ECAI 2016
G.A. Kaminka et al. (Eds.)

doi:10.3233/978-1-61499-672-9-1818

Planning Tourist Agendas for Different Travel Styles

Jesus Ibañez and **Laura Sebastia** and **Eva Onaindia**[1]

Abstract. This paper describes *e-Tourism2.0*, a web-based recommendation and planning system for tourism activities that takes into account the preferences that define the travel style of the user. *e-Tourism2.0* features a recommender system with access to various web services in order to obtain updated information about locations, monuments, opening hours, or transportation modes. The planning system of *e-Tourism2.0* models the taste and travel style preferences of the user and creates a planning problem which is later solved by a planner, returning a personalized plan (agenda) for the tourist. *e-Tourism2.0* contributes with a special module that calculates the recommendable duration of a visit for a user and the modeling of preferences into a planning problem.

1 INTRODUCTION

A **Recommender System** (RS) [13] is a personalization tool aimed to provide the items that best fit the individual tastes of people. A RS infers the user preferences by analyzing the available user data, information of other users and of the environment. The target of the extensively popularized Tourism RSs (TRSs) is to match the user preferences with the leisure resources and tourist activities of a city [15] by using some initial data, usually explicitly provided by the user. The relevance of TRSs relies in their capacity of automatically inferring the user preferences, through an explicit or implicit feedback of the user, as well as providing the user with a personal tourist activity agenda. Typically, TRSs use a hybrid approach of recommendation techniques such as demographic, content-based or collaborative filtering [2] and they are confined to recommendations within a delimited area or city since tourism infrastructure is usually developed to promote the tourism demand in particular spots [6, 11].

The latest developments in TRSs share a common mainstream, that of providing the most user-tailored tourist proposal. Hence, some tools like *SAMAP* [3] elicits a tourist plan with recommendations about the transportation mode, restaurants and bars or leisure attractions such as cinemas or theaters, all this accompanied with a detailed plan explanation. Scheduled routes presented in a map along with a timetable are nowadays a common functionality of many TRSs, like *e-Tourism* [6], which also include context information such as the opening and closing hours of the Points Of Interest (POIs) to visit and the geographical distances between POIs to compute the time to move from one place to another. Some other tools allow the user to interact with the plan or develop interfaces specifically designed to be used in mobile devices ([12, 16]). Personalization is interpreted in *CT-Planner* [9] as emphasizing the concept of interactive assistance between the user and a tour advisor, where the advisor offers several plans, learns the tourist preferences, requests feedback from the users and customizes the plans accordingly. *CT-Planner* also accounts for user preferences like the walking speed or reluctance to walk, in which case the planner will suggest short walking distances in the plan.

Recent advances in TRSs go one step ahead towards personalization and propose to adapt the duration of the visits to the user preferences. For instance, *PersTour* [10] calculates a personalized duration of a visit to POI using the POI popularity and the user interest preferences, which are automatically derived from real-life travel sequences based on geotagged photos. And the work in [14] considers user preferences based on the the number of days of the trip and the pace of the tour, that is, whether the user wants to perform many activities in one day or travel at a more relaxed pace.

In this paper, we present *e-Tourism2.0*, a TRS that draws upon the recommendation model and planning module of *e-Tourism* [6] and significantly enhances the personalization of the recommendations. *e-Tourism2.0* improves *e-Tourism* in two main aspects:

- *context-aware tool*: it establishes a connection to several web services to capture up-to-date context information such as opening hours of POIs to visit, location of POIs, ratings of users, modes of transport in the city, etc.;
- *preference temporal planning*: it handles a full range of user preferences such as the user interest in visiting a POI, the pace of the tour (relaxed vs busy) and variable durations of the visits within a temporal interval; all these preferences represent the user travel style. *e-Tourism2.0* uses OPTIC [1], a state-of-the-art planner that addresses the full set of preferences defined in PDDL3.0 language [7].

This paper is organized as follows. Section 2 summarizes the main aspects of *e-Tourism*. Section 3 explains the procedure to calculate the recommended duration of an activity for a given user and section 4 details the construction of the planning problem and the encoding of the user preferences within the planning problem. Section 5 shows several cases of study to test whether the defined preferences are taken into account correctly by the planner and last section concludes.

