Project Number 288094

eCOMPASS

eCO-friendly urban Multi-modal route PlAnning Services for mobile uSers

STREP
Funded by EC, INFSO-G4(ICT for Transport) under FP7

eCOMPASS – TR – 010

A Survey on Algorithmic Approaches for Solving Tourist Trip Design Problems

Damianos Gavalas, Charalampos Konstantopoulos, Konstantinos Mastakas,
Grammati Pantziou, Yiannis Tasoulas

October 2012

Project Number 288094

eCOMPASS

eCO-friendly urban Multi-modal route PlAnning Services for mobile uSers

STREP
Funded by EC, INFSO-G4(ICT for Transport) under FP7

eCOMPASS – TR – 010

A Survey on Algorithmic Approaches for Solving Tourist Trip Design Problems

Damianos Gavalas, Charalampos Konstantopoulos, Konstantinos Mastakas, Grammati Pantziou, Yiannis Tasoulas

October 2012

A Survey on Algorithmic Approaches for Solving Tourist Trip Design Problems

Damianos Gavalas[1], Charalambos Konstantopoulos[2], Konstantinos Mastakas[3], Grammati Pantziou[4], and Yiannis Tasoulas[2]

[1]Department of Cultural Technology and Communication, University of the Aegean, Mytilene, Greece, email: dgavalas@aegean.gr
[2]Department of Informatics, University of Piraeus, Piraeus, Greece, email: {konstant,jtas}@unipi.gr
[3]Department of Mathematics, University of Athens, Athens, Greece, email: kmast@math.uoa.gr
[4]Department of Informatics, Technological Educational Institution of Athens, Athens, Greece, email: pantziou@teiath.gr

Abstract

The tourist trip design problem (TTDP) refers to a route-planning problem for tourists interested in visiting multiple points of interest (POIs). TTDP solvers derive daily tourist tours i.e., ordered visits to POIs, which respect tourist constraints and POIs attributes. The main objective of the problem discussed is to select POIs that match tourist preferences, thereby maximizing tourist satisfaction, while taking into account a multitude of parameters and constraints (e.g., distances among POIs, visiting time required for each POI, POIs visiting days/hours, entrance fees, weather conditions) and respecting the time available for sightseeing in daily basis. The aim of this work is to survey models, algorithmic approaches and methodologies concerning tourist trip design problems. Recent approaches are examined, focusing on problem models that best capture a multitude of realistic POIs attrbutes and user constraints; further, several interesting TTDP variants are investigated. Open issues and promising prospects in tourist trip planning research are also discussed.

1 Introduction

Tourists that visit a destination for one or several days, are facing the problem to decide which points of interest (POIs) would be more interesting to visit and to determine a route for each trip day, i.e., which POIs to visit as well as visit order among them. This is a challenging quest that involves a number of constraints such as the visiting time required for each POI, the POI's visiting days/hours, the travelling distance among POIs, the time available for sightseeing in daily basis and the "degree of satisfaction" (termed "profit") associated with the visit to each POI (based on personal profile and peferences). A number of different problems may be defined by considering different parameters and constraints of the above general problem, termed as the "tourist trip design problem" (TTDP) [114].

Mobile tourist guides may be used as tools to derive solutions to TTDP [36], [68], [67], [80]. Based on a list of personal interests and preferences, up-to-date information for the sight and information about the visit (e.g. date of arrival and departure, accommodation address, etc), a mobile guide can suggest near-optimal and feasible routes that include visits to a series of sights, and to recommend the order of each sight's visit along the route [114].

A number of web and mobile applications have recently incorporated tourist route recommendations within their core functionality [113], [2], [53]. In effect, most are TTDP solvers (e.g. the City Trip Planner [1], the mtrip [2]) taking into account several user-defined parameters within their recommendation logic (days of visit, preferences upon POI categories, start/end location, visiting pace/intensity), while also allowing the user to manually edit the derived routes, e.g. add/remove POIs. Recommended tours are visualized on maps [1], [2], [53], allowing users to browse informative content on selected POIs. Some tools also offer augmented reality views of recommended attractions (e.g., [2]).

The modeling of a TTDP is approached considering the following input data (see Figure 1):

- A set of candidate POIs, each associated with a number of attributes (e.g. type, location, popularity, opening days/hours, etc).
- The travel time among POIs calculated using multi-modal routing information among POIs, i.e. tourists are assumed to use all modes of transport available at the tourist destination, including public transportation, walking and bicycle.
- The "profit" of each POI, calculated as a weighted function of the objective and subjective importance of each POI (subjectivity refers to the users' individual preferences and interests on specific POI categories).
- The number of routes that must be generated, based upon the period of stay of the user at the tourist destination.
- The anticipated duration of visit of a user at a POI which derives from the average duration and the user's potential interest for that particular POI.
- The daily time limit T that a tourist wishes to spend on visiting sights; the overall daily route duration (i.e. the sum of visiting times plus the overall time spent moving from a POI to another which is a function of the topological distance) should be kept below T.

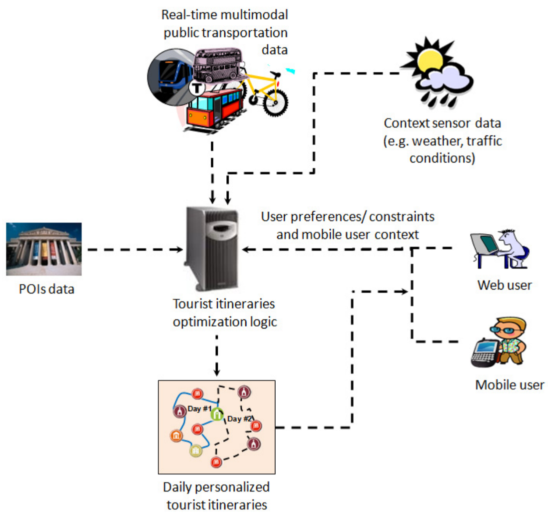


Figure 1: Input data and recommended itineraries in TTDP.

By solving a TTDP we expect to derive daily, ordered visits to POIs, while respecting user constraints and POIs attributes. High quality TTDP solutions should feature POI recommendations that match tourist preferences and near-optimal feasible route scheduling. The algorithmic and operational research literature include many route planning problem modeling approaches, which may be used for different versions of TTDP. A well-known optimization problem that may formulate a simple version of TTDP is the orienteering problem (OP) [105]. The OP is based on the orienteering game, in which several locations with an associated profit have to be visited within a given time limit. Each location may be visited only once, while the aim is to maximize the overall profit collected on a single tour. Clearly, the OP may be used to model the simplest version of the TTDP wherein the POIs are associated with a profit (i.e. user satisfaction) and the goal is to find a single tour that maximizes the profit collected within a given time budget (time allowed for sightseeing in a single day).

Extensions of the OP have been successfully applied to model more complicated versions of the TTDP. The team orienteering problem (TOP) [29] extends the OP by considering multiple tours (i.e. daily tourist itineraries). The TOP with time windows (TOPTW) considers visits to locations within a predefined time window (this allows modeling opening and closing hours of POIs). The time-dependent TOPTW (TDTOPTW) considers time dependency in the estimation of time required to move from one location to another and therefore, it is suitable for modeling multi-modal transports among POIs. Several further generalizations exist that allow the modeling of even more complicated versions of the TTDP, e.g. the multi-constraint team orienteering problem with time windows (MC-TOPTW) takes into account multiple user constraints such as the overall budget that may

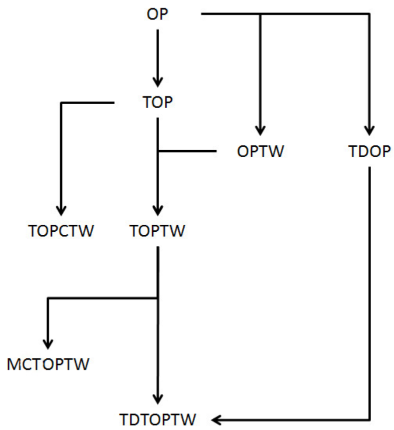


Figure 2: Optimization problems relevant to the TTDP (arrows denote problem extensions/generalizations).

be spent for POI entrance fees. A non-exhaustive illustration of the optimization problems with relevance to the TTDP as referred to in the literature is given in Figure 2.

