Received March 2, 2020, accepted March 17, 2020, date of publication March 20, 2020, date of current version March 31, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2982241

Network-Aware Placement Optimization for Edge Computing Infrastructure Under 5G

ALEJANDRO SANTOYO-GONZÁLEZ AND CRISTINA CERVELLÓ-PASTOR
Department of Network Engineering, Universitat Politècnica de Catalunya (UPC), 08860 Barcelona, Spain

Corresponding author: Alejandro Santoyo-González (alejandro.santoyo@entel.upc.edu)

This work was supported by the Ministerio de Economía y Competitividad of the Spanish Government under Project TEC2016-76795-C6-1-R and Project AEI/FEDER, UE.

ABSTRACT Edge Computing has grown into a key solution for coping with the stringent latency requirements of 5G scenarios. Nevertheless, the edge node placement problem raises critical concerns regarding deployment and operational expenditures (i.e., mainly due to the number of nodes to be deployed), current backhaul network capabilities, non-technical placement limitations, etc. In this paper, a novel framework called **EdgeON** is presented aiming at reducing the overall expenses when deploying and operating an Edge Computing (EC) network, taking into account the usage and characteristics of the in-place backhaul network. The framework implements several placement and optimization strategies targeting the heavily constrained network-aware Edge Node Placement Problem (ENPP). The results obtained by our solution are promising, achieving an average of 30% less Edge Nodes (ENs) deployed and 25% higher average usage ratio when compared to other widely used heuristics. Furthermore, our strategy achieved a score offset of less than 2% in comparison to the implemented Mixed Integer Linear Programming (MILP).

INDEX TERMS 5G, edge computing, NFV, MILP, optimization problems.

I. INTRODUCTION

Under 5G networking, a user-centered ecosystem providing seamless integration between users and devices is to be achieved. Such ecosystem, based on smart interconnection, artificial intelligence-based systems and automated self-aware orchestration, comes at the cost of stringent technical requirements. 5G use cases such as Enhanced Mobile Broadband (eMBB) and Massive Machine-type Communications (mMTC) will severely stretch the limits of current networking platforms as nearly 1 million interconnected devices per squared kilometer are to be supported [1]. This poses complex challenges regarding radio resource allocation, data transmission, routing/processing and Quality of Service (QoS) delivery [1], [2]. Smart Cities and e-Health deployments entail strict data rate demands, whereas Autonomous Driving and Industry Monitoring will require nearly 100% reliability and millisecond-level latency [3]. In addition to severe QoS and Quality of Experience (QoE) needs, 5G is required to enforce extremely high security and privacy for e-Banking, Security Monitoring, Traffic Safety and Mobile Health [4]. Moreover, overall power consumption is to be drastically reduced to ensure long-time battery life and *green* networking [3].

In this context, the remote datacenter model has become inefficient and unable to cope with the rising technical demands. By providing an end-to-end communication delay of around 60 to 100 ms, current remote clouds are unable to guarantee the required 1 ms round-trip latency and stable jitter for delay-sensitive and location-aware use cases [2], [5], [6]. Privacy and security concerns are additionally stretching the cloud capabilities as the use of applications working over distributed platforms increases (e.g., blockchain-based systems, multimedia delivery systems, etc.). Scalability has additionally grown into a critical concern given the massive amounts of data to be processed [1]. For instance, deep data analysis mechanisms to accurately segment and generate maximum value from each customer are causing critical bottlenecks in the data transmission systems.

Edge Computing (EC) has become a solid alternative to the traditional remote datacenter-based service scheme. By bringing computing, storage and networking resources to the users' vicinity, EC aims at reducing the physical and logical distance between hosts and end-users, while satisfying

The associate editor coordinating the review of this manuscript and approving it for publication was Fuhui Zhou.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

the requirements of distributed resource-intensive applications and delay-sensitive ecosystems. Concretely, the authors in [7]–[10] claim that EC is able to effectively reduce around 20% of the average response time and 90% of the north-south traffic when compared to a remote cloud service architecture, while significantly improving scalability. However, a distributed set of ENs raises critical concerns regarding Capital Expenditures (CAPEX) and Operational Expenditures (OPEX), deployment strategies, QoS and QoE [11]–[13]. On the other hand, cost-effectively deploying an EN network is extremely challenging due to the numerous tradeoffs involved. For instance, placing as few as possible high-capacity ENs leverages economy of scale but can lead to performance degradation or unmet demands, while increasing the number of ENs directly increases the overall CAPEX/OPEX of the network [12], [14]. Moreover, deploying the EC service infrastructure at the Radio Access Network (RAN) nodes, following the Mobile Edge Computing (MEC) approach, is often seen as the best solution for the EN site selection problem. Nevertheless, this is commonly ineffective or impractical since: 1) Base Stations (BSs) are typically placed at unfeasible locations with very limited physical equipment space (e.g., macro-cell towers located at the top of a remote hill, small-cells placed at street cabinets, etc.) and, 2) placing an EN at each RAN site will significantly increase the overall network costs due to oversizing [15], [16].

Under these circumstances, the EN placement strategy becomes crucial. Since 5G ultra-dense networking will drastically change the placement of mobile BSs, cache servers, etc., and thousands of ENs are to be deployed, the economical feasibility of the 5G/EC ecosystem is tied to the efficiency of the capacity planning and deployment strategies, i.e., the EN placement methods.

By optimizing the EN placement, the deployment/operation cost savings can be significantly increased and user requirement satisfaction can be effectively enforced. However, most capacity planning studies assume that the service infrastructure has been already deployed focusing on the resource allocation and capacity problem, thus overlooking the need to optimize the location selection procedures [17], [18]. Extensive research has been found regarding problems closely related to the EN placement optimization: Facility Location Problems (FLPs), datacenter, BS and generic server placement (e.g., cache servers) [19]–[28]. Additionally, few articles were found targeting the edge server placement problem [17], [29]–[31]. Several limitations prohibit the use of these studies to effectively place an EN network under 5G constraints.

FLP solutions, for instance, cannot be directly applied for the EC infrastructure deployment due to typical cost function simplicity, traditional convergence into a specific operational problem (e.g., Weber, coverage, etc.) and lack of non-technical restrictions analysis [25]. Datacenter and generic server placement strategies overlook the need for a shared and geographically distributed infrastructure where the member nodes must cooperatively solve offloaded tasks while maintaining minimum latency levels. In addition, the lack of flexibility forces these models to be discarded when applied to the ultra-dense networking demands of 5G networking [28], [32]. BS placement is mostly done based on tessellation and clustering methods that may not be suitable for 5G traffic patterns and service trends under ultra-dense 5G networks [19], [20]. Finally, the edge server placement solutions found have not been tailored to 5G requirements, while covering a limited set of specific scenarios, overlooking the underlying network capacity constraints and over-simplifying the user demand distribution through traditional clustering approaches [17], [29], [30].

Given the aforementioned context, the **main goal** of this work is to propose a platform to solve the network-aware multi-objective Edge Node Placement Problem (ENPP) under strict 5G service constraints. To this aim, our core **contributions** can be summarized as follows:

1) A rigorous definition of the network-aware multi-objective ENPP through a MILP mathematical model.
2) A novel network-aware framework called "**EdgeON**" for telcos and operators to adapt to their particular needs and use cases when planning the deployment of an EC network under custom service requirements.
3) A thorough evaluation of the framework's core heuristic to solve the ENPP by comparing it to: a) widely used algorithms addressing multi-objective placement problems and, b) a MEC deployment approach.

To achieve our goals, the remaining of this paper is structured as follows: Section II presents a review of current literature and Section III mathematically defines the ENPP. Section IV showcases the architecture of the proposed framework whereas Section V presents the evaluation and results. Finally, the conclusions of the paper are explained in Section VI.

II. RELATED WORK AND MOTIVATION

The ENPP can be directly linked to other infrastructure placement problems such as FLPs, mobile BS and cache-enabled node placement, etc. However, there are some important differences that stand out after a detailed analysis of these problem types within the 5G context.

The articles in [28] and [32] are two of the few publicly available papers referred to the datacenter placement optimization problem. On the former, key insights are provided regarding the physical aspects of service infrastructure placement such as energy consumption, build and land costs, etc. The limitations of this work when directly applied to our particular problem are: its inability to deal with the exponential number of nodes to deploy in a small to medium-sized 5G service area and the communication restrictions between users and services as result of the latency constraints forcing the formulation of a "*coverage*" problem. The work in [32], aims to place all components of a fog network based on micro datacenters and a Long-Reach Passive Optical Network (LRPON). In this case, the limited scope of the formulation is a key restriction for the ENPP solution, as the interconnecting methods under 5G are expected to be significantly diverse, whereas in this case the whole formulation depends on the particularities of the LRPON and its components.

BS placement strategies, as in [19], [20], [33], [34], commonly use tessellation methods to select the optimal site for mobile BS. Furthermore, their typical concerns are mostly related to radio resource allocation problems (e.g., interference, etc.) and rigorous traffic density modeling. Further investigation in the field of mobile networking has attempted to optimize the location selection of the remaining network components. An example can be found in [35], where novel metrics are proposed for the placement of the Serving Gateways and Packet Data Network Gateways. In summary, the proposal adds new metrics such as the end-to-end connection and service/application types to the process of selecting the most suitable data anchor gateway for a given host-to-host communication.

