TRIP PLANNING FUNCTIONALITIES: STATE OF THE ART AND FUTURE

PIETER VANSTEENWEGEN* and WOUTER SOUFFRIAUF†‡

*Department of Industrial Management, Ghent University, Zwijnaarde, Belgium
‡Centre of Industrial Management, Katholieke Universiteit, Leuven, Belgium
‡dyNAVic, Leuven, Belgium

When tourists visit a city or region, they usually cannot visit every point of interest available as they are constrained in time and budget. Tourist recommender applications help tourists by facilitating personal selection. Providing adequate tour scheduling support for these kinds of applications is a daunting task for the application developer. The objective of this article is to demonstrate how existing models from the field of Operations Research (OR) fit this scheduling problem and enable a wide range of tourist trip planning functionalities. Using the Orienteering Problem (OP) and its extensions to model the tourist trip planning problem allows us to deal efficiently with a number of practical planning problems.

Key words: Trip planning; Recommender systems; Orienteering Problem (OP); Operations research

Introduction

Many tourists visit a region or a city for one or more days. It usually is not possible to visit every tourist attraction or cultural heritage site during such a limited period, so the tourist has to make a selection of what he/she believes to be the most valuable points of interest (POI). This personal selection is based on information found on websites, in articles in magazines, or in guidebooks from specialized book stores or libraries. Once the selection is made, the tourist decides on a route, keeping in mind the opening hours of the POIs and the available time.

Tourists face several difficulties when following that procedure. Information provided in guidebooks can be out of date (e.g., opening hours may change). Also, guidebooks cannot provide temporal information: temporary exhibitions in museums change all the time, some POIs are (partly) closed due to renovation, and theaters change their program regularly (Dunlop et al., 2004). Tourists have to combine the information from different sources and decide which information is the most reliable. Moreover, selecting the most valuable POIs (i.e., those of the greatest interest to the tourist) is not easy. Usually tourists will be happy if they devise a somewhat attractive and feasible schedule, but they have no idea whether better schedules are possible.

Some guidebooks acknowledge these problems and propose generic visitor tours through a city or region. Of course, these tours are constructed in order to satisfy the interests of the majority rather
than the specific interests of individuals (Cheverst, Davies, Mitchell, Friday, & Efstratiou, 2000). Generic visitor tours do not take user context into account (e.g., the start and end location, the available time, the current time, the weather, etc.). Kramer, Modsching, and ten Hagen (2006) have analyzed the diversity of gathered tourist interest profiles and conclude that they are surprisingly diverse. This conclusion supports the idea of creating personalized tours instead of proposing generic visitor tours. Furthermore, tourists today want to use their free time in an optimal way and they expect to be well informed on what a city or specific POI can offer (Keyson, 2004; Oppermann & Specht, 1999).

Web-based decision support applications are excellent aids for tourists who want real support for planning problems. Based on an interest profile, up-to-date POI information and trip information, a (near-)optimal and feasible selection of POIs and a route between them can be suggested (Vansteenwegen & Van Oudheusden, 2007). Also, most tourists today move within a limited crowded area of very attractive POIs. Kramer et al. (2006) state that a system enabling personal selection and routing of POIs will help to spread tourists more evenly across the destination region, which helps to prevent crowds.

Since World War II, the science of Operation Research (OR) has been applied to a vast range of problems in different sectors. OR is concerned with applying mathematics, statistics, optimization technology, etc., to provide decision makers with (near-)optimal solutions to complex problems in military contexts, manufacturing, transportation, logistics, finance, etc. However, the field of tourism, in particular personalized trip planning, has been largely ignored.

Godart (2001) uses the Travelling Salesperson Problem (TSP) as a starting point to plan trips. His TSP with Activities and Lodging Selection (ALS) automatically selects POIs and lodging. The Multiple Objectives extension (MOTSP-ALS) minimizes transport and accommodation costs at the same time. Finally, Preference Based MOTSP-ALS also maximizes the attractiveness of the lodging and the activities. This complicated model turned out to be difficult to solve. This research resulted in a web-based tour planner, YourTour (http://www.yourtour.com).