In this article we survey exact, approximate and heuristic approaches for solving the TTDP and interesting variants of the TTDP. Section 2 and 3 present algorithmic techniques for solving optimization problems that are employed for modeling different versions of the TTDP. Specifically, Section 3 surveys algorithmic approaches for solving single tour versions of the TTDP i.e. problems aiming at finding a single tour that maximizes the profit under certain constraints (OP and OPTW), and Section 3 surveys algorithmic approaches dealing with multiple tour versions of the TTDP (TOP, TOPTW, TDTOPTW). It is noted that particular emphasis is given to algorithmic techniques for solving problems highly relevant to more complicated and realistic versions of the TTDP (e.g. TOPTW and TDTOPTW). Section 4 highlights combinatorial problems that may be used for modeling variants of the TTDP and surveys algorithmic approaches dealing with such problems. Finally, Section 5 concludes the paper providing new prospects in tourist route planning research. Specifically, we discuss (i) quality improvements upon existing solution approaches, (ii) modeling TOPTW generalizations, (iii) modeling problems relevant to TTDP and (iv) employing parallel computing techniques to design new heuristics for the TTDP.

2 Single tour TTDP solution approaches

2.1 Orienteering Problem (OP)

The Orienteering Problem (OP) was introduced by Tsiligirides [105] named after a sport game called orienteering. Other names used for OP are Selective Traveling Salesperson

Problem (STSP) [74], Maximum Collection Problem (MCP) [64] and Bank Robber Problem [12]. OP can be formulated as follows: Let $G = (V, E)$ be an edge-weighted graph with profits (rewards or scores) on its nodes. Given a starting node s, a terminal node t and a positive time limit (budget) B, the goal is to find a path from s to t (or tour if $s = t$) with length at most B such that the total profit of the visited nodes is maximized (see Figure 3).

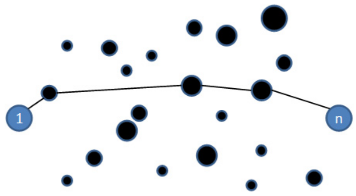


Figure 3: OP illustration. Circles' radius denote nodes' profit.

OP can be formulated as an integer programming problem as follows [109]: Let N be the number of nodes labelled by $1, 2, \ldots, N$ where $s = 1$ and $t = N$, p_i be the profit of visiting node i and c_{ij} be cost of traveling from i to j. For every path from 1 to N, if node i is followed by node j we set the variable x_{ij} equal to 1 or equal to 0 otherwise. Finally, u_i denotes the place of node i in the path. With this notation we have the following relations:

$$\max \sum_{i=2}^{N-1} \sum_{j=2}^{N} p_i x_{ij}, \tag{1}$$

s.t.

$$\sum_{j=2}^{N} x_{1j} = \sum_{i=1}^{N-1} x_{iN} = 1, \tag{2}$$

$$\sum_{i=1}^{N-1} x_{ir} = \sum_{j=2}^{N} x_{rj} \leq 1, \text{ for all } r = 2, \ldots, N-1, \tag{3}$$

$$\sum_{i=1}^{N-1} \sum_{j=2}^{N} c_{ij} x_{ij} \leq B, \tag{4}$$

$$2 \leq u_i \leq N, \text{ for all } i = 1, 2, \ldots, N, \tag{5}$$

$$u_i - u_j + 1 \leq (N-1)(1 - x_{ij}), \text{ for all } i, j = 2, \ldots, N, \tag{6}$$

$$x_{ij} \in \{0, 1\}, \text{ for all } i, j = 1, \ldots, N. \tag{7}$$

The objective function (1) is to maximize the total profit of visited nodes. Constraint (2) ensures that the path starts at node 1 and ends at node N. Constraint (3) ensures that the path starting at node 1 and ending at node N is connected and each node is visited at most once. Constraint (4) ensures that the path meets the time budget. Finally, constraints (5) and (6) ensure that there are no closed subtours.

The most important types of OP considered so far depend on whether the graph is undirected or directed (undirected OP or directed OP), whether the nodes have different or the same profits (weighted OP or unweighted OP), whether there is no fixed terminal node but only a fixed starting node called root (rooted OP) or whether there are not fixed end points at all (unrooted OP) and their combinations. OP is harder than rooted OP, which in turn is harder than unrooted OP, since algorithms for rooted OP can be used to solve unrooted OP by considering each node of the graph as the root. Likewise, OP can be used to solve rooted OP by considering as the starting node the root and each node of the graph as the finishing node.

OP is NP-hard (e.g. see [58], [74]). Hence, exact solutions for OP are only feasible for graphs with a small number of nodes. Some of the exact algorithms proposed for the OP are based on branch-and-bound [74, 88] and branch-and-cut [54, 48]. There exist a number of approximation algorithms for the above variants of OP, however, with high complexity. Note that rooted OP is APX-hard (e.g. see [22], where it is proved that rooted OP is NP-hard to approximate to within a factor of $\frac{1481}{1480}$).

Some helpful remarks concerning the approximability of certain OP variants are the following:

- In the approximation algorithms for the OP, the input graph can be restricted to graphs having nodes with unit profit since Korula [69, Lemma 2.6] proved that an a-approximation algorithm for OP with unit profits yields an $a(1+O(1))$-approximation algorithm for weighted OP. The basic idea is to use a standard scaling technique to adjust the weights into integers from 1 to n^2, where n is the number of nodes, and then to transform the graph to a new graph with at most n^3 nodes having unit profits. A solution with the above approximation is derived for the weighted OP by applying an a-approximation algorithm on the newly transformed graph.

- An approach for approximating the unrooted OP in undirected graphs comes from approximation algorithms for the k-TSP problem (find a tour of minimal length while visiting at least k nodes). The basic idea is to break such a tour into pieces bounded by B and then pick the one with the largest profit (for more details, see [16]).

- Usually, the approximation algorithms for OP have highest complexity in directed graphs than in undirected graphs (e.g. see [31]).

One of the first works for approximating the rooted OP is that of Arkin et al. [12] that gives an $(2 + \epsilon)$-approximation algorithm for OP restricted to points in the 2-dimensional plane. The fundamental idea to approximate the rooted OP in undirected graphs was presented by Blum et al. in [21], [22]. They use, as an intermediate step, the solution of the min-excess $(s-t)$ path problem (find a minimum-excess [1] path connecting fixed nodes s and t that visits at least k nodes or collecting at least k profit). The basic idea is to guess[2] the profit P_{OPT} of the optimal solution of the rooted OP and try to compute for every node the min-excess path from the root to the node that collects at least a fixed fraction of P_{OPT}, until a path is found with length at most B. In this work they obtain a 4-approximation algorithm for rooted OP in undirected graphs by using a $(2+\epsilon)$-approximation to the min-excess $(s-t)$ path problem. In fact, most subsequent approximation algorithms (e.g. see [31]) use the solution of a min-excess path problem as an intermediate step.

Later, Bansal et al. [18] give a 3-approximation algorithm for OP in metric spaces. In their approach they show that a $(2+\epsilon)$-approximation to the min-excess $(s-t)$ path problem can be used to obtain a 3-approximation for OP, hence, improving the previous result by Blum et al. [21], [22].

Chen et al. [34] present a PTAS for the rooted OP in $\mathbb{R}^d$, where every location has unit profit. In order to create the PTAS, an approximation algorithm is presented for the $k-$TSP in $\mathbb{R}^d$ based on Mitchell's approximation algorithm for the $k-$TSP [81] and Arora's work on the same problem[13].

Chekuri and Pal [33] give an $O(\log n)$-approximation algorithm for solving the OP in directed graphs that runs in quasi-polynomial time. In their formulation of OP, called submodular OP, the total profit of the nodes visited is not necessarily the sum of the profit of each node but has the submodular property, i.e., for subsets A, B of the set of nodes the total weight f satisfies the inequality: $f(A \cup B) \leq f(A) + f(B) - f(A \cap B)$.

Chekuri et al. [31] give approximation algorithms for the OP in directed and undirected graphs. In particular, they give a $(2+\epsilon)$-approximation algorithm for the undirected OP with running time $n^{O(1/\epsilon)}$ and an $O(\log^2 OPT)$ approximation algorithm for directed OP, where OPT denotes the number of nodes in an optimal solution. They follow Blum et al. focusing on the k-stroll problem (i.e. find a minimum length $s-t$ path that visits at least k nodes) and give bi-criteria approximations for k-stroll in directed and undirected graphs with respect to the path length and the number of nodes visited.

Nagarajan and Ravi [84] give an $O(\frac{\log^2 n}{\log\log n})$-approximation algorithm for OP in directed graphs, by approximately solving a number of problems in the following order: from minimum ratio ATSP to directed k-path problem, then to the minimum excess problem and finally to OP in directed graphs. First, they present a polynomial time $O(\frac{\log^2 n}{\log\log n})$ bi-criteria approximation algorithm for the directed k-TSP problem (find a minimum length tour that contains a specified root and at least another k nodes), by using an $O(\frac{\log^2 n}{\log\log n})$-approximation algorithm for minimum ratio ATSP problem, due to Asadpour et al [15]. They reduce the directed k-path problem to the directed OP More specifically, they go through from directed k-path problem to directed minimum excess problem and finally to OP in directed graphs.

