QoS Constrained Semi-Persistent Scheduling of Machine Type Communications in Cellular Networks

Goksu Karadag, Recep Gul, Yalcin Sadi and Sinem Coleri Ergen

***Abstract*—The dramatic growth of machine-to-machine (M2M) communication in cellular networks brings the challenge of satisfying the Quality of Service (QoS) requirements of a large number of M2M devices with limited radio resources. In this paper, we propose an optimization framework for the semi-persistent scheduling of M2M transmissions based on the exploitation of their periodicity with the goal of reducing the overhead of the signaling required for connection initiation and scheduling. The goal of the optimization problem is to minimize the number of frequency bands used by M2M devices to allow fair resource allocation of newly joining M2M and human-to-human communications. The constraints of the problem are delay and periodicity requirements of M2M devices. We first prove that the optimization problem is NP-hard, then propose a polynomial-time heuristic algorithm employing a fixed priority assignment according to the QoS characteristics of devices. We show that this heuristic algorithm provides an asymptotic approximation ratio of 2.33 to the optimal solution for the case where the delay tolerances of devices are equal to their periods. Through extensive simulations, we demonstrate that the proposed algorithm performs better than the existing algorithms in terms of frequency band usage and schedulability.**

***Index Terms*—scheduling, machine type communications, cellular networks, QoS constraints**

I. Introduction

M2M communication is defined as the conveyance of sensing and actuation data among machines to perform sensing, processing, decision making and acting on decisions without any human supervision in the communication cycle. Total automation of devices without including human effort in mobile communication together with the advancement in inexpensive sensors and devices have led to a variety of applications in smart grid, smart city, smart home, vehicular telematics, health services and industrial environment [1], [2]. These applications mostly require seamless coverage over a large area to facilitate the communication of M2M devices and M2M servers in different network domains. Therefore, cellular network will be widely used in collecting M2M data. M2M devices are expected to be connected to the cellular network either directly or through M2M gateways that collect data from M2M devices using short-range technologies. By 2019, more than 40% of all connected devices are projected to be M2M devices [3]. Almost all existing M2M applications are based on GPRS due to the advantages of low device cost, high geographic coverage, international interoperability and immediate business entry [4]. However, the limited capacity of GPRS cannot support large number of M2M devices expected to be deployed in the near future. Dedicated M2M cellular architectures, such as SigFox and LORA [5], are built to provide high coverage with very low cost connectivity and long battery lifetime. However, they can only support very low throughput, on the order of a few bytes per minute. Thus, exploiting existing LTE infrastructure and providing a native support in 5G for fast growing M2M services is of paramount importance.

The conventional connection-oriented data communication in LTE requires a user equipment (UE) in idle mode to make a connection before sending data to the base station (BS). The UE first initiates random access procedure by transmitting a randomly selected preamble, out of all preambles with equal probability, to the BS. The BS responds with a random access response, including the identity of the detected preamble and an initial uplink resource grant for the transmission of a connection setup request message. Upon reception of random access response, the UE sends connection setup request message by using the initial uplink resource grant. The BS then responds with the connection setup response message to the UE. If UE succeeds in the random access procedure, it switches to the connected mode, sends a scheduling request and buffer status report to the BS and receives an uplink grant from the BS for sending the data at the higher layers. Following the

Goksu Karadag and Sinem Coleri Ergen are with the department of Electrical and Electronics Engineering, Koc University, Istanbul, Turkey, e-mail: {gkaradag16,sergen}@ku.edu.tr. Recep Gul is with the department of Information Technology and Electrical Engineering, ETH Zurich, Zurich, Switzerland, e-mail: guelr@nari.ee.ethz.ch. Yalcin Sadi is with the department of Electrical and Electronics Engineering, Kadir Has University, Istanbul, Turkey, e-mail: yalcin.sadi@khas.edu.tr.
Sinem Coleri Ergen acknowledges the financial support by the Turkish Academy of Sciences (TUBA) within the Young Scientist Award Program (GEBIP) and METU-Prof. Dr. Mustafa Parlar Foundation Research Encouragement Award.

transmission of the data, the UE is disconnected from the BS.

The usage of M2M communications in LTE networks optimized for human-to-human (H2H) communications results in efficiency and congestion problems [6]. First, M2M devices generate small amount of data, in contrast to H2H communication with high data rates. The size of the signaling packets used in the random access procedure and at higher layers is much larger than that of the payload to be sent by M2M devices, resulting in low efficiency. Second, the number of M2M devices within a cell can be significantly large and high number of M2M devices may be activated simultaneously by an external event. The large number of M2M devices trying to access eNodeB within a short period of time results in severe congestion. Third, the uplink-to-downlink ratio for M2M devices is much larger than that of H2H communications. The large size of signaling packets required to request the transmission of small size data packets again decreases the efficiency of the network. Fourth, M2M devices usually require high energy efficiency due to battery dependent operation and wide range of quality-of-service (QoS) performance in terms of delay and reliability. The lack of consideration of these constraints in LTE results in suboptimal performance. Finally, M2M devices generate data at mostly predetermined times, in mostly periodic manner, at predetermined locations with low or no mobility as opposed to highly mobile and unpredictable H2H devices, such as smart phones. These features have been exploited only in a limited manner in the literature.

Up to now, several M2M communication studies have focused on increasing the success rate and decreasing the delay of M2M devices due to the high number of accessing devices. These works mostly analyze and optimize the candidate 3GPP solutions for controlling RAN overload [7], including access class barring (ACB), random access resource separation, M2M specific backoff, slotted access methods. ACB methods aim to minimize the congestion for the higher priority devices by the optimization and transmission of the ACB related parameters, including a probability factor and barring timer for different classes, by eNodeB in case of network load [7], [8], [9], [10]. Devices start random access procedure with a probability factor corresponding to their classes, and perform random backoff, while considering the barring timer value, otherwise. Random access resource separation methods either split the available preambles or allocate different random access slots to M2M and H2H devices, mostly to minimize the effect of high number of M2M devices on H2H devices [7], [8]. M2M specific backoff schemes reduce the overload by assigning different backoff timers to M2M and H2H devices, with the goal of spreading access attempts of devices in time in order to reduce congestion [8], [11]. Slotted access method is based on the assignment of dedicated random access slots to M2M devices based on their identity and RA cycle parameter broadcast by eNodeB [7]. However, the usage of large RA cycles in the case of overload may lead to large delays. Apart from 3GPP solutions, novel mechanisms have also been proposed to solve overload control problem introduced by massive M2M accesses, e.g. [12], [13], [14], [15]. [12] proposes novel preamble collision resolution rather than collision avoidance for massive number of M2M devices. If the preamble of an M2M device has collided, the collision resolution ensures the random access reattempts from a reserved set of preambles. The rate of collision is used in the optimization of the number of preambles in each reserved set. [13], [14] embed the transmission of the small M2M data into the random access process by attaching data to connection setup request message in the third stage to minimize delay. [15] proposes a self-optimization framework to achieve maximum M2M throughput based on the adaptation of the resource block composition and access barring parameter according to the amount of available resource blocks and M2M traffic load. All of these methods aim to improve the delay, throughput and success rate of M2M transmissions, however fail to provide any QoS guarantees as they employ a random access procedure. Moreover, none of these schemes exploit the periodicity of M2M communications to reduce the random access overhead.

Another set of prior studies focus on the dynamic scheduling of M2M communications at each Transmission Time Interval (TTI) based on the assumption that BS knows their channel conditions, data backlog states and delay tolerance. [16], [15], [17], [18], [19] propose an uplink scheduling algorithm, considering both channel condition and maximum delay tolerance. [20] proposes a delay dependent scheduling based on giving higher priority to the M2M devices exceeding their delay threshold until they are served. None of these works, however, consider the delay and signaling overhead of random access and periodic update of channel conditions and UE buffer status over some dedicated control channel in the evaluation of the performance of these scheduling algorithms [21]. Dynamic scheduling of small packets are expected to cause substantial control signaling overhead. Furthermore, these studies do not exploit periodicity of MTC devices.

Semi-persistent scheduling has been proposed for Voice over Internet Protocol (VoIP) in the past [22], [23], [24], [25], recently being extended for M2M communications [26], [27] exploiting the predetermined periodic nature of M2M communications. Semi-persistent scheduling is based on the allocation of a sequence

of TTI-resource unit chunks and fixed modulation to a UE for a certain amount of time. Unlike persistent scheduling that also preallocates resources for retransmissions, which may cause mismatch between allocated and actually needed resources, semi-persistent scheduling adopts dynamic scheduling for retransmissions on unused resource units. This fixed allocation both guarantees meeting QoS requirements and decreases control signaling overhead in downlink control channel and uplink random access. [23], [25] demonstrates the higher performance of semi-persistent scheduling of VoIP compared to dynamic scheduling. [22] examines the feasibility of semi-persistent scheduling with initial random access to determine the VoIP capacity as a function of the maximum number of VoIP terminals that can be allowed provided that their random access delay does not exceed a predefined delay constraint. The extension of semi-persistent scheduling for M2M communications requires considering their differentiating features from VoIP, including less dynamic nature and wider range of QoS requirements. M2M devices generate traffic mostly with the same period over a longer duration at a fixed location, in contrast to VoIP calls arriving randomly with short durations. Besides, the packet generation periods varies in the range between 1 ms and several minutes for M2M communications compared to $10 - 40$ ms for multimedia applications, requiring novel scheduling algorithms.