Vansteenwegen and Van Oudheusden (2007) advocate the use of the OP and its extensions to solve Tourist Trip Design Problems (TTDP). The OP integrates automated selection of locations with finding the shortest path, and is therefore highly appropriate to model TTDPs. The objective of this article is to demonstrate how a wide range of real-life tourist trip planning functionalities can be enabled by using the OP. First, an overview of systems that compose tours of POI visits is presented. The next section discusses the wide range of planning-related functionalities that are offered by these systems and compares them based on the functionality they offer. Then how the OP and its extensions can be used to model trip planning functionalities is explained. The City Trip Planner system is then evaluated in detail. Finally, conclusions are presented.

State of the Art

Instead of recommending prepackaged tours, or sorting POIs by estimated interest value as recommender systems do, scheduling approaches typically try to determine the combination of POIs that maximize the joint interest. Soo and Liang (2001) present a software agent that recommends a trip plan through dialogue with the user. Custom trips to the city of Taipei, China, are proposed by first letting the tourist select his hotel(s) and next automatically filling the available time with POI visits in a nearest-neighbor fashion. Ardissono, Petrone, Segnan, and Torasso (2002) and Ardissono, Goy, Petrone Segnan, and Torasso (2003) describe their INteractive TouRist Information GUidE (INTRIGUE) for the city of Torino, Italy. It is a fuzzy logic-based recommender system that is able to provide an explanation why a recommendation has been made. Moreover, INTRIGUE has a tour scheduling functionality that enables composing group tours, taking into account opening hours and locations of the POIs and time restrictions of the tourist group members. Unfortunately, very little technical details on the scheduling algorithm are provided.

Suna and Lee (2004) present a multiagent system that advises personalized tourist routes using a vector-based recommendation technique to calculate personal interest values in geographical objects. A shortest path algorithm minimizes the normal cost of arcs in the road network divided by
their personal interest values in order to calculate personalized point-to-point routes. A tourist trip is calculated by modeling the POI selection problem as a prize collection TSP, which is to be solved with the method of Dell’Amico, Maffioli, and Sciomachen (1998). Despite the promising ideas, Suna and Lee (2004) do not evaluate the system thoroughly.

Maruyama, Shibata, Murata, Yasumoto, and Ito (2004a, 2004b) present P-Tour, a personal navigation device that calculates tourist routes. They use a variant of the TSP with profits that aims at finding a circuit that minimizes travel costs minus collected profit (Feillet, Dejax, & Gendreau, 2005). The P-tour routing algorithm selects and routes a number of POIs that are defined by a location, a visiting duration, an importance score, and an optional constraint on arrival time. Maruyama et al. (2004a) maximize the weighted sum of (1) the importance of selected POIs, (2) the importance of selected POIs that satisfy a time restriction, minus (3) the total travel distance, while Maruyama et al. (2004b) only use (2) minus (3). The route search engine achieves a gap of less than 2% from the optimal solution in 15 seconds, for one small instance of 30 POIs. Users have to enter personal importance scores themselves for each POI.

Shiraishi et al. (2005a, 2005b) extend P-Tour in two ways. First, they search for undesirable situations during the execution of the planned route (wrong route, behind or ahead of schedule) and warn the user in an appropriate manner. Second, the route search engine is extended to take multiple conflicting evaluation functions into account: they contrast the weighted sum function of Maruyama et al. (2004b) together with the total travel expense. Kinoshita et al. (2006) extend P-Tour for multiple days. The total set of POIs is partitioned across different areas. Every day a selection of POIs of a predetermined area is to be visited, including accommodation to spend the night. A 3-day instance is solved in 19.6 seconds with a gap of 0.7% from optimality, taking into account the prepartitioning of the POIs, which possibly discards significantly better solutions. Nagata, Murata, Shibata, Yasumoto, and Ito (2006) extend the P-Tour system in order to plan group tours. Every member of the tourist group is allowed to state a preference value, a duration, and a latest arrival time for each POI. Also, every member has a starting and an end point, a time restriction on the total tour, and a speed. The objective is to find a schedule that forks and joins the group along the way to visit POIs.