[1] excess of an $s-t$ path is the difference of the path length from the shortest $s-t$ path.

[2] i.e., try exhaustive search

Table 1 summarizes the approximation algorithms for the OP in directed and undirected graphs and their approximation ratio.

Table 1: Approximation algorithms for the OP

Reference	Directed OP	Undirected OP	Approximation Ratio	Time
Blum et al. [21]	✓		4	polynomial
Bansal et al. [18]	✓		3	polynomial
Chekuri et al. [31]	✓		$(2+\epsilon)$	polynomial
Chekuri and Pal [33]		✓	$O(\log n)$	quasi-polynomial
Chekuri et al. [31]		✓	$O(\log^2 OPT)$	polynomial
Nagarajan & Ravi [84]		✓	$O(\frac{\log^2 n}{\log\log n})$	polynomial

For practical applications, many researchers propose heuristics to tackle the OP, based on different approaches. Some representative methods are discussed in the sequel. Tsiligirides [105] presents two algorithms for OP. A stochastic algorithm based on Monte-Carlo techniques that constructs a large number of routes and picks the one with the maximum profit and a deterministic heuristic algorithm, that partitions the geographic area into concentric circles and restricts the allowed routes into the sectors defined by the circles.

In [58] a center-of-gravity heuristic for the OP is presented where the solution tour is constructed by the cheapest insertion procedure according to a combined measure for node selection. Golden et al. in [57] improve the center-of-gravity heuristic by rewarding nodes associated with above-average tours while penalizing those associated with below-average tours.

In [87] Ramesh et al. propose a four-phase heuristic. After choosing the best solution from iterations over a set of three phases (node insertion, edge exchange and node deletion), a fourth phase is entered, where one attempts to insert unvisited nodes into the tour.

In [116] the authors apply a neural network approach to solve the OP. They derive an energy function and learning algorithm for a modified, continuous Hopfield neural network.

Chao et al. [28] propose a heuristic algorithm for OP that proceeds as follows. Initially, the set of nodes is partitioned in a greedy way into paths each with length bounded by B and the current solution is the path with the most profit. Then an iterative method is employed. At each iteration a local search procedure is applied to improve the current solution. However, if a better solution is not found, a solution with slightly less profit is accepted. At the end of the iteration a perturbation move is applied, wherein a number of nodes (that depends on the current iteration) with the smallest ratio of profit to insertion cost are removed from the solution.

In [55] a tabu search heuristic for the unrooted OP is presented. The algorithm iteratively inserts clusters of nodes in the current tour or removes a chain of nodes. Compared to the previous approaches, this method reduces the chance to get trapped in a local optimum. Tests performed by the authors on randomly generated instances with up to 300 nodes show that the algorithm yields near-optimal solutions.

2.2 Orienteering Problem with Time Windows (OPTW)

In OP with Time Windows (OPTW) each node of the graph G can be visited only within one or more specific time intervals (windows) which may be different for each node (see Figure 4). Vansteenwegen et al. [109] argue that time windows significantly affect the nature of OP and its respective algorithmic approaches. For instance, reducing the travel time by reordering scheduled visits, is no longer appropriate due to the time windows. Actually, it has been proved that OPTW is NP-hard even on the line [106].

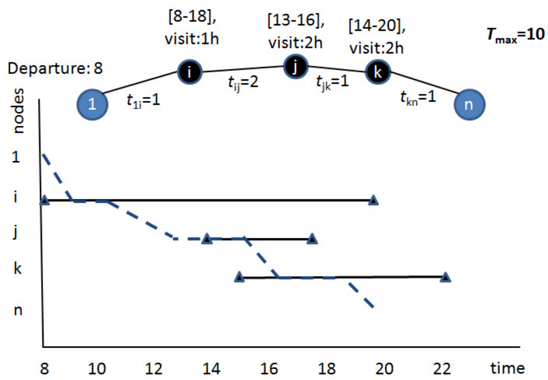


Figure 4: OPTW illustration (dashed lines denote the sheduled route, while triangles opening/closing times)

Righini et al. [91] give two exact dynamic programming algorithms for OPTW. The first algorithm uses bidirectional search and the label of each node u used in the algorithm, is a binary vector representing the nodes included in the path ending at u. In the second method, the state space relaxation (SSR) [38] is applied, where the label is only an integer denoting the number of visits along the path. Since in the second method a node may be visited more than once due to the reduced information kept at each label, the authors correct this by applying the decremental SSR (DSSR) method [90], which is an iterative algorithm optimally solving the relaxed problem with the additional constraint that a specific set of nodes cannot be visited more than once.

Kantor and Rosenwein [63] proposed two heuristics for solving the OPTW. The first, the insertion heuristic, incrementally builds the solution and at each step it selects the node with the highest ratio of profit over insertion cost as the next node to be inserted in the path. The second heuristic, the tree heuristic, is employed when the time windows constraints are tight and the input graph nodes are relatively few. By a depth first search exploration of the input graph, it maintains a number of partial solutions simultaneously and repeatedly inserts new nodes in these partially constructed paths as long as the attempted insertion satisfies the problem constraints and some heuristic criteria quantifying the potential solution improvement yield from this insertion.

Also, a number of OPTW approximation algorithms have been proposed in the literature. Bansal et al. [18] gave an $(3 \log^2 n)$-approximation algorithm for OPTW. The main idea is to partition the nodes into different groups according to their time windows and in such a way that OPTW can be solved in each group ignoring time windows. The final solution is derived by stitching the solutions of these subproblems using a dynamic programming approach.

Chekuri and Kumar [32] gave a 5-approximation algorithm for OPTW with at most k distinct time windows that runs in time polynomial in $(n\Delta)^k$, where Δ is the maximum distance in the metric space and n is the number of nodes. They utilize an approximation algorithm for the maximum coverage problem with group budget constraints[3] and a 3-approximation algorithm of Bansal et al [18] for OP.

Later, Chekuri and Pal [33] gave an $O(\log OPT)$-approximation algorithm for rooted OPTW in directed graphs where the total weight of the nodes visited has the submodular property. Their approach, based on a variant of an algorithm for directed s-t connectivity due to Savitch [92], is recursive and greedy and runs in quasi-polynomial time. An application of this algorithm can be found in [37] where travel itineraries for a city are constructed from information collected in the social breadcrumb *Flickr* about the preferences of tourists visiting the city.

Also, Chekuri et al. [31] inspired by the technique of Bansal et al. [18] proved that an α-approximation algorithm for OP yields an $O(\alpha \max \{\log OPT, \log L\})$ approximation algorithm for OPTW in directed (and undirected) graphs, where OPT denotes the number of nodes in an optimal solution and L is the ratio of the longest to the shortest time window.

Finaly, Frederickson et al. [50] proposed approximation algorithms for the travelling repairman problem (TRP) in a metric graph or a tree. TRP is a variant of OPTW, which aims at finding a path passing through the maximum number of nodes with each node visited within its time window. First, the algorithm trims all time windows into subwindows with specific ends and then for the nodes of each time window, the optimum k-path from s to t [30] is solved. Last, the solutions found for each time window are combined into a solution to the TRP by applying a dynamic programming approach. For the case that all time windows have equal length, it is proved that the optimal solution for the trimmed time windows is within factor of 3 from the optimal solution before trimming. Using the above result, the algorithm has a 3-approximation ratio with running time $O(n^4)$ when the input graph is a tree and a $(6+\epsilon)$-approximation for a general graph with $n^4 \cdot n^{O(\frac{1}{\epsilon^2})}$

[3]Given an integer k and a collection of subsets, of a set S, partitioned into groups, pick k subsets of that collection such that the cardinality of their union is maximized with the restriction that at most one set is picked from each group.

running time. Then, the authors generalize their method for time windows with different lengths and they derive an $O(\log L)$-approximation algorithm where L is the ratio of the maximum to minimum time length of all input windows.

3 Multiple tour TTDP solution approaches

3.1 Team Orienteering Problem (TOP)

The extension of the OP to multiple tours was defined as the Team Orienteering Problem by Chao et al. [29]. The TOP first appeared in the literature with the name Multiple Tour Maximum Collection Problem (MTMCP) by Butt and Cavalier [25]. TOP is an extension of OP where the goal is to find k paths (or tours) each with length bounded by B, that have the maximum total collected profit (each non-starting, non-terminal node is visited at most once along the k paths) (see Figure 5). TOP is NP-hard and APX-hard since OP is a special case of TOP.

