Intelligent Routing System for a Personalised Electronic Tourist Guide

Ander Garcia[a],
Maria Teresa Linaza[a],
Olatz Arbelaitz[b] and
Pieter Vansteenwegen[c]

[a] Dept. of Tourism, Heritage and Creativity
Visual Communication Technologies VICOMTech, Spain
{agarcia, mtlinaza}@vicomtech.org

[b] Dept. of Computer Architecture and Technology
University of the Basque Country, Spain
olatz.arbelaitz@ehu.es

[c] Centre for Industrial Management
Katholieke Universiteit Leuven, Belgium
Pieter.Vansteenwegen@cib.kuleuven.be

Abstract

When tourists are at a destination, they typically search for information in the Local Tourist Organizations. There, the staff categorizes tourists' profile and restrictions. Combining this information with their up-to-date knowledge about the local attractions, weather and public transportation, they suggest a personalised route for the tourist agenda. This paper presents an intelligent routing system for a Personalised Electronic Tourist Guide to fulfil the same task. This system improves the automatic route creation functionality of existing PETs to solve better the needs of tourists in several aspects: i) it includes public transportation, ii) it takes varying travelling times into account, adapting to real circumstances as rush-hours, iii) it calculates routes in real time to react to unexpected events, iv) it applies last generation heuristics from Operations Research to create routes efficiently, even in destinations with a large number of point of interests and a dense public transportation network.

Keywords: Personalised Electronic Tourist Guide, Team Orienteering Problem Time Windows, public transportation.

1 Introduction

Nowadays, the creation of on-site personalised tourist routes that take into account the profile and other restrictions of tourists and up-to-date tourist information is a time consuming task that is often done by the staff of the Local Tourist Organizations (LTOs).

When tourists are at a destination and search for information in the LTO, the staff categorises their profile (cultural, romantic, family …) and restrictions (time, money, transportation means…). Combining this information with their knowledge about local attractions (location, prices, timetables, how appealing they are for a certain tourist profile…), they suggest a personalised route for the tourist agenda. However, this route does not take into account new circumstances that may happen during the visit (longer time in the visit of an attraction, changes in the weather, delay in transportation …).

The problem a tourist faces (Brown & Chalmers, 2003) is to decide which attractions to visit or to filter and to select what activities to do. The next step is to time-sequence these attractions or activities and to decide how to move from one attraction to the following one. Tourists solve this problem many times during the trip due to deviations from the original planning.

Among others, a Personalised Electronic Tourist Guide (PET), which is a mobile hand-held device, should perform the same task fulfilled by the LTO (Vansteenwegen & Van Oudheusden, 2007), solving the problem summarised by Brown and Chalmers (2003). A PET should provide an integrated solution for route planning that adapts to new circumstances. Furthermore, transportation information was identified as one of the most appreciated functionalities of a PET (Beer et al, 2007; Schmidt-Belz et al., 2003; Stroobants, 2006).

This paper presents an intelligent routing system for a PET, which gathers information about the profile and context of the tourists and attractions. It combines this information, using a novel algorithm capable of creating personalised tourist routes. Such routes react to events in real time, taking into account public transportation to improve the mobility of tourists. The system can target large destinations with an important number of attractions due to the application of efficient algorithms from Operations Research (OR) (Vansteenwegen et al., 2008). The system will be validated during the summer of 2009 at the city of San Sebastian.

This paper has been organised as follows. Section 2 summarizes the related work. Section 3 introduces the system, including the final validation scenario. At section 4 the details of the system and the algorithm are presented. The paper concludes with the evaluation of the system and some conclusions.

2 Related work

2.1 Mobile tourist guides

As power capabilities, network technologies and portability of mobile devices have improved, several applications aiming at creating mobile tourist guides have been implemented. They have been named Mobile Tourist Guide (MTG), Personal Navigation System for tourism (PNS), Electronic Tourist Guides (ETG) or Personalised Electronic Tourist Guides (PET). There are several reviews of existing PETs available for interested readers (Kramer et al, 2007; Souffriau et al., 2008). Table 1 summarizes the routing capabilities of many prototypes.

