

Personalized Travel Itineraries with Multi-access Edge Computing Touristic Services

Felipe Fonseca, Lefteris Mamatas, Aline Carneiro Viana, Sand L Correa, Kleber V Cardoso

▶ **To cite this version:**

Felipe Fonseca, Lefteris Mamatas, Aline Carneiro Viana, Sand L Correa, Kleber V Cardoso. Personalized Travel Itineraries with Multi-access Edge Computing Touristic Services. IEEE Globecom 2019 - IEEE Global Communications Conference, Dec 2019, Big Island, Hawaii, United States. hal-02285929

HAL Id: hal-02285929
https://inria.hal.science/hal-02285929

Submitted on 13 Sep 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Personalized Travel Itineraries with Multi-access Edge Computing Touristic Services

Felipe F. Fonseca*, Lefteris Mamatas†, Aline C. Viana‡, Sand L. Correa*, Kleber V. Cardoso*

*Instituto de Informática
Universidade Federal de Goiás
Goiânia, Brazil
fonsecafel@gmail.com,{sand,kleber}@inf.ufg.br

†Department of Applied Informatics
University of Macedonia
Thessaloniki, Greece
emamatas@uom.edu.gr

‡INRIA
Universitè Paris Saclay
Paris, France
aline.viana@inria.fr

***Abstract*—The 5G networks enable new touristic services with challenging communication requirements, such as augmented reality (AR) applications, and allow the visitors to enjoy a touristic experience that involves both the physical and virtual space. Here, we propose a novel multi-user travel itinerary planning framework based on an optimal problem formulation that considers both individual trip itinerary (e.g., tourist's preferences, time or cost) and touristic service constraints (e.g., nearby edge cloud resources and application requirements). The main idea is to maximize the itinerary score of individual visitors, while also optimizing the resource allocation at the edge. We consider two services, video streaming and AR, and evaluate our framework using data from Flickr. Results demonstrate gains up to 100% in the resource allocation and user experience in comparison with a state-of-the-art solution adapted to this scenario.**

***Index Terms*— Personalized Travel Itineraries, ESPPRC, Multi-access Edge Computing, Network Slicing.**

I. Introduction

5G networks and relevant communication paradigms, including the Multi-Access Edge Computing (MEC) [1] and network slicing [2], bring an unprecedented level of flexibility in the network and cloud resource allocation, while improving user experience. For example, the MEC provides edge clouds to dynamically allocate resources near the users, i.e., to reduce the involved communication delays and extend the processing capabilities of the mobile devices. Furthermore, network and cloud slicing [3] allow the partitioning and the allocation of resources to satisfy diverse service demands. These features have the capability to: (i) provide personalised services; (ii) isolate utilized resources between applications; and (iii) support new applications or services, while providing quality of experience improvements to the existing ones.

Tourism is one of the largest industries in the world [4] and the 5G networks are foreseen as an important enabler for next-generation touristic services [5]. With the improved communication and computation capabilities provided by 5G technologies, tourists will be able to benefit from: (i) virtual, augmented or mixed reality (VR/AR/MR) applications enriching their sight-seeing activities; (ii) integrated transport, accommodation, and entertainment services; (iii) and new social-networking based ways to communicate with other tourists. Consequently, visitors (users) will be targeting for an advanced, combined physical and virtual touristic experience.

One of the most time consuming activities of a tourist is to plan the trip itinerary. To make this task easier, recommendation algorithms such as [6], [7] have been proposed for automatically and efficiently generating personalized itineraries for a single tourist. Usually, such algorithms are proposed as a combinatorial problem called the *Orienteering problem*, where given a graph with nodes representing the Points of Interest (POIs) of the city and edges being the travel routes between nodes, the algorithms find a max-benefit tour for a visitor considering factors such as popularity of the POIs, the target visitor's preferences, distance between the POIs, and the visitor's time limit to complete the tour. In the last few years, several variants of this problem have been proposed. For example, the authors in [8] studied the problem focusing in the case wherein the trip duration takes more than a single day. In [9], the problem is extended to take into account the crowdedness of the POIs, while in [10] the authors investigate the problem considering the queuing times at the attractions. There are also some works [11] where the problem is redefined to deal with a group of tourists travelling together but presenting different preferences.

However, in order to fulfill the physical and virtual expectations of tourists, recommendation algorithms for next-generation touristic services should also consider which service applications the visitor will consume in the POIs and how the content of such applications will be delivered to him given the available 5G Information and Communications Technology (ICT) infrastructure [12]. To the best of our knowledge, no previous work have tackled the problem of recommending personalized itineraries for tourists in the 5G context.