The MEC paradigm, Enhanced Small Cells and other concepts and platforms such as the proposed in [36]–[38], significantly differ in their deployment location considerations. While some solutions (e.g., Small Cell Clouds and Mobile Cell Clouds) assume to place the computation capacities within the RAN sites, others encourage a further away placement of the resources at centralized datacenters but introducing new components and inter-working procedures to guarantee better performance.

Very few articles are available on EC infrastructure placement, most likely due to the youth of the related technologies and the lack of operational deployments. Furthermore, the papers found throughout this research mainly cover quite specific scenarios. Therefore, a broad view of the problem with a more general solution method remains an open question.

The QoS-aware placement of Fog Computing (FC) nodes is solved in [39] based on the "*k-means*" algorithm to find the best network gateways to place the fog nodes such that the overall latency is minimized. This work lacks a flexible capacitated formulation and assumes that each node transmit data to only one fog node, thus reducing the applicability of the solution to real-life scenarios. The authors in [17] present two core problems: 1) the minimization of the number of Access Points (APs) co-located with an arbitrary edge server to guarantee customer demand satisfaction and, 2) the efficient task assignment to the edge servers. Graph theory is used to transform the presented problem into the minimum dominating set problem and a solution based on a greedy and Simulated Annealing (SA) algorithms is developed to find the near-optimal solutions. When compared to the ENPP, this paper is limited by the translation of the delay constraint to a simplistic Euclidean distance-based model and the use of a clustering approach, thus lacking the adaptability required to deal with the EN placement under 5G requirements.

Similarly, a framework to solve the edge server placement within a geographical topology is showcased in [30]. As in [40], this work uses a clustering approach in order to simplify the overall problem complexity, thus incurring in the above mentioned limitations. In [29] the edge server placement problem is tackled for mobile edge computing environments in future smart cities. The novelty of this study lies in the multi-objective optimization model, aiming at both delay minimization and overall workload balance. This work assumes to know in advance the number of edge servers to be placed and uses the distance to estimate the network delay, thus limiting the applicability of the results.

Overall, the key open issues regarding the optimized placement of ENs for 5G networks are: 1) no available network-aware formulation, which has become mandatory to satisfy the required 5G Key Performance Indicators (KPIs) and avoid performance degradation in the long run, 2) limited scope, as the vast majority of the current models are unable to represent the underlying complexity of a 5G-EC ecosystem due to, for instance: unrealistic cost models and over-simplified delay constraints (i.e., commonly based on Euclidean distance), etc., 3) unrealistic assumptions made to simplify the problem complexity (e.g., number of nodes to deploy assumed to be known in advance) and, 4) lack of a flexible and extensible platform to solve the ENPP in a cost-effective manner [17], [29]–[31].

All this considered, our work is tailored to a 5G-EC ecosystem based on three key aspects. First, we address the strict latency and reliability demands of upcoming 5G use cases through a heavily constrained problem formulation. Our solution ensures ultra-low latency demand compliance while bounding the maximum allowed delay between endpoints, thus guaranteeing that the delay requirements for most of the identified 5G use cases are satisfied. Additionally, we propose a reliability demand analysis based on node redundancy allowing our solution to provide fault tolerance for sensitive scenarios. Secondly, we propose a solution strategy based on a framework characterized by flexibility and applicability, without incurring in rigid assumptions and abstraction models inapplicable to 5G scenarios. Namely, a key element of our approach is that we avoid abstracting the underlying demand distribution through clustering techniques. We argue that clustering mechanisms become ineffective to model future 5G demand distribution due to severe geographical interlacing amongst use cases (i.e., and thus amongst service demands/types) as a result of the expected deployment of dense microcells, the high number of devices within the service areas and the still unknown evolution of user demand patterns make current models outdated [3], [17], [30]. Finally, we propose a network-aware model and solution platform allowing our approach to prevent under and oversubscribed deployments, thus affecting the overall cost and performance of the 5G-EC networks.

For these reasons, the following sections present a multi-objective network-aware ENPP model and solution strategy tailored to 5G scenarios.

III. PROBLEM FORMULATION

The multi-objective ENPP aims at reducing the overall cost of deploying an EN network while ensuring that the capacity usage ratio (i.e., used capacity vs. maximum allowable EN capacity) per EN is maximized and the number of ENs is minimized. We assume that the underlying network topology (i.e., assumed to be a fully connected undirected graph) is composed by the set of nodes N and the set of links L. The set N is formed by the set of Traffic Generators (TGs), denoted as T (i.e., to abstract the EN placement from the end-user distribution characteristics without loss of accuracy and generality [41]), the nodes from the Internet Service Provider (ISP) backhaul network, existing Central Offices (COs) and Internet Service Providers-PoPs (ISP-PoPs), etc. Table 1 summarizes the variables and parameters used for the problem formulation.

TABLE 1. Complete glossary of symbols for the problem formulation.

Symbol	Params.	Vars.	Description
α_i^t		✓	1 if a TG at t is served by an EN at i, 0 otherwise
υ_i		✓	1 if an EN is placed at i, 0 otherwise
γ_{ij}^{te}		✓	fraction of the network demand of TG t served by EN e routed through link (i,j)
ψ_{ij}^{te}		✓	1 if link (i,j) is active and routing demand (i.e., $\gamma_{ij}^{te} > 0\ \forall(i,j) \in L, e \in E, t \in T$), 0 otherwise
χ_i		✓	ratio of in-use EN capacity such that $\chi_i \in (0,2]$
μ_i^t		✓	fraction of the compute demand of TG t served by an EN at i
κ_i^t		✓	fraction of the network demand of TG t served by an EN at i
F_i		✓	upfront costs of deploying an EN at i
ι_i^t		✓	cost of interconnecting an EN at i with a TG at t
θ_i		✓	cost of an EN with capacity (Cc_i, Cn_i) at i
τ	✓		cost per compute capacity unit
σ	✓		cost per network capacity unit
M_t	✓		computing demand of TG at t
K_t	✓		network demand of TG at t
A_t	✓		1 if a TG at t aggregates ultra-low latency services, 0 otherwise
R_t	✓		1 if a TG at t requires at least two serving ENs (i.e., main and backup) due to the reliability requirements of the aggregated services, 0 otherwise
Cc_i	✓		maximum compute capacity assigned to the EN at i
Cn_i	✓		maximum networking capacity assigned to (or available at) the EN at i
B_{ij}	✓		link bandwidth $(\forall(i,j) \in L)$
D_{ij}	✓		link delay $(\forall(i,j) \in L)$
P_i	✓		processing delay on node $i \in N$
D_M	✓		maximum delay allowed between a TG an its serving EN
D_U	✓		maximum delay allowed between a TG with ultra-low latency requirements an its serving EN

Considering that any $i \in N$ is a potential EN site, the optimization functions for the network-aware ENPP can be defined as follows:

$$\text{Min} \sum_{\forall i \in N} \theta_i \cdot \upsilon_i + \sum_{\forall i \in N} \sum_{\forall t \in T} \iota_i^t \cdot \alpha_i^t + \sum_{\forall i \in N} F_i \cdot \upsilon_i \quad (1)$$

$$\text{Min} \sum_{\forall i \in N} \upsilon_i \quad (2)$$

$$\text{Max} \sum_{\forall i \in N} \chi_i \cdot \upsilon_i \quad (3)$$

where,

$$\theta_i = \tau \cdot (Cc_i - \sum_{\forall t \in T} \mu_i^t) + \sigma \cdot (Cn_i - \sum_{\forall t \in T} \kappa_i^t) \quad \forall i \in N \quad (4)$$

$$\iota_i^t = \sum_{\forall (i,j) \in L} \sigma \cdot \gamma_{ij}^{te} \quad \forall e, t \in N, T \quad (5)$$

$$\chi_i = \frac{\sum_{\forall t \in T} \mu_i^t}{Cc_i} + \frac{\sum_{\forall t \in T} \kappa_i^t}{Cn_i} \quad \forall i \in N \quad (6)$$

$$\alpha_i^t, \upsilon_i, \psi_{ij}^{te} \text{ binary} \quad \forall i,\ e \in N,\ t \in T,\ (i,j) \in L \quad (7)$$

$$\theta_i, \iota_i^t, \chi_i, \beta_{ij} \geq 0 \quad \forall i \in N,\ t \in T,\ (i,j) \in L \quad (8)$$

$$\kappa_i^t, \mu_i^t, \gamma_{ij}^{te} \in [0,1] \quad \forall i,\ e \in N,\ t \in T,\ (i,j) \in L \quad (9)$$

$$Cc_i, Cn_i \geq 0 \quad \forall i \in N \quad (10)$$

Equation (1) minimizes the overall cost of deployment. The first addend accounts for the operating costs of deploying an EN at i. These expenses are found through (4) based on two elements: 1) the processing capacity deployed at i, calculated by subtracting the maximum allowable processing capacity (Cc_i) and the capacity required to satisfy the processing demands of the TGs served by the EN at i and, 2) the networking capacity deployed, calculated following the same approach but considering the maximum allowable networking capacity (Cn_i) and the TG networking demands routed through the EN at i. Each addend in (4) is multiplied by a capacity-to-cost conversion factor to return a valid cost. The second addend in (1) comprises the cost of interconnecting an EN at i with a TG at t, calculated using (5) based on the bandwidth of the active links. The third addend in (1) represents all upfront deployment costs. These fixed expenses are estimated for each potential EN site selected as EN and it is calculated based on its interconnecting and operational costs when serving a TG (hence, F_i is defined as a variable in Table 1). The objective function in (2) aims at minimizing the number of deployed ENs while (3) seeks to maximize the EN capacity usage ratio with χ_i calculated through (6). Restrictions (7) to (10) define the variables and parameters on the model.

