Prescriptive Analytics for MEC Orchestration

Alberto Ceselli, Marco Fiore, Angelo Furno, Marco Premoli, Stefano Secci,
Razvan Stanica

HAL Id: hal-01740816

https://hal.sorbonne-universite.fr/hal-01740816v1

Submitted on 22 Mar 2018

Prescriptive Analytics for MEC Orchestration

Alberto Ceselli*, Marco Fiore†, Angelo Furno‡, Marco Premoli*, Stefano Secci§, Razvan Stanica¶

* Università Degli Studi di Milano, Dept. of Computer Science, Crema, Italy. Email: {firstname.lastname}@unimi.it
† CNR–IEIIT, Torino, Italy. Email: marco.fiore@ieiit.cnr.it
‡ Univ Lyon, IFSTTAR, ENTPE, LICIT UMR_T9401, F-69675, Lyon, France. Email: angelo.furno@ifsttar.fr
§ Sorbonne Université, CNRS, LIP6, F-75005 Paris, France. Email: stefano.secci@sorbonne-universite.fr
¶ Univ Lyon, INSA Lyon, Inria, CITI, F-69621, Villeurbanne, France Email: razvan.stanica@insa-lyon.fr

Abstract—Orchestrating network and computing resources in Mobile Edge Computing (MEC) is an important item in the networking research agenda. In this paper, we propose a novel algorithmic approach to solve the problem of dynamically assigning base stations to MEC facilities, while taking into consideration multiple time-periods, and computing load switching and access latency costs. In particular, leveraging on an existing state of the art on mobile data analytics, we propose a methodology to integrate arbitrary time-period aggregation methods into a network optimization framework. We notably apply simple consecutive time period aggregation and agglomerative hierarchical clustering. Even if the aggregation and optimization methods represent techniques which are different in nature, and whose aim is partially overlapping, we show that they can be integrated in an efficient way. By simulation on real mobile cellular datasets, we show that, thanks to the clustering, we can scale with the number of time-periods considered, that our approach largely outperforms the case without time-period aggregations in terms of MEC access latency, and at which extent the use of clustering and time aggregation affects computing time and solution quality.

I. INTRODUCTION

The softwarization of networks is an innovative trend expected to transform the mobile access environment in the coming years. It is an evolution accompanied by the virtualization of network functions and application servers, which can be operated running virtualization clusters close to, or at cellular base stations and mobile network points of presence [1]. The type of functions that can be virtualized ranges from traffic load balancers and multimedia (de)coders to mobile core functions such as those of the Long Term Evolution (LTE) Evolved Packet Core (EPC) [2]. Application servers can also be run in such facilities, so that the end-to-end user experience benefits from low access latency [3].

An illustration of this evolution is given in Figure 1. Figure 1a depicts a legacy 4G environment, where the user accesses remote applications via cellular access, in such a way that its wireless signals are processed at Base Band Unit (BBU) nodes integrated to cellular Base Stations (BSs), its traffic is routed through the EPC (composed of four main functions), before reaching the Internet border on the way to the application server. Figure 1b shows instead a fully cloudified environment, where radio-network elements such as the BBU, EPC functions, mobile phone remotely executable applications, as well as application servers (possibly synchronized with a remote cloud) are all virtualized in

(a) Legacy 4G access network.
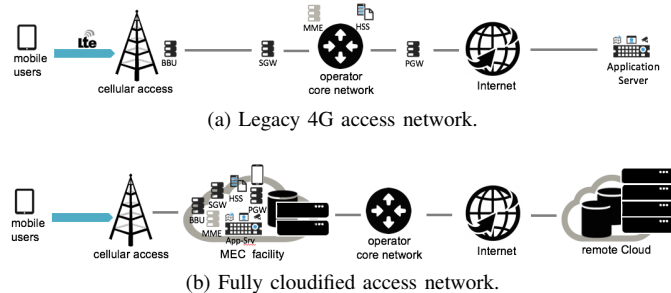

(b) Fully cloudified access network.

Fig. 1: Mobile access network evolution with edge computing.

potentially the same place, called Mobile Edge Computing (MEC) facility. Such a scenario is an extreme one, coping with the virtualization of a variate set of hardware, but that could correspond to the reality in the coming decade. In any case, the virtualization of a subset of these functions is a certainty, as demonstrated by different ongoing projects in the industry, for instance those regarding the virtualization of EPC functions (as announced by Orange Spain in fall 2017), of radio-access network elements (as announced by China Mobile in 2011), or of application servers (as encompassed in some reference MEC use-cases [1]).

Among the virtualizable nodes at MEC facilities, we can distinguish nodes that are strictly serving a subset of the BSs of an operator (e.g., vBBU and vEPC nodes), and nodes that serve single or multiple users (e.g., virtualized mobile device environment for computation offloading, virtualized application servers), possibly behind different BSs. An important amount of traffic can therefore be aggregated at MEC facilities, depending on the type of virtualized functions that are run at these edge delivery points. The management of virtualized nodes running at MEC facilities encompasses service and network management operations mainly related to: *i)* BS-to-MEC facility association, and *ii)* user-to-virtual machine (VM) association (a VM being in turn associated to a MEC facility). This kind of association decisions imply the execution of VM-level MEC orchestration operations, such as VM scaling up/down (increase/decrease of computing resources such as memory, processor, storage), Virtual Network Function (VNF) scaling in/out (more or less VM instances running a given VNF), VM migration, VM creation or destruction.

At the time being, the telco industry is more focused on the virtualization of the nodes that serve a subset of cellular

antennas (e.g., vBBU, vEPC), instead of working directly at the user-VM granularity, mainly because of scalability concerns. Therefore, one shall consider BS-to-MEC facility switching decisions as critical ones. In this paper, we indeed propose a MEC orchestration framework that primarily optimizes BS-to-MEC facility association over time, based on a spatiotemporal grouping of the BSs, while integrating VM workload adaptations across MEC facilities.

BS-to-MEC facility switching operations can not be reasonably expected to run continuously, as this would incur in traffic loss and overhead due to traffic handover, but to occur only at certain points in time (*e.g.*, once every thirty minutes). Hence, introducing an implicit time discretization of the orchestration system appears appropriate. In order to identify suitable discrete-time profiles of the traffic demand, different strategies can be employed. The simplest option is to aggregate the demand observed at each BS during every time step in a recent reference period, exploiting training data, using one reference profile for each time step. Another option to identify suitable discrete-time profiles of the traffic demand is to use temporal clustering analytics on the historical data, so as to group together time slots that feature very similar distributions of the mobile traffic demand across the BSs.