Wu, Murata, Shibata, Yasumoto, and Ito (2009) extend the P-Tour system to take the weather forecast into account. For each POI, a timetable is given that contains probabilities for fine, cloudy, and rainy weather, every hour. POI preference values are dependent on the current weather while visiting it. The objective is to construct a decision tree that maximizes the total expected satisfaction degree. Limited experimental results are presented: an instance of 6 POIs is optimally solved in 6 seconds, compared to 16 hours with brute force search, and on an instance with 20 POIs, the quality of the greedy construction heuristic is improved 17.9% on average. Overall, P-Tour and its extensions provide interesting ideas, but fail to provide extensive computational experiments.

The Dynamic Tour Guide (DTG) of ten Hagen, Kramer, Hermkes, Schumann, and Mueller (2005) calculates personal tours on-the-fly. A tour is a collection of so-called Tour Building Blocks (TBB): sights, restaurants, etc. An ontology consists of a tree of concept classes and describes each TBB. TBBs receive Interest Matching Points by a semantic matching algorithm, representing the interest of the user. After the assignment of Interest Matching Points, an algorithm constructs a tour by maximizing the sum of the Interest Matching Points within a given time frame. The algorithm keeps a list of candidate TBBs, sorted by use of gain (i.e., the Interest Matching Point divided by the cost needed to visit the TBB). From the list, the TBBs are removed and inserted in the tour randomly until the available time runs out.

J. Lee, Kang, and Park (2007) present a tourist tour planning system for the Jeju province in Korea. They adopt the interest estimation method of Kang, Kim, and Cho (2006) in order to compute vector-based similarities between POIs and users. Next, they add a maximum length constraint to their TSP formulation of the planning problem. Actually, without mentioning it, they use the OP as a model. However, they tackle the problem by solving 2^3 distinct TSPs of n POIs, instead of 1 OP, which is a computationally very expensive approach. Their high performance cluster manages to offer a solution within 5 seconds when n < 22.
Castillo et al. (2008) present a multiagent-based system for planning tourist visits. A user agent first captures the user’s interest. Next, a Case-Based Reasoning agent predicts interesting activities. Finally, the planning agent takes these interesting activities as input and outputs a plan. The planning takes the following items into account: opening hours, preferences of the user, prices of meals and transports, locations, and multimodal means of transport. Two types of goals can be specified: totally and partially instantiated goals (e.g., visit a specific museum, respectively a type of museum). The planning problem is translated to predicate logic. Predicate logic AI planning modules use tree search to come up with a feasible plan and route to perform the activities. There is no integration of selection and routing, and no evaluation of the proposed system is presented.

C.-S. Lee, Chang, and Wang (2009) present a recommendation system that allows planning personalized travel routes to Tainan City, China. Their ontology-based multiagent system consists of a context decision agent and a travel route recommendation agent. The context decision agent first finds concepts of the ontology that match the tourist’s requirements. Next, the travel route recommendation agent uses fuzzy logic to select and sort a top three of historic sites and a top five of local gourmet food stores. A TSP that deals with these eight locations is solved. Experimental results present two small examples as evaluation, but no performance benchmarks. The system makes a distinction between selection and routing.

Niaraki and Kim (2009) developed a method for personalizing route planning network impedances. They evaluate multiple criteria that are defined in an ontology describing road segments. The user states his preferences for attributes such as traffic volume, safety, POI presence, etc., based on which the weights in the road graph are calculated. Common shortest path algorithms such as Dijkstra (Gallo & Pallattino, 1986) and A* (Pearl, 1984) can be used to calculate personal point-to-point routes.

