Distributed task offloading in MEC networks for temporary peaks in demand

Camilo Anzola-Rojas
Universidad de Valladolid
Valladolid, Spain
0000-0002-8489-2426

Ramón J. Durán Barroso
Universidad de Valladolid
Valladolid, Spain
0000-0003-1423-1646

Ignacio de Miguel
Universidad de Valladolid
Valladolid, Spain
0000-0002-1084-1159

Juan Carlos Aguado
Universidad de Valladolid
Valladolid, Spain
0000-0002-2495-0313

Noemí Merayo
Universidad de Valladolid
Valladolid, Spain
0000-0002-6920-0778

Patricia Fernández
Universidad de Valladolid
Valladolid, Spain
0000-0001-5520-0948

Rubén M. Lorenzo
Universidad de Valladolid
Valladolid, Spain
0000-0001-8729-3085

Evaristo J. Abril
Universidad de Valladolid
Valladolid, Spain
0000-0003-4164-2467

***Abstract*—Multi-Access Edge Computing (MEC) network planning is performed considering a forecast of estimated workload in each coverage zone with the aim of offloading computationally expensive tasks from user's devices to the nearest MEC Data Center (MEC-DC). Nevertheless, in some scenarios, these forecasts are exceeded temporarily due to sudden peaks in demand in a determined MEC-DC, making its planned computing resources (*i.e.,* MEC servers) scarce, and introducing the need of a strategy to face this increment in demand. In this paper, we propose and evaluate an Integer Linear Programming (ILP) model for optimizing the task offloading considering a previously defined MEC network topology. Our model is based on the possibility of offloading some tasks to MEC-DCs different to the initially planned (nearest to the user) one, as long as the latency requirements are met, and the allocated server has enough idle computing power. Results show that the proposed strategy considerably increases the capacity of the network to face sudden workload increments compared to an approach that only assigns the nearest MEC server to every user.**

Keywords—Resource allocation, computation offloading, Multi Access Edge Computing MEC, resource optimization, Network operation.

I. INTRODUCTION

Multi-access Edge Computing (MEC) [1] networks need careful planning before deployment in order to reduce the cost and guarantee the technical requirements of the applications to be run on the servers. This planning is usually performed based on a forecast of how much computing power will be needed in a determined zone, and based on this forecast, a MEC Data Center (MEC-DC) in a nearby zone is provisioned with the required number of servers. Despite these demand forecasts are usually accurate enough under normal conditions, there are specific scenarios in which the network has to process more workload than usual, e.g., social events, concerts, sports matches, or any type of crowd. In such cases a static resource allocation in which all tasks are offloaded to the nearest MEC-DC, yields to the overload of this MEC-DC, meaning a high number of unsuccessful tasks due to the sudden increment in demand. Moreover, since these workload peaks can be temporary, it is not convenient to deploy more MEC servers just to face these demand increments, hence a cheaper and temporary solution must be envisaged.

Population decreasing is a growing issue in many zones around the world, such as some small villages of Spain. In such villages this problem is increased given the low offer of some services like hospitals, shops, entertainment, and good internet connections. The lack of services makes people move away from these villages, and the lack of people hinders the improvement of the services, producing positive feedback on the problem and making it harder to solve. One step towards overcoming this issue is to provide these villages with communication networks capable of offering last-generation services to the local population, as well as to potential visitors.

When it comes to sparsely populated areas, the network resources are planned considering a low workload given the low population in the coverage zone. Therefore, it is especially important to implement a strategy to face temporary peaks in resources demand, allowing to offer stable and reliable services in crowdy events and encouraging the realization of such events in these zones without a big economic investment that is not fully exploited after finishing the event.

In this paper, we propose and validate a computation offloading scheme considering a scenario just like the one mentioned above, in which the demand of resources is increased in a specific zone of a MEC network in a sparsely populated area, causing the nearest MEC-DC to be uncapable of handling all the incoming requests. In our proposal, we introduce the possibility of allocating resources of a MEC-DC different to the closest one to the user. Results show that the proposed scheme fulfils the latency constraints of MEC services, and considerably increases the success rate of the

Funding Information. Ministerio de Ciencia e Innovación / Agencia Estatal de Investigación (MCIN/AEI/10.13039/501100011033) (PID2020-112675RB-C42), Consejería de Educación de la Junta de Castilla y León and European Regional Development Fund (VA231P20), and the EU H2020 research and innovation programme under the MSCA grant agreement No 953442 (IoTalentum).

offloaded tasks compared to the static allocation of only the nearest MEC-DC to all tasks.

