Traffic Shifting based Resource Optimization in Aggregated IoT Communication

Amirahmad Chapnevis*, İsmail Güvenç†, and Eyuphan Bulut*
*Dept. of Comp. Science, Virginia Commonwealth University, Richmond, VA 23284
†Dept. of Elec. and Comp. Engineering, North Carolina State University, Raleigh, NC 27607
Email: {chapnevisa, ebulut}@vcu.edu, iguvenc@ncsu.edu

2020 IEEE 45th Conference on Local Computer Networks (LCN) | 978-1-7281-7158-6/20/$31.00 ©2020 IEEE | DOI: 10.1109/LCN48667.2020.9314781

***Abstract*—Aggregated Internet of Things (IoT) communication aims to use core network resources efficiently by providing cellular access to a group of IoT devices over the same subscriber identity. Leveraging the low data rates and long data sending intervals of IoT devices, several of the IoT devices in the same serving area of the core network are grouped together and take turns to send their data to their servers without causing overlaps in their communication. In this paper, we take this approach further and benefiting from the flexibility in data sending schedules, we aim to increase savings in cellular resources by shifting (delaying or performing earlier) the regular traffic patterns of IoT devices slightly. To this end, we consider two different traffic shifting models, namely, consistent and inconsistent shifting. We first solve the optimal aggregation of IoT devices under each model by using Integer Linear Programming (ILP). In order to avoid the high complexity of ILP solution, we then develop a heuristic based solution that runs in polynomial time. Through simulations, we show that heuristic based solution provides close to optimal results in various scenarios and shifting based aggregated communication offers more resource optimization (i.e., smaller number of bearers needed to connect all IoT devices) than the aggregated communication with no shifting.**

***Index Terms*—5G, cellular network, clustering, core network, Internet of Things (IoT), machine type communications (MTC).**

I. Introduction

Internet of Things (IoT) technology has enabled many devices to be connected to collect and exchange data in various applications including smart cities [1]–[3], environmental monitoring [4]–[6], and home automation [7]. The massive increase in the number of such machine-type devices (MTD) has generated new challenges in cellular communication due to limited wireless spectrum and scarce resources available in the core network of mobile operators. Several studies [8], [9] have been performed recently to address these challenges from different aspects. There are also several standardization efforts by organizations such as 3GPP and IEEE in order to develop the next generation (5G) IoT standards (e.g., LTE-M, NB-IoT [10]) and allow massive access.

In this paper, we study the efficient utilization of core network resources (e.g., data paths or *bearers* in Evolved Packet Core (EPC)) in order to provide scalable communication architecture for massive number of MTDs. Gateways in existing cellular core networks are primarily designed to handle the traffic from mobile users. That is, the resources and limitations are set to respond efficiently to the current communication characteristics of mobile users. However, the traffic characteristics of the machine type devices (MTDs) in the IoT network are different. Thus, especially considering that these devices tend to rarely send data, connectivity resources are wasted if each MTD directly connects to the macrocell base station (BS) and the core network individually. Note that putting the IoT devices into power saving mode (PSM) [11] (introduced in 3GPP Release 12 in order to optimize the device power consumption by turning its radio off) during the times they are not sending data will release the channel and reduce the load on the macro BS. However, as the device remains registered with the network, the core network resources will still be used. That is, for example in EPC, only the connection between Serving Gateway (SGW) and Mobility Management Entity (MME) will be deleted, but Packet Data Network Gateway (PGW) and MME will still keep the information about the device connection and continue consuming memory resources at these gateways.

One approach to connect such nearby IoT devices efficiently is to connect them to a local IoT gateway having a dedicated line and let them achieve their data communication with their servers and the rest of the Internet over this local gateway. This can be in the form of a star topology and can also be extended through forming a device-to-device (D2D) communication network [9], [12], [13] among these devices. Such an approach will work as long as the bandwidth requirements for devices could be supported by the utilized D2D technology (e.g., Bluetooth Low Energy (BLE), WiFi-direct) and the capacity of the single backhaul connectivity from the gateway to the macro BS can handle all traffic from the connected devices. On the other hand, it will only be applicable for nearby devices which are in D2D communication range of each other.

In order to provide a more scalable solution in wider areas and also use core network resources efficiently, recently, the concept of *aggregated communication* has been introduced [14] for IoT devices. It aims to connect a group of IoT devices with same data sending intervals over the same subscriber identity and have them take turns for their data communication. Core network treats the communication from all these devices as if it is coming from a single device which is turning on and off (i.e., establishing bearer and releasing it); thus, it maintains only a single bearer for all of them, yielding huge resource saving. This approach has also been extended for IoT devices that have different data sending intervals [15] for additional optimization in resource utilization. In this study,

we aim to take these approaches further and achieve additional saving by shifting the regular traffic patterns of IoT devices slightly (e.g., less than a threshold time), especially for IoT devices who have some flexibility in sending their periodically collected data to their servers within their long data sending intervals. We study two different shifting models (i.e., consistent and inconsistent shifting) and find the optimal grouping of IoT devices using Integer Linear Programming (ILP). We then develop a heuristic based solution which runs much faster and provides close to optimal results. We also perform simulations with different settings and show that traffic shifting, even it is small, can provide remarkable additional saving in the number of bearers actively used over no shifting based aggregation.

The rest of the paper is organized as follows. We provide background information and discuss the related work in Section II. In Section III, we discuss the details of the proposed traffic shifting based aggregated communication for IoT devices. We first discuss the ILP based optimal solution and then elaborate on the heuristic based solution that runs faster. In Section IV, we present the evaluation of the proposed approaches under different shifting models and different settings. Finally, we end up with conclusion and outline the future work in Section V.

II. Background

A. Aggregated IoT Cellular Communication

Overview. Aggregated IoT communication is first proposed in [14], [16] to efficiently use the core network resources for IoT devices which have usually low data rates and long data sending intervals. The IoT devices that have a common data sending interval (e.g., sensors from the same company performing the same function at different places) are assigned a common International Mobile Subscriber Identity (IMSI) and let the core network consider them as the same device. The data communication of each device over this common connection line is achieved by having them take turns without overlapping their traffic patterns.

Note that the IMSI sharing based aggregated communication reduces the utilization of core network resources such as the number of cellular bearers, for which there is usually a limit on core network gateways e.g., PGW in EPC. Considering all the IoT devices in the service region of a core network gateway, which usually covers hundreds of base stations or eNodeBs, it provides a resource optimization in a wider area compared to earlier approaches. On the other hand, in these studies [14], [16], only the devices that share a common data sending interval are considered and the list of devices that will share the same subscriber ID or IMSI (which is achieved at the initial provisioning of these devices with multiple instances of the same physical cellular SIM) are pre-determined and not allowed to change. In a more recent work [15], this aggregation method has been extended considering all IoT devices with varying data upload cycles and with a dynamically determined list of devices that will share the same subscriber ID. Dynamic grouping of devices is achieved through new generation subscriber ID solutions including but not limited to virtual SIMs [17] and e-SIM cards [18], [19]. These solutions help subscribers change their mobile operators without changing their SIM cards but could easily be used for online provisioning of the network connectivity for IoT devices and assign them a new subscriber ID dynamically [20].