Yu and Chang (2009) developed a framework for the personalized recommendation of hotels, restaurants, and POIs. They combined these three functionalities in a tour recommendation process that recommends a personalized tour based on the user’s current time and location and his interests. A prototype was built for the city of Taipei, Taiwan.

We developed the City Trip Planner (http://www.citytripplanner.com), which is a web-based tourist decision support system that proposes city trips tailored to the user’s context and personal interests (Vansteenwegen, Souffriau, Vanden Berghe, & Van Oudheusden, 2011). The system plans city visits of multiple days, with for each POI multiple time windows which can differ from day to day. Moreover, lunch breaks can also be scheduled and the local tourist office can suggest a few POIs to be included in a trip. The City Trip Planner integrates the selection of POIs and the routing between them. To the best of the authors’ knowledge, such an integrated system is unique. It uses the OP to model trip planning problems and fast heuristic algorithms (Vansteenwegen, Souffriau, Vanden Berghe, & Van Oudheusden, 2009) to solve them. A detailed evaluation and analysis of the performance of the City Trip Planner is presented below.

### Planning Functionality for Tourist Decision Support

This section first discusses the wide range of planning-related functionalities that are offered by the tour scheduling systems. Next, a comparison between these different systems is provided, based on the functionality they offer.

**Personal Interest Estimation** quantifies the interest of the tourist in a particular POI, the appropriateness of a hotel or the “beautifulness” of a scenic route. This quantified value can be used to sort POIs and hotels when presenting them to the user. Attributes of the user, which are collected into his/her user profile, are to be matched with attributes of the location or activity.

**Selection and Routing** automatically presents a customized route based on the user’s current location, his/her destination, and his/her available time, which limits the length of the route. This limit implies a selection of POIs, in order to keep the length of the route feasible. When combined with personal interest estimation, the resulting route is tailored to the user’s interest.

**Mandatory POIs** can be considered as “must
see.” Whenever the tourist is in the neighborhood of a mandatory POI, it should be presented at the top of his/her preference list. A POI can be determined mandatory by the provider of the POI information. A tourist can also indicate a POI as mandatory, if he/she is aware of a POI he/she certainly wants to visit (i.e., because the POI is recommended by friends or family).

**Mandatory Types** enable the tourist to state that a tour or a trip should contain at least one visit of a certain type (e.g., a visit to a church). Mandatory types extend the concept of mandatory POIs.

**Weather Dependency** influences the estimated appreciation of POIs by taking the weather forecast into account. When rain is expected during a given (part of the) day, the personal interest estimation in outdoor visits can be decreased for that period. It is better not to visit some of those POIs or visit those during another (part of the) day.

**Dynamic Recalculation** is needed when unexpected events occur. Especially during a tourist trip, planning is frequently revised. On vacation, people often do something different from what was anticipated. Maybe tourists stay longer than planned in a particular place. This could mean that another POI for that day has to be skipped or that some POIs are shifted to another day. A certain POI might turn out to be less appealing and be left early. Dynamic recalculation detects these infeasibilities and presents a new plan to the user, in “real time.”

**Multiple Day Decision Support** enables planning for multiple days. The user receives a selection of POI visits for a series of days. Each POI visit only appears once in the total selection.

**Opening Hours** should be taken into account when visiting the “interior” of POIs. Therefore, the route of selected POIs should take into account the time of the scheduled POI visits, making sure that each visit is planned when the POI is open. The opening hours of each POI are defined by means of a calendar. In the simplest case, a POI is open for a consecutive period during the day, with one opening time and one closing time, for all days. However, POIs can be closed during lunch, resulting in two opening periods during the same day. Moreover, opening hours tend to differ on different days: a POI can be closed on Sunday afternoon.

**Budget Limitations** arise when the tourist has a maximum amount of money to spend. Next to the time budget of the selection and routing functionality, a money budget further constrains the selection of POI visits.

**Max-n Type** constrains the selection of POIs by allowing to state a maximum number of certain types of POIs, per day or for the whole trip; for example, maximum two museum visits on the first day or maximum three churches during the whole trip.