Table 1. Routing capabilities of existing PETs (based on Vansteenwegen, 2008)

	HIPPIE	HyperAudio	Cyberguide	Gulliver's Genie	GUIDE	CRUMPET	Taeneb	Mobility Agebt	eNarro	etPlannet	DTG	PNS
Location awareness	x	x	x	x	x	x	x	x	x	x	x	x
Personalised	x	x	x	x	x	x	x	x	x	x	x	
Standard tour available					x				x			
Navigation guidance between attractions	x		x		x	x		x			x	x
Selection assistance	x		x		x	x	x	x		x		
Automated selection										x	x	
Route creation			x		x			x	x	x	x	x
Integration of selection and route creation											x	
Multiple days												
Public transport												

Dynamic Tour Guide (DTG) (Kramer et al., 2007) is the most advanced and mature PET found in the literature. The system allows strong personalisation based on ontologies. Moreover, it creates routes in real time (less than five seconds) grouping attractions into Tourist Building Blocks (TBB). However, the algorithm applied in DTG for routing creation is very simple and it has important restrictions. It can only create proper solutions for a small number of TBBs and one day routes. No public transportation is taken into account.

PNS (Maruyama et al., 2004) applies a genetic algorithm to calculate routes. Nevertheless, tourists have to manually enter the attractions they want to visit with their details (visiting time, duration and value). The system needs nearly ten seconds to obtain a route for just twelve attractions.

2.2 Routing algorithms

The routing algorithm presented in this paper is based on extensions of the Orienteering Problem (OP) (Tsiligirides, 1984). In the OP, several locations with an associated score have to be visited in order to obtain a total trip score. Each player can visit each attraction only once. The objective is to obtain a total trip score as high as possible without violating a time restriction. Its generalization to multiple players is known as the Team Orienteering Problem (TOP). When locations have an associated time window, the problem is called TOP with Time Windows (TOPTW) (Savelsbergh, 1985). The Multi Constrained TOPTW (MCTOPTW) extends TOPTW adding multiple constraints such as money budget or travelled distance.

The Time Dependent OP (TDO) (Fomin & Lingas, 2002) is an extension of the OP where the time needed to travel from a location *i* to a location *j* depends on the leaving time from location *i*. Finally, the proposed algorithm Multi Path TOPTW (MPTOPTW) extends the previous algorithms including multiple paths to go from one location to another. The duration of at least one of these paths is dependent on the departure time. MPTOPTW is valid to simulate public transportation.

To the authors' understanding, there is no algorithm able to solve MPTOPTW out of the extensions of OP. In the field of public transport, an algorithm to calculate optimum itineraries in an urban public transport system has been proposed, including a stop at an intermediate point (Zografos and Androutsopoulos, 2008). In Hong-Kong, a similar system was tested using a different approach (Chiu, Lee and Leung, 2005). Both of them were designed to be used as standalone solutions. Thus, they are not suitable to be used within a wider algorithm required to solve the MPTOPTW, which has to calculate several times the best way to move from one place to another.

3 Objectives of the system

The objective of the system is the improvement of the route creation functionality for PETs using the latest advances in OR. The system goes beyond the state of the art in route creation functionality, as none of the existing PETs apply the latest OR algorithms to solve similar problems. Moreover, although the inclusion of public transportation has been identified as functionality with a great added value for tourists, this has not been achieved in current PETs.

The intelligent route creation system proposed in this paper calculates routes in real time (less than 5 seconds) according to the profile and restrictions of tourists (time, money, travelled distance …), weather conditions and public transportation. Routes are updated in real time when necessary (longer visit to an attraction, weather change, missing a bus). The system will be validated using the latest technological developments, both on client devices and web technologies.

It must be mentioned that the work presented in this paper focuses on the route creation functionality. The creation of tourists' profile is kept as simple as possible due to great advances on this area such as DTG Planner (Höpken et al., 2006).