2 *e-Tourism2.0* TOOL

e-Tourism [6] was developed as a web application to generate recommendations about personalized tourist tours in the city

[1] Universitat Politècnica de València, Valencia, Spain, Email: {jeibrui, lstarin, onaindia}@dsic.upv.es

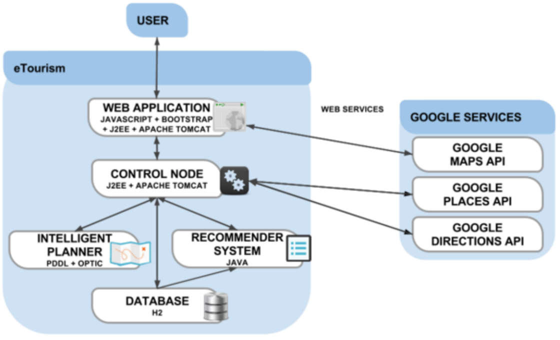

Figure 1. *e-Tourism2.0* Architecture

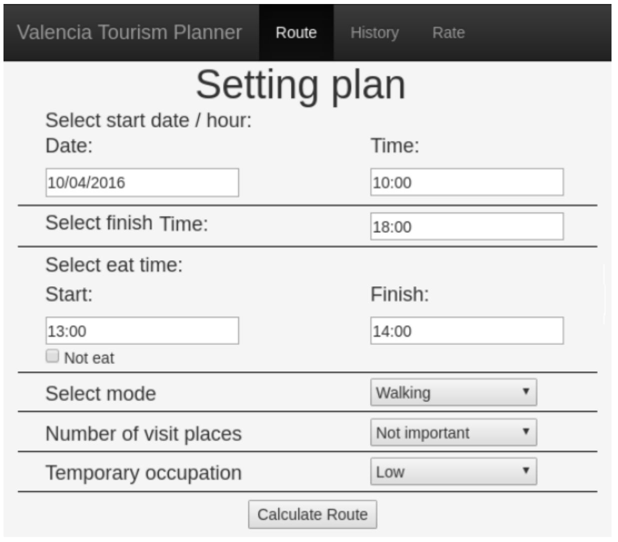

Figure 2. *e-Tourism2.0* system: agenda preferences.

of Valencia (Spain). It was intended to be a service for foreigners and locals to become deeply familiar with the city and plan leisure activities. *e-Tourism* makes recommendations based on the user's tastes, her demographic classification, the places visited by the user in former trips and, finally, her current visit preferences. One of the main components of *e-Tourism* is the planning module, aimed at scheduling the recommended activities. Thus, the output of *e-Tourism* is a real agenda of activities which not only reflects the user's tastes but also provides details on when to perform the recommended activities. Specifically, the construction of the agenda takes into account duration of the activities to perform, the opening hours of the places to visit and the geographical distances between places (time to move from one place to another). All this information is compiled into a planning problem that can be formulated as a *Constraint Satisfaction Problem* or as an *Automated Planning Problem* [8].

Let us take three tourists, Rose, Mark and David, interested in visiting Valencia. Rose and Mark like visiting museums, but Rose likes museums more than Mark. Both decide to visit the *National Museum of Ceramics*. Rose wishes to visit the museum for 2h30min, whereas Mark only wants to be there for about 1h30min. Moreover, since this is Mark's first time in Valencia, he would like to include quite a few POIs in his agenda, namely 5 POIs, and not to have much spare time between activities. Rose, however, visited Valencia last year and she would like to explore in depth two museums that she already visited last time. Therefore, she would like her agenda to contain only these two visits over a full day and no much free time between them. In contrast, David has been in Valencia several times and he would rather include in his agenda two or three quick visits and spare time to walk around and sit in a terrace to have a beer. These three examples show different travel styles around two preferences: the number of visits and the time spent in each visit. *e-Tourism2.0* handles taste preferences of the user as well as this new type of *travel style preferences*.

The *e-Tourism2.0* architecture is composed of five subsystems (Figure 1): the control node, responsible of coordinating the whole recommendation-planning process, the web application, the recommender system, the intelligent planner and the database.