**Scenic Routes** allow visiting beautiful routes, next to interesting locations. When moving from one POI to the next, a tourist will not mind a small detour through a car-free street with medieval façades. Although this is not the shortest path between the two POIs, it will be appreciated more than a walk through a regular street. A scenic route can also be a beautiful walk along the river side.

**Hotel Selection** automatically selects appropriate hotels when visiting a region for multiple days. The personal interest in the different hotels can be estimated, based on attributes of the hotels such as comfort. The automated selection mechanism will need to take the price of a stay into account, in function of the budget of the total trip. Integrating the selection of hotels on the trip planning leads to better results, nicer trips, than first fixing the hotels and then plan the POIs to visit in between.

**Public Transportation** takes into account metro, train, and bus schedules when traveling between POIs. These alternatives to walking need to be considered when distances between POIs are large. Some worthwhile attractions, easy to reach by bus or metro in only 10 minutes, but at least 1 hour away on foot, will never be included in the trip without the integration of public transportation. Using public transportation can save considerable amounts of time that could be spend on POI visits.

**Group Profiles** enable planning for groups of tourists, which differs considerably from single-tourist planning, as a group of tourists may have a broad, possibly conflicting, range of interests. Possible strategies include optimizing the joint interests of the group members by selecting locations they are all interested in, or taking turns and alter-
nating interests so that no one feels he has been left out.

Table 1 presents a match of the existing tour scheduling approaches with the different functionalities presented above. Only those functionalities that appear in two or more approaches are included. Unknown features have been marked as "?". For the sake of completeness, Castillo et al. (2008) are the only ones to mention mandatory types and public transportation, Shiraishi et al. (2005a, 2005b) are the only ones to tackle dynamic recalculation, Wu et al. (2009) weather dependency, Niaraki and Kim (2009) scenic routes, and Kinoshita et al. (2006) hotel selection.

All of the presented approaches use some form of personal interest estimation, except for P-Tour and its extensions, which in turn present a wide range of planning functionalities. A number of approaches offer integrated selection and routing. Soo and Liang (2001) and Yu and Chang (2009) use a nearest neighbor approach, which iteratively adds the closest available visit to the tour. Suna and Lee (2004), P-Tour and its extensions, except Wu et al. (2009), use the profitable tour problem as a basis for selection and routing. Ten Hagen et al. (2005) use a tree-based search, while Lee et al. (2007) combine a selection problem with a TSP. Vansteenwegen, Souffriau, Vanden Berghe, et al. (2011) use fast metaheuristics to deal with extensions of the OP.

Based on these developments, it can be concluded that providing automated POI selection and routing is an upcoming trend in tourist recommender applications. It appears that a large amount of research effort is still required in order to devise efficient tourist decision support techniques that are able to propose customized tours with acceptable response times. Providing adequate planning support for these kinds of applications is a huge research opportunity in the field of OR.

### Modeling the Tourist Trip Planning Problem

This section explains how the OP and its extensions can be used to model trip planning functionalities. This approach is applied, for instance, in the City Trip Planner mentioned above. An OP is a mathematical optimization problem that consists of a set of locations that are determined by coordinates, a visiting duration, and a score. The pairwise travel times between the locations are known. The goal is to find a tour that maximizes the total score earned by visiting locations. The start and end of the tour do not need to coincide. The total travel and visiting time cannot exceed a predetermined value, which is called the time budget. Each location can be visited at most once.

---

Table 1
Overview of Tourist Trip Planning Functionalities and Tour Scheduling Approaches

