that there exist k different payload and packet generation rate combinations. The weight of sensor s_i is set as: $\omega_{s_i} = \frac{\rho_i r_i}{\sum_{j=1}^{k} \rho_j r_j}$. Similar to sensors, each group has a weight. We are not handling the grouping issue here, and assume that the groups are already formed, following some criteria. The weight of group g_i is set as: $\omega^{g_i} = \frac{\mu_i}{\sum_{j=1}^{m} \mu_j}$, where μ_i is the cumulative weight of all nodes in group g_i and m is the number of groups. The cumulative data generated by s_i in a beacon period is computed as: $\rho_i r_i T_{BI}$, where T_{BI} is the duration of a beacon period. Similarly, the cumulative data generated by group g_j in a beacon period is computed as: $\sum_{s_i \in g_j} \rho_i r_i T_{BI}$. C_{s_i} and C^{g_j} represent the cumulative pending demands of sensor s_i and group g_j, respectively. From sampling rates and payload sizes of sensors, and the cumulative data received from each sensor, the AP can keep track of C_{s_i} and C^{g_j}, $\forall$ i, j.

The weight assignment of sensors and groups explained above is based on the amount of data that they generate per second. This is one possible way of weight assignment; weights can also be assigned based on other criteria such as the applications the sensors support, importance of the data the group members carry, and so on.

A. Analytical Background

Here we describe the normalized throughput of group g in a RAW slot. Let $s(t)$ be a stochastic process for the backoff stage at time t and $b(t)$ be the backoff counter of a given node. Now, assuming constant and independent collision probability p_c for each packet, the bi-dimensional process $\{s(t), b(t)\}$ can be represented as a DTMC as given in [9]. The transmission probability of a single node is:

$$p_t = \frac{2(1-2p_c)}{(1-2p_c)(CW_{min}+1) + p_c CW_{min}(1-(2p_c)^u)}, \tag{1}$$

where CW_{min} is the initial contention window size and u is the maximum number of retransmission attempts. In a RAW slot, a node encounters a collision if at least one of the remaining $(|g|-1)$ nodes also transmit, where $|g|$ is the cardinality of g. Thus, the collision probability is:

$$p_c = 1-(1-p_t)^{|g|-1}. \tag{2}$$

Eq. (1) and (2) can be solved using nonlinear methods. The probability of at least one node to initiate transmission in a slot is given as: $P_{tr} = 1-(1-p_t)^{|g|}$.

The success probability of group g can be computed by the probability that exactly one station transmits on the channel under the condition that at least one node initiates transmission, and is given as:

$$p_s(g) = \frac{|g| p_t (1-p_t)^{|g|-1}}{1-(1-p_t)^{|g|}}. \tag{3}$$

The success probability of a node s_i can be computed as:

$$p_s(s_i) = \frac{p_t (1-p_t)^{|g|-1}}{1-(1-p_t)^{|g|}}. \tag{4}$$

Now, the normalized throughput of g, T_g, can be obtained as:

$$\frac{p_s(g) P_{tr} P_l}{(1-P_{tr})\delta_t + P_{tr} p_s(g) su_l + P_{tr}(1-p_s(g)) co_l}, \tag{5}$$

where, P_l is the average payload size, su_l is the average length of a successful transmission, co_l is the average length of a collision, and δ_t is the duration of an idle slot. And, the normalized throughput of s_i, T_{s_i}, can be computed as:

$$\frac{p_s(s_i) P_{tr} P_l}{(1-P_{tr})\delta_t + P_{tr} p_s(s_i) su_l + P_{tr}(1-p_s(s_i)) co_l}. \tag{6}$$

IV. Fair scheduling in IEEE802.11ah Networks

Consider an IEEE 802.11ah network that consists of n sensors. Assume that these sensors are partitioned into K groups and we have R RAW slots available in each beacon period. The AP has to allocate these RAW slots to K groups in such a way that the throughput of the network is maximized and the unfairness across the groups is minimized. The RAW allocation can be represented in the form of a matrix G, which is of size $K \times R$. g_{it} (the element in the i-th row and t-th column of G) is set to 1, if the t-th RAW slot is allocated to group g_i, otherwise it is set to 0. Let A_{g_i} be the average throughput of group g_i per RAW slot, in the past, and RS_{g_i} be the number slots required to transmit C^{g_i} amount of data assuming the average throughput per RAW slot is A_{g_i}, that is, $RS_{g_i} = \lceil C^{g_i}/A_{g_i} \rceil$.