In order to solve the multi-objective optimization model, equations (1), (2) and (3) are linearly combined using a "weighted sum" approach to obtain a single objective function [42]:

$$\text{Min } \omega_1 \cdot TC + \omega_2 \cdot NE - \omega_3 \cdot UR \quad (11)$$

where TC is the total cost of the EC network, calculated through (1), NE is the total amount of ENs deployed estimated using (2), UR is the capacity usage ratio of the ENs obtained through (3) and $\omega_1, \omega_2, \omega_3 \geq 0$.

The set of restrictions from (12) to (15) define how the model manages the TG demand and EN capacity interrelation. Both (12) and (13) ensure that the amount of demand of a TG served by one or more already selected ENs, does not exceed the TG total demand. Likewise, constraints (14) and (15) guarantee that the amount of demand served by an EN does not exceed the EN maximum capacity.

The v_e variable ensure that restrictions from (12) to (15) are enforced for the locations where an EN has been already placed.

$$\sum_{\forall e \in N} \mu_e^t \cdot v_e = 1 \quad \forall t \in T \tag{12}$$

$$\sum_{\forall e \in n} \kappa_e^t \cdot v_e = 1 \quad \forall t \in T \tag{13}$$

$$\sum_{\forall t \in T} \mu_e^t \cdot v_e \cdot M_t \leq Cc_e \quad \forall e \in N \tag{14}$$

$$\sum_{\forall t \in T} \kappa_e^t \cdot v_e \cdot K_t \leq Cn_e \quad \forall e \in N \tag{15}$$

The restrictions required to define the behavior and interrelation among a selected EN at e (i.e., where $v_e = 1$), serving a TG at t (i.e., where $\alpha_e^t = 1$) and their capacities and demands, respectively, is regulated by the constraints from (16) to (18). Both (16) and (17) imply that if a TG is served by a given EN, that EN will serve a fraction of the TG demand higher than zero. Meanwhile, (18) forces to zero the compute demand served by any EN potential location where an EN is not placed.

$$\text{if } \alpha_e^t = 1 \Leftrightarrow \mu_e^t > 0 \quad \forall e,\ t \in N, T \tag{16}$$

$$\text{if } \alpha_e^t = 1 \Leftrightarrow \kappa_e^t > 0 \quad \forall e,\ t \in N, T \tag{17}$$

$$\text{if } \upsilon_e = 0 \Leftrightarrow \sum_{\forall t \in T} \mu_e^t = 0 \quad \forall e \in N \tag{18}$$

Modeling the network-aware nature of the ENPP under strict latency constraints was challenging. Our approach, showcased from (19) to (21), models the EN-TG interconnection using "*flow conservation*" conditions. Such strategy allowed us to significantly simplify the problem definition when compared to a traditional path-based analysis, while reducing the overall computation time. Through (19) and (20) the demand entering and exiting both source and destination nodes must be equal to the total demand of the source, considering the reliability requirements of the TGs (i.e., ensuring that each TG with ultra-high reliability requirements is served by at least two ENs). By ensuring that a main and at least one backup EN serve each TG with ultra-high reliability demands, the model guarantees that no service disruption occurs in the event of a failure within the main EN. Similarly, (21) forces the amount of demand entering and exiting any node in between source and destination to be zero.

$$\sum_{\substack{\forall e \in N \\ |e \neq t}} \left(\sum_{\substack{(j,i) \in L \\ |i=t}} \gamma_{ij}^{te} - \sum_{\substack{(j,i) \in L \\ |i=t}} \gamma_{ji}^{te} \right) \geq 1 + R_t \quad \forall t \in T \tag{19}$$

$$\sum_{\substack{\forall e \in N \\ |e \neq t}} \left(\sum_{\substack{(j,i) \in L \\ |j=e}} \gamma_{ij}^{te} - \sum_{\substack{(j,i) \in L \\ |j=e}} \gamma_{ji}^{te} \right) \geq 1 + R_t \quad \forall t \in T \tag{20}$$

$$\sum_{(i,j) \in L|_{j \neq e}^{i \neq t}} \gamma_{ij}^{te} - \sum_{(j,i) \in L|_{i \neq e}^{j \neq t}} \gamma_{ji}^{te} = 0 \quad \begin{matrix} \forall e, t \in N, T \mid e \neq t, \\ n \in N \setminus \{e, t\} \end{matrix} \tag{21}$$

Since the amount of capacity for each link is limited, (22) guarantees that this capacity is not exceeded for any link in the EN-TG path selected. Restriction (23) defines a link as "*active*" (i.e., $\psi_{ij}^{te} = 1$) whenever it is used to route any amount of existing TG demands (i.e., $\gamma_{ij}^{te} > 0$). The constraint in (24) showcases the case where a TG is to be selected as EN in order to serve itself (in case it is required) and no "*active*" network link/path is therefore required. In the event of a TG at t being served by an EN at e (i.e., $\alpha_e^t = 1, v_e = 1$), (25) and (26) force the routed demand to be greater than zero and viceversa.

$$\sum_{\forall e \in N} \sum_{\forall t \in T} \gamma_{ij}^{te} \cdot K_t \leq B_{ij} \quad \forall (i,j) \in L \tag{22}$$

$$\text{if } \gamma_{ij}^{te} > 0 \Leftrightarrow \psi_{ij}^{te} = 1 \quad \forall e,\ t, (i,j) \in N, T, L \tag{23}$$

$$\text{if } e = t \Rightarrow \sum_{\forall (i,j) \in L} \psi_{ij}^{te} = 0 \quad \forall e,\ t \in N, T \tag{24}$$

$$\text{if } \sum_{\forall (i,j) \in L} \gamma_{ij}^{te} > 0 \Leftrightarrow \alpha_e^t = 1 \quad \forall e,\ t \in N, T \tag{25}$$

$$\text{if } \sum_{\forall (i,j) \in L} \gamma_{ij}^{te} > 0 \Leftrightarrow v_e = 1 \quad \forall e,\ t \in N, T \tag{26}$$

The 5G latency requirements are comprehensively modeled through (27) and (28). A maximum latency is assumed in constraint (27) for any EN-TG assignment, such that most of the 5G use cases are met for every TG. In addition, (28) was defined to guarantee ultra-low latency requirement satisfaction.

$$\sum_{\forall (i,j) \in L} (D_{ij} + P_i) \cdot \psi_{ij}^{te} + v_e \cdot P_e \leq D_M \quad \forall e,\ t \in N, T \tag{27}$$

$$\text{if } A_t = 1 \Rightarrow \sum_{\forall (i,j) \in L} (D_{ij} + P_i) \cdot \psi_{ij}^{te} + v_e \cdot P_e \leq D_U \quad \forall e,\ t \in N, T \tag{28}$$

The core aim with (27) and (28) is to ensure latency demand satisfaction for a comprehensive set of 5G use cases. For instance, setting $D_U = 1$ ms and forcing the Round-Trip Time (RTT) on the EN-TG service path -i.e., for TGs aggregating traffic from ultra-low latency 5G use cases- to be lower than D_U, enforces strict compliance of 5G requirements as presented in [2].

The propagation and processing delays for any path selected to interconnect e and t were considered in both (27) and (28) (further details on how the path delays are calculated are provided in Section IV-B).

IV. EdgeON ARCHITECTURE

By proposing a framework to solve the ENPP we aim at providing a useful tool (fully adaptable and extensible) for operators to use when planning the deployment of an EN network.

EdgeOn comprises a main (i.e., vertical) module containing all the base models used by the framework, three core processing stages, and an output/visualization phase (see Fig. 1). The **Input Processing** stage takes as input and normalizes the 5G use case requirements data (e.g., latency,

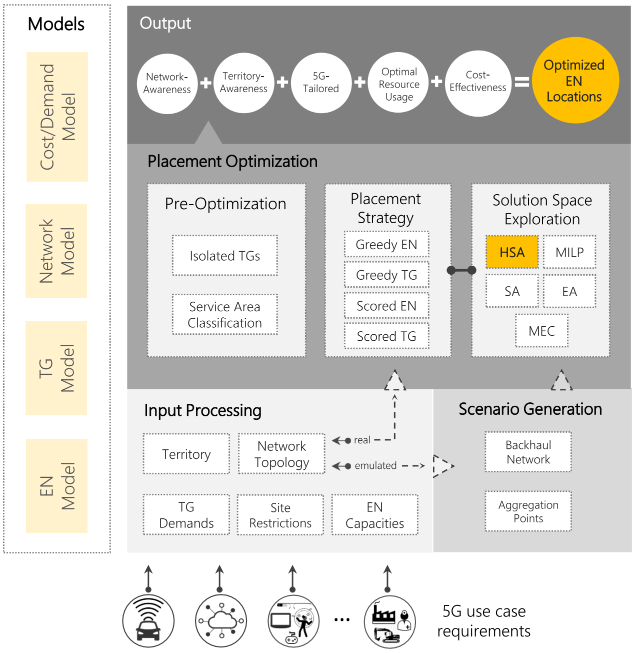


FIGURE 1. EdgeON architecture.

reliability, etc.) in order to tailor the EN ranked locations to pre-defined 5G demand values. Furthermore, a given territory of interest, network topology (see Fig. 2), EN maximum networking/computing capacity and aggregated traffic demands (i.e., TG demands) are assumed to be inputted. In addition to accepting real network topology data as input, the **Scenario Generation** stage of **EdgeOn** implements a network emulator based on the Python library Networkx,[1] to provide test scenarios accepting as input an arbitrary number of TGs and network nodes, distributed over a given number of cities (i.e., the topology generator returns an arbitrary number of Wide Area Network (WAN) networks interconnected by a high-speed backbone, thus emulating a country-sized network). In case a real topology is inputted, the **Scenario Generation** stage is bypassed and **EdgeOn** moves on to the **Placement Optimization** phase.