<table>
<thead>
<tr>
<th>Personal Interest Estimation</th>
<th>Distinct Selection and Routing</th>
<th>Integrated Selection and Routing</th>
<th>Mandatory POIs</th>
<th>Multiple Days</th>
<th>Opening Hours</th>
<th>Budget Limitations</th>
<th>Group Profiles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Soo and Liang (2001)</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Suna and Lee (2004)</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maruyama et al. (2004a, 2004b)</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Shiraishi et al. (2005a, 2005b)</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kinoshita et al. (2006)</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nagata et al. (2006)</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wu et al. (2009)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ten Hagen et al. (2005)</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J. Lee et al. (2007)</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Castillo et al. (2009)</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>C.-S. Lee et al. (2009)</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Niaraki and Kim (2009)</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Yu and Chang (2009)</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vansteenwegen et al. (2011)</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>
All locations with a score represent POIs. The score represents the estimated personal interest of the tourist in the POI, and can be calculated by means of POI recommender techniques. The time budget obviously represents the maximum amount of time the tourist has available for visiting POIs and traveling between them. A solution to the OP represents a tourist route. It is obvious that solving the OP entails an integrated solution of the selection and routing problem a tourist faces. This section indicates how each planning functionality from above can be modeled by (an extension) of the OP.

Mandatory POIs can be included as locations with a score that is higher than the sum of the scores of the nonmandatory POIs. As a consequence, the mandatory POIs will always be preferred over the other POIs. A quality algorithm will always include these mandatory POIs in the trip, if that is feasible.

Dynamic recalculation can be achieved by a fast algorithm, capable of offering immediately a new solution in case of unexpected events that lead to a new TTDP instance. When a certain visit takes much more time than expected, the remainder of the trip needs to be adjusted. The ending point and time of the TTDP instance stay the same, the starting point and time will be the current place and time of the tourist, and, obviously, POIs that were already visited will not be considered anymore. To make sure the trip is not changed too much during each recalculation, the objective function could be modified to favor POIs included in the original plan.

The “Team OP (TOP)” extends the OP by allowing multiple tours, each limited by a time budget. Again, each location can be visited at most once. The TOP allows modeling TTDPs for multiple days. Each tour or vehicle represents 1 day from a multiday tourist trip.

Furthermore, the TOP can be extended with multiple constraints, in which each location has Z “attributes.” These attributes correspond to characteristics of the POI: an entrance fee, does the POI belong to a certain type, etc. Z additional constraints are defined, which limit the selection of vertices. In the envisioned tourist application, these additional constraints enable to model budget limitations to spend on entrance fees, max-n types for each day and for the whole trip, and mandatory POI types. These three planning functionalities are now explained one by one.

In case of budget limitations, a POI attribute is used to represent the entrance fee for a POI, and an extra constraint defines the money budget available to spend. Max-n types are modeled in a similar way: a particular POI attribute is set to 1 if the POI is of a particular type, 0 otherwise, and the constraint defines the maximum number of visits of that type in the whole trip. Note that the model also enables max-n type and budget constraints to be defined per day (e.g., visit maximum one church on the first day, or spend at most 100 euro on the second day). In this case, extra constraints are added for that particular day.

Mandatory POI types (e.g., visit at least n churches) are a bit more complicated to model. Firstly, a copy is made of every POI of the mandatory type and the copied POI receives a score that is higher than the sum of all regular visits (cf. a mandatory POI). Secondly, two types of constraints are required in order to obtain the wanted result. (1) For each couple of POIs, an extra constraint is added, indicating that the original POI and the copied POI cannot be visited both. (2) For all copied POIs of the considered type, one extra attribute is added and set to 1 for a copied POI and 0 for an original POI. A constraint is added, which limits the total selection of copied POIs to the preferred minimum number of POIs of that type. When during the search this preferred number is not yet reached, the copied POIs will be preferred over their original counterparts (and all other POIs) due to their higher score. But as soon as the minimum number of POIs of that type is reached, all other copied POIs of that type will not influence the search any further. However, the original POI visits will still be considered with their normal scores.

The most studied extension of the TOP is the TOP with Time Windows (TOPTW). In this extension, each location is assigned a TW, with an opening time and a closing time. A visit to a location can only start during this time window. On arrival before opening, waiting is allowed, until opening, in order to collect the score. A feasible solution does not violate any TW constraint. The TOPTW enables modeling opening hours of POIs.
in the TTDP. However, only one TW can be defined per location, implying that a POI can have only one opening and closing time per day. Moreover, in the case of multiple days, a POI has an identical opening and closing time for any day in the standard TOPTW.