Towards filling this gap, this paper introduces n5GTour (*next-generation 5G Touristic services*), a framework for the optimized design of recommending personalized multi-user itineraries. n5gTour expands the tourist preferences to consider the efficient allocation of edge cloud resources providing advanced applications' performance. We present the problem formulation and a solution that finds the highest achievable itinerary score of individual visitors, while optimizing the resource allocation at the edge. We evaluate the effectiveness of n5GTour using POI datasets of four different cities in Europe and the Flickr dataset. We compare our results with (a modified

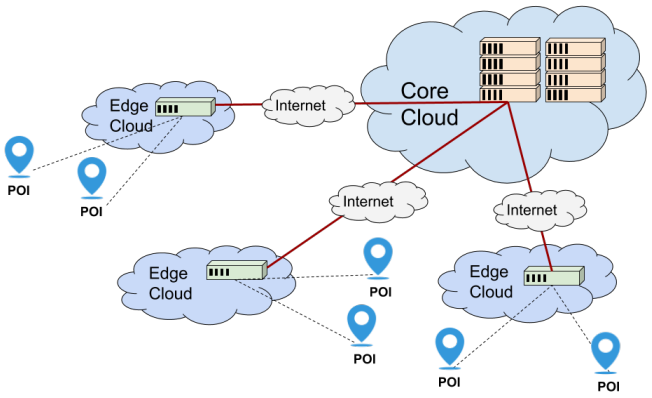


Fig. 1. Tourist spot with ICT infrastructure.

version of) an state-of-the-art solution for recommending personalized single-user trip itinerary, named PersTour [6]. In order to take into account the user experience, we introduce a metric that combines traditional recommendation scoring with application demand attendance. The results show that n5GTour achieves gains from near 20% up to 100% in both aspects, i.e., resource allocation and user experience, while presenting performance similar to PersTour in classic metrics for evaluation of personalized itinerary recommendation. In summary, we show that: (i) our solution provides general improved user experience, through proper resource allocation, without sacrificing individual tourist preferences, and (ii) since 5G technologies offer a broader knowledge on the available resources, this should be explored by service providers, e.g., to improve users experience or to offer new applications.

Paper organization: our system model is introduced in Section II; Section III provides the problems formulation and the solution methods; experimental results are shown in Section IV; conclusions are presented in Section V.

II. System Model

We consider a Metropolitan Tourist Centre (MTC) responsible for offering tourist information facilities to those visitors aiming at enriching their visiting experience. The MTC will leverage the (multi-access) edge clouds deployed near the POIs, potentially operated by a telecom service provider which is geographically present in the city, as well as a larger cloud infrastructure, as shown in Fig. 1. The cloud infrastructure will maintain all content required by the services offered to the visitors, while edge clouds will be preferably used by applications that can benefit from low-latency in the access to the computing resources. We next describe the four basic aspects of our system model.

1) POIs – For a city with $|\mathcal{V}|$ POIs, we consider a complete non-oriented graph $G = (\mathcal{V}, \mathcal{E})$, with $\mathcal{V} = \{v_1, \ldots, v_{|\mathcal{V}|}\}$ being the set of vertices representing the POIs and $\mathcal{E} = \{(v_i, v_j) \mid v_i, v_j \in \mathcal{V}\}$ the set of edges with cost $c_{i,j}$ representing the travel time (in seconds) between vertex v_i to vertex v_j. Each POI $v_i \in \mathcal{V}$ is characterized by the following attributes:

- $Pop(v_i) \in \mathbb{Z}$: the POI's popularity;
- $Avg(v_i) \in \mathbb{R}$: the expected time in seconds one should spend in the POI to enjoy what it has to offer;
- $\mathcal{C}_{v_i}$: one or more categories that describes the nature of the POI such as Historical, Cultural, Park, Museum, etc.;
- $Lat(v_i), Long(v_i)$: a location expressed in terms of latitude and longitude.

2) Services – In each POI, the MTC offers a set $\mathcal{A} = \{a_1, \ldots, a_{|\mathcal{A}|}\}$ of service applications to enrich the visitor's experience. Each service $a_i \in \mathcal{A}$ has particular requirements in terms of network and computing resources, expressed as:

- $\lambda_{a_i}^{min}$, and $\lambda_{a_i}^{max}$: minimum and maximum bandwidth demand in Mbps (respectively);
- $\rho_{a_i}^{min}$, and $\rho_{a_i}^{max}$: minimum and maximum processing load in reference cores (respectively);

A service $a_i \in \mathcal{A}$ is said to have strict requirements when either $\lambda_{a_i}^{min} = \lambda_{a_i}^{max}$, or $\rho_{a_i}^{min} = \rho_{a_i}^{max}$. Otherwise the service is considered to have elastic requirements. Whenever possible, services should run on the edge clouds, since this results in faster response time and increased visitor experience. When no resource is available at the edge, services with elastic requirements can still run at the cloud infrastructure with degraded performance.

3) ICT infrastructure – For each POI, we consider that two types of ICT infrastructure resources are available: connectivity and computing ones. In each POI, a connectivity resource is provided, for example, by a wireless access point. We define Λ_{v_i} as the maximum connectivity resource available at POI $v_i \in \mathcal{V}$. On the other hand, computing resources are provided by MEC hosts. We define $\mathcal{B} = \{b_1, \ldots, b_{|\mathcal{B}|}\}$ the set of all MEC hosts, and for each $b_i \in \mathcal{B}$ we define C_{b_i} as being the maximum computing resource available.