3.1 Application scenario: San Sebastian

San Sebastian is a beautiful city located at the North of Spain, just 20 kilometers away from France. San Sebastian's picturesque coastline makes it a popular beach resort, being one of the most relevant tourist places of the North of Spain. The villages near the city are easily reached and they offer amazing cultural, gastronomic and architectural experiences. Due to the small size of the Basque Country (less than 200 km long in any direction), the system assumes that tourists do not change their accommodation during the days of their stay. Thus, the starting and ending location for each day are the accommodation of the tourist. The maximum tour length is 3 days, leading to an actual average stay of a tourist in San Sebastian of 2 days.

Fig. 1. Overview of the public transportation network of San Sebastian

San Sebastian has around 200.000 habitants and it is best visited combining public transportation with short walks. The public transportation (see Figure 1) offers a good bus service with a fixed timetable through a dense network.

3.2 Use cases

As it has been mentioned before, the system targets tourists that are in the destination. Evaluations of existing PETs (Kramer et al., 2007) showed that although tourists are accepting these systems, they also enjoy ignoring the plan. In this case, they use the system to access information about the destination while they explore it. The proposed system offers tourist both choices with the following options:

- Creation and edition of the tourist profile, restrictions, attractions to be visited, and the location of the accommodation.
- Creation and adaptation of the route anytime. The system will update the route when new events are detected, after confirmation by the tourist.
- Visualization and exploration of the route, viewing and evaluating the attractions in a GoogleMaps-based interface.
- Free browse of the destination, accessing information about attractions and public transportation.

4 System overview

4.1 System architecture

The system is based on a thin client architecture (see Figure 2). There are three differentiated elements: client, agents and server. The client provides the interface (see Figure 3) for the final user and communicates the position of the tourist to the server. The HTML based client has been developed using Google Web Toolkit (GWT) and Google Gear. GWT allows developing web applications using native Java code, automatically transforming this code into HTML and JavaScript. Thus, the developer is isolated from browser specific issues. New versions of GWT make the existing code compatible with new browser versions. Furthermore, Google Gears is an Open Source project that enables more powerful web applications. Gears allows among others, to access the location data of the client. The interface is based on Google Maps to display geo-referenced information.

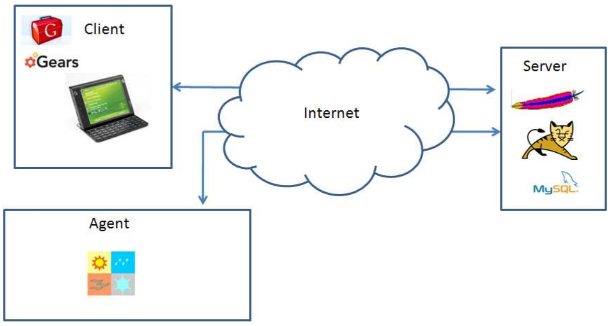

Fig. 2. Architecture of the system

Agents are responsible of detecting changes in the context and communicating them to the system. Each agent tracks a specific variable (weather, traffic jams ...) and updates its state on the server. The current prototype has only one agent that periodically queries a public weather Web Service to check for changes in the weather. The agent looks for the forecast within three days, so that weather is taken into account when creating routes. Future versions will tackle more aspects of the context such as the state of the traffic, demonstrations, long queues at attractions,...

Finally, the server collects all the information sent by the client and the agent. It combines this information with the rest of the data to create and update personalised tourist routes in real time. The server side is based on an Apache web server with a Apache Tomcat application server and a Mysql database manager. The prototype has been tested on a HTC X7500 Advance device, which has built in WiFi, 3G and GPS functionalities.