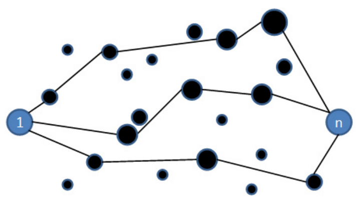


Figure 5: TOP illustration. Circles' radius denote nodes' profit.

TOP can be formulated as an integer programming problem as follows [109]: Further to the notation for OP, given the integer k, let x_{ijm} be equal to 1 if node i is followed by node j in path m or equal to 0 otherwise, y_{im} be equal to 1 if node i is visited in path m or equal to 0 otherwise and u_{im} be the position of node i in path m. With this notation we have the following relations:

$$\max \sum_{m=1}^{k} \sum_{i=2}^{N-1} p_i y_{im}, \tag{8}$$

s.t.

$$\sum_{m=1}^{k} \sum_{j=2}^{N} x_{1jm} = \sum_{m=1}^{k} \sum_{i=1}^{N-1} x_{iNm} = k, \tag{9}$$

$$\sum_{m=1}^{k} y_{rm} \leq 1, \text{ for all } r = 2, \ldots, N-1, \tag{10}$$

$$\sum_{i=1}^{N-1} x_{irm} = \sum_{j=2}^{N} x_{rjm} = y_{rm}, \text{ for all } r = 2, \ldots, N-1, m = 1, \ldots, k \quad (11)$$

$$\sum_{i=1}^{N-1} \sum_{j=2}^{N} c_{ij} x_{ijm} \leq B \text{ for all } m = 1, \ldots, k, \quad (12)$$

$$2 \leq u_{im} \leq N, \text{ for all } i = 1, 2, \ldots, N, \; m = 1, \ldots, k \quad (13)$$

$$u_{im} - u_{jm} + 1 \leq (N-1)(1 - x_{ijm}), \text{ for all } i, j = 2, \ldots, N, \; m = 1, \ldots, k \quad (14)$$

$$x_{ijm}, y_{im} \in \{0, 1\}, \text{ for all } i, j = 1, \ldots, N, \; m = 1, \ldots, k \quad (15)$$

The objective function (8) is to maximize the total profit of visited nodes. Constraints (9) and (10) ensure that each of the k paths starts at node 1 and ends at node N and that each non-starting, non-terminal node is visited at most once. Constraint (11) ensures that each path starting at node 1 and ending at node N is connected. Constraint (12) ensures that the path meets the time budget. Finally, constraints (13) and (14) ensure that there are no closed subtours.

Exact algorithms for TOP are presented by Butt et al. [26] and Boussier et al. [24]. Butt et al. [26] give an algorithm that optimally solves TOP by solving the relaxation of the problem with the column generation technique together with a branch and bound technique for deriving increasingly better solutions. Specifically, the problem is formulated as a set-partitioning problem and then a column generation procedure is applied. When applying the branch and bound technique, the solution space is partitioned around a specific node pair $\{u, v\}$ with one subspace containing solutions where both u, v belong to the same tour and the other one containing solutions where these two nodes cannot be part of the same tour. The combination of column generation and branch-and-bound technique (also known as branch-and-price in the literature) has also been applied in [24] for optimally solving the TOP. The selection of the new columns to be included at each step of column generation is reduced to solving an instance of Elementary Shortest Path Problem with Resource Constraint by using a dynamic programming approach. Finally, in a branch and bound phase, different branches are created according to either whether a node should be visited or not or whether a particular edge should be included in a tour or not.

Blum et al. [22] present an approximation algorithm for variants of TOP in undirected graphs, where the paths have a common start point and not a fixed end point or they are mutually disjoint. Their main idea is to iteratively apply algorithms for rooted OP setting already visited node profits to zero. For the former case, applying this procedure using an α-approximation algorithm for rooted OP, an $1/(1-e^{-\alpha})$ approximation ratio is obtained. While, in the latter case where the paths are mutually disjoint, using an α-approximation algorithm for rooted OP, an $(\alpha + 1)$ approximation ratio is obtained.

In the sequel, we outline the most important heuristic approaches for TOP (see Table 2). The first heuristic algorithm (BC) for TOP was presented by Butt and Cavalier [25]. They proposed a greedy algorithm that constructs the k tours successively. Every pair of nodes obtains a weight that gives an estimate of how advantageous it is to include both nodes in the same tour. Every tour initially contains the depot and the node pair with the greater weight. Then, at each step the node belonging to the heaviest pair of nodes with one of these nodes already in the tour is added to tour provided that this insertion is feasible.

Table 2: TOP Heuristic Algorithms

Reference	Algorithm	Technique
Butt and Cavalier [25]	BC	Greedy Insertions
Chao et al. [29]	CGW	Local Search
Tang and Miller-Hooks [102]	TMH	Tabu Search
Archetti et al.[10]	SVN, FVN	Variable Neighbourhood Search
	TS	Tabu search
Ke et al. [65]	ASe, ADC, ARC, ASi	Ant Colony Optimization
Vansteenwegen et al. [110]	GLS	Guided Local Search
Vansteenwegen et al. [112]	SVNS	Variable Neighbourhood Search
Souffriau et al. [99]	FPR, SPR	GRASP with Path Relinking
Bouly et al. [23]	MA	Genetic Algorithm
Muthuswamy et al. [83]	PSO	Discrete Particle Swarm Optimization

The heuristic algorithm (CGW) for the TOP presented by Chao et al. in [29] extends the one presented by the same authors for the OP in [28]. The main differences of the two algorithms are two. Firstly, in TOP the current solution contains the k (instead of one) most profitable paths. Secondly, in TOP there are two perturbation moves instead of one that holds for OP. The first move is identical for TOP and OP. In the second move of the TOP algorithm, a number of nodes with the lowest profit are removed from the paths of the solution.

In [102] a tabu search heuristic (TMH) for TOP is proposed by Tang and Miller-Hooks, comprising three basic steps: initialization, solution improvement and evaluation. TMH is embedded within an adaptive memory procedure that alternates between small and large neighborhood stages during the solution improvement phase. Both random and greedy procedures for neighborhood solution generation are employed, and infeasible as well as feasible solutions are explored in the process. The heursitic has been compared against CGW heuristic.

Archetti et al. [10] presented three metaheuristics solving the TOP. After defining a number of local search moves that can be applied in the solution space of the problem at hand, they present a tabu search heursitic and two variable neighborhood search [60] heuristics (the fast variable neighborhood search - FVN and the slow variable neighborhood search - SVN) which iteratively apply local search moves for gradually improving the solution derived at each step. The authors compare their algorithms with TMH and CGW and they show that each of the proposed heuristics improves the performance of TMH and CGW on average. They also show that FVN represents a fair compromise between solution quality and computational effort.

An Ant Colony Optimization-based heuristic algorithm (ACO) is proposed by Ke et al. [65] for TOP. Specifically, an iterative procedure is followed wherein the ants generate k feasible tours by succesively inserting promising edges from previous iterations associated with relatively low cost and high profit in their endnodes. Four methods, i.e., the sequential (ASe), the deterministic-concurrent (ADC), the random-concurrent (ARC) and the simultaneous (ASi) methods, are proposed to construct candidate solutions in the framework of ACO. The authors compare these methods with several existing approaches. The results obtained by ASe are as good as the results obtained by Archetti et al. [10], however they are faster to obtain. Therefore, it appears that ASe is a very good compromise between solution quality and computational effort.

A guided local search [115] metaheuristic algorithm (GLS) for the TOP is presented by Vansteenwegen et al. [110]. A solution to the problem is initialized as in CGW ([29] and a local search procedure is applied to improve it. Finally, guided local search is employed to ameliorate the effectiveness of the local search. In [112] Vansteenwegen et al. propose a Skewed Variable Neighbourhood Search (SVNS) framework for the TOP. The algorithms apply a combination of intensification and diversification procedures. The diversification procedures remove a chain of points in each path. The available budget spread over different paths within the current solution is gathered into a single path in the new solution. The intensification procedures try to increase the score or to decrease the travel time in a path. The SVNS algorithm clearly outperforms the GLS algorithm.

In [97] the authors employ the Greedy Randomised Adaptive Search Procedure (GRASP) to solve TOP. GRASP is a metaheuristic originally introduced by Feo and Resende [47]. GRASP performs a number of iterations that consist of a constructive procedure followed by a local search approach. The constructive procedure, based on a ratio between greediness and randomness, inserts nodes one by one until all paths are full. Thus, a new initial solution is generated during every iteration. Then, the initial solution is improved by the local search procedure which alternates between reducing the total time of the solution and increasing its total profit, until the solution is locally optimal. The different iterations are independent and the best solution found is saved and returned as a result. In [99] Souffriau et al. introduce a GRASP with Path Relinking metaheuristic approach for solving the TOP. The goal of the Path Relinking extension is to avoid the independence of the different iterations of the GRASP by adding a memory component, i.e. a pool of elite solutions consisting of a number of best solutions. At each iteration the best solution, considered for insertion into the pool of elite solutions, is returned by a procedure that takes as arguments a starting solution and a guiding solution and visits the solutions on the virtual path in the search space that connects the staring and the guiding solution. A fast variant (FPR) and a slow variant (SPR) of the approach are tested using a large set of test instances from the literature. The two heuristics are compared against other state-of-the-art approaches. The quality of the results of the slow variant is comparable to the quality obtained by the best algorithms of Archetti et al. [10] and Ke et al. [65].