4) Visitors and itineraries – We consider a set $\mathcal{U} = \{u_1, \ldots, u_{|\mathcal{U}|}\}$ of visitors, where each visitor $u \in \mathcal{U}$ is characterized by the following attributes:

- $Attr^u(cat) \in \mathbb{R}$: the visitor's interest in category cat, where: $cat \in \bigcup_{v_i \in \mathcal{V}} \mathcal{C}_{v_i}$;
- $Int^u(v_i) \in \mathbb{R}$: the visitor's interest in POI v_i according to the POI's categories, i.e.: $Int^u(v_i) = \sum_{cat \in \mathcal{C}_{v_i}} Attr^u(cat)$;
- $v_s^u \in \mathcal{V}$: the place where the itinerary needs to begin;
- $v_f^u \in \mathcal{V}$: the place where the itinerary needs to finish;
- $Q^u \in \mathbb{R}$: the visitor's time budget for the entire itinerary;
- $\mathcal{A}_u \in 2^{\mathcal{A}}$: the set of services the visitor is willing to use;
- $\lambda_u^{min} = \sum_{a_i \in \mathcal{A}_u} \lambda_{a_i}^{min}$: minimum bandwidth demand in Mbps required during a POI visit;
- $\lambda_u^{max} = \sum_{a_i \in \mathcal{A}_u} \lambda_{a_i}^{max}$: maximum bandwidth demand in Mbps required during a POI visit;
- $\rho_u^{min} = \sum_{a_i \in \mathcal{A}_u} \rho_{a_i}^{min}$: minimum processing load in reference cores required during a POI visit;
- $\rho_u^{max} = \sum_{a_i \in \mathcal{A}_u} \rho_{a_i}^{max}$: maximum processing load in reference cores required during a POI visit.

We define an itinerary $I^u = (v_s^u, \ldots, v_f^u)$ of a visitor u as a sequence of POI visits, starting at $v_s^u \in \mathcal{V}$ and ending

at $v_f^u \in \mathcal{V}$. In addition, each itinerary I^u has a score (profit) given by: $\sum_{v_i \in I^u} \alpha Int^u(v_i) + (1-\alpha) Pop(v_i)$, where $0 \leq \alpha \leq 1$, and α can be customized to give more emphasis either on the visitor interest or on the POI's popularity.

Furthermore, let $I_{\mathcal{E}}^u$ be the set of edges traversed during an itinerary I^u and let $Avg(v_i)Int^u(v_i)$ be the recommended visit time for a given POI v_i. We say that I^u is valid, if the sum of the time spent visiting each POI $v_i \in I^u$ and the time travelling between POI visits is less than the budget Q^u, i.e, $\sum_{v_i \in I^u} Avg(v_i)Int^u(v_i) + \sum_{(v_i,v_j) \in I_{\mathcal{E}}^u} c_{i,j} \leq Q^u$.

Finally, we define $\mathcal{I}^{u*}$ as the set of valid itineraries of visitor u where the itinerary score is maximum.

n5GTour Design Problem – Given the set $\mathcal{V}$ of POIs with their amount of connectivity resource and the set $\mathcal{B}$ of MEC hosts that provides edge computing resources for such POIs; the set $\mathcal{A}$ of service applications offered by the MTC and their connectivity and processing demands; the set $\mathcal{U}$ of visitors, with their candidates itineraries $\bigcup_{u \in \mathcal{U}} \mathcal{I}^{u*}$ and the services they want to use, determine the set $\mathcal{I}^*$ of itineraries, so that:

- $\mathcal{I}^*$ contains exactly one itinerary for visitor u;
- the itineraries in $\mathcal{I}^*$ together do not exceed connectivity and processing capacity in the POIs at any given time;
- $\mathcal{I}^*$ takes into account the demands of the services chosen by the visitors;
- $\mathcal{I}^*$ maximizes the sum of scores for the chosen itineraries and prioritizes resource allocation at the edge.

III. Optimization Framework

We solve the n5GTour problem in two phases. In the first phase (A), given the set of POIs of the city, the set of paths connecting them, the set of visitors and their preferences, we find the set $\bigcup_{u \in \mathcal{U}} \mathcal{I}^{u*}$ of candidates itineraries for all visitors $u \in \mathcal{U}$. This set is computed by finding the set $\mathcal{I}^{u*}$ of candidate itineraries for each visitor $u \in \mathcal{U}$. Then, in the second phase (B), given the set $\bigcup_{u \in \mathcal{U}} \mathcal{I}^{u*}$, the service preferences for each visitor u as well as the service constraints, and the ICT infrastructure resource constraints, we determine the set $\mathcal{I}^*$.

A. Phase A: Personalized Single-User Itinerary Recommendation Problem

The problem of finding the set $\mathcal{I}^{u*}$ of itineraries with maximum score for a visitor u given a complete non-oriented graph $G = (\mathcal{V}, \mathcal{E})$ (with vertices representing the POIs and edges with costs $c_{i,j}$ representing the travel time between the vertices), constrained to a time budget Q^u and visitor's preferences ($Attr^u(cat)$, $Int^u(v_i)$, v_s^u,v_f^u) is known as the personalized single-user itinerary recommendation problem. We model this problem as a special case of the *Shortest Path Problem with Resource Constraints* (SPPRC) [13], which is known as *Elementary* SPPRC (ESPPRC).