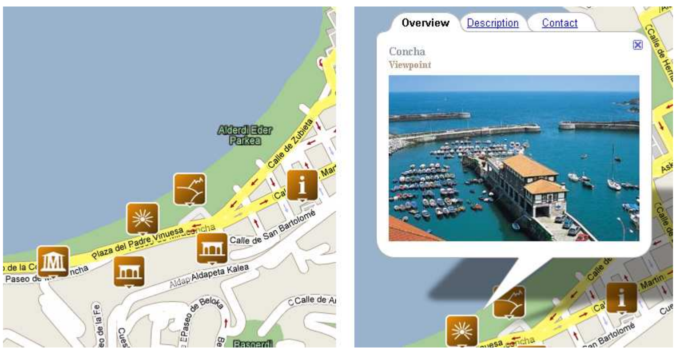

Fig. 3. Screenshots of the client

4.2 Data model

All the information required by the system has been modelled to be stored on a database. The main entities the system stores are the attractions, public transportation, the context and the routes. LTOs are responsible for updating this information, which is a crucial factor to create high quality routes. In order to help in the management of this information, a simple authoring tool based on questionnaires has been developed.

4.2.1 Attractions

Attractions are composed by places to visit (museums, buildings ...) and activities to do. Different types of data are stored for each attraction:

- Descriptive information: name, description, image, location, opening hours, prices and a link to a web page with additional information.
- Customization information to assign a different value to each tourist profile. Each attraction has a value between 0 and 100 for each profile. Some attractions can be clustered into groups, so that each group includes similar attractions, for example, museums of a similar topic and layout. Each time a tourist visits an attraction, the system records the visit and asks for feedback to compare the expected value to the real value. Tourists can rank visited attractions within four categories: dislike a lot (0-25), dislike (26-50), like (51-75), or like a lot (76-100). When a tourist expresses he/she has liked or disliked an attraction of a group, the value of the rest of the attractions of the group is increased or decreased multiplying it by a different percentage: dislike a lot (50%), dislike (75%), like (125%), like a lot (150%). If a tourist has explicitly marked he wants to visit or omit an attraction, its value is changed to 100 or 0 automatically.
- Adaptation information to change the default value of the attraction to react to the context. The relation between each context condition and the attraction value is represented by a percentage. For each attraction, the context conditions that are taken into account are the weather (sun/rain), and the hour of the day (dawn, morning, midday, evening, nightfall, night). For example, a nice viewpoint will be much more appealing on a sunny dawn than on a rainy afternoon.

4.2.2 Public transportation

Public transportation has been modelled taking into account that service frequencies vary according to the day of the week (working day, Saturday, Sunday, special days) and that there are different time windows within a day, each one with different frequencies and different travelling times.

The database stores the information in a human friendly way, keeping it simple to update and to fix errors. At night, a batch procedure is executed to transform this information into a more complex format specifying the travel times between attractions for the next three days. The algorithm accesses the transformed data to improve its execution speed creating new routes.

4.2.3 Context

Context stores information about the weather forecast for the following three days. The forecast is obtained from Yahoo Weather public Web Service. The forty eight weather condition codes reported by Yahoo are mapped into two simplified states: rainy (freezing rain, showers, snow ...) and sunny (clear, fair, sunny ...).

4.2.4 Routes

Each route is stored in the database, including the attractions visited each day and the time required to go from one attraction to the next one. Details about each movement are saved (detailed travelling times, origin, and destination).

As routes adapt in real time to the behaviour of tourists and changes in weather conditions, they must be recalculated while tourists are on the destination. When a new route is calculated, the previous one is taken as the basis template from the starting time of the visit till the current time. Then, the system creates a new route starting from the current position and taking into account that some attractions have been visited and that some resources (time and money) have been consumed. Tourists can customize the proposed route inserting, moving or deleting attractions.

4.3 Multi Path Team Orienteering Problem with Time Windows

The planning problem that needs to be solved is modelled as a Multi Path Team Orienteering Problem with Time Windows (MPTOPTW). A MPTOPTW requires the following input data: a list of attractions (ID, position, customized value for each day, opening and visiting times, cost); information about the public transportation network; preferences of the tourist; the current position of the tourist; and the available resources (time and money). Based on this information, the algorithm creates a personalised route with an ordered list of attractions to visit, including the details to move from one attraction to the following one.

