Efficient Network Planning for Internet of Things With QoS Constraints

Ilias Gravalos, Prodromos Makris, Kostas Christodoulopoulos, and Emmanouel A. Varvarigos

***Abstract*—In the Internet of Things (IoT) era, a vast number of (smart) end devices forward their traffic to the Internet, either by direct communication to LTE networks, or by multihop transmissions to a specific gateway. Acquiring both types of communication capabilities for IoT end devices would be unnecessarily costly. Instead, for a cost effective IoT infrastructure, only devices performing as gateways could be fully equipped with such capabilities, while the remaining devices could have simple low cost wireless transceivers, of differing transmission specifications, to forward/relay the traffic toward a gateway. Furthermore, the IoT devices network should comply with specific quality of service (QoS) requirements, specified for each IoT device. In this context, the IoT network planning problem, where we have to select the number of gateways and their locations along with respective transceivers for IoT devices, is key to provide a low cost and QoS aware IoT infrastructure. We formulate the planning problem as an integer linear program (ILP) that minimizes the total cost of the devices deployed in the network, while achieving the mandatory QoS requirements. We also present a heuristic algorithm of lower complexity that was observed to provide solutions near the optimal ones, in scenarios that we tracked optimal solutions with the ILP. A variety of performance evaluation results exhibits the effectiveness of the proposed algorithms in terms of network cost and efficiency.**

***Index Terms*—Cost optimization, fog computing, gateway placement, Internet of Things (IoT), sensor network, smart city.**

I. Introduction

THE EVOLUTION of wireless (either local or wide area) communications and technologies, along with the processing capabilities of mobile devices, has given a significant boost to the Internet of Things (IoT) concept [1]. The main idea of the IoT is that ordinary electronic devices (sensors, smart home appliances, traffic monitoring devices, surveillance cameras, etc.) equipped with necessary transmission hardware, are able to exchange data, directly or through the Internet, in order to provide respective information on events, or act to fix or prevent undesired incidents. In addition specific devices' information could be made available to users. As a result, the IoT can enable a diversity of applications, such as smart grid, smart cities, home and industrial surveillance, home and industrial automation, e-learning, or upgrade a multitude of public services, such as traffic monitoring, healthcare, public lightning, parking and many others that would improve the quality of residents' life and moreover save world's resources.

The diversity of devices and applications that must be accommodated by the IoT concept brings forward the need for new architectures, communication protocols, models, and services to bridge the required technologies. In particular, the IoT infrastructure conveys a multitude of data types that correspond to different applications, each with specific quality of service (QoS) requirements, which must be processed at different units. Within this context many IoT standards have been proposed from major groups, such as the World Wide Web Consortium, Internet Engineering Task Force, EPCglobal, Institute of Electrical and Electronics Engineers (IEEE), and the European Telecommunications Standards Institute. In these standards, significant attention has been paid on the architecture specification, communication and application protocols, as well as service discovery protocols [1], [2].

A benefit of IoT deployment is that the generated data from embedded devices (like smart meters and sensors) can be processed to extract analytics that could profit individual citizens or business administrators. This corresponds to a vast amount of data that requires significant network, storage, and computing resources. To efficiently transmit and process the data at low cost the "fog computing" concept has emerged [1], as a bridge between the IoT end devices and the cloud services that provide sufficient computing resources for data analytics. In the fog computing paradigm, the intermediate gateway devices can perform some lightweight processing tasks and data aggregation (taking decisions distributedly), thus reducing data transfers and processing load on the clouds.

A typical fog computing architecture is illustrated in Fig. 1, comprising of: 1) numerous IoT end devices that generate the data; 2) gateways that serve as bridge interfaces to extend the connection to other networks; and 3) cloud servers that connect to a hierarchical network forming a tree infrastructure. Regarding data exchange, the IoT end devices transmit the data to the gateways, either directly or by

Manuscript received June 20, 2017; revised March 5, 2018 and May 9, 2018; accepted June 6, 2018. Date of publication June 20, 2018; date of current version November 14, 2018. This work was supported by the European Union's Horizon 2020 research and innovation programme under Grant Agreement No. 761989 in the context of the 5G-PHOS project. The authors would also like to acknowledge the contributions of their colleagues from the EU-funded FP7 VIMSEN consortium (Grant Agreement No. 619547). *(Corresponding author: Ilias Gravalos.)*

I. Gravalos and K. Christodoulopoulos are with the Computer Engineering and Informatics Department, University of Patras, 26500 Patras, Greece (e-mail: grabalos@ceid.upatras.gr).

P. Makris is with the Computer Engineering and Informatics Department, University of Patras, 26500 Patras, Greece, and also with Computer Technology Institute and Press "Diophantus," 26500 Rio, Greece.

E. A. Varvarigos was with the Computer Engineering and Informatics Department, University of Patras, 26500 Patras, Greece. He is now with the School of Electrical and Computer Engineering, Institute of Communication and Computer Systems, National Technical University of Athens, 10682 Athens, Greece.

Digital Object Identifier 10.1109/JIOT.2018.2849327

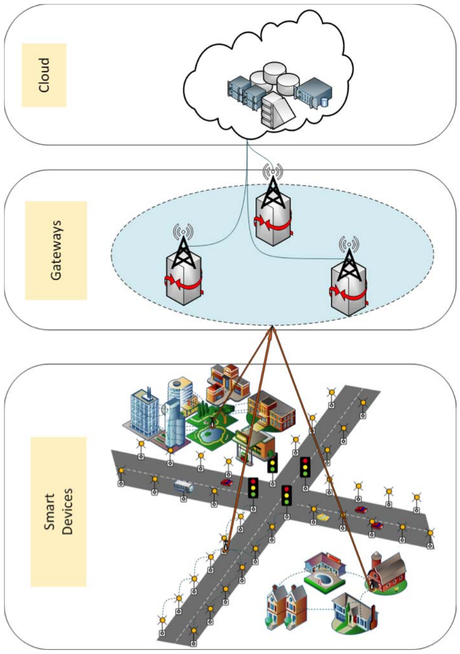


Fig. 1. Fog network architecture.

multihop transmissions through adjacent IoT end devices that act as relays (using communication protocols for MANET and WSN [3] or IEEE 802.15.4 and 802.11ah for WMN). Then, the gateways perform some processing and forward the data to the cloud servers directly through the WAN (e.g., LTE-A).

The IoT (smart) end devices can be integrated in single board computers with onboard sensors, or through standard interfaces for short range communication subsystems (like radio frequency identifiers, ultrawide bandwidth, or near field communication). This would enable data reception from nearby sensors/smart meter devices and forward it to upper layer devices or adjacent IoT end devices. Thus, IoT end devices are low cost conventional devices that operate with minimal hardware for wireless transmission using lightweight communication protocols. In this paper, we assume that the cost of the IoT devices varies according to the cost of the low-end transmission equipment (LeTE). Thus, the transmission capabilities of LeTE define the device's cost. On the other hand, gateway devices, in addition to data exchange from/to sensor/meter devices, must also provide Internet communication. In order to support both functionalities and improve mesh networking flexibility, a gateway must be equipped with multiple interfaces for wireless transmission among different access technologies. Moreover, the core hardware of a gateway must have adequate resources to support minor data processing functionalities.

A fog network infrastructure deployment for the IoT must be both bandwidth- and cost-effective. Hence, reducing the installation cost of the network would make IoT attractive for respective vendors and service providers. Since gateway devices are much more expensive than simpler IoT smart devices, a cost effective IoT network infrastructure can be achieved by efficiently selecting the gateways and IoT end devices required for the network to be established. Toward this end, we consider in this paper the problem of planning a QoS-aware and efficient IoT network. We assume a given topology of facilities (where metering and/or sensor devices are installed *a priori*), whose transmission of monitoring values is connected to specific QoS requirements, and a set of potential IoT end devices, to be deployed at these facilities, with different specifications (such as data rate, transmission range, and cost). The problem is formulated as an integer linear program (ILP) that minimizes the overall network cost, by deciding the right number and location of gateways along with suitable LeTE in IoT end devices to optimize the overall installation cost without compromising the related QoS requirements. Following this approach we obtain the optimal solution. Our simulation results indicate that the proposed solution achieves significant cost savings against the baseline solution of placing gateways in all locations. We also observe a tradeoff between cost effectiveness and QoS provisioning.

Since ILP complexity is nonpolynomial and thus intractable for medium to large topologies, we propose a heuristic algorithm, referred to as device selection adaptive to QoS algorithm (DESAQoS) to provide fast solutions. The heuristic is a k-means-based algorithm that clusters the facilities and determines suitable IoT end devices at each facility in order to ensure cost effective paths toward cluster head points (gateways) providing QoS. The computational complexity of the algorithm is analyzed theoretically, while we observed that it finds solutions near the optimal ones in the experiments that we were able to track the optimal solution. The results show that the proposed heuristic achieves solutions close to the respective ones obtained by the proposed ILP, which are considered optimal for the problem as formulated in this paper.

The remainder of this paper is organized as follows. In Section II, we refer to the previous studies and comment on related work with respect to our work's novelty points. In Section III, we provide a detailed description of the assumed network model and formulate the ILP problem. The respective heuristic approach is presented in Section IV. Obtained solutions of the proposed approaches over random topologies are presented in Section IV. Finally, Section V concludes this paper.