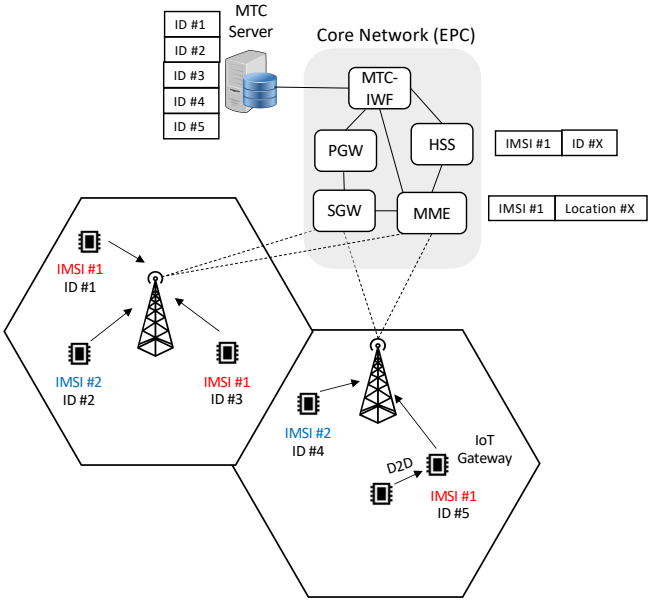


Fig. 1: Overview of aggregated IoT communication in EPC, as a representative of mobile core network.

Procedure updates. Once the MTDs that will share the same connectivity (and subscriber ID) are determined (by the mobile network operator (MNO) or by a central network authority if multiple MNOs are involved), the previous work [14]–[16] addresses the necessary minimal changes that need to be made in the traditional call flows of several operations under this IMSI sharing model.

- *Attach.* When a new IoT device turns on, it sends an attach request to the core network. If the current time slot is in use by another IoT device that is sharing the same IMSI with this new device, its request is rejected and a new request is made after an assigned back-off timer expires. The procedure is repeated until a successful attachment is accomplished.
- *Data Communication.* The time is divided into equal slots and each device sharing the same link takes turns to connect and send their data to their corresponding destinations. A guard time is introduced between the time slots to avoid potential overlap that may occur due to delay in communication.
- *Paging.* Home Subscriber Server (HSS) coordinates with MTC server to keep track of the active IoT device of an aggregated cellular line, and manages the paging of the right device accordingly.

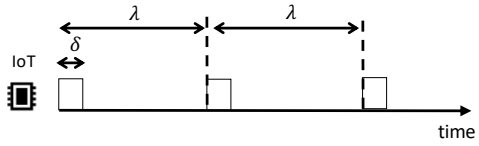
Consider the EPC network in Fig. 1, as a representative core network architecture which is currently the most common

system in use. The IoT devices that share the same IMSI are considered as the same device by the core network. However, the list of the IoT devices using the same IMSI are still being tracked by the MTC server in the background through the usage of external identifiers (EID) and MTC interworking function (MTC-IWF) [11] that is serving as an intermediary function between the core network and the MTC server. Note that MTC server does not deal with IP addresses and cellular IDs (e.g., IMSI), which is managed by PGW, and just uses external identifiers (EID) to communicate with the IoT devices. The mapping of IMSI and application port ID to EID is achieved through communication of MTC-IWF with HSS. The interested readers can refer to [14]–[16] for further details.

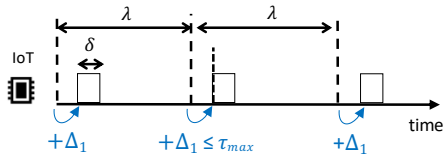
B. Related work

There are several studies proposing solutions to the incoming tsunami of connection demands from massive number of IoT devices. These solutions include modifications and re-architecturing of core network and its functions [21], separating the control and user planes with Software Defined Networks (SDN) and Network Function Virtualization (NFV) (e.g., [22], MMLite [23], CleanG [24], [25], Softcell [26]) and device side based solutions (e.g., virtual bearers [27], group-based communication [28]). While some of these approaches are promising and yet to be tested in actual deployments, most of them come with some limitations for practical applications. For example, the solution proposed in [27] requires devices to be in D2D communication range of each other, and the solution proposed in [28] requires devices to be in the same eNB service area. Similarly, while a lightweight, functionally decomposed and stateless MME design is proposed in [23], the optimization and resource saving happens in only one core network gateway, thus the solution is limited and does not provide benefit to the entire core network.

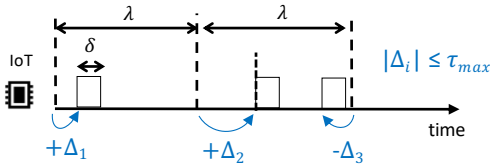
Different from these works, a more scalable and practical approach using an *aggregated communication* model is studied in [14]–[16] without changing the current architecture of core network drastically. The proposed aggregated model groups a set of IoT devices and let them share the same subscriber identity and take turns for their actual data communication. Since the data communication happens infrequently for most of the machine type IoT devices (e.g., humidity measurement in field two times a day) and there is usually some flexibility especially when the collected data is not critical, we consider the shifting of scheduled communication times (to an earlier or later time) slightly to further decrease the number of active cellular bearers used. To this end, we study both consistent and inconsistent shifting models and obtain the optimal grouping through an ILP-based solution and design a fast heuristic based solution. With simulations, we show that shifting based aggregated communication can provide remarkable saving in the number of bearers used over no shifting case, and heuristic solution can provide close to optimal results under different settings. Next, we elaborate on the shifting models, optimal ILP based solution and heuristic based approach.



(a) Original traffic without shifts



(b) Consistently shifted traffic



(c) Inconsistently shifted traffic

Fig. 2: The shifting models considered and their impact on the original traffic pattern.

III. Traffic Shifting based Aggregated Communication

A. Assumptions

IoT Traffic Model. We assume that there are M IoT devices (i.e., MTDs) denoted by set $G = \{I_1, I_2 \dots I_M\}$ and each of them sends their data (e.g., measurements, computations) to their server in some constant intervals. However, their data sending intervals and the required connectivity duration within each of these intervals could be different due to different application specific requirements (e.g., the size of the data collected). To this end, we assume that for each device i, the duration of data upload happens at every λ_i time units and each data upload occurs for a duration of δ_i time units, starting at s_i and ending at e_i, within each λ_i duration (i.e., $\delta_i = e_i - s_i$). We have chosen this model for the sake of simplicity, however, it could be extended to more complicated models (e.g., Gaussian distribution with a mean) which will be the subject of our future work. We assume that the time is also divided into equal slots and all time related parameters are a multiple of the slot size.

Traffic Shifting. As it is shown in Fig. 2, we consider two different types of shifting for the IoT traffic model considered:

- *Consistent shifting.* The timing of each data upload instance for an IoT device is consistently shifted with an amount less than a given time threshold, which is denoted with $\tau_{\max}$. This can result in data upload both earlier and later than the originally scheduled data upload time.

Notations	Description
I_i	MTD or IoT device i
M	Number of MTDs
G	The set of all MTDs
G_i	The group of MTDs on bearer i
λ_i	Data sending interval of MTD i
δ_i	Duration of data communication in each data sensing interval for MTD i
s_i	Starting time of data communication within each interval by MTD i
e_i	Ending time of data communication within each interval by MTD i
$\mathcal{T}$	Least Common Multiple (LCM) of data sending intervals (λ) of all MTDs
$\mathcal{T}_j$	Least Common Multiple (LCM) of data sending intervals (λ) of all MTDs in bearer or group j
b_i	Set to 1 if bearer i is used by at least one MTD and at any time (otherwise 0)
b_{ik}	Set to 1 if bearer i is used by at least one MTD at time slot k (otherwise 0)
b_{ijk}	Set to 1 if MTD i uses bearer j at the time slot k (otherwise 0)
$\tau_{\max}$	Maximum allowed shifting
$\mathcal{A}_j, \mathcal{U}_j, \mathcal{B}_j$	Active timeline, Utilization, and Border scores used in heuristic based approach
S	Overall score of merging two bearers in heuristic based approach

TABLE I: Notations and their descriptions.