2.1 Tourist agenda

We developed a new web-based interface which can be accessed through different devices such as computers, smartphones, tablets, etc. The first step in the construction of the tourist agenda is to **build the user model**. The user registers in the system and enters her personal details and general preferences. With this information the system builds an initial user profile. Besides, each time the user enters the system for a new visit she will be requested to introduce her specific preferences for the current visit, shown in Figure 2: the date of the visit *date*, her available time slot (T_s^{tour}, T_e^{tour}), the time interval reserved for lunch $(T_s^{lunch}, T_e^{lunch})$, the mode of transport she prefers - walking, driving or public transport -, her initial location $location_{initial}$ and final destination $location_{final}$. Moreover, she also indicates her preferences related to her travel style: $pref_{\#visits}$ indicates if the user prefers to include many or few visits in the tour or has no preference over it; and $pref_{occupation}$ indicates if the user prefers to obtain an agenda with a high or a low temporal occupation or has no preference over it.

The second step is to **generate a list of activities that are likely of interest to the user** by means of the *Generalist Recommender System Kernel (GRSK)*, which uses a mixed hybrid recommendation technique. A detailed description of GRSK can be found in [5]. The *intelligent planner* is in charge of **calculating the tourist agenda**, scheduling the activities recommended by the GRSK according to the restrictions of the environment and user preferences with respect to the configuration of the agenda. Figure 4 shows two agendas computed for a particular user and a map with the path she should follow. When the user logs again in the system, she is asked to **rate the activities** in the last recommended plan (through the option *Rate* in the top bar menu of Figure 2). The information obtained from these ratings is further used to improve the user profile and provide more suitable recommendations.

2.2 Database

The database schema of *e-Tourism2.0* is shown in Figure 3. We manage two sets of tables: those used for the recommendation process and those used for the planning process. Table `places` stores information about the POIs to recommend such as the name or the geographical coordinates. Table `users` contains personal details of the user, such as the name and other demographic data (this is neglected in Figure 3 for the sake of clarity). These two tables are used in both processes. The information used by the GRSK is: (1) tables `preferences`, `places_preferences` and `users_preferences`, which store the characteristics of the POIs to recommend and the user preferences inferred by the GRSK, respectively[2]; (2) tables `history` and `history_data`, which store the past interaction of the user with the system. The planner uses the information in table `timetables`, which stores a list of opening hours for each POI, and `movements_time`, that keeps the estimated and actual travelling time between two locations according to the value of `travel_mode` (see Figure 3).

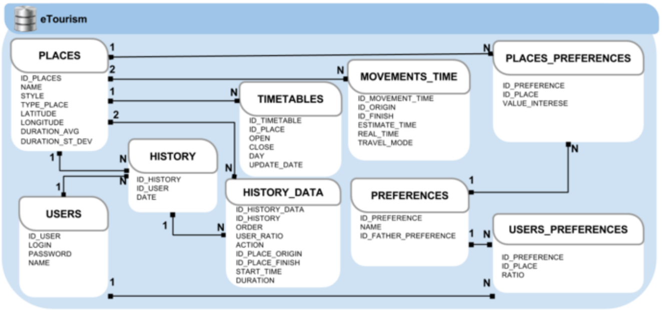

Figure 3. *e-Tourism2.0* system: database.

2.3 External data sources

As explained above, *e-Tourism2.0* accesses various web services in order to obtain some up-to-date information about location of restaurants and POIs, opening hours, transportation modes, etc. For obtaining this information, we selected the Google location and mobility web services, specifically:

- Google Directions[3] for obtaining a route (path) between two given coordinates, addresses or name of places. It is also possible to add some intermediate points in the path and to select the travel mode (walking, cycling, driving or with public transport)
- Google Places[4] for obtaining information about a given place. In *e-Tourism2.0*, this service has been used to elicit the opening hours of the places to visit and to find restaurants close to an specific place.
- Google Maps[5] for the visualization of the map along with the route provided to the user with the recommended places to visit.

Information like the catalog of POIs or the route between two places is stored in the database, which allows us to accelerate the process of calculating the recommendations and the plan. However, since information can become obsolete and needs to be updated, Google web services are periodically queried to update the data (see section 4 for more details).