Given a graph, the SPPRC finds the shortest path between a starting vertex and an ending vertex, fulfilling a set of constraints upon a set of given resources. Resource constraints represent the value that a resource ℓ has to be so that the vertex can be visited and are usually modeled as resource windows $[a_i^\ell, b_i^\ell]$ for each vertex of the graph and each resource ℓ, with $a_i^\ell, b_i^\ell \in \mathbb{R}$, and $a_i^\ell < b_i^\ell$. The ESPPRC has the additional constraint of finding a path that is elementary, e.g., a path in which no vertex is visited more than once. The SPPRC is similar to a *Multiple-criteria Problem* that is *NP-hard* [14].

In order to solve the personalized single-user itinerary recommendation problem as an ESPPRC, we use a dynamic programming algorithm with labels. The mapping is straightforward. Given the graph $G = (\mathcal{V}, \mathcal{E})$ of POIs, starting from a path $P = (v_s^u)$ (the desired starting point), the algorithm makes successive viable extensions, systematically building new paths. A viable path is the one that respect the time budget Q^u and ends at v_f^u (the desired ending point). In the personalized single-user itinerary recommendation problem, the sum of the time spent in the POIs (vertices) and travelling between them (edges) is the resource. Since we have a single resource (time), we use a resource window a_i and b_i and set them $a_i = 0$ and $b_i = Q^u - Avg(v_i)Int^u(v_i)$ for all $v_i \in \mathcal{V}$.

We define R_i as the label associated with path P_i. Associated with each label, we also have compressed information to speed up the dominance step of the algorithm. One of such information is the score of path P_i, denoted by $Profit(R_i)$ and given by $\sum_{v_i \in P_i} \alpha Int^u(v_i) + (1-\alpha) Pop(v_i)$. The score represents how interesting the itinerary is for the visitor. We use this information to discard paths, and keep only those with greater scores. At the end, the algorithm returns the set of all non-dominated paths, i.e, Pareto optimal paths. The optimal solution is part of this set and is the one with the highest profit.

B. Phase B: n5GTour Design Problem

Once we have computed the set $\mathcal{I}^{u*}$ for each visitor $u \in \mathcal{U}$, we can solve the n5GTour problem. Particularly, we formalize and solve this problem as a Mixed Integer Linear Programming (MILP) model, as described below.

Objective function – Let $\mathcal{I} = \bigcup_{u \in \mathcal{U}} \mathcal{I}^{u*}$ be the set of candidate itineraries for all the visitors. In addition to the sequence of POI visits and the time to spend at each POI, each itinerary $I_j^{u*} \in \mathcal{I}^{u*}$ also has a score given by $Score(I_j^{u*})$. This score is the $Profit(R_i)$ associated with $P_i = I_j^{u*}$.

We define the indicator function $\phi(I_j^{u*}, v_i) \in \{0, 1\}$ to represent POI visitation in an itinerary, with $\phi(I_j^{u*}, v_i) = 1$ if itinerary I_j^{u*} visits POI v_i and $\phi(I_j^{u*}, v_i) = 0$ otherwise. We consider that all itineraries in $\mathcal{I}$ start and end during a given period of H hours, and that ICT infrastructure resources (computing and connectivity) for these itineraries are allocated in a per minute basis. Since H hours contain $60H$ minutes, we define $1 \leq t \leq 60H$ time units when ICT infrastructure resources are allocated.

Given an itinerary I_j^{u*}, a POI v_i and a time instant t, we define the indicator function $\psi(I_j^{u*}, v_i, t) \in \{0, 1\}$, with $\psi(I_j^{u*}, v_i, t) = 1$ if itinerary I_j^{u*} is visiting POI v_i during t, and $\psi(I_j^{u*}, v_i, t) = 0$ otherwise. This information can be derived since we know the start time, sequence of visits, and visit duration for each itinerary.

We define the set of decision variables $y_{I_j^{u*}} \in \{0,1\}$ to represent itinerary choice, so that $y_{I_j^{u*}} = 1$, if the itinerary I_j^{u*} from the set $\mathcal{I}^{u*}$ of candidate itineraries of visitor u will compose the solution; and $y_{I_j^{u*}} = 0$ otherwise. For each POI visit, a MEC host has to provide computing resources for the visitor. We define the decision variable $z^{b_n}_{I_j^{u*},v_i} \in \{0,1\}$ to represent MEC host allocation, so that $z^{b_n}_{I_j^{u*},v_i} = 1$, if MEC host $b_n \in \mathcal{B}$ will be responsible to provide computing resources during a visit to POI $v_i \in \mathcal{V}$ in itinerary $I_j^{u*} \in \mathcal{I}^{u*}$; and $z^{b_n}_{I_j^{u*},v_i} = 0$ otherwise.