This drawback can be overcome by extending the TOPTW to the Multiple Constraint TOP with Multiple Time Windows. Each location is extended by allowing $W$ TWs for each day, instead of one. Also, the TWs can be different on different days. This enables modeling opening hours of POIs by allowing multiple opening hours per day, and different opening hours on different days. The multiple constraints extension described above can be used to model multiple TWs for one POI (on the same day or on different days) by multiple POIs with a single time window: POIs with multiple TWs are split up into different POIs on the same location, with one TW each, and an extra constraint allows at most one visit to location; only one of the different POIs can be actually visited. Thanks to this way of modeling, problems with different time windows on different days and multiple time windows per day can be solved with the same techniques as the regular Multiple Constraint TOPTW (MC-TOPTW). An algorithm to deal with the MC-TOPTW in real time is available and described in Souffriau, Vansteenwegen, Vandenberghe, & Van Oudheusden (2010a).

Weather dependency can also be modeled by splitting up a POI: when rain is expected in the afternoon, an outdoor POI can be split up in a POI with a TW before noon with a higher score and a POI with a TW in the afternoon with a lower score, and only one of both POIs can be visited. However, determining the most appropriate scores for both POIs is not straightforward.

Scenic Routes can be modeled as “arc POIs” in the orienteering problem. Instead of being placed at a given location, an arc POI starts in one location and ends in another and has a visiting time and possibly opening hours. An orienteering problem with both regular and arc POIs is called a “Mixed Orienteering Problem” (Vansteenwegen, Souffriau, & Van Oudheusden, 2011). The personal interest score of the arc POI can be determined in the same way as for normal POIs.

In order to model Hotel Selection in the OP, the starting and ending point of each day should be selected from a limited set of possible hotels. This makes the problem significantly more difficult to solve, but solution techniques for related problems such as the “travelling salesperson problem with hotel selection” (Vansteenwegen, Souffriau, & Sørensen, in press) illustrate that obtaining high-quality solutions in real time is still possible.

The integration of Public Transportation in the model also makes it much more complicated. Public Transportation can be modeled by “time-dependent travel times.” This means that the length of the travel time between two POIs depends on the moment the first POI is left. Optimization problems with time-dependent travel times are well known in the field of OR and appear to be challenging.

Introducing Group Profiles is possible without changing the OP model that is used, but it requires changing significantly the way in which the interest estimation in each POI is determined. Another possible approach is multiobjective optimization, in which each objective corresponds to one person’s interests. This would significantly complicate the OP model and the solution strategies.

This section illustrated that all planning functionalities of the previous section can be modeled by variants of the OP. The next section will evaluate the City Trip Planner, a publicly available trip planning system that implements an extension of the OP to assists tourists.

The City Trip Planner (and the OP model) was evaluated by user statistics and feedback. The user statistics were collected by means of Google Analytics (www.google.com/analytics), an online tool providing traffic statistics to a website. Data were collected for a period of 6 months after the public launch of the system (July 7, 2009 to January 5, 2010). There were 37,663 unique visitors that used the system 45,599 times, resulting in an average of 1,449 visitors per week or 207 visitors per day. Thirty-four percent of the users actually started planning a trip and two thirds of the users who started the trip planning wizard also complete it;
TRIP PLANNING FUNCTIONALITIES

20% of the tourists receiving a trip suggestion decided to use the print or download functionalities. Apparently, many users just try out the website or use it to get a first idea about the touristic opportunities of the city. Nevertheless, the city tourist offices are enthusiastic about the number of visitors of the application.

An average visit takes only 4 minutes and 5 seconds and is composed of 7.69 page views. Inputting user data takes on average 1.5 minutes. The user spends 2 minutes on average checking his personal trip proposal and reading the detailed POI information. Averaged over all trips, 1 day of a trip consists of 26 activities on average, including start, lunch break, and end. Many of these activities are only short visits (5 minutes or less) to look at a statue or at the facade of a building.