3 RECOMMENDATION OF THE VISIT DURATION

The GRSK of *e-Tourism2.0* elicits the list of POIs or activities to include in the travel agenda of the user according to her preferences. This list is an ordered set of tuples of the form: $\left\langle a, Pr^a \right\rangle$, where a denotes the recommended activity and $Pr^a \in [0, 300]$ is the estimated degree of interest of the user in activity a.

For each activity a, we assign a duration in average, denoted by μ_a, which represents the recommendable duration of a for a typical tourist. The value of μ_a joint with σ_a define a normal distribution $X(\mu_a, \sigma_a^2)$. This is used by the GRSK to return a time interval that encompasses the minimum and maximum recommendable duration of a for the user according to Pr^a. Following the definition of the normal distribution, σ_a is computed as μ_a divided by α, so that, 68% of tourists spend $[\mu_a - \mu_a/\alpha, \mu_a + \mu_a/\alpha]$ minutes in visiting a, whereas about 4% of the tourists spend less than $\mu_a - 2 * \mu_a/\alpha$ or more than $\mu_a + 2 * \mu_a/\alpha$ minutes. In our experiments, we set $\alpha = 5$ and we empirically tested that consistent durations are returned. Our future objective is to estimate this distribution by studying the actual behaviour of tourists by means of an analysis of Twitter interactions, similarly to the analysis described in [10].

Once the normal distribution $X(\mu_a, \sigma_a^2)$ for each activity is defined, the recommended interval $(dur^a_{min}, dur^a_{max})$ is computed as $(X(Pr^a/300/2), X(Pr^a/300))$. That is, the values of probability that leave an area of the corresponding argument on the right. For example, let's assume that the a=*National Museum of Ceramics* has $\mu_a = 180$ and, therefore, $\sigma_a = 36$, meaning that a typical tourist would spend 180 minutes visiting this museum, and the dispersion for the other tourists is 36 minutes. Then, by the normal distribution, 68% of the tourists spend between [144,216] minutes in this visit and approximately 4% of the tourists spend less than 108 or more than 252 minutes. If the GRSK determines a degree of interest of 100 out of 300 for a given user, the duration interval will be $[145, 164]$, whereas if Pr^a is 260, the duration interval will be $[174, 220]$.

In [10], the visit duration is adjusted with the category of the activity a and the interest of the user in the category. However, durations in *eTourism2.0* are more accurate because we consider the degree of interest of the user in a, not in the category of a. Moreover, since the GRSK returns a tuple of the form $\left\langle a, Pr^a, dur^a_{min}, dur^a_{max} \right\rangle$ for each a, the planner can select the most appropriate duration within the interval the according to the travel style preferences of the user.

[2] A more detailed explanation about the domain ontology can be found in [4].
[3] https://developers.google.com/maps/documentation/directions/
[4] https://developers.google.com/places/
[5] https://developers.google.com/maps/documentation/javascript/

4 PLANNING PROBLEM SOLVING

The Control node receives the list of the recommended activities along with the recommended duration interval from the GRSK and generates the planning problem. Planning a set of recommended activities for a tourist requires some functionalities: (1) temporal planning and management of durative actions (e.g., duration of visits, time spent in transportation, etc.); (2) ability of reasoning with temporal constraints (e.g., scheduling the activities within the opening hours of places, planning the tour within the available time slot of the tourist, etc.) and (3) ability of reasoning with the tourist preferences (e.g., selecting the preferred activities of the user for planning the tour). Reasoning with time constraints and preferences simultaneously is a big challenge for current temporal planners.

Among the few automated planners are capable of handling temporal planning problems with preferences, we opted for OPTIC because it handles the version 3.0 of the popular Planning Domain Definition Language (PDDL) [7], including non-fixed durations and soft goals. Soft goals are preferences that we wish to satisfy in order to generate a good plan, but that do not have to be achieved in order for the plan to be correct. We need to identify and describe the preferences in PDDL3.0 as well as stating how the satisfaction, or violation, of these constraints affects the quality of a plan. Thus, the violation costs (penalties) associated to the preferences are considered at the time of selecting the best tourist plan; i.e., the plan that satisfies most tourist preferences and thereby minimizes the violation costs. This section describes the automatic generation of the corresponding planning problem in PDDL3.0.