II. Related Work

The computation offloading and resource allocation problem in MEC networks has been previously studied. The work presented in [2] proposes an offload resource allocation algorithm for vehicular networks combining cloud computing, mobile edge computing and local computing, aiming at minimizing delay. In [3], authors propose a scheme to maximize user task completion within a tolerable time period while minimizing energy consumption, considering a network consisting of only one base station embedded with a MEC server, where the scheme determines the portion of the task to be offloaded to the server. The problem of computation offloading of vehicular applications is modeled in [4], where four Deep Reinforcement Learning (DRL) algorithms and four heuristics are compared in order to decide where each task should be executed. In [5], Lin *et al.,* formulate the computation offloading decision making problem as a MEC server selection problem, where the objective is to minimize the communication cost, defined as both transmitting energy consumption and communication delay. The work presented in [6] formulate the offloading decision as a multiclass classification problem and formulate the MEC resource allocation as a regression problem using a multi-task learning (MTL) based feedforward neural network (MTFNN) model.

In this paper, we consider a scenario where the resources demand (workload) suddenly increases in specific points of a MEC network based on optical-fiber point to point connections. This demand increase gives rise to the necessity of distributing the additional tasks among different MEC-DCs with the aim of maximizing the successfully offloaded tasks processed by the limited available resources. This paper addresses the mentioned problem considering a sparsely populated scenario where a MEC network based on optical fiber connections is planned for relatively low workloads. To test our proposed scheme, we simulated it over a network topology designed using the method proposed in [7], which presents a planning strategy for MEC networks in sparsely populated areas, minimizing the deployment cost of optical connections and MEC servers, and implementing a ring topology to interconnect MEC-DCs and the Wide Area Network Gateway (WAN Gateway).

III. MEC resource allocation problem

In MEC networks, users arrive at a determined time and request specific features for their connections depending on the service type they plan to use. In our problem definition, we consider the latency requirements of each application (user service), and the computing capacity that each of them needs.

Considering a scenario in which there are more users than their nearest MEC-DC is capable of attending simultaneously, we design an ILP formulation for maximizing the number of successfully processed tasks, considering the relation between the propagation delay experimented by the user and his latency limit.

The notation used in this model is summarized in Table 1. All symbols are parameters (previously known values), except S_{un}, which is a binary variable (value to be found) and represents if a task of user u is successfully performed in MEC-DC n.

TABLE 1: MODEL NOTATION

Symbol	Meaning
N	Number of MEC-DCs
U	Number of users
z_{ul}	$z_{ul} \in \{0,1\}, u \in [1,U], l \in [1,N]$. If $z_{ul}=1$ indicates that user u is in the local zone of MEC-DC l.
D_u	Distance limit of user u (kms).
d_{ln}	Distance in kms between MEC-DC l and MEC-DC n. $l \in N, n \in N$.
C_n	Processing capacity (MEC servers) of MEC-DC n. $n \in N$.
c_u	Processing requirement of user u. $u \in U$.
δ	Importance factor for relative delay.
S_{un}	Binary variable. $S_{un}=1$ indicates that the task of user $u$is successful at MEC-DC n.

According to the described notation, the objective of the formulation is given by (1). In essence we want to maximize the successful transmissions given by the sum of S_{un} along all users and MEC-DCs. The negative term in the formula aims to reduce the relative latency of each task. We define the relative latency of a task as the ratio between the distance d_{ln} to the allocated MEC-DC, and the maximum allowed latency D_u of the task. The parameter δ can be set according to the level of importance given to the delay compared to the successful tasks.

$$max\colon \sum_{l=1}^{N}\sum_{n=1}^{N}\sum_{u=1}^{U} z_{ul}S_{un}\left(1-\delta\frac{d_{ln}}{D_u}\right) \tag{1}$$

And (3)-(5) define the constraints of the model, described below.

$$S_{un} \in \{0,1\} \qquad \forall\, u \in U, n \in N \tag{2}$$

$$S_{un}z_{ul}d_{ln} \le D_u \qquad \forall\; u \in U, n \in N, l \in N \tag{3}$$

$$\sum_{n=1}^{N} S_{un} \le 1 \quad \forall\; u \in U \tag{4}$$

$$\sum_{u=1}^{U} S_{un}c_u \le C_n \quad \forall\; n \in N \tag{5}$$

According to (2), for each user u and MEC-DC n, S_{un} can be either zero or one. Constraint (3) makes sure that for a successful task, the distance between the user's nearest MEC-DC and the allocated one is not more than the distance limit for the specific user application. Constraint (4) states that every user will have at most one successful task in only one MEC-DC. Finally, constraint (5) only allows a task allocation in a MEC-DC if the center has enough processing power available.