A. Fair scheduling Across Groups

The RAW allocation in IEEE 802.11ah networks can be formulated as a multi-objective optimization problem (**P1**) as follows:

$$\min_G \left[\sum_{i=1}^{K} \left(C^{g_i} - \Big(\sum_{t=1}^{R} g_{it} A_{g_i} \Big) \right), \right.$$
$$\left. \sum_{i=1}^{K} \sum_{j=1, j\neq i}^{K} \left| \frac{\sum_{t=1}^{R} g_{it} A_{g_i}}{\omega^{g_i}} - \frac{\sum_{t=1}^{R} g_{jt} A_{g_j}}{\omega^{g_j}} \right| \right], \tag{7}$$

such that

$$\sum_{t=1}^{R} g_{it} \begin{cases} \geq 0, \ \leq RS_{g_i}, & \text{if } C^{g_i} > 0 \\ = 0, & \text{if } C^{g_i} = 0 \end{cases} \forall\ i \tag{8}$$

$$g_{it} \in \begin{cases} \{0,1\}, & \text{if } C^{g_i} > 0 \\ \{0\}, & \text{if } C^{g_i} = 0 \end{cases} \forall\ i, t \tag{9}$$

$$\sum_{i=1}^{K} g_{it} = 1, \forall\ t, \sum_{i=1}^{K} \sum_{t=1}^{R} g_{it} \leq R. \tag{10}$$

Constraints (8) and (9) indicate that a group is allocated a RAW slot if and only if that group has some pending demand and the number of slots allocated to a group is upper bounded by the number of slots required to clear its pending demand. Constraint (10) indicates that each RAW slot is allocated to at most one group, and the total number of RAW slots allocated to all groups is at most the number of available RAW slots. Problem (**P1**) is hard to solve in real time, since it is a variant of *the exact cover problem*, which is NP-complete. Hence, we develop a heuristic method to obtain a schedule in real-time.

1) Heuristic Fair Allocation for Groups: To perform fair scheduling, the AP maintains the cumulative service received by each group g in a variable W_g. Whenever the AP receives a packet from a member of g, it increments W_g by the size of the received packet. Each RAW slot is allocated to the group that receives the least normalized cumulative service, compared to the other groups. To identify such a group, for each group g, its normalized cumulative service (Υ_g) is computed as: $\Upsilon_g = \frac{W_g}{\omega^g}$. Then, among all groups, the group g_{min} with the least normalized service is selected and a RAW slot is allocated to g_{min}. Now, we expect that g_{min} will receive $A_{g_{min}}$ service in the allocated RAW slot, and temporarily increment the received service of g_{min}, $W_{g_{min}}$, by $A_{g_{min}}$. If a RAW slot is allocated to a group for the first time, then we do not have information about its average per-slot throughput. Hence, its cumulative service is incremented

Algorithm 1 Fairness across groups

```
BEGIN:
initialization: G_1 is the list of groups with pending demands;
for i = 1 to |G_1| do
    Ŵ_{g_i} = W_{g_i}; ϒ_{g_i} = Ŵ_{g_i} / ω^{g_i};
end for
for i = 1 to R do
    Find the group g with the least ϒ_g value;
    Ŵ_g += A_g; ϒ_g = Ŵ_g / ω^g;
end for
END;
```

by its normalized throughput computed using (5). Now, using the updated cumulative service of g_{min}, its normalized cumulative service is recomputed. The above explained process is repeatedly executed until each RAW slot is allocated to some group. The pseudo-code for this method is given in Algorithm 1. The worst case complexity of this algorithm is $O(|G_1|R)$.

B. *Fair Scheduling of Nodes in a Group*

The method explained above handles fair allocation across the groups, but not fair allocation among the sensors in a group. Intuitively, the sensors with a higher weight should get more access to the channel than the sensors with a lower weight. Hence, to maintain fairness among the sensors, their success probabilities should be in proportion to their weights [10]. An important parameter that decides the success probability of a sensor is its contention window size. Hence, by selecting optimal contention window sizes for various sensors, fairness can be maintained among them. The initial contention window sizes of all sensors in group g_1 can be represented as a vector **cw**. Now, the optimal contention window size selection for the sensors in g_1 can be formulated as a multi-objective optimization problem (**P2**) as follows:

$$\min_{\mathbf{cw}} \left[\sum_{i=1}^{|g_1|} \left(C_{s_i} - \left(T_{s_i} \right) \right), \sum_{i=1}^{|g_1|} \sum_{j=1, j \neq i}^{|g_1|} \left| \frac{T_{s_i}}{\omega_{s_i}} - \frac{T_{s_j}}{\omega_{s_j}} \right| \right], \quad (11)$$

such that

$$\frac{P_{tr} p_s(s_i)}{\omega_{s_i}} = \frac{P_{tr} p_s(s_j)}{\omega_{s_j}}, \forall\, i, j. \quad (12)$$