4.1 Initial state

The specific values of the variables of a problem are described in the initial state by means of predicates and functions. The predicates and functions for an activity are:

- The interval duration of an action (activity) a is defined through the functions `(min_visit_duration ?a)` and `(max_visit_duration ?a)`. They will be assigned the values dur^a_{min} and dur^a_{max} returned by the GRSK, respectively.
- An activity a has an opening hour and a closing hour that are specified by a timed-initial literal: `(at` t_{open} `(open` a`))` and `(at` t_{close} `(not (open` a`)))`, to indicate when the activity is not longer available.

The duration of moving from one location p_j to another location p_k is defined by the function `(travelling_time` p_j p_k`)` that returns the time in minutes needed to travel from p_j to p_k by using the travel mode indicated by the user. If the duration of this action is not available in the DB from a past user, an estimated duration is calculated with the *Haversine* formula, used for calculating Earth distances, and the classical *uniform linear motion* formula, where *speed* depends on the mode of transport, and adding a small correction θ for awaiting times:

$$EstimTime(A,B) = \frac{Haversine(A,B)}{speed} + \theta * Haversine(A,B)$$

The predicate `(person_at ?l)` is used to represent the location of the user and the function `(total_available_time)` returns the available time of the user, which is initially set to $T_{finish} = T_e^{tour} - T_s^{tour}$.

We must note that web services are queried to obtain the initial data of the planning problem and that most of these data (timetables, distances between monuments) are stored in the database in order to keep the number of queries as low as possible and quickly retrieve the data during planning. In case a particular distance is not found in the database during the construction of a plan, we estimate the distance with the Haversine formula explained above, thus avoiding access to web services at planning time. Estimated times will be then updated after the planning process with the actual values by querying the corresponding web services.

4.2 Goal and preferences

We handle two types of goals: *hard goals*, that represent the realization of an activity that the user has specified as mandatory (e.g., the final destination at which the user wants to finish up the tour `(person_at id_hotelastoria)`); and *soft goals or preferences*, that represent the realization of a desirable but non-compulsory activity; e.g., visiting the *National Museum of Ceramics*: `(preference p1 (visit_location id_museumceramics))`.

The objective is to find a plan that achieves all the hard goals while minimizing a plan metric to maximize the preference satisfaction. This is expressed in the form of penalties, so that when a preference is not fulfilled, a penalty is added to the metric. Specifically, we define three types of penalties: for non-visited POIs, travelling times and the non-fulfillment of other configuration parameters of the agenda.

The *penalty for non-visited places*, aimed to help the planner to select the activities with a higher priority for the user, is calculated as the ratio between the priority of the activities not included in the plan Π and the priority of the whole set of recommended activities RA:

$$P_{non\_visited} = \frac{\sum_{a \in RA - \Pi} Pr^a}{\sum_{a \in RA} Pr^a}$$

The *penalty for movements* forces the planner to reduce the time spent in travelling from one location to another, so that closer activities are visited consecutively. This penalty is calculated as the duration of the *move* actions of Π, Π_m:

$$P_{move} = \sum_{a \in \Pi_m} dur(a)$$

Initially, the user defines her travel style preferences (see section 2): $pref_{\#visits}$ represents the preference for the number of visits and $pref_{occupation}$ denotes the user preference for the time to be spent in the visits or, conversely, for the free time between activities. The idea of combining both preferences is to give response to the different travel styles described in section 2. For example, Rose would set $pref_{\#visits}$ to "few" and $pref_{occupation}$ to "high". In order to take into account these preferences, two penalties are included.

$P_{\#visits}$ is the penalty that considers the user preference for the number of visits. It takes into account the number of visits in the plan with respect to the number of recommended activities:

$$\begin{cases} \frac{(|RA|-|\Pi_v|)}{|RA|} * T_{finish} & : \quad pref_{\#visits} = many \\ \frac{|\Pi_v|}{|RA|} * T_{finish} & : \quad pref_{\#visits} = few \\ 0 & : \quad pref_{\#visits} = indifferent \end{cases}$$

$P_{occupation}$ is the penalty that considers the user preference for the temporal occupation. Similarly to $P_{\#visits}$, $P_{occupation}$ takes into account the time that remains available in the plan with respect to the total time of the user.