IV. Case study: Allocation in Valladolid province

For testing the model, we used as basis a simulated network topology from a previous work [7], this topology is suitable for our purpose since it was proposed for a sparsely populated area, which is the Valladolid province of Spain, and given that it is one of our previous works, we have enough knowledge and data about it. Some data about Valladolid province can be found in Table 2. Valladolid province has many small villages with low population, which make this province a suitable case study for studying sudden peaks in networks planned for low workloads. In this study we assume that the workload of each BSs is proportional to the population near it. We combined a population dataset from [8] and a BS's location dataset from [9], and we generated a merged dataset which we used to our studies and which is available at [10].

For the performed simulations, we assume that a single MEC-DC has an overload, while the rest of them have some available capacity, hence, the load that cannot be handled locally must be forwarded to other locations. Note that both the model and the simulation can handle simultaneous overload of multiple nodes, but to make the results easier to show and interpret, in this paper we only study an example of one overloaded node.

TABLE 2. VALLADOLID PROVINCE CHARACTERISTICS

Characteristic	***Value***
Area (km²)	8111
Total population	517975
Main city's population	295639
Minimum population	18
Average population	2314

Fig. 1 depicts an illustrative example of the load distribution scheme. There is a set of Base Stations (BSs), some of which are equipped with a MEC-DC and the others are connected to the nearest MEC-DC through an optical fiber link (blue connections). All BSs with MEC-DCs are connected with each other in a ring topology according to [7] (red connections). A single node faces an overload (green cross), and some other MEC-DCs (green circles) assist in processing part of this overload, while some other nodes (red circles) do not assist.

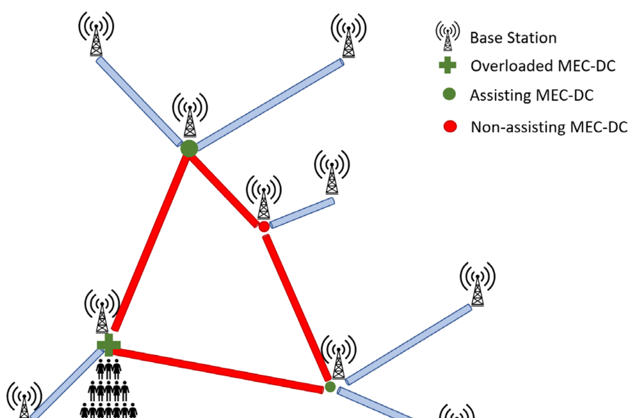

Fig. 1. Load distribution simplified scheme

For the simulation of our proposal, we work over a MEC network topology designed by the method proposed in [7] when it is configured for a peak capacity of 3% of simultaneously connected population. For our tests, we assume that 50% of the full capacity of servers is occupied, therefore when an overload occurs in one of the MEC DCs, there is available computing capacity in other nodes. We generate network traffic following a mixed traffic distribution of 70% of video traffic, 15% of car traffic, 10% of smart factory and 5% of augmented/virtual reality, and we used the latency and processing requirements of each task type according to [11]. In our simulation we consider the propagation latency constraint as a maximum distance in kilometers, this value is defined for each task type and corresponds to D_u in the formulation (Section III). Taking the required computing power of each type of task, and assuming that each MEC server can process 75 simultaneous users under the mentioned mixed traffic profile, we obtain the fraction of a server that each task needs, which corresponds to the value of c_u in the formulation. A summary of the described values can be seen in Table 2.

TABLE 3. TASK REQUIREMENTS

Service	**Requirements**	
	D_u (Maximum distance in km)	***c_u (Required server fraction)***
Vehicle collisions warning	1000	0.00019761
Video streaming	1000	0.00459559
Smart Factories	100	0.00889706
AR/VR	1000	0.18382353

To implement and test the model, we used the Python-based Pyomo optimization tool, with the IBM CPLEX optimizer. We ran simulations for different values of δ in the objective function (1). As explained in Section III, the greater the value of δ, the greater is the importance of the delay that the system considers. A graphic representation of load distribution after applying the formulation with δ=0 is presented in Fig. 2. In this test, an overload of 3200 users is generated in one MEC-DC (green cross), and this extra load is distributed among other MEC-DCs (green circles). In this representation the size of green circles is proportional to the external load that the respective MEC-DC is processing. In this case, all MEC-DCs are collaborating to handle the tasks of the overloaded node. A reference of 10 km is shown in the lower right corner of the figure to give an idea of the distances considered.