We also define decision variables $p_{I_j^{u*},v_i} \in \mathbb{R}$ and $q^{b_n}_{I_j^{u*},v_i} \in \mathbb{R}$ to represent, respectively, the amount of connectivity resources (in Mbps) allocated during a visit to POI $v_i \in \mathcal{V}$ in itinerary $I_j^{u*} \in \mathcal{I}^{u*}$, and the amount of computing resources (in reference cores) allocated at MEC host $b_n \in \mathcal{B}$ during a visit to POI $v_i \in \mathcal{V}$ in itinerary $I_j^{u*} \in \mathcal{I}^{u*}$. The objective of the model is to maximize the sum of scores of the chosen itineraries and maximize resource allocation at the edge. Considering $Norm(value)$ a generic function that normalizes a value, we define the following objective function:

$$\max_{y,z,p,q} \sum_{I_j^{u*}\in\mathcal{I}} y_{I_j^{u*}} Norm(Score(I_j^{u*})) + \sum_{I_j^{u*}\in\mathcal{I}} \sum_{v_i\in\mathcal{V}} \frac{Norm(p_{I_j^{u*},v_i})}{2|I_j^{u*}|} + \sum_{I_j^{u*}\in\mathcal{I}} \sum_{v_i\in\mathcal{V}} \sum_{b_n\in\mathcal{B}} \frac{Norm(q^{b_n}_{I_j^{u*},v_i})}{2|I_j^{u*}|}. \quad (1)$$

Our goal is to choose the itineraries with highest scores that will lead visitors to POIs that interests them the most. However, since we also want to maximize resource allocation at the edge to improve visitor's experience, for each chosen itinerary, we also add to the objective function the amount of connectivity and computing resources allocated at the edge for each POI visit. Normalization is used to give equal weights to itinerary score and resource allocation. Thus, our model may choose some itineraries that do not have the highest score for some users individually, but provide better resource allocation and visitor's experience to the set of visitors as a whole.

Itinerary choice constraints – For each visitor $u \in \mathcal{U}$, we need to ensure that one itinerary from the set $\mathcal{I}^{u*}$ of candidate itineraries is chosen. We ensure that with the constraints:

$$\sum_{I_j^{u*}\in\mathcal{I}^{u*}} y_{I_j^{u*}} = 1, \quad \forall u \in \mathcal{U}, \quad (2)$$

$$y_{I_j^{u*}} \in \{0,1\}, \quad \forall I_j^{u*} \in \mathcal{I}. \quad (3)$$

If an itinerary is chosen to compose a solution, we also need to allocate one MEC host to each POI visit to provide computing resources. We achieve this with the constraints:

$$\sum_{b_n\in\mathcal{B}} z^{b_n}_{I_j^{u*},v_i} = y_{I_j^{u*}} \phi(I_j^{u*}, v_i), \quad \forall I_j^{u*} \in \mathcal{I}, \quad \forall v_i \in \mathcal{V}, \quad (4)$$

$$z^{b_n}_{I_j^{u*},v_i} \in \{0,1\}, \quad \forall I_j^{u*} \in \mathcal{I}, \quad \forall v_i \in \mathcal{V}. \quad (5)$$

Service demand constraints – For each visitor and chosen itinerary, we also need to select at least the minimum required for the set of chosen applications to execute properly, taking care of to not allocate more than it is needed. We ensure proper allocation of connectivity resources with the constraints:

$$p_{I_j^{u*},v_i} \geq \lambda_u^{min} y_{I_j^{u*}} \phi(I_j^{u*}, v_i), \quad \forall I_j^{u*} \in \mathcal{I}, \quad \forall v_i \in \mathcal{V}, \quad (6)$$

$$p_{I_j^{u*},v_i} \leq \lambda_u^{max} y_{I_j^{u*}} \phi(I_j^{u*}, v_i), \quad \forall I_j^{u*} \in \mathcal{I}, \quad \forall v_i \in \mathcal{V}, \quad (7)$$

$$p_{I_j^{u*},v_i} \in \mathbb{R}, \quad \forall I_j^{u*} \in \mathcal{I}, \quad \forall v_i \in \mathcal{V}. \quad (8)$$

We ensure proper allocation of computing resources with the following constraints:

$$q^{b_n}_{I_j^{u*},v_i} \geq \rho_u^{min} z^{b_n}_{I_j^{u*},v_i} \phi(I_j^{u*}, v_i), \quad \forall I_j^{u*} \in \mathcal{I}, \quad \forall v_i \in \mathcal{V}, \quad \forall b_n \in \mathcal{B}, \quad (9)$$

$$q^{b_n}_{I_j^{u*},v_i} \leq \rho_u^{max} z^{b_n}_{I_j^{u*},v_i} \phi(I_j^{u*}, v_i), \quad \forall I_j^{u*} \in \mathcal{I}, \quad \forall v_i \in \mathcal{V}, \quad \forall b_n \in \mathcal{B}, \quad (10)$$

$$q^{b_n}_{I_j^{u*},v_i} \in \mathbb{R}, \quad \forall I_j^{u*} \in \mathcal{I}, \quad \forall v_i \in \mathcal{V}, \quad \forall b_n \in \mathcal{B}. \quad (11)$$