The key modules of the framework -i.e., **Placement Strategy** and **Solution Space Exploration**- are executed within this stage. These two steps are tightly coupled, since any method on the **Solution Space Exploration** can use one or more algorithms from the **Placement Strategy** module to generate feasible placement solutions (i.e., TG-EN pairings considering all the underlying restrictions). The current version of **EdgeON** implements four placement algorithms and five solution space exploration methods. Finally, the framework returns an optimized placement solution within the final **Output** stage. All phases of the framework are detailed further in the following subsections.

A. PRE-OPTIMIZATION MODULE

The **Pre-Optimization** module aims at reducing the overall problem complexity (as the number of TGs and potential ENs is decreased) by finding the "*isolated*" TGs and dividing the territory of interest into Service Areas [43]. In this regard, an "*isolated*" TG is defined as follows:

Definition 1: A TG t is said to be "*isolated*" when there is no potential EN site e (i.e., other than itself) within the territory or service area analyzed such that:

$$delay(t, e) \leq D \tag{29}$$

where D is the maximum delay allowed between a TG and its serving EN (i.e., D_M or D_U according to the latency requirements of the TG). Checking the territory of interest in search for isolated TGs is done through Algorithm 1. The $delay(t, e)$ value is calculated using the Networkx embedded *shortest_path*()[2] function to estimate the shortest path delay between an EN at e and TG at t. Namely, after the shortest path between e and t is found, the path delay is calculated considering the sum of the processing and propagation delays of the links and nodes in the path (i.e., the former is assumed to be a fixed known value, the latter is calculated for each link based on the distance and assuming direct fiber connections, Section V specifies the values selected or each parameter). The directly connected nodes or "*neighbors*" for each TG -i.e., obtained by calling t.*neighbors* in the

[1] https://networkx.github.io/

[2] https://networkx.github.io/documentation/stable/reference/algorithms/shortest_paths.html.

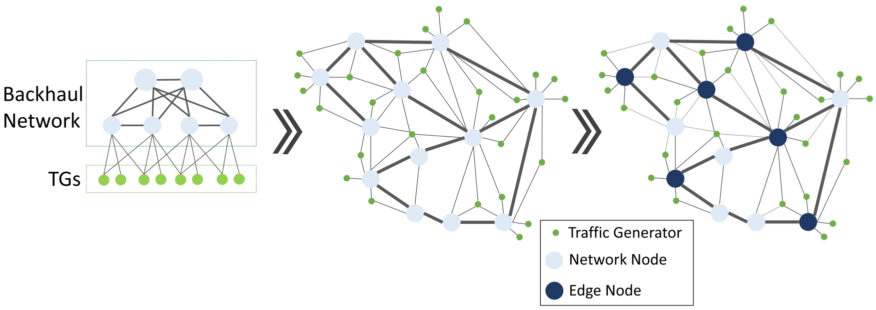


FIGURE 2. Network-aware solution process executed by EdgeON. Logical network diagram on the leftmost image, geographical node distribution on the center and rightmost images (the latter showing an example optimal EN site set).

Algorithm 1 Isolated TG Check

```
Input: $D_M, D_U$
Output: $T^s$

for $t \in T$ do
    for $e \in t.neighbors$ do
        if $A_t = 1 \wedge delay(t, e) < D_U$ then
            # Save $e$ as EN candidate for $t$
            $t.candidates \leftarrow e$
        else if $A_t = 0 \wedge delay(t, e) < D_M$ then
            # Save $e$ as EN candidate for $t$
            $t.candidates \leftarrow e$
        end if
        if $t.candidates = \varnothing$ then
            $T^s \leftarrow t$
        end if
    end for
end for

return $T^s$
```

pseudo-codes shown later in this section- are assumed to be known in advance based on the inputted (or generated) topology, although they can be easily found using Networkx available tools in case a generated topology is used. By determining the "*isolated*" TGs, the resources and execution time required to solve the problem can be effectively reduced as these nodes are immediately upgraded to ENs without loss of generality and accuracy.

On the other hand, the **Service Area Classification** method within the **Pre-Optimization** module aims at a further reduction of the ENPP difficulty. We argue that in rural areas where the user density is typically low and thus TGs are scattered over large geographical areas, a co-location strategy can be used to deploy the ENs. This co-location approach reduces overall costs by minimizing CAPEX, as the required EN capacity is low with high probability and, for instance, a co-located cabinet-based EN-RAN solution, based on wireless connectivity, can be used.

After completing the pre-optimization phase, **EdgeON** is able to execute the core modules of the ENPP solution.

B. PLACEMENT STRATEGIES

Although **EdgeON** only requires one placement strategy to solve the ENPP, the reasons to implement several in this study were twofold: a) to comprehensively evaluate different solving approaches in order to find the most suitable one for the ENPP as formulated in Section-III and, b) to provide potential users of **EdgeON** with a flexible platform and set of methods to easily adapt to their needs and use cases. For this reason, two algorithm types (i.e., EN-TG pairing methods) and two different implementations for each type were developed as placement strategies: "*greedy*" and "*scored*". The former greedily pairs TGs and ENs considering the TG requirements, available EN capacities, network usage, etc. The latter enhances the greedy strategy by scoring either the TGs or ENs in order to consider the impact of the ENs selected so far over the new EN selection. The placement strategies developed are: Greedy EN (EN-G), Greedy TG (TG-G), Scored EN (EN-SG) and Scored TG (TG-SG).

The pseudo-code for the implementations of the"*greedy*" and "*scored*" strategies are showcased in Algorithm 2 and Algorithm 3. Both methods start by sorting the TG set T such that the more demanding TGs (e.g., $A_t = 1$ or $R_t = 1$) are processed first (Lines 4 and 5 in Algorithm 2 and Algorithm 3). From Line 7 to Line 20 in both placement strategies, each TG t is then analyzed and paired to any EN e to which a feasible path is found through $best\_path(e, t)$. This function is based on a modified version of the Depth-First Search algorithm implemented by Networkx and explained in [44]. It searches and scores all simple paths from e to t (i.e., simple paths with enough network capacity on nodes and links to route t demands) and returns the best path.

Algorithm 2 Greedy EN (EN-G)

```
1:  Input: $N, L, D_M, D_U$
2:  Output: $E$
3:
4:  $T^{hr} = \{t \mid A_t = 1 \; \forall t \in T\}$
5:  $sort(T)$
6:
7:  for $t \in T$ do
8:      $randomize(t.candidates)$
9:      for $e \in t.candidates$ do
10:         if $is\_feasible(e) = True$ then
11:             $p_{et} = best\_path(e, t)$
12:             if $p_{et} \neq \varnothing \wedge e.avail\_capacity > 0$ then
13:                 $E \leftarrow e$
14:             end if
15:             if $K_t = 1 \wedge len(t \in [T^e, \forall e \in E]) \geq 1 + R_t$
                then
16:                 Remove $t$ from $T^{hr}$
17:             end if
18:         end if
19:     end for
20: end for
21:
22: for $t \in T^{hr}$ do
23:     for $e \in N$ do
24:         if $is\_feasible(e) = True$ then
25:             $p_{et} = best\_path(e, t)$
26:             if $p_{et} \neq \varnothing \wedge e.avail\_capacity > 0$ then
27:                 $E \leftarrow e$
28:             end if
29:         end if
30:     end for
31: end for
32: return $E$
```

Algorithm 3 Greedy TG (TG-G)

```
1:  Input: $N, L, D_M, D_U$
2:  Output: $E$
3:
4:  $T^{hr} = \{t \mid A_t = 1 \; \forall t \in T\}$
5:  $sort(T)$
6:
7:  while $T \neq \varnothing$ do
        Select random EN site $e$
8:      if $is\_feasible(e) = True$ then
9:          for $t \in T$ do
10:             $p_{et} = best\_path(e, t)$
11:             if $p_{et} \neq \varnothing \wedge e.avail\_capacity > 0$ then
12:                 $E \leftarrow e$
13:             end if
14:             if $K_t = 1 \wedge len(t \in [T^e, \forall e \in E]) \geq 1 + R_t$ then
15:                 Remove $t$ from $T^{hr}$
16:             end if
17:         end for
18:         Remove fully served $t$ from $T$
19:     end if
20: end while
21:
22: for $t \in T^{hr}$ do
23:     for $e \in N$ do
24:         if $is\_feasible(e) = True$ then
25:             $p_{et} = best\_path(e, t)$
26:             if $p_{et} \neq \varnothing \wedge e.avail\_capacity > 0$ then
27:                 $E \leftarrow e$
28:             end if
29:         end if
30:     end for
31: end for
32: return $E$
```

The path scoring is executed considering three path attributes: total delay from source to target, number of hops, cost (i.e., according to the cost of the active links and the capacity required in the routing nodes), energy consumption (i.e., according to the number of hops, link usage, interconnection technology, etc.). In case a valid path is found and the EN at e has enough capacity to serve t (Lines 10-12 and 24 in Algorithm 2 and Lines 8-11 and 24 in Algorithm 3), the EN-TG pairing occurs. The reliability requirement satisfaction is checked in Lines 15-17 and 14-16 for Algorithm 2 and Algorithm 3 respectively. From Lines 22 to 31, the TGs with high reliability requirements not yet satisfied are served by greedily choosing suitable ENs. It is worth noticing that a feasibility check looking for non-technical limitations is performed for each e to guarantee that only restriction-free sites are evaluated. In summary, Algorithm 2 greedily selects a feasible EN site to serve each TG, while Algorithm 3 does the opposite process by greedily assigning TGs to each EN.