In this paper, we explore the two options above, proposing a prescriptive analytics approach integrating advanced temporal clustering into a mathematical programming formulation of the addressed MEC orchestration problem. The clustering returns a limited number of profiles, each of which corresponds to time intervals where the mobile network presents a similar distribution of the demand. It is then possible to feed the optimization framework with a small number of profiles, with the risk of decreasing the solution quality, since typical profiles can only approximate the actual MEC network load at a specific time step. We assess in the paper the computational and quality aspects of our prescriptive analytics approach, as compared to basic time aggregation in the orchestration.

The manuscript is structured as follows. Section II draws the necessary background. Our network model is described in Section III, while our orchestration algorithm is described in Section IV. Section V reports numerical results. Section VI concludes the paper.

II. BACKGROUND

We draw in this section the necessary background on edge computing and virtualization and on the integration of data analytics in network optimization problems.

A. Edge computing and network virtualization

In a MEC infrastructure, virtualization clusters – called 'MEC facilities' or 'MEC hosts' in the standardization documents [1], or 'cloudlets' in academic jargon [5] – are connected to access network nodes within a few hops, to deliver access to application servers running as VMs. Various operations dealing with the changing mobile access demand can be applied to orchestrate the resulting cloud-network system, which include BS to MEC facility dynamic assignment, VM

capacity rescaling (addition or removal of computing power in terms of live memory or virtual processors) and VM migration (a VM state is moved from one MEC facility to another one). An 'orchestrator' is in charge of instantiating such decisions into the MEC infrastructure. Each orchestration action comes at a cost, often referred to as 'migration' or 'switching' cost, as it can require synchronizing states and reconfiguring network equipment and servers, across a geographical network under stringent performance guarantees. The technology to perform MEC orchestration operations is being experimented since many years [6]. It commonly takes into consideration changing states of the network in time and space, related to user mobility and digital usages behavior.

These dynamics are being considered for the management of not only application servers, but also of the network services needed to deliver resilient access to applications. Indeed, 5G systems will also build on new networking paradigms such as Network Function Virtualization (NFV) and Software Defined Networking (SDN) in order to, on the one hand, support the orchestration of virtualized network functions and, on the other hand, provide to core network switches the necessary features to support flow management that may be needed when applying fine-grained orchestration decisions [7].

Eventually, for mobile access networks and in particular cellular networks, the physical facility delivering application and network function VMs is expected to be the same, as already discussed in Figure 1, located in access network aggregation points of presence. Such a convergence is also clearly appearing in standardization efforts related to MEC and NFV systems [8], [9], with equivalent interfaces between virtualization layer and orchestration system components.

A significant amount of work exists in the area of MEC and mobile-access NFV orchestration. A common problem addressed is the virtualization cluster placement within the access network, as considered in [10] for application VMs, in [4] for the EPC functions and in [11] for radio-access functions. A different orchestration dimension is the one related to VM migration and rescaling across a given set of MEC facilities, as a function of user mobility, as addressed in [12] for application VMs and in [13] for the EPC functions. Finally, in the area of virtualized radio-access network orchestration, the problem of clustering, i.e., assigning a set of BSs to BBUs was also extensively studied, as for instance in [14].

B. Data-driven mobile networks

A further step in this area, only marginally addressed to date, is to investigate how to integrate the result of data analytics in the instrumentation of MEC orchestration decisions, related to placement, migration, rescaling and clustering operations, along the lines traced in [15], [16].

Virtualized networks where significant resources are placed in proximity of the radio access open substantial new scenarios for the dynamic management of system operations. Solutions based on data analytics are in particular expected to play a critical role: knowledge inferred by mining traffic measurements and Key Performance Indicators (KPIs) will fuel effective

orchestration policies for the deployment and re-allocation of resources across mobile edge computing facilities. The vision of 'data-driven' (also referred to as 'cognitive') network management is attracting the interest of a growing research community [17], [18], and is supported by major players in the 5G ecosystem [19].

Due to the very recent emergence of relevant use cases, solutions to extract useful information from massive amounts of mobile traffic data records and to employ it for network configuration are still in their infancy. Data analytics for mobile network traffic based on clustering or spectral analysis have revealed regular macroscopic structures [20], [21] that are highly predictable [22]. Actual experiments of data-driven network management have mainly focused on optimizing video streaming services [23], [24] and controlling core network congestion [25], [26]. However, as of today there is almost no practical demonstration of how MEC can benefit from data-driven paradigms. The single application we could identify is the data-driven BBU-to-BS clustering approach in [27], where however the interconnection network is not modeled.

In this paper, we present a first application of data-driven networking in the context of MEC orchestration, and more precisely clustering decisions, considering both network and systems constraints. Specifically, we leverage existing analytics for the spatiotemporal classification of traffic, and extract long-timescale patterns in the spatial distribution of the mobile traffic demand. We then employ these patterns to guide the operation of MEC facilities so that the user Quality of Service is maximized, by their integration in orchestration algorithms based on mathematical programming.

C. Network optimization

The orchestration problem we address is to find groups of BSs for their association to MEC facilities, in a multi-period setting such that the BS-to-MEC facility association can change across periods. In the area of network optimization, this requires to tackle a multi-period extension of the famous Generalized Assignment Problem (GAP) [28].

We point to [29] for a detailed review on the GAP and its extensions. Despite the large body of research available on the GAP, we are not aware of many papers directly dealing with its multi-period extensions. In [30], the authors face a single-source allocation problem with a flexible model and an effective algorithm; however, their model does not handle limited capacity, which is a crucial feature in our application. The multi-period allocation problem discussed in [31], in which a dual ascent technique is adapted to telecommunication networks applications, is similarly missing the handling of capacities.