Resource capacity constraints – Finally, we have to make sure that the performed allocations do not exceed the amount of available connectivity and computing resources, at any given minute considered. We ensure this with the constraints:

$$\sum_{I_j^{u*}\in\mathcal{I}} \psi(I_j^{u*}, v_i, t) p_{I_j^{u*},v_i} \leq \Lambda_{v_i}, \quad \forall v_i \in \mathcal{V}, \quad 1 \leq t \leq 60H, \quad (12)$$

$$\sum_{I_j^{u*}\in\mathcal{I}} \sum_{v_i\in\mathcal{V}} \psi(I_j^{u*}, v_i, t) q^{b_n}_{I_j^{u*},v_i} \leq C_{b_n}, \quad \forall b_n \in \mathcal{B}, \quad 1 \leq t \leq 60H. \quad (13)$$

IV. Evaluation

In this section, we present the experimental evaluation of our optimization framework.

A. Datasets and Preprocessing

In order to obtain realistic results, we use real data extracted from Google Places API to derive four POI datasets comprising points of interest of four cities in Europe: London, Madrid, Barcelona, and Athens (one dataset per city). These cities are important from a touristic point of view and guarantee variety and diversity in our evaluation. For each city, we chose the first 30 POIs with the highest rating, i.e. $|\mathcal{V}| = 30$. In each dataset, each POI is identified by an id, name, latitude, and longitude ($Lat(v_i)$, $Long(v_i)$), and a set of categories it belongs to ($\mathcal{C}_{v_i}$). The cost $c_{i,j}$ of an edge connecting two POIs is computed using The Google Matrix Distance API in the walking mode.

To obtain POI popularity ($Pop(v_i)$), average POI visit duration ($Avg(v_i)$) and visitor interest in a given category ($Attr^u(cat)$), we preprocess a dataset of geo-tagged photos using the Flickr API, using an approach similar to [6]. We consider photos taken from 01-01-2017 to 12-31-2018 for the four considered cities. Then, for each city, we combine its photo dataset with the POI dataset to generate sequences

S of visits. To this end, we apply the following approach. First, we match each photo to the corresponding POI using the haversine formula and the POI's and photo's latitude and longitude. Second, we sort the photos by user and time and, for each user, we group photos within eight hours into a tour, i.e. $H = 8$. Then, for each tour, we take the times of the first and last photo associated with a given POI as the arrival and departure time in that POI, respectively. Finally, we discard sequences with repeated POI visits or with less than three POI visits. We end up the preprocessing phase with four sets of sequences of visits, one for each city.

Given the set $\mathcal{S}$ of sequences of visits of a given city, where each sequence has the form $S = \{(v_i \in V, arr(v_i), dep(v_i)), \ldots, (v_k \in V, arr(v_k), dep(v_k)\}$, with $arr(v_i)$ and $dep(v_i)$ being arrival and departure time at POI v_i, respectively, and considering $\mathcal{S}_u$ the set of all sequences for a given user $u \in \mathcal{U}$, we compute $Pop(v_i)$ and $Avg(v_i)$ as:

$$Pop(v_i) = \sum_{S \in \mathcal{S}} \sum_{v_x \in S} \delta(v_x = v_i)$$

$$Avg(v_i) = \frac{\sum\limits_{S \in \mathcal{S}} \sum\limits_{v_x \in S} (dep(v_x) - arr(v_x))\delta(v_x = v_i)}{Pop(v_i)},$$

where $\delta(v_x = v_i) = \{\frac{1 \quad \text{if } v_x = v_i,}{0 \quad \text{otherwise.}}$.

Assuming $\gamma(cat \in \mathcal{C}_{v_x}) = \{\frac{1 \quad \text{if } cat \in \mathcal{C}_{v_x},}{0 \quad \text{otherwise.}}$, we compute visitor's interest in a category cat as:

$$Attr^u(cat) = \sum_{S \in \mathcal{S}_u} \sum_{v_x \in S} \frac{(dep(v_x) - arr(v_x))}{Avg(v_x)} \gamma(cat \in \mathcal{C}_{v_x}).$$

B. Service Application and ICT Resources

We consider that visitors can make use of two applications ($|\mathcal{A}| = 2$): augmented reality and video streaming. The requirements of an augmented reality MEC Service were taken from [15] while the requirements of a video streaming application were based in the Netflix stream service website. Table I shows the bandwidth and processing requirements of both applications. Based on [16], we assume the amount of connectivity resources available at each POI as $\Lambda_{v_i} = 75$ Mbps. As for computing resources, we consider two MEC hosts ($|\mathcal{B}| = 2$), each one with 37.5 Reference Cores (RCs), with one RC being equivalent to the processing power of an Intel Haswell i7-4770 3.40GHz CPU [15].