In order to enhance the TG-EN pairing, both the Greedy EN and Greedy TG algorithms were modified resulting in the Scored EN and Scored TG algorithms (see the **Placement Strategy** in Fig. 1). These strategies rely on enhanced pairing methods scoring each EN potential site -i.e., based on its current usage ratio, capacity cost and non-technical limitations[3]- and each TG to be served, i.e, based on its demand (processing, networking, latency, reliability), impact on the EN capacity usage ratio and number of serving ENs. The path delay calculation includes the transmission and propagation delays corresponding to the links and network nodes traversed from source to target.

C. SOLUTION SPACE EXPLORATION

Given the strictly constrained and multi-objective nature of the ENPP, the key optimization procedure to be executed goes beyond the TG-EN pairing. Namely, the critical mechanism when solving the ENPP is the exploration of the solution space in order to determine the Pareto front. However, the ENPP defined in this research can be derived to be NP-hard due to its Multi-criteria Multi-attribute FLP nature, basically combining several FLPs problem characteristics [45]–[47]. In summary, the ENPP implies the analysis of all possible EN-TG combinations and feasible network paths in order to find the minimum cost solution, with no possible combination splitting (i.e., to reduce runtime) due to the latency restrictions. Furthermore, a simplified variant of the ENPP (i.e., a network-agnostic formulation) has been already proven to be NP-hard in [29]. Due to these reasons, exact methods were discarded to solve the ENPP for mid to large amounts of nodes (cf. Fig. 3, showcasing the exponential growth in runtime for the MILP model). Nonetheless, the MILP model presented in Section III is still included within **EdgeON** for evaluation purposes on small-sized and controlled testing scenarios.

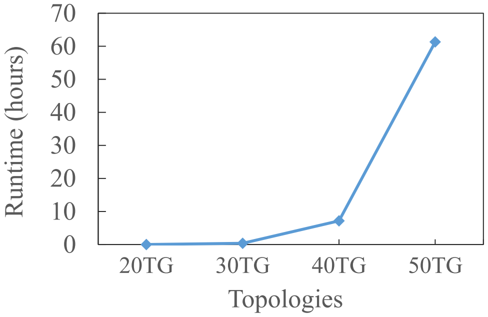


FIGURE 3. Runtime for the MILP model (measured in hours).

Currently, **EdgeON** implements four solution space analysis methods (i.e., Traditional Simulated Annealing (TSA), Hybrid Simulated Annealing (HSA) and, Evolutionary Algorithm (EA)) and a widely used approach for EN placement

[3] If the EN potential site is a PoP -e.g., a CO, a ISP-PoP, etc.- a score bonus is added to enforce using PoPs as ENs given their potential lower CAPEX/OPEX when compared to, for instance, deploying ENs at TG sites.

(i.e., MEC, where the ENs are co-located with the RAN nodes). These algorithms are among the most used to solve complex placement problems and were selected based on their flexibility to be adapted to the particularities of the ENPP, namely, its non-convergent nature within the FLP problem set and the added difficulties of a network-aware formulation. Nevertheless, due to the promising results of the HSA placement solution (cf. [41], [48] and [43]) this is the default ENPP mechanism used by **EdgeON**. The pseudo-code of the adapted HSA proposed and implemented in the current version of **EdgeON** is showcased in Algorithm 4 where the *score*() function refers to (11).

The key elements in HSA are the memory structures, "*intensification*" and "*diversification*" mechanisms inherited from widely used FLP solution strategies, namely, the Tabu Search algorithm. Such elements combined with the SA core provide HSA with a strong ability to escape local optima and thoroughly explore the problem solution space.

In summary, the heuristic works as follows: Lines 4-9 initialize the algorithm variables, for instance, an initial solution i_s is generated and recorded as best solution (i.e., b_s) in Line 5 and as overall best solution (i.e., b_{se}, best solution after a temperature cycle) in Line 6, the "*bad solution*" w_s is set to *None* (cf. Line 7) and the initial number of neighbor solutions n_s is estimated (cf. Line 8). HSA relies on three types of neighbor solutions: "*good*", "*bad*" and "*random*". The "*good*" and "*bad*" solutions are based on the EN-TG pairing inherited from previous solutions with superior and inferior score values respectively, as explained in [41], [43], [48]. The idea is to thoroughly explore the solution space by analyzing randomly selected low-performance solutions while pushing the algorithm towards the EN-TG pairing found by the better solutions. To reduce the probability of generating unfeasible solutions (i.e., usually due to TGs with ultra-high reliability requirements unable to be served by more than one EN because of prior EN-TG assignments, latency constraints and capacity issues on the remaining EN potential sites and available paths) the TG set is sorted to ensure that the TGs with ultra-high reliability requirements are processed first (cf. Lines 11 and 12). The neighbor solutions set is generated in Line 16 and scored in Line 17. Furthermore, in case that a score improvement is found by the checks performed in Lines 19-26, b_s is updated accordingly.

At this point, due to the HSA core based on the SA "*cooling*" scheme, a worst solution may still be saved as best solution in order to escape local optima (cf. Line 24). The number of solutions within the neighbor set is optimized from Line 32 to Line 44 according to the results obtained in each cycle and a random value between 0 and 1. Additionally, the temperature values are decreased at a slow or fast rate within this code segment (i.e., through the α_s and α_f values respectively, in Lines 36 and 42). These decrease factors depend on whether the score of the best solution of the preceding iteration cycle (i.e., b_s) is better or not than the best recorded solution so far (i.e., b_{se}).

Algorithm 4 Hybrid Simulated Annealing

```
 1: Input: $M, EN, TG, td_t, D_{max}, A, B, C, T$ maximum temperature, $T_{min}$ minimum temperature, $I$ counter, $n_s$ number of neighbor solutions, $\alpha_f$ fast $T$ decrease factor, $\alpha_s$ slow $T$ decrease factor
 2: Output: $b_{se}$
 3:
 4: $i_s \leftarrow gen\_sol()$
 5: $b_s \leftarrow i_s$
 6: $b_{se} \leftarrow i_s$
 7: $w_s \leftarrow \varnothing$
 8: $n_s \leftarrow num\_neig(T, T_{min})$
 9: $i \leftarrow 0$
10:
11: $T^{hr} = \{t \mid A_t = 1 \; \forall t \in T\}$
12: $sort(T)$
13:
14: while $T < T_{min}$ do
15:   while $i < I$ do
16:     $N \leftarrow neig\_set(b_s, w_s, n_s)$
17:     $S \leftarrow score(N)$
18:
19:     if $score(S[0]) < score(b_s)$ then
20:       $b_s \leftarrow S[0]$
21:     else
22:       $p \leftarrow ap(T, score(S[0]), score(b_s))$
23:       if $p > random(0, 1)$ then
24:         $b_s \leftarrow S[0]$
25:       end if
26:     end if
27:
28:     $w_s \leftarrow rand\_sol(S)$
29:     $i \leftarrow i + 1$
30:   end while
31:
32:   if $score(b_s) < score(b_{se})$ then
33:     $T \leftarrow T * \alpha_f$
34:     $p \leftarrow 1 - ap(T, score(b_s), score(b_{se}))$
35:     if $p > random(0, 1)$ then
36:       $n_s \leftarrow decrease(n_s, \alpha_f)$
37:     end if
38:   else
39:     $T \leftarrow T * \alpha_s$
40:     $p \leftarrow ap(T, score(b_s), score(b_{se}))$
41:     if $p > random(0, 1)$ then
42:       $n_s \leftarrow decrease(n_s, \alpha_f)$
43:     end if
44:   end if
45: end while
46: return $E$
```

D. OUTPUT

The last stage of the framework returns the best solution obtained containing the set of EN locations to place the service infrastructure at the edge of the 5G network and the network paths, link and node usage regarding the TG-EN interconnection. Additionally, **EdgeON** optionally provides both static and interactive charts depicting the deployment details and the performance of the selected placement solutions.

V. EVALUATION AND RESULTS

To evaluate **EdgeON**'s suitability to solve the proposed ENPP we conducted experiments on emulated network topologies varying the number of TGs, the placement strategies and the solution space exploration mechanisms.

The testbed used was developed using the **Scenario Generation** tool embedded within **EdgeON**. Namely, we emulated a geographical area (i.e., a 2D map grid formed by (x, y) coordinate pairs with a 1 m separation step) and, in each experiment, we varied the network topology placed within this area. Each topology generated was formed by a scattered set of TGs and network nodes (i.e., interconnecting the TGs) randomly scattered resembling a WAN network surrounded by rural territory (i.e., where TGs are separated by a higher distance). All network topologies were generated through the Python library Networkx (i.e., as mentioned in Section IV) as fully connected undirected graphs with all edges assumed to be fiber optic links. Overall, 9 topologies were tested, with the number of TGs ranging from 20 to 100 nodes (with an increase step of 10 nodes) and the number of network nodes assumed to be half the amount of the TGs within each topology.