Although our problem does not require to decide the location of the facilities, which is instead assumed to be optimized in a prior strategic planning [10] and given as input, one may expect features and computational challenges similar to those of multi-period location problems [33]. Recent approaches on that field include [34]: the authors face a multi-period concentrator location and dimensioning problem, providing

MILP formulations and reduction techniques, and solving to optimality in less than one hour of computation instances with up to 30 clients, 10 candidate location sites and 15 time periods, or 100 clients, 30 candidate locations and 5 time periods. In [35] the authors introduce exact methods for a capacitated multi-period facility location problem in which, however, unlike our case, the demand of each client can be fractionally served by multiple facilities. Large scale instances with up to 200 facilities, three periods and an arbitrary number of clients could be solved with their algorithms.

III. MEC NETWORK ORCHESTRATION MODEL

We elaborate our reference MEC network orchestration model along the following generic lines. BSs have associated mobile traffic demand, that changes over time. Each MEC facility has a certain capacity, limiting the overall amount of demand it can serve simultaneously. BSs must be assigned to MEC facilities; each new assignment implies a cost for each user connected to the BS in terms of latency for communicating with the associated MEC server. Due to capacity limits, it might not always be a good decision to assign each AP to its MEC facility of minimum latency; furthermore, since demand changes over time, an assignment pattern would hardly remain an efficient one over the whole planning horizon. We therefore leave the option of changing assignments over time, taking into account that each change implies a switching cost for the network, for example in terms of signaling to move session data of active users. An optimization problem therefore arises, that is to assign BSs to MEC facilities over time, respecting capacity constraints and minimizing a combination of users (assignment) and network (switching) costs.

Before providing a more formal problem statement and mathematical formulation, we describe the data analytics problem we address to instrument the orchestration algorithm.

A. Data analytics

The data analytics we adopt to drive our resource orchestration problem is inspired by the temporal classifier of mobile network traffic introduced by [37]. The classifier leverages an agglomerative hierarchical clustering with fine-tuned distance measures, and allows detecting long time periods during which the geographic distribution of the mobile traffic demand does not vary significantly. The results presented in the original paper show that, *e.g.*, the aggregate demand of voice calls and text messages switches among a very small number of possible spatial configurations during a whole week.

We employ the classifier above as a building block, and proceed through the 4 phases, also summarized in Fig. 2:
Phase 1: We collect substantial measurement data from an operational mobile network. The data captures the demand for a major mobile service in two large-scale metropolitan regions for a period of several consecutive months. Details on the data collection are provided in Sec. V-A.
Phase 2: For a subset of the collected data, representing our training set, we compute the typical weekly average demand, by aggregating all data collected at the same time of the week.
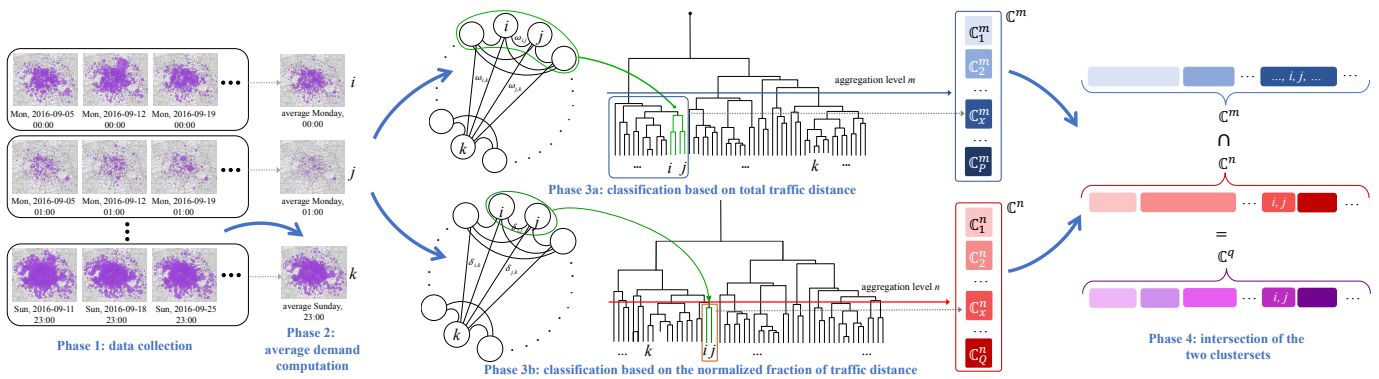
Fig. 2: Workflow for the classification of network usage profiles. The final intersection cluster set $\mathbb{C}_q$ is used for the training.
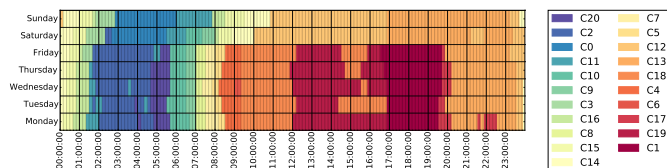

Fig. 3: Sample cluster set of 10-minute time instants for a mobile service. The plot outlines the existence of 20 temporal classes of spatial distributions of the demand, during each daytime (abscissa) of different weekdays (ordinate).

For instance, the representative offered load on Monday, 4:00 pm at one antenna is the average of all measurements on Mondays in the training set, at that specific time and antenna. Clearly, time needs to be discretized in order to obtain a finite set of time instants: we thus assume that each time instant refers in fact to a period of duration T.

Phase 3: Following the methodology suggested in [37], we run two separate instances of the classifier on the average week, considering two distance metrics to compute the similarity of demands at diverse time instants. The first such metric is the difference of total traffic volumes, which tends to cluster together time instants with equivalent total demands. The second metric is the difference of the normalized fraction of traffic at each antenna, which groups together time instants that feature comparable spatial distributions of the demand.

Phase 4: We derive the intersection of the two cluster sets, obtaining our final set of time instant classes. The rationale is that such an intersection yields classes that have *i)* equivalent total traffic volumes, that are *ii)* distributed in the same way across antennas. In other words, the demands in time instants that belong to the same class are similar from all viewpoints.

Figure 3 shows an example of the final cluster set for our reference mobile service, when considering that each time instant spans $T = 10$ minutes. In the specific case under study, our approach categorizes all 10-minute time periods in a week into just 20 classes, *i.e.*, spatial configurations of the mobile service demand. The fact that 20 classes capture the diversity of offered loads in more than 1,000 time instants underscores how the demand for our target mobile service shows significant

regularity over time. The emergence of 20 classes is also good news for our case study, as it implies that a small number of MEC facility deployments can be sufficient to accommodate all possible spatial dynamics in the traffic.