TABLE I
APPLICATIONS REQUIREMENTS

Application	$\boldsymbol{\lambda^{min}}$	$\boldsymbol{\lambda^{max}}$	$\boldsymbol{\rho^{min}}$	$\boldsymbol{\rho^{max}}$
Augmented reality	1 Mbps	10 Mbps	0.1 RC	1 RC
Video streaming	1.5 Mbps	25 Mbps	None (0)	None (0)

C. Baseline Algorithm

We test our algorithm against PersTour [6], an state-of-the-art solution for recommending personalized single-user itinerary. However, PersTour does not take into account the ICT resources and the application usage in its itinerary generation. Thus, to perform the evaluation, we use the itinerary generated by PersTour as input to phase B of our solution. However, since PersTour produces only a single itinerary, this workaround may lead to infeasible solutions. When this happens, we solve the problem relaxing the constraints dealing with minimum allocation for services. In addition, because PersTour is desigend for single-user recommendation, to produce itineraries for all the users, we run PersTour $|\mathcal{U}|$ times.

D. Evaluation Metrics

Similar to other works related to recommendation algorithms [6]–[11], we use metrics as **Recall**, **Precision** and, **F-score** to evaluate our framework. In addition, we introduce two new metrics, **Allocation** and **Ux** to assess, respectively, the effectiveness of n5GTour on allocating resources in the edge and the overall user experience provided by the recommended itinerary.

- **Recall** – the fraction of POIs that exists in the user's real life travel sequence (generated in the preprocessing phase) that also exists in the generated itinerary I_j^{u*}. If we consider S^u as the user's real life travel sequence, we define $Recall(I_j^{u*}) = \frac{|I_j^{u*} \cap S^u|}{|S^u|}$.
- **Precision** – the fraction of POIs that exists in the generated itinerary that also exists in the user's real life travel sequence. If we consider S^u as the user's real life travel sequence, we define $Precision(I_j^{u*}) = \frac{|I_j^{u*} \cap S^u|}{|I_j^{u*}|}$.
- **F-score** – the harmonic average of **Precision** and **Recall**, representing the model accuracy.
- **Allocation** – the fraction of resources allocated according to a visitor's demand. This is only calculated for visitor's itineraries that make use of at least one application and it is defined as:

$$Allocation(I_j^{u*}) = \sum_{v_i \in \mathcal{V}} \frac{p_{I_j^{u*}, v_i}}{2|I_j^{u*}|\lambda_u^{max}} + \sum_{v_i \in \mathcal{V}} \sum_{b_n \in \mathcal{B}} \frac{q^{b_n}_{I_j^{u*}, v_i}}{2|I_j^{u*}|\rho_u^{max}}.$$

- **Ux** – the relation between allocated resources for an itinerary and an visitor interest in this itinerary. The idea is that high **Allocation** alone does not make an itinerary good/better, and **Ux** also embeds visitor interest within it. Defined as $Ux(I_j^{u*}) = Allocation(I_j^{u*})Norm(Profit(I_j^{u*}))$.

E. Results

For each city, we draft 250 visitor profiles ($|\mathcal{U}| = 250$) from the set of all profiles generated after our prepossessing phase described in Section IV-A. For each visitor, we also draft one application profile from $2^{|\mathcal{A}|}$, the power set for the set of applications $\mathcal{A}$. Note that, because of this, some visitors may wish to use no application/service, which is also the truth in real life. For n5GTour, we run phase A (considering $\alpha = 0.5$) to generate the set of candidate itineraries for all visitors, and then supply these sets, along with applications profiles and

resources availability to phase B, which then chooses one itinerary for each visitor and allocate resources for it. For PersTour, we run it one time for each visitor providing only the visitor profiles.

Since we evaluate generated itineraries against real life travel sequences, we employ leave-one-out evaluation. That is, each visitor profile is built using all travel sequences of a visitor, except the one we want to test. The starting POI, ending POI, and budget of this left out real travel sequence are also used as the respective starting POI, ending POI, and budget of the visitor profile.

All experiments were performed at a virtual machine (VM) that runs Debian 9 GNU/Linux and is configured with 16 vCPUs, 64GB RAM, and 40GB of virtual disk. The VM is hosted in a server with two Intel Xeon Silver 4114 processors running at 2.20GHz. Phase A of n5GTour was implemented in C++. Both PersTour and phase B of n5GTour were implemented using Python 2.7.12, docplex 2.8.125, and IBM CPLEX 12.8.0 as the solver.

We assess the metrics defined in Section IV-D for itineraries generated by both n5GTour and PersTour. Due to the lack of space, we summarize our results to the average values shown in Table II. Despite having the ability of sometimes choosing itineraries with lower scores in favor of allocating more resources to provide better application performance, n5GTour outperforms PersTour in all metrics other than three **Recall** results. The greatest difference appears when comparing **Allocation** and **Ux** metrics, meaning that n5GTour has the ability to choose itineraries that will provide better application usage experience and a better user experience when considering user interest as well. Our best result is with Athens dataset, with 115.68% and 106.87% performance gains in **Allocation** and **Ux**, respectively. Coincidentally, Athens is also the city with the largest set of candidate itineraries generated in phase A, with 9922 in total, showing that n5GTour tends to perform better with larger input datasets.