Bouly et al. [23] propose a genetic algorithm (MA) for TOP enhanced with local search techniques. A population of chromosomes is constructed where a chromosome is a sequence of nodes from which a solution to TOP is obtained by applying a PERT[4] like technique. A child chromosome is produced by a couple of chromosomes by applying a crossover technique followed by a local search procedure with a certain probability. Computational results are compared with those of different methods such as CGW, TMH, the slow VNS algorithm (SVN), and the sequential method in the framework of ACO. It appears that MA outperforms SVN in terms of efficiency and is quite equivalent in terms of stability ([23])

Muthuswamy et al. [83] tackle the TOP using discrete particle swarm optimization (PSO), creating one tour at a time. At each step a population of particles is generated such that each particle represents a feasible tour. Then, using PSO particles are heading for more profitable solutions (tours). The whole procedure is enhanced with local search techniques.

In the survey article of Vansteenwegen et al. [109], a summary of the performance of the best TOP algorithms is given. The comparisons are based on 157 benchmark instances ([29]). For each algorithm, the number of times the best known solution is found, is given together with the average gap to the best solution and the average computational time.

3.1.1 Team Orienteering Problem with Time Windows (TOPTW)

The TOP with Time Windows (TOPTW) introduced by Vansteenwegen P. [107], extends TOP adding the constraint of limited time availability of serviced nodes (this corresponds to the opening and closing hours of a POI). Exact solutions for TOPTW are feasible for graphs with very restricted number of nodes (e.g. see the work by Z. Li and X. Hu [78] which is used on networks of up to 30 nodes).

Li and Hu formulated the Team Orienteering Problem with Capacity Constraint and Time Window (TOPCTW) [78] (an extension of TOPTW where each "customer" has a demand and the serving vehicle has a capacity limitation) and obtained exact solutions using an integer linear programming solver. However, this approach is inappropriate for real-time applications.

Given the complexity of the problem, the main body of TOPTW literature exclusively involves heuristic algorithms. Notably, existing methods are metaheuristics that involve, (a) an insertion step (adds a visit to one of the k tours) iteratively performed until a first feasible solution (or a set of feasible solutions) is obtained, and (b) a sort of local search step that aims at escaping from local optima. Those two steps are repeated until a termination criterion is met. Depending on the insertion step principle, existing methods

[4] program evaluation review technique

are designated either as deterministic (those that always produce the same solution for given problem instances) or as stochastic or probabilistic (those that involve a degree of randomness in solutions generation). Probabilistic methods are generally shown to yield high quality solutions (as they perform more extensive search of the solution space) at the expense of increased execution time.

Labadi et al. [70] propose a local search heuristic algorithm for TOPTW based on a variable neighbourhood structure. In the local search routine the algorithm tries to replace a segment of a path with nodes not included in a path that offer more profit. For that, an assignment problem related to the TOPTW is solved and based on that solution the algorithm decides which arcs to insert in the path.

Lin et al. [79] propose a heuristic algorithm based on simulated annealing (SA) for TOPTW. On each iteration a neighbouring solution is obtained from the current solution by applying one of the moves swap, insertion or inversion, with equal probability. If the new solution is more profitable than the current and with a probability depending on the difference of profits of the two solutions in the opposite case, the new solution is adopted and becomes the current one. After applying the above procedure for a certain number of iterations the best solution found so far is further improved by applying local search.

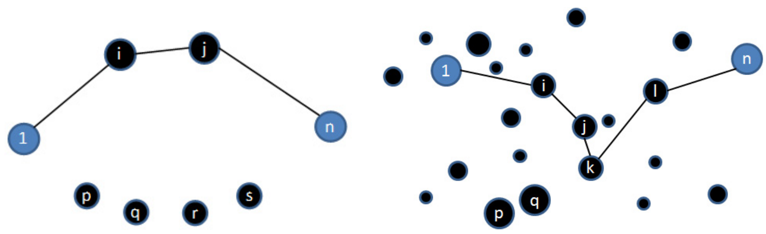
The Iterated Local Search (ILS) heuristic proposed by Vansteenwegen et al. [111] is the fastest known algorithm proposed for TOPTW [109]. ILS defines an "insertion" and a "shake" step. The insertion step adds, one by one, new visits to a tour, ensuring that all subsequent visits (those scheduled after the insertion place) remain feasible, i.e. they still satisfy their time window constraint. For each visit i that can be inserted, the cheapest insertion time cost is determined. For each of these visits the heuristic calculates a ratio, which represents a measure of how profitable is to visit i versus the time delay this visit incurs. Among them, the heuristic selects the one with the highest ratio for insertion. The shake step is used to escape from local optima. During this step, one or more visits are removed in each tour in search of non-included visits that may either decrease the tour time length or increase the overall collected profit.

Overall, ILS represents a fair compromise in terms of speed versus deriving routes of reasonable quality. However, ILS presents a number of shortcomings:

- In the insertion step, ILS may be attracted and included into the solution some high-score nodes isolated from high-density topology areas. This may trap ILS and make it infeasible to visit far located areas with good candidate noses due to prohibitively large travelling time (possibly leaving considerable amount of the overall time budget unused). For instance, in Figure 6(a), the itinerary $\{1, p, q, r, s, n\}$ would yield more profit and fully utilize the available time budget, compared to the solution $\{1, i, j, n\}$.
- During the insertion step, ILS rules out candidate nodes with high profit value as long as they are relatively time-expensive to reach (from nodes already included in routes). This is also the case even when whole groups of high profit nodes are located

within a restricted area of the plane but far from the current route instance. In case that the route instance gradually grows and converges towards the high profit nodes, those may be no longer feasible to insert due to overall tour time constraints. For instance, in Figure 6(b), ILS inserts i, l, j and k. Although p and q have larger profit value, they are not selected on the first four insertion steps since they are associated with large $Shift$ values. On the next step, q is associated with the highest $Ratio$, however its insertion violates the tour feasibility constraint; hence, it is not performed.

- The ILS shake step examines a very narrow space of alternative solutions. For instance ILS neglects swaps among visits included on the same or different itineraries which could potentially decrease the involved tours' length, thereby creating room for accommodating new visits until a new local optima is reached.



(a) (b)

Figure 6: Weaknesses of ILS.

Montemanni and Gambardella proposed an ant colony system (ACS) algorithm [82] to derive solutions for a hierarchical generalization of TOPTW, wherein more than the k required routes are constructed. At the expense of the additional overhead, those additional fragments are used to perform exchanges/insertions so as to improve the quality of the k tours. The algorithm comprises two phases:

- Construction phase: Ants are sent out sequentially; when at node i, an ant chooses probabilistically the next node j to visit (i.e. to include into the tour) based on two factors:
  - The pheromone trail τ_{ij} (i.e. a measure on how good it has been in the past to include arc (i, j) in the solution).
  - The desirability n_{ij}, (a node j is more desirable when it is associated with high profit, it is not far from i, and its time window is used in a suitable way).
- Local search: performed upon the solutions derived from construction phase, aiming at taking them down to a local optimum.

ACS has been shown to obtain high quality results (that is, low average gap to the best known solution) at the expense of prolonged execution time, practically prohibitive for online applications.

Tricoire et al. [104] deal with the Multi-Period Orienteering Problem with Multiple Time Windows (MuPOPTW), a generalization of TOPTW, wherein each node may be assigned more than one time window on a given day, while time windows may differ on different days. Both mandatory and optional visits are considered. The motivation behind this modelling is to facilitate individual route planning of field workers and sales representatives. The authors developed two heuristic algorithms for the MuPOPTW: a deterministic constructive heuristic which provides a starting solution, and a stochastic local search algorithm, the Variable Neighbourhood Search (VNS), which considers random exchanges between chains of nodes.

Vansteenwegen et al. [109] argue that a detailed comparison of TOPTW solution approaches (i.e. ILS, ACS and the algorithm of Tricoire et al. [104]), is impossible since the respective authors have used (slightly) different benchmark instances. Nevertheless, it can be concluded that ILS has the advantage of being very fast, while ACS and the approach of Tricoire et al. [104] (2010) have the advantage of obtaining high quality solutions.