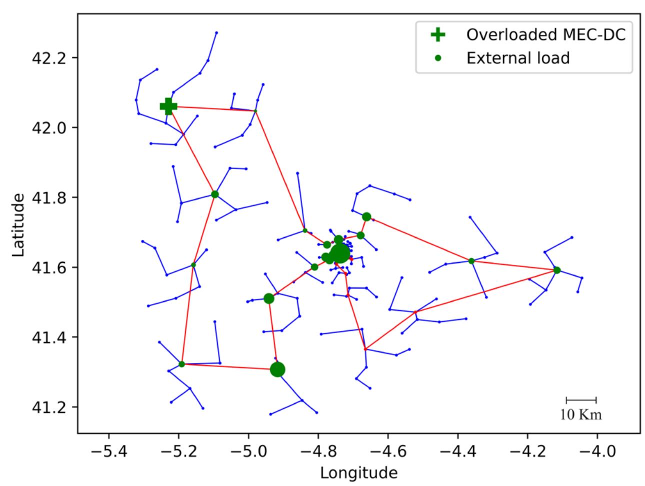

Fig. 2. Load distribution for δ=0

Another graphic result can be appreciated in Fig. 3. This time we ran the simulation for δ=5, meaning that the relative delay is considered for the optimization. By looking at the size of green circles it can be appreciated that the load is more concentrated in the MEC-DCs that are near the overloaded one, there are even some nodes that are not assisting with the overload at all (red points). This is due to the effect of δ, which in this case, different from previous one, makes the system to prioritize the offloading with shorter distances.

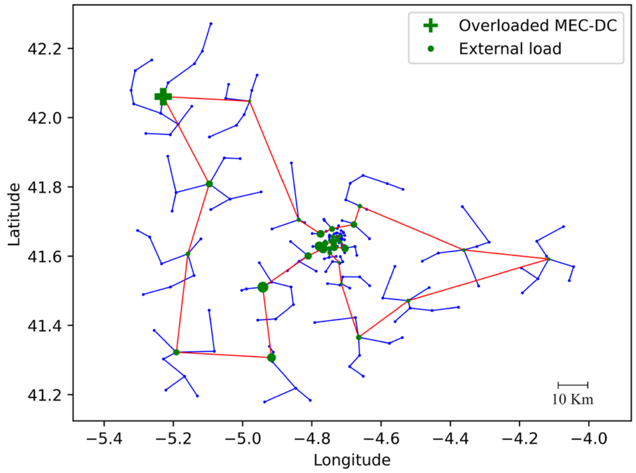

Fig. 3. Load distribution for δ=5

Different test results are illustrated in Fig. 4. In the vertical axis, the tasks success rate is shown, and the horizontal axis includes different values of additional users in logarithmic scale. We ran simulations with different numbers of additional users (overload) for four techniques: the blue plot corresponds to a scheme without collaboration between MEC-DCs to handle the overload, in such a way that all BSs are only able to offload their tasks to their nearest MEC-DC. The network topology we used for our tests has 32 MEC-DCs, and we ran a simulation placing the overload in each one of them after which we computed the average and the 95% confidence intervals, depicted with horizontal bars in figures 4 and 5. As can be expected, the success rate decreases rapidly with the overload increment.

Results are noticeably different for the other three approaches in the figure. Red, green and yellow lines represent respectively the results of the ILP formulation for δ=0, δ=2, and δ=5. The most remarkable aspect is the improvement in success ratio when the formulation for collaborative offloading is applied. By the point where the additional users are 12800, the success rate of the "Nearest MEC" approach, has dropped to nearly zero, while for the three cases of the formulation, the success rate has only started to decrease. It is also worth mentioning that, as δ increases, the success ratio for large amounts of users slightly decreases, because the system gives importance to the relative delay, sometimes at the expense of successful tasks.