The link delay was assumed to be calculated based on the distance between the vertices and each link was assigned either 1 or 10 Gbps capacity based on the link type, i.e., lower bandwidth for the links connecting the TGs to the core network nodes (i.e., access links) and higher bandwidth for the backbone network links (i.e., links where no vertex is a TG). In addition, each routing node within the network was assumed to have a typipcal processing delay of 0.05 ms (i.e., for IP forwarding) [50]. The maximum networking and processing capacities were set to 300 units (i.e., generic units were used to model the bandwidth/processing capacities for the ENs and network nodes) for each EN, while the same network capacity value was assigned to each network node. To obtain this capacity value we ran **EdgeON** 10 times for each topology with randomly selected capacity values. The goal was to find an arbitrary capacity value forcing the worst placement conditions for most of the topologies -i.e., when the majority of the TGs must be served by more than one EN, thus resulting in drastic capacity imbalance and complex EN-TG pairing. Moreover, each TG within each topology was assigned a random processing and networking demand ranging from 20 to 100 units, along with random latency and reliability requirements. The conversion factors τ and σ were set to 10000 $/unit and 700 $/unit to model the general operating costs of deploying an EN considering a realistic scenario [51]. Table 2 summarizes the parameter values used of the scenario generation, while Table 3 to Table 5 present the input parameter values used for the solution space exploration algorithms.

To simulate 5G heavily constrained use cases regarding, for instance, latency and reliability, we assumed a RTT of 1 ms for ultra-low delay requirements and 10 ms for the remaining 5G scenarios (i.e., $D_U = 0.5$ ms and $D_M = 5$ ms). The 1 ms RTT ensures compliance with the identified demands for

TABLE 2. Parameter values.

Model	Param.	Unit	Value	Details
Network	Cn_i	-	300	Generic capacity units were used
	B_{ij}	Gbps	1 - 10	Lower bandwidth for access links, higher bandwidth for core network links
	D_{ij}	ms	-	Estimated based on the distance between nodes assuming a direct fiber link and a propagation delay of 5 μs/km [49]
	P_i	ms	0.05	Typical processing delay for IP forwarding
EN	Cc_i	-	300	Generic capacity units were used
TG	M_t	-	20 - 100	A random processing demand is assigned to each TG
	K_t	-	20 - 100	A random networking demand is assigned to each TG
	A_t	-	0 - 1	Randomly set to 1 (ultra-low latency) or 0 for each TG
	R_t	-	0 - 1	Randomly set to 1 (ultra-high reliability) or 0 for each TG
Cost	τ	$/unit	10000	Cost per generic capacity unit
	σ	$/unit	700	Cost per generic capacity unit

TABLE 3. Input parameters for EA.

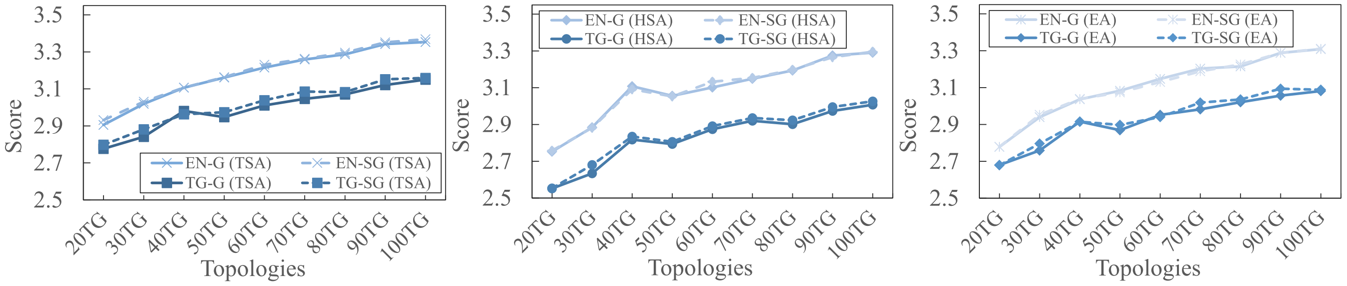
Parameter	Value
Num. Generations	100.00
Num. Individuals	100.00
Mutation rate	0.01

TABLE 4. Input parameters for the HSA.

Parameter	Value
Minimum Temperature	0.0001
Maximum Temperature	1.0000
Temperature Iterations	10.000
Fast Alpha	0.8000
Slow Alpha	0.9500
Num. Neighbors	10.000

TABLE 5. Input parameters for the TSA.

Parameter	Value
Minimum Temperature	0.0001
Maximum Temperature	1.0000
Temperature Iterations	10.000
Alpha	0.9500
Num. Neighbors	10.000



(a) Performance using TSA.

(b) Performance using HSA.

(c) Performance using EA.

FIGURE 4. Evaluation of the placement strategies for all solution space exploration algorithms and network topologies with TG values ranging from 20 to 100 TGs. The naming convention is as follows: Greedy EN → EN-G, Greedy TG → TG-G, Scored EN → EN-SG and, Scored TG → TG-SG.

5G ultra-low latency use cases [2], [5]. Meanwhile, the maximum RTT allowed of 10 ms, for any EN-TG pairing, guarantees that most 5G use cases can be met for any TG and its serving ENs [2].

For the objective function we arbitrarily selected the normalized weights $\omega_1 = 0.35$, $\omega_2 = 0.33$, $\omega_3 = 0.32$. Similarly, arbitrary values were selected for the weights in the *best_path*(e, t) function.

The first step towards a comprehensive evaluation of **EdgeON**'s capabilities was to determine the best placement strategy to solve the ENPP, due to the critical impact of the EN-TG pairing on the overall performance of the solution. To this aim, we repeatedly ran the TSA, HSA and EA algorithms for all placement strategies and topologies. The results are showcased in Fig. 4.

Taking into account that the lower the score the better the performance, for all the topologies analyzed (i.e., named after the number of TGs on the topology), the Greedy EN (EN-G) and Scored EN (EN-SG) were significantly outperformed by both the Greedy TG (TG-G) and Scored TG (TG-SG). The reason is that greedily assigning feasible ENs to each TG results in a poor usage ratio balance and higher number of ENs when compared to selecting random ENs and greedily pairing them with suitable TGs, considering the underlying capacities and TG requirements. Consequently, we discarded EN-G and EN-SG as placement strategies in favor of TG-G and TG-SG for the remaining of our experiments.

A different perspective to further analyze the placement strategies performance is shown in Fig. 5. Crosschecking the charts in Fig. 4 and Fig. 5 evidences the superiority of TG-G and TG-SG for any solution space exploration mechanism. For all topologies analyzed, both TG-G and TG-SG outperformed the remaining placement strategies, resulting in significantly lower costs, lower number of deployed ENs and higher average usage ratio, thus lowering the overall score. In addition, Fig. 5 depicts how TG-G performed slightly better than TG-SG for all algorithms and the majority of topologies analyzed. Consequently, we set TG-G as the default placement strategy to solve the ENPP using **EdgeON**.

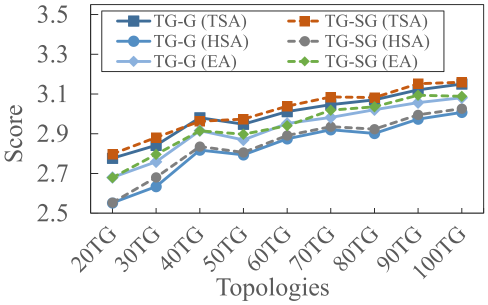


FIGURE 5. Performance results for the TG-G and TG-SG placement strategies for all topologies analyzed.

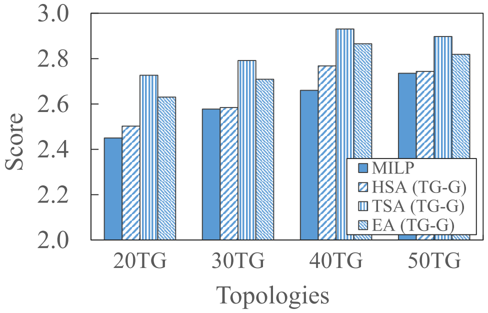
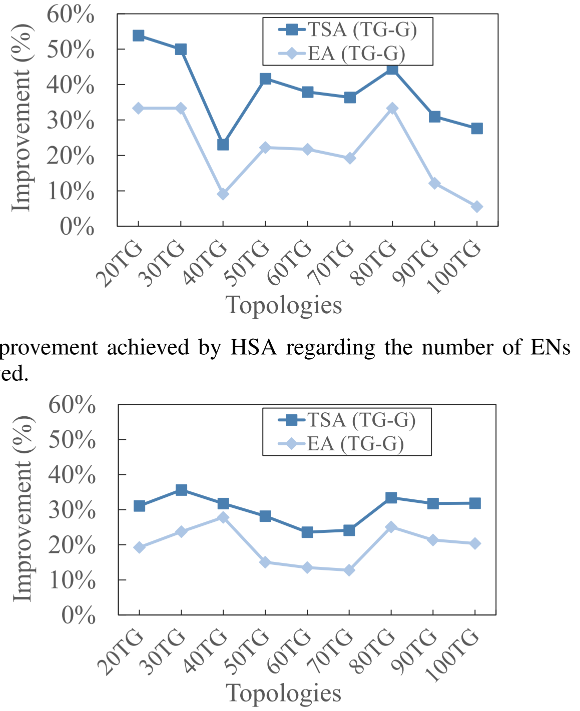


FIGURE 6. Performance results for the MILP model compared to the solution space exploration heuristics (i.e., HSA, TSA, EA).