The algorithm to solve the MPTOPTW applies Iterated Local Search (ILS) (Lourençco, Martin and Stützle 2002). ILS is a metaheuristic method based on iteratively building sequences of solutions generated by an embedded heuristic called local search. This leads to much better solutions than repeating random trials of the same heuristic. The solution found by the local search is perturbed to create a new solution. Then, the best solution is taken as the new starting solution for the local search. The process is repeated until a termination criteria is met. The ILS metaheuristic can be summarized as in Figure 4.

```
s0 = GenerateInitialSolution
s* = LocalSearch(s0)
while termination condition not met do
|      s' = Perturbation(s*)
|      s*' = LocalSearch(s')
|      s* = AcceptanceCriterion(s*, s*')
end
```

Fig 4. Diagram of ILS

The implemented local search heuristic is based on an Insert Step that tries to add new visits to a route, one by one (Vansteenwegen et al., 2008), using two main concepts. The first one is *Wait*, the time a tourist has to wait for an attraction to be opened. The second one is *MaxShift*, which represents the maximum delay in the arrival to an attraction without causing a route alteration. For each attraction *i*, *MaxShift* is calculated as the sum of *Wait* and *MaxShift* of the next location *i+1*, unless its time windows ends earlier. For a feasible insertion of a new attraction *j* between visits *i* and *k*, besides not violating any of the constraints, the total time consumption of visit *j*, named $shift_j$, should be limited to the sum of *Wait* and *MaxShift* of visit *k*. Thanks to *MaxShift* and *Wait*, it is not required to check the time windows of all other visits in a tour, to determine the feasibility of a given local search move.

For each visit that can be inserted, the smallest insertion time (*shift*) is determined. For each of these visits a ratio, which ponders the value of the attraction with the cost (time, money and distance) required to visit it, is calculated. Among them, the one with the highest ratio is selected for insertion. Then, the Insert Step is repeated until no attraction can be inserted.

After an insertion, all other visits should be updated. Visits after the insertion require an update of the waiting time (*Wait*), the arrival time, the start of the service, the leave time and *MaxShift*. Every time a visit requires a waiting time, the time shift for the end of that service and all following services is reduced by this waiting time. Visits before the insertion may require an update of *MaxShift*.

To calculate the cost of going from one attraction to the next one, the algorithm chooses the fastest mean between public transportation and going on foot. The walking time only depends on the position of the attractions. However, when using the public transportation, there is a walking time to and from the nearest stops, a waiting time till the transport arrives and a traveling time (including possible transfers). Knowing the leaving time from the departure attraction and the details of the public transportation network (starting time of the services, frequencies, locations of the stops, traveling times between stops), it is possible to calculate all the traveling times and to choose the best option.

The perturbation phase (Figure 4) is based on a shake movement that removes consecutive attractions from a tour. After the removal, all visits following the removed visits are shifted forward as much as possible, in order to avoid unnecessary waiting. A list containing attractions removed recently avoids removing a previously removed visit without first trying to find a different visit.

Although it is possible to include advanced acceptance functions which use the search history to decide on the best solution, in this case the new solution obtained by the shake movement is always accepted (Acceptance Criterion in Figure. 4). The heuristic always continues the search from the perturbed solution, it never returns to the best found solution to continue. This is called iterated local search with a random walk acceptance criterion (Lourençoet al. 2003). Of course, the best found solution is always kept on memory. Once the termination criteria is met (maximum number of iterations without improvement or maximum allowed time), the system returns the best solution found.

4.4 Example

Table 2 shows a simplified example of a tour proposal for a sunny morning. For each attraction the travel times are summarised. The waiting time refers to the waiting time at the bus stop. When public transportation is required, detailed information is available as follows: i) walk 3 minutes until you reach the bus stop at Sancho El Sabio street number 1, ii) wait for 5 minutes and take bus 27 for 5 stops (10 minutes) until you reach the stop at Avda Zumalakarregi number 3, iii) walk 7 minutes until you arrive to the destination.