II. Related Work

Irrespectively of the underlying topology and protocols (WSN, WMN, M2M, or IoT), gateways placement in wireless networks as well as clustering and coverage problems in wireless ad-hoc networks, have long received considerable attention [4]–[6]. In the simpler approach, the clustering problem refers to finding the minimum number of cluster head nodes that collect and aggregate the data of the nodes in the cluster. This approach is applied in base station scenarios, where only one hop communications is involved. When multihop paths are used, the establishment of hierarchical networks with data aggregators is preferred for efficiency purposes.

Chandra *et al.* [7] explored the placement of gateways, which they call Internet transit access points (ITAPs), in wireless neighborhood networks and sensor networks by accounting for link capacity, wireless interference, and variable traffic constraints. They study multihop networks, in which house or sensor nodes send or forward data to servers on the Internet via ITAPs. They propose ILP formulations and placement algorithms to obtain the minimum number of ITAPs for a given topology under three different wireless link models (ideal link and two variants of a general link model). The link model is associated with the throughput on the links, which bounds the path length. However, they assume that every node is equipped with identical transmission devices and thus all wireless links on the network are the same. Furthermore, the set of potential ITAP locations are points in the plane that can be reached by a set of nodes via a wireless link, but not including nodes' locations as candidate ITAP locations, which does not always provide the minimum network installation cost.

Efficient gateway placement with QoS constraints in WMN has been studied in [8]–[11]. Li *et al.* [8], also, studied the optimal placement of a given number of gateways on a wireless mesh backbone network in order to achieve maximum throughput. They formulate the related problem as an ILP and provide a greedy algorithm that selects the gateways' locations in order to optimize the cross-layer throughput. They take into account the capacity reduction on wireless links due to interference in case of simultaneous transmissions. The aim of the algorithm is to provide interference-free link scheduling. In their study the installation cost is predefined since the number of gateways is given *a priori*. Aoun *et al.* [9] considered the capacity of every link as a QoS constraint. Given the link capacities and an inequality that correlates the total one-hop capacity of the network with the expected path length, the authors bound the maximum number of hops from a node to the gateway in order to achieve QoS. Another polynomial time heuristic algorithm for gateway placement considering QoS, was proposed in [10]. This algorithm performs an initial clustering, where nodes are one hop away from the selected cluster head nodes. Then split, merge and shift functions are performed among the clusters in order to reduce their number and hence the number of gateways as long as QoS are satisfied. The installation cost is not taken into account in these studies. He *et al.* [11] considered the installation cost of the WMN, formulating the problem as an ILP. The authors also propose two variants of a heuristic that attempts to minimize the number of gateways (reducing the installation cost), while maximizing the gateway to Internet throughput and minimizing the path length from nodes to gateways. The heuristics are based on greedy dominating tree set partitioning to obtain the cost effective trees with gateways at the root. However, the aforementioned studies do not consider differences in the nodes, assuming that every node is equipped with identical specification devices and thus all wireless links on the network are the same and of equal cost.

In several other studies [12]–[14], optimization methods, such as genetic algorithm, simulation annealing, and tabu search meta-heuristics have been applied to minimize the number of gateways in wireless networks without, however, considering the connection polymorphism of real wireless networks due to transceivers with different characteristics and the overall installation cost of the network. In a recent study [15], the mobility of wireless mesh network client devices is also considered, regarding placement of gateways in dynamic WMNs. The authors propose a social-based swarm optimization method that exploits the social relationship notion of users, in which groups with similar interests move with high probability to the same direction.

In this paper, our aim is to reduce the number of gateways and place appropriate LeTE in IoT devices to lower the overall installation cost of the IoT network. Furthermore, in contrast to the previous works, we consider different IoT (nongateway) communication devices to be deployed at the facilities' locations. These devices vary in their transmission capabilities and consequently in their cost. Each device is selected with respect to: 1) its transmission specifications, in order to ensure sufficient capacity for its own and transient flows and 2) its cost, so as achieve an overall lower network cost. This paper is an extension of our previous study [16]. In particular, we include a novel heuristic algorithm to address the planning problem and we also provide a more complete set of performance results that in addition to considering a wider set of parameters, they also evaluate the performance of the proposed heuristic.

III. System Model and Gateway Placement Formulation

In our problem setting, we consider a set of nodes (representing facilities) placed at specific locations. Each node represents a point that generates corresponding metering data and utilizes a respective IoT end device. The IoT end device is capable of transmitting/receiving data to/from the Internet through gateways. This communication, which is associated with specific QoS requirements, can be achieved either directly or through multihop transmissions. In the first case, the IoT end device must be equipped with suitable hardware and software to function as a gateway. In the latter case, it is preferred for the IoT end device to utilize simpler and lower cost transmission equipment, in order to transmit its data to nearby devices until a gateway is reached. The simple transmission equipment can be selected from a set of devices with different transmission specifications and (consequently) cost. The LeTE defines the type of IoT end device to be utilized. Henceforth, the IoT devices and LeTE definitions will refer to the same object. The permissible locations for the placement of the gateways must be specified as well. The candidate locations of a gateway include the positions of the IoT end devices and some intermediate points that are discovered through Voronoi diagrams (the discovery phase is explained in Section III-B).

In summary, we assume two classes of devices: 1) gateways and 2) a set of IoT end devices (lower-end transmission devices with different capabilities). IoT devices are placed at the given locations, while gateways can be placed at these locations (replacing the IoT related device there) or at the Voronoi points. Within this context, we pursue to optimize the placement of the gateways and the selection of suitable IoT transmission devices in order to minimize the installation cost of the network, while respecting predefined QoS requirements. An illustrative example of the proposed gateway placement problem is depicted in Fig. 2. The five nodes in Fig. 2(a) must be equipped with suitable devices to form the IoT network that

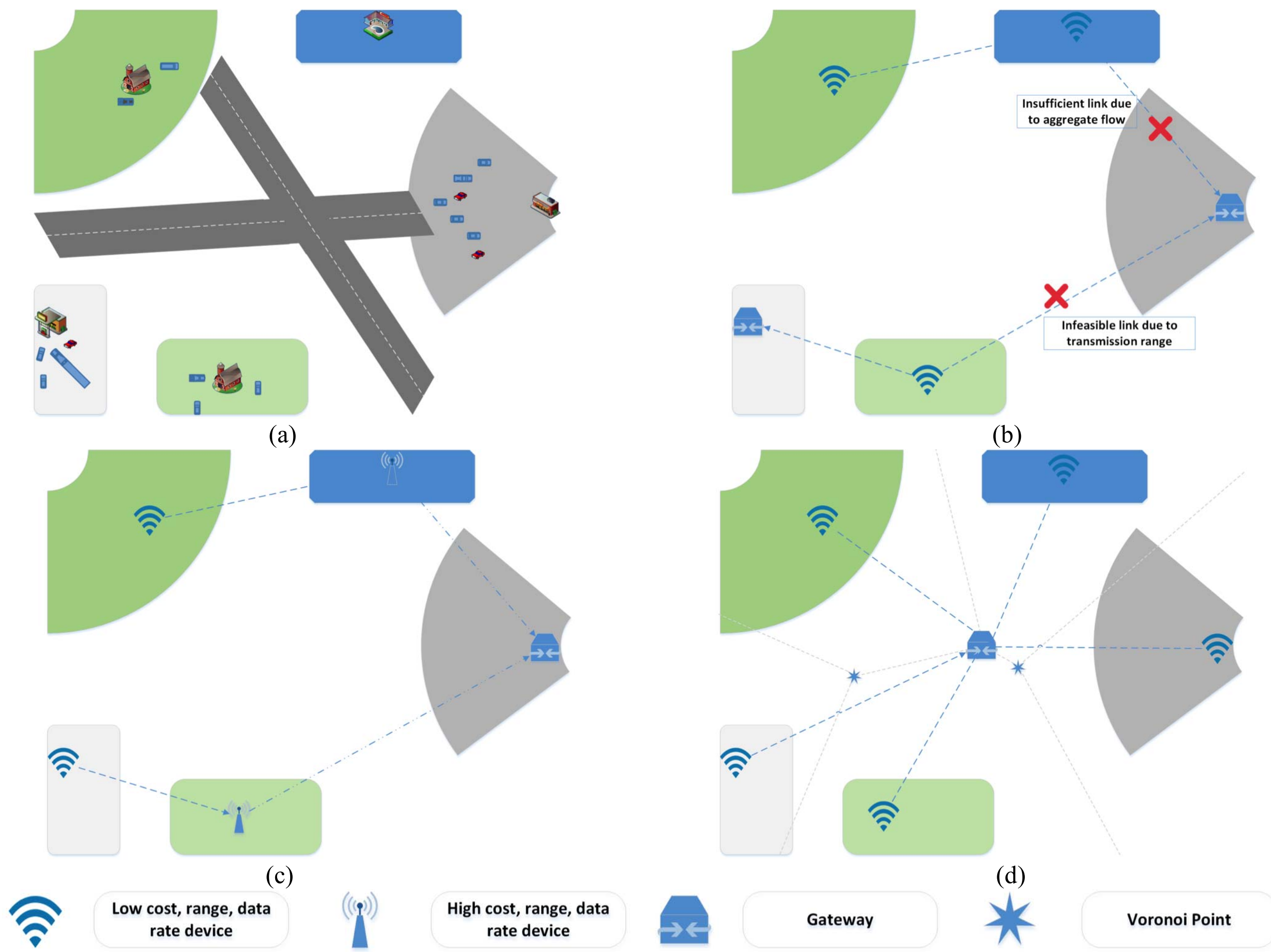


Fig. 2. Instances of the proposed model. (a) Typical district with IoT wireless devices, (b) infeasible network setup due to both insufficient capacity (upper link) and transmission range (lower link), (c) proposed network setup, and (d) also utilizing Voronoi point.

is able to communicate with the world, through gateways. Due to the procedure, several deployments [Fig. 2(b) and (c)] are considered until the one that is feasible and cost efficient is found. In Fig. 2(d), the advantage of intermediate gateway candidate points is highlighted.

A. Low-End Transmission Equipment Specifications

To optimize an IoT network, we assume a set D of different transmission equipment that can be placed in specific locations to forward the data to some gateway points. Each LeTE $d \in D$:

1) operates at a specific effective data rate R_d bits/s;
2) reaches a fixed transmission range B_d based on the device technology;
3) has a specific cost price P_d by the vendor.

The devices' effective data rate, R_d, is defined so as to account for the MAC overhead and interference in the network. To be more specific, we assume that the MAC protocol utilized to resolve interference has a specific efficiency H_d, and thus is taken as a portion of the nominal data rate of the device $R_{d,\text{nominal}}$ provided by the vendor, that is $R_d = H_d{\cdot}R_{d,\text{nominal}}$.

MAC protocols provide techniques to synchronize the transmission of nodes in wireless networks and avoid interference. The idea is to coordinate so that nodes whose transmission range overlap should exclusively transmit on the shared medium. Following this approach, Gupta and Kumar [17] proved that the maximum throughput per node in a random network is upper bounded by $\Theta\ (R_{d,\text{nominal}}/\sqrt{\xi})$, where ξ is the number of interfering nodes.

We will adopt a similar approach in our model, in order to estimate the value of H_d. To be more specific, we will assume that the efficiency of each LeTE device depends on the number of devices ξ_d within its transmission range B_d, and we will approximate H_d by $\sqrt{(1/\xi_d)}$.

Alternatively, to allow simultaneous transmissions to different receivers without interference, FDM/TDM channel distribution between interfering transmitters that communicate with independent recipients could be assumed. The decisions on the specific channels/slots for these transmissions can be obtained by typical channel allocation algorithms, the discussion of which is outside of this paper.

B. Gateway Candidate Location Points Discovery

A gateway could be located at any place of the IoT area so that all or a portion of the IoT end devices could reach it

TABLE I
SUMMARY OF PARAMETERS

Parameter	Description
	Constants
$V_L \subseteq V_G$	The set of facility locations
F_i	The arrival rate at facility i in bits/sec
R_d	The service rate for LeTE device $d \in D$
P_V	The price of a gateway and p_d of a device
S_{nm}	The distance between nodes n, m
M	A big number (in bits/sec)
	Variables
y_i^d	Boolean variable ($\{0,1\}$) that indicates whether node i deploys device d or not
z_v	Boolean variable ($\{0,1\}$) that indicates whether a gateway is deployed at position $n \in V_G$ or not
l_{nm}^{iv}	Boolean variable ($\{0,1\}$) that indicates whether link (n, m) belongs to the path from node i to gateway v
f_n	Float variable equal to the load (the sum of traffic) on bits/ sec that is generated or cross the LeTE at location $n \in V_L$
d_{nm}^{iv}	Positive float variable equal to the queuing delay introduced at the LeTE at location $n \in V_L$

either directly or over a multihop path. To narrow the options, we consider as an initial set of candidate positions for the gateways, the positions of the facilities. Assuming that every end device on a facility has at least one neighbor it can directly communicate with utilizing some of the available LeTE devices, the initial set of facilities could be sufficient. However, there may be cases where a gateway placed at the centroid of a specific subarea would be more efficient for data aggregation. To also cover these cases, we extend the aforementioned set by the Voronoi vertices (intersection of Voronoi edges) of the Voronoi diagram [18] that is generated by the given facility locations. In particular, we construct a Voronoi diagram with the facilities being the site points The Voronoi vertices are the intersection points of three or more Voronoi edges thus making them equidistant points of corresponding three or more IoT facility locations. The final set V_G of candidate gateway locations is derived by excluding any Voronoi vertices whose distance from the neighboring site points is larger than the largest B_d, meaning that a physical connection from any facility is infeasible.

C. Optimal Gateway and Transmission Device Placement Formulation

Given the set of facility locations V_L along with their QoS requirements, the complete set of feasible gateway locations V_G, and the set of LeTE device we want to select the gateways and the LeTE devices so as to the minimize the implementation cost of the network. The problem is a variation of the single source uncapacitated facility location problem (SSUFLP) [19] with QoS constraints. To optimally solve this problem in the following we present an ILP formulation. The parameters and variables of the ILP are summarized in Table I. The objective and the constraints of the ILP are as follows:

$$\text{minimize} \sum_{i\in V_L}\sum_{d\in D} P_d y_i^d + \sum_{v\in V_G} P_V z_v \tag{1}$$

subject to
For all $i \in V_L$,

$$\sum_{v\in V_G} \sum_{\substack{m \neq i \in V_L \\ m \neq v}} l_{im}^{iv} + \sum_{v\in V_G} l_{iv}^{iv} = 1. \tag{C1}$$

For all $i \in V_L$,

$$\sum_{v\in V_G}\sum_{n\in V_L} l_{nv}^{iv} = 1. \tag{C2}$$

For all $i, m \neq i \in V_L, m \neq v \in V_G$,

$$\sum_{n\in V_L} l_{nm}^{iv} = \sum_{\substack{k \in V_L \\ k \neq v}} l_{mk}^{iv} + l_{mv}^{iv}. \tag{C3}$$

For all $i \in V_L$,

$$\sum_{d\in D} y_i^d \leq 1. \tag{C4}$$

For all $i, n, m \neq v \in V_L, v \in V_G$,

$$S_{nm} \cdot l_{nm}^{iv} \leq \sum_{d\in D} B_d \cdot y_n^d. \tag{C5}$$

For all $i, n \in V_L, v \in V_G$,

$$S_{nv} \cdot l_{nv}^{iv} \leq \sum_{d\in D} B_d \cdot y_n^d. \tag{C6}$$

For all $v \in V_G, i, n \in V_L$,

$$l_{nv}^{iv} \leq z_v. \tag{C7}$$

For all $v \in V_G, i, n, m \neq v \in V_L$,

$$l_{nm}^{iv} \leq z_v. \tag{C8}$$

As (1) implies, we aim to minimize the total cost of the IoT network infrastructure deployment. In order to assure that we obtain a feasible network, we apply suitable constraints. Constraint (C1) ensures that the corresponding traffic, originating from facility i, will be transmitted through a single outgoing link, and, as (C2), (C7), and (C8) ensure, will be destined to a single gateway. Moreover, (C3) ensures that no traffic will be lost from intermediate nodes. Each facility is constrained to deploy a communication device, as (C4) imposes, which combined with (C5) and (C6), implies that the device can be either a gateway or one LeTE from the set D. Furthermore, the latter constraints ensure that the communication between two facilities can be achieved if the receiver is inside the range of the transceiver in means of Euclidian distance.

The latency introduced to the flows should be limited according to specific QoS requirements. To formally define this, we can assume Poisson flows, exponential packet lengths, and Kleinrock's independence approximation, and so the queue of each LeTE is described by the M/M/1 queuing model. In constraint (C9), we first calculate the load of each

LeTE. Then, in constraint (C10), we assume an average packet of length X bits and we calculate the average queuing delay introduced at each location. In case the transceiver is a gateway, then a big rate $M>>1$ is used to make the delay equal to zero. Finally, the average accumulated latency of each flow $i \in V_L$ is constrained in (C11) to satisfy the QoS latency limit K_i, taking also into account the propagation delay (C denotes the speed of light). Note that the case, where the packet lengths are deterministic (M/D/1 queue), can be formulated similarly, since the queue length and the delay in an M/D/1 queue is half of that of the M/M/1 queue that is studied in the following.

For all $n \in V_L$,

$$f_n = \sum_{i \in V_L} \sum_{v \in V_G} \sum_{m \in V_G} F_i \cdot l_{nm}^{iv}. \tag{C9}$$

For all $n \in V_L$,

$$d_n = \frac{X}{\sum_{d \in D} R_d \cdot y_n^d + M \cdot z_n - f_n}. \tag{C10}$$

For all $i \in V_L, v \in V_G$,

$$\sum_{n \in V_L} \sum_{m \in V_L} \left(d_n + \frac{S_{nm}}{C} \right) \cdot l_{nm}^{iv} \leq K_i. \tag{C11}$$

Constraint (C10) essentially is based on an M/M/1 approximation of the queue delay, assuming Poisson traffic, and exponential packet lengths. Since constraint (C10) is nonlinear, in the ILP formulation we do not introduce constraints (C10) and (C11), but we simplify them, introducing the following two linear constraints (C12) and (C13). To be more specific, (C12) constrains the load at each facility to be less than $\varepsilon \in (0, 1)$ of the transmission rate of the selected LeTE (gateways are assumed of infinite rate, by setting $M>>1$).

For all $n \in V_L$,

$$f_n \leq \varepsilon \cdot \left(\sum_{d \in D} R_d \cdot y_n^d + M \cdot z_n \right) \tag{C12}$$

where we remind that the rates are measured in bits/s.

Constant $\varepsilon \in (0, 1)$ is chosen sufficiently far from 1 (e.g., 0.8) to guarantee that the link is operating sufficiently far from the total link capacity (e.g., at 80%) so that the queuing delays (due to the randomness of the arrivals and of the packet lengths) are relatively small.

Then (C13) calculates the queuing delay d_{nm}^{iv} (which is by definition a positive float) that each flow (*i, v*) experiences in a link *(n, m)*, which under light load $f_n < \varepsilon \cdot R_d$ equals the transmission delay

$$d_{nm}^{iv} \geq -M + M \cdot l_{nm}^{iv} - M \cdot z_n + \sum_{d \in D} \frac{X}{R_d} \cdot y_n^d + \frac{S_{nm}}{C}. \tag{C13}$$

Alternatively, one could think of the right hand side of (C13) as modeling the delay of a D/D/1 (as opposed to an M/M/1 or M/D/1) queue, where aggregated arrivals f_n are deterministic with a constant bit rate (CBR) of rate $f_n < \varepsilon \cdot R_d$ (where R_d is the capacity of the link) and packets are of constant length equal to X bits. For CBR generated traffic and constant packet lengths, queuing delays are 0 as long as the arrival rate is less than the service rate even for ε very close to 1.

Constraints (C12) and (C13) can be used to replace (C10), and are applied together with (C9) and (C11) to satisfy the QoS latency limit K_i as follows.

For all $i \in V_L, v \in V_G$,

$$\sum_{n \in V_L} \sum_{m \in V_L} d_{nm}^{iv} \leq K_i. \tag{C14}$$

IV. Heuristic Algorithm

The optimal solution for the placement problem can be obtained by solving the aforementioned ILP formulation, using an ILP solver. However, the complexity of the problem is exponential (variation of the SSUFLP [19]) and the running time of such a solver is prohibited for larger topologies. Thus, we propose a heuristic algorithm to plan the IoT network infrastructure in a cost effective manner, to be referred to as the DESAQoS. The heuristic approximates very well the optimal solutions obtained by the ILP in significantly less time and with less computational resources. Previously proposed gateway placement heuristic algorithms [9]–[11] cannot be applied to this problem since they assume a predefined connected topology from which respective spanning trees and acyclic graphs can be obtained. In our system model, the type IoT devices to be installed at each facility are variables to be jointly decided by the planning problem, and hence the connectivity among facilities is not known *a priori*. A high-level presentation of the DESAQoS is illustrated in Algorithm 1. The algorithm consists of different phases that are executed iteratively until the solution is achieved. The phases are described in the following.

A. Initialization Phase

Given the locations of the facilities V_L and the set D of different LeTE, the algorithm obtains y different connected sets for the topology. In a connected set S_q the facilities that comprise it can communicate with each other, either directly or via multihop paths. This means that for each facility $i \in S_q$ there is a device $d \in D$ with transmission range enough to communicate with at least one other facility $j \in S_q$, and adequate data rate to serve the corresponding facility's flow. However, there must be no link from S_q to any other set. These properties are summarized in the following:

$$\bigcup_{q=1}^{y} S_q = V_L, \ \bigcap_{q=1}^{y} S_q = \emptyset.$$

Such distinct sets would result in cases of sparse topologies with high density locally (a municipality composed of many villages would be an example). Subsequently this phase would advance the complexity of the clustering phase, which will be applied for each small set at a time, rather in the aggregated set.

The procedure begins for the most inexpensive LeTE $d_1 \in D$ and from an arbitrary facility $i \in V_L$. The facilities that are in the transmission range of i for device d_1 expand the set S_1. The same procedure is executed for each newly added facility

Algorithm 1 DESAQoS

```
input: V_L, V_G, D
output: deviceAtFacility

/* Initialization Phase */
undiscoveredNodes ← V_L
intermediates ← Voronoi()
while undiscoveredNodes ≠ Ø
  S_q ← undiscoveredNodes[rand()]
  for d ∈ D
    for i ∈ S_q
      for n ∈ undiscoveredNodes
        if n in transmission Range of i
          S_q ∪ {n}
          undiscoveredNodes - n
  q = q + 1
for g ∈ intermediates
  for each S_q
    if |S_q|== 1
      if g in transmission Range of i ∈ S_q
        S_g ∪ {i}
/* Clustering and Path Discovery Phase */
for each S_q
  clusterNum = 1
  while latestCost < previousCost
    v, C ← k-means(S_q, clusterNum)
    for each C_λ ∈ C
      for each i ∈ C_λ
        setPath(i, v)
  clusterNum += 1
```

in order to further expand the set with remaining undiscovered facilities. The procedure is repeated from the beginning for each LeTE $d \in D$ until the set cannot be extended any further, in which case the procedure is executed from another undiscovered facility j that would create a new set S_2, and the S_3 and so on, until all facilities in V_L are discovered. After exploiting the connectivity between facilities, another round considering the Voronoi points is initiated, for the facilities that are not interconnected (every set of unary length). The procedure is the same, with each facility tested for a direct link to one of the intermediate points. In order to form a new set, the following constraint must be met:

$$\left|S_q\right| p_v \geq \sum_{n \in S_q} p_d^n + p_v$$

which imposes that the set is most cost effective than the trivial with all facilities deploying a gateway. From the newly created sets we keep the ones that cover all facilities and have the less aggregate cost. At this phase the device deployment cannot capture QoS constraints, but it is a promising indicator. We include equality as well, to favor improved scalability (newly added facilities can exploit the existing topology and utilize a marginal cost efficient cluster).

At the end of the phase there are y distinct sets with a low-end transmission device being assigned at each facility. At the current state the devices are assigned with connectivity as the only criterion. Thus, there is a network in which each device can reach and be reached by another and in the following phases the gateways and the paths are defined, ensuring QoS.

Algorithm 2 setPath

```
paths ← K-Shortest Paths(i, v)
pathFound ← From paths select the path
 that satisfies (C12),(C14) and optimizes
 Eqs (2), (3)
if pathFound = Ø
  for p ∈ paths
    Try new devices d ∈ D at bottleneck
    facilities to satisfy Eqs. (C5), (C6),
    (C12), (C14) for all paths traversing
    the facilities.

    if p adapted successfully
    adpPaths ∪ p

  pathFound ← From paths select the path
  that optimizes Eqs (2), (3)
  if pathFound = Ø
    break
    Increase #clusters and execute from
    the beggining

Store the selected devices
Refresh the available bandwidth for the
facilities in the path subtracting F_i.
```

B. Clustering Phase

For each set S_q, obtained in the previous phase, the algorithm separately applies the procedure of gateway and LeTE placement. Thus, for the set S_q, this operation begins by applying the k-means clustering algorithm [20] to obtain a centroid point in which the gateway could be placed. At this point the set S_q is assumed to be a single cluster C_λ and the k-means algorithm attempts to minimize the Euclidian distance between facilities in S_q and centroids, thus providing an intermediate point in the area of the set. In order to ensure that the gateway can be reached by at least one facility in S_q, the algorithm defines as the gateway location, the point $v \in V_G$ (it is straightforward that $v \in S_q$) that is closest to the point obtained by k-means. Here, let us notice that a single gateway for the set could not be adequate due to QoS constraints of the comprised facilities. This will be determined in the path discovery phase described below. In such case, the k-means algorithm is applied more times to generate new clusters along with the respective centroids. At each iteration, the number of clusters (and thus gateway points) is increased by one until the QoS are satisfied.

C. Path Discovery Phase

In order to estimate the final most cost-effective network, the LeTE devices must be decided for each facility in order to ensure a path to the gateway while satisfying the respective QoS constraints [(C12)–(C14)]. To this end, the algorithm investigates and defines the paths from each facility to the gateway of the corresponding cluster. Here we mention that each cluster is a list of facilities sorted in increasing order to the distance from the gateway point. Therefore, for each facility $i \in C_\lambda$ (starting from the closest one to the gateway) the algorithm generates the available paths to the gateway $v \in C_\lambda$. In order to maintain low complexity, each newly added facility n (as long as it can satisfy the flow constraint of i) extends the path with facilities that can successfully transmit to, and are toward the gateway. Then for each of them, facility n generates the corresponding different candidate extensions. The path that satisfies the QoS constraints and optimizes (2), which ensures that it is the most cost effective, is selected. If multiple paths achieve the same cost in (2), the one that optimizes (3), which ensures that the path is the most bandwidth effective, is selected as the path from i to v

$$\min \frac{\sum_{n \in \text{path}(i,v)} p_n}{|\text{path}(i,v)| - 1} \tag{2}$$

$$\max \frac{\sum_{n \in \text{path}(i,v)} r_n}{|\text{path}(i,v)| - 1} \tag{3}$$

where r_n is the available bandwidth left at LeTE n.

The algorithm generates all different paths from i for every device $d \in D$, starting from the one with the less price. If a path for a device is found then the iteration terminates for i and continues to the next facility in the cluster. In case there is no path that satisfies the QoS constraints for facility i, then adaptation operations are performed for every candidate path generated previously. In the following, we present two adaptation methods that are applied depending on the problem at hand (i.e., depending on the QoS requirement that is not met).

1) Adapt Path to Flow Constraint: If a path comprises of one or more links that fail to meet the constraint of (C12) (which means that the remaining data rate at corresponding devices is not enough to serve the flow of facility i), while it satisfies (C14), the algorithm tries to replace the devices with one that offers adequate data rate to serve the total flows traverse the relay facility. If there are multiple such devices, the less expensive one is selected. Furthermore, the new devices must not compromise the connectivity of previous installed paths that use these facilities as relay nodes. If such a device exists then the path becomes a candidate path for the communication from i to v.

2) Adapt Path to Delay Constraint: In case the path imposes delays that violate (C14), then the algorithm iteratively changes the devices on the facilities along the path with other devices with longest transmission range and/or higher data rates and re-estimates new paths toward the gateway that satisfy (C14). The notion behind this adaptation is that higher data rate will reduce the delay in a straightforward manner. Moreover, larger transmission ranges may generate paths that reduce the length of the route, since paths are extended toward the gateway. Hence, the connection between end points tends to be a straight line. The candidate devices to replace the existing ones must also operate at adequate data rates to transmit the aggregate flows that traverses the respective facilities.

Among the paths generated by adaptation procedures, the one optimizing (2), (3) is selected. In case the adaptation operations fail to discover a suitable path from facility i to gateway v, path discovery phase is terminated and the algorithm returns to clustering phase in order to produce more gateways (and clusters) to be deployed at the corresponding set. The number of clusters is increased by one at each iteration. The procedure for a set is repeated until the most cost effective solution satisfying the QoS constraints is found.

D. Complexity Analysis

In the initialization phase of the algorithm each facility is checked for connection with the facilities in V_L that have not been discovered yet, until all facilities are discovered. This would require $|V_L|$ iterations. In the worst case, where each facility becomes a single-element set, the procedure must be executed for every device resulting into an arithmetic series from $|V_G| - 1$ to 2 facilities. Thus, the worst case running time would be $|D|(|V_G|/2)(|V_G| + 2)$. This initialization phase removes the independent facilities from the facilities that are examined in subsequent phases, so in a sense it is a separate procedure, and its complexity is added to the one reported next.

The clustering phase operation is dominated by the k-means algorithm. In general k-means optimal solution is proven to be an NP-hard problem [21] and its computational complexity is exponential (or super-polynomial) [22]. However, many heuristics have been proposed to provide local optimum solutions in near linear-time in the number of points [20], [23], [24]. To be more specific in our implementation we used the algorithm presented in [20].

In the path discovery phase, the algorithm attempts to discover the cost effective path for each facility. For the worst case we assume a single cluster that contains all the facilities in V_L (this means that the initialization phase has not excluded any facility due to not satisfying a connectivity or flow constraint). During each path discovery, each facility added to the path has to check for the next hop facility among the ones that are not already in the path. Since, the aforementioned topology will be dense, discovering every path from a facility to the gateway would result into exponential number of paths. To mitigate the complexity we implement a K-shortest loopless path approach [25]. Thus, we select the K-shortest paths, in means of number of hops that optimize (3). Having a multitude of paths is necessary in case that there is no feasible path and adaptation must be applied. Thus, the time complexity is $O(K|V_L|(|E_L| + |V_L| \log |V_L|)$ for each facility resulting into $O(K|V_L|^2(|E_L| + |V_L| \log |V_L|)$ complexity considering all $|V_L|$ facilities. $|E_L|$ is the number of edges, which in the worst case (very dense network) can reach up to $|V_L|^2$. Clearly all phases of the proposed algorithm have polynomial execution time.

V. Performance Results

We conducted a number of simulation experiments to evaluate the effectiveness of the proposed schemes, under a variety of network planning scenarios. We considered two kinds of topologies: 1) random topologies, with different numbers of

TABLE II
DEVICE SPECIFICATIONS

Communication Devices	Data Rate	Transmission Range	Cost (Euros)
Device1	2 Mb/s	70 m (no antenna)	14
Device2	1 Mb/s	200 m (with antenna)	21
Device3	250 kb/s	330 m (with antenna)	21
Gateway	LTE-A		55

facilities randomly placed in a plane according to a 2-D uniform distribution (for each reported simulation experiment we generated ten different random topologies with the same number of facilities and their results were averaged for the presented value) and 2) mesh-like topologies (such structured topologies can represent a smart city or highway scenario with lampposts as facilities equipped with sensors, traffic cameras, etc). In what follows we will present some randomly selected instances of the topologies in a 1000 m $\times$ 1000 m plane, in order to explain the details of the devices' selection, as well as an overall performance comparison between the ILP solution and the heuristic. The candidate communication equipment to be utilized for the fog network correspond to custom devices assembled with commercial hardware and software. A gateway is assumed to utilize a Raspberry Pi 2 chip (RPi 2) and multiple antennas, while IoT devices rely on Arduino Uno R3 board with single transceiver. Hence, the cost of each device is the sum of the respective vendors' cost for each component. The data rate and transmission range values were found in the literature and were obtained from real field measurements of the distance at which a device can transmit with zero packet loss [26]. The measurements obtained for each device individually and thus the MAC protocol efficiency H, taken equal to 85% was applied to define the useful data rate of the devices (as discussed in Section III-A). Table II summarizes the specifications of the devices under consideration.

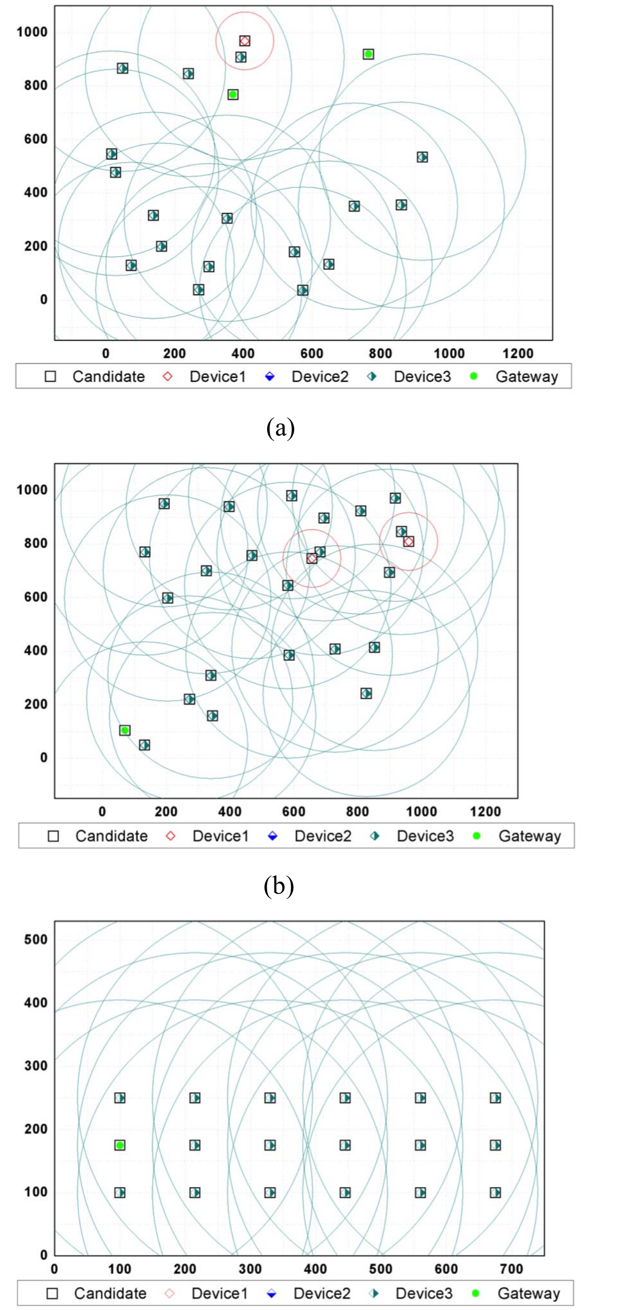


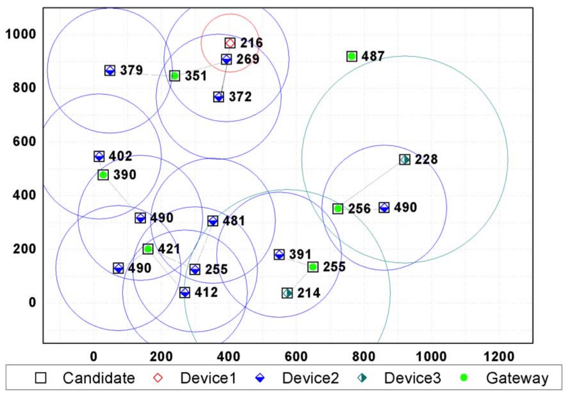
Fig. 3. Network planning for (a) 20 facilities, (b) 25 facilities, and (c) mesh-like 3 $\times$ 6 facilities topology.

A. ILP Without QoS Constraints Scenario

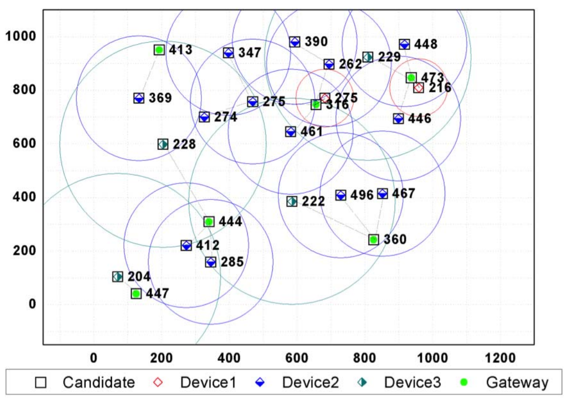
Fig. 3 depicts the obtained networks for 20 and 25 facilities randomly placed at the field (type a) and a mesh-like 3 $\times$ 6 topology (type b). The solutions were obtained using CPLEX [27], [28] to solve the formed ILP problems for the case where no flow constraints applied [only constraints (C1)–(C8) where used]. The results showcase that the proposed formulation can optimally design the desired network placing the gateways and LeTE devices so that the facilities are assigned to the corresponding gateways either with a single or a multihop transmission. Furthermore, they show that as the topology becomes denser the network becomes more efficient. Utilization of transmission devices instead of gateways results into savings of up to 60%, compared with the case where a gateway is deployed at each facility.

B. ILP With QoS Constraints Scenario

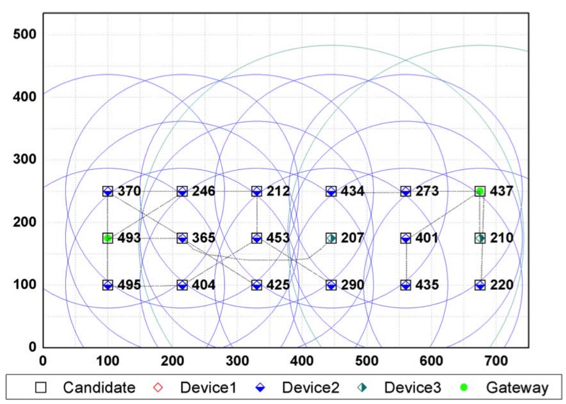
For the above scenarios, the only significant characteristic for the configuration of the network is the transmission range. We also consider the performance obtained through our formulation when flow constraints are also applied. In this case, each facility is assigned a predefined flow uniformly selected between 200 and 500 Kb/s. The range of the load is dependent on the data rates of the devices so as not only to exclude any of them but also highlight the differences on selection between them. Furthermore, suitable delay constraints have



(a)

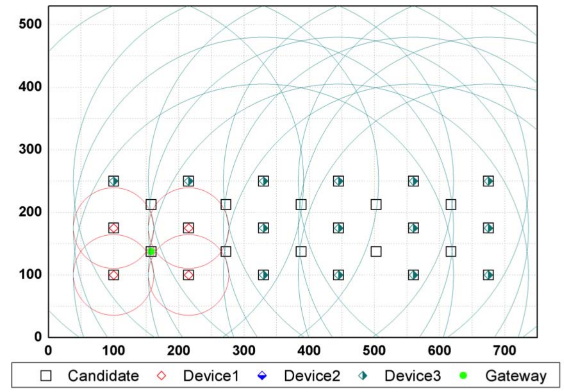


(b)

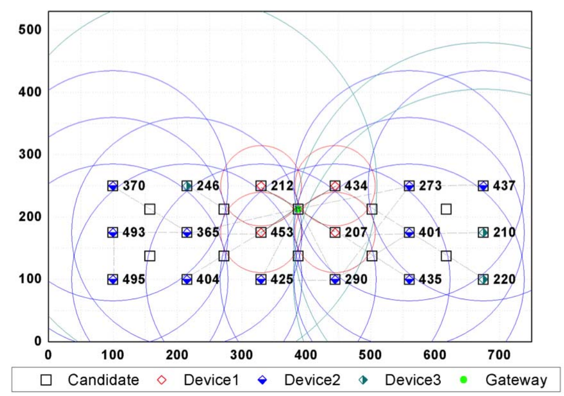


(c)

Fig. 4. Network planning for (a) 20 facilities, (b) 25 facilities, and (c) 3 × 6 mesh-like topology considering flow constraints.



(a)



(b)

Fig. 5. Network planning for 3 × 6 topology considering Voronoi points (a) without and (b) with QoS constraints.

been imposed, randomly selected in the range of 10 s to 30 min (in compliance with the services' tolerable delays defined in [2]). This pair of values consist the respective QoS constraints for each facility. Fig. 4 shows that our approach can capture such QoS restrictions and the network configuration adapts to the given constraints. Here, the difference of device class 2 and class 3 becomes obvious since the respective data rate is taken into account. Moreover, since the data rate of each device is limited, the links that are close to the gateway in Fig. 3(b) and (c) become congested and thus more gateways must be utilized as Fig. 4 depicts. The QoS constraints bound the length of the paths toward the gateway, thus limiting the number of simultaneous transmissions from links adjacent to a gateway.

C. ILP Considering Intermediate Points Scenario

Previous scenarios consider solely the locations of the facilities as the set of the candidate gateway points. In cases where intermediate points can be utilized for gateways placement, Fig. 5 shows that the use of Voronoi points as candidate positions can be more efficient and further decrease the overall installation cost. As can be seen, the cost savings may be more significant when QoS constraints are applied, where more gateways should be utilized to satisfy the network requirements, according to the basic approach. In this case, Voronoi points provide additional alternatives to position gateways, which may result into network configuration with less gateways utilizing more low cost LeTE devices. In general, considering only the installation cost, placing a gateway at a Voronoi point will add extra cost since otherwise there would be no device in that point. To benefit from such an approach this additional cost should be counterbalanced by the replacement of nearby IoT end devices with other lower cost devices. Thus, savings can be achieved either due to the difference in the cost of LeTE or in the number of the facilities that replace the LeTE.

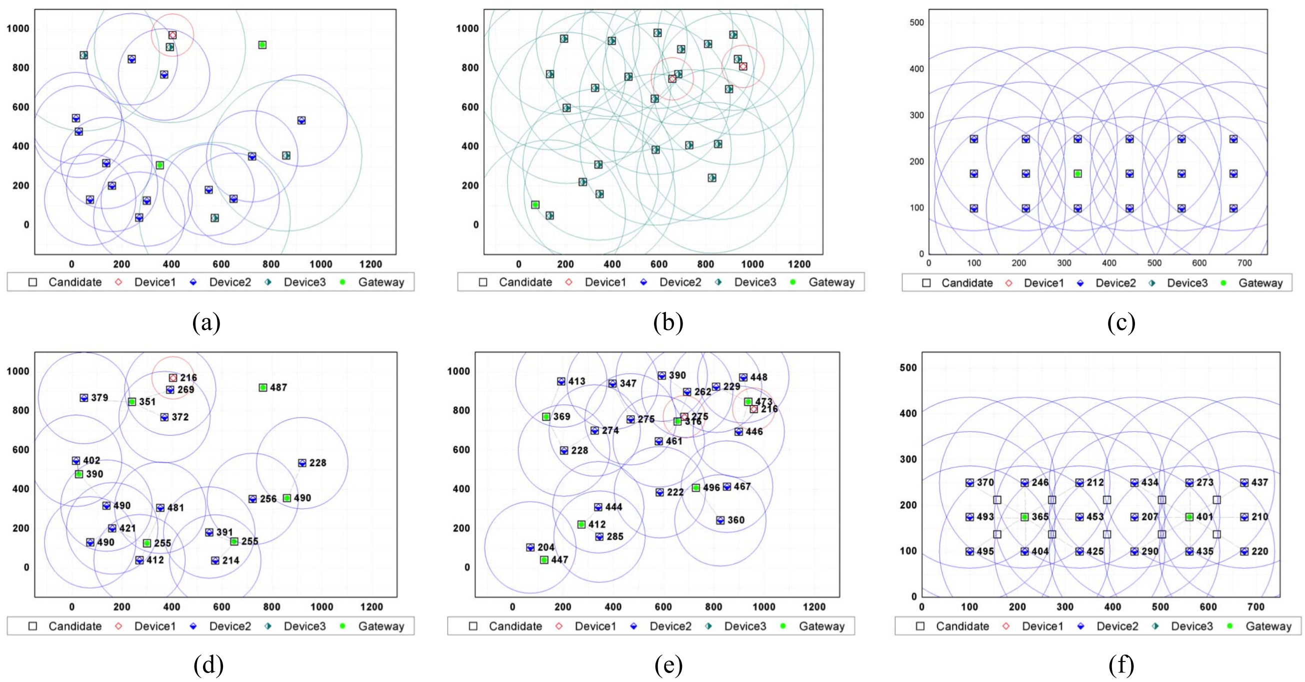


Fig. 6. DESAQoS results for (a) 20 facilities, (b) 25 facilities, (c) 3 × 6 mesh, (d) 20 facilities with QoS, (e) 3 × 6 mesh with QoS, and (f) 3 × 6 mesh with QoS and Voronoi points.

D. DESAQoS Performance Evaluation

Corresponding simulations were conducted to assess the performance of the proposed heuristic. Fig. 6 shows the established network of DESAQoS for the same topologies of the previous scenarios. In particular Fig. 6(a) to (c) correspond to Fig. 3, where no QoS constraints were applied. Fig. 6(d) and (e) presents QoS scenarios (as in Fig. 4). Comparing these instances, it is evident that the proposed heuristic yields solutions near to optimal ones (as obtained by the ILP). Especially in the cases where the QoS constraints are not considered the overall installation cost is exactly the same for both approaches. The number of gateways and the LeTE selected for each facility are proportional. The difference is at the positions selected for gateway deployment. The DESAQoS always selects centroid points due to its clustering nature. This fact is of insignificant importance when QoS constraints are not taken into account. However, when QoS must be met the inner points of the topology are more efficient and thus are preferred by the ILP as well.

Fig. 7 corroborates the aforementioned performance as well. In Fig. 7, we present a more comprehensive comparison, where average values of different topology instances are depicted. Regarding the case where QoS constraints are applied, for each topology instance we generate three different QoS pairs for each facility as explained in Section V-B. We compare the performance in terms of average cost and average execution time for three algorithms: 1) the ILP, referred to as IGP; 2) the DESAQoS heuristic; and 3) the baseline ALL-GW solution, where we place gateways in all locations. The extensions of the ILP and DESAQoS algorithms where the QoS constraints are applied are denoted by IGP-C, and DESAQoS-C, respectively.

In Fig. 7(b), we graph the execution time of the related solutions. As expected, the execution time of the IGP solution is the worst, since the execution of the ILP grows exponentially to the number of facilities in the plane. To obtain results we limited the running time of the ILP algorithm to 3 h. Applying this bound we were able to obtain optimal solutions for at most 30 facilities. Up to that point the DESAQoS was shown to calculate solutions that were optimal, as calculated by the ILP in the non-QoS scenarios, and very close to optimal in the QoS scenarios. In that case the difference is on average less than the cost of a two gateways. The execution time of the DESAQoS is polynomial and quite lower than the optimal ILP. The baseline solution of placing a gateway at each facility (ALL-GW) was quite more expensive, and the savings obtained by the proposed solutions increase as the number of nodes increases.

We also conducted experiments to evaluate the performance of the proposed schemes with respect to the area coverage density (ACD). As ACD we define the ratio of the aggregate circle area of the facilities to the area of the squared field. In particular the value of ACD is given by

$$\mathrm{ACD} = \frac{\sum_{i \in V_L} \sum_{d \in D} y_i^d \pi B_d^2}{W^2} \tag{4}$$

where B_d is the largest transmission range of the given LeTE devices set and W is the edge length of the squared field. The above equation can be simplified to

$$\mathrm{ACD} = |V_L| \frac{\pi B_d^2}{W^2}. \tag{5}$$

Within this context, in Fig. 8 we consider a constant number of 25 facilities and variable field dimensions. Again, the obtained cost of DESAQoS is near to the optimal solution,

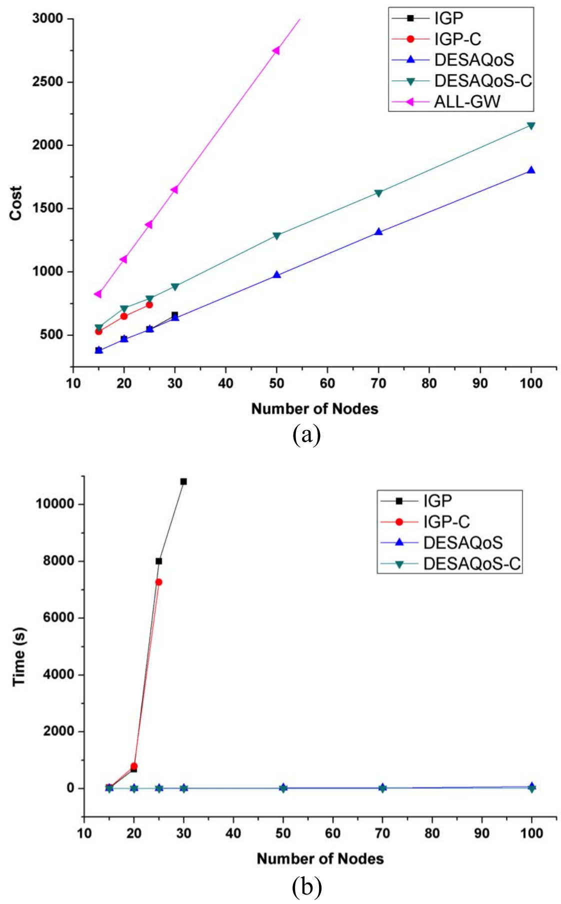


Fig. 7. Average (a) total installation cost and (b) execution time comparison with respect to the number of facilities (1000 m × 1000 m field).

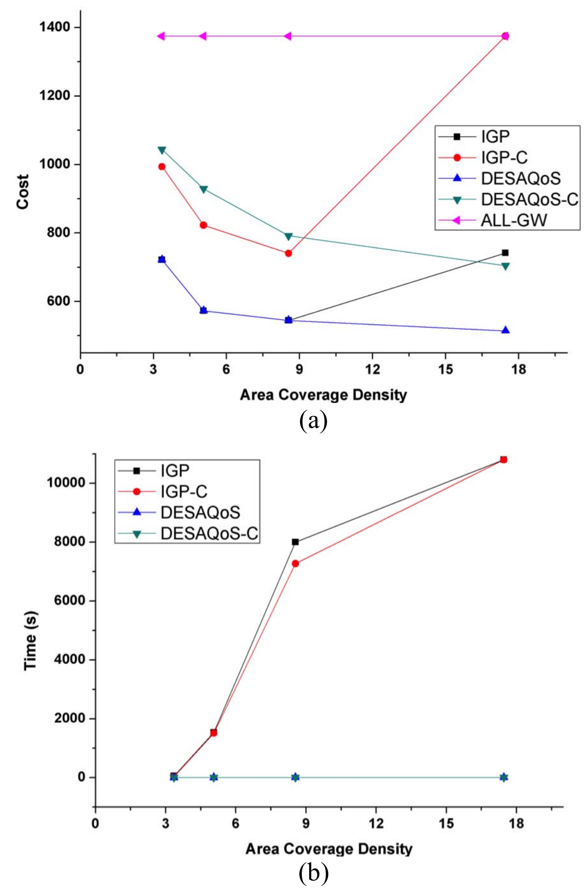


Fig. 8. Average (a) total installation cost and (b) execution time comparison for 25 facilities and varying ACD values.

with significant lower execution time. Since the execution time of the ILP was limited to 3 h, for certain problems of high ACD the IGP algorithm was not able to find good solutions within that time limit. This explains the points where the cost of IGP is greater than DESAQoS. Fig. 8(a) also shows that as the ACD increases the installation cost decreases. This is because in more dense environments there are more connections between facilities and thus smaller need for extra gateways. This fact also results into an increase in execution time of IGP.

Fig. 9 exhibits the impact of interference in the proposed models. For this set of simulations the effective data rates R_d were adapted to account for the efficiency of a collision avoidance MAC protocol. As expected, the efficiency is affected by the average number of nodes that are not allowed to transmit simultaneously under the collision detection approach and thus depends on the density of the facilities in the field. Following the discussion in Section III-A for a (random) topology under simulation, the efficiency H_d of each LeTE d was calculated as follows:

$$H_d = \frac{1}{\sqrt{\xi_d}} \tag{6}$$

$$\xi_d = \begin{cases} |V_L| \frac{\pi (2B_d)^2}{W^2} = 4 \cdot \text{ACD} \\ |V_L|, \textit{if } \text{ACD} > 1/4 \end{cases} \tag{7}$$

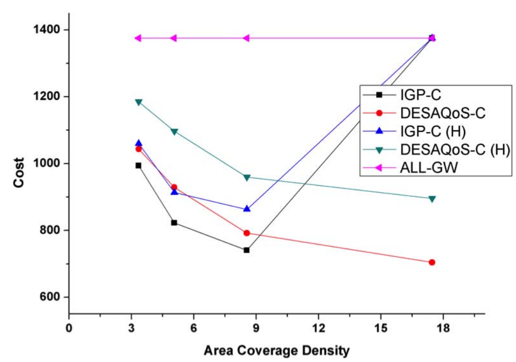


Fig. 9. Average total installation cost comparison when accounting for interference (algorithms indicated with "H") for 25 facilities and varying ACD values.

where ξ is the average number of facilities (assuming uniform node distribution) within the transmission range of an LeTE (more accurately inside the interference range of the LeTE which is taken twice the nominal effective transmission range) as a proportion to the number of facilities placed in the squared field. Note here that the value of ξ cannot exceed the total number of facilities in the topology. In Fig. 9, the lines indicated with H correspond to the performance of IGP and DESAQoS when QoS is considered and interference is taken into account.

Similar to the previous results, the total network cost increases as the network becomes more disconnected. Furthermore, as expected, the average cost is increased compared to the scenario presented in Fig. 8, where interference is assumed static irrespective of topology's density. However, the benefits in cost, derived from the proposed technique, are significant approaching 40% compared to the all gateway approach. The execution times, of the ILP-based algorithm, were observed to be slightly lower due to the fact that some connections are excluded earlier (due to the lack of sufficient capacities in LeTE devices), which resulted in fewer branches that need to be checked by the ILP algorithm.

A case where DESAQoS tends to yield solutions that are further from optimal is when Voronoi points are also utilized for gateway placement. We observed that the clustering algorithm was highly unlikely to generate centroids that are close enough to Voronoi points, and that these selections did not benefit the overall device installation. This is shown in Fig. 6(f) where Voronoi points do not benefit the network installation, like the ILP approach does, resulting into additional gateway. However, as Fig. 7 depicts, the proposed heuristic also provides significant efficiency compared to the naïve case where each facility deploys a gateway.

VI. Conclusion

Motivated by the emergence of fog computing and IoT, we considered the gateway placement and LeTE allocation problem and proposed an optimal ILP formulation and a heuristic algorithm to minimize the IoT network installation cost, while satisfying specific QoS requirements. The LeTE devices differ in their transmission capabilities, which also relate to their cost. The effectiveness of the proposed ILP formulation for the infrastructure planning problem is evaluated by simulation results for several topologies and traffic scenarios. The results exhibit significant savings compared to the case where each facility deploys a gateway. We observed savings up to 52% of the overall installation cost that increase to 60% when centroid points are considered as gateway candidate positions in case of QoS restrictions. The proposed heuristic algorithm was observed to have satisfactory performance, yielding total installation cost that was near optimal and, in special cases, identical to that obtained with optimal ILP algorithm.

References

[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2347–2376, 2015.

[2] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for smart cities," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.

[3] P. Bellavista, G. Cardone, A. Corradi, and L. Foschini, "Convergence of MANET and WSN in IoT urban scenarios," *IEEE Sensors J.*, vol. 13, no. 10, pp. 3558–3567, Oct. 2013.

[4] A. A. Abbasi and M. Younis, "A survey on clustering algorithms for wireless sensor networks," *Comput. Commun.*, vol. 30, nos. 14–15, pp. 2826–2841, 2007.

[5] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: A survey," *Comput. Netw.*, vol. 47, no. 4, pp. 445–487, 2005.

[6] M. Younis and K. Akkaya, "Strategies and techniques for node placement in wireless sensor networks: A survey," *Ad Hoc Netw.*, vol. 6, no. 4, pp. 621–655, 2008.

[7] R. Chandra, L. Qiu, K. Jain, and M. Mahdian, "Optimizing the placement of integration points in multi-hop wireless networks," in *Proc. ICNP*, 2004, pp. 271–282.

[8] F. Li, Y. Wang, X.-Y. Li, A. Nusairat, and Y. Wu, "Gateway placement for throughput optimization in wireless mesh networks," *Mobile Netw. Appl.*, vol. 13, no. 1–2, pp. 198–211, 2008.

[9] B. Aoun, R. Boutaba, Y. Iraqi, and G. Kenward, "Gateway placement optimization in wireless mesh networks with QoS constraints," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 11, pp. 2127–2136, Nov. 2006.

[10] Y. Drabu and H. Peyravi, "Gateway placement with QoS constraints in wireless mesh networks," in *Proc. IEEE 7th Int. Conf. Netw. (ICN)*, 2008, pp. 46–51.

[11] B. He, B. Xie, and D. P. Agrawal, "Optimizing the Internet gateway deployment in a wireless mesh network," in *Proc. IEEE Int. Conf. Mobile Adhoc Sensor Syst. (MASS)*, 2007, pp. 1–9.

[12] T. Vanhatupa, M. Hännikäinen, and T. D. Hämäläinen, "Genetic algorithm to optimize node placement and configuration for WLAN planning," in *Proc. IEEE Wireless Commun. Syst. (ISWCS)*, 2007, pp. 612–616.

[13] F. Xhafa, A. Barolli, C. Sánchez, and L. Barolli, "A simulated annealing algorithm for router nodes placement problem in wireless mesh networks," *Simulat. Model. Pract. Theory*, vol. 19, no. 10, pp. 2276–2284, 2011.

[14] S. C. Ho, "An iterated Tabu search heuristic for the single source capacitated facility location problem," *Appl. Soft Comput.*, vol. 27, pp. 169–178, Feb. 2015.

[15] C.-C. Lin, P.-T. Tseng, T.-Y. Wu, and D.-J. Deng, "Social-aware dynamic router node placement in wireless mesh networks," *Wireless Netw.*, vol. 22, no. 4, pp. 1235–1250, 2016.

[16] I. Gravalos, P. Makris, K. Christodoulopoulos, and E. A. Varvarigos, "Efficient gateways placement for Internet of Things with QoS constraints," in *Proc. GLOBECOM*, 2016, pp. 1–6.

[17] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 388–404, Mar. 2000.

[18] F. Aurenhammer, "Voronoi diagrams—A survey of a fundamental geometric data structure," *ACM Comput. Surveys*, vol. 23, no. 3, pp. 345–405, 1991.

[19] V. Verter, "Uncapacitated and capacitated facility location problems," in *Foundation of Location Analysis*, vol. 155. Boston, MA, USA: Springer, 2011, pp. 25–37.

[20] S. P. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inf. Theory*, vol. IT-28, no. 2, pp. 129–136, Mar. 1982.

[21] D. Arthur and S. Vassilvitskii, "How slow is the k-means method?" in *Proc. 22nd Annu. Symp. Comput. Geom.*, 2006, pp. 144–153.

[22] A. Vattani, "K-means requires exponentially many iterations even in the plane," *Discr. Comput. Geom.*, vol. 45, no. 4, pp. 596–616, 2011.

[23] T. Kanungo *et al.*, "An efficient k-means clustering algorithm: Analysis and implementation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 881–892, Jul. 2002.

[24] S. G. Kolliopoulos and S. Rao, "A nearly linear-time approximation scheme for the Euclidean *k*-median problem," *SIAM J. Comput.*, vol. 37, no. 3, pp. 757–782, 2007.

[25] J. Y. Yen, "An algorithm for finding shortest routes from all source nodes to a given destination in general networks," *Quart. Appl. Math.*, vol. 27, no. 4, pp. 526–530, 1970.

[26] EU VIMSEN Project. *The VIMSEN Communication Infrastructure Design, Deliverable D4.1*. Accessed: Apr. 1, 2016. [Online]. Available: http://www.ict-vimsen.eu/images/jdownloads/Open_Data/public%20deliverables_pdf/vimsen_ir4.1_final_060220150.pdf

[27] *IBM ILOG CPLEX for Python*. Accessed: Oct. 2016. [Online]. Available: http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/index.html

[28] S. Mitchell, M. Osullivan, and I. Dunning, *PuLP: A Linear Programming Toolkit for Python*, Univ. Auckland, Auckland, New Zealand, 2011. [Online]. Available: http://www.optimization-online.org/DB_FILE/2011/09/3178.pdf

Ilias Gravalos received the diploma degree in computer engineering and informatics, M.Sc. degree, and Ph.D. degree in computer science and engineering from the University of Patras, Patras, Greece, in 2008, 2011, and 2016, respectively.

He was an Adjunct Assistant Professor with the Technological Educational Institute of Western Greece, Missolonghi, Greece, and a Researcher with the Computer Technology Institute "Diophantus," Rio, Greece, and the University of Patras. He is currently a Research Fellow with the High Speed Communications Networks Laboratory, School of Electrical and Computer Engineering, National Technical University of Athens, Athens, Greece. He has participated in several EU-funded research projects within the areas of optimization and routing protocols for wireless/IoT and optical networks.

Prodromos Makris was born in Samos, Greece, in 1985. He received the B.Sc., M.Sc., and Ph.D. degrees in context-aware resource management for mobile and fixed networking systems from the University of the Aegean, Mytilene, Greece, in 2007, 2009, and 2013, respectively.

He is currently a Senior Researcher with the National Technical University of Athens, Athens, Greece, serving as a Project and Technical Coordinator Assistant of the H2020-ICT SOCIALENERGY Project. From 2013 to 2016, he was a Post-Doctoral Researcher on context-aware resource management in smart energy networks being also the Project and Technical Coordinator Assistant of the FP7-ICT VIMSEN Project. During the last decade, he has been actively participating in several other national and EC-funded projects (e.g., FP6-IST-UNITE, FP7-ICT-HURRICANE, FP7-ICT-PASSIVE, FP7-ICT-COGEU, and so on). He has over 35 publications in international conferences and journals.

Kostas (Konstantinos) Christodoulopoulos received the diploma degree in electrical and computer engineering, the M.Sc. degree in advance computing from the Imperial College of London, London, U.K., in 2004, and the Ph.D. degree from the Computer Engineering and Informatics Department, University of Patras (UoP), Patras, Greece, in 2009.

He is a Senior Researcher with the School of Electrical and Computer Engineering, National Technical University of Athens, Athens, Greece. He was an Adjunct Assistant Professor with CEID-UoP and a Senior Researcher with Computer Technology Institute Greece, Trinity College Dublin, Dublin, Ireland, and IBM Ireland, Dublin. He has authored or co-authored over 100 peer-reviewed journal and conference papers and he has over 2400 Google Scholar citations with an *H*-index of 24. His current research interests include optimization, algorithms and protocols for optical telecom, and datacom networks.

Dr. Christodoulopoulos was a recipient of the IEEE/OSA JLT Best Paper Award in 2011, the Best Paper Award of Europar in 2013 and the E-science conferences, and several Best Reviewer Awards in 2007. He recently became an Associate Editor of the IEEE/OSA JOURNAL OF OPTICAL COMMUNICATIONS AND NETWORKING. He has given invited talks at OFC in 2015, ONDM in 2017, and ICTON (multiple years).

Emmanouel (Manos) A. Varvarigos received the diploma degree in electrical and computer engineering from the National Technical University of Athens, Athens, Greece, in 1988, and the M.S. and Ph.D. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 1990 and 1992, respectively.

He has held faculty positions with the University of California at Santa Barbara, Santa Barbara, CA, USA, as an Assistant and later an Associate Professor, the Delft University of Technology, Delft, The Netherlands, as an Associate Professor, and the University of Patras, Patras, Greece, as a Professor. He is currently a Professor of electrical and computer engineering with the National Technical University of Athens (NTUA), Athens, Greece. Since 2017, he has been a Professor and the Head of Electrical and Computer Systems Engineering Department, Monash University, Melbourne, VIC, Australia, on leave from NTUA. From 2003 to 2016, he was the Director and then the Scientific Director of the Greek School Network, the main public ISP in Greece connecting over 20 000 schools and other educational units, which has a major role in the development of network technologies and telematic services in Greece. He has participated in over 35 U.S.- and EU-funded research projects in the areas of networking, smart energy grids, and grid and cloud computing, and in many national research projects, and has been the Consortium Coordinator in seven of them.

Dr. Varvarigos has served on the Organizing and Program Committees of over 150 international conferences and has over 350 publications in refereed international journals and conferences.