Labadi et al. [71], [72] recently proposed a method that combines the greedy randomized adaptive search procedure (GRASP) with the evolutionary local search (ELS). GRASP generates independent solutions (using some randomized heuristic) further improved by a local search procedure. ELS generates multiple copies of a starting solution (instead of a single copy generated in ILS) using a random mutation (perturbation) and then applies a local search on each copy to yield an improved solution. GRASP-ELS derives solutions of comparable quality and significantly less computational effort to ACS. Compared to ILS, GRASP-ELS gives better quality solutions at the expense of increased computational effort. Table 3 summarizes the performance of GRASP-ESP, ILS and ACS presented in [72]. This comparison is based on a number of sets of instances: sets c100, r100, rc100 and c200, r200, rc200 designed by Solomon [95] and pr01-10, pr11-20 designed by Cordeau et al. [40]. The table reports for each method and for different number of tours k ($k = 1, ..., 4$), the average gap to the best known solution and the average computational time, over all instance sets. It appears that GRASP-ELS derives solutions of comparable quality and significantly less computational effort to ACS; compared to ILS, GRASP-ELS gives better quality solutions but it needs more computational effort.

Garcia et al. introduced the Multi-Constrained Team Orienteering Problem with Time Windows (MCTOPTW) [51]; each visit in the MCTOPTW is associated with a number of attributes; the sum of those attributes values is bounded by a max value (e.g. the sum of attractions entrance fee should not exceed an overall budget or the total time spent in parks cannot exceed a given time threshold). The proposed algorithm is based on ILS [111], incorporating two different aspects: (a) The feasibility check of visit insertions caters for checking constraints in addition to time feasibility; (b) the ratio function determining the candidate visit to be inserted is adapted so as to associate each attribute constraint with

Table 3: Comparison of TOPTW Metaheuristics

# of tours k	ILS		GRASP-ELS		ACS	
	Gap (%)	Time (s)	Gap (%)	Time (s)	Gap (%)	Time (s)
k=1	2.1	1.2	0.1	2.6	0.9	811.4
k=2	1.9	2.6	0.1	7.5	1.0	1812.9
k=3	1.9	3.6	0.1	8.8	0.8	1588.0
k=4	1.8	4.7	0.2	11.7	0.5	1286.9

a special weight and include the available quantity of each constraint on the route. For instance, if the total entrance fee constraint is assigned a relatively high weight, the algorithm favors insertions of visits with relatively low entrance fee, even more so if currently selected visits sum to low overall fee (relatively to the fee threshold).

Souffriau et al. [98] studied the Multi-Constraint Team Orienteering Problem with Multiple Time Windows (MCTOPMTW), in effect an extension of MCTOPTW which allows defining different/ multiple time windows for different days. The proposed MCTOPMTW algorithm is based on a hybrid ILS-GRASP approach: GRASP yields an initial solution (GRASP involves a degree of randomness in the insertion phase) and the 'shake' routine of ILS is used thereafter to derive an improved solution. The authors report that the ILS-GRASP algorithm yields fairly quality solutions, while achieving computation time suitable for online applications.

3.2 Time Dependent Team Orienteering Problem with Time Windows (TDTOPTW)

Time-dependent route planning incorporates time dependency in calculating cost of edges, i.e. travelling times among nodes. Time dependency is useful for modeling transfers among nodes through multimodal public transportation. Time-dependent graphs has been investigated in almost all variants of the orienteering problems, from the basic OP to the TOPTW.

Time Dependent OP (TDOP) was introduced by Formin and Lingas [49]. TDOP is MAX-SNP-hard since a special case of TDOP, time-dependent maximum scheduling problem is MAX-SNP-hard [100]. An exact algorithm for solving TDOP is given by Li et al [77] using a mixed integer programming model and a pre-node optimal labeling algorithm based on the idea of dynamic programming. Moreover, Li [76] proposes an exact algorithm for TDTOP based again on dynamic programming principles. However, both algorithms are of exponential complexity. Fomin and Lingas [49] give a $(2+\epsilon)$ approximation algorithm for rooted and unrooted TDOP (which runs in polynomial time if the ratio R between the maximum and minimum traveling time between any two sites is constant). When considering unrooted TDOP, its running time is $O\big((2R^2(\frac{2+\epsilon}{\epsilon}))!\frac{2R^2}{\epsilon}n^{2R^2(\frac{2+\epsilon}{\epsilon})+1}\big)$, and for rooted TDOP its running time increases by the multiplicative factor $O(\frac{Rn}{\epsilon})$ (the key idea is derived from Spieksma's algorithm [100] for Job Interval Selection Problem, which employs a divide-and-conquer approach). First, the problem is split in smaller ones. Exact solutions are found to each smaller problem and later combined (stitch) to obtain an approximate solution.

Abbaspour et al. [3] investigated a variant of Time Dependent OP with Time Windows (TDOPTW) in urban areas, where the nodes are partitioned into the POIs (associated with profits and time windows) and multimodal transportation stops which do not have profit. A genetic algorithm is proposed for the problem that uses as a subroutine another genetic algorithm for solving the shortest path problem between POIs.

TDTOPTW is the problem that better models more complicated and realistic TTDP requirements among all problems and approaches surveyed in this article. TDTOPTW is particulary complex as it adds time dependency of arcs to TOPTW. Zenker et al. [119] described a tourism-inspired problem that refers to TDTOPTW and presented ROSE, a mobile application assisting pedestrians to locate events and locations, moving through public transport connections. ROSE incorporates three main services: recommendation, route generation and navigation. The authors identified the route planning problem to solve and they described it as a multiple-constrained destination recommendation with time windows using public transportation. However, no algorithmic solution to this problem has been proposed.

The work of Garcia et al. [51], [52] is the first to address algorithmically the TDTOPTW and is based on the algorithm by Vansteenwegen et al. [111] for the TOPTW. The authors present two different approaches to solve TDTOPTW, both applied on real urban test instances. The first approach involves a pre-calculation step, computing the average travel times between all pairs of POIs, allowing reducing the TDTOPTW to a regular TOPTW. A repair procedure introduces the real travel times between the POIS of the derived TOPTW solution. In case that the TOPTW solution is infeasible (due to violating the time windows of POIs included in the solution), a number of visits are removed. The second approach considers direct public transportations, without transfers, and assumes only periodic service schedules. It modifies the insert procedure of the TOPTW ILS heuristic [111] by introducing a few new concepts and formulas to keep the concepts updated, and making possible the local and efficient evaluation of the possible insertion of an extra POI. The authors propose two variants of the second approach that take transfers into account. The fist variant is based on precalculating all required values for each pair of POIs. To reduce the number of calculations, the notion of the "period of a transfer connection" is used, defined as the least common multiple of the periods of all services involved in the transfer. The second variant models transfers as direct connections. The waiting time at the transfers is approximated by half of the period of the second service of the transfer. The authors tested all approaches for a set of instances based on real data for a city with around 50 POIS and with high frequency of public transportation. Based on the results of the tests the following can be concluded:

- The second approach (real travel time with no transfers) gives good solutions only for cities with a small number of POI to POI connections that are unfeasible without transfers. The approach needs low computation time (the same order of magnitude with the TOPTW algorithm [111]).
- In the case that the average travel times are good approximations of the real travel times, the first approach (average travel time approach) gives only slightly worse solutions compared to the second approach and its variants (real travel time approaches). This happens only when we have high frequency of public transportation. The computation time of the first approach is comparable with the one of the second approach.
- Both variants of the second approach improve the results obtained by the real travel time approach with no transfers (considering transfers widens the search space and leads to better results [52]). The first variant, i.e. the real time approach based on the precalculation, is not appropriate for big cities with a large number of POIS, as a lot of memory is required to store the precalculated values and retrieving the values is too time consuming. The second variant is less accurate than the first one but it is more suitable for bigger cities.

4 TTDP variants

Clearly, the combinatorial problems discussed in Section 2 and Section 3 closely match the TTDP modeling requirements. However, a large number of relevant problems investigated in the optimization algorithms literature could also capture various aspects and modeling parameters of TTDP variants and closely related problems. Algorithmic approaches to solve such problems are reviewed herein, explaining their utility in addressing TTDP variants and closely related problem requirements.

The Travelling Salesman Problem with Profits (TSPP) is a bicriteria generalization of TSP with two conflicting objectives. In TSPP we are given a network in which nodes are associated with profits and links with travel costs, and the goal is to find a tour (which starts and finishes at a specified node - the depot) over a subset of nodes such that the collected profit is maximized while the travel cost is minimized. The problem was introduced under the name multiobjective vending problem in [66]. In [19] the authors gave the first exact Pareto fronts (sets of non-dominated solutions) for TSPP instances obtained from classical TSP instances, available in the TSPLIB [89]. In [62] a hybrid meta-heuristic was presented that yields high-quality approximations of the efficient frontier for TSPP.