- *Inconsistent shifting.* The timing of each data sent is delayed or scheduled earlier with a time difference less than the allowed shifting threshold. However, there is no consistency requirement. That is, the starting time of each data upload instance of an MTD can be shifted to an earlier time or delayed to a later time, independent from the decisions made for the timing of the other data upload instances of the same MTD.

Note that *inconsistent shifting* model gives more flexibility to the data uploads from each IoT device or MTD, hence lets more opportunity to group IoT devices without overlapping their traffic especially for those with varying data sending intervals. On the other hand, with *inconsistent shifting*, the optimization problem gets harder and the amount of information that needs to be maintained by each IoT device increases (i.e., they need to know the patterns of shifting amount needed in each upload). Thus, overhead in managing it can be more compared to that of *consistent shifting* especially when multiple IoT devices are grouped together with a very large common data sending interval. This interval can be computed by longest common multiple (LCM) of data sending intervals of all devices in the same group, which will be denoted by $LCM(\lambda_i, \forall i)$. The notations used throughout the paper and their descriptions are summarized in Table I.

B. ILP based Optimal Solution

The objective of aggregating the traffic from multiple MTDs is to minimize the number of bearers actively used by all devices under the given scenario and optimize the cellular resources. When there is no shifting allowed in the originally scheduled traffic patterns of MTDs, the grouping of the devices (which will define the number of bearers needed) will be possible to some extent, as there will be overlaps between the traffic patterns of different devices. If some of them can shift their uploading times slightly (i.e., less than $\tau_{\max}$) within their long data sending intervals, there will be more opportunity to decrease the number of groups and the number of actual bearers that will be used, and thus increase the resource saving. We define the optimization model for both consistent and inconsistent shifting models as:

$$\min \quad \sum_{j=1}^{M} b_j \tag{1}$$

$$\text{s.t.} \quad b_j = \min\left\{1, \sum_{k=1}^{\mathcal{T}} b_{jk}\right\}, \forall j \in [1, M] \tag{2}$$

$$b_{jk} = \min\left\{\sum_{i=1}^{M} b_{ijk}, 1\right\}, \forall j \in [1, M], \forall k \in [1, \mathcal{T}] \tag{3}$$

$$\sum_{i=1}^{M} b_{ijk} \leq 1, \forall j \in [1, M], \forall k \in [1, \mathcal{T}] \tag{4}$$

$$\exists! \Delta \in [-\tau_{\max}, +\tau_{\max}] : \sum_{d=1}^{\delta_i} b_{ij(r\lambda_i + ((s_i+\Delta+d) mod(\lambda_i))} = \delta_i \quad \forall i, j \in [1, M], \forall r \in [0, \mathcal{T}/\lambda_i - 1] \tag{5}$$

For Inconsistent Shifting only:

$$\sum_{k=1}^{\mathcal{T}} b_{ijk} = (0)||(\delta_i \mathcal{T}/\lambda_i), \forall i, j \in [1, M] \tag{6}$$

For Consistent Shifting only:

$$b_{ij((r-1)\lambda_i + d)} = b_{ij(r\lambda_i + d)}, \forall d \in [1, \lambda_i] \quad \forall i, j \in [1, M], \forall r \in [1, \mathcal{T}/\lambda_i - 1] \tag{7}$$

where,

$$\mathcal{T} = LCM\{\lambda_1, \ldots \lambda_M\}$$

$$b_{ijk} = \begin{cases} 1, & \text{if } I_i \text{ uses bearer } j \text{ at time slot } k, \\ 0, & \text{otherwise.} \end{cases}$$

Optimization formula in (1) aims to minimize the number of bearers used actively. The usage of each bearer (which could be up to M when each MTD uses a separate one) is determined by (2) and (3), by checking if there is at least one bearer using it at any time slot. (4) limits usage of each slot by a single MTD at most and (5) requires that there exists at least one and only one *shifting* (Δ) amount between $-\tau_{\max}$ and $\tau_{\max}$ which makes all δ_i consecutive slots utilized for a given MTD i (i.e., I_i) at a given bearer j.

Depending on the shifting model that will be used, there is also one more separate constraint defined for each. We use (7) to achieve consistent shifting between the different

data sending intervals of an MTD at the bearer it uses. If inconsistent shifting is allowed, then we simply discard (7), and let the (6) be the decider only, which requires that the total number of slots used by each MTD i at each bearer j should be either equal to zero or to the total data communication need in the entire common time frame (i.e., $\mathcal{T}$), which is simply computed by multiplying the data communication need (i.e., δ_i) in each data sending interval and the total number of repetitions of that MTD's data sending (i.e., $\mathcal{T}/\lambda_i$).

Once all IoT device traffic characteristics are known by the mobile network operator (MNO), it can run this optimal model and determine which IoT device will be in which bearer and update their network registration information (e.g., IMSI) through an online provisioning process as discussed in Section II. On the other hand, while the proposed ILP model will find the optimal (i.e., minimum) number of bearers possible that can allocate all MTD traffic, its running time will be very long even with a small number of MTDs (e.g., 10-15) in the network. Thus, if the optimization model has to be run frequently (e.g., when the set of IoT devices or their traffic characteristics change), it may not be a practical solution. To this end, in the next section, we provide a heuristic based solution with a reduced complexity.

C. Greedy Heuristic based Solution

1) Overview: In order to aggregate the traffic of multiple MTDs on the minimum number of bearers possible, we consider an iterative approach and try to select the best option at every step greedily. Initially, we assume that each device is on a separate bearer or group. Then, we first find all eligible bearer pairs that can be merged under the current shifting model. This is determined by checking if there is an overlapping allocated time slot by both of these bearers. Out of all eligible bearer pairs, we then find the pair that provides the best score and merge these two bearer traffic into one bearer (we call it *root bearer*), and release the other one. In consecutive steps, we go through all other single MTD bearers again and check if they are eligible to be merged with this root bearer traffic. Among eligible ones, we find the one that gives the best score and bring its traffic into the root bearer. This process continues until no more single MTD bearer is eligible to be added into the current root bearer. Then, we continue the process with the formation of a new root bearer out of the remaining single MTD bearers not aggregated yet. We again find the pair of bearers that gives the best score, merge their traffic on one of them and try to add other bearer traffic on this bearer one by one until no more eligible bearer remains. Here, note that, if there is no eligible pair of bearers that can be merged and assigned as root bearer, we stop the entire process and leave each of the single MTD bearers as a separate bearer without any aggregation. This greedy approach is provided in Algorithm 1. Root bearer formation is done in lines 4-12 and addition of other bearers on it one by one is done in lines 17-34. If no more root bearer that can be obtained by merging two single MTD bearers is possible, each remaining MTD is kept on its own bearer as shown in lines 37-41.

Algorithm 1: Greedy Heuristic-based Aggregation

```
1  G = {I_1, I_2 ... I_M},  S_max = 0
2  α = 0 // Next bearer id to assign MTDs
3  while |G| > 0 do
4      foreach (I_x, I_y) s.t. I_x, I_y ∈ G, I_x ≠ I_y do
5          if I_x and I_y are eligible to be merged then
6              Find score S(I_x, I_y) based on selected
               shifting model
7              if S(I_x, I_y) > S_max then
8                  S_max = S(I_x, I_y)
9                  (I_x^max, I_y^max) = (I_x, I_y)
10             end
11         end
12     end
13     if S_max ≠ 0 then
14         G_α = {I_x^max, I_y^max}
15         G = G \ G_α
16         E = G,  S_max = 0
17         while |E| > 0 do
18             foreach I_z ∈ E do
19                 if I_z can be merged on G_α then
20                     Find score S(I_z, G_α) based on
                       selected shifting model
21                     if S(I_z, G_α) > S_max then
22                         S_max = S(I_z, G_α)
23                         I_z^max = I_z
24                     end
25                 end
26             end
27             if S_max ≠ 0 then
28                 G_α = G_α ⋃ {I_z}
29                 E = E \ {I_z}
30                 S_max = 0
31             else
32                 E = ∅
33             end
34         end
35         α = α + 1
36     else
37         foreach I ∈ G do
38             G_α = {I}
39             G = G \ G_α
40             α = α + 1
41         end
42     end
43 end
```

2) Score Function: In this iterative and greedy heuristic based approach, the critical part is the score function. As we target to merge as many MTD traffic as possible on a single bearer, we select the root bearer as well as the next added bearers to it such that the allocated time slots in the entire timeline are distributed in a way that adding new MTD traffic will be easier. To this end, we consider three different criteria:

- *Active Timeline* ($\mathcal{A}$): It is the duration from the first allocated time slot until the last allocated one. So, for bearer or group j, G_j, we find the minimum start time and maximum end time of all IoT devices on this bearer, and take the difference:

$$\begin{aligned}\mathcal{A}_j &= e^j_{max} - s^j_{min}, \text{where} \\ & s^j_{min} = \min\{s_i,\ \forall I_i \in G_j\} \\ & e^j_{max} = \max\{e_i,\ \forall I_i \in G_j\}\end{aligned}$$

- *Utilization* ($\mathcal{U}$): It refers to the percentage of time slots allocated within the active timeline. Given that $b_{ijk} = 1$ when MTD i allocates bearer j at time slot k, for all MTDs on a given bearer or group j, G_j, we calculate

$$\begin{aligned}\mathcal{U}_j &= \left(\sum_{k=s^j_{min}}^{e^j_{max}} a_k\right)/\mathcal{A}_j, \text{ where} \\ a_k &= \begin{cases}1, & \text{if } \exists I_i \in G_j \text{ s.t. } b_{ijk} = 1 \\ 0, & \text{otherwise}\end{cases}\end{aligned}$$

- *Border Score* ($\mathcal{B}$): This indicates how close the active timeline is to the end points of the entire timeline. As the allocated time slots get close to the sides of the entire timeline, the likelihood of allocating another MTD to the same bearer increases. Thus, we first find the minimum of distances to the start and end of entire timeline from the start and end of active timeline and take their sum. That is, for bearer or group j, G_j, we compute

$$\mathcal{B}_j = \min\{\mathcal{T}_j - s^j_{min}, s^j_{min}\} + \min\{\mathcal{T}_j - e^j_{max}, e^j_{max}\}$$

We consider these criteria in a prioritized manner. That is, we first prefer the cases that provide shorter active timeline. Then, for those cases with the same active timeline, we prefer higher utilization. Finally, for cases where active timeline duration and the utilization is the same, we give priority to cases that are closer to the borders. In order to reflect this prioritization in the score function, we define it as:

$$S = 2\mathcal{T}(\mathcal{T} - \mathcal{A}) + \mathcal{U}\mathcal{A} + \frac{1}{2 + \mathcal{B}} \tag{8}$$

Here, as minimum possible $\mathcal{B}$ is zero, i.e., when the first and the last time slots of the entire timeline are allocated, third term can be at most 1/2. However, any increase in $\mathcal{U}$ will increase the second term minimum by 1 (i.e., one more time allocated within the active timeline), and it will change the total score value more than the maximum of third term can do (i.e., 1/2) thus will be preferred. Finally, any decrease in $\mathcal{A}$ will make first term contribute more than maximum possible contribution (i.e., $\mathcal{T}$) of second term, thus it will give priority to a decrease in $\mathcal{A}$ over an increase in $\mathcal{U}$.

Consider the example in Fig. 3 with two MTDs. We compute active timeline as $\mathcal{A} = 12$, utilization as $\mathcal{U} = (2 + 5 + 2)/12 = 75\%$, and border score as $\mathcal{B} = 4 + 4 = 8$. Then, the score function is, $S = 2 \times 20 \times (20 - 12) + 9/12 \times 12 + 1/(2 + 8) = 329.1$.

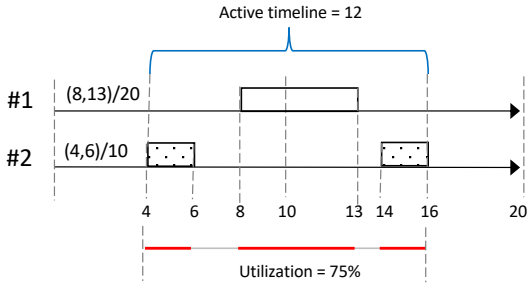
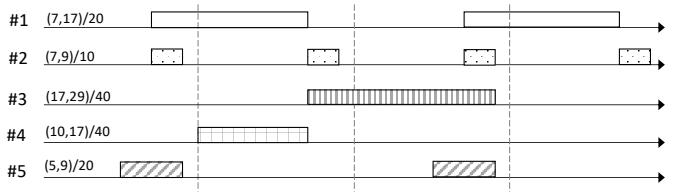


Fig. 3: Score calculation on an example with two MTDs.

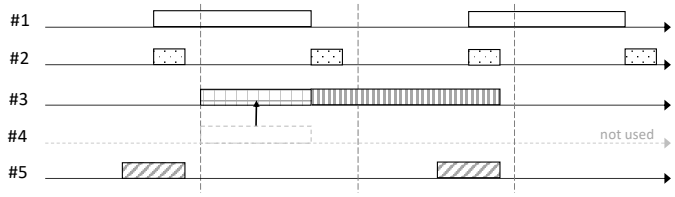
3) Running time: There can be at most $\binom{M}{2}$ single MTD bearer pairs that need to be checked to find the best candidate for a root bearer. If any other single MTD bearers can be added to the current root bearer, the cost of finding the best one will be less than $\mathcal{O}(M)$. If none can be added to the root bearer and a new root bearer needs to be determined over score comparison of pairs in the remaining set of single MTD bearers, there will be another $\binom{M-2}{2}$ pairwise comparison. If the process always continues with root bearer selection (formed by two single MTD bearers) without adding any third bearer, which will be the worst case scenario, overall there will be $\mathcal{O}(M^3)$ eligibility check and score calculation.

Note that score calculation will have the same cost for different shifting models, however, eligibility check will have different cost. For no shifting, it will only require one to one comparison of each time slot within $\mathcal{T}$ to see if there is an overlap. In consistent shifting, each bearer traffic can be shifted between $[-\tau_{max}, \tau_{max}]$ range of its original traffic pattern. Thus, it will require comparison of $\mathcal{O}(\tau^2_{max})$ combinations, each with $\mathcal{T}$ cost. With inconsistent shifting, as each repetition of the data sending interval within the common timeline can be shifted in different amounts, there will be huge cost to cover all cases. Thus, we apply another heuristic for inconsistent shifting eligibility check in order to reduce the complexity while still benefiting from inconsistent shifting of repeating data sending intervals. To this end, we first find the best consistent shifting amount of all repetitions of data sending intervals for an MTD. Then, by keeping the first repetition at that shifting amount, we consider the consistent shifting of remaining repetitions of data sending intervals and find the best consistent shifting amount for them. We continue until all repetitions are assigned a shifting amount by this way. At the end, through this approach we decrease the complexity of eligibility check for inconsistent shifting from $\mathcal{O}(\tau_{max}{}^{\frac{\mathcal{T}^2}{\lambda_i \lambda_j}})$ to $\mathcal{O}(\tau^2_{max} \max\{\frac{\mathcal{T}}{\lambda_i}, \frac{\mathcal{T}}{\lambda_j}\})$. Through simulations, we also find that this extended heuristic based eligibility check ($H2$) performs as good as high complexity one ($H1$) in around 90% of the cases.

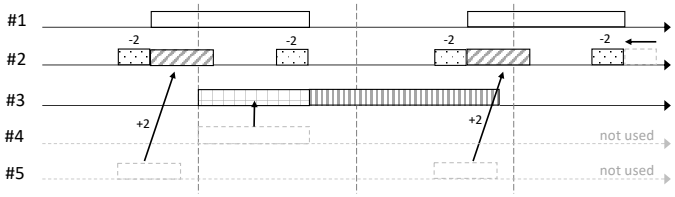
To summarize, the overall run time complexity of heuristic solution in no shifting, consistent shifting and inconsistent shifting based aggregation models are $\mathcal{O}(\mathcal{T}M^3)$, $\mathcal{O}(\tau^2_{max}\mathcal{T}M^3)$, and $\mathcal{O}(\tau^2_{max}\max\{\frac{\mathcal{T}}{\lambda_i}, \frac{\mathcal{T}}{\lambda_j}\}\mathcal{T}M^3)$, respectively.



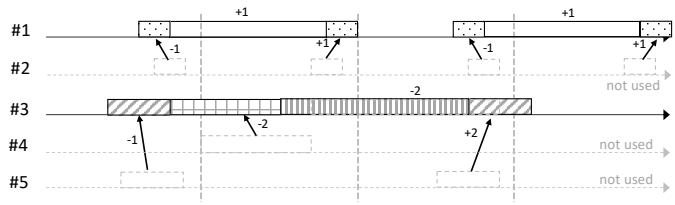
(a) Original traffic without grouping



(b) Grouping MTDs with no shifting



(c) Grouping MTDs with consistent shifting



(d) Grouping MTDs with inconsistent shifting

Fig. 4: The reduction in the number of active bearers needed to carry the traffic of 5 MTDs under different shifting models.

D. Toy Example

In this part, we provide a sample run of heuristic based solution on an example set of MTDs shown in Fig. 4. We have 5 MTDs, each of which initially uses a separate bearer as shown in Fig. 4a. The traffic patterns are also shown in Fig. 4a. That is, for example, MTD #1 is sending its data between 7-17th time units in every 20 time units. As the LCM of the data sending intervals of these 5 MTDs is 40, we show all the repetitions of data communication for each device in this entire common timeline. When the heuristic based algorithm is run with no shifting model, it finds that there are only three eligible pairs that can be merged to obtain the first root bearer, namely, (#2, #4), (#3, #4), and (#4, #5). Calculating their score function, it selects (#3, #4) as the one with the best score and merges their traffic on one of these bearers. Trying to add other bearers on this root bearer does not help further, thus a new pairwise checking process starts among the remaining single MTD bearers (i.e., #1, #2, #5). As there is no eligible pair of bearers that can be merged without any shifting, each remaining MTD is kept on a separate bearer and the algorithm completes for this case with 4 active bearer usage for 5 MTD traffic as shown in Fig. 4b.

With consistent shifting, with $\tau_{max} = 2$, heuristic based algorithm considers shifting of each MTD's traffic in range of $[-\tau_{max}, +\tau_{max}]$ during each pairwise merge eligibility check of bearers. This time, in addition to the previous three pairs found in no shifting case, thanks to the flexibility through shifting, the algorithm also finds three more pairs that are eligible to be merged, namely, (#1, #5), (#2, #5), and (#3, #5). However, (#3, #4) still provides the best score, thus is selected to form the initial root bearer. As no other bearer can be added to this root bearer, another pairwise merge eligibility check starts again with remaining single MTD bearers. Out of two eligible pairs (i.e., (#1, #5) and (#2, #5)), (#2, #5) provides the best score when MTD #5's traffic is shifted +2 time slots and MTD #2's traffic is shifted -2 time slots consistently for each of their data sending time repetitions. After they are merged, since MTD #1 cannot be added on this new root bearer, and it is the only one left, it is left on its own bearer. Overall, only 3 bearers are used for 5 MTD traffic as shown in Fig. 4c.

Finally, with inconsistent shifting, all pairs of bearers except one, i.e., (#1, #4) will be eligible to be merged initially thanks to the independent shifting flexibility of each repetition of data sending times of each MTD traffic. Again, (#3, #4) is selected out of all pairs, as it still provides the best score. Once they are merged, MTD #5 could also be added on this bearer, with -1 time slot shifting for its first data sending time and with +2 time slot shifting for its second data sending time. Note that MTD #3 and MTD #4 are also shifted -2 time slots compared to their original schedule. Once no more single MTD bearer can be added on this bearer, a new root bearer selection process starts. Since there are only two remaining single MTD bearers left, i.e., MTD #1 and MTD #2, and they are eligible to be merged under inconsistent shifting with given $\tau_{max} = 2$, they are selected with the shifting amounts that provide the best score to their merging. That is, MTD #2 is shifted -1 time slots for its first and third data sending times, and +1 time slots for its second and fourth data sending times. MTD #1 is shifted +1 time slots consistently to accommodate MTD #2 on its own bearer. At the end, only 2 bearers are used, yielding 60% of resource (i.e., bearer) saving for 5 MTDs as shown in Fig. 4d.

Note that, when we run ILP based solution on this example, we also receive the same number of bearer usage in each setting as in heuristic based algorithm. Heuristic based solution may not always find the optimal solution as ILP solution, however, as it will be shown in simulations, it provides close to optimal results in most of the settings.

Parameter	Traffic Load		
	Low	Medium	High
Data communication per interval (δ in % within λ)	10-15%	15-25%	25-50%
Number of MTDs (M)	5-50		
Maximum shifting allowed (τ_{max})	0-6 time slots		
Data sending interval (λ) array	{10,20,40} time slots		
Start time for data sending (s_i)	Uniformly distributed in λ_i		
End time of data sending (e_i)	$s_i + \delta_i$ if it is $\leq \lambda_i$		

TABLE II: Simulation parameters.

IV. Evaluations

In order to evaluate the proposed traffic shifting based aggregated IoT communication and compare the performance of heuristic based approach to the ILP based solution, we develop a custom simulator in Java and perform simulations under different settings.

A. Simulation Setting

The simulation parameters and their values are summarized in Table II. To determine the data upload pattern of each MTD in the format of $[s_i - e_i]/\lambda_i$ with $\delta_i = e_i - s_i$, we first set the data upload/sending interval to a value randomly selected from the set $\{10, 20, 40\}$min. Then, we randomly assign a data communication duration, δ_i, from a given range. To this end, we use three different traffic loads; namely, (i) low, (ii) medium and (iii) high. In the low traffic case, we assume 10-15% of the data sending interval or λ_i is used for data communication, while 15-25% and 25-50% is considered for medium and high traffic loads, respectively. Then, to decide the start time of the data communication within the data sending interval, we select a random value from $[0, \lambda_i - \delta_i]$ and set s_i to that value. The end time of data communication is then set to $s_i + \delta_i$ automatically. For main simulations, we use an MTD count from 5 to 50, as running ILP solution takes very long when MTD count is more. However, we also provide some results with larger number of MTDs for only heuristic based approach.

B. Performance Metrics

In order to evaluate the performance of the proposed traffic shifting based aggregated communication models, we use the *percentage of saving* in the number of cellular lines (i.e., bearers) as the main metric. For a given MTD count M, if the number of bearers sufficient to carry the traffic from all MTD devices with the considered aggregated traffic model is found as X, then the saving is defined as

$$\left(\frac{M - X}{M} \times 100\right)\%.$$

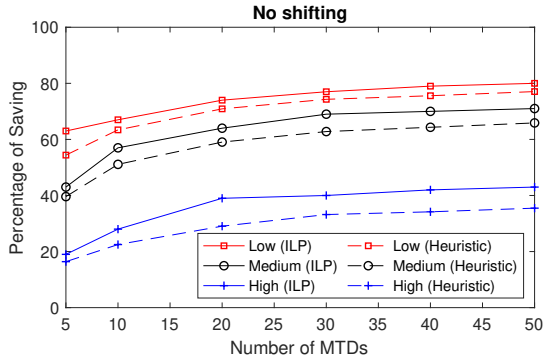
We look at the impact of number of MTDs, maximum threshold allowed for shifting and the impact of traffic load on this metric for each of the aggregation models, namely, (i) no shifting, (ii) consistent shifting, (iii) inconsistent shifting. We compare the performance of ILP based optimal solution and heuristic based approach in each setting. Moreover, we show the running time comparison of these solutions in different settings. All results presented are average of 20 runs.

Note that while aggregated IoT communication has previously been studied in [14], [16] with a no shifting model, their solution assumes that only MTDs with the same data sending interval (λ) and the same data communication duration (δ) within each interval will share the same bearer. These studies mainly focus on modifications of call flows to realize IMSI sharing based aggregated communication and do not propose how to actually group IoT devices if their traffic patterns are different, thus their solution is not applicable to our setting directly. Because of this, we could not compare the proposed solutions with an existing work in the literature but no shifting case could be considered as a benchmark solution thus we apply our solutions with $\tau_{max} = 0$ to obtain no shifting results and understand the additional savings offered by shifting based models.

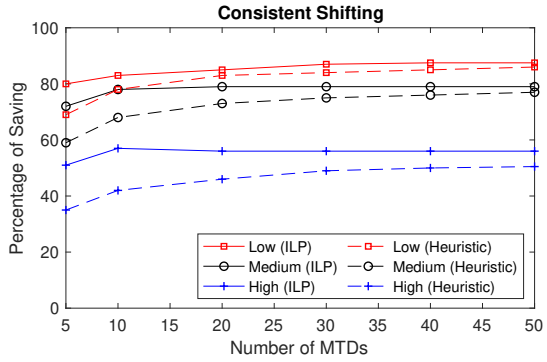
C. Results

Fig. 5 shows the impact of number of MTDs on the percentage of saving with different traffic loads. The results show that as the MTD count increases in the system, in no shifting case, the percentage of saving increases and finally converges to a value. The rate of increase however is different in different traffic loads. While the highest percentage of saving is achieved in low traffic model, as the number of MTDs increases the increase in the percentage of saving is the least in the low traffic model and the highest in the high traffic model. This is expected because as the traffic load increases, the opportunity to aggregate more MTDs in a single bearer increases more compared to the case in low traffic model, which already achieves a high percentage of saving with smaller number of MTDs. With consistent and inconsistent shifting we clearly see more saving than it is in no shifting case. Inconsistent shifting achieves slightly higher saving even with a small maximum shifting threshold (i.e., $\tau_{max} = 3$). Moreover, the saving converges to a value quickly as MTD count increases in both cases. On the other hand, we clearly see that heuristic based solution can provide close to optimal results in most of the cases and exhibits a similar trend. The gap between heuristic and ILP results is larger in high traffic case as it gets harder for the heuristic based solution to find better groupings in the highly utilized timelines of MTDs.

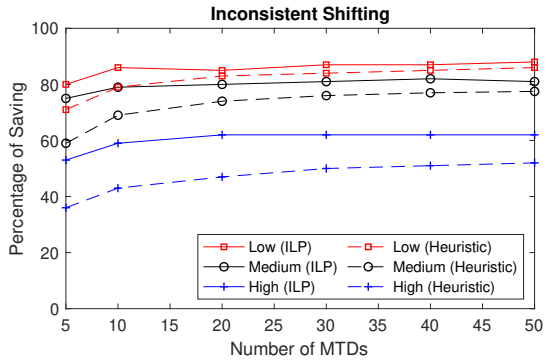
Fig. 6 shows the impact of τ_{max} when MTD count is fixed at 20 (we used a small MTD count not to have very long run time for ILP solution). Note that in the case of no shifting the results will not change but we are providing them to observe the benefit of proposed models over this benchmark model. We see that as threshold increases, there is more saving achieved in all traffic load models. However, we see that in low traffic, the convergence happens more quickly than in medium traffic whose convergence happens more quickly than high traffic case. The saving achieved with inconsistent shifting over consisting shifting is also much clear when the traffic load is higher as inconsistent shifting's flexibility will be more



(a)

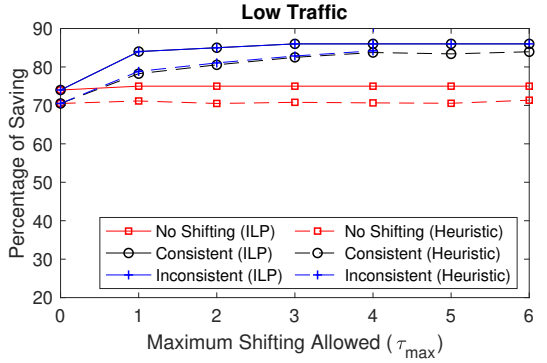


(b)

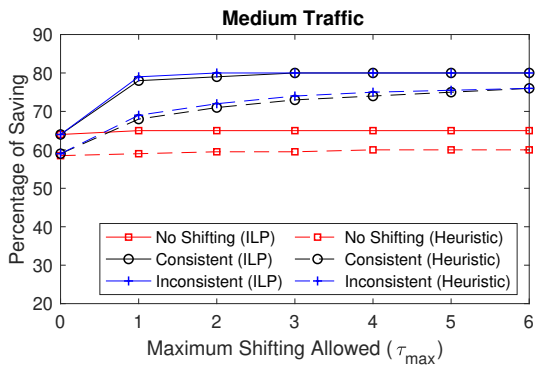


(c)

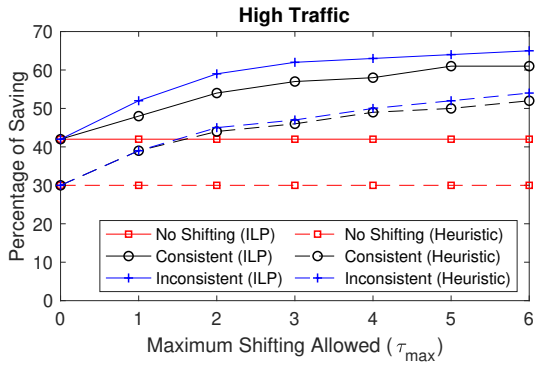
Fig. 5: Percentage of savings in the number of cellular lines used with (a) no shifting, (b) consistent shifting, and (c) inconsistent shifting based aggregation considering low, medium and high traffic patterns ($\tau_{max} = 3$ for (b) and (c)).



(a)



(b)

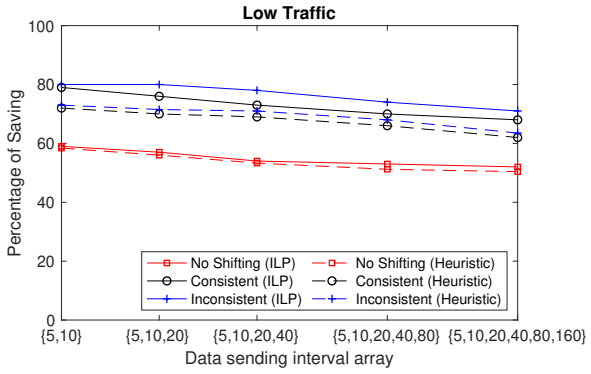


(c)

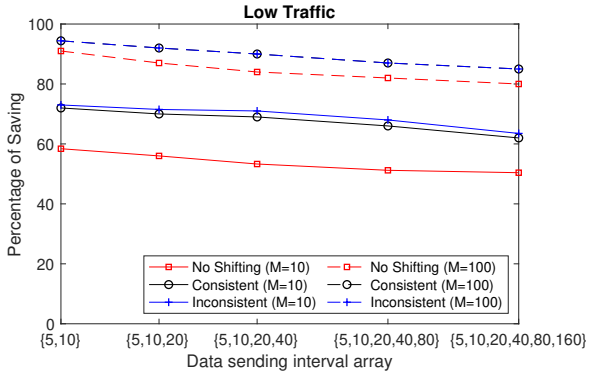
Fig. 6: Impact of maximum shifting threshold ($\tau_{\max}$) on the percentage of savings in the number of cellular lines used with (a) low, (b) medium and (c) high traffic patterns (M = 20).

effective in crowded schedules. Moreover, heuristic solution in general provides closer results to ILP solution. However, as the traffic density gets higher, we see that especially with no shifting model, the gap between heuristic and ILP results increases.

In Fig. 7 we look at the impact of the array from which the data sending intervals of the MTDs are selected on the percentage of saving. As both graphs show, with more options and larger λ values, the saving reduces in all algorithms but inconsistent shifting provides the best. In Fig. 7a, we observe that heuristic approach achieves within 10% range of ILP results in consistent and inconsistent shifting, while providing very close results to ILP in no shifting case. In Fig. 7b, we show heuristic only results with 10 and 100 MTDs as getting ILP results with 100 MTDs was not possible due to very long time. As Fig. 7b shows, with more MTDs, aggregated communication offers more benefit in all cases including no shifting case. This is because with more MTDs, there are more opportunities to group more MTDs. The performance of inconsistent and consistent shifting get very close with more

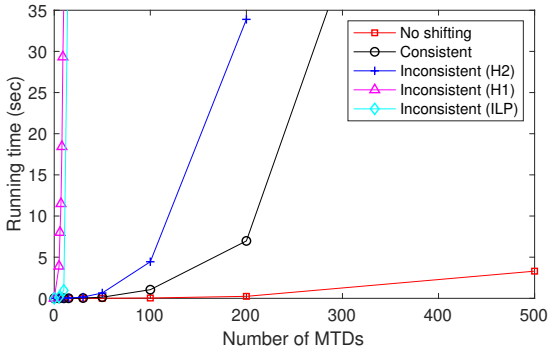


(a)

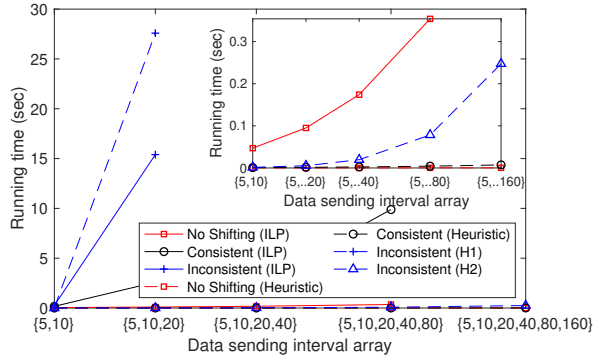


(b)

Fig. 7: a) Impact of data sending interval array on the percentage of saving ($M = 10$ and $\tau_{\max} = 3$), b) Percentage of saving obtained by Heuristic algorithm with 10 and 100 MTDs using different data sending interval arrays ($\tau_{\max} = 3$).



(a)



(b)

Fig. 8: Running time comparison a) with different number of MTDs, and b) with different data sending interval arrays (M=10). $\tau_{\max} = 3$ in both graphs.

MTDs and no shifting case also can provide closer results to them.

Finally, in Fig. 8, we compare the running times of ILP and heuristic based solutions. From Fig. 8a, we clearly see that heuristic based solutions run much faster than ILP solutions. Here, we also show the running time difference of inconsistent model with the extended heuristic (shown with H2, which is the one used in earlier graphs as well) discussed in Section III-C3 and without it (shown with H1). We see that running inconsistent shifting without extended heuristic is as costly as ILP model thus it is not applicable in practice for large-scale systems. However, with extended heuristic it could be applicable relatively easily. Fig. 8b shows the impact of array sizes on running times. As the results clearly show, ILP models have longer running times compared to heuristic models. Heuristic approach (i.e., H2) with inconsistent shifting indeed runs faster than ILP model for no shifting. Moreover, as it is shown in Fig. 9, H2 outperforms (i.e., results in fewer or equal number of active bearers) H1 in more than 90% of the cases on average in various scenarios. Overall, these results show that heuristic approach, even with inconsistent shifting

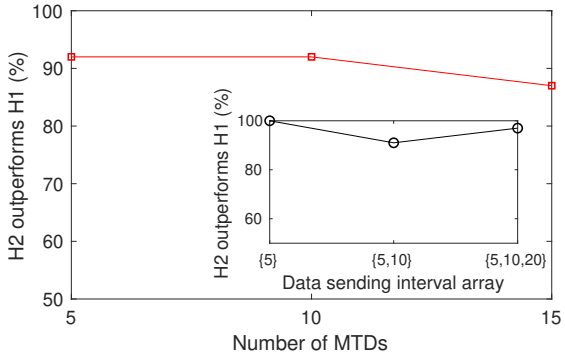


Fig. 9: Comparison of extended heuristic (H2) for inconsistent shifting model to high complexity heuristic version (H1) in terms of aggregation performance.

model, is applicable for large scale systems thanks to its much faster running time while producing close to optimal results.

V. Conclusion

In this paper, we study traffic shifting based aggregated communication model for IoT devices. The proposed model not only lets the devices use the same subscriber identity and take turns during their communication with the core network but also considers slight shifting in the original traffic patterns of devices for further saving in the resource utilization, namely the number of actively used bearers, in the core network. We consider two different shifting models which consistently and inconsistently shift the traffic for the devices at every data upload time, respectively. Using ILP, we obtain the optimal grouping of IoT devices under each shifting model. We also develop a heuristic based solution with a polynomial time complexity. Through simulations, we show that we can obtain up to 40% additional saving with shifting models and proposed heuristic based solution runs fast, is scalable and can provide closer to optimal ILP results in most of the scenarios.

In our future work, we will consider more complicated traffic models and deploy the proposed system on real devices. We will also evaluate the performance of the proposed system and algorithms in dynamic environments in which the existing IoT devices can leave and new IoT devices can join the network, thus groupings need to be updated.

VI. Acknowledgments

This work was supported in part by National Science Foundation awards CNS-1815603, CNS-1814727.

References

[1] W. Ejaz, M. Naeem, A. Shahid, A. Anpalagan, and M. Jo, "Efficient energy management for the Internet of Things in smart cities," *IEEE Communications Magazine*, vol. 55, no. 1, pp. 84–91, 2017.

[2] F. Yucel and E. Bulut, "Clustered crowd gps for privacy valuing active localization," *IEEE Access*, vol. 6, pp. 23 213–23 221, 2018.

[3] F. Yucel, M. Yuksel, and E. Bulut, "QoS-based budget constrained stable task assignment in mobile crowdsensing," *IEEE Transactions on Mobile Computing*, 2020.

[4] S. M. Hernandez and E. Bulut, "Lightweight and standalone IoT based WiFi sensing for active repositioning and mobility," in *21st International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM) (WoWMoM 2020)*, Cork, Ireland, Jun. 2020.

[5] F. Montori, L. Bedogni, and L. Bononi, "A collaborative Internet of Things architecture for smart cities and environmental monitoring," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 592–605, 2017.

[6] S. M. Hernandez and E. Bulut, "Performing wifi sensing with off-the-shelf smartphones," in *2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*. IEEE, 2020, pp. 1–3.

[7] A. Yang, C. Zhang, Y. Chen, Y. Zhuansun, and H. Liu, "Security and privacy of smart home systems based on the Internet of Things and stereo matching algorithms," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 2521–2530, 2019.

[8] Y. Xiao, M. Krunz, and T. Shu, "Multi-Operator network sharing for massive IoT," *IEEE Communications Magazine*, vol. 57, no. 4, pp. 96–101, 2019.

[9] R. M. Huq, K. P. Moreno, H. Zhu, J. Zhang, O. Ohlsson, and M. I. Hossain, "On the benefits of clustered capillary networks for congestion control in machine type communications over LTE," in *24th International Conference on Computer Communication and Networks (ICCCN)*. IEEE, 2015, pp. 1–7.

[10] 3GPP, "Standards for IoT," Dec. 2016. [Online]. Available: http://www.3gpp.org/news-events/3gpp-news/1805-iot_r14

[11] 3GPP, "Architecture enhancemens to facilitate communication with packet data networks and applications," v15, 2017. [Online]. Available: TS23.682

[12] F. Hussain and A. Ferworn, "Distributed slot allocation in capillary gateways for Internet of Things networks," in *IEEE 84th Vehicular Technology Conference, VTC Fall 2016, Montreal, QC, Canada, September 18-21, 2016*. IEEE, 2016, pp. 1–6.

[13] O. A. Amodu and M. Othman, "Machine-to-machine communication: An overview of opportunities," *Comput. Networks*, vol. 145, pp. 255–276, 2018. [Online]. Available: https://doi.org/10.1016/j.comnet.2018.09.001

[14] M. Ito, N. Nishinaga, Y. Kitatsuji, and M. Murata, "Reducing state information by sharing IMSI for cellular IoT devices," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 1297–1309, 2016. [Online]. Available: https://doi.org/10.1109/JIOT.2016.2587823

[15] E. Bulut and I. Güvenç, "Dynamically shared wide-area cellular communication for hyper-dense IoT devices," in *Proc. of the IEEE 43rd Conference on Local Computer Networks Workshops (LCN Workshops)*. IEEE, 2018, pp. 64–69.

[16] M. Ito, N. Nishinaga, Y. Kitatsuji, and M. Murata, "Aggregating cellular communication lines for IoT devices by sharing IMSI," in *2016 IEEE International Conference on Communications, ICC 2016, Kuala Lumpur, Malaysia, May 22-27, 2016*. IEEE, 2016, pp. 1–7. [Online]. Available: https://doi.org/10.1109/ICC.2016.7510812

[17] TechNews, "Xiaomi's miui now features virtual sim card for overseas travels," 2017. [Online]. Available: http://technews.co/2015/03/25/xiaomis-miui-now-features-virtual-sim-card-for-overseas-travels/.

[18] McKinsey, "E-sim for consumers – a game changer in mobile telecommunications?" 2016. [Online]. Available: https://www.mckinsey.com/industries/telecommunications/our-insights/e-sim-for-consumers-a-game-changer-in-mobile-telecommunications

[19] GSMA, "Remote sim provisioning for machine to machine," 2017. [Online]. Available: http://www.gsma.com/connectedliving/embedded-sim/.

[20] L. Militano, G. Araniti, M. Condoluci, I. Farris, and A. Iera, "Device-to-device communications for 5g internet of things," *EAI Endorsed Trans. Internet Things*, vol. 1, no. 1, pp. 1–15, 2015.

[21] S. Sakurai, G. Hasegawa, N. Wakamiya, and T. Iwai, "Performance evaluation of a tunnel sharing method for accommodating M2M communication to mobile cellular networks," in *Proc. IEEE GLOBECOM Workshops*. IEEE, 2013, pp. 157–162.

[22] V.-G. Nguyen, A. Brunstrom, K.-J. Grinnemo, and J. Taheri, "SDN/NFV-based mobile packet core network architectures: A survey," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1567–1602, 2017.

[23] V. Nagendra, A. Bhattacharya, A. Gandhi, and S. R. Das, "MMLite: A scalable and resource efficient control plane for next generation cellular packet core," in *Proc. of the 2019 ACM Symposium on SDN Research*, 2019, pp. 69–83.

[24] A. Mohammadkhan and K. K. Ramakrishnan, "Re-architecting the packet core and control plane for future cellular networks," in *27th IEEE International Conference on Network Protocols, ICNP 2019, Chicago, IL, USA, October 8-10, 2019*. IEEE, 2019, pp. 1–4.

[25] A. Mohammadkhan, K. K. Ramakrishnan, A. S. Rajan, and C. Maciocco, "CleanG: A clean-slate EPC architecture and controlplane protocol for next generation cellular networks," in *Proceedings of the 2016 ACM Workshop on Cloud-Assisted Networking, CAN@CoNEXT 2016, Irvine, California, USA, December 12, 2016*. ACM, 2016, pp. 31–36.

[26] X. Jin, L. E. Li, L. Vanbever, and J. Rexford, "Softcell: Scalable and flexible cellular core network architecture," in *Proceedings of the ninth ACM conference on Emerging networking experiments and technologies*, 2013, pp. 163–174.

[27] K. Samdanis, A. Kunz, M. I. Hossain, and T. Taleb, "Virtual bearer management for efficient MTC radio and backhaul sharing in LTE networks," in *Proc. IEEE Int. Symp. Personal Indoor Mobile Radio Commun. (PIMRC)*, 2013, pp. 2780–2785.

[28] Y. Jung, D. Kim, and S. An, "Scalable group-based machine-to-machine communications in LTE-advanced networks," *Wireless Networks*, vol. 25, no. 1, pp. 63–74, 2019.