The second step on **EdgeON**'s analysis was to thoroughly assess the solution space exploration strategies. The idea within this step was to evaluate **EdgeON**'s ability to find the best near-optimal solution using our in-house heuristic (i.e., HSA) tested against the exact method in a controlled test scenario -i.e., reduced number of nodes- and against widely used heuristics commonly applied to other placement problems. The findings of these tests are depicted in Fig. 6 and Fig. 7. The former showcases the superiority of HSA when compared to the other heuristics, with an average score offset of around 1.5% compared to the MILP model,[4] i.e., with EA and TSA achieving a score offset of 6% and 8% respectively. The significantly better score offset obtained by HSA when compared to TSA and EA (cf. Fig. 7) resulted from its performance improvements in terms of number of ENs and average capacity usage ratio. Overall, Fig. 7 illustrates that HSA deployed an average of nearly 40% less ENs

[4] Results shown for topologies with less than 50 TGs due to the exponential increase in runtime for the MILP model when applied to topologies with more than 50 nodes.



(a) Improvement achieved by HSA regarding the number of ENs deployed.

(b) Improvement achieved by HSA regarding the average EN capacity Usage Ratio.

FIGURE 7. **HSA results for Num. ENs and usage ratio compared to TSA and EA (using TG-G in all cases). The HSA improvement percentage is depicted.**

than TSA and 20% less than EA. Moreover, HSA achieved a 30% and 20% higher average usage ratio when compared to TSA and EA respectively.

Finally, to further validate HSA's suitability for EN deployment within 5G networking, we tested it against a commonly preferred strategy to locate ENs: the MEC approach, where as mentioned above, the service infrastructure (i.e., the EN) is arbitrarily co-located with the RAN nodes. As expected, Fig. 8 evidences how using MEC can lead to a rather inefficient EC network deployment when compared to HSA, since it results in lower usage ratio, higher number of deployed ENs and performance degradation due to overlooking the in-place backhaul network capacity. In summary, the MEC approach placed an average of 71% more ENs than HSA (using TG-G as placement strategy) and resulted in 50% less average usage ratio for the vast majority of the analyzed scenarios (cf. Fig. 9).

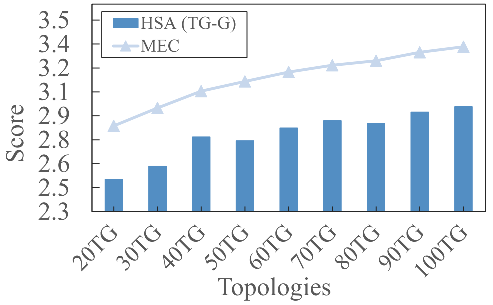


FIGURE 8. **Performance results for HSA and MEC using TG-G as placement strategy.**

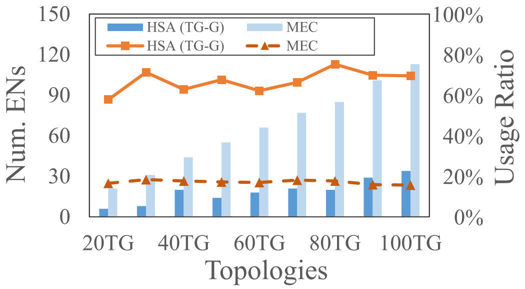


FIGURE 9. **Performance results for HSA and MEC using TG-G as placement strategy in terms of average usage ratio (lines, left *y* axis) and Num. ENs (columns, right *y* axis).**

The aforementioned results encourage the evaluation and test of **EdgeON** on real-life scenarios and network topologies. Furthermore, its modular implementation ensures an easy-to-use and extensible platform for operators to adapt to their requirements and use cases.

VI. CONCLUSION

This paper presents a novel and rigorous definition of the network-aware ENPP under heavily constrained 5G scenarios and a tailored framework to solve this problem based on several placement strategies and optimization heuristics. A key goal of this work was to provide a useful tool for current operators and telcos to use when planning the deployment of an EC network. As a result, we developed the **EdgeON** framework focusing on flexibility and extensibility, while comprising a thorough analysis of the technical and non-technical aspects and costs of the network-aware EN placement.

To validate the capabilities of **EdgeON**, the performance of its core placement optimization solution, based on an in-house heuristic (i.e., HSA), was thoroughly assessed. The promising results obtained encourage its use to solve the network-aware ENPP under strict 5G use case requirements. Namely, significant improvements were achieved regarding the number of ENs deployed and average usage ratio (i.e., around 30% lower and 25% higher, respectively, compared to the remaining tested heuristics). Moreover, an average score offset of just 2% was obtained when testing our heuristic against an exact method (i.e., MILP model).

Future work and open research questions to improve **EdgeON**'s capabilities and performance include the study of EN online placement feasibility for 5G use cases and the development of a mobile/desktop application implementing an extended version of **EdgeON** (i.e., enhancing its support for real-life network topologies, extended set of optimization parameters, etc.).

REFERENCES

[1] S. K. Sharma and X. Wang, "Toward massive machine type communications in ultra-dense cellular IoT networks: Current issues and machine learning-assisted solutions," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 1, pp. 426–471, 1st Quart., 2020.

[2] B. Blanco, J. O. Fajardo, I. Giannoulakis, E. Kafetzakis, S. Peng, J. Pérez-Romero, I. Trajkovska, P. S. Khodashenas, L. Goratti, M. Paolino, E. Sfakianakis, F. Liberal, and G. Xilouris, "Technology pillars in the architecture of future 5G mobile networks: NFV, MEC and SDN," *Comput. Standards Interfaces*, vol. 54, pp. 216–228, Nov. 2017.

[3] W. Xiang, K. Zheng, and X. Shen, *5G Mobile Communications*. New York, NY, USA: Springer, 2017.

[4] I. Ahmad, T. Kumar, M. Liyanage, J. Okwuibe, M. Ylianttila, and A. Gurtov, "Overview of 5G security challenges and solutions," *IEEE Commun. Standards Mag.*, vol. 2, no. 1, pp. 36–43, Mar. 2018.

[5] I. Parvez, A. Rahmati, I. Guvenc, A. I. Sarwat, and H. Dai, "A survey on low latency towards 5G: RAN, core network and caching solutions," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 3098–3130, 2018.

[6] S. E. Elayoubi, M. Fallgren, P. Spapis, G. Zimmermann, D. Martin-Sacristan, C. Yang, S. Jeux, P. Agyapong, L. Campoy, Y. Qi, and S. Singh, "5G service requirements and operational use cases: Analysis and METIS II vision," in *Proc. Eur. Conf. Netw. Commun. (EuCNC)*, Jun. 2016, pp. 158–162.

[7] B. Varghese, N. Wang, D. S. Nikolopoulos, and R. Buyya, *Feasibility of Fog Computing*. New York, NY, USA: Springer, 2019.

[8] M. M. Hussain, M. S. Alam, and M. M. S. Beg, "Feasibility of fog computing in smart grid architectures," in *Proc. 2nd Int. Conf. Commun., Comput. Netw.*, vol. 46, C. R. Krishna, M. Dutta, and R. Kumar, Eds. Singapore: Springer, 2019, pp. 999–1010.

[9] P. Bellavista and A. Zanni, "Feasibility of fog computing deployment based on Docker containerization over RaspberryPi," in *Proc. 18th Int. Conf. Distrib. Comput. Netw. (ICDCN)*. Hyderabad, India: ACM Press, 2017, pp. 1–10.

[10] C.-T. Kuo, V. Chang, and C.-L. Lei, "A feasibility analysis for edge computing fusion in LPWA IoT environment with SDN structure," in *Proc. Int. Conf. Eng. Technol. (ICET)*, Aug. 2017, pp. 1–6.

[11] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322–2358, Aug. 2017.

[12] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016.

[13] A. C. Baktir, A. Ozgovde, and C. Ersoy, "How can edge computing benefit from software-defined networking: A survey, use cases, and future directions," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2359–2391, Jun. 2017.

[14] H. Tanaka, M. Yoshida, K. Mori, and N. Takahashi, "Multi-access edge computing: A survey," *J. Inf. Process.*, vol. 26, pp. 87–97, Feb. 2018.

[15] I. Hadzic, Y. Abe, and H. C. Woithe, "Server placement and selection for edge computing in the ePC," *IEEE Trans. Services Comput.*, vol. 12, no. 5, pp. 671–684, Sep. 2019.

[16] I. Hadzic, Y. Abe, and H. C. Woithe, "Edge computing in the ePC: A reality check," in *Proc. 2nd ACM IEEE Symp. Edge Comput. (SEC)*. San Jose, CA, USA: ACM Press, 2017, pp. 1–10.

[17] F. Zeng, Y. Ren, X. Deng, and W. Li, "Cost-effective edge server placement in wireless metropolitan area networks," *Sensors*, vol. 19, no. 1, p. 32, 2019.

[18] T.-W. Kuo, B.-H. Liou, K. C.-J. Lin, and M.-J. Tsai, "Deploying chains of virtual network functions: On the relation between link and server usage," *IEEE/ACM Trans. Netw.*, vol. 26, no. 4, pp. 1562–1576, Aug. 2018.