$$\begin{cases} T_{finish} - \sum_{a\in\Pi} dur(a) & : \quad pref_{occupation} = high \\ \sum_{a\in\Pi} dur(a) & : \quad pref_{occupation} = low \\ 0 & : \quad pref_{occupation} = indifferent \end{cases}$$

Both penalties return a value in the interval $[0, T_{finish}]$. The combination of all these penalties defines the plan metric or optimization function to minimize by the planner:

$$P_{total} = P_{non\_visited} + P_{move} + P_{\#visits} + P_{occupation}$$

4.3 Actions

Three different types of actions are defined in this tourism domain. Due to space restrictions, we will only focus on the **visit** action. The input parameters of this action are the activity to perform `?a` and the user `?y`. The duration of the action is defined within the interval `(min_visit_time ?a)` and `(max_visit_time ?a)`. Moreover, this duration must be smaller than the remaining available time `(total_available_time)`. The planner will choose the actual duration of the action according to these constraints. The conditions for this action to be applicable are: (1) the user must be located in `?a` during the whole execution of the action; (2) the POI `?a` is open during the whole execution of the action and (3) the activity `?a` has not been performed yet. The effects of the action assert that (1) the activity is done, (2) the number of visited locations is increased and (3) the user available time is updated according to the activity duration. The action to perform the activity of *having lunch* is similarly defined to the action `visit`. The action of *moving* between locations essentially modifies the current location of the user, the available time of the user and the time spent in travelling from one location to another according to the duration stored in the database.

Regarding the periodical update of the information, only the location of restaurants and distances between restaurants and monuments are not retrieved beforehand because the list of restaurants is rather changeable. The planner deals with a 'dummy' restaurant, which is instantiated to a real restaurant that matches the user's tastes after planning.

5 CASES OF STUDY

In this section, we show some cases of study and we analyze whether the resulting plans of the OPTIC planner are compliant with the user preferences. We use two metrics to measure the plan quality:

$$O_\Pi = \frac{\sum_{a\in\Pi} dur(a)}{T_{finish}} \qquad U_\Pi = \frac{\sum_{a\in\Pi_v} (Pr^a * dur(a))}{\sum_{a\in\Pi_v} dur(a)}$$

O_Π is the occupation rate of the plan; i.e., the total time during which the user is performing some action (visiting, moving or having lunch). U_Π is the utility of the plan, defined as the rate between the priority of the activities performed in a given interval and the total duration of such activities. U_Π returns a value in [0, 300].

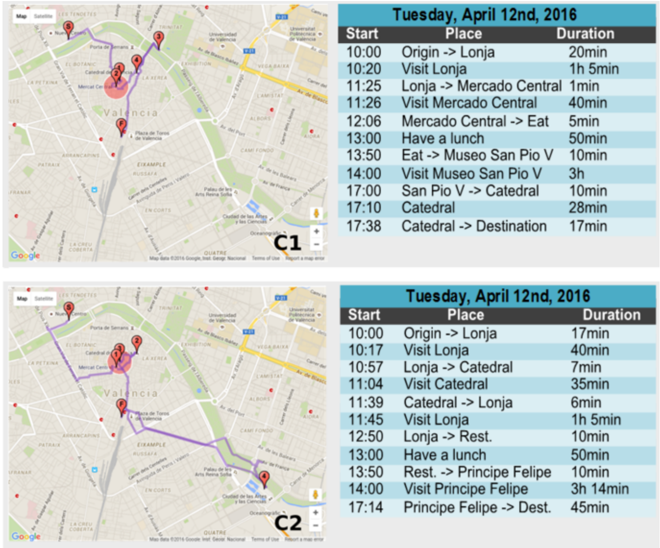

Figure 4. Plan generated for case studies C1 and C2

First, we performed a comparison to see how the selection of the mode of transport affects the final plan. Figure 4 shows the paths obtained for two cases: C1 and C2. C1 represents a basic case, where the user only specifies he would rather walk. In this case, the focus of the planner is on finding the best route taking into account the degree of interest of the user in the POIs, the opening hours and the reduction of the walking time. The resulting plan is an agenda with $O_\Pi = 90.625\%$ and $U_\Pi = 217.22$. The case C2 differs from C1 in that the user can either walk, or use the public transport when the distance between two consecutive places is greater than a threshold. In this case, the system generates routes that include POIs in which the user is highly interested but are far away from each other, returning an agenda with a higher utility. For example, the user is advised to use the public transport to visit *Museo Principe Felipe*, given that this POI is not within walking distance of the previous visited POI in the plan. In this case, $O_\Pi = 64.57\%$ and $U_\Pi = 251.62$.