The objective function seeks to minimize the pending traffic while ensuring fairness among the sensors. Constraint (12) indicates that the success probability of a sensor, which is mainly affected by its contention window size, should be in proportion to its weight.

Problem ($\mathbf{P2}$) is a non-linear integer programming problem, and it is hard to solve in real-time. Hence, we develop a heuristic method that improves the success probability of the nodes that have received a lower normalized service than others. Similar to groups, the AP maintains the cumulative service received by each sensor s in U_s. Also, each sensor keeps track of the cumulative service it has received. Now, consider a group g_1 to which R_1 RAW slots are allocated in a frame, where $R_1 \geq 1$. For each sensor $s_i \in g_1$, the AP computes its normalized cumulative service as: $\upsilon_{s_i} = \frac{U_{s_i}}{\omega_{s_i}}$. Let $s_{max} \in g_1$ be the sensor that has received the highest normalized service in g_1, and $s_{min} \in g_1$ be the sensor that has received the least normalized service. If a fair allocation had taken place, then in proportion to the cumulative service received by s_{max}, the cumulative service received by s_{min} should have been $\frac{U_{s_{max}} \omega_{s_{min}}}{\omega_{s_{max}}}$. But, the cumulative service received by s_{min} is $U_{s_{min}}$. Hence, the extra service that s_{min} should receive in comparison to s_{max} is: $\upsilon_{diff} = (\frac{U_{s_{max}} \omega_{s_{min}}}{\omega_{s_{max}}} - U_{s_{min}})$.

To reduce unfairness among the nodes in g_1, s_{min} alone or along with some other nodes that have received a lesser service than s_{max}, use a smaller contention window size than the normal one. Let the set of sensors that use a reduced contention window size be a_{g_1}. Initialize a_{g_1} with s_{min}. Now, using (6), the throughput of s_{min}, $T_{s_{min}}$, is computed while assuming s_{min} transmits exclusively in R_1 slots. If $T_{s_{min}} \leq \upsilon_{diff}$, then *only* s_{min} reduces its contention window size, by setting CW_{max} as xCW_{min}, where x is a constant. Otherwise, the other sensors that use a reduced contention window size are identified according to the following process.

To identify the other members of a_{g_1}, first we fix the contention window size of s_{min} as follows. When s_{min} transmits exclusively in R_1 slots, then its expected normalized throughput can be expressed as given in (6). We constrain this throughput to be at least υ_{diff} and compute the maximum possible value of CW_{min}, cw^{max}. That is, we solve $\frac{R_1 R_d p_s(s_{min}) P_{tr} P_l \gamma}{(1-P_{tr})\delta_t + P_{tr} p_s(s_{min}) su_l + P_{tr}(1-p_s(s_{min}))co_l} \geq \upsilon_{diff}$ for cw^{max}, where R_d is the duration of a RAW slot (in seconds) and γ is the transmission rate. Then, from the other sensors in g_1, the sensor s_1 with the least υ value is selected and temporarily included in a_{g_1}. Now, CW_{min} is set as cw^{max}, and the normalized throughput of s_{min} in R_1 slots is computed when all sensors in a_{g_1} contend for access. If this throughput is less than υ_{diff}, then s_1 is removed from a_{g_1}, and a_{g_1} is finalized. Otherwise, s_1 remains in a_{g_1}, and the other sensors in g_1 are considered in the increasing order of their υ values and included in a_{g_1}, provided their inclusion does not result in the throughput of s_{min} in R_1 slots to drop below υ_{diff}. After finalizing a_{g_1}, the AP includes the following information in a beacon: the slots allocated to g_1, cw^{max}, and the normalized cumulative service (υ_l) of the last sensor that was included in a_{g_1}. The same process is repeated with the other scheduled groups, and then the AP transmits the beacon. After receiving the beacon, a sensor $s_1 \in g_1$ for which $\upsilon_{s_1} \leq \upsilon_l$, sets its CW_{min} and CW_{max} as cw^{max} and xcw^{max}, respectively, for the current beacon period. The pseudo-code for this method is given in Algorithm 2, and its worst case complexity is $O(|g_1|)$. We call the heuristic method for fair allocation across groups and this method together as "group fair and sensor fair scheduler (GF-SFS)".