Table 2. Route for a sunny morning

From	To	Travel time				Visit Duration	Score	Money
		total	walking	waiting	bus			
Hotel Amara	Peine Vientos	25	10	5	10	30	90	0
Peine Vientos	Concha	20	20	0	0	75	85	30
Concha	Urgul	15	15	0	0	90	85	0
Urgul	Hotel Amara	20	5	5	10	0	0	0

If suddenly the weather changes and it starts raining, the value of some attractions changes, mostly the outdoor ones. The weather agent notifies this change and the server recalculates the route taking into account the new values. In the previous examples the value of Peine de los Vientos and Urgul (outdoor locations) would decrease to 40 and some other attractions would be more attractive. Table 3a shows the composition of the new route.

Table 3. New route for a rainy morning (a) and extended visit (b)

From	To	Travel time	Visit Duration	Score	Money
Hotel Amara	Concha	25	75	85	60
Concha	Kursaal	30	75	75	30
Kursaal	San Telmo	15	90	75	0
San Telmo	Hotel Amara	20	0	0	0

(a)

From	To	Travel time	Visit Duration	Score	Money
Hotel Amara	Concha	25	75	85	60
Concha	Kursaal	30	135	75	30
Kursaal	San Vicente	15	10	55	0
San Vicente	Hotel Amara	20	0	0	0

(b)

While following the proposed route, if the tourist expends more time than expected in an attraction, the route is recalculated again. On the previous example, if the tourist leaves Kursaal later than expected, the previous route would become infeasible. Thus a new route would include a shorter visit before returning to the hotel. Table 3b shows this route.

5 Evaluation

Although the real final evaluation will be held at the LTO of San Sebastian during the summer of 2009 offering the system to real tourists, the prototype has been tested on the laboratory. All computations have been carried out on a personal computer Intel Core 2 Quad with 2.40GHz processors and 2 GB Ram.

First a version of the algorithm not including public transportation has been tested. It has been compared against the exact method proposed by Boussier et al. (2006) for the Selective Vehicle Routing Problem with Time Windows (SVRPTW). Problems with 50 and 100 points and up to 10 routes have been solved. The average gap with the optimal solution is around 5% and the average execution time is around 2.5 seconds. The total time for the whole test set (90 problems) is 150 seconds, while it took around two hours to Boussier's algorithm and they were not able to solve all the problems with their exact solution.

Moreover, new custom test sets have been designed based on the available test sets for the TOPTW. This test set is composed by 68 problems of up to 288 locations; 1 and 2 constraints; and 1 and 2 routes. For 68 problems with one route and two extra constraints the average gap with the best results is only 2.2%, and the worst gap is 8.9%, and with the upper bound 3.1% and 12.3 %. The average computation time is 0.1 seconds. Adding one route the gaps are 0.8% and 6.8% with the best results, and 11.8% and 25.30% with the theoretical upper bound. The average execution time is 0.4 seconds. Authors expect the detailed results of both tests to be published on 2009.

Regarding the inclusion of the public transportation, problems initialised with data about the main bus lines and attractions of San Sebastian are solved in less than 1 second. Authors are working on extending the custom test set to include public transportation in a general way. Once this final custom test set is finished, other researchers are going to be able to develop and test new approaches for MPTOPTW.

6 Conclusions and future work

Current devices meet the technical requirements of a PET, such as combining computational power, location and communication capabilities. Although some PETs are already available, none of them is able to create personalised routes automatically in real time, on a medium or big destination, including public transportation.

This paper presents an intelligent routing system for PETs based on a novel algorithm that is able to create personalised routes in real time for large destinations taking into account public transportation. Moreover, it reacts to the actions and behaviors of tourists, events and weather changes. The PET is based on a thin client architecture and it takes advantage of the latest technological advances. The evaluation of the system will be held during the summer of 2009 at the city of San Sebastian.

Future work targets three areas. The first one is the implementation of the system on a larger destination. The second one focuses on increasing the algorithm's speed, applying parallelisation techniques, and its functionality: supporting scenic routes/walks, different lodgings and supporting groups of tourists traveling together. The final one is the improvement of the social aspects of the system, storing, sharing and adding travel experiences to better help tourists on the destination.