In the next experiment, we selected a fixed initial and final locations, the available time slot, time reserved for lunch and transportation means, and we generated a set of cases with all the possible combinations of $pref_{\#visits}$ and $pref_{occupation}$. The results are shown in Table 1. Columns *#visits* and *occupation* indicate the value of the preferences $pref_{\#visits}$ and $pref_{occupation}$, respectively. Column *#POIs* shows the number of POIs included in the agenda, whereas columns *move* and *visit* indicates the percentage of the time devoted to move and visit actions, respectively. Finally, columns O_Π and U_Π indicate the occupation rate and the utility of the plan.

The results show that the preferences indicated by the user are effectively reflected in the agenda. We can observe that

#visits	*occupa*	#POIs	move	visit	O_Π	U_Π
Indiff	Indiff	3	7.2	28.7	52.59	220.96
Indiff	High	4	21.6	61.64	99.97	242.35
Indiff	Low	2	9.81	15.37	42.22	216.74
Many	Indiff	4	10.37	41.11	68.14	246.74
Many	High	4	21.66	61.64	99.97	242.35
Many	Low	4	10.37	37.59	64.62	217.68
Few	Indiff	2	6.6	19.44	42.77	236.66
Few	High	3	21.66	60.9	99.23	242.01
Few	Low	2	9.81	15.37	42.22	216.74

Table 1. Cases of study with different travel styles

when only one preference is set, clearly the other preference influences the final result of the agenda. For example, when $pref_{\#visits}$ is set to 'Indiff', the difference in O_Π is more than 57%. This also happens when $pref_{occupation}$ is set to 'Indiff', where the number of visited POIs goes from 4 to 2, depending on the value of $pref_{\#visits}$.

When $pref_{\#visits}$ is set to 'Many', the number of POIs is the highest (4), but we can observe a clear difference in O_Π depending on the value of $pref_{occupation}$: if it is set 'High', O_Π almost reaches 100%; and if it set to 'Low', then the value of O_Π is lower than the value obtained when $pref_{occupation}$ is 'Indiff'. We can find a similar situation when the number of visits is 'Few', where the only difference is that number of POIs to include in the agenda increases in 1 when $pref_{occupation}$ is set to 'High'.

In the resulting plans, that we do not show due to space restrictions, we have observed that when $pref_{occupation}$ is'Low', irrespective of the number of visits, the duration of the activity is usually set to the minimum value of the duration interval returned by GRSK. This is reflected in that the time of *visit* when $pref_{occupation}$ is 'Low' is always lower than the *visit* times when $pref_{occupation}$ is 'High' or 'Indiff'. Obviously, U_Π is also the lowest in these cases and the highest utility is always obtained when O_Π is also the highest.

The percentage of the time devoted to travelling actions is usually around 10%, except in the cases where the $pref_{occupation}$ is 'High'. This is because, in this particular case of study, the user must travel to a distant POI to obtain a high value of occupation.

The tourist-tailored plans obtained in the cases of study are the result of the planner's performance and of a faithful and consistent modeling of the user preferences and corresponding penalties.

6 CONCLUSIONS

This paper describes *e-Tourism2.0*, an enhanced recommendation and planning system for tourist activities in the city of Valencia (Spain). *e-Tourism2.0* offers a personalized recommendation of the duration of the visits suited to the interest of the user in the place to visit. It also handles user preferences related with the configuration of the agenda, particularly travel style preferences in terms of the number of places to visit and the desired temporal occupation of the tour.

We tested the adaptiveness of the plans to the user preferences through some cases of study. From the results we can conclude that an accurate modeling of the user preferences is very relevant to obtain plans that effectively reflect the tastes and travel style preferences of the tourist.