V. Simulation Results

For performance evaluation, the IEEE 802.11ah module for ns-3 (version 3.23) [11] is used. Performance of GF-SFS is

Algorithm 2 Fairness among the sensors within group g_1

BEGIN:

$a_{g_1} = \phi$;
for $j = 1$ to $|g_1|$ **do**
  $\upsilon_{s_j} = \frac{U_{s_j}}{\omega_{s_j}}$;
end for
Find the node s_{min} with the least υ value;
Find the node s_{max} with the highest υ value;
$\upsilon_{diff} = \upsilon_{s_{max}}\omega_{s_{min}} - U_{s_{min}}$;
push s_{min} onto a_{g_1}; $\upsilon_l = \upsilon_{s_{min}}$; $cw^{max} = CW_{min}$;
$T^e_{s_{min}}$ be the expression that denotes the throughput of s_{min} when it transmits solely in the slots allocated to g_1; $Eval(T^e_{s_{min}})$ be its value;
if $Eval(T^e_{s_{min}}) > \upsilon_{diff}$ **then**
  Solve equation ($T^e_{s_{min}} \geq \upsilon_{diff}$) for the highest CW_{min}, assign the result to cw^{max}; $g_1 = g_1$ - $\{s_{min}\}$;
  while g_1 $! = \phi$ **do**
    select s_j with the least υ value;
    temporarily push s_j onto a_{g_1} and compute the throughput of s_{min}, $T_{s_{min}}$, when all nodes in a_{g_1} contend for channel access in the slots allocated to g_1;
    if $T_{s_{min}} > \upsilon_{diff}$ **then**
      $\upsilon_l = \upsilon_{s_j}$; $g_1 = g_1$ - $\{s_j\}$;
    else
      $a_{g_1} = a_{g_1}$ - $\{s_j\}$; break;
    end if
  end while
end if
Output a_{g_1}, cw^{max}, and υ_l;

END;

TABLE I
MAC PARAMETERS FOR SIMULATIONS.

Parameter	Value
Frequency	900 MHz
Node distribution	Random
CW_{min}	15
CW_{max}	1023
AIFSN	3
MAC header	legacy header
RTS/CTS	not enabled
Cross slot boundary	enabled
Wi-Fi mode	MCS2, 2Mhz
Number of slots per group	variable

evaluated in an IoT network where the number of sensors vary between 50 and 500, and the number of groups vary between 5 and 50. Depending upon their packet generation rates and packet sizes, the sensors can be divided into four service classes (class-I to class-IV) with even cardinalities. The respective packet sizes of the sensors belonging to these service classes are 256, 256, 512, and 128 bytes. Two traffic generation scenarios are considered for evaluation. In the first scenario called "saturated mode", the cumulative traffic generated by all sensors is around the maximum throughput of the channel. In the second scenario called "overloaded mode", the cumulative traffic generated is much more than the capacity of the channel. Sensors are partitioned into K groups uniformly randomly. The "load balancing group formation (LBGF)" method [3], and "random grouping (RAND)" where sensors are partitioned into groups randomly, are considered for performance evaluation of GF-SFS. Some more simulation parameters are given in Table I. su_l and co_l are computed using the expressions given in [9].

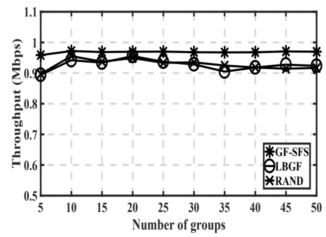

Fig. 2. Number of groups vs network throughput.

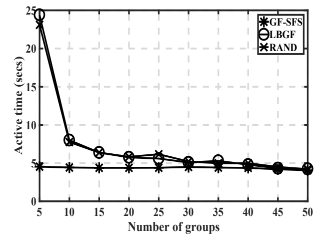

Fig. 3. Number of groups vs active time of class-I.

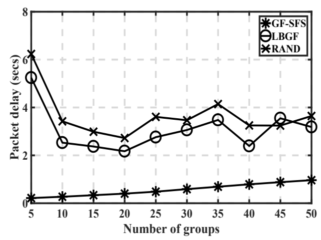

Fig. 4. Number of groups vs packet delay of class-I.