[19] S. Wang and C. Ran, "Rethinking cellular network planning and optimization," *IEEE Wireless Commun.*, vol. 23, no. 2, pp. 118–125, Apr. 2016.

[20] S. Wang, W. Zhao, and C. Wang, "Budgeted cell planning for cellular networks with small cells," *IEEE Trans. Veh. Technol.*, vol. 64, no. 10, pp. 4797–4806, Oct. 2015.

[21] S. Hasan, S. Gorinsky, C. Dovrolis, and R. K. Sitaraman, "Trade-offs in optimizing the cache deployments of CDNs," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Apr. 2014, pp. 460–468.

[22] I. Gravalos, P. Makris, K. Christodoulopoulos, and E. A. Varvarigos, "Efficient gateways placement for Internet of Things with QoS constraints," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2016, pp. 1–6.

[23] Y. Zhang, D. Li, and M. Tatipamula, "The freshman handbook: A hint for server placement in online social network services," in *Proc. IEEE 18th Int. Conf. Parallel Distrib. Syst.*, Dec. 2012, pp. 588–595.

[24] Z. Ulukan and E. Demircioglu, "A survey of discrete facility location problems," *Int. J. Social Behav., Educ., Econ., Bus. Ind. Eng.*, vol. 9, no. 7, pp. 2487–2492, 2015.

[25] R. Z. Farahani, M. Steadieseifi, and N. Asgari, "Multiple criteria facility location problems: A survey," *Appl. Math. Model.*, vol. 34, no. 7, pp. 1689–1709, Jul. 2010.

[26] M. Barbati, "Models and algorithms for facility location problems with equity considerations," Ph.D. dissertation, Universita degli Studi di Napoli Federico II, Naples, Italy, 2013.

[27] A. Boloori Arabani and R. Z. Farahani, "Facility location dynamics: An overview of classifications and applications," *Comput. Ind. Eng.*, vol. 62, no. 1, pp. 408–420, Feb. 2012.

[28] I. Goiri, K. Le, J. Guitart, J. Torres, and R. Bianchini, "Intelligent placement of datacenters for Internet services," in *Proc. 31st Int. Conf. Distrib. Comput. Syst.*, Jun. 2011, pp. 131–142.

[29] S. Wang, Y. Zhao, J. Xu, J. Yuan, and C.-H. Hsu, "Edge server placement in mobile edge computing," *J. Parallel Distrib. Comput.*, vol. 127, pp. 160–168, May 2019.

[30] N. Mohan, A. Zavodovski, P. Zhou, and J. Kangasharju, "Anveshak: Placing edge servers in the wild," in *Proc. Workshop Mobile Edge Commun. (MECOMM)* Budapest, Hungary: ACM Press, 2018, pp. 7–12.

[31] H. Yin, X. Zhang, H. H. Liu, Y. Luo, C. Tian, S. Zhao, and F. Li, "Edge provisioning with flexible server placement," *IEEE Trans. Parallel Distrib. Syst.*, vol. 28, no. 4, pp. 1031–1045, Apr. 2017.

[32] W. Zhang, B. Lin, Q. Yin, and T. Zhao, "Infrastructure deployment and optimization of fog network based on MicroDC and LRPON integration," *Peer-to-Peer Netw. Appl.*, vol. 10, no. 3, pp. 579–591, May 2017.

[33] S. Zhou, D. Lee, B. Leng, X. Zhou, H. Zhang, and Z. Niu, "On the spatial distribution of base stations and its relation to the traffic density in cellular networks," *IEEE Access*, vol. 3, pp. 998–1010, 2015.

[34] J. Kosmerl and A. Vilhar, "Base stations placement optimization in wireless networks for emergency communications," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC)*, Jun. 2014, pp. 200–205.

[35] T. Taleb and A. Ksentini, "On efficient data anchor point selection in distributed mobile networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2013, pp. 6289–6293.

[36] S. Barbarossa, S. Sardellitti, and P. Di Lorenzo, "Communicating while computing: Distributed mobile cloud computing over 5G heterogeneous networks," *IEEE Signal Process. Mag.*, vol. 31, no. 6, pp. 45–55, Nov. 2014.

[37] F. Lobillo, Z. Becvar, M. A. Puente, P. Mach, F. Lo Presti, F. Gambetti, M. Goldhamer, J. Vidal, A. K. Widiawan, and E. Calvanesse, "An architecture for mobile computation offloading on cloud-enabled LTE small cells," in *Proc. IEEE Wireless Commun. Netw. Conf. Workshops (WCNCW)*, Apr. 2014, pp. 1–6.

[38] I. Giannoulakis, E. Kafetzakis, I. Trajkovska, P. S. Khodashenas, I. Chochliouros, C. Costa, I. Neokosmidis, and P. Bliznakov, "The emergence of operator-neutral small cells as a strong case for cloud computing at the mobile edge," *Trans. Emerg. Telecommun. Technol.*, vol. 27, no. 9, pp. 1152–1159, Sep. 2016.

[39] P. Maiti, J. Shukla, B. Sahoo, and A. K. Turuk, "QoS-aware fog nodes placement," in *Proc. 4th Int. Conf. Recent Adv. Inf. Technol. (RAIT)*, Mar. 2018, pp. 1–6.

[40] M. Bouet and V. Conan, "Mobile edge computing resources optimization: A geo-clustering approach," *IEEE Trans. Netw. Service Manage.*, vol. 15, no. 2, pp. 787–796, Jun. 2018.

[41] A. Santoyo-González and C. Cervelló-Pastor, "Latency-aware cost optimization of the service infrastructure placement in 5G networks," *J. Netw. Comput. Appl.*, vol. 114, pp. 29–37, Jul. 2018.

[42] J.-H. Cho, Y. Wang, I.-R. Chen, K. S. Chan, and A. Swami, "A survey on modeling and optimizing multi-objective systems," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1867–1901, 3rd Quart., 2017.

[43] A. S. González and C. C. Pastor, "Edge computing node placement in 5G networks: A latency and reliability constrained framework," in *Proc. 6th IEEE Int. Conf. Cyber Secur. Cloud Comput. (CSCloud), 5th IEEE Int. Conf. Edge Comput. Scalable Cloud (EdgeCom)*, Jun. 2019, pp. 183–189.

[44] R. Sedgewick, *Algorithms in C, Part 5: Graph Algorithms*, 3rd ed. Reading, MA, USA: Addison-Wesley, 2001.

[45] R. Z. Farahani, N. Asgari, N. Heidari, M. Hosseininia, and M. Goh, "Covering problems in facility location: A review," *Comput. Ind. Eng.*, vol. 62, no. 1, pp. 368–407, Feb. 2012.

[46] L.-Y. Wu, X.-S. Zhang, and J.-L. Zhang, "Capacitated facility location problem with general setup cost," *Comput. Oper. Res.*, vol. 33, no. 5, pp. 1226–1241, May 2006.

[47] Z. Zhu, F. Chu, and L. Sun, "The capacitated plant location problem with customers and suppliers matching," *Transp. Res. E, Logistics Transp. Rev.*, vol. 46, no. 3, pp. 469–480, May 2010.

[48] I. Leyva-Pupo, A. Santoyo-González, and C. Cervelló-Pastor, "A framework for the joint placement of edge service infrastructure and user plane functions for 5G," *Sensors*, vol. 19, no. 18, p. 3975, 2019.

[49] M. H. Eiselt and F. Azendorf, "Accurate measurement of propagation delay in a multi-span optical link," in *Proc. Int. Topical Meeting Microw. Photon. (MWP)*. Ottawa, ON, Canada: IEEE, Oct. 2019, pp. 1–3.

[50] R. Ramaswamy, N. Weng, and T. Wolf, "Characterizing network processing delay," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, vol. 3, 2004, pp. 1629–1634.

[51] D. Hardy, M. Kleanthous, I. Sideris, A. G. Saidi, E. Ozer, and Y. Sazeides, "An analytical framework for estimating TCO and exploring data center design space," in *Proc. IEEE Int. Symp. Perform. Anal. Syst. Softw. (ISPASS)*, Apr. 2013, pp. 54–63.

ALEJANDRO SANTOYO-GONZÁLEZ received the B.Sc. degree in telecommunication and electronic engineering from the Havana University of Technology Jose Antonio Echeverria, Havana, Cuba. He is currently pursuing the Ph.D. degree with the Universitat Politècnica de Catalunya (UPC), Barcelona, Spain.

He has experience as a System Engineer and a Solution Manager at Huawei Technologies Co., Ltd. He is also a Researcher with UPC. His main research interests include edge computing, NFV, optimization problems, and 5G networking.

CRISTINA CERVELLÓ-PASTOR received the M.Sc. and Ph.D. degrees in telecommunication engineering from the Barcelona School of Telecommunications Engineering (ETSETB), Universitat Politècnica de Catalunya (UPC), Barcelona, Spain.

She is currently an Associate Professor and the Head of the Department of Network Engineering, UPC. Being part of BAMPLA research group, she has been responsible and actively participated in diverse national and European competitive projects (NOVI, FEDERICA, ATDMA, A@DAN, Euro-NGI, Euro-FGI, and EURO-NF) and private funding R&D projects. In parallel, she has published diverse papers in national and international journals and conferences. She has been supervising thesis in the field of management, optimal resource allocation, topology discovery, and routing in SDN/NFV and 5G.