ACKNOWLEDGEMENTS

This work has been partly supported by the Spanish MINECO under the project TIN2014-55637-C2-2-R, and the Valencian project PROMETEOII/2013/019.

REFERENCES

[1] J. Benton, Amanda Jane Coles, and Andrew Coles, 'Temporal planning with preferences and time-dependent continuous costs', in *Proc. Int. Conference on Automated Planning and Scheduling*, (2012).

[2] Joan Borràs, Antonio Moreno, and Aïda Valls, 'Intelligent tourism recommender systems: A survey', *Expert Syst. Appl.*, **41**(16), 7370–7389, (2014).

[3] Luis A. Castillo, Eva Armengol, Eva Onaindia, Laura Sebastia, Jesús González-Boticario, Antonio Rodríguez, Susana Fernández, Juan D. Arias, and Daniel Borrajo, 'samap: An user-oriented adaptive system for planning tourist visits', *Expert Syst. Appl.*, **34**(2), 1318–1332, (2008).

[4] Inma Garcia, Sergio Pajares, Laura Sebastia, and Eva Onaindia, 'Preference elicitation techniques for group recommender systems', *Information Sciences*, **189**, 155–175, (2012).

[5] Inma Garcia, Laura Sebastia, and Eva Onaindia, 'On the design of individual and group recommender systems for tourism', *Expert Systems with Applications*, **38**(6), 7683–7692, (2011).

[6] Inma Garcia, Laura Sebastia, Eva Onaindia, and Cesar Guzman, 'e-Tourism: a tourist recommendation and planning application', *International Journal on Artificial Intelligence Tools (WSPC-IJAIT)*, **18**(5), 717–738, (2009).

[7] Alfonso Gerevini, Patrik Haslum, Derek Long, Alessandro Saetti, and Yannis Dimopoulos, 'Deterministic planning in the 5th International Planning Competition: PDDL3 and experimental evaluation of the planners', *Artificial Intelligence*, **173**(5-6), 619–668, (2009).

[8] Ghallab M., Nau D., Traverso P., *Automated Planning. Theory and Practice.*, Morgan Kaufmann, 2004.

[9] Yohei Kurata and Tatsunori Hara, 'Ct-planner4: Toward a more user-friendly interactive day-tour planner', in *Information and Communication Technologies in Tourism 2014 - Proceedings of the International Conference in Dublin, Ireland, January 2014*, pp. 73–86, (2014).

[10] Kwan Hui Lim, Jeffrey Chan, Christopher Leckie, and Shanika Karunasekera, 'Personalized tour recommendation based on user interests and points of interest visit durations', in *24th International Joint Conference on Artificial Intelligence, IJCAI*, pp. 1778–1784, (2015).

[11] Antonio Moreno, Aida Valls, David Isern, Lucas Marin, and Joan Borràs, 'Sigtur/e-destination: ontology-based personalized recommendation of tourism and leisure activities', *Engineering Applications of Artificial Intelligence*, **26**(1), 633–651, (2013).

[12] Arturo Montejo Ráez, José M. Perea-Ortega, Miguel Angel García Cumbreras, and Fernando Martínez Santiago, 'Otiŭm: A web based planner for tourism and leisure', *Expert Syst. Appl.*, **38**(8), 10085–10093, (2011).

[13] Paul Resnick and Hal R. Varian, 'Recommender systems', *Communications of the ACM*, **40**(3), 56–58, (1997).

[14] Beatriz Rodríguez, Julián Molina, Fátima Pérez, and Rafael Caballero, 'Interactive design of personalised tourism routes', *Tourism Management*, **33**(4), 926–940, (2012).

[15] Steffen Staab, Hannes Werthner, Francesco Ricci, Alexander Zipf, Ulrike Gretzel, Daniel R. Fesenmaier, Cécile Paris, and Craig A. Knoblock, 'Intelligent systems for tourism', *IEEE Intelligent Systems*, **17**(6), 53–64, (2002).

[16] Pieter Vansteenwegen, Wouter Souffriau, Greet Vanden Berghe, and Dirk Van Oudheusden, 'The city trip planner: An expert system for tourists', *Expert Syst. Appl.*, **38**(6), 6540–6546, (2011).