References

Beer, T., Fuchs, M., Höpken, W., Werthner, H., & Rasinger, J. (2007). CAIPS: A Context-Aware Information Push Service in Tourism. *Information and Communication Technologies in Tourism 2007*.

Brown, B., & Chalmers, M. (2003). Tourism and mobile technology. *8th European Conference on Computer Supported Cooperative Work, Helsinki, Finland.*

Chiu, D. K. W., Lee, O., & Leung, H. F. (2005). A Multi-Modal Agent Based Mobile Route Advisory System for Public Transport Network. *38th Annual Hawaii International Conference on System Sciences (HICSS'05), Hawaii.*

Fomin, F. V., & Lingas, A. (2002). Approximation algorithms for time-dependent orienteering, *Information Processing Letters* (Vol. 83, pp. 57-62): Elsevier.

Höpken, W., Fuchs, M., Zanker, M., Beer, T., Eybl, A., Flores, S., et al. (2006). etPlanner: An IT Framework for Comprehensive and Integrative Travel Guidance. *Information and Communication Technologies in Tourism* 2006.

Kramer, R., Modsching, M., ten Hagen, K., & Gretzel, U. (2007). Behavioural Impacts of Mobile Tour Guides. *Information and Communication Technologies in Tourism 2007.*

Lourenco, H.R., Martin O., and Stuetzle T.. *Iterated Local Search*. In: F. Glover and G. Kochenberger, editors, Handbook of Metaheuristics. pages 321-353, Kluwer Academic Publishers, Norwell, MA, 2002

Maruyama, A., Shibata, N., Murata, Y., Yasumoto, K., & Ito, M. (2004). A Personal Tourist Navigation System to Support Travelling Multiple Destinations with Time Restrictions. *18th International Conference on Advanced Information Networking and Applications*

Savelsbergh, M. W. P. (1985). Local search in routing problems with time windows. *Annals of Operations Research, 4*(1), 285-305.

Schmidt-Belz, B., Laamanen, H., Poslad, S., & Zipf, A. (2003). Location-based mobile tourist services-first user experiences. *Information and Communication Technologies in Tourism 2003.*

Souffriau, W., Vansteenwegen, P., Vertommen, J., Vanden Berghe, G., & Van Oudheusden, D. (2008) A Personalised Tourist Trip Design Algorithm for Mobile Tourist Guides. *Applied Artificial Intelligence*: to appear.

Stroobants, R. (2006). Mobile Tourist Guides. *Katholieke Universiteit Leuven, Leuven.*

Tsiligirides, T. (1984). Heuristic methods applied to orienteering. *The Journal of the Operational Research Society, 35*(9), 797-809.

Vansteenwegen, P., and Van Oudheusden, D. (2007) The Mobile Tourist Guide: An OR Opportunity. *OR Insights,* 20 (3), 21-27.

Vansteenwegen, P., Souffriau, W., Vanden Berghe, G., and Van Oudheusden, D. (2008) Metaheuristics for tourist trip planning. *Metaheuristics in the Service Industry* (Geiger, M., Habenicht, W., Sevaux, M., Sörensen, K., eds.), Lecture Notes in Economics and Mathematical Systems, Springer Verlag: to appear.

Vansteenwegen, P. (2008) *Planning in tourism and public transportation.* PhD dissertation, Katholieke Universiteit Leuven, Centre for Industrial Management, Belgium, ISBN: 978-90-5682-949-0.

Zografos, K. G., & Androutsopoulos, K. N. (2008). *Algorithms for Determining Optimum Itineraries in a Multimodal Urban Transportation Network.* IEEE Transactions on Intelligent Transportation Systems, Tokyo, Japan

Acknowledgements

The authors would like to thank the Basque Government for partially funding this work through the neurebide and etourgune projects and to the Centre for Industrial Management of the Katholieke Universiteit Leuven for hosting Ander Garcia as a guest researcher during 2008. Pieter Vansteenwegen is a post doctoral research fellow of the “Fonds Wetenschappelijk Onderzoek- Vlaanderen (FWO)”.