There are three single-criterion variants of TSPP based on how the two objectives of maximizing the collected profit and minimizing the travel cost are handled:

(i) The OP seeks for a tour that maximizes the total collected profit while maintaining the travel cost under a given value, i.e., the travel cost objective is stated as a constraint.

(ii) The Profitable Tour Problem (PTP) introduced in [43], searches for a tour that maximizes the collected profit minus the travel cost, i.e., the two objectives are combined in one objective function.

(iii) The Prize Collecting TSP (PCTSP) introduced in [17] aims at finding a tour that minimizes the travel cost, with the total tour profit being not smaller than a given value, i.e., the profit objective is stated as a constraint.

TSP is a special case of both PTP and PCTSP and, therefore, the two problems belong to the class of NP-hard problems. Bienstock et al. [20] developed the first approximation algorithm for PTP with a performance guarantee bound of 5/2. This bound was improved in [56] where a $2 - 1/(n-1)$-approximation algorithm was given, where n is the number of nodes. Awerbuch et al. [16] gave an approximation algorithm for the PCTSP based on an approximation algorithm for the k-minimum-spanning-tree problem ([14]). There also exists literature on exact, heuristic and metaheuristic algorithms for PTP and PCTSP as well as variants of these problems (see [46] for a survey).

A number of OP variants have been introduced in the literature to model TTDP variants as well as other practical problems:

1. The Generalized Orienteering Problem (GOP), wherein each node of the network is assigned a set of benefit values. For example, in the case of a POI, the benefit values may be related to natural beauty, cultural interest, historical significance, educational interest. The overall objective function may comprise any combination of the different benefits. Nonlinear objective functions make the GOP more difficult to solve than OP. In [116] a heuristic was designed to solve GOP using artificial neural networks, while in [117] a straightforward genetic algorithm was given that yields comparable results. In [94] an iterative algorithm was presented for the problem.

2. The Multi-Objective Orienteering Problem (MOOP) is the multi-objective extension of the OP which was formulated in [93] as follows. Each node (POI) may be assigned to different categories (e.g., culture, history, leisure, shopping) and provide different benefits for each category. The aim of MOOP is to find all Pareto efficient solutions without violating the maximum travel cost restriction. In [93] two metaheuristic solution techniques for the bi-objective OP were presented. The first is an adaptation of the Pareto Ant Colony Optimization metaheuristic developed by Doerner et al. [44]. The second is a multi-objective extension of VNS [60].

3. The following stochastic variants of the OP have been studied in the literature:

   - The Orienteering Problem with Stochastic Profits (OPSP), in which the nodes are associated with normally distributed profits. The problem was introduced in [61] and aims at finding a tour that starts and finishes at the depot, visits a subset of nodes within a time limit, and maximizes the probability of collecting more than a prespecified target profit level. In [61] the authors present an exact solution approach based on a parametric formulation of the problem for solving small problem instances and a Pareto-based bi-objective genetic algorithm for larger instances that is based on the conflict between high mean profit and low variance in a solution.

- The Stochastic Orienteering problem (SOP), in which each node is associated with a deterministic profit and a random service time. The visit time of a POI is not known until the visit is completed. The problem combines aspects of both the stochastic knapsack problem with uncertain item sizes and the OP. The stochastic orienteering problem was introduced in [59] where an $O(\log \log B)$-approximation algorithm was presented.
- The Orienteering Problem with Stochastic Travel and Service Times (OPSTS) which was introduced in [27], wherein both travel and service times are stochastic. If a node is visited, a reward is received, but if it is not, a penalty may be incurred. This problem reflects the challenges of an employee of a company who, on a given day, may have more customers to visit than he can serve. In [27] heuristics for general problem instances and computational results for a variety of parameter settings were given.

4. The OP with Compulsory Vertices (OPCV) discussed in [54], models the variant of OP in which a subset of the nodes has to be visited. In TTDP modeling, these compulsory nodes may be significant POIs that should be included in any itinerary. Gendreau et al. ([54]) developed a branch-and-cut algorithm to solve to optimality problem instances with up to 100 nodes, some of which are compulsory.

The Vehicle Routing Problem (VRP) can be described as the problem of designing optimal delivery or collection routes from a depot to a number of nodes subject to certain constraints. The most common constraints are (i) capacity constraints i.e., a demand is attached to each node and the sum of weights loaded on any route may not exceed the vehicle capacity, (ii) time constraints over individual routes, (iii) time windows, and (iv) precedence relations between pairs of nodes. Although many variants of the classical VRP have been studied based on different constraints (e.g., the Capacity-constrained VRP (CVRP), the Time or Distance constrained VRP (DVRP), the Vehicle Routing Problem with Time Windows (VRPTW), etc.) only a few can model tourist trip design problems. In the sequel, we discuss two problems that can formulate useful variants of TTDP: the DVRP and the Minimum Path Cover Problem (MPCP).

In DVRP, given a depot node r and a distance constraint D the goal is to find a minimum cardinality set of tours originating from r and corresponding to routes for vehicles, that covers all the nodes in the network ([73], [75], [85]). Each tour is required to have length at most D. DVRP may formulate the following problem: We are given a set of POIs and we are asked to determine the minimum number of days that will be needed to visit all POIs without violating the constraint of the available time per day. The unrooted version of DVRP, defined as the minimum path cover problem (MPCP) in [11], seeks for the minimum number of paths each of length at most D, that cover all the nodes of the network. Note that in MPCP, the paths may start and end at any two nodes. MPCP can be reduced to DVRP by adding a depot node that is located at some large distance L from all nodes, and setting the distance constraint to $D + 2L$. In [75] DVRP was studied under the objectives of total distance and number of tours. It was shown that the optimal solutions under both objectives are closely related, and any approximation guarantee for one objective implies a guarantee with an additional loss of factor 2, for the other objective. In [85] the authors presented an $(O(\log 1/\epsilon), 1 + \epsilon)$-approximation algorithm: i.e., for any $\epsilon > 0$, the algorithm provides a solution violating the length bound by a $1+\epsilon$ factor, while using at most $O(\log 1/\epsilon)$ times the optimal number of tours. The algorithm partitions the nodes of the network into subsets, according to their distance from the depot, and solves the unrooted DVRP with appropriate distance bounds on each subset. To solve the unrooted DVRP the $3-$approximation algorithm for the minimum path cover problem of Arkin et al.([11]) is employed that proceeds as follows. First, it guesses the solution value of k and then finds k paths with total length at most $2kD$ that cover the nodes of the network. Finally, it cuts the paths into smaller paths with length less than or equal to D.

The above variants of VRP assume that all nodes must be visited and there is no profit collected when visiting a node. Archetti et al. ([10]) name the extension of TSP with profits to multiple tours as Vehicle Routing Problem with Profits (VRPP). In VRPP visiting the whole set of nodes is not compulsory; a profit is collected when visiting a node, while collecting the profits is distributed over several vehicles with limited capacity. Known variants of the VRPP is the Prize-Collecting VRP (PCVRP), the Capacitated Profitable Tour Problem (CPTP) [8], and the VRP with profits and time deadlines (VRPP-TD). In PCVRP the main objective is a linear combination of three objectives: minimization of total distance traveled, minimization of vehicles used, and maximization of prizes collected [103]. In CPTP the objective is to maximize the difference between the total collected profit and the total travel cost [8]. In VRPP-TD, in addition to the capacity constraints, there are node-specific temporal constraints referred to as time deadlines. The objective function is the same with the function of CPTP [4].

The extension of the OP to multiple tours, i.e., the TOP, is a special case of VRP with profits. Archetti et al. ([8]) introduced the Capacitated Team Orienteering Problem (CTOP) as a TOP with an additional constraint, i.e., a nonnegative demand is associated with each node and the total demand in each tour may not exceed the given capacity constraint. They present exact and heuristic algorithms that are extentions of schemes for solving the TOP. The exact algorithm is an adaptation of a branch-and-price scheme first presented in [24], while the heuristic algorithms are based on the heursitic solutions for TOP given in [10]. In [7] a new branch-and-price scheme is presented to solve the CTOP.

A column-based heuristic is applied at each node of the branch-and-bound tree in order to obtain primal bound values.