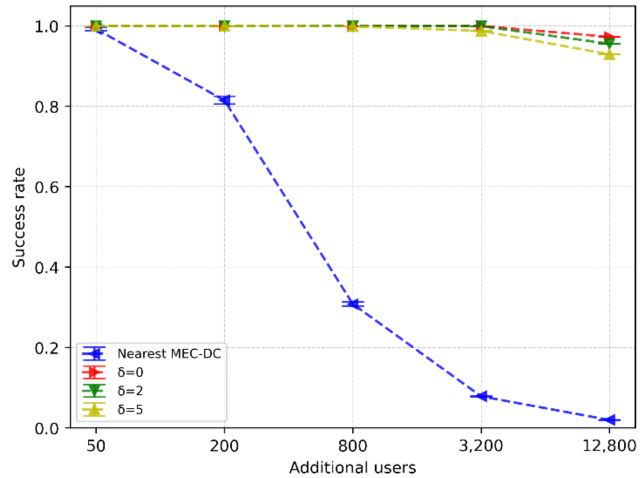

Fig. 4. Success rates vs additional users

The obtained releative delay after running the simulations were measured for each user, and the average of these values were calculated. In Fig. 5, the average relative delays for the different approaches is shown. The "nearest MEC" approach has zero delay because all tasks are processed in the local node, but at the expense of considerably less successful tasks as we saw in Fig. 4. The different options of the formulation present lower delay values as δ increases. Remind that these values are relative delays, i.e., the ratio between the obtained distance and the maximum allowed, and from constraint (3), we know beforehand that all successful tasks fullfill the latency requirement, that is why all values in Fig. 4 are smaller than one.

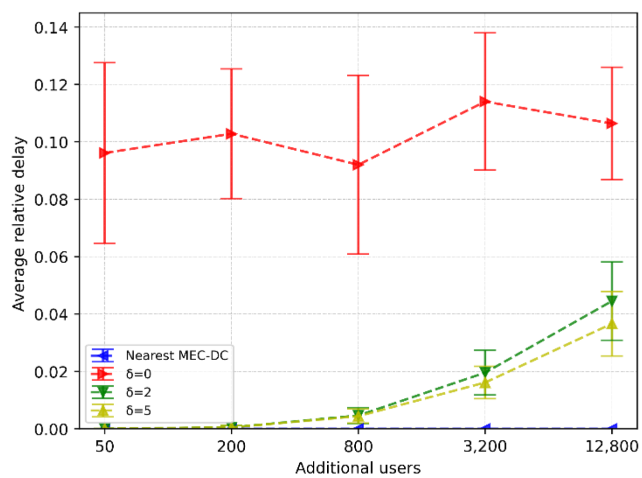

Fig. 5. Average relative delay vs additional users

ILP models solving time increases with the size of the problem. In this case, when the number of additional users grow, the required time to solve the model is also increased as Fig. 6 shows for the different values of δ. Even though all options take more time for higher number of users, the model with δ=0 presents a notable increment in time against the others (note the logarithmic scale in both axis). This result suggests that including the delay factor in the objective function (making δ>0) is convenient because, at the expense of a relatively small decrement in successful rate (Fig. 4), it reduces the relative delay (Fig. 5), as well as the solving time when the number of additional users is high.

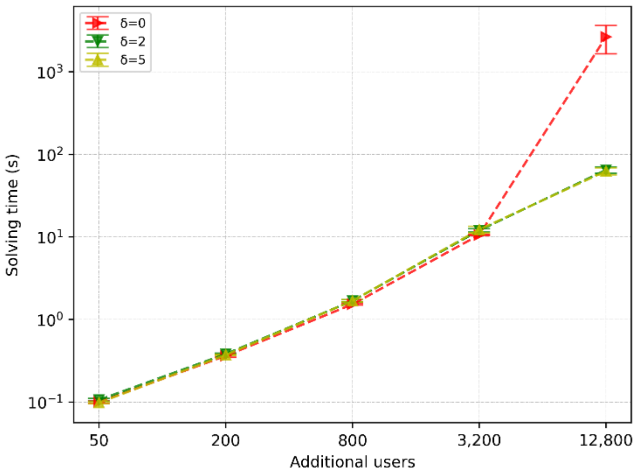

Fig. 6. Solving time vs additional users

V. Conclusion

MEC networks are designed to work under determined conditions. Regarding computing power (number and capacity of servers) there are limitations that essentially depend on the estimation of the workload that each MEC-DC will face during its operation. However, under some circumstances, the estimated workload is exceeded in determined zones of the network, e.g., social events or other crowds. These circumstances are easier to arise when it comes to sparsely populated areas where the networks are envisaged for handling a relatively low number of tasks. In this work we designed and validated a model in which the MEC-DCs of a previously designed network can collaboratively share the excess workload in such cases. Results show that the proposed approach considerably increases the success rate of tasks when an overload occurs in a node of the network, while maintaining the latency requirements of MEC applications. Moreover, by adjusting one parameter of the system, it is possible to tune the importance of the successful tasks against the relative delay. The proposal of this paper is developed under the assumption that there is a central entity that knows the state of all MEC-DCs and the requirements of all users, in such a way that this entity can solve the ILP model and forward the tasks to the more convenient MEC-DC for their processing. ILP gives the optimal solution to the problem assuming that the model is accurate, however, ILP solving is a time-consuming process, hence it is important to pay attention to the delay induced by this process for the task completion, especially regarding applications requiring a fast response, such as vehicle collisions warnings and smart factories. Two aspects are proposed for improving the study in future works: First, to implement a heuristic technique that guarantees a solution in reasonable time, and second, to consider a more realistic delay model to consider aspects such as processing and scheduling time, rather than only propagation time.

References

[1] Q.-V. Pham *et al.*, “A Survey of Multi-Access Edge Computing in 5G and Beyond: Fundamentals, Technology Integration, and State-of-the-Art,” Jun. 2019, [Online]. Available: http://arxiv.org/abs/1906.08452

[2] Y. Cui, Y. Liang, and R. Wang, “Intelligent Task Offloading Algorithm for Mobile Edge Computing in Vehicular Networks,” in *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, IEEE, May 2020, pp. 1–5. doi: 10.1109/VTC2020-Spring48590.2020.9128803.

[3] S. Birhanu Engidayehu, T. Mahboob, and M. Young Chung, “Deep Reinforcement Learning-based Task Offloading and Resource Allocation in MEC-enabled Wireless Networks,” in *2022 27th Asia Pacific Conference on Communications (APCC)*, IEEE, Oct. 2022, pp. 226–230. doi: 10.1109/APCC55198.2022.9943689.

[4] M. Ferens *et al.*, “Deep Reinforcement Learning Applied to Computation Offloading of Vehicular Applications: A Comparison,” in *2022 International Balkan Conference on Communications and Networking (BalkanCom)*, IEEE, Aug. 2022, pp. 31–35. doi: 10.1109/BalkanCom55633.2022.9900545.

[5] X. Lin, H. Zhang, H. Ji, and V. C. M. Leung, “Joint computation and communication resource allocation in mobile-edge cloud computing networks,” in *2016 IEEE International Conference on Network Infrastructure and Digital Content (IC-NIDC)*, IEEE, Sep. 2016, pp. 166–171. doi: 10.1109/ICNIDC.2016.7974557.

[6] B. Yang, X. Cao, J. Bassey, X. Li, and L. Qian, “Computation Offloading in Multi-Access Edge Computing: A Multi-Task Learning Approach,” *IEEE Trans Mob Comput*, vol. 20, no. 9, pp. 2745–2762, Sep. 2021, doi: 10.1109/TMC.2020.2990630.

[7] C. Anzola-Rojas *et al.*, “Improving Joint Planning of MEC and Fiber Deployment with Duct and Optical Cable Sharing,” in *2022 IEEE International Mediterranean Conference on Communications and Networking, MeditCom 2022*, 2022. doi: 10.1109/MeditCom55741.2022.9928653.

[8] Instituto Nacional de Estadística, “Valladolid: Población por municipios y sexo. (2904).” Accessed: Mar. 27, 2023. [Online]. Available: https://www.ine.es/jaxiT3/Datos.htm?t=2904#!tabs-grafico

[9] Ministerio de asuntos económicos y transformación digital, “Niveles de Exposición.” Accessed: Mar. 27, 2023. [Online]. Available: https://geoportal.minetur.gob.es/VCTEL/vcne.do

[10] C. Anzola-Rojas, “GitHub - GCOdeveloper/CyL-base-stations-dataset.” Accessed: May 27, 2023. [Online]. Available: https://github.com/GCOdeveloper/CyL-base-stations-dataset

[11] F. Spinelli and V. Mancuso, “Toward Enabled Industrial Verticals in 5G: A Survey on MEC-Based Approaches to Provisioning and Flexibility,” *IEEE Communications Surveys and Tutorials*, vol. 23, no. 1, pp. 596–630, Jan. 2021, doi: 10.1109/COMST.2020.3037674.