Another interesting observation is that time instants in a same class are typically contiguous. Also this aspect plays in favor of our objective: the temporal consistence of spatial configurations entails that MEC resource allocation profiles remain valid throughout quite long timespans, and the number of switches between profiles is reduced. The expectation (confirmed by our numerical evaluation) is that these intrinsic properties of the mobile service demand can make a data-driven approach for MEC deployment highly effective.

B. Orchestration Optimization Model

Our MEC orchestrator includes an optimization core for performing prescriptive analytics on a tactical level. We adapt models and methods from [38]. In particular, we build dynamic assignment plans detailing, for each time slot, the set of BSs to be connected to each MEC facility and, as a by-product, the set of switching operations to be performed between subsequent time slots. We consider a periodic single-assignment operational policy, that is, in each time slot each BS is assigned to exactly one MEC facility, and the last time slot is assumed to be followed by the first one.

The task details are the following.

Input. We assume to be given the set of BSs, the set of MEC facilities and a discretization of the time horizon in a set T of time slots. We also assume to be given *i)* for each BS, the mobile traffic demand that has to be accommodated in each time slot, *ii)* the capacity of each MEC facility, *iii)* the physical distance between each BS and each MEC facility and the network distance between each pair of MEC facilities (that is, a measure directly proportional to the network latency, including packet processing latency at intermediate nodes, and physical distance).

Output. We expect, as output of the optimization core, an assignment plan: for each BS and each time slot, an indication of the MEC facility where traffic needs to be routed. As a side result, we expect a switching plan, that is a boolean value for each BS and each pair of MEC facilities for each time slot,

indicating whether that BS switches at that time between a particular pair of MEC facilities, or not.

Requirements. The assignment plan satisfies the following conditions: *i)* the overall demand assigned to each MEC facility at each time slot must not exceed its capacity, *ii)* each BS is connected to exactly one MEC facility at each time slot, *iii)* assignment and switching plans must be coherent.

Objective. The plans must target a trade-off between the minimization of network- and user-related costs. The former is generated by the change of BS-MEC facility associations in consecutive time slots, which produces some overhead due to the necessity of migrating VMs. The latter is instead the latency experienced by the user with the current BS-MEC facility association. The relative weight of the network- and user-related costs in the objective function is represented by suitable parameters, set to equal weights in our experiments.

A sample instance with three APs (squares), two MEC facilities (circles) and two time-slots (left and right parts) is depicted in Figure 4: AP 2 is assigned to MEC facility A at $t = 1$ and MEC facility B at time $t = 2$, therefore a switching operation from A to B needs to be performed.

Formally, our orchestration task can be modeled with the following Mathematical Program:

$$\min \alpha \sum_{t \in T} \sum_{i \in A} \sum_{\substack{(j,k) \in \\ K \times K}} d_i^t l_{jk} y_{ijk}^t + \beta \sum_{t \in T} \sum_{i \in A} \sum_{k \in K} d_i^t m_{ik} x_{ik}^t \tag{1}$$

$$\text{s.t.} \sum_{i \in A} d_i^t x_{ik}^t \leq C_k \qquad \forall t \in T, \forall k \in K \tag{2}$$

$$\sum_{k \in K} x_{ik}^t = 1 \qquad \forall i \in A, \forall t \in T \tag{3}$$

$$x_{ik}^t = \sum_{l \in K} y_{ilk}^t \qquad \substack{\forall i \in A, \forall k \in K \\ \forall t \in T \setminus \{1\}} \tag{4}$$

$$x_{ik}^t = \sum_{l \in K} y_{ikl}^{t+1} \qquad \substack{\forall i \in A, \forall k \in K \\ \forall t \in T \setminus \{T\}} \tag{5}$$

$$x_{i,k}^t \in \{0,1\} \qquad \substack{\forall i \in A, \forall k \in K \\ \forall t \in T} \tag{6}$$

$$y_{i,k',k''}^t \in \{0,1\} \qquad \substack{\forall i \in A, \forall t \in T \\ \forall k', k'' \in K} \tag{7}$$

where A is the set of BSs, K is the set of MEC facilities, T is the set of time slots, d_i^t is the demand of BS $i \in A$ during time slot $t \in T$, C_k is the capacity of MEC facility $k \in K$ and l_{jk} (resp. m_{ik}) is the distance from facility j to facility k (resp. from BS i to facility k). Assignment plans are encoded by variables x_{ik}^t, which take value 1 if BS i is assigned to facility k at time t, 0 otherwise. Switching plans are encoded by variables y_{ijk}^t, which are 1 if traffic from BS i must be switched from facility j to facility k at time t, 0 otherwise. Constraints (2) and (3) model requirement *i)* and

ii), respectively. Collectively, constraints (4) and (5) ensure assignment and switching plans to be coherent. The first term in the objective function (1) models switching costs, while the second term models assignment costs.
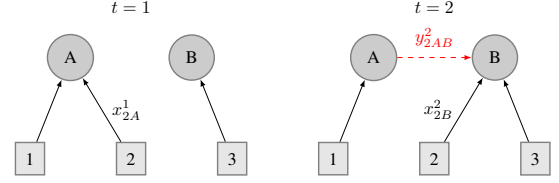

Fig. 4: x and y variables

IV. RESOLUTION ALGORITHM

Problem (1) – (7) is NP-Hard in general [29]. When the size of the MEC network is large, the resolution of the orchestration problem requires ad-hoc algorithms. A decomposition approach can be employed: when the requirements on MEC facility capacities are considered in 'soft' form, penalizing their violation with suitable multipliers in the objective function, the problem disaggregates in independent subproblems per BS.

In particular, when a penalty multiplier is fixed for each capacity unit violation, a minimum cost path problem in a suitable graph needs to be solved independently for each BS; these problems admit polynomial time algorithms, and can therefore be computed efficiently also on large scale networks and fine grained time discretizations.

In turn, we are able to obtain a set of optimal multipliers through Dantzig Wolfe Reformulation and Column Generation methods [38]. Strictly speaking, the solution found by means of Dantzig Wolfe Reformulation might be fractional, that is some BSs might be fractionally assigned to more than a single MEC facility in some time slots. In that case, we run a heuristic, selectively rounding these fractions, and thereby always choosing a single MEC facility. Our heuristic works as follows. We iterate over each time slot in sequential order. In each time slot, for each BS, the MEC facility having highest fractional assignment is retrieved: we sort the set of BSs according to these assignment values (from highest to lowest). Then, we iterate over each BS following this order, assigning a single MEC facility to each BS; whenever an integer assignment would yield a capacity violation, an alternative MEC facility is selected for that BS, still in order of non-increasing fractional assignment values in the particular time slot. Our computational experiments revealed this heuristic to be highly effective.