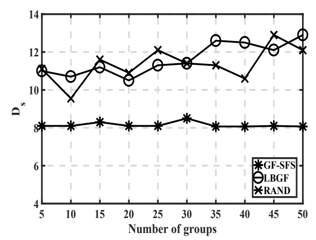

Fig. 5. Number of groups vs highest D_s.

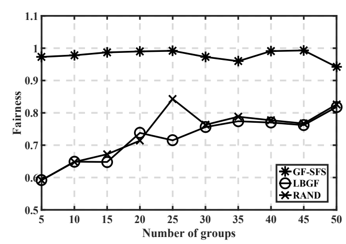

Fig. 6. Number of groups vs fairness (class-I).

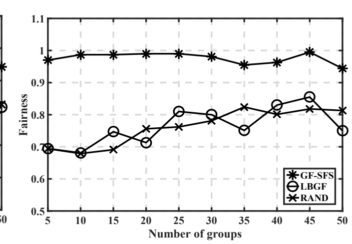

Fig. 7. Number of groups vs fairness (class-II).

When fair allocation takes place in the network, as time progress, the ratio of the cumulative services of sensors s_i and s_j should approach the ratio of their weights [5]. The deviation in the service ratio of these sensors is computed as:

$$D_s = \frac{U_{s_i}}{U_{s_j}} - \frac{\omega_{s_i}}{\omega_{s_j}}. \quad (13)$$

The maximum D_s among all sensor pairs, network throughput, node active time, packet delay, and fairness [12] are considered as the performance evaluation metrics. Each simulation runs for 300 seconds, and the presented results are averaged over 10 simulation runs.

A. Saturated Mode

In this set of simulations, the number of sensors is fixed at 500, and the number of groups increases from 5 to 50. The sampling rates (in Hz) of the sensors of classes I to IV are 1, 0.4, 1, and 0.8, respectively. During RAW allocation, GF-SFS gives priority to the groups that have received a lesser normalized cumulative service than others. In addition, the sensors that have received a lesser normalized cumulative service than others are given prioritized access to the channel by decreasing their contention window sizes. As a result, the collision probability within each scheduled group reduces and consequently, GF-SFS obtains a better throughput, compared to LBGF and RAND, as shown in Figure 2. Due to reduction in the collision probability in each scheduled group, the active time of each node and packet delay using GF-SFS are lower, compared to LBGF and RAND, as shown in Figures 3 and 4. For a fixed number of sensors, as the number of groups increases, the collision probability decreases. Thus, with increasing number of groups, the node active times of the three methods gradually converge. As the number of groups increases, the duration of beacon period increases, and the frequency with which a node can access the channel decreases. As a result, we observe a slightly increasing trend in the packet delay with GF-SFS, as the number of groups increases. GF-SFS maintains a reasonably good fairness among the sensors even when a large number of sensors are contending for access. Hence, its D_s values shown in Figure 5 are much better and consistent, compared to LBGF and RAND.

B. Overloaded Mode

In this set of simulations, the fair allocation capability of the methods is evaluated when the network is overloaded. The traffic generation rates (in Hz) of the sensors of classes I to IV are 5, 1.5, 5, and 1, respectively. Two cases are considered for performance evaluation: in case-I, the number of sensors is fixed at 500 and the number of groups increases from 5 to 50; and in case-II, the number of groups is fixed at 50 and the number of sensors increases from 50 to 500. The fairness results of classes I and II in case-I are shown in Figures 6 and 7, respectively. Similar results are observed for classes III and IV, and not shown here due to space limitations. With varying number of groups, the contention within each group varies. However, due to its fair allocation across groups and within each group, GF-SFS consistently maintains the fairness of each service class at a higher level, compared to LBGF and RAND. This observation is true in case-II also, as shown in Figures 8 and 9. Figures 10 and 11 show the network throughput of three methods in case-I and II, respectively. In case-I, when the number of groups vary between 5 and 30, the contention within each group is very high. In such high contention situations, GF-SFS achieves a significantly better throughput, compared to LBGF and RAND. In case-II, owing to a large number of groups, a small number of sensors contend for channel access in each group. Thus, the improvement in throughput with GF-SFS is slightly lower is case-II than in case-I.

VI. Conclusions

The IEEE 802.11ah standard is capable of supporting large-scale networks such as IoT and M2M networks. For fair bandwidth allocation across the groups of nodes, this paper formulates RAW allocation as a multi-objective optimization problem. Also, fairness maintenance among the sensors in each group is modeled as an integer programming problem. Then, heuristic methods with a lower complexity are developed to achieve a fair allocation in real time. Results from extensive simulations conducted in a dense IoT network show that the proposed scheduler achieves significantly higher fairness and throughput, compared to the existing methods, as the number of sensors and groups increase.

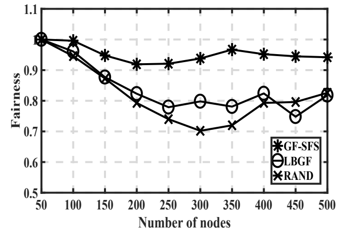

Fig. 8. Number of nodes vs fairness (class-I).

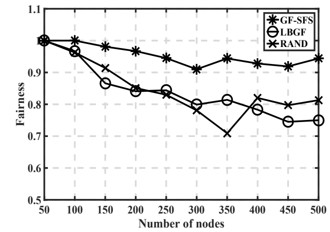

Fig. 9. Number of nodes vs fairness (class-II).

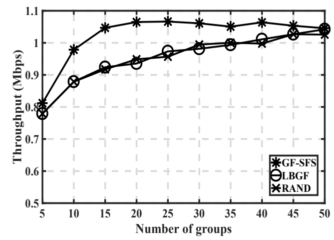

Fig. 10. Number of groups vs network throughput.

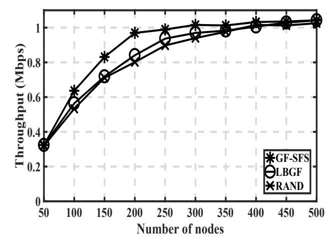

Fig. 11. Number of nodes vs network throughput.

References

[1] IEEE 802.11ah task group, tgah functional requirements and evaluation methodology rev. 5. available [online].

[2] L. Zheng, M. Ni, L. Cai, J. Pan, C. Ghosh, and K. Doppler, "Performance analysis of group-synchrinized DCF for dense IEEE 802.11 networks", *IEEE Trans. on Wireless Comm.*, vol. 13, no. 11, pp. 6180-6192, 2014.

[3] T. C. Chang, C. H. Lin, K. C. J. Lin, and W. T. Chen, "Load-balancing sensor grouping for IEEE 802.11ah networks", *Proc. IEEE GLOBECOM*, pp. 1-6, 2015.

[4] C. W. Park, D. Hwang, and T. J. Lee, "Enhancement of IEEE 802.11ah MAC for M2M communications", *IEEE Communications Letters*, vol. 18, no. 7, pp. 1151-1154, 2014.

[5] H. Luo and S. Lu, "A topology-independent fair queueing model in ad hoc wireless networks", *Proc. IEEE ICNP*, pp. 325-335, 2000.

[6] H. Luo, S. Lu, and V. Bharghavan, "A new model for packet scheduling in multihop wireless networks", *Proc. ACM MOBICOM*, pp. 76-86, 2000.

[7] K. Ogawa, Y. Sangenya, M. Morikura, K. Yamamoto, and T. Sugihara, "IEEE 802.11ah based M2M networks employing virtual grouping and power saving methods", *In Proc. IEEE VTC*, pp. 1-5, 2013.

[8] N. Nawaz, M. Hafeez, S. A. R. Zaidi, D. C. McLemon, and M. Ghogho, "Throughput enhancement of restricted access window for uniform grouping scheme in IEEE 802.11ah", *In Proc. IEEE ICC*, pp. 1-7, 2017.

[9] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function", *IEEE J. Sel. Areas. Commun.*, vol. 18, no. 3, pp. 535-547, 2000.

[10] D. Qiao and K. G. Shin, "Achieving efficient channel utilization and weighted fairness for data communications in IEEE 802.11 WLAN under the DCF", *In IEEE QoS*, pp. 227-236, 2002.

[11] L. Tian, S. Deronne, S. Latré, and J. Famaey, "Implementation and validation of an IEEE 802.11ah module for ns-3", *Proc. Workshop on ns-3*, pp. 49-56, 2016.

[12] R. Jain, A. Durresi, and G. Babic, "Throughput fairness index: an explanation", *ATM Forum Document Number: ATM Forum/990045*, Feb. 1999.