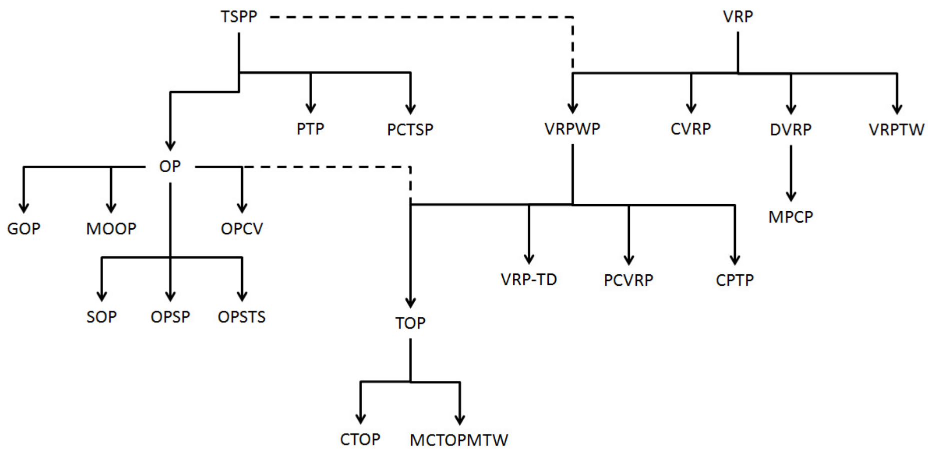


Figure 7: Variants of TTDP (solid arrows denote problem variants, while dashed arrows denote generalizations).

In all the above cited problems (see Figure 7) the sites/customers are represented by the nodes of a network. Also, the network nodes are associated with profits and/or demands. There is a limited literature on arc routing problems with profits i.e., problems in which the sites/customers are represented by the arcs of a network and the profits/demands are associated with the arcs. One such problem is the Prize-collecting Rural Postman Problem (PRPP) defined in [6]. In PRPP the arcs are associated with profits and costs, and the objective is to find a tour that maximizes the difference between the collected profit and the travel cost. Note that PRPP is the arc routing counterpart of the profitable tour problem (PTP). Problems related to post delivery and garbage collection can be modelled using PRPP, which has been studied from the algorithmic point of view in [45] and in [5].

In [9] the undirected Capacitated Arc Routing Problem with Profits(CARPP) was considered which is the arc routing counetrpart of the capacitated TOP (CTOP): a profit and a nonnegative demand is associated with each arc and the objective is to determine a path for each available vehicle in order to maximize the total collected profit, without violating the capacity and time limit constraints of each vehicle. The authors consider an application where carriers can select potential customers for transporting their goods. Another potential application is the creation of personalized bicycle trips. Based on the biker's personal interests, starting and ending point and the available time, a personalized trip can be composed using the selection of arcs that better match with the cyclist's profile.

The study of the combination of the orienteering problem and the arc routing problem with profits, under the name Mixed Orienteering Problem (MOP), where profits are associated to nodes as well as to arcs, is proposed in [109]. This problem is very interesting in the context of tourist trip planning as variants of MOP can be used to formulate TDDP variants where certain routes may be of tourist interest, in addition to attractions. To the best of our knowledge, no research has been done on MOP.

5 New prospects in tourist route planning problems

5.1 Quality improvement upon existing solution approaches

Evidently, extensions of the elementary OP problem (such as the TOPTW and the TD-TOPTW), which strongly resemble TTDP modeling, are particularly complex; hence, even heuristic approaches that derive high quality approaches (e.g. the algorithms that deal with TOPTW [72], [82], [104]) cannot meet the real-time execution requirement of TTDP web/mobile applications. Notably, ILS [29] and the algorithm of Garcia et al. [52] (proposed for TOPTW and TDTOPTW, respectively) significantly reduce execution time. However, it appears that several promising new directions exist to further improve the quality of solutions derived by those algorithms.

For instance, the insertion phase of ILS overlooks attractive candidate nodes (i.e. POIs associated with relatively high profit) located far from currently selected nodes, as the insertion of such nodes would considerably increase route travel time. This is also the case when considering larger groups of nearby attractive candidate nodes located far from the current solution's nodes. In such scenarios, the increased travel cost of visiting the first node would be soon compensated by successive visits to other nearby interesting sites; yet, deterministic approaches like ILS will fail to incorporate such groups of nodes into derived solutions as they examine candidate nodes individually. In fact, realistic TTDP problem instances are likely to match such high-profit node distribution patterns. Namely, city tourist attraction maps typically include distinct areas (possibly far located from each other or from tourist hotels) with high density of must-see POIs.

A way around this problem would be the identification of node clusters located in close proximity with relatively high average profit, prior to executing the insertion phase. Several cluster analysis algorithms such as the k-means or the fuzzy c-means clustering [118] could serve for partitioning available nodes into separate groups (clusters). Certainly, the clustering criteria could be adjusted to incorporate several attributes (in addition to distance), such as the feasibility of successive visits to cluster nodes with respect to their time windows. Thereafter, several alternative insertion criteria could be examined to bias solutions including such node clusters. It is noted that the clustering procedure could be performed offline to save online queries execution time.

Another characteristic of TTDP overlooked by TOPTW algorithms is the fact that, typically, POIs time windows largely overlap. This fact could be utilized to effectively reduce TOPTW to TOP and thereafter apply perturbations typically used in TOP algorithms (such as 2-opt exchanges) to further improve derived solutions.

5.2 Modeling and solving TOPTW generalizations

The state-of-the-art relevant to the OP and VRP families of problems presented in previous sections, reveals that little has been done in regards with tourist trip design problems that have more complicated requirements and constraints, e.g. allowing modeling multiple user constraints and transfers through public transportation. This highlights a promising field of research which calls for modeling and solving extensions of TOPTW and TDTOPTW that take into account realistic TTDP issues or constraints like the following:

- Weather conditions: museums may be more appropriate to visit than open-air sites in rainy or relatively cold days, while the contrary may be true in sunny days; hence, route planning could take into account weather forecast information in recommending daily itineraries.
- Accessibility features of sites should be taken into account when recommending visits to individuals with motor disabilities.
- Tourists are commonly under inflexible budget restrictions when considering accommodation, meals, means of transport or visits to POIs with entrance fees. Hence, next to the time budget, money budget further constrains the selection of POI visits.
- Recommended tourist routes that exclusively comprise POI visits and last longer than a few hours are unlikely to be followed closely. Tourists typically enjoy relaxing and breaks as much as they enjoy visits to POIs. A realistic route should therefore provide for breaks either for resting (e.g. at a nearby park) or for a coffee and meal. Coffee and meal breaks are typically specific in number, while respective recommendations may be subject to strict time window (e.g. meal should be scheduled around noon) and budget constraints.
- The assumption of POIs having periodic time windows is invalid. POIs typically operate at specific days weekly, possibly with varying opening and closing hours. Hence, TTDP modelling should take into account multiple time windows.
- Max-n Type [96] constrains the selection of POIs by allowing stating a maximum number of certain types of POIs, per day or for the whole trip. e.g. maximum two museum visits on the first day. Likewise, mandatory visits (i.e. tours including at least one visit to a POI of certain type, such as a visit to a church) could also be asked for.
- Tourists commonly prefer strolling downtown rather than visiting museums. In such cases, tourists may prefer to walk along routes featuring buildings and squares with historical value or routes with scenic beauty. Such routes are likely to be preferred also when moving among POIs, e.g. a detour through a car-free street along a medieval castle walls would be more appreciated than following a shortest path though streets with car traffic.

5.3 Modeling and solving relevant problems

Modeling and solving of problems relevant to TTDP represents another promising research direction. For instance, hotel selection is often a cumbersome task for tourists unfamiliar with hotels and POI locations or with the structure of the public transportation network in the tourist destination area. This is even more true when planning long road trips across large geographic areas (in such scenarios, changing accommodation in daily basis is common) [108]. Several criteria could apply in hotel recommendation, including cost, amenities or cost-for-profit (i.e. select an affordable hotel suitably located so as to maximize the overall profit collected from POI visits throughout the whole trip). Restaurants selection is equally important as meal/dinner breaks are mandatory, while constrained by several - often contradictory - user preferences (e.g. budget, diet preferences, favorite cuisine) and restaurant characteristics (e.g. menu, price list, opening hours).

Another example is the problem of determining the minimum number of days that one needs to visit all selected POIs without violating the constraint of the available time per day. This problem may be formulated using the distance constrained vehicle problem (DVRP) described in Section 4. Other interesting variants of TTDP may be formulated using the mixed orienteering problem, also discussed in Section 4.

5.4 Fast tourist routes updates

Existing TTDP solutions deal with tourist queries for multiple days' route planning, considering routes with the same starting/ending location. However, there is no provision for user deviations from the originally planned routes, although such deviations are highly probable to occur.

Dynamic rescheduling functionality should detect route invalidation (infeasibility) and present a new route schedