IoT-Cloud Service Optimization in Next Generation Smart Environments

Marc Barcelo, Alejandro Correa, Jaime Llorca, Antonia M. Tulino, *Fellow, IEEE*,
Jose Lopez Vicario, and Antoni Morell

***Abstract*—The impact of the Internet of Things (IoT) on the evolution toward next generation *smart environments* (e.g., smart homes, buildings, and cities) will largely depend on the efficient integration of IoT and cloud computing technologies. With the predicted explosion in the number of connected devices and IoT services, current centralized cloud architectures, which tend to consolidate computing and storage resources into a few large data centers, will inevitably lead to excessive network load, end-to-end service latencies, and overall power consumption. Thanks to recent advances in network virtualization and programmability, highly distributed cloud networking architectures are a promising solution to efficiently host, manage, and optimize next generation IoT services in smart environments. In this paper, we mathematically formulate the service distribution problem (SDP) in IoT-Cloud networks, referred to as the IoT-CSDP, as a minimum cost mixed-cast flow problem that can be efficiently solved via linear programming. We focus on energy consumption as the major driver of today's network and cloud operational costs and characterize the heterogeneous set of IoT-Cloud network resources according to their associated sensing, computing, and transport capacity and energy efficiency. Our results show that, when properly optimized, the flexibility of IoT-Cloud networks can be efficiently exploited to deliver a wide range of IoT services in the context of next generation smart environments, while significantly reducing overall power consumption.**

***Index Terms*—Internet of things, cloud, edge computing, service optimization, energy efficiency, smart cities.**

I. Introduction

THE Internet of Things (IoT) refers to the network of objects, devices, machines, vehicles, buildings, and other physical systems with embedded sensing, computing, and communication capabilities, that sense and share real-time information about the physical world [1], [2]. When connecting the IoT to the Cloud, vast amounts of data collected from multiple locations can be processed and analyzed to create meaningful information for the end users. At the same time, the intrinsic limitations of lightweight mobile devices (e.g., battery life, processing power, storage capacity) can be alleviated by taking advantage of the extensive resources in the Cloud [3], [4]. The resulting IoT-Cloud paradigm enables a new breed of services and applications (e.g., health system monitoring, traffic control, energy management, vehicular networking), expected to define the essence of next generation *smart* environments (e.g., smart homes, smart buildings, smart cities, smart grids) [5], [6].

However, with the predicted explosion in the number of IoT services and connected devices, traditional centralized cloud architectures, in which computing and storage resources are concentrated in a few large data centers, will inevitably lead to excessive network load, end-to-end service latencies, and unbearable energy costs [7], [8]. In order to meet the tight QoS requirements associated with real-time IoT applications while maximizing overall efficiency, cloud architectures are becoming increasingly distributed [7], with the presence of small cloud nodes at the edge of the network, referred to as cloudlets [9], micro-clouds [10], fog nodes [11], or simply edge cloud nodes [12]. may be organized into hierarchical layers, each with different computing and storage capabilities [13]. In addition, thanks to recent advances in mobile computing [14] and device layer virtualization [15], even end devices can become part of this highly distributed networked compute platform, creating what we refer to as *IoT-Cloud networks*. These are highly virtualized heterogeneous platforms that offer computing, storage, and networking services with unprecedented distribution and proximity to the end user. Compared to traditional centralized clouds, IoT-Cloud networks provide increased flexibility in the allocation of resources to IoT services, and a clear advantage in meeting their stringent latency, mobility, and location-awareness constraints.

In the IoT-Cloud network paradigm, services take the form of virtual network slices of the shared physical infrastructure, which can flexibly and elastically consume compute, storage, and network resources according to changing service requirements. A key aspect driving both performance and efficiency is the actual placement of the service functions, as well as the routing of network flows through the appropriate function instances. In the context of cloud networks, [16] introduced the cloud service distribution problem (CSDP), where the goal is to find the placement of virtual functions and the routing of network flows that meets QoS requirements, satisfies resource capacities and minimizes overall infrastructure cost. The authors provide a network-flow based linear programming

Manuscript received January 31, 2016; revised May 21, 2016 and September 15, 2016; accepted October 16, 2016. Date of publication October 25, 2016; date of current version December 29, 2016. This work was supported by the Spanish Government under Project TEC2014-53656-R. *(Corresponding author: Marc Barcelo.)*

M. Barcelo, A. Correa, J. L. Vicario, and A. Morell are with the Universitat Autònoma de Barcelona, Barcelona 08193, Spain (e-mail: marc.barcelo@uab.cat; alejandro.correa@uab.cat; jose.vicario@uab.cat; antoni.morell@uab.cat).

J. Llorca is with Nokia Bell Labs, Holmdel, NJ 07733, USA (e-mail: jaime.llorca@nokia.com).

A. M. Tulino is with Nokia Bell Labs, Holmdel, NJ, USA, and also with Università degli Studi di Napoli Federico II, Naples 80124, Italy (e-mail: antoniamaria.tulino@unina.it).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSAC.2016.2621398

solution that optimizes the distribution of cloud services with arbitrary function relationships (e.g., service chaining) over a distributed cloud network. However, the work in [16] does not take into account the increased flexibility that arises when introducing the access network and the device layer into the virtualized infrastructure, aspects that are critical for the efficient delivery of IoT services [17].

In this paper, we formalize the IoT-CSDP as a service placement and resource allocation problem that goes beyond traditional information services and cloud architectures to include next generation IoT services and IoT-Cloud infrastructures. Our contributions can be summarized as follows:

- We introduce a flexible mathematical model for IoT-Cloud networks that characterizes the capacity, efficiency, and reliability of sensing, computing, and transmission resources across end device, access, and cloud layers.
- Building on the cloud service model introduced in [16], we characterize a generic IoT service via a directed rooted graph that encodes the relationship between the service functions that act on the source information flows to create the final augmented information that needs to be delivered to the end users.
- We formally define the IoT service distribution problem (IoT-CSDP) as the problem of finding the placement of IoT service functions and routing of network flows across the IoT-Cloud infrastructure that satisfy end user demands while minimizing overall operational cost. We formulate the IoT-CSDP as a minimum cost mixed-cast network information flow problem on a properly augmented graph that allows capturing unicast and multicast flows, and admits optimal polynomial-time solutions.
- We evaluate the solution to the IoT-CSDP in an illustrative set of next generation smart environments that require the joint orchestration of device, access, and cloud layers, and show the impact of the most relevant system parameters. Our results show that the proposed optimization model and associated flow-based solution enables fully exploiting the flexibility of IoT-Cloud networks to efficiently host, manage, and optimize a wide range of IoT services in the context of next generation smart environments. When compared to conventional centralized cloud approaches, our solution achieves up to 80% overall power consumption reductions while guaranteeing much more stringent end-to-end latency constraints.

The remainder of this paper is organized as follows: Section II reviews related work. Section III presents the IoT-Cloud network paradigm. Section IV describes the system model. Section V introduces the IoT-CSDP and its flow-based formulation. Section VI presents simulation results from the solution to the IoT-CSDP in the context of illustrative smart environments: smart cities, smart buildings, and smart transportation systems. Finally, Section VII summarizes the paper and presents the main conclusions.

II. Related Work

In the context of integrating end devices into the Cloud, special attention has been given to wireless sensors networks (WSNs) due to their relevance in the IoT [18]. Misra *et al.* [19] present a theoretical model for Sensor-Cloud architectures and study the performance improvements that can be obtained over traditional WSN architectures. Mitton *et al.* [20] describe a Sensor-Cloud architecture that makes use of the Contiki Operating System to provide end users with access to the data acquired by different heterogeneous sensing infrastructures. Aslam *et al.* [21] propose WSN-SOrA, a service oriented architecture that orchestrates service provisioning for embedded networked systems in large scale WSNs. Zhu *et al.* [22] propose a sensory data processing framework that reduces the amount of data forwarded to the Cloud by processing data at the gateway layer using monitoring, filtering, prediction, compression, and recommendation techniques. Zhu *et al.* [23] address sensor data usefulness and WSN reliability by proposing a WSN-Cloud integration scheme in which WSN gateways selectively transmit information to the Cloud based on the time and priority features of the users' requested data, and wireless sensors execute a priority-based sleep scheduling algorithm to reduce their power consumption. In [24], a trust and reputation calculation and management system with authentication is proposed to address some of the security concerns that arise when integrating WSNs into the Cloud. Zhu *et al.* [25] propose a position-based sleep-scheduling mechanism that reduces the active time of duty-cycled sensors in order to enhance the lifetime of wireless sensors connected to the Cloud. This problem is also addressed in [26], where the authors propose a management framework to control the duty cycle of sensors in IoT environments that may be performing multiple tasks with different quality-of-information requirements.

In addition to wireless sensors, the efficient integration of smart devices, such as smartphones and connected vehicles, has also attracted the attention of the research community. Mukherjee *et al.* [27] analyze the use of smartphones as shared edge computing devices, taking advantage of their increasing processing capabilities. In [28], software defined networking (SDN) and edge computing technologies are combined to enable programmable vehicular ad hoc networks (VANETs).

The work in [29] presents a model for fog computing architectures with edge or fog nodes forming an intermediate layer between device and cloud layers. The authors show how the fog layer enables relevant energy savings when supporting IoT applications, but do not provide any optimization strategy to distribute the service functions over the entire infrastructure. Farris *et al.* [30] develop a federation strategy among IoT devices and clouds so they can share their computing resources in order to maximize the total number of executed tasks. Aazam and Huh [31] present an IoT resource management system in which micro data centers co-located with network gateways at the edge of the network perform billing, pricing, and resource reservation for IoT services. They also propose a probabilistic resource estimation model that takes into account the fluctuating connectivity behavior of IoT devices [32]. While existent literature provides a significant amount of studies describing alternative models and architectures that illustrate the advantages, limitations, and challenges associated with IoT-Cloud networks, to the best of our knowledge, this is the first work that mathematically formalizes the problem of optimal distribution of generic IoT services over IoT-Cloud networks, taking into account the heterogeneous nature of sensing, transmission, and computing resources across the physical infrastructure, as well as the unique function inter-relationships, mixed-cast flow nature, and tight QoS requirements of IoT services.

III. IoT-Cloud Networks

IoT-Cloud networks result from the convergence of distributed cloud networks and the IoT. These can take advantage of the ubiquitous sensing capabilities of the IoT and the virtually unlimited computing resources of the Cloud to offer a new class of services that create *augmented* information from the cloud-based analysis of IoT-based data. In IoT-Cloud networks, sensors, smartphones, connected vehicles, are not simple endpoints, but smart sensing, storing, and computing resources that can be jointly orchestrated with the rest of the cloud infrastructure [33]. In this new paradigm, and thanks to recent advances in network functions virtualization (NFV) and software defined networking (SDN) technologies [34], service functionality can be dynamically allocated across the resulting highly distributed platform and flows can be routed through the appropriate service functions in order to maximize end devices' battery life, optimize service performance, and minimize overall operational cost.

The main advantages of IoT-Cloud networks with respect to traditional centralized cloud approaches are:

- *Low Latency:* service functions can be placed at the edge of the network in close proximity to the end users to support real-time services.
- *High Reliability:* service functions can be replicated across a highly distributed platform for increased fault tolerance and disaster recovery.
- *Reduced Operational Cost:* service functions that require large inputs or produce large outputs can be placed close to their respective sources and/or destinations for reduced network load and associated operational cost.
- *High Flexibility:* virtualization technologies allows sharing the heterogeneous physical infrastructure among multiple services that can elastically tap into a rich pool of resources without the need of dedicated deployments.
- *Location Awareness and Mobility Support:* the current location of mobile users can be leveraged to provide personalized contextual services as well as service mobility.
- *Scalability:* the distribution of computing and storage resources close to the sources of information allows scaling the number of connected devices and services without saturating the transport network.

IoT-Cloud networks hence emerge as an ideal platform for the implementation of IoT services in the context of a wide range of smart environments, such as smart grids, smart mobility, smart buildings, and smart cities. For instance, in smart grids, the data collected by smart meters can be pre-analyzed at the edge of the network reducing the amount of data sent to cloud data centers, saving both computing and transport resources, and significantly reducing service latency. In smart mobility services, the analysis of data collected at the edge of the network can significantly increase responsiveness to sudden events, such as vehicle collisions, improving the efficiency and safety of transportation networks. In smart buildings, while centralized cloud resources can be used for complex data analysis, edge cloud nodes are ideally suited for low-latency real-time physical system control and actuation.

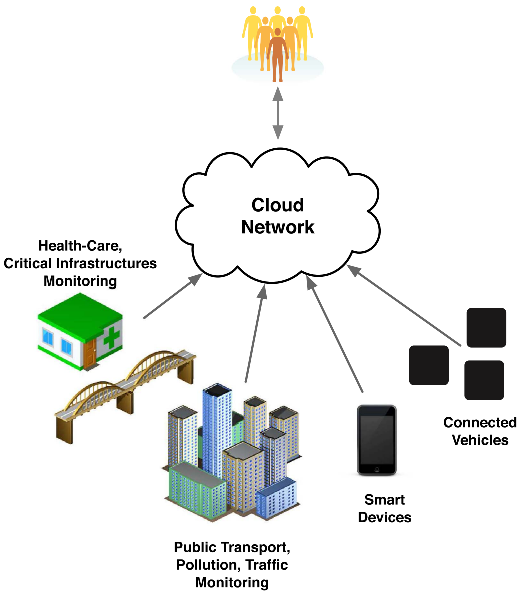

Cloud Network
Health-Care, Critical Infrastructures Monitoring
Connected Vehicles
Smart Devices
Public Transport, Pollution, Traffic Monitoring

Fig. 1. IoT-Cloud network resulting from the integration of IoT devices into the cloud network infrastructure.

IV. System Model

After describing the suitability of IoT-Cloud networks for the delivery of IoT services in future smart environments, we now introduce the mathematical network and service models that will be used to optimize the distribution of IoT services over IoT-Cloud networks.

What is a directed graph?

A. Network Model

We refer to an IoT-Cloud network as the converged platform that results from the integration of programmable IoT devices into the cloud infrastructure (Fig. 1).

We model an IoT-Cloud network as a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with V vertices and E edges representing the set of network nodes and links, respectively. A node in $\mathcal{V}$ may represent an end device, an access point, or a cloud node (at possibly different hierarchical layers). Each node is characterized by its energy resources (e.g., power grid, battery), processing resources (e.g., processor, microprocessor), and data acquisition or sensing resources (e.g., camera, sensors, I/O interfaces). We denote by c_u^{pr} and c_u^{sn} the data processing and sensing capacities (in bits per second or bps) at node $u \in \mathcal{V}$, and by e_u^{pr} and e_u^{sn} the data processing and sensing unit energy costs (in Watts per bps) at node $u \in \mathcal{V}$, respectively. Nodes are interconnected via wireless or wireline links, each characterized by their transmission capacity and unit energy cost. We use c_{vu}^{tr} and e_{vu}^{tr} to denote the capacity (in bps), and the unit energy cost (in Watts per bps) of link $(v, u) \in \mathcal{E}$, respectively.

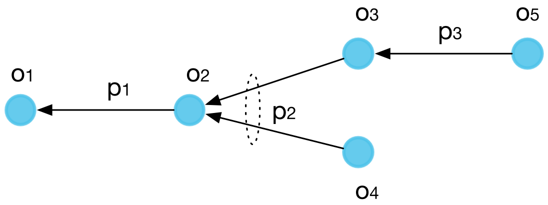


Fig. 2. An example of a service graph, $\mathcal{T}_\phi = (\mathcal{A}_\phi, \mathcal{O}_\phi)$, with $|\mathcal{O}_\phi| = 5$ objects, $|\mathcal{A}_\phi| = 4$ edges, and $|\mathcal{P}| = 3$ virtual functions.

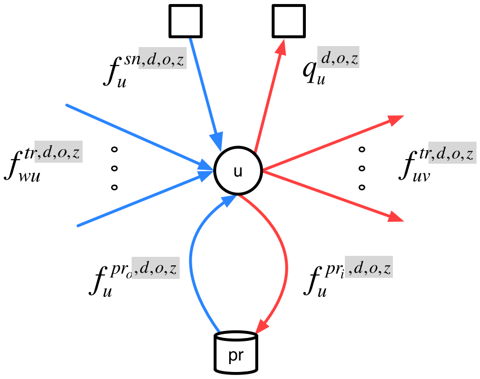


Fig. 3. Input/output user-object flows at node $u \in \mathcal{V}$.

B. Service Model

We denote by $\mathcal{O}$ the set of information objects or flows that can be captured, processed or transported over the IoT-Cloud network, and by $\mathcal{P}$ the set of virtual functions that can process information objects as part of an offered service. A given information object $o \in \mathcal{O}$ is, in general, the output of a function $p_o \in \mathcal{P}$ that requires the set of objects $\mathcal{Z}(o)$ as input. For example, in a mobility assistance service, the final object delivered to the end user may be personalized mobility information resulting from the analysis of data collected by multiple sensing devices (e.g., GPS devices, wireless sensors, cameras).

We represent a generic IoT-Cloud service ϕ by a directed rooted graph $\mathcal{T}_\phi = (\mathcal{A}_\phi, \mathcal{O}_\phi)$. For any node $o \in \mathcal{O}_\phi$, there is a set of incoming edges $\{(z, o) \in \mathcal{A}_\phi : z \in \mathcal{Z}(o)\}$, as shown in Fig. 2. Hence, the set of objects $\mathcal{Z}(o) \subset \mathcal{O}$ required to generate object o via function p_o are represented as the children of o in the service graph $\mathcal{T}_\phi$. In particular, the root of the service graph $r_\phi \in \mathcal{O}$ represents the final information object that needs to be delivered to the end user(s). The service graph hence encodes the relationship between the virtual functions that act on the source objects to create the final object that needs to be delivered to the end users. When a user requests service ϕ, the user is, in essence, requesting the final information object or flow represented by the root of the service graph r_ϕ.

we use $H_{d,o}$ to denote the maximum delay allowed for the delivery of content object $o \in \mathcal{O}$ at destination $d \in \mathcal{V}$, where we denote by h_{vu} and r_{vu} the transport delay (in seconds) and reliability (in terms of packet delivery ratio) associated with link $(v, u) \in \mathcal{E}$, respectively, and by h_u the processing delay at cloud node $u \in \mathcal{V}$. In addition, we use E_u (in Watts) to denote the maximum power consumption allowed at battery-powered node $u \in \mathcal{V}$ in order to guarantee its minimum lifetime requirements.

C. Flow Model

We model the sensing, processing, and transmission of information across the IoT-Cloud network in terms of the following user-object and global flows:

- User-object flows: are characterized by a triplet (d, o, z), which indicates that the given flow is carrying information of object $z \in \mathcal{O}$ used to deliver final product $o \in \mathcal{O}$ at destination $d \in \mathcal{V}$. In particular, $f_{vu}^{tr,d,o,z}$, $f_{vu}^{sn,d,o,z}$, $f_u^{pr_i,d,o,z}$ and $f_u^{pr_o,d,o,z}$ indicate the fraction of object z carried/captured/processed by edge $(v, u) \in \mathcal{E}$ or node $u \in \mathcal{V}$ for final product $o \in \mathcal{O}$ at destination $d \in \mathcal{V}$, respectively. Note that we differentiate between $f_u^{pr_i,d,o,z}$ and $f^{pr_o,d,o,z}$, which denote the input and output flows of the processing unit at node $u \in \mathcal{V}$ associated with triplet (d, o, z). Fig. 3 illustrates the network flows associated with a given triplet (d, o, z) at node $u \in \mathcal{V}$, where pr represents the processing unit that hosts virtual functions and $q_u^{d,o,z}$ is a binary demand parameter that indicates if node $u \in \mathcal{V}$ requests object $z \in \mathcal{O}$. Note that $q_u^{d,o,z} = 0$ if $u \neq d$ or $z \neq o$, since users only request final information objects for themselves.
- Global flows: f_{vu}^{tr}, f_u^{sn} and f_u^{pr} determine the total amount of information flow carried, captured, and processed at a given physical link or node, respectively.

V. The Internet of Things - Cloud Service Distribution Problem (IoT-CSDP)

The IoT-CSDP builds on the linear CSDP [16] and extends it via the characterization of the access network and the device layer. In particular, the IoT-CSDP complements the CSDP as follows:

- The sensing or data acquisition capabilities of end devices, such as sensors, video cameras or RFID tags, are considered.
- The capacities and energy costs of processing and transmission resources are modeled across the entire heterogeneous IoT-Cloud network, including core, metro, access, and end devices.
- The limited energy resources of battery powered devices are taken into account in order to guarantee their minimum lifetime requirements.
- The reliability of links is considered in order to characterize the possible packet losses and associated retransmissions, particularly relevant in low power wireless links.
- The end-to-end latency is modeled by considering the delay contributions along the entire service path, from the nodes that generate the source data to the delivery of the final augmented information to the end users.

- The IoT-CSDP captures the unique nature of IoT services, typically characterized by a multicast upstream phase in which sensed data that can be used for multiple services and end users is uploaded to edge cloud nodes, and a typically unicast downstream phase in which specific information resulting from the processing of sensed data is delivered in a personalized manner to the end users.
- Given that the cloud network infrastructure is shared among multiple services, the IoT-CSDP assumes a load-proportional cost (e.g., energy) model, in which the cost of a given service is proportional to its use of the physical infrastructure. This results in a significantly reduced complexity linear program that enables faster reactions to variations of users' service demands.

A. Mathematical Formulation

The IoT-CSDP is formulated as a minimum cost mixed-cast flow problem as follows:

1) Objective Function: We define a generic linear cost function characterized by the energy costs of the IoT-Cloud network resources:

$$\underset{f^{tr},f^{sn},f^{pr}}{\text{minimize}} \sum_{(v,u)\in\mathcal{E}} e_{vu}^{tr} f_{vu}^{tr} + \sum_{u\in\mathcal{V}} \left(e_u^{sn} f_u^{sn} + e_u^{pr} f_u^{pr} \right) \quad (1)$$

2) Generalized Flow Conservation Constraints: User-object flows must satisfy demand and flow conservation. By modeling the demand $q_u^{d,o,z}$ as part of the outgoing flow of node u, we can use the following generalized flow conservation constraints:

$$q_u^{d,o,z} + f_u^{pr_i,d,o,z} + \sum_{w\in\mathcal{N}^+(u)} f_{uw}^{tr,d,o,z}$$
$$= f_u^{sn,d,o,z} + f_u^{pr_o,d,o,z} + \sum_{v\in\mathcal{N}^-(u)} f_{vu}^{tr,d,o,z} \quad \forall u, d, o, z. \quad (2)$$

Flow conservation constraints state that the outgoing flow associated with a given triplet (d, o, z) must be equal to the incoming flow for that same triplet, for any node $u \in \mathcal{V}$. As illustrated in Fig. 3, the outgoing flow is composed of the outgoing transport flows, the processing flow leaving node u towards the processing unit, and the demand flow; while the incoming flow is composed of the incoming transport flows, the capturing flow, and the processing flow going out of the processing unit.

In addition, each processing unit must satisfy the following flow conservation constraints:

$$f_u^{pr_o,d,o,z} \le f_u^{pr_i,d,o,y} \quad \forall u, d, o, z, y \in \mathcal{Z}(z). \quad (3)$$

These make sure that in order to have a processed flow z for demand (d, o), a flow associated with each of the input objects required to generate z, $y \in \mathcal{Z}(z)$, for demand (d, o), must be present at the input of the processing unit.

3) Function Availability Constraints: These constraints allow restricting the set of virtual functions that can be implemented at a given node:

$$f_u^{pr_o,d,o,z} = 0 \quad \forall u, d, o, z, p_z \notin \mathcal{P}_u, \quad (4)$$

where $P_u \subset P$ is the set of virtual functions available at node $u \in \mathcal{V}$.

4) Source Constraints: These constraints initialize the availability of source/sensed/captured information objects at their respective sources:

$$f_u^{sn,d,o,z} = 0 \quad \forall u, d, o, z \notin O_u, \quad (5)$$

where $O_u \subset \mathcal{S}$ denotes the set of objects that are sourced/captured at node u, with $\mathcal{S} \subset \mathcal{O}$ denoting the set of available source information objects. In addition, we need to make sure that source objects $\mathcal{S} \subset \mathcal{O}$ are not created in the network:

$$f_u^{pr_o,d,o,z} = 0 \quad \forall u, d, o, z \in \mathcal{S}. \quad (6)$$

5) Mixed-Cast Constraints: Mixed-cast constraints allow modeling the unicast or multicast nature of flows via the corresponding relationship between user-object and global flows. Since a single captured object can be used to satisfy multiple demands, captured flows are said to be multicast. Hence, the user-object that capture flows for the same object, but for different destinations, are allowed to overlap:

$$f_u^{sn,d,o,z} \le f_u^{sn,z} \quad \forall u, d, o, z, \quad (7)$$
$$\sum_{z\in\mathcal{O}} f_u^{sn,z} B_z = f_u^{sn} \quad \forall u, \quad (8)$$

where B_z denotes the bitrate of object z, and therefore f_u^{sn} is defined in bps.

On the other hand, transport and processing flows can either be unicast or multicast. In particular, for IoT services, the transport of captured objects that may be useful to create multiple final objects for multiple users are modeled as multicast flows. The multicast or unicast nature of transport flows that carry processed objects may, however, be modeled as unicast flows if the given processed object is personalized for a specific destination. Processing flows follow the same reasoning and are hence modeled as unicast if the processed flow is destined for a unique destination, and as multicast, otherwise. Accordingly, we use the following constraints for multicast transport and processing flows:

$$f_{vu}^{tr,d,o,z} \le f_{vu}^{tr,z} \quad \forall u, d, o, z, \quad (9)$$
$$\sum_{z\in\mathcal{O}} f_{vu}^{tr,z} B_z = f_{vu}^{tr'} \quad \forall u, \quad (10)$$
$$f_u^{pr_o,d,o,z} \le f_u^{pr,z} \quad \forall u, d, o, z, \quad (11)$$
$$\sum_{z\in\mathcal{O}} f_u^{pr,z} B_z \gamma_z = f_u^{pr} \quad \forall u. \quad (12)$$

Note that user-object flows are sized by the rate in bps of the object, B_z. In addition, output processing flows are scaled by a factor γ_z that captures possible relative flow size changes associated with the generation of object z from its input $\mathcal{Z}(z)$ via function p_z. As a result, $f_{vu}^{tr'}$ and f_u^{pr} are defined in bps.

When transporting or processing personalized objects destined for a specific destination, the corresponding transport and processing flows are modeled as unicast. In this case, user-object flows cannot overlap and must be added across both objects and destinations:

$$\sum_{d\in\mathcal{V}}\sum_{o\in O}\sum_{z\in O} f_{vu}^{tr,d,o,z} B_z = f_{vu}^{tr'} \quad \forall(v,u), \tag{13}$$

$$\sum_{d\in\mathcal{V}}\sum_{o\in O}\sum_{z\in O} f_{u}^{pr_o,d,o,z} B_z \gamma_z = f_{u}^{pr} \quad \forall u. \tag{14}$$

Note that at this point, transport global flows are denoted by $f_{vu}^{tr'}$, since the actual total transport flows need to account for possible retransmissions in the presence of unreliable links, as shown in equation (19) in the following subsection.

6) QoS Constraints:

- Latency: We use $\delta_{d,o,z}$ to denote the (local) delay associated with a particular user-object flow (d,o,z). This is computed as the weighted sum of the delay associated with the transport and processing of (d,o,z) flows. That is, $\forall d,o,z$:

$$\delta_{d,o,z} = \sum_{(v,u)\in\mathcal{E}} f_{vu}^{tr,d,o,z} h_{vu} + \sum_{u\in\mathcal{V}} f^{pr_o,d,o,z} h_u, \tag{15}$$

  where h_{vu} and h_u (seconds) are the average transport delay at link (v,u) and processing delay at node u, respectively. The h_{vu} values include the queuing time that each object spends at node v before being forwarded, as well as the propagation time of link (v,u). The h_u values depend on the processing capacity of node u.

  Note that in (15), without loss of generality, we assume input processing flows to have zero delay and capture all processing delay with the output processing flows, as described in [16]. We then use $\delta_{d,o,z}^{ag}$ to denote the aggregate delay associated with user-object flow (d,o,z), computed as the sum of the local delay, $\delta_{d,o,z}$, plus the maximum across the aggregate delays of all of its input flows $\delta_{d,o,y}^{ag}$, $\forall y \in \mathcal{Z}(z)$, as:

$$\delta_{d,o,z}^{ag} = \delta_{d,o,z} \quad \forall d,o,z \in \mathcal{S}, \tag{16}$$

$$\delta_{d,o,z} + \delta_{d,o,y}^{ag} \le \delta_{d,o,z}^{ag} \quad \forall d,o,z,y \in \mathcal{Z}(z). \tag{17}$$

  Finally, the aggregate delay associated with the delivery of final object o at destination d is constrained according to the specified maximum latency requirements $H_{d,o}$ (in seconds):

$$\delta_{d,o,o}^{ag} \le H_{d,o} \quad \forall d,o. \tag{18}$$

- Reliability: The reliability of links has a relevant impact on the traffic flow due to the retransmissions caused by packet losses, particularly in low power wireless links. The average number of retransmissions required in link $(v,u) \in \mathcal{E}$ is modeled as the reciprocal of its packet delivery ratio:

$$f_{vu}^{tr} = f_{vu}^{tr'} / r_{vu} \quad \forall(v,u). \tag{19}$$

  The value of the packet delivery ratio, r_{vu}, is between zero (i.e., disconnected nodes) and 1 (i.e., ideal link), and therefore it increases the actual transport flow.

- Battery Lifetime: In order to guarantee battery-powered devices lifetime requirements, the total power consumption at node $u \in \mathcal{V}$, considering transport, processing and sensing, must be lower than its maximum power consumption E_u (in Watts). Then:

$$\sum_{(u,w)\in\mathcal{E}} f_{uw}^{tr} e_u^{tx} + \sum_{(v,u)\in\mathcal{E}} f_{vu}^{tr} e_u^{rx} + f_u^{sn} e_u^{sn} + f_u^{pr} e_u^{pr} \le E_u \quad \forall u, \tag{20}$$

  where e_u^{tx} and e_u^{rx} denote the transmission and reception cost (in Watts per bps), respectively.

7) Capacity Constraints: Global flows must satisfy capacity constraints:

$$f_{vu}^{tr} \le c_{vu}^{tr} \quad \forall(v,u), \tag{21}$$

$$f_{u}^{sn} \le c_{u}^{sn} \quad \forall u, \tag{22}$$

$$f_{u}^{pr} \le c_{u}^{pr} \quad \forall u. \tag{23}$$

8) Integer/Fractional Flow Constraints: User-object flows can either be binary or fractional. That is, $\forall(v,u), u, d, o, z$:

$$f_{vu}^{tr,d,o,z},\ f_u^{pr_i,d,o,z},\ f_u^{pr_o,d,o,z},\ f_u^{sn,d,o,z} \in \{0,1\}, \tag{24}$$

or

$$f_{vu}^{tr,d,o,z},\ f_u^{pr_i,d,o,z},\ f_u^{pr_o,d,o,z},\ f_u^{sn,d,o,z} \in [0,1]. \tag{25}$$

The use of fractional flow variables allows splitting service flows over multiple paths, providing added flexibility to optimize the use of IoT-Cloud resources. In addition, the resulting linear program admits optimal polynomial time solutions. However, the use of fractional flows challenges the precise computation of end-to-end latencies in mesh network topologies [35]. In the next section, we choose to solve the IoT-CSDP with binary flow variables in order to precisely model end-to-end delay in mesh network topologies. While the complexity of the resulting integer linear program increases exponentially with the number of binary variables, the IoT-CSDP can be efficiently solved to within a few minutes in all tested scenarios.

We remark that the solution to the IoT-CSDP is centralized, i.e., it requires collecting global information about service demands and network resources at a centralized network controller, and disseminating the solution to all network nodes. In this work, as in [35], we assume that service demands and network resources change at a longer time-scale than the time needed to implement a new IoT-Cloud configuration, leaving aspects related to distributed implementations of our solution to handle faster dynamics as part of future work.

VI. Evaluation

In this section, we present results from the solution to the IoT-CSDP for illustrative IoT services in smart environments, obtained via the linear programming solver Xpress-MP. We analyze and compare the efficiency of the IoT-Cloud solution, which finds the optimal location of IoT service functions exploiting the full flexibility of the IoT-Cloud infrastructure with: i) a conventional cloud approach, in which all service functions are centralized at the highest cloud layer, ii) the recently proposed cloudlet or fog approach [36], in which the processing of IoT services is handled by micro clouds located one hop away from the end devices, and iii) a fully distributed approach, in which all service functions are executed at the end devices.

TABLE I

TYPICAL CAPACITIES AND EFFICIENCIES OF IOT-CLOUD NETWORK RESOURCES

	Capacity	Efficiency
Cloud Node (HO)	53.5 Million MIPS	500 MIPS/W
Cloud Node (IO)	26 Million MIPS	200 MIPS/W
Cloud Node (EO)	13 Million MIPS	133 MIPS/W
Micro Cloud Node (MC)	6.5 Million MIPS	100 MIPS/W
Wireless Sensor/Actuator	1 MIPS	480 MIPS/W
Smart Device	2000 MIPS	50-1000 MIPS/W
Connected Vehicle	2000 MIPS	1000 MIPS/W
Optical Link	4480 Gbps	12.6 nJ/bit
4G Link (Down/Up)	72/12 Mbps	76.2/19 μJ/bit
WiFi Link	150 Mbps	300 nJ/bit
ZigBee Link	250 kbps	100 nJ/bit

A. Simulation Details

We consider a hierarchical IoT-Cloud network architecture composed of three main layers: i) a cloud layer, in which cloud nodes are organized into 3 tiers, i.e., a head office (HO) node representing the largest centralized data center, intermediate offices (IOs), and end offices (EOs), ii) an access layer, composed of base station (BS) nodes hosting micro-clouds (MCs) or cloudlets [9], and iii) a device layer, containing wireless sensors, smart devices (e.g., smartphones, tablets, smart glasses), and connected vehicles. The specific configuration of the IoT-Cloud network will be described for each of the simulated scenarios.

Precisely modeling the processing capacity (in MIPS) and energy efficiency (in MIPS/W) of the IoT-Cloud nodes is difficult due to the limited information disclosed by cloud and network operators. In this paper, we compute approximate values using information extracted from [37] (for cloud equipment) and [38] (for wireless sensors/actuators), which are presented in Table I. Note that due to resource consolidation and multiplexing gains, the efficiency of cloud nodes increases at higher layers. Regarding the efficiency of smart devices, given their heterogeneity, we consider a representative range of values that go from 50 to 1000 MIPS/W [39]. We assume that connected vehicles are equipped with similar processors [40]. Finally, note that the relatively high processing efficiency of sensors/actuators is due to the use of low-power microprocessors. However, they are very limited in terms of battery and computing capacity.

In terms of communication technologies, cloud nodes and base stations are connected via optical links, smart devices use 4G or WiFi, and wireless sensors communicate using the ZigBee protocol. In order to model ZigBee links, which are particularly lossy due to their low transmission power, we assume that these are time-varying and follow the log-distance path loss model described in [41] for urban micro-cell scenarios. Then, the path losses (PL) are modeled as follows:

$$PL(dB) = PL_0(dB) + 20 - 12.5\gamma + 10\gamma\log(d) + 3\log(f_c/5) + X, \tag{26}$$

where PL_0 is the path loss at the reference distance d_0 (1m), γ is the path loss exponent, d is the communication distance, f_c is the system frequency (in GHz) and $X \sim \mathcal{N}(0, \sigma^2)$ is zero mean Gaussian noise that models the shadowing effects. This model assumes γ=2.8 and σ=4 dB [41]. Moreover, a collision-free MAC layer is assumed to obtain results independent from the specific MAC mechanism [42]. Taking into account the general characteristics of ZigBee transceivers, we assume f_c=2.4 GHz, $PL_0(dB)$=35 dB [43], a transmit power of 3 dBm, a sensitivity value of -91 dBm [44] and omnidirectional antennas. We also consider that the packets generated at wireless sensors have a length of 127 bytes (i.e., standard packet size in IEEE 802.15.4). On the other hand, we assume that the channel losses in optical, 4G, and WiFi links do not have a relevant impact on their reliability, compared to ZigBee links [41].

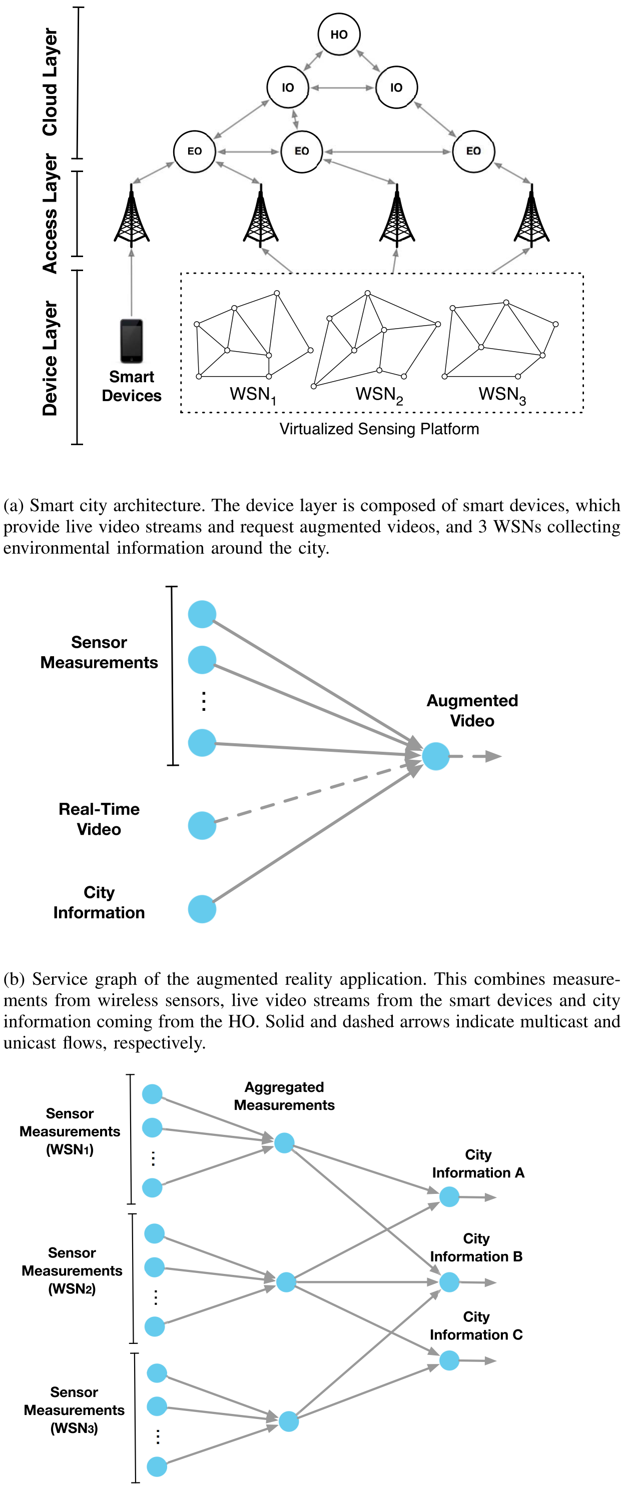
The average transport delay values considered in the simulations include the effect of both queuing and propagation delays. While the first is generally higher for the wireless links at the device and access layers due to their limited transmission capacities, the second is typically higher for the optical links connecting cloud nodes due to the longer distances among them. Taking this into account, we assume an average delay of 5 ms for wireless links, while the delay of optical links depends on the hierarchy level of the cloud node. In particular, we assume that a packet needs an additional 2 ms to reach an end office, 7 ms to reach an intermediate office, and 15 ms to reach a head office. The processing delay mainly depends on the particular processing capabilities of each node. In the simulation results, we assume that wireless nodes have an average processing delay of 4 ms, while cloud nodes, which can process larger amounts of bits simultaneously, have a processing delay of 1 ms.

In the following, we compute the solution to the IoT-CSDP in a number of representative smart environments (i.e., smart city, smart building, and smart mobility) to evaluate the impact of different system parameters in the IoT-Cloud solution, such as transmission and processing efficiencies, number of users, and latency requirements.

B. Smart City

We consider the smart city architecture illustrated in Fig. 4(a), in which end users request information using their smart devices. Since smart cities may provide different types of IoT services, we solve the IoT-CSDP for two relevant services: augmented reality and city monitoring.

1) Augmented Reality Service: We consider an augmented reality service in which users consume video streams that result from the combination of the real time video stream captured by the camera of their own smart device, contextual information downloaded from the Internet, and data from the city collected by the WSNs. In particular, we assume that the augmented video results from the combination of 200 kbps of source video from the user device, 50 kbps of city information, and 1 packet/s from each wireless sensor. In terms of processing complexity, we assume that 2000 instructions per bit are required to generate the augmented video. In order to take into account current network access technologies, we assume that half of the users access the network using 4G and the other half use a WiFi connexion. Note that both captured and augmented video must be sent via unicast, but the contextual



(a) Smart city architecture. The device layer is composed of smart devices, which provide live video streams and request augmented videos, and 3 WSNs collecting environmental information around the city.

(b) Service graph of the augmented reality application. This combines measurements from wireless sensors, live video streams from the smart devices and city information coming from the HO. Solid and dashed arrows indicate multicast and unicast flows, respectively.

(c) Service graph of the city monitoring application. The city information is obtained through the analysis of the aggregated measurements collected by 3 different WSNs. Sensor readings are allowed to be sent via multicast, but the city information can only be sent via multicast for multiple users requesting the same information simultaneously.

Fig. 4. Smart city environment

information coming from the internet and the wireless sensors can be sent via multicast, as the same data can be used for multiple destinations. The service graph for this application is shown in Fig. 4(b).

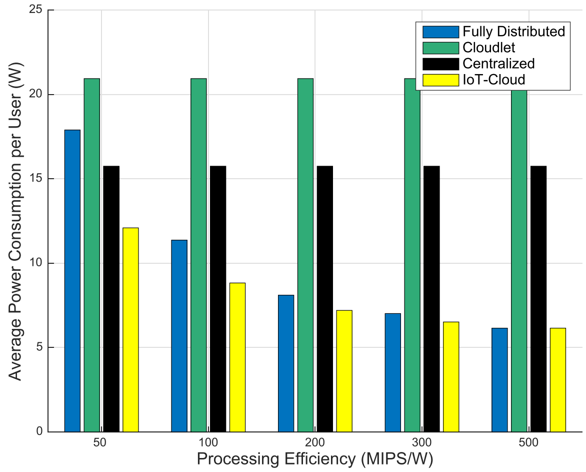


Fig. 5. Average power consumption per user of the augmented reality application for different smart device processing efficiencies.

In Fig. 5, we show the average power consumption for different smart device processing efficiencies. The simulation results show that the IoT-Cloud solution adapts the offloading decisions according to the processing efficiency of end devices. Specifically, note that if the device processing efficiency is sufficiently high, i.e., 500 MIPS/W, the IoT-Cloud solution allocates the augmented video processing function at the end devices. Otherwise, some processing tasks have to be offloaded to the cloud in order to reduce the overall processing cost. We show that for efficiencies between 50 and 300 MIPS/W the IoT-Cloud uses hybrid solutions, in which the placement strategy is not uniform for all users. In these cases, the processing tasks of WiFi users are offloaded to the Cloud, while the network access cost of 4G devices keeps their video processing functionality at the end devices.

We can also observe that the cloudlet solution yields an overall power consumption that is even higher than that of the centralized approach. This is because the micro cloud nodes have lower processing efficiencies than the larger cloud data centers, and this does not get compensated by the savings in terms of transport cost. Note that we consider the use of highly efficient optical fiber links throughout the cloud layer.

2) *City Monitoring Service:* Thanks to the deployment of large scale wireless sensor networks, smart cities can provide real-time information about the city and its infrastructures. In this section, we consider a city monitoring service in which end users request city information that is generated from the analysis of data collected by wireless sensors, such as in [45]. The service graph of this application is shown in Fig. 4(c). We evaluate the impact of the city information bitrate and the number of simultaneous users requesting the same information.

In Fig. 6, we compare the power consumption for different user demands in terms of bitrate. A high bitrate means that the user requests detailed information, while low bitrates indicate that the service only provides the most relevant information. We assume that end users request City Information B (See Fig. 4(c)), which receives the aggregated measurements of WSN_1, WSN_2 and WSN_3 as input. We assume smart

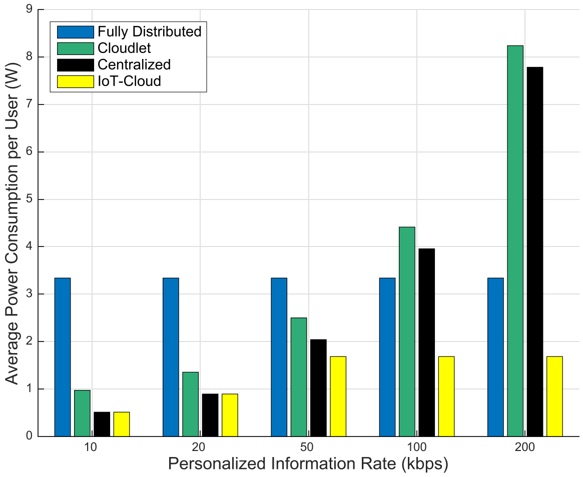
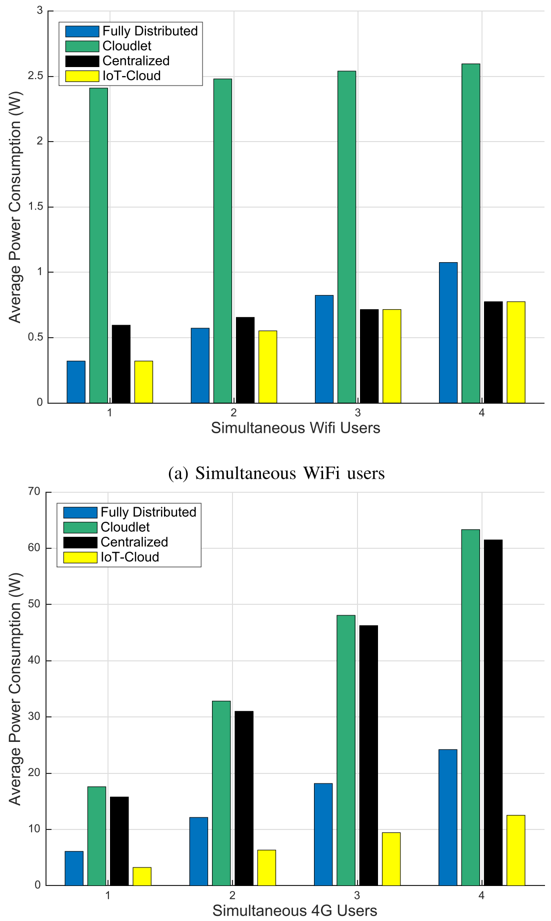


Fig. 6. Average power consumption per user of a city monitoring application for different personalized information rates.

devices have a processing efficiency of 1000 MIPS/W. The processing complexity of analyzing the data from each WSN is 500 instructions per bit, while the complexity of generating the personalized information requested by the user is 5000 instructions per bit. An average rate of 1 packet per second is generated by each wireless sensor. We consider the same number of WiFi and 4G users, and an average number of 4 users simultaneously requesting the same information in total. Note that the sensor readings can be sent via multicast, since they can be used to generate different final information objects. However, personalized information can be sent via multicast only when multiple users request the same information at the same time. Otherwise, these must be sent via unicast. Nevertheless, we assume that the transmission from the base stations to the end users is always unicast, as neither WiFi nor 4G multicast technologies are yet commercially available. As we can observe, if users request a low bitrate the IoT-Cloud solution tends to consolidate the processing functionalities at the HO in order to reduce the total processing cost. As a result, the overall power consumption is reduced by more than 84% compared to the fully distributed solution. However, the centralized cloud approach becomes less efficient at high bitrates due to the transport cost from the cloud nodes to the smart devices, particularly to the 4G devices. Then, the IoT-Cloud solution pushes some processing functionalities closer to the users in order to reduce transport costs, at the expense of reducing the consolidation gain. In particular, the IoT-Cloud solution moves the function that generates the personalized data to the end devices for bitrates above 20 kbps, while keeping the sensor data analysis at the cloud layer. We can observe that the overall power consumption of the centralized and cloudlet solutions are severely affected by the requested bitrate, being their cost more than 78% higher than the cost of the IoT-Cloud solution if the bitrate exceeds 200 kbps.

In Fig. 7, we compare the power consumption for different average number of users requesting simultaneously the same information. In this case, we separate the power consumption of WiFi and 4G users to observe the impact of simultaneous users in each case. The bitrate of the city information is assumed to be 200 kbps and the processing efficiency of the smart devices is 1000 MIPS/W. In Fig. 7a, we observe that if there are more than 2 simultaneous WiFi users, it is more energy efficient to consolidate the processing tasks at the HO than to process them locally, even if this increases the total transport cost. Note that if the information requested is generated at the HO, the network can take more advantage of multicast delivery. However, for users accessing the network using 4G (see Fig. 7b), the additional transport cost of 4G networks cannot be compensated even with 4 simultaneous 4G users, being the overall cost of the centralized solution 80% higher than that of the IoT-Cloud solution. Nevertheless, it is important to note that if wireless multicast streaming schemes were considered, the impact of processing consolidation would be more relevant in both cases. We again observe the higher power consumption of the cloudlet solution due to its low processing efficiency. This is particularly relevant in Fig. 7a due to the higher significance of processing costs compared to transport costs in WiFi networks.

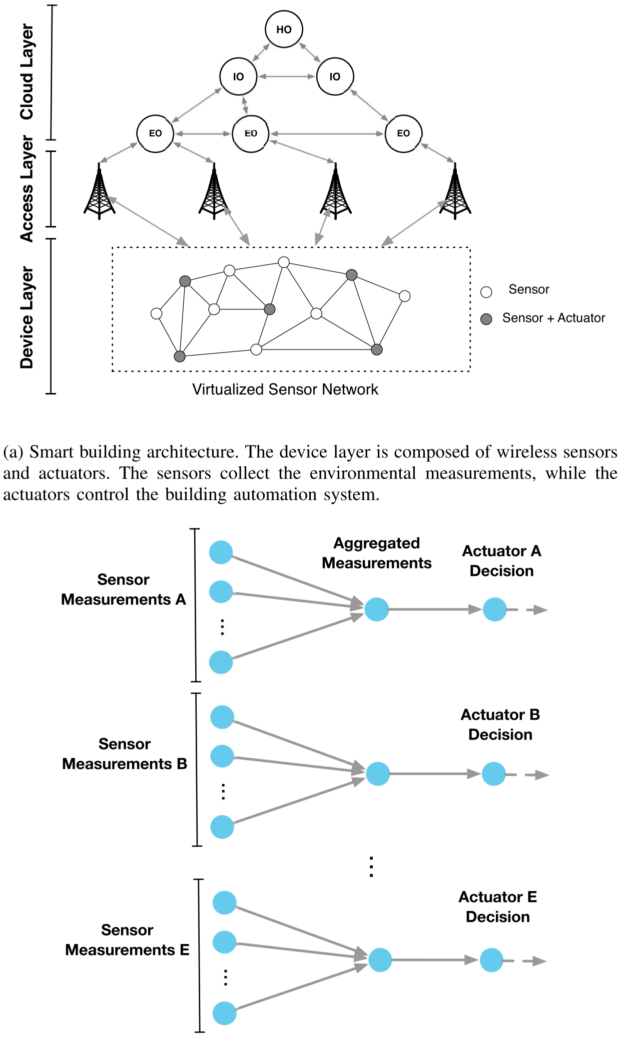
We remark that the effect of the number of users is specially relevant under the presence of multicast flows. Note that if all flows are unicast, then, under the adopted linear cost model, the total power consumption would be proportional to the total number of users, since each transport/processing



(a) Simultaneous WiFi users

(b) Simultaneous 4G users

Fig. 7. Average power consumption per user of a city monitoring service for different number of simultaneous users.



(a) Smart building architecture. The device layer is composed of wireless sensors and actuators. The sensors collect the environmental measurements, while the actuators control the building automation system.

(b) Service graph of the smart building application. The sensor readings are first aggregated, to reduce the transport flow, and then analyzed to control the building automation system using the wireless actuators. Solid and dashed arrows state for multicast and unicast flows, respectively.

Fig. 8. Smart building environment

function needs to be conducted individually for each end user.

C. Smart Building

In this section, we consider an IoT service that controls a building automation system. A WSN that collects 5 different types of measurements (e.g., humidity, temperature, light, smoke, occupancy) is assisted by the Cloud to manage 5 actuators distributed across the building (See Fig. 8(a)). According to the sensor readings, and after a simple data analysis, the actuators manage different autonomous systems. In terms of processing complexity, both data aggregation and data analysis require 50 instructions per bit. We also assume that the measurements from wireless sensors can be combined using data aggregation techniques as long as they contain the same type of information. This reduces traffic by a compression rate that depends on the spatial and temporal correlation of data. A realistic compression rate of 50% [46] is assumed in these simulations. The sensor readings may be sent via multicast, but the actuator decisions are sent via unicast. However, note that in this case the distribution mode does not have any impact in the total traffic flow, since every object has only one possible destination, and hence there are no multicast opportunities. The service graph associated with this service is shown in Fig.8(b).

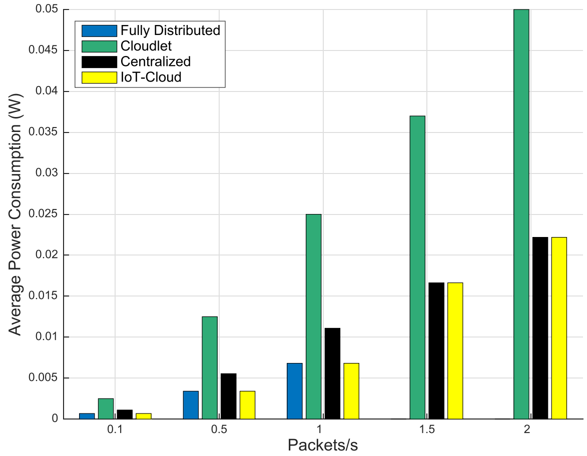
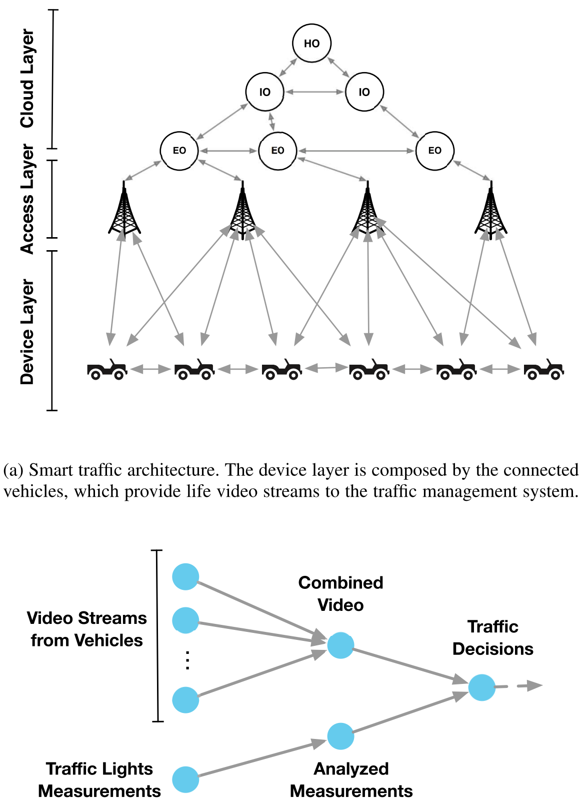


Fig. 9. Average power consumption of an autonomous sensing actuation service for different packet generation rates. Note that missing values indicate that the solution is infeasible.

In order to observe the impact of the limited battery capacity of wireless actuators, we constrain their average power consumption according to their expected battery lifetime. Note that battery replacement is a costly operation and therefore nodes should not deplete their batteries before their scheduled maintenance. With this in mind, we only allow solutions that assure a minimum lifetime of one year for the wireless actuators. Assuming that each actuator is equipped with a battery storing 27 kJ [47], this translates into a maximum power of $E_u = 856\mu$W.

In Fig. 9, we compare the power consumption for different packet generation rates at the wireless sensors. A high bitrate increases the accuracy of the service, but it also increases the power consumption of the wireless sensors and actuators. Note that when the average power consumption of a given solution is not shown, it indicates that such solution is infeasible, i.e., it violates QoS or capacity constraints. In scenarios that require more than 2 packets/s from each sensor, the processing tasks are fully offloaded to the Cloud to reduce the power consumption of the wireless actuators. However, if the packet rate can be reduced, the limited processing capacity of actuators can be used to reduce the traffic sent to the Cloud. As we can observe, the device layer can control the actuators without the assistance of the cloud infrastructure as long as the packet rate is lower than 1.5 packets/s. As a result, the total transport consumption is reduced. However, the processing tasks have to be offloaded to the Cloud at higher rates in order to satisfy the minimum lifetime requirements of the wireless actuators. Therefore, in this case, the IoT-Cloud solution corresponds to the centralized solution. It is worth noting that the optimized IoT-Cloud solution only uses the cloud resources when they are strictly necessary.



(a) Smart traffic architecture. The device layer is composed by the connected vehicles, which provide life video streams to the traffic management system.

(b) Service graph of the smart mobility application. The measurements collected at the traffic lights are combined with the video streams of vehicles. Solid and dashed arrows indicate multicast and unicast flows, respectively.

Fig. 10. Smart mobility environment

D. Smart Mobility

In this section, we consider a service that interconnects city vehicles among each other, as well as with city traffic lights. This enables optimizing city traffic, reducing congestion, and increasing road safety. We assume the scenario in Fig. 10(a), in which vehicles are connected to other nearby vehicles and to their closest base stations. In addition, each base station has 5 traffic lights connected to it, each generating 127 bytes per second of traffic information. The vehicles generate video streams of 200 kbps. The streams from nearby vehicles are combined and compressed using a scaling factor of 0.5. The measurements from the traffic lights and the video streams are then jointly analyzed to predict traffic jams and manage traffic lights accordingly. In particular, we consider the service graph shown in Fig. 10(b). In terms of processing complexity, we assume that processing the traffic measurements requires 50 instructions per bit, the combination of the video streams 2000 instructions per bit, and the generation of the traffic decisions requires 5000 instructions per bit. We further assume that vehicles' embedded computing resources have a processing efficiency of 1000 MIPS/W.

In Fig. 11, we show the average power consumption for different latency requirements. Low end-to-end service latencies are specially relevant for smart mobility services in which processed information is used to coordinate city traffic in real-time. It is important to remark that there exists a minimum end-to-end service latency dictated by the network topology and the adopted solution approach. As shown in Fig. 10(a), source streams need to be sent at least to the base stations (5 ms), which control the traffic lights, then forwarded to be jointly processed (i.e., at least 4 ms), and finally output decisions must be sent back to the base stations (i.e., at least 2 ms) that control the traffic lights. Hence, the minimum achievable transport delay is 11 ms. In addition, we need to consider the processing delay, which is at least 2 ms if the processing tasks are placed at a cloud location (i.e., 1 ms for each processing function). Note that this minimum end-to-end latency can only be achieved with the IoT-Cloud and cloudlet solutions. A fully distributed solution can only achieve a minimum latency of 28 ms (i.e., 20 ms for transport and 8 ms for processing). Finally, the centralized cloud solution has a minimum latency of 55 ms (i.e., 53 ms for transport and 2 ms for processing). This can be observed in Fig. 11, where maximum latency requirements below 28 ms can only be satisfied by the optimized IoT-Cloud solution. It is worth noting that the IoT-Cloud optimal solution does not need to place the virtual functions at the access layer to achieve its minimum latency. Since the information processed at the micro cloud nodes needs to go through the cloud layer for sharing purposes, placing the virtual functions at the edge of the cloud layer (EOs) reduces the processing cost without increasing the end-to-end latency.

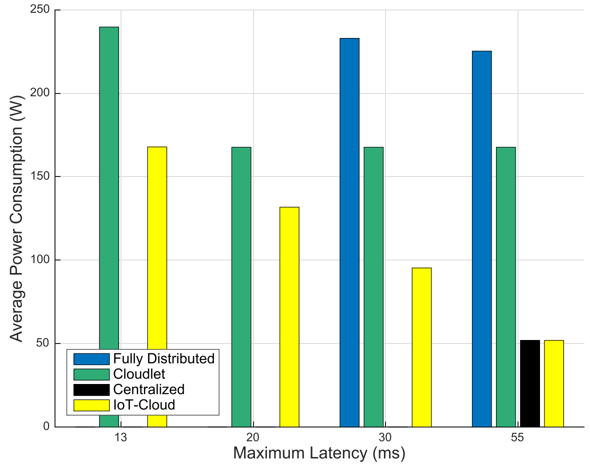


Fig. 11. Average power consumption of a traffic management service for different latency requirements. Note that missing values indicate that the solution is infeasible.

The results presented in this paper clearly show the benefits that arise from the flexibility of an optimized IoT-Cloud solution and the limitations of current centralized, fully distributed, and cloudlet solutions for the efficient delivery of next generation IoT services. On one hand, centralizing all processing tasks at a single core-level data center is shown to be efficient in terms of processing costs, while incurring transport costs and end-to-end latencies that become unsustainable for increasingly bandwidth-intensive low-latency IoT services. On the other hand, placing all processing functionality at the device layer, while appropriate for low complexity operations, can seriously compromise the desirable lightweight, long battery life properties of end devices. An intermediate solution, such as placing the processing tasks at the access layer, improves service latencies compared to a centralized solution, but it also reduces overall processing efficiency due to the lower computing capabilities of micro cloud nodes. In contrast, we have shown that an optimized IoT-Cloud solution overcomes these limitations by placing service functions at different layers of the IoT-Cloud infrastructure as a function of the input parameters, striking the right balance between processing efficiency, transport efficiency, and end-to-end latency.

VII. Conclusions

The confluence of distributed cloud networking and the Internet of Things (IoT) enables a new class of services that create augmented information from the cloud analysis of IoT data. IoT-Cloud networks reduce the distance between end users and cloud resources using edge cloud nodes distributed across the network, in order to support the key low latency, mobility, and location-awareness requirements of IoT services. We proposed a comprehensive optimization framework to enhance the efficiency of IoT services in next generation smart environments. We formulated the service distribution problem (SDP) in IoT-Cloud networks (IoT-CSDP) as a min-cost mixed-cast flow problem. The solution to the IoT-CSDP determines the optimal placement of service functions and the routing of network flows, taking into account the heterogeneous capacities and efficiencies of sensing, computing, and transport resources across the distributed IoT-Cloud infrastructure. We solved the IoT-CSDP for the optimization of IoT services in multiple smart environments. Results show that the IoT-CSDP solution captures the critical tradeoffs that appear in IoT-Cloud platforms due to the heterogeneity of IoT services, cloud network technologies, and end user devices. When compared to current solutions, smart IoT services optimized over a fully virtualized IoT-Cloud platform are shown to guarantee stringent QoS requirements in terms of reliability, battery lifetime, and end-to-end latency, while reducing overall power consumption by more than 80%.

Motivated by these promising results, interesting directions for future work include implementing the proposed solution in a cloud computing testbed, designing approximation algorithms of improved computational complexity, and studying the benefit of distributed optimization techniques to enable local reactions to fast system dynamics.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.

[2] 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, 4th Quart., 2015.

[3] C. Zhu, V. C. M. Leung, L. Shu, and E. C.-H. Ngai, "Green Internet of Things for smart world," *IEEE Access*, vol. 3, pp. 2151–2162, 2015.

[4] N. Alhakbani, M. M. Hassan, M. A. Hossain, and M. Alnuem, "A framework of adaptive interaction support in cloud-based Internet of Things (IoT) environment," in *Proc. 7th Int. Conf. Internet Distrib. Comput. Syst. (IDCS*, Calabria, Italy, Sep. 2014, pp. 136–146.

[5] K. Su, J. Li, and H. Fu, "Smart city and the applications," in *Proc. ICECC*, Sep. 2011, pp. 1028–1031.

[6] C. Perera, C. H. Liu, and S. Jayawardena, "The emerging Internet of Things marketplace from an industrial perspective: A survey," *IEEE Trans. Emerg. Topics Comput.*, vol. 3, no. 4, pp. 585–598, Dec. 2015.

[7] S. Sarkar, S. Chatterjee, and S. Misra, "Assessment of the suitability of fog computing in the context of Internet of Things," *IEEE Trans. Cloud Comput.*, Oct. 2015.

[8] D. Bouley, "Estimating a data center's electrical carbon footprint," Schneider Electric, USA, White Paper 66, 2015.

[9] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for VM-based cloudlets in mobile computing," *IEEE Pervasive Comput.*, vol. 8, no. 4, pp. 14–23, Oct./Dec. 2009.

[10] W. Shiqiang *et al.*, "Mobile micro-cloud: Application classification, mapping, and deployment," in *Proc. Annu. Fall Meeting ITA (AMITA)*, New York, NY, USA, Oct. 2013.

[11] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the Internet of Things," in *Proc. 1st Ed. MCC Workshop Mobile Cloud Comput.*, New York, NY, USA, 2012, pp. 13–16.

[12] H. Chang, A. Hari, S. Mukherjee, and T. Lakshman, "Bringing the cloud to the edge," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Apr./May 2014, pp. 346–351.

[13] B. Addis, D. Ardagna, B. Panicucci, M. S. Squillante, and L. Zhang, "A hierarchical approach for the resource management of very large cloud platforms," *IEEE Trans. Depend. Sec. Comput.*, vol. 10, no. 5, pp. 253–272, Sep. 2013.

[14] L. Lei, Z. Zhong, K. Zheng, J. Chen, and H. Meng, "Challenges on wireless heterogeneous networks for mobile cloud computing," *IEEE Wireless Commun.*, vol. 20, no. 3, pp. 34–44, Jun. 2013.

[15] S. Nastic, S. Sehic, D. H. Le, H. L. Truong, and S. Dustdar, "Provisioning software-defined IoT cloud systems," in *Proc. Int. Conf. Future Internet Things Cloud (FiCloud)*, Aug. 2014, pp. 288–295.

[16] M. Barcelo, J. Llorca, A. M. Tulino, and N. Raman, "The cloud service distribution problem in distributed cloud networks," in *Proc. IEEE ICC*, London, U.K., Jun. 2015, pp. 344–350.

[17] I. Stojmenovic and S. Wen, "The Fog computing paradigm: Scenarios and security issues," in *Proc. Fed. Conf. Comput. Sci. Inf. Syst. (FedCSIS)*, Sep. 2014, pp. 1–8.

[18] A. Alamri, W. S. Ansari, M. M. Hassan, M. S. Hossain, A. Alelaiwi, and M. A. Hossain, "A survey on sensor-cloud: Architecture, applications, and approaches," *Int. J. Distrib. Sensor Netw.*, vol. 2013, Feb. 2013, Art. no. 917923.

[19] S. Misra, S. Chatterjee, and M. S. Obaidat, "On theoretical modeling of sensor cloud: A paradigm shift from wireless sensor network," *IEEE Syst. J.*, Nov. 2014.

[20] N. Mitton, S. Papavassiliou, A. Puliafito, and K. S. Trivedi, "Combining cloud and sensors in a smart city environment," *EURASIP J. Wireless Commun. Netw.*, vol. 2012, no. 1, pp. 247–257, Aug. 2012.

[21] M. S. Aslam, S. Rea, and D. Pesch, "Service provisioning for the WSN cloud," in *Proc. IEEE 5th Int. Conf. Cloud Comput. (CLOUD)*, Jun. 2012, pp. 962–969.

[22] C. Zhu, H. Wang, X. Liu, L. Shu, L. T. Yang, and V. C. M. Leung, "A novel sensory data processing framework to integrate sensor networks with mobile cloud," *IEEE Syst. J.*, vol. 10, no. 3, pp. 1125–1136, Sep. 2016.

[23] C. Zhu, Z. Sheng, V. C. M. Leung, L. Shu, and L. T. Yang, "Toward offering more useful data reliably to mobile cloud from wireless sensor network," *IEEE Trans. Emerg. Topics Comput.*, vol. 3, no. 1, pp. 84–94, Mar. 2015.

[24] C. Zhu, H. Nicanfar, V. C. M. Leung, and L. T. Yang, "An authenticated trust and reputation calculation and management system for cloud and sensor networks integration," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 1, pp. 118–131, Jan. 2015.

[25] C. Zhu, V. C. M. Leung, L. T. Yang, X. Hu, and L. Shu, "Collaborative location-based sleep scheduling to integrate wireless sensor networks with mobile cloud computing," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2013, pp. 452–457.

[26] C. H. Liu, J. Fan, J. W. Branch, and K. K. Leung, "Toward QoI and energy-efficiency in Internet-of-Things sensory environments," *IEEE Trans. Emerg. Topics Comput.*, vol. 2, no. 4, pp. 473–487, Dec. 2014.

[27] A. Mukherjee, H. S. Paul, S. Dey, and A. Banerjee, "ANGELS for distributed analytics in IoT," in *Proc. IEEE World Forum Int. Things (WF-IoT)*, Mar. 2014, pp. 565–570.

[28] N. B. Truong, G. M. Lee, and Y. Ghamri-Doudane, "Software defined networking-based vehicular adhoc network with fog computing," in *Proc. IFIP/IEEE Int. Symp. Integr. Netw. Manage. (IM)*, May 2015, pp. 1202–1207.

[29] S. Sarkar and S. Misra, "Theoretical modelling of fog computing: A green computing paradigm to support IoT applications," *IET Netw.*, vol. 5, no. 2, pp. 23–29, 2016.

[30] I. Farris, L. Militano, M. Nitti, L. Atzori, and A. Iera, "Federated edge-assisted mobile clouds for service provisioning in heterogeneous IoT environments," in *Proc. IEEE 2nd World Forum Internet Things (WF-IoT)*, Dec. 2015, pp. 591–596.

[31] M. Aazam and E.-N. Huh, "Fog computing micro datacenter based dynamic resource estimation and pricing model for IoT," in *Proc. IEEE 29th Int. Conf. Adv. Inf. Netwo. Appl. (AINA)*, Mar. 2015, pp. 687–694.

[32] M. Aazam, M. St-Hilaire, C.-H. Lung, and I. Lambadaris, "PRE-Fog: IoT trace based probabilistic resource estimation at fog," in *Proc. 13th IEEE Annu. Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2016, pp. 12–17.

[33] J. Bungo, "Embedded systems programming in the cloud: A novel approach for academia," *IEEE Potentials*, vol. 30, no. 1, pp. 17–23, Jan./Feb. 2011.

[34] "The programmable cloud network—A primer on SDN and NFV," Bell Labs Strategic, New Jersey, USA, White Paper, Jun. 2013.

[35] J. Llorca and A. Tulino, "The content distribution problem and its complexity classification," Alcatel-Lucent, Boulogne-Billancourt, France, Tech. Rep., 2013.

[36] M. Satyanarayanan *et al.*, "An open ecosystem for mobile-cloud convergence," *IEEE Commun. Mag.*, vol. 53, no. 3, pp. 63–70, Mar. 2015.

[37] A. Vishwanath, F. Jalali, K. Hinton, T. Alpcan, R. W. A. Ayre, and R. S. Tucker, "Energy consumption comparison of interactive cloud-based and local applications," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 4, pp. 616–626, Apr. 2015.

[38] M. Ilyas and I. Mahgoub, *Smart Dust: Sensor Network Applications, Architecture and Design*. Boca Raton, FL, USA: CRC Press, 2006.

[39] P. Zheng and L. Ni, *Smart Phone and Next Generation Mobile Computing*. San Mateo, CA, USA: Morgan Kaufmann, 2010.

[40] *Qualcomm Automotive Infotainment*. accessed on Jan. 2016. [Online]. Available: https://www.qualcomm.com/products/automotive/infotainment

[41] P. Kyösti, J. Meinilä, L. Hentilä, X. Zhao, and T. Jämsä, "WINNER II channel models," EBITG, TUI, UOULU, CU/CRC, NOKIA, Finland, Tech. Rep. IST-4-027756, 2007.

[42] V. G. Guimaraes, A. Bauchspiess, and R. M. D. Moraes, "Dynamic timed energy efficient and data collision free MAC protocol for wireless sensor networks," *IEEE Latin Amer. Trans.*, vol. 13, no. 2, pp. 416–421, Feb. 2015.

[43] A. Martínez-Sala, J.-M. Molina-Garcia-Pardo, E. Egea-López, J. Vales-Alonso, L. Juan-Llacer, and J. García-Haro, "An accurate radio channel model for wireless sensor networks simulation," *J. Commun. Netw.*, vol. 7, no. 4, pp. 401–407, Dec. 2005.

[44] *AT86RF230: Low Power 2.4 GHz Transceiver for ZigBee*, IEEE Standard 802.15.4, 6LoWPAN, RF4CE, and ISM Applications, Atmel, 2009.

[45] C. Csiszar and D. Foldes, "Analysis and modelling methods of urban integrated information system of transportation," in *Proc. Smart Cities Symp. Prague (SCSP)*, Jun. 2015, pp. 1–10.

[46] C. Cappiello and F. A. Schreiber, "Experiments and analysis of quality- and energy-aware data aggregation approaches in WSNs," in *Proc. 10th Int. Workshop Quality Databases*, Aug. 2012, pp. 1–8.

[47] A. Sivagami, K. Pavai, D. Sridharan, and S. A. V. S. Murty, "Estimating the energy consumption of wireless sensor node: Iris," *Int. J. Recent Trends Eng. Technol.*, vol. 3, no. 4, pp. 141–143, May 2010.

Marc Barcelo received the M.Sc. degree (Hons.) in electrical engineering and the Ph.D. (*cum laude*) degree from the Universitat Autònoma de Barcelona (UAB) in 2010 and 2015, respectively. From 2009 to 2015, he was with the UAB's Telecommunications and Systems Engineering Department, and where he was involved in several international research and development projects, such as ADIBEAM and SINTONIA. In 2014, he was an International Research Intern with Nokia Bell Labs, Holmdel, NJ, USA. He joined Ikerlan, Mondragón, Spain, in 2016, where he is currently a Research Scientist with the Cybersecure IoT Department. He has authored over 20 papers in recognized international journals and conferences. Currently, his research efforts are focused on the design of efficient communication architectures for the Internet of Things. In 2011, he received the COIT/AEIT Prize for his M.Sc. thesis.

Alejandro Correa received the B.Sc. and M.Sc. degrees (Hons.) in electrical engineering from Universitat Autònoma de Barcelona (UAB) in 2008 and 2010, respectively, where he is currently pursuing the Ph.D. degree with the Telecommunications and Systems Engineering Department, where he has been involved in several research and development projects, such as ATLANTIDA and 3SENS. He has authored eight international journals and seven international conferences. His main research interests include pedestrian indoor positioning systems and wireless sensor networks.

Jaime Llorca received the B.E. degree in electrical engineering from the Universitat Politècnica de Catalunya, Barcelona, Spain, in 2001, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Maryland, College Park, MD, USA, in 2003 and 2008, respectively. He held a post-doctoral position with the Center for Networking of Infrastructure Sensors, University of Maryland, from 2008 to 2010. He joined Nokia Bell Labs, Holmdel, NJ, USA, in 2010, where he is currently a Research Scientist with the Network Algorithms Department. His research interests include energy efficient networks, distributed cloud networking, content distribution, resource allocation, network information theory, and network optimization. He was a recipient of the 2007 Best Paper Award at the IEEE International Conference on Sensors, Sensor Networks, and Information Processing, the 2016 Best Paper Award at the IEEE International Conference on Communications, and the 2015 Jimmy H.C. Lin Award for Innovation.

Antonia M. Tulino (S'00–M'03–SM'05–F'13) received the Ph.D. degree in electrical engineering from Seconda Università degli Studi di Napoli, Italy, in 1999. She held research positions with Princeton University, the Center for Wireless Communications, Oulu, Finland, and the Università degli Studi del Sannio, Benevento, Italy. In 2002, she joined the Faculty of the Università degli Studi di Napoli Federico II. In 2009, she joined Bell Labs. She has been a Principal Investigator of several research projects sponsored by the European Union and the Italian National Council. Her research interests lie in the area of communication systems approached with the complementary tools provided by signal processing, information theory, and random matrix theory. Since 2011, she has been a member of the Editorial Board of the IEEE TRANSACTIONS ON INFORMATION THEORY. She has received several paper awards and the 2009 Stephen O. Rice Prize in communications theory for the best paper published in the IEEE TRANSACTION ON COMMUNICATION in 2008. She was selected by the National Academy of Engineering for the Frontiers of Engineering Program in 2013.

Jose Lopez Vicario received the degree in electrical engineering and the Ph.D. (*cum laude*) degree from the Universitat Politècnica de Catalunya, Barcelona, in 2002 and 2006, respectively, and the M.B.A. degree from the IESE Business School, Universidad de Navarra. Since 2006, he has been an Associate Professor with the Universitat Autonòma de Barcelona, where he has been involved in digital communications and signal processing and several research and development projects, such as IST FP6 SATNEX, ESA DINGPOS, ESA ADIBEAM, CENIT ATLANTIDA, CENIT SINTONIA (as PI), PROFIT INTERRURAL (as PI), INFOREGIO XALOC (as PI), AVANZA DESSO (as PI), and RECERCAIXA 3SENS. He has wide expertise on statistical signal processing. He has authored 24 papers in recognized international journals and around 50 papers in conferences. Currently, his research efforts are focused on the development of data mining algorithms covering different areas, such as wireless sensor networks, twitter analysis, financial applications, and air traffic management.

Antoni Morell received the M.Sc. degree in electrical and electronic engineering and the Ph.D. degree from the Universitat Politècnica de Catalunya (UPC), Barcelona, in 2002 and 2008, respectively. He was with the Signal Theory and Communications Department, UPC, from 2002 to 2005. He joined Universitat Autònoma de Barcelona in 2005, where he is currently an Associate Professor. He teaches courses on communications, signals, and systems. He has been involved in over ten research and development projects, national and international, being the Principal Investigator in two of them. He has expertise in optimization techniques applied to communications, also in the field of wireless sensor networks. He has authored over 40 papers in recognized international journals and conference proceedings. His current research interests include smart data applications of machine learning and inertial-aided indoor positioning, and routing/access strategies for WSNs.