Furthermore, we employ several techniques for speeding up our algorithms, like the use of Lagrangean fixing procedures to reduce the search space. In particular, we initialize column generation using greedy heuristics: for each time slot, BSs are sorted by non-increasing demand and each BS is associated to a profitable MEC facility. The most profitable MEC facility is considered to be that involving no switching cost, if enough residual capacity is available, or the nearest one with enough residual capacity, otherwise.

We implemented our algorithms in C++, using CPLEX 12.6 [39] to solve the master LP subproblems, running tests on an Intel i7 4GHz workstation equipped with 32 GB of RAM. Before describing the results on orchestration time period assessment and MEC performance, we describe the dataset.

A. Dataset

The dataset used in our study was collected in the core network of Orange, a major European mobile operator, during three months in 2016. It describes the traffic generated by several millions of mobile subscribers in the French metropolitan areas of Lyon and Paris, for a specific mobile service, *i.e.*, Facebook. More precisely, the data was recorded by monitoring IP sessions at the 3G and 4G core network gateways. A combination of Deep Packet Inspection (DPI) and proprietary fingerprinting tools was employed to infer application-level information on Facebook user sessions. The approach allows to determine the volume of all content traffic related to the Facebook mobile service, including streaming content or messaging, accessed through the app or web interfaces.

The rationale for the choice of Facebook is that it represents a prominent mobile service, generating around 20% of the compound downlink and uplink demand in the network. It is also an example of typical service that could benefit from the improved quality of service granted by a MEC infrastructure.

It is important to remark that the traffic is aggregated at the antenna sector level. This ensures that the information in the dataset is a combination of the Facebook sessions of many users, hence it does not contain personal data or raises privacy issues. The dataset is composed of twelve weeks of traffic demands, aggregated by 10-minutes time-periods.

B. Time-period granularity

In order to identify suitable demand discrete-time profiles, we evaluated six different time-period aggregations:

- aggregating consecutive 10-minute periods to form a period of four, two and one hour ('4H', '2H' and '1H' in the remainder). The resulting training sets consist of 42, 84 and 168 time-periods, respectively. Using shorter time periods for this strategy revealed to be too complex to solve; at the same time we found tests on longer time periods not informative, as already the 4H case is dominated by 2H and 1H aggregations, and the experimental trend is clear, as discussed in the remainder;
- aggregating 10-minute periods belonging to the same clustering profile generated as presented in Subsection III-A, with clustersets of one hour ('1HC'), 30-minute ('30MC') and 10-minute ('10MC'); the resulting training sets differs for Lyon and Paris dataset: for Lyon dataset, time-periods are 99 for '1HC', 171 for '30MC' and 260 for '10MC'; for Paris dataset, time-periods are 60 for '1HC', 160 for '30MC' and 141 for '10MC'.

We observe that the number or time-periods in 2H is similar to that of 1HC. The same can be observed for 1H and 30MC.

For the training set, we use the first 4 weeks of the dataset to build the typical average week used by the clustering approach in the classification process. As recommended in [37], we also tested the construction of the typical week using the median value of the demand, but the results only marginally differed with respect to the average week, so we avoid reporting them.

Lyon dataset contains demands from 332 BSs, while Paris dataset has 1907 BSs. We set three cardinalities of facilities for the dataset of Lyon (10, 20 and 30, resp.) and two cardinalities of facilities for the dataset of Paris (20 and 50, resp.). The location of the facilities was generated by a k-medoid algorithm, using the coordinates of the BS locations as input data. Distances between BS and facilities were computed using the Haversine formula [40]. Parameters α and β of objective function (1) were both set to 1. The resulting training set is composed of 60 instances.

a) Benchmark: As benchmark for our methodology, we considered a baseline approach without the time-period aggregation we propose with our model, therefore with a single time-period, leading to a single assignment for every BS to a MEC facility over the week and no switching of assignment among MEC facilities during the week. We computed a single time-period demand averaging demands of all time-periods in our dataset; we used this single-time average demand to train our model for every city and every facility cardinality (5 training instances). We solved the corresponding problem with the ILP general solver of CPLEX, stopping the resolution when an optimality gap lower than 1% was reached. We label such instances as 'S' in the remainder.

b) Training Computational Results: In Figure 5 we present the box-plots of the execution times of the training sets, in logarithmic scale (base 10), highlighting each the different time-period granularity (a boxplot shows a box bars indicating the minimum, 1st quartile, median, 3rd quartile, maximum). We can notice that the consecutive 1H case has the highest execution time (up to 10 hours of executions), followed by the 30MC case, while a lower time is required by 4H and 2H and 1HC, with S as fastest approach. In addition to the plot, we found that the average execution time for 'S' is 10 seconds, for '4H', '2H' and '1HC' is less than five minutes (165.1 s, 273.5 s and 268.8 s, resp.), for '1H' is more than 2 hours (9022.6 s), for '30MC' is slightly more then 1 hour (4515.6 s) and finally for '10MC' half an hour (2012.9 s). Having similar number of time-periods, '2H' and '1HC' (resp. '1H' and '30MC') require similar training time.

In Figure 6 we present the box-plots of the optimality gap of the training sets, still highlighting each the different time-period granularity. As specified previously, 'S' training was stopped as soon as an optimality gap less than 1% was reached. We can notice little difference among different aggregations: with respect to the worst-case performance, '4HC' has the worst result, while the clustering cases have better performances; with respect to the median performance, all aggregations show similar results in the range 2–5%. In addition to the plot, we found that the average optimality gap for '4H' is around 5%, for '2H', '1H' and '1HC' is around
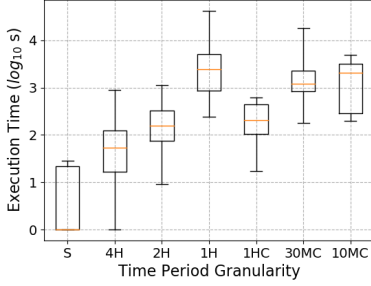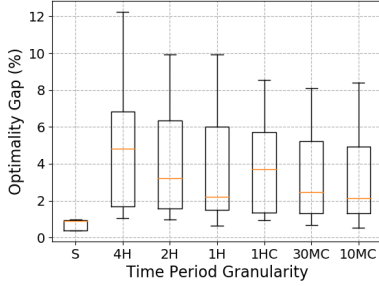
Fig. 5: Execution Time

Fig. 6: Optimality Gap

4%, while '30MC' and '10MC' is around 3.3%.

C. Cost components assessment

We tested the assignments generated by our algorithm against the original 10-minute periods demands in our datasets, considering all twelve weeks separately. That is, for every training instance we have twelve tests with a different demand.

First, we compare the performance of the time-periods aggregation from the point of view of the MEC access latency costs, that is defined by the objective function of model (1).

We present the costs in three parts:

- the assignment cost considering the distance between a BS and MEC facility and the demand of the BS (component $\beta \sum_{t \in T} \sum_{i \in A} \sum_{k \in K} d_i^t m_{ik} x_{ik}^t$ in (1));
- the switching cost considering the distance between MEC facilities in considering time-slots and the demand of the BS (component $\alpha \sum_{t \in T} \sum_{i \in A} \sum_{\substack{(j,k) \in \\ K \times K}} d_i^t l_{jk} y_{ijk}^t$ in (1));
- and the total cost (1).

We do not present the absolute value of the assignment and total costs, rather for every test week we compute the percentage difference between the lowest cost among those obtained with the seven time-period aggregations (4H, 2H, ... 10MC, plus the benchmark S) and the cost obtained with the given time-period aggregation. For example, while testing with 10 facilities, trained with average reference week, let us assume that for test week 1 the minimum cost $\bar{c}_1$ is given by the training assignment generated by '30MC' aggregation: the percentage difference of the costs of the test week 1 of every time-period aggregation is computed as $(c_1 - \bar{c}_1)/\bar{c}_1$. Hence, '30MC' will have a percentage gap of 0 for test week 1, in the scenario with 10 facilities trained with average week.

In Figure 7 we present the box-plots of the percentage gaps of the costs. Every figure contains a separate box-plot for each time-period aggregation method. We can notice that:

- the positive effect of clustering can be evaluated by comparing '1H' with '30MC': they yield a similar number of time-periods, but the latter allows slightly faster training, producing at the same time solutions of lower costs;
- w.r.t. assignment costs (Fig. 7a), 'S' always leads to the highest cost, i.e. longer MEC access latency, on median 20% higher than the minimum; all other aggregations show similar results, except '4H' which has slightly worse results (on median 8% higher cost than the minimum): the lowest average cost is given by '30MC', '1H' retrieves a cost higher on median of 1%, while '2H','1HC' and '10MC' show worsening of about 2-4%;
- on the contrary, w.r.t. switching costs (Fig. 7b), with the exception of 'S' which always ensures no switching costs, '4H' always leads to the lowest cost, i.e., the least number of MEC facility switching, and on median all other aggregations retrieve a switching costs from 2 to 8 times the cost given by '4H';
- however, the huge difference in the switching cost does not lead to a significant change in the total cost (Fig. 7c): this latter is composed mostly of the aggregation costs and it shows similar difference gaps.

In order to further analyze this behavior, in Figures 8 and 9 we present two indices regarding the assignments and the switching arising from the training:

- given that every BS has to be assigned to a MEC facility in every time-period, in our model the best option corresponds to the nearest facility. Therefore, we compute for every case the percentage of times a BS has not been assigned to its nearest MEC facility, that we present in Figure 8 as single box-plots for every time-period aggregation. We can notice that 'S' has the highest median non-nearest assignments (more than 30% of the assignments); '4H' and '1HC' have similar median behavior with 26% of non-nearest assignments, and all other aggregations show a value around 22%. This behavior better explains the poor performance of 'S' and '4H' for what concerns assignment costs.
- in Figure 9 we present the percentage of times a switching occurs in any time-period for any BS (i.e. number of switching over $(|T| - 1) \cdot |A|$), as box-plot for each time-period aggregation: we can notice that this value is considerably low for '4H' (on average less than 0.5% of the time a switching occurs) and is on average low for every aggregation (the highest value is 1.25% of the times). This behavior also explains the low effect of the switching costs in the total cost computation.

D. Computing capacity violation

Given that the BS-MEC facility assignment is computed using a reference week, it can generate a violation of the MEC facility capacity given by the change of the demand pattern in

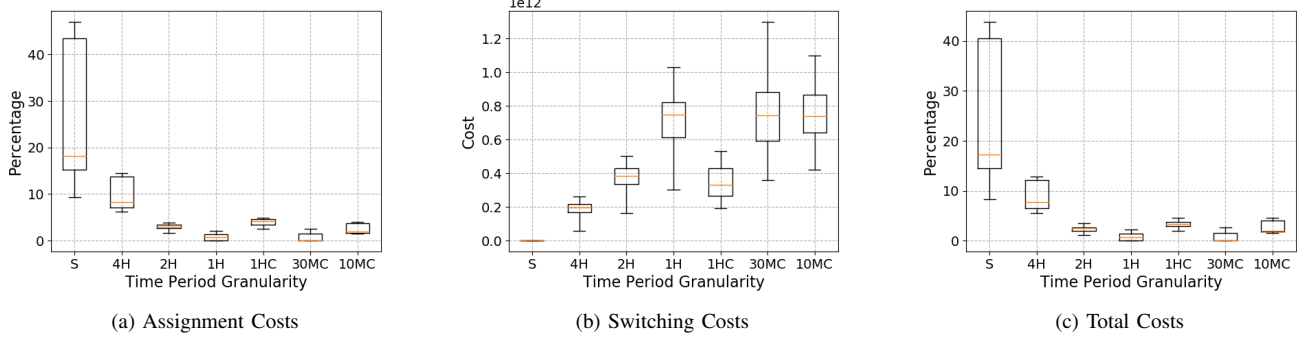| (a) Assignment Costs | (b) Switching Costs | (c) Total Costs |

Fig. 7: MEC access latency and switching costs gaps.

the test week from the reference week. In order to measure the violation of capacity, we introduce three indices:

- average capacity excess ('SUM-SUM' in the remainder):

$$\sum_{t \in T} \sum_{k \in K} \frac{\max\{\sum_{i \in A} d_i^t x_{ik}^t - C_k, 0\}}{C_k \cdot |K| \cdot |T|}$$

- percentage number of times a capacity is exceeded ('SUPPORT' in the remainder):

$$\frac{|\{(t,k) : \sum_{i \in A} d_i^t x_{ik}^t - C_k < 0, \forall t \in T, \forall k \in K\}|}{|K| \cdot |T|}$$

- average of excess, only when a violation occurs ('SUM-SUM-SUPPORT'):

$$\frac{\sum_{t \in T, k \in K : d_i^t x_{ik}^t - C_k < 0} \left(\frac{\sum_{i \in A} d_i^t x_{ik}^t - C_k}{C_k} \right)}{|\{(t,k) : \sum_{i \in A} d_i^t x_{ik}^t - C_k < 0, \forall t \in T, \forall k \in K\}|}$$

In Figure 10 we present box-plots of these three indices, in logarithmic scale (base 10). We can notice that:

- the 'SUM-SUM' index (Fig. 10a) is rather low for every time-period aggregation, only the '10MC' show a slightly higher median value, but it is less than the 0.05% for both the reference weeks;
- the 'SUPPORT' index (Fig. 10b), i.e. the percentage of time-periods in which a MEC facility has a capacity violation, does not show particular differences between the time-periods aggregation; '4H' shows a lower third quartile, that is however always lower than 0.5% for every time-period aggregation;
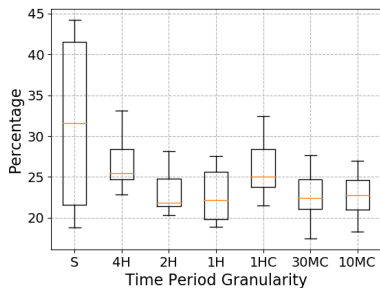
- the 'SUM-SUM-SUPPORT' index (Fig. 10c), i.e. the average violation computed only when violations occur, show a different behaviour for the '10MC' aggregation: while the median value is almost constant for all aggregations, '10MC' in the worst-case can violates a MEC capacity of more than 350% (i.e. it assign to a facility an amount of demand that is more than three times its capacity); this behaviour would advise against '10MC'.

VI. CONCLUSIONS

We presented in this paper a MEC orchestration framework that (i) enables taking orchestration decisions on base station to MEC facility assignments, and that (ii) at an arbitrary time period granularity within a reference horizon hence taking into consideration load variations along time, while (iii) supporting advanced spatio-temporal clustering among base stations based on network data analytics. It is, as of our knowledge, the first effort of this type.

We show that - by extensive simulations against real network data of an application that could benefit from MEC - with our framework we (a) largely outperform baseline orchestration decision without time-period aggregation by a order of magnitude in terms of MEC access latency, (b) scale with the number of time periods by leveraging on spatio-temporal clustering of base stations, and (c) identify which time-periods and aggregation techniques better allow minimizing MEC access latency and facility switching costs.

As a further work we plan at refining the clustering algorithms so as to anticipate factors in the preprocessing phase that could enhance the quality of the orchestration solutions.
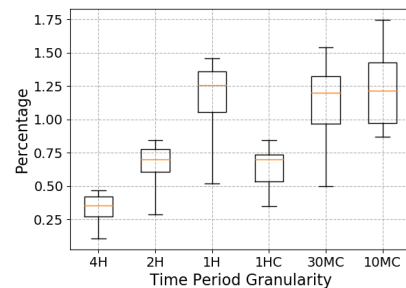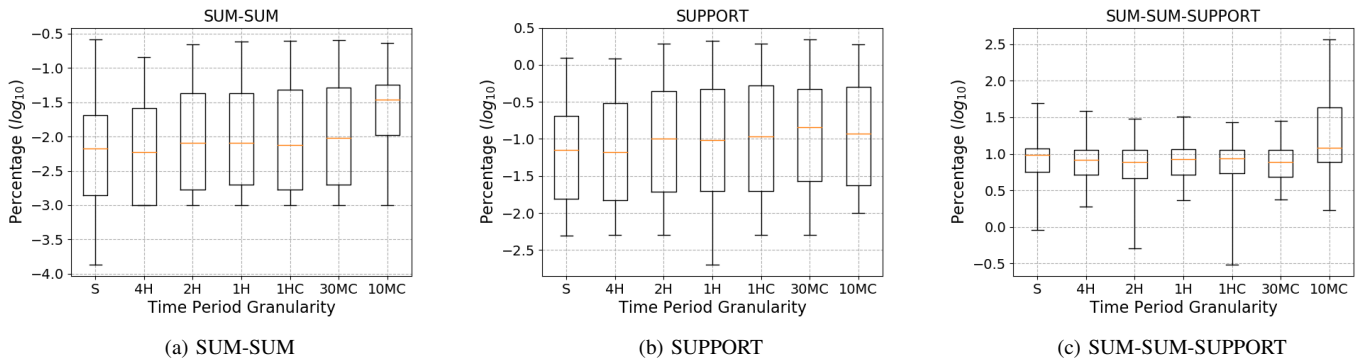

Fig. 8: Non-Nearest Assignments

Fig. 9: Switching Occurrences

| (a) SUM-SUM | (b) SUPPORT | (c) SUM-SUM-SUPPORT |

Fig. 10: Capacity violation measures.

REFERENCES

[1] M. Patel et al., "Mobile-Edge computing introductory technical white paper", ETSI MEC, Tech. Rep., 2014.

[2] H. Hawilo, A. Shami, M. Mirahmadi, R. Asal, "NFV: state of the art, challenges, and implementation in next generation mobile networks (vEPC)", *IEEE Network*, vol. 28, no. 6, pp. 18–26, 2014.

[3] A. Aijaz et al., "Realizing the tactile internet: Haptic communications over next generation 5g cellular networks", *IEEE Wireless Communications*, vol. 24, no. 2, pp. 82–89, April 2017.

[4] D. Dietrich et al., "Network function placement on virtualized cellular cores", in *Proc. of COMSNETS 2017*.

[5] M. Satyanarayanan, "Cloudlets: at the leading edge of cloud-mobile convergence", in *Proc. of ACM QoSA 2013*.

[6] D. Lindemeier, "MEC proofs of concept," ETSI, [Online]: http://www.etsi.org/technologies-clusters/technologies/multi-access-edge-computing/mec-poc

[7] P. Rost et al., "Mobile network architecture evolution toward 5G," *IEEE Communications Magazine*, vol. 54, no. 5, pp. 84–91, May 2016.

[8] "Mobile Edge Computing (MEC); Framework and Reference Architecture", ETSI DGS/MEC-003Arch, V1.1.1 (2016-03).

[9] "Network Functions Virtualisation (NFV); Architectural Framework", ETSI RGS/NFV-002, V1.2.1 (2014-12).

[10] A. Ceselli, M. Premoli, S. Secci, "Mobile edge cloud network design optimization", *IEEE Transactions on Networking*, vol. 25, no. 3, 2017.

[11] C. Colman-Meixner et al., "Resilient cloud network mapping with virtualized BBU placement for cloud-RAN", in *Proc. of ANTS 2016*.

[12] S. Secci, P. Raad, P. Gallard, "Linking virtual machine mobility to user mobility", *IEEE Transactions on Network and Service Management*, vol. 13, no. 4, pp. 927–940, Dec 2016.

[13] T. Taleb, A. Ksentini, "Follow me cloud: interworking federated clouds and distributed mobile networks", *IEEE Network*, vol. 27, no. 5, pp. 12–19, 2013.

[14] MY. Lyazidi, N. Aitsaadi, R. Langar, "Dynamic resource allocation for Cloud-RAN in LTE with real-time BBU/RRH assignment", in *Proc. of IEEE ICC 2016*.

[15] H. Assem, T. S. Buda, L. Xu, "Initial use cases, scenarios and requirements", *H2020 5G-PPP CogNet, Deliverable D2.1*, 2015.

[16] K. Zheng et al., "Big data-driven optimization for mobile networks toward 5G", *IEEE Network*, vol. 30, no. 1, pp. 44–51, January 2016.

[17] H. Yao et al., "A novel framework of data-driven networking", *IEEE Access*, vol. 4, pp. 9066–9072, 2016.

[18] J. Jiang, V. Sekar, I. Stoica, H. Zhang, *Unleashing the Potential of Data-Driven Networking*. Springer International Publishing, 2017.

[19] "EC H2020 5G infrastructure PPP pre-structuring model, version 2.0," 5GPPP, http://5g-ppp.eu, April 2014.

[20] A. Furno et al., "A tale of ten cities: Characterizing signatures of mobile traffic in urban areas", *IEEE Transactions on Mobile Computing*, vol. 16, no. 10, October 2017.

[21] A. Furno, M. Fiore, R. Stanica, "Joint spatial and temporal classification of mobile traffic demands," in *Proc. of IEEE INFOCOM 2017*.

[22] J. Wang et al, "Spatiotemporal Modeling and Prediction in Cellular Networks: A Big Data Enabled Deep Learning Approach", in *Proc. of INFOCOM 2017*.

[23] A. Ganjam et al., "C3: Internet-scale control plane for video quality optimization", in *Proc. USENIX NSDI 2015*.

[24] J. Jiang et al., "CFA: A practical prediction system for video QoE optimization", in *Proc. of USENIX NSDI 2016*.

[25] K. Winstein, H. Balakrishnan, "TCP ex machina: Computer-generated congestion control", in *Proc. of ACM SIGCOMM 2013*.

[26] M. Dong et al., "PCC: Re-architecting congestion control for consistent high performance", in *Proc. of USENIX NSDI 2015*.

[27] L. Chen et al., "Complementary Base Station Clustering for Cost-Effective and Energy-Efficient Cloud-RAN", in *Proc. of UIC 2017*.

[28] S. Martello, P. Toth, *Knapsack Problems: Algorithms and Computer Implementations*. New York, NY, USA: John Wiley & Sons, Inc., 1990.

[29] D. R. Morales, H. E. Romeijn, "The generalized assignment problem and extensions", in *Handbook of Combinatorial Optimization*, D.-Z. Du and P. M. Pardalos, Eds., 2005, pp. 259–311.

[30] R. Freling, H. E. Romeijn, D. R. Morales, A. P. M. Wagelmans, "A branch-and-price algorithm for the multiperiod single-sourcing problem", *Operations Research*, vol. 51, no. 6, pp. 922 – 939, 2003.

[31] I. Murthy, P. K. Seo, "A dualascent procedure for the file allocation and join site selection problem on a telecommunications network", *Networks*, vol. 33, no. 2, pp. 109 – 124, 3 1999.

[32] I. Murthy, "Solving the multiperiod assignment problem with start-up costs using dual ascent", *Naval Res. Log.*, vol. 40, pp. 325–344, 1993.

[33] S. Nickel, F. S. da Gama, "Multi-period facility location," in *Location Science*, G. Laporte, S. Nickel, and F. S. da Gama, Eds. Springer, 2015, pp. 289–310.

[34] I. Gourdin, O. Klopfenstein, "Multi-period capacitated location with modular equipments", *Comp. Op. Res.*, vol. 35, no. 3, pp. 661–682, 2008.

[35] J. Castro, S. Nasini, F. Saldanha-da Gama, "A cutting-plane approach for large-scale capacitated multi-period facility location using a specialized interior-point method", *Mathematical Programming*, vol. 163, no. 1, pp. 411–444, 2017.

[36] R. Halper, S. Raghavan, M. Sahin, "Local search heuristics for the mobile facility location problem", *Computers & Operations Research*, vol. 62, pp. 210 – 223, 2015. [Online].

[37] A. Furno, D. Naboulsi, R. Stanica, M. Fiore, "Mobile demand profiling for cellular cognitive networking", *IEEE Transactions on Mobile Computing*, vol. 16, no. 3, March 2017.

[38] A. Ceselli, M. Fiore, M. Premoli, S. Secci, "Optimized Assignment Patterns in Mobile Edge Cloud Networks", *Comp. Op. Res*, 2018.

[39] *IBM ILOG CPLEX 12.6 User Manual*. IBM corp., 2013, accessed: 2016-11-01.

[40] Shumaker, B. P., and R. W. Sinnott. "Astronomical computing: 1. Computing under the open sky. 2. Virtues of the haversine.", Sky and telescope 68 (1984): 158-159.