Semi-persistent scheduling algorithms developed for M2M communications aim to meet the QoS constraints of M2M devices over a wide range [26], [27], [28]. The basic idea is to group M2M devices based on their QoS characteristics, including packet arrival rate, maximum tolerable jitter and acceptable probability of jitter violation. The scheduling algorithm in [27] considers the case with zero acceptable jitter violation probability. The algorithm assigns an allocated access grant time interval (AGTI) to the clusters according to their priority, at the beginning of their packet generation period. Each AGTI comprises L allocation units. Each M2M device in each cluster is assigned one allocation unit to transmit at most one packet in the corresponding AGTI. If AGTIs for different clusters overlap, the AGTI of the cluster with lower priority is delayed. This scheduling is extended with the additional opportunistic scheduling of the clusters with nonzero acceptable jitter violation probability in [26] and scheduling of Poisson modeled event-driven traffic in [28]. Since these studies allocate the entire AGTI to a cluster, the M2M devices in the lower priority clusters may suffer from high delay and may not even meet their jitter constraints in the presence of massive M2M deployment. Moreover, these algorithms do not consider the adverse effect of AGTI based scheduling on H2H communications.

Radio resource allocation schemes should address the effective partition of resources between M2M and H2H communications so that QoS requirements of both can be satisfied. Such co-existence has only been considered for dynamic scheduling algorithms in a limited context. Most scheduling algorithms give strict priority to H2H over M2M without providing any QoS guarantee for M2M devices [18], [29]. A solution for this problem is to give high priority to voice, video, M2M services of real-time communication over normal priority traffic such as buffered video, data services and M2M non-real time data services [30], [31], or allocate both H2H and M2M using utility based scheduling [32]. All of these works adopt dynamic scheduling without considering the associated signaling overhead. The extension to semi-persistent scheduling is an open problem. Moreover, these studies do not provide any QoS guarantees exploiting the periodic nature of M2M transmissions.

In this paper, we propose a novel semi-persistent scheduling algorithm to guarantee satisfying the delay requirements of periodic real-time M2M communication with minimum usage of the frequency spectrum. The proposed framework aims to achieve fair allocation of radio resources between M2M and H2H communication by making efficient use of the scarce spectrum and exploiting the unique characteristics of M2M communication while satisfying their QoS requirements. Frequency spectrum minimization is introduced for the first time in the literature with the goal of minimizing the effect of real-time M2M devices on newly arriving or non-real time M2M and H2H applications. The original contributions of the paper are as follows:

- We provide a semi-persistent scheduling framework based on the persistent scheduling of the periodic M2M communication to meet their maximum tolerable jitter constraints, inclusion of newly arriving periodic real-time M2M communication via a call admission control algorithm and dynamic scheduling of event-triggered M2M and H2H considering the priorities among them.
- We formulate the radio resource allocation with the objective of minimizing the number of frequency bands used by the real-time periodic M2M devices while meeting their stringent timing requirements as a binary integer programming problem. We prove that the optimization problem is NP-hard. Frequency band minimization problem is introduced for the first time in the literature.
- We propose an efficient fathoming based smart enumeration search algorithm, called Efficient Depth-First Search Algorithm (EDFS), to obtain the optimal solution. The algorithm is based on the depth-first search method for branch and bound technique. Although this decreases the runtime compared to

binary integer programming formulation, it still requires an exponential runtime in the number of M2M devices.

- We propose a polynomial time heuristic algorithm, called Minimum Frequency First-Fit Allocation (MFFFA) Algorithm, for the radio resource allocation problem of M2M devices. The main feature of the proposed algorithm is to employ a priority assignment based on the transmission period of devices and allocate as many devices as possible to a frequency band as long as the delay requirement of each device is satisfied. We provide the worst case performance of the proposed algorithm with respect to optimal solution under certain conditions.
- We propose a call admission mechanism to effectively manage the admission of new devices. We formulate an optimization problem with the goal of serving maximum number of devices while satisfying the QoS requirements of both existing and newly arriving devices. We prove that the resulting problem is again NP-hard. We then propose a polynomial time heuristic algorithm, called First Fit Occupied Bands (FFOB) Algorithm, based on the principle that the frequency band should be used efficiently to serve as many devices as possible.
- The superior performance of the proposed algorithms compared to previously proposed efficient random access methods and persistent scheduling algorithms has been demonstrated for different number of devices and varying delay requirement values via extensive simulations.

The rest of the paper is organized as follows. Section II describes the system model and assumptions. Section III describes the semi-persistent scheduling framework. Section IV provides the formulation of the optimization problem and the proof of its NP-hardness. Section V describes the proposed efficient smart enumeration based exponential time search algorithm. Section VI presents the proposed polynomial time heuristic radio resource allocation algorithm and the analysis of its worst case performance under certain conditions. Section VII gives the call admission control scheme. Section VIII provides the performance evaluation of the proposed resource allocation algorithm. Finally, Section IX concludes the paper.

II. SYSTEM MODEL AND ASSUMPTIONS

The system model and assumptions are detailed as follows.

1) We consider a cellular network with a base station serving a large number of M2M devices with diverse traffic characteristics and H2H devices, as shown in Fig. 1-a. Most M2M devices are time-triggered, generating data periodically, with period

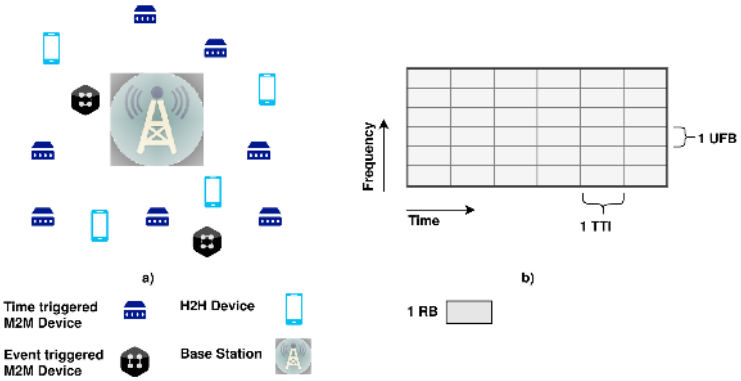


Fig. 1: a) System architecture, b) Time-frequency resource allocation

p_i for node i [33]. Example applications include smart grid, e-health [34], intelligent transportation [35], [36] and industrial supply systems [33]. There may be further event-triggered M2M devices and H2H devices generating data at random times.

2) Each device is allocated time-frequency resource elements called *Resource Blocks* (RBs), as shown in Fig. 1-b. In LTE, a resource block is a time-frequency unit with 0.5 ms time duration and 180 kHz bandwidth. The length of minimum scheduling unit [37] is an integer multiple of resource block length, thus providing a time granularity for scheduling. RB-based granularity is expected to be preserved in 5G cellular networks, even though the size of an RB may change [38]. For periodic data generating devices, multiple RBs may be allocated in a period but these RBs do not have to be allocated consecutively.
3) We define Unit Frequency Band (UFB) as the frequency band of 1 RB width, as shown in Fig. 1-b. Each device is assigned to one UFB; i.e. once a device is allocated to a UFB, all packets of that device will be transmitted on that particular UFB. This partitioned scheduling is preferred as it provides lower scheduling overhead without allowing the packets of the same device to migrate to the different bands [39].
4) The QoS (Quality of Service) requirement of M2M devices is represented by maximum allowable delay that we call *delay tolerance*, denoted by d_i for node i. Satisfying delay requirements is critical especially in safety critical operations such as navigational data communications [40], health care applications [41], and real-time control systems [42].
5) Time-triggered M2M devices are assigned priorities in the decreasing order of their periods, denoted by p_i for node i; i.e. a lower period implies a higher priority. Devices with lower periods have lower delay tolerances (since a packet must be

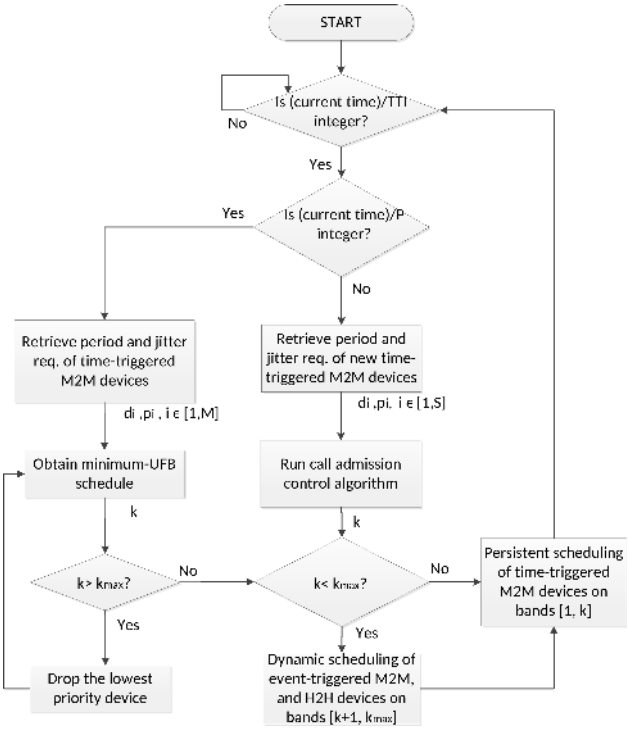


Fig. 2: Scheduling Framework

transmitted before the next packet is generated), thus giving priority to low period devices ensures that their strict delay requirements are satisfied. In the case of equality of periods, devices with lower delay tolerances are prioritized. If delay tolerances are also equal, then devices with higher transmission times are prioritized. If transmission times are also equal, they are randomly assigned priorities.

6) Time-triggered M2M devices are given priority within a certain number of UFBs, denoted by k_{max}. k_{max} may be determined according to traffic density, number of devices or channel condition. Once time-triggered M2M devices are allocated within k_{max} bands, event-triggered M2M and H2H devices can be scheduled if resources are available. The scheduling outside these k_{max} bands where time-triggered M2M devices are not allocated is out of scope of this paper.

III. SCHEDULING FRAMEWORK

The base station uses semi-persistent scheduling to allocate resources to time-triggered M2M devices, and dynamic scheduling to include event-triggered M2M and H2H devices in the schedule. Semi-persistent schedule is regenerated with period P, much larger than the update period of dynamic scheduling, denoted by transmission time interval (TTI). Semi-persistent scheduling is enabled by the fact that the data generation times of time-triggered M2M devices are pre-known given their data generation period. Since the signaling required to initiate connection and request resources for the transmission of data is eliminated, both signalling and scheduling overhead is reduced. There may still be time-triggered M2M devices joining and leaving the network between the regeneration times of the semi-persistent schedule. For those arriving the network, call admission algorithm is executed, considering both existing and newly arriving devices. The time-triggered M2M devices leaving the network are excluded from the schedule.

The proposed scheduling framework is shown in Fig. 2. Semi-persistent schedule is updated regularly with period P while allowing the inclusion of newly arriving and exclusion of leaving time-triggered M2M devices in the semi-persistent schedule and scheduling of event-triggered M2M and H2H every TTI. The period P may be determined according to the traffic density of time-triggered M2M devices and k_{max}. With low traffic density and low bandwidth usage, the schedule can be updated over longer intervals; i.e. P can be larger, since the resources are not so scarce, hence, do not require frequent optimization. On the other hand, if traffic density for time-triggered M2M devices is high and a large number of UFBs are allowed to be used, then the schedule can be updated over shorter intervals. In the construction of the semi-persistent schedule, the goal is to use minimum number of UFBs for the allocation while satisfying the period and delay requirements of the time-triggered M2M devices. If the regenerated schedule cannot allocate these devices within maximum number of available bands k_{max}, then the lowest priority devices are dropped such that the remaining devices can be allocated within k_{max} bands. In between the regeneration times of the semi-persistent schedule, the newly arriving time-triggered M2M devices are allocated using call admission control algorithm. Call admission control algorithm assumes the pre-allocation of previously assigned devices and aims to minimize the total number of UFBs used following the allocation of new devices. Call admission control algorithm guarantees the usage of at most k_{max} UFBs by not admitting the lowest priority devices if needed. Following the allocation of all time-triggered M2M devices, at any TTI, if there are still idle bands within k_{max}, then event-triggered M2M and H2H devices can be scheduled. In order to schedule these devices, the base station may use any previously proposed dynamic scheduling algorithm for cellular networks such as [43].

IV. DESCRIPTION OF THE OPTIMIZATION PROBLEM

In this section, we first provide the motivation for the objective of the minimization of the number of UFBs occupied by time-triggered M2M devices, then formulate

the QoS constraints of periodic M2M devices, give the formulation of the resulting optimization problem, and finally prove its NP-hardness.

A. Objective Function

The objective function of the minimization of the number of UFBs is motivated by the spectrum scarcity, separation of resources allocated to machines and humans, and presence of newly joining time-triggered machines, event-driven devices and human-to-human communication with various QoS requirements as follows:

- Massive number of M2M devices consume scarce radio resources that are already strained by H2H communications [44]. Sustaining acceptable QoS with these scarce resources requires efficient utilization of the available spectrum resources [45]. This can be achieved by minimizing number of UFBs allocated to M2M devices.
- Frequency bands reserved for M2M and H2H devices need to be separated for fair allocation of resources and exploitation of pre-determined traffic generation characteristics of time-triggered M2M devices. Many M2M applications have strict timing requirements; such as medical applications, assisted living, industrial control and navigational data communications [46], [47], [40]. If H2H and M2M bands are not separated and H2H communications are given priority, then these delay sensitive M2M applications suffer excessive delays and resource starvation due to H2H devices. Similarly, many H2H applications such as online gaming, internet browsing, video streaming, and VoIP have strict latency requirements [48], [23]. If M2M communications are given priority, these delay-sensitive H2H applications may experience performance degradation due to massive number of M2M devices allocated to the same bands. Moreover, the exploitation of the packet generation characteristics of time-triggered devices through semi-persistent scheduling requires a separation from the devices with random packet generation characteristics. The number of frequency bands allocated to time-triggered M2M devices must be minimized in order to serve more M2M devices and provide more resources for H2H communications.
- The generated semi-persistent schedule should allow the scheduling of new devices generating packets at random times within their strict delay constraints. Due to strict delay requirements, there must be sufficient radio resources immediately available for such devices. This requires an efficient usage of radio resources by time-triggered M2M devices so that some idle resources are available for such delay sensitive devices.

Previous semi-persistent scheduling algorithms developed for M2M communications fail to provide any such objective function, allocating the entire AGTI to a cluster, resulting in adverse effect on the delay performance of lower priority clusters of M2M devices and H2H communications [26], [27], [28]. On the other hand, the objective function of previous random access or dynamic scheduling based M2M communication studies include maximization of success rate [7], [8], [9], [10], [11], [12], minimization of delay [7], [8], [16], [15], [17], [18], [19], [30], [31], [32] and maximization of throughput [15]. These studies, however, fail to combine these objectives with QoS constraints due to random access procedure, or do not consider the delay and signaling overhead of the associated random access and periodic update of channel conditions in dynamic scheduling.

B. QoS Constraints

The QoS constraints of time-triggered M2M devices must ensure that delay tolerances are never violated, i.e. worst case delay is less than the corresponding delay tolerance. For any device, the worst case delay occurs when the device wants to transmit a packet at the same time with all higher priority devices. In that case, the device has to wait for all higher priority devices to transmit their data. The mathematical expression for the worst case delay serves as a computationally simple sufficient condition for satisfying delay tolerances.

Let N, δ_i^* and τ_i be the number of time-triggered devices on the same band, delay bound and transmission time of device i, respectively. Assume that devices are ordered according to their priorities; i.e. if device i is prior to device l, then $i < l$. The QoS constraint is formulated based on the extension of the delay bound formulation in [26] for variable transmission times as follows:

$$\delta_i^* = \tau_i + \sum_{l=1}^{i-1} \lceil \frac{p_i}{p_l} \rceil \tau_l \leq d_i, \tag{1}$$

for $\quad i \in [1, N]$.

C. Formulation of Optimization Problem

The optimization problem for minimizing the number of UFBs used by the time-triggered M2M devices while satisfying their period and delay tolerance constraints is formulated as follows:

minimize

$$\sum_{k=1}^{k_{max}} y_k \tag{2a}$$

subject to

$$\sum_{k=1}^{k_{max}} x_{ik} = 1, \quad i \in [1, N] \tag{2b}$$

$$\sum_{i=1}^{N} x_{ik} \leq N y_k, \quad k \in [1, k_{max}] \tag{2c}$$

$$\tau_i + \sum_{l=1}^{i-1} \lceil \frac{p_i}{p_l} \rceil \tau_l x_{lk} \leq d_i + (1 - x_{ik}) T_i, \quad i \in [1, N], k \in [1, k_{max}] \tag{2d}$$

variables

$$y_k \in \{0, 1\},\ x_{ik} \in \{0, 1\},\ i \in [1, N], k \in [1, k_{max}] \tag{2e}$$

where $T_i = \tau_i + \sum_{l=1}^{N} \lceil \frac{p_i}{p_l} \rceil \tau_l - d_i$ ensuring that the inequality in Eqn. (2d) is always satisfied when $x_{ik} = 0$. The variables of the problem are $y_k, k \in [1, k_{max}]$, binary variable taking value 1 if there exists a device allocated to UFB k, and 0 otherwise; $x_{ik}, i \in [1, N], k \in [1, k_{max}]$, binary variable taking value 1 if device i is allocated to UFB k, and 0 otherwise. The objective is to minimize the number of UFBs used. Eqn. (2b) states that each device is scheduled to one UFB. Equation (2c) represents that a UFB is used if there exists at least one device allocated to that band. Eqn. (2d) provides delay bound constraint. This optimization problem is an Integer Programming problem.

D. NP-Hardness of Optimization Problem

***Theorem** 1:* The optimization problem (2) is NP-Hard.

Proof: We reduce the NP-hard 3-partition problem to our scheduling problem. Consider a set of $3A$ positive integers, $S = \{a_1, a_2, .., a_{3A}\}$, where $\frac{B}{4} < a_i < \frac{B}{2}$ and $B \in \mathbb{Z}^+$ for all $i \in [1, 3A]$ and $\sum_{i=1}^{3A} a_i = AB$. The 3-partition problem aims at answering the question of whether the set S can be divided into A disjoint sets $S_1, .., S_A$ such that for each $m \in [1, A]$, $\sum_{a_i \in S_m} a_i = B$ is satisfied. Note that each disjoint set has exactly 3 elements since otherwise, $\frac{B}{4} < a_i < \frac{B}{2}, i \in [1, 3A]$ and $\sum_{a_i \in S_m} a_i = B$, $m \in [1, A]$ constraints yield a contradiction.

Let us define a problem instance where the delay tolerance and the period are equal to B, i.e. $d_i = p_i = B$, and transmission time $\tau_i = a_i$, where $\frac{B}{4} < a_i < \frac{B}{2}$ and $B \in \mathbb{Z}^+$ for all $i \in [1, 3A]$, $\sum_{i=1}^{3A} a_i = AB$, $k_{max} = A$. The necessary and sufficient condition for the schedulability of these devices is that the node set is divided into A disjoint sets $S_1, .., S_A$ such that the sum of their transmission times is not larger than the period; i.e. $\sum_{a_i \in S_m} a_i \leq B$ for $m \in [1, A]$. Using the contradiction argument of the 3-partition problem, exactly 3 devices need to be scheduled at each band. Since the sum of all transmission times is satisfied with equality, i.e. $\sum_{i=1}^{3A} a_i = AB$, the sum of the transmission times of the devices assigned to a specific band should also be equal to B, i.e. $\sum_{a_i \in S_m} a_i = B$. Then, this problem has a solution if and only if given instance of 3-Partition Problem has a solution. Since this construction is carried out in polynomial time, the problem of whether a set of devices S with integer transmission times is schedulable on k_{max} bands is NP-hard. Obviously, if the problem of finding minimum number of bands that can schedule the set S can be solved in polynomial time, then the problem whether the set S can be scheduled on k bands can be solved in polynomial time as well, for any $k \in \mathbb{Z}^+$. Thus, the problem of finding minimum number of bands to schedule set S is also NP-hard. □

V. Efficient Depth-First Search Algorithm

A straightforward search algorithm is based on the enumeration of all possible assignments of the M2M devices to the frequency bands such that each device is allocated to only one frequency band and the devices are allocated to RBs in the order of their priorities, and checking whether delay tolerances of the devices are satisfied. The optimal solution is then the minimum of the number of the frequency bands used by the feasible assignments. The complexity of this search is $O(Nk_{max}^N)$. The delay tolerance of each of the N nodes needs to be checked for k_{max}^N possible band allocations.

We will propose an efficient pruning based search algorithm based on the construction of a tree and development of pruning conditions to fathom the branches of the tree, by exploiting the problem structure to decrease the complexity of the brute-force search. In the search algorithm, we use a tree structure for the assignment of the M2M devices to the frequency bands with each node represented by $z_r = (i_1, i_2, ..., i_N)$, where the devices are enumerated in the order of decreasing priority. The root of the tree is $(0, 0, ..., 0)$, representing that none of the devices are allocated to any band yet. In the $j-th$ level of the tree, each of the nodes in the previous level is branched into k_{max} nodes, representing the allocation of the $j-th$ node to the corresponding frequency band. The leaves of the tree represent the assignment of the nodes to the frequency bands, without including any zero entry. The proposed algorithm is based on the construction of this tree from the root by using depth-first search (DFS) and pruning of the nodes during the construction without checking their descendants in the case the following conditions are met:

1) The allocation of the device on the particular band that the node represents violates the delay bound for the device.
2) The allocation of the device on the particular band that the node represents results in a worse solution than the best feasible solution already obtained.

For both conditions, note that descendant nodes do not change the allocation of devices represented by the parent node. Therefore, if the delay bound is violated for a device on a particular band, the violation cannot be reverted on descendant nodes. Similarly, if the allocation

of the device requires more bands than the best solution obtained so far, then the search through descendant nodes will not decrease the number of required bands. The DFS enables to quickly obtain a feasible solution that can be used as an upper bound for subsequent search, thereby eliminating solutions that are far from optimal.

Algorithm 1 Efficient Depth-First Search Algorithm (EDFS)

1: **initialize variables:** $bestFeasible$=N, F={all nodes}, CN= zeros(N,1), $optimalAllocation$=zeros(N,1);
2: **while** $F \neq \varnothing$ **do**
3: **if** $R_{CN} = \varnothing$ **and** CN is not at level N **then**
4: $F \leftarrow F \setminus CN$;
5: $CN \leftarrow$ Parent of CN;
6: **else if** CN satisfies any pruning condition **then**
7: $F \leftarrow F \setminus (R^*_{CN} \cup CN)$;
8: $CN \leftarrow$ Parent of CN;
9: **else if** CN is not at level N **then**
10: $CN \leftarrow$ Child in R_{CN} with minimum band value;
11: **else if** max $(CN) \leq bestFeasible$ **and** delay bound is not violated **then**
12: $bestFeasible$=max(CN);
13: $optimalAllocation = CN$;
14: $CN \leftarrow$ Parent of CN;
15: $F \leftarrow F \setminus (R_{CN} \cup CN)$;
16: $CN \leftarrow$ Parent of CN;
17: **end if**
18: **end while**

The Efficient Depth-First Search Algorithm (EDFS), given in Algorithm 1, is described next. The algorithm starts the search from the root node by initializing the current node, denoted by Current Node (CN), to a zero vector. The best feasible solution and the band allocation vector corresponding to optimal solution, denoted by $bestFeasible$ and $optimalAllocation$, are initialized to N and zero vector, respectively (Line 1). Note that there may be multiple $optimalAllocation$ vectors satisfying the optimal band number. Our algorithm stores only one of the $optimalAllocation$ vectors corresponding to optimal band number. F is the set of unexplored nodes and initialized to all nodes. R_{CN} refers to unexplored children of node CN. R^*_{CN} contains all unexplored descendants of node CN. If the current node does not have any unexplored children and is not at the level N; i.e. it is not one of the leaves of the search tree, then the current node is marked explored, returning to parent node (Lines 3-5). If the current node has children to be explored but satisfies at least one of the the pruning conditions, then the node and all its descendants are marked explored, returning to parent node (Line 6-8). If the current node has children to be explored and does not satisfy pruning conditions, then the algorithm proceeds with the unexplored child node with minimum band value (Lines 9-10). If the current node is at the level N and provides a better feasible solution than $bestFeasible$, $bestFeasible$ is updated while storing CN in the $optimalAllocation$ vector (Lines 11-13). Since from each parent node, we move to the unexplored child node with the minimum band value, all other unexplored nodes of the same parent will provide a worse solution. Thus, the current node is updated with the parent node, while marking the parent and all unexplored children explored, and moving the next parent node again (Lines 14-16). The algorithm terminates when all nodes are explored (Line 2). The output of the algorithm is $optimalAllocation$. If $optimalAllocation$ is a zero vector, then no feasible solution exists.

VI. MINIMUM FREQUENCY FIRST-FIT ALLOCATION ALGORITHM

Although the EDFS algorithm described in Section V decreases the complexity of the straightforward search algorithm with a smart pruning mechanism, the complexity of the algorithm is still exponential, which may not be manageable with the increasing number of H2H and M2M devices in LTE/5G stations [27]. The proposed polynomial time heuristic algorithm is closely related to the EDFS algorithm with two features decreasing the runtime complexity. First, instead of exploring all the nodes in the tree, the heuristic algorithm explores only one branch of the tree starting from the root node and moving to the feasible child node with the minimum band number at each step. Second, the devices with the same traffic and QoS characteristics are grouped into clusters with the goal of assigning the nodes in bulks instead of one-by-one.

A. Algorithm Description

Algorithm 2 Minimum Frequency First-Fit Allocation (MFFFA) Algorithm

Input: d_i^c, p_i^c, τ_i^c, N_i for $i \in [1, M]$
Output: K, UFB^k, for $k \in [1, K]$
$k = 1$;
while $\sum_{i\in[1,M]} N_i \neq 0$ **do**
$\mathbf{UFB}^k = zeros(1, M)$;
for $i = 1 : M$ **do**
$crossDelay = 0$;
for $j = 1 : i - 1$ **do**
$crossDelay = crossDelay + \mathbf{UFB}^k_j \lceil \frac{p_i^c}{p_j^c} \rceil \tau_j^c$;
end for
$remDelay = d_i^c - crossDelay$;
if $remDelay > 0$ **then**
$\mathbf{UFB}^k_i = \min(N_i, \lfloor \frac{remDelay}{\tau_i^c} \rfloor)$;
end if
$N_i = N_i - \mathbf{UFB}^k_i$;
end for
$k++$;
end while
$K = k$;

We propose the Minimum Frequency First-Fit Allocation (MFFFA) Algorithm, given in Algorithm 2, as described next. MTC devices are clustered into M clusters based on packet arrival period, maximum allowable delay and transmission time, denoted by p_i^c, d_i^c and τ_i^c for cluster i, respectively. N_i denotes the number of unallocated devices from cluster i. $crossDelay$ is the extra delay the device experiences due to devices in higher priority clusters. $\mathbf{UFB}^k$ is an $M-$dimensional vector, with the $i-th$ entry, denoted by $\mathbf{UFB}_i^k$, storing the number of devices from cluster i in band k. $\mathbf{UFB}^k$ is initialized to a zero vector (Line 5). The algorithm allocates the devices starting from the highest-priority cluster (Line 6). First, $crossDelay$ on each cluster i from higher priority clusters is calculated (Lines $7-10$). Then, the difference between the delay tolerance and experienced delay due to higher priority clusters, denoted by $remDelay$, is computed to determine the number of devices from cluster i that can be allocated in band k. If the calculated number is larger than the number of remaining devices from cluster i, then all remaining devices in cluster i are allocated to that particular band. Otherwise, the number of remaining devices from this cluster will be updated for the allocation to the following bands (Lines $11-15$). The algorithm stops when there are no remaining unallocated devices from the clusters (Line 4).

B. Algorithm Illustration Through An Example

In Fig. 3, we describe the workings of our algorithm through an example. Let the number of clusters be 3. The cluster parameters are given by $d_1^c = 2, p_1^c = 2, \tau_1^c = 1, N_1 = 3, d_2^c = 3, p_2^c = 3, \tau_2^c = 1, N_2 = 2, d_3^c = 3, p_3^c = 6, \tau_3^c = 1, N_3 = 4$. The first, second and third clusters are denoted by A, B and C, respectively. We start allocating devices from cluster A. Since there is no higher priority cluster, there is no delay imposed by other clusters on A, setting $crossDelay$ to 0. The $remDelay$ is $d_1 - crossDelay = 2 - 0 = 2$. We can allocate $remDelay/\tau_1 = 2/1 = 2$ devices from cluster A on the first UFB, updating $\mathbf{UFB}_1^1 = 2$. Now, we try cluster B on the first UFB. The delay imposed on cluster B from cluster A on the first band is, $\mathbf{UFB}_1^1 * \lceil \frac{p_2}{p_1} \rceil * \tau_1 = 2 * \lceil 3/2 \rceil * 1 = 4$. However, $4 > d_2 = 3$, i.e. the delay imposed on cluster B is larger than its tolerance. Thus, we cannot allocate any device from cluster B on the first UFB. Similarly, the delay imposed by cluster A on cluster C is $2 * \lceil 6/2 \rceil * 1 = 6$ and $6 > d_3 = 3$. Thus, no device from cluster C can be allocated on the first band. We move to the second UFB. We still have the third device from cluster A. We allocate this device on the second UFB. Now, we proceed to cluster B, the second highest priority cluster. The cross delay is $\lceil 3/2 \rceil * 1 = 2$. The remaining delay

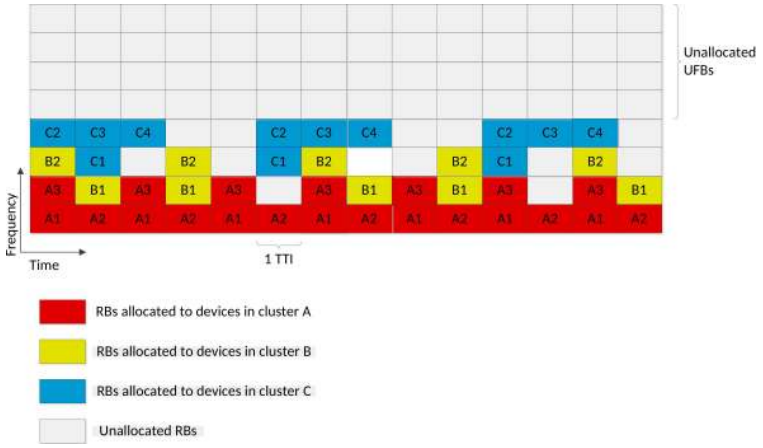


Fig. 3: Algorithm Description: An example

is $d_2 - crossdelay = 3 - 2 = 1$. We can allocate $remdelay/\tau_2 = 1/1 = 1$ device from cluster B on the second band. For cluster C, the devices on the second UFB would impose $\lceil 6/2 \rceil + \lceil 6/3 \rceil = 5 > d_3$ delay, thus cluster C cannot be allocated on the second UFB. With this procedure, we allocate the remaining devices on the third and the fourth UFB.

C. Worst Case Performance Analysis

In this section, we find the approximation bound of the MFFFA algorithm under certain conditions. For this purpose, we will need *Lemma 1* and *Lemma 2*.

Consider a set of M2M devices $S = \{a_1, .., a_N\}$, where each device a_j has implicit deadline, period p_j and transmission time τ_j RB, with τ_j being an integer, for $j \in [1, N]$. For each device a_j, define a set $S'_j = \{b_{j_1}, .., b_{j_{\tau_j}}\}$, where each device $b_{j_i}, i \in [1, \tau_j]$, has 1 RB transmission time and period p_j. Define S' as $\cup_{j=1}^{N} S'_j$.

***Lemma** 1:* Set S is schedulable on a single band if and only if set S' is schedulable on a single band.

Proof: Assume set S is schedulable. Then, τ_j RBs are allocated within each period p_j of device a_j. Allocate each $b_{j_i}, i \in [1, \tau_j]$ at the time instant when the $i-th$ RB of device a_j is allocated. Since this holds for an arbitrary j, the set S' is also schedulable.

Assume set S' is schedulable. Then, all devices in subset S'_j are scheduled within time interval $(t, t + p_j)$. There are τ_j devices with 1 RB transmission time in subset S_j. Allocate the $i-th$ RB of device a_j at the time instant when device b_{j_i} is allocated, for $i = 1, .., \tau_j$. With this allocation scheme, all τ_j RB's of device a_j are allocated in time interval $(t, t + p_j)$. Since this holds for an arbitrary j, the set S is schedulable. □

***Lemma** 2:* Let $\tau_j = 1$ for $j \in [1, N]$. For any scheduling policy, the preemptive allocation yields the same schedule as the non-preemptive allocation.

Proof: Assume that a packet from the device a_i starts transmission at time $t = t'$ RB, where $t' \in \mathbb{Z}^+$. This

implies that all higher priority devices that generated a packet up to time t' RB have completed their transmission. Since the devices generate data at integer multiples of 1 RB, there can be no preemption in $(t', t'+1)$ RB interval. Since transmission times are equal to 1 RB, the packet from device a_i completes its transmission at time $t = t' + 1$ RB without any preemption. □

1) Worst Case Performance With Implicit Deadlines: We first find the approximation bound of the MFFFA algorithm under the following conditions: Devices generate data at integer multiples of 1 RB and devices have implicit deadlines; i.e. their delay tolerances are equal to their periods. In order to find the approximation bound, we will use the approximation bound provided for Rate-Monotonic-First-Fit (RM-FF) algorithm in [49]. RM-FF algorithm greedily allocates devices onto bands starting from the highest priority device. However, RM-FF allows preemption whereas MFFFA does not. Also, MFFFA orders devices in increasing period, lower period implying higher priority, whereas RM-FF allocates devices with respect to any given priority order.

***Theorem** 2:* Let K_0 denote the minimum number of UFB bands required to schedule the set S and K be the minimum number of UFB bands required to schedule the set S by MFFFA algorithm. Then, the following relation holds:

$$K \leq [2 + \frac{(3 - 2^{\frac{3}{2}})}{2^{\frac{4}{3}} - 2}]K_0 + 1 \approx 2.33K_0 + 1 \qquad (3)$$

Proof: The schedulability of the set S is equivalent to that of $S^{'}$ containing devices of 1 RB transmission time due to *Lemma 1*. Moreover, the preemptive schedule of the set S' is equivalent to its non-preemptive schedule due to *Lemma 2*. Thus, MFFFA is an instance of the RM-FF algorithm, where the given priority order coincides with the order of increasing period. Therefore, the performance bound of the RM-FF algorithm in Eqn. (3) given in [49] can be used for MFFFA. □

2) Worst Case Performance With Simply Periodic Set of Devices: Next, we find the approximation bound of the MFFFA algorithm under the following conditions: The transmission time of devices is 1 RB and devices are simply periodic; i.e. for any two devices a_i and a_j with periods $p_i < p_j$, p_j is an integer multiple of p_i. Devices have implicit deadlines and generate data at integer multiples of 1 RB.

***Theorem** 3:* Let K_0 denote the minimum number of UFB bands required to schedule the set S and K be the minimum number of UFB bands required to schedule the set S by MFFFA algorithm. Then, the following relation holds:

$$K \leq 1.5K_0 \qquad (4)$$

Proof: The proof is based on the demonstration of the equivalence between our scheduling problem and bin-packing problem, and using the bound derived for the First-Fit Decreasing (FFD) algorithm proposed for bin-packing problem. First, we show that our scheduling problem is equivalent to bin-packing problem. Bin packing problem can be described as follows: We have objects with different sizes and bins with identical capacities. We need to pack these objects into bins such that the number of bins used is minimized. Let us consider a bin-packing problem. The size of an object corresponds to the utilization of a device, i.e. ratio of the transmission time of a device to its period. Each UFB is associated with a bin. Accordingly, the sum of the utilizations of the devices assigned to each UFB gives the total size of the objects in a bin. To complete the equivalence, we will use the following result in [50]: For a preemptive, simply periodic system with implicit deadlines, the rate monotonic (RM) algorithm, an algorithm that allocates devices based on the priority order in which lower period implies higher priority, is schedulable on a uniform processor if and only if the total utilization of devices is equal to or less than 1. Therefore, if we use RM algorithm at each UFB, total utilization of devices allocated to a UFB cannot exceed 1, resulting in bin capacity of 1. The objective of minimizing the number of UFBs is then equivalent to the objective of minimizing the number of bins in bin-packing problem.

Based on this equivalence, we will use the bound of the FFD algorithm proposed for bin-packing problem in [51]. FFD first sorts items in non-increasing order of their sizes and places them in the lowest indexed bin as they appear, i.e. First-Fit principle. On the other hand, MFFFA first orders devices in increasing period, i.e. non-decreasing utilization, and allocates them with nonpreemptive RM algorithm on individual UFBs based on First-Fit principle. Further note that since we use devices generating data at integer multiples of RBs with 1 RB transmission time, preemptive RM schedule is equivalent to nonpreemptive RM schedule due to *Lemma 2*. Thus, MFFFA is equivalent to First Fit Decreasing algorithm proposed for bin packing problem, for which the approximation bound is proven to be 1.5. □

VII. CALL ADMISSION CONTROL

Call admission control algorithm aims to manage the admission and resource allocation of newly joining time-triggered M2M devices between the regeneration times of the semi-persistent schedule. Given the allocation of the existing time-triggered M2M devices, the newly arriving time-triggered M2M devices are scheduled with the goal of using minimum number of UFBs while providing their QoS guarantees. The usage of at most k_{max} UFBs is guaranteed by not admitting lowest priority devices as needed.

A. Call Admission Optimization Problem

Call admission optimization problem aims to minimize the number of bands used throughout the semi-persistent period P while satisfying the QoS requirements of both existing and newly arriving devices. The devices arriving earlier are given higher priority since they are scheduled before those arriving later in time. If two devices arrive at the same TTI, then the priority ordering is the same as the one described in Section II. Once the devices are allocated resources within the schedule, the resource allocation is not updated until the next regeneration time of semi-persistent schedule. The optimal allocation at each TTI requires the knowledge of the requirements of the devices arriving both before and after that TTI within two regeneration times of the semi-persistent schedule, since the objective is to minimize the number of UFBs and allocation cannot be changed until the next schedule regeneration. Thus, the optimization problem is offline. Although the resource allocation algorithms must be online since we do not assume the availability of any information on newly arriving devices, this optimization problem is useful in providing a theoretical lower bound on the UFB usage.

The call admission optimization problem is exactly the same as the optimization problem described in Section IV, with the exception of a modified priority ordering. This problem is NP-hard. In order to prove this, take a specific instance of the problem where the number of TTIs between two regeneration times of the schedule is equal to 1. In that specific instance, the call admission optimization problem is equivalent to the optimization problem provided in Section IV, which is proved to be NP-hard in Section IV-C. Therefore, solving the call admission optimization problem requires an exponential runtime in the number of newly arriving devices. The optimization problem further requires an offline algorithm, which is not possible to implement in practice. Thus, we propose an online fast and efficient call admission heuristic algorithm for the solution of this problem.

B. Call Admission Algorithm

The proposed heuristic call admission algorithm is shown in Fig. 4. First Fit Occupied Bands (FFOB) Algorithm is first executed to allocate newly arriving devices to the UFBs some parts of which are already occupied by existing devices. If there are still unallocated devices after running the FFOB algorithm, the base station allocates remaining devices to new bands by using MFFFA algorithm. If call admission mechanism requires more bands than k_{max}, then the lowest priority device is dropped. This continues until the allocation requires at most k_{max} UFB bands.

FFOB algorithm is similar to MFFFA except its execution on the UFBs some parts of which are already assigned to the previously arrived devices. Similar to

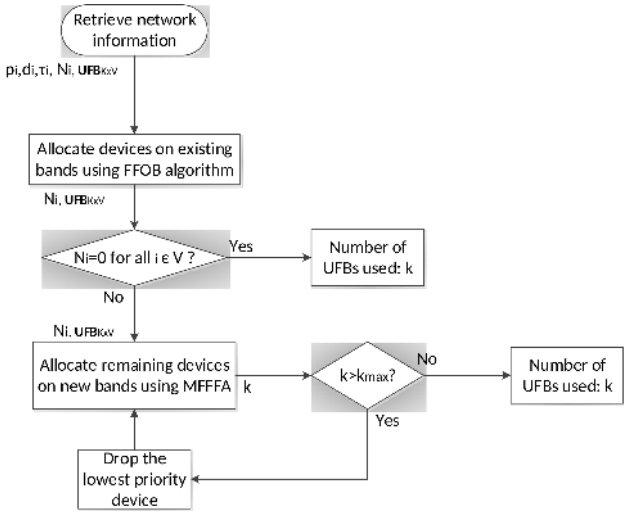


Fig. 4: Call admission control mechanism

MFFFA, devices are ordered according to their priorities, and then are allocated to the lowest number UFB on which they can be allocated. Unlike MFFFA that aims to allocate all devices, FFOB leaves the allocation of the devices that cannot be allocated within already occupied bands to the MFFFA algorithm for their allocation on new bands as shown in Fig. 3. Let the delay tolerance, period, transmission time and number of newly arriving devices, denoted by d_i^c, p_i^c, τ_i^c, N_i, respectively, for $i \in [1, M]$, be given. Let $\mathbf{UFB}^k$ be an $M-$dimensional vector, with the $i-th$ entry, denoted by $\mathbf{UFB}_i^k$, storing the number of devices from cluster i in band k. The input of the FFOB algorithm is $\mathbf{UFB}_i^k$ containing the number of devices from cluster $i \in M$ that are already allocated on band k. The output of the algorithm is the number of unallocated devices after running the algorithm, N_i, and updated $\mathbf{UFB}_i^k$. Starting from the highest priority cluster, cross delay, the delay experienced due to devices already allocated to band k, is calculated (Lines 5-8). Note that unlike MFFFA, devices experience an additional delay from all devices that are already assigned. Then, the remaining delay is computed to determine the number of devices that can be additionally allocated from the corresponding cluster with exactly the same procedure applied in MFFFA (Lines 11-12).

VIII. PERFORMANCE EVALUATION

The goal of the simulations is to compare the performance of the proposed scheduling and call admission algorithms to that of previously proposed algorithms, including clustering-based scheduling algorithm [26], basic LTE without ACB, and LTE system with dynamic optimal ACB [6], and optimal solution in terms of schedulability and usage of frequency band. In clustering-based scheduling algorithm, denoted by MAM, devices are grouped into clusters according to their periods and delay tolerances. Clusters are assigned

Algorithm 3 **First Fit Occupied Bands Algorithm**

Input: d_i^c, p_i^c, τ_i^c, K, N_i for $i \in [1, M]$, UFB^k, for $k \in [1, K]$
Output: N_i for $i \in [1, M]$, UFB^k, for $k \in [1, K]$
for $i = 1 : M$ **do**
  for $k = 1 : K$ **do**
    $crossDelay = 0$;
    for $j = 1 : M$ **do**
      $crossDelay = crossDelay + \mathbf{UFB}_j^k * \lceil \frac{p_i^c}{p_j^c} \rceil * \tau_j^c$;
    end for
    $remDelay = d_i^c - crossDelay$;
    if $remDelay > 0$ **then**
      $\mathbf{UFB}_i^k = \mathbf{UFB}_i^k + \min(N_i, \lfloor \frac{remDelay}{\tau_i} \rfloor)$;
      $N_i = N_i - \min(N_i, \lfloor \frac{remDelay}{\tau_i} \rfloor)$;
    end if
  end for
end for

priorities according to their periods such that clusters with lower periods have higher priorities. The transmission times of all devices are assumed to be equal to 1 RB. The clusters are assigned to TTIs such that devices in the same cluster are allocated resources at the same TTI using different UFBs and no device from other clusters can be assigned to this TTI. Since MAM assumes the availability of unlimited number of frequency bands, we additionally include a mechanism for the case in which the allocation of a cluster exceeds the frequency band limit: If the allocation of a cluster requires more UFBs than the maximum number of UFBs in a TTI, the cluster is subdivided into new clusters, each new cluster fitting within the maximum number of frequency bands. The call admission procedure proposed for MAM allocates the newly coming device to the existing TTIs if the device belongs to an existing cluster and there are available resource blocks in the corresponding TTI, and creates a new cluster for the device otherwise. The basic LTE refers to random access method that does not use any access class barring method, and is denoted by basic LTE. In the event of collision at the random access, colliding devices wait for a backoff time. Packet losses occur under the following conditions: If the number of retransmissions for a packet exceeds the maximum number of retransmissions or if a device produces the next packet before successfully transmitting a packet. For the LTE system with dynamic optimal ACB, denoted by DACB, the base stations dynamically set the ACB parameter to its optimal value obtained in [6]. With respect to physical layer, in order to get the best possible result out of random access based solutions we will make the following assumptions as in [52]: the channel is ideal, there is no signal loss due to radio propagation problems, and once a preamble is successfully transmitted, the device accesses the channel. LTE parameters are adopted from [53]: backoff time = 20 ms, maximum number of retransmissions = 10, number of preambles = 54, random access opportunity period = 5 ms. Finally, the optimal solution is denoted by OPT.

Simulation results are obtained based on 1000 random network topologies, in which devices are uniformly distributed within a circle of radius r with the base station at the center. Simulation of the medium access protocols is performed in an event-based simulator developed in MATLAB, called M2MSCHEDULE, and simulation of the optimal algorithm is performed in GAMS, a high-level modeling platform for expressing and solving linear, nonlinear and mixed integer optimization problems. Both simulators are publicly available in [54]. We assume that the transmission time of all M2M devices is equal to 1 RB for comparison to the MAM algorithm. Each TTI comprises of 100 RBs. The packet generation period of the clusters is chosen from $[10, 2000]$ ms, based on practically used values in [26], and delay tolerance of clusters is uniformly chosen in the range from 10 ms to the randomly chosen packet generation period unless otherwise stated and denoted by uniform deadline case.

A. Schedulability Performance

Fig. 5 shows the schedulability percentage of different access mechanisms for different number of M2M devices. The schedulability percentage is defined as the percentage of the total number of packets that are successfully sent (i.e. packets that are not lost) out of the total number of packets generated. The number of clusters is fixed to 12. The schedulability performance of the MFFFA algorithm outperforms both variations of random access mechanisms and previous schedule based algorithm due to the proposed efficient resource allocation mechanism. MFFFA succeeds in allocating all devices within the frequency band limit with a slight drop below 100% for around 5000 devices, where the frequency band limit is reached. The schedulability of the MAM algorithm deteriorates drastically with increasing number of devices. Available frequency bands fail to suffice for increasing number of devices, causing clusters that reach the frequency band limit to be subdivided into new clusters. This increase in the number of clusters results in increased delay and decreased schedulability percentage. On the other hand, the schedulability performance of random access based methods deteriorates as the number of devices increases with the increasing number of collisions in the network. As expected, the deterioration in performance is smaller in DACB than basic LTE due the dynamic optimization of ACB parameter.

Fig. 6 shows the schedulability percentage of different scheduling algorithms for different number of M2M clusters. The number of devices is fixed to 2000. As random access based methods do not depend on clustering, they are not included. The smart resource allocation mechanism of the MFFFA algorithm achieves much

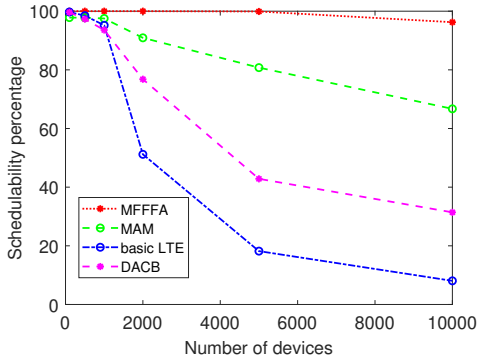


Fig. 5: Schedulability percentage of MFFFA, MAM, basic LTE and DACB algorithms for different number of M2M devices, where $k_{max} = 100$.

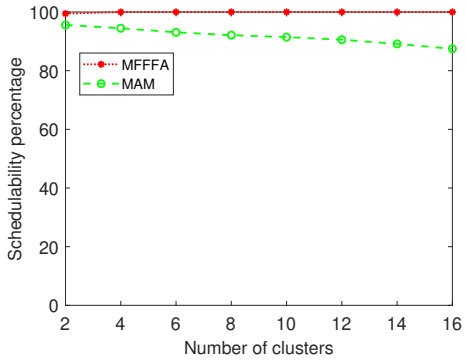


Fig. 6: Schedulability percentage of MFFFA and MAM algorithms for different number of M2M clusters, where $k_{max} = 100$.

better schedulability performance than MAM algorithm for different number of clusters. MAM algorithm fails to allocate all devices within the frequency band limit for any given cluster number. Each cluster occupies a single TTI in MAM, resulting in increasing delay and decreasing schedulability percentage with increasing number of clusters.

B. Frequency Band Usage Performance

Fig. 7a and Fig. 7b show the number of frequency bands used by different scheduling algorithms for different number of clusters for the uniform deadline and implicit deadline cases, respectively. The number of devices belonging to each cluster is uniformly chosen from $[10, 50]$. For both cases, the average number of frequency bands used by MAM approaches to its maximum very quickly as the number of devices increases. The number of frequency bands used by MAM depends only on the number of devices in the cluster since each device in the cluster uses one frequency band. As the number of clusters increases, the probability that a cluster has a higher number of devices also increases. MFFFA, on the other hand, significantly outperforms MAM for different number of clusters. Furthermore, MFFFA performs very close to the OPT. Since the implicit deadline case is less restrictive than the uniform deadline case, MFFFA and OPT are able to allocate the same number of devices using fewer bands whereas the performance of MAM remains the same.

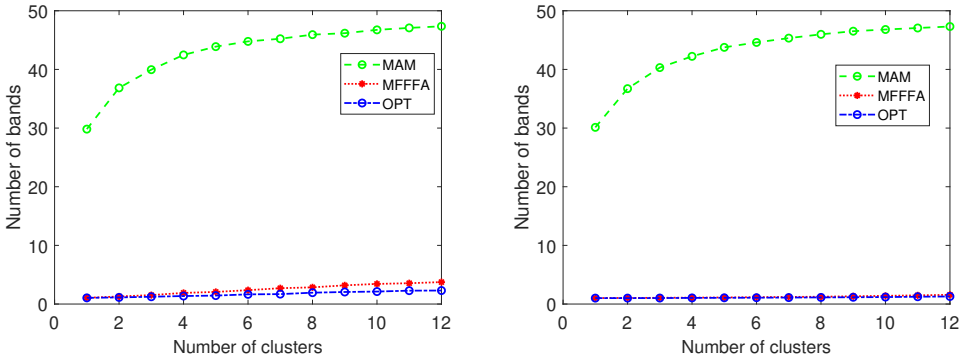
Fig. 8a and Fig. 8b show the number of frequency bands used by different scheduling algorithms for different number of devices for the uniform deadline and implicit deadline cases, respectively. The number of clusters is 12. Different number of devices are uniformly distributed among clusters. As stated before, the number of frequency bands used by the MAM depends only on the maximum number of devices within the clusters. Thus, for both cases, as the number of devices increases, the number of frequency bands used by the MAM increases. MFFFA, on the other hand, again performs much better than MAM. The effect of implicit deadlines is similar to Fig. 7.

C. Performance Evaluation of Proposed Call Admission Control Mechanism

Fig. 9 shows the number of frequency bands used by various call admission control mechanisms, i.e., the proposed call admission control mechanism (FFOB), call admission mechanism proposed for MAM [26] and offline optimal solution. The algorithm starts with 12 clusters and 50 devices randomly distributed among these clusters. The simulation duration is 20 s. The arrival of devices for each cluster at any TTI is modeled as a Poisson process with mean $\lambda = 0.001$. Semi-persistent schedule update period is set to 10 s. The proposed call admission mechanism is very efficient in the frequency band usage. The number of frequency bands occupied in MAM increases much more rapidly than that of the proposed algorithm FFOB as new devices arrive to the network. Since MAM algorithm does not use the scarce frequency resources efficiently, high traffic load leads to resource starvation. Further note that call admission algorithm needs to be restarted periodically so that the base station can allocate devices in an efficient manner. In other words, at the schedule update points only, we drop the condition that early arriving devices have the priority for FFOB. Thus, all the devices that are present at those points are treated with respect to their periods. This explains why the optimal solution, which also updates its schedule at schedule update points, improves at the schedule update point (at 10 s).

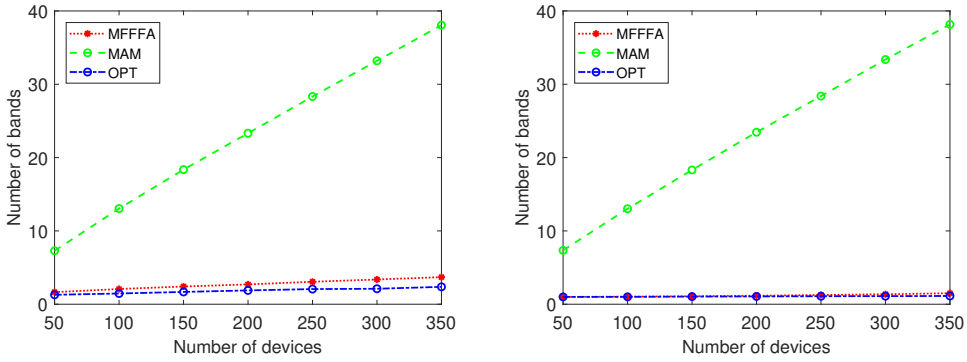
IX. CONCLUSION

We study the problem of M2M packet scheduling with QoS guarantees in cellular networks, particularly considering the radio resource starvation problem in 5G technology with dramatically growing numbers of M2M applications and devices. Accordingly, we propose a novel optimization framework with the objective of



(a) Uniform deadline case (b) Implicit deadline case

Fig. 7: Number of frequency bands used by MFFFA, MAM and OPT for different number of clusters.



(a) Uniform deadline case (b) Implicit deadline case

Fig. 8: Number of frequency bands used by MFFFA, MAM and OPT for different number of devices.

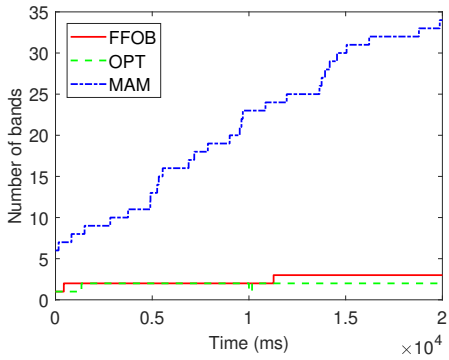


Fig. 9: Number of frequency bands used by FFOB, MAM and OPT in call admission control as time progresses.

minimizing the number of frequency bands occupied by a given set of M2M devices, while guaranteeing their delay and periodicity constraints. The optimization problem is proven to be NP-hard upon which we provide an efficient heuristic scheduling algorithm. We establish performance guarantees for the proposed heuristic algorithm by constructing an approximation bound to the optimal. We further provide a call admission control scheme to dynamically manage the arrivals of new devices to the network. Extensive simulations demonstrate that the proposed algorithm performs much better than the existing algorithms with very close performance to that of the optimal solution in minimizing the frequency band usage and maximizing schedulability. Furthermore, the proposed call admission control mechanism demonstrates the robustness of the proposed framework in a dynamic environment with changing traffic load and characteristics.

REFERENCES

[1] G. Wu, S. Talwar, K. Johnsson, N. Himayat, and K. D. Johnson, "M2m: From mobile to embedded internet," *IEEE Communications Magazine*, vol. 49, no. 4, pp. 36–43, April 2011.

[2] J. C. L. Zhou, D. Wu and Z. Dong, "When computation hugs intelligence: Content-aware data processing for industrial iot," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 1657–1666, June 2018.

[3] R. Pepper, "The rise of m2m devices," October 2015, 3rd BEREC Stakeholder Forum.

[4] A. G. Gotsis, A. S. Lioumpas, and A. Alexiou, "M2m scheduling over lte: Challenges and new perspectives," *IEEE Vehicular Technology Magazine*, vol. 7, no. 3, pp. 34–39, Sept 2012.

[5] L. Vangelista, A. Zanella, and M. Zorzi, "Long-range iot technologies: The dawn of lora," in *Future Access Enablers for*

Ubiquitous and Intelligent Infrastructures. Springer, 2015, pp. 51–58.

[6] S. Duan, V. Shah-Mansouri, Z. Wang, and V. Wong, "D-acb: Adaptive congestion control algorithm for bursty m2m traffic in lte networks," *IEEE Transactions on Vehicular Technology*, vol. PP, no. 99, pp. 1–1, 2016.

[7] G. T. . V11.0.0, "Study on ran improvements for machine-type communications," Sep. 2011.

[8] G. T. R. W. . R2-104662, "Mtc simulation results with specific solutions," Aug. 2010.

[9] J.-P. Cheng, C. H. Lee, and T.-M. Lin, "Prioritized random access with dynamic access barring for ran overload in 3gpp lte-a networks," in *IEEE GLOBECOM Workshops (GC Wkshps)*, Dec. 2011, pp. 368–372.

[10] S.-Y. Lien, T.-H. Liau, C.-Y. Kao, and K.-C. Chen, "Cooperative access class barring for machine-to-machine communications," *IEEE Transactions on Wireless Communications*, vol. 11, no. 1, pp. 27–32, Jan. 2012.

[11] X. Yang, A. Fapojuwo, and E. Egbogah, "Performance analysis and parameter optimization of random access backoff algorithm in lte," in *IEEE Vehicular Technology Conference (VTC)*, Sep. 2012, pp. 1–5.

[12] M. S. Ali, E. Hossain, and D. I. Kim, "Lte/lte-a random access for massive machine-type communications in smart cities," *IEEE Communications Magazine*, vol. 55, no. 1, pp. 76–83, Jan. 2017.

[13] S. Cherkaoui, I. Keskes, H. Rivano, and R. Stanica, "Lte-a random access channel capacity evaluation for m2m communications," in *Wireless Days (WD)*, Mar. 2016.

[14] Y. Chen and W. Wang, "Machine-to-machine communication in lte-a," in *IEEE Vehicular Technology Conference (VTC)*, Sep. 2010, pp. 1–4.

[15] D. T. Wiriaatmadja and K. W. Choi, "Hybrid random access and data transmission protocol for machine-to-machine communications in cellular networks," *IEEE Transactions on Wireless Communications*, vol. 14, no. 1, pp. 33–46, Jan 2015.

[16] C. Y. Oh, D. Hwang, and T. J. Lee, "Joint access control and resource allocation for concurrent and massive access of m2m devices," *IEEE Transactions on Wireless Communications*, vol. 14, no. 8, pp. 4182–4192, Aug 2015.

[17] A. Lo, Y. Law, and M. Jacobsson, "A cellular-centric service architecture for machine-to-machine (m2m) communications," *IEEE Wireless Communications*, vol. 20, no. 5, pp. 143–151, Oct. 2013.

[18] A. S. Lioumpas and A. Alexiou, "Uplink scheduling for machine-to-machine communications in lte-based cellular systems," in *IEEE GLOBECOM*, Dec. 2011, pp. 353–357.

[19] A. G. Gotsis, A. S. Lioumpas, and A. Alexiou, "Analytical modelling and performance evaluation of realistic time-controlled m2m scheduling over lte cellular networks," *Transactions on Emerging Telecommunications Technologies*, vol. 24, no. 4, pp. 378–388, June 2013.

[20] I. M. D.-L. et al., "Evaluation of latency-aware scheduling techniques for m2m traffic over lte," in *European Signal Processing Conference (EUSIPCO)*, Aug. 2012, pp. 989–993.

[21] N. Afrin, J. Brown, and J. Y. Khan, "Design of a buffer and channel adaptive lte semi-persistent scheduler for m2m communications," in *2015 IEEE International Conference on Communications (ICC)*, June 2015, pp. 5821–5826.

[22] J. B. Seo and V. C. M. Leung, "Performance modeling and stability of semi-persistent scheduling with initial random access in lte," *IEEE Transactions on Wireless Communications*, vol. 11, no. 12, pp. 4446–4456, December 2012.

[23] D. Jiang, H. Wang, E. Malkamaki, and E. Tuomaala, "Principle and performance of semi-persistent scheduling for voip in lte system," in *International Conference on Wireless Communications, Networking and Mobile Computing*, Sept 2007, pp. 2861–2864.

[24] "Persistent Scheduling in E-UTRA," Sorrento,Italy, Tech. Rep., January 2007, 3GPP TSG RAN WG1 Meeting 47bis, R1-070098.

[25] M. Rinne, M. Kuusela, E. Tuomaala, P. Kinnunen, I. Kovacs, and K. Pajukoski, "A performance summary of the evolved 3g (e-utra) for voice over internet and best effort traffic," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 7, pp. 3661–3673, Sep. 2009.

[26] S. Y. Lien and K. C. Chen, "Massive access management for qos guarantees in 3gpp machine-to-machine communications," *IEEE Communications Letters*, vol. 15, no. 3, pp. 311–313, March 2011.

[27] S. Y. Lien, K. C. Chen, and Y. Lin, "Toward ubiquitous massive accesses in 3gpp machine-to-machine communications," *IEEE Communications Magazine*, vol. 49, no. 4, pp. 66–74, April 2011.

[28] A. G. Gotsis, A. S. Lioumpas, and A. Alexiou, "Evolution of packet scheduling for machine-type communications over lte: Algorithmic design and performance analysis," in *2012 IEEE Globecom Workshops*, Dec 2012, pp. 1620–1625.

[29] S. Y. Shin and D. Triwicaksono, "Radio resource control scheme for machine-to-machine communication in lte infrastructure," in *International Conference on ICT Convergence (ICTC)*, Oct. 2012, pp. 1–6.

[30] J. Ding, A. Roy, and N. Saxena, "Smart m2m uplink scheduling algorithm over lte," *Elektronika IR Elektrotechnika*, vol. 19, no. 10, pp. 138–144, 2013.

[31] M. K. Giluka, N. Rajoria, A. C. Kulkarni, V. Sathya, and B. R. Tamma, "Class based dynamic priority scheduling for uplink to support m2m communications in lte," in *IEEE World Forum Internet Things (WF-IoT)*, Mar. 2014, pp. 313–317.

[32] K. Zheng, F. Hu, W. Wang, W. Xiang, and M. Dohler, "Radio resource allocation in lte-advanced cellular networks with m2m communications," *IEEE Communications Magazine*, vol. 50, no. 7, pp. 184–192, Jul. 2012.

[33] M. T. Islam, A. e. M. Taha, and S. Akl, "A survey of access management techniques in machine type communications," *IEEE Communications Magazine*, vol. 52, no. 4, pp. 74–81, April 2014.

[34] Z. Fan, R. J. Haines, and P. Kulkarni, "M2m communications for e-health and smart grid: an industry and standard perspective," *IEEE Wireless Communications*, vol. 21, no. 1, pp. 62–69, February 2014.

[35] S. C. Ergen and A. Sangiovanni-Vincentelli, "Intra-vehicular energy harvesting wireless networks," *IEEE Vehicular Technology Magazine*, vol. 12, no. 4, pp. 77–85, Dec. 2017.

[36] Y. Sadi and S. C. Ergen, "Optimal power control, rate adaptation and scheduling for uwb-based intra-vehicular wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 1, pp. 219–234, Jan. 2013.

[37] E. Dahlman, S. Parkvall, and J. Skold, *4G: LTE/LTE-advanced for mobile broadband*. Academic press, 2013.

[38] G. Wunder, P. Jung, M. Kasparick, T. Wild, F. Schaich, Y. Chen, S. T. Brink, I. Gaspar, N. Michailow, A. Festag, L. Mendes, N. Cassiau, D. Ktenas, M. Dryjanski, S. Pietrzyk, B. Eged, P. Vago, and F. Wiedmann, "5gnow: non-orthogonal, asynchronous waveforms for future mobile applications," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 97–105, February 2014.

[39] M. N. Shehzad, A.-M. Déplanche, Y. Trinquet, and R. Urunuela, "Overhead control in real-time global scheduling." in *RTNS*, 2011, pp. 45–52.

[40] L. Doyle and J. Elzey, "Successful use of rate monotonic theory on a formidable real time system," in *Real-Time Operating Systems and Software, 1994. RTOSS '94, Proceedings., 11th IEEE Workshop on*, May 1994, pp. 74–78.

[41] K.-C. Chen, "Machine-to-machine communications for healthcare," *Journal of Computing Science and Engineering*, vol. 6, no. 2, pp. 119–126, 2012.

[42] K. K. Chintalapudi and L. Venkatraman, "On the design of mac protocols for low-latency hard real-time discrete control applications over 802.15.4 hardware," in *Information Processing in Sensor Networks, 2008. IPSN '08. International Conference on*, April 2008, pp. 356–367.

[43] A. Jain and I. H. Hou, "R-pf: Enhancing service regularity for legacy scheduling policy," *IEEE Transactions on Wireless Communications*, vol. 15, no. 1, pp. 258–266, Jan 2016.

[44] V. Misic and J. Misic, *Machine-to-Machine Communications: Architectures, Technology, Standards, and Applications*. CRC Press, 2014.

[45] A. Ali, W. Hamouda, and M. Uysal, "Next generation m2m cellular networks: challenges and practical considerations," *IEEE Communications Magazine*, vol. 53, no. 9, pp. 18–24, September 2015.

[46] A. Rajandekar and B. Sikdar, "A survey of mac layer issues and protocols for machine-to-machine communications," *IEEE Internet of Things Journal*, vol. 2, no. 2, pp. 175–186, April 2015.

[47] M. Weiner, M. Jorgovanovic, A. Sahai, and B. Nikoli, "Design of a low-latency, high-reliability wireless communication system for control applications," in *2014 IEEE International Conference on Communications (ICC)*, June 2014, pp. 3829–3835.

[48] J. Penttinen, *Wireless Communications Security: Solutions for the Internet of Things*. Wiley, 2016.

[49] Y. Oh and S. H. Son, "Allocating fixed-priority periodic tasks on multiprocessor systems," *Real-Time Systems*, vol. 9, no. 3, pp. 207–239, 1995.

[50] J. Liu, *Real-Time Systems*. Prentice Hall, 2000.

[51] D. Simchi-Levi, "New worst-case results for the bin-packing problem," *Naval Research Logistics*, vol. 41, no. 4, p. 579, 1994.

[52] M. Koseoglu, "Lower bounds on the lte-a average random access delay under massive m2m arrivals," *IEEE Transactions on Communications*, vol. 64, no. 5, pp. 2104–2115, May 2016.

[53] *Study on RAN Improvements for Machine Type Communications*, September 2011, 3GPP TR 37.868 V11.0.0.

[54] QoS Constrained Semi-Persistent Scheduling of Machine Type Communications in Cellular Networks. [Online]. Available: https://goo.gl/8ThVQB