The website allows the users to give feedback in a questionnaire. During the first 6 months, 56 visitors filled in this form, of which 25 reported small technical malfunctions. The results of the other 31 users are summarized here. Almost all users “disagree completely” or “disagree” with the statements: “Completing the forms to construct my profile took too much time,” “The whole process to receive a personal trip took too much time,” and “I had to change a lot before receiving a satisfying trip.” At the same time, almost all users “agree” or “agree completely” with the statements: “The proposed attractions meet my interests,” “It was always immediately clear what was expected from me,” and “The City Trip Planner is very clear and easy to use.” When asked for remarks or other suggestions, some users suggested extra functionalities such as enabling to save the trip for the next visit or include booking of restaurants and hotels. Others suggested translating the content to other languages, as it is currently only available in English and Dutch. Overall, apart from the technical malfunctions reported, the remarks concerning the functionality were (very) positive. A more extensive evaluation, with more user statistics, is available in Vansteenwegen, Souffriau, Vanden Berghe, et al. (2011).

The City Trip Planner system proves that the OP and its extensions are very appropriate to model personalized trip planning. If the tourist trip planning problem is modeled as an (extension of the) OP, a large battery of algorithms are readily available for reusage. The interested reader is referred to Vansteenwegen, Souffriau, and Van Oudheusden (2011) for a recent survey on solution techniques for the OP (and its extensions).

Conclusions

Providing adequate tour scheduling support for tourist decision support applications is a daunting task for the application developer. An overview of systems that compose tours of POI visits is presented. The wide range of planning-related functionalities that are offered by these systems is discussed and the systems are compared based on these functionalities. However, the cited papers provide almost no information about the usefulness of their technologies. Next, this article demonstrates that existing OR models enable a wide range of current and future tourist trip planning functionalities.

We present and evaluate the basic OP model, as it integrates selection and routing of tourist attractions. Mandatory POIs can be easily incorporated and dynamic recalculation is achieved by using fast solution techniques, which present solutions in nearly real time. The model is iteratively extended with multiple tours, multiple constraints, time windows, and multiple time windows. This article explains how these extensions can be used to enable the planning of multiple days, budget limitations, max-n types, mandatory POI types, taking opening hours of POIs into account, and how to tackle weather dependency. We don’t mention solution methods in this article, because it focuses only on the state-of-the-art functionalities and modeling of the problem. Solution methods are explained in other papers and a survey is available in Vansteenwegen, Souffriau, and Van Oudheusden (2011). Moreover, the publicly available system, called “City Trip Planner,” shows that algorithms are available to deal with the OP and its variants for real-life tourist planning problems. The system is evaluated based on user statistics and feedback. In addition to the City Trip Planner, we have also incorporated the OP model into another web-based tourist decision support system, namely an online cycle route planner for the province of East-Flanders that offers personalized cycle routes based on user preferences. This web-
based system is extended with an SMS service that provides cyclists “in the field” with routes on demand (Souffriau et al., 2010b). Future work includes incorporating decision support for scenic routes, hotel selection, public transportation, and group profiles.

Biographical Notes
Pieter Vansteenwegen is an assistant professor at the Department of Industrial Management of Ghent University. In 2008, he obtained a Ph.D. in Operations Research at the Faculty of Engineering of the Katholieke Universiteit Leuven. His research is about the development of metaheuristics to deal with complex real-life problems in real time. He is cofounder of the spin-off company dyNAVic, commercializing his Ph.D. research on personalized tourist guides.

Wouter Souffriau obtained his Ph.D. in Operations Research in 2010 at the Faculty of Engineering of the Katholieke Universiteit Leuven. His research focused on the design and implementation of metaheuristics to deal with automated tourist trip planning. He is cofounder of the spin-off company dyNAVic which is commercializing his Ph.D. research about the City Trip Planner (http://www.citytripplanner.com). He currently works full-time for dyNAVic.

References