TABLE II
AVERAGE VALUE OF THE METRICS, ROUNDED TO 3 DECIMAL PLACES

	Recall	Precision	F-score	Allocation	Ux
n5GTour - London	0.816	0.45	0.543	0.817	0.42
PersTour - London	0.845	0.433	0.541	0.663	0.351
Gain - London	-3.59%	**+4.03%**	**+0.42%**	**+23.28%**	**+19.35%**
n5GTour - Madrid	0.77	0.533	0.592	0.57	0.248
PersTour - Madrid	0.788	0.496	0.574	0.459	0.206
Gain - Madrid	-2.33%	**+7.33%**	**+3.12%**	**+24.27%**	**+20.28%**
n5GTour - Barcelona	0.772	0.478	0.555	0.665	0.397
PersTour - Barcelona	0.771	0.464	0.546	0.501	0.307
Gain - Barcelona	**+0.04%**	**+2.91%**	**+1.71%**	**+32.86%**	**+29.38%**
n5GTour - Athens	0.859	0.325	0.456	0.564	0.485
PersTour - Athens	0.863	0.301	0.432	0.261	0.234
Gain - Athens	-0.47%	**+7.98%**	**+5.61%**	**+115.68%**	**+106.87%**

V. CONCLUSION

Is this paper, we presented n5GTour, an optimization framework for recommending personalized itineraries for a set of visitors. Our model considers POI popularity, user interest, application demand, and network/computing resources, when deciding which itineraries to recommend. We presented two algorithms. First, a dynamic programming algorithm to solve the ESPPRC problem and to generate sets of Pareto optimal itineraries for users, while considering POI popularity and user interest. Second, a MILP model that, given sets of candidate itineraries for a set of users, application demand, and available computing/network resources, chooses one itinerary for each user and allocate the corresponding resources. As future work, we intend to include a degree of uncertainty in the model, since not all visitors may follow the recommendation and, for example, may extrapolate the visitation time.

REFERENCES

[1] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, and D. Sabella, "On multi-access edge computing: A survey of the emerging 5g network edge cloud architecture and orchestration," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1657–1681, 2017.

[2] X. Foukas, G. Patounas, A. Elmokashfi, and M. K. Marina, "Network slicing in 5g: Survey and challenges," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 94–100, 2017.

[3] F. S. D. Silva *et al.*, "NECOS Project: Towards Lightweight Slicing of Cloud Federated Infrastructures," in *2018 4th IEEE Conf. on Network Softwarization and Workshops (NetSoft)*, June 2018, pp. 406–414.

[4] Statista Team, "Direct and total contribution of travel and tourism to the global economy from 2006 to 2017," https://www.statista.com/statistics/233223/travel-and-tourism–total-economic-contribution-worldwide/, 2017, [Last access: Oct-24-2018].

[5] UK Department for Digital, Culture, Media & Sport, "5G Smart Tourism," https://www.gov.uk/government/case-studies/5g-smart-tourism, 2018, [Last access: Oct-27-2018].

[6] K. H. Lim, J. Chan, C. Leckie, and S. Karunasekera, "Personalized trip recommendation for tourists based on user interests, points of interest visit durations and visit recency," *Knowledge and Information Systems*, vol. 54, no. 2, pp. 375–406, 2018.

[7] I. Brilhante, J. A. Macedo, F. M. Nardini, R. Perego, and C. Renso, "Where shall we go today?: Planning touristic tours with tripbuilder," in *Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management*, 2013, pp. 757–762.

[8] Z. Friggstad, S. Gollapudi, K. Kollias, T. Sarlos, C. Swamy, and A. Tomkins, "Orienteering algorithms for generating travel itineraries," in *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. ACM, 2018, pp. 180–188.

[9] X. Wang, C. Leckie, J. Chan, K. H. Lim, and T. Vaithianathan, "Improving personalized trip recommendation by avoiding crowds," in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, 2016, pp. 25–34.

[10] K. H. Lim, J. Chan, S. Karunasekera, and C. Leckie, "Personalized itinerary recommendation with queuing time awareness," in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2017, pp. 325–334.

[11] L. Fan, L. Bonomi, C. Shahabi, and L. Xiong, "Multi-user itinerary planning for optimal group preference," in *Advances in Spatial and Temporal Databases*, 2017, pp. 3–23.

[12] Y. Fu, S. Wang, C. Wang, X. Hong, and S. McLaughlin, "Artificial Intelligence to Manage Network Traffic of 5G Wireless Networks," *IEEE Network*, vol. 32, no. 6, pp. 58–64, Nov 2018.

[13] S. Irnich and G. Desaulniers, *Shortest Path Problems with Resource Constraints*. Springer US, 2005, pp. 33–65.

[14] G. Desaulniers, J. Desrosiers, and M. M. Solomon, Eds., *Column generation*, 1st ed. Springer Science & Business Media, 2006.

[15] A. Garcia-Saavedra, G. Iosifidis, X. Costa-Perez, and D. J. Leith, "Joint optimization of edge computing architectures and radio access networks," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 11, pp. 2433–2443, 2018.

[16] F. Malandrino, C. F. Chiasserini, G. Avino, M. Malinverno, and S. Kirkpatrick, "From megabits to cpu ticks: Enriching a demand trace in the age of mec," *IEEE Transactions on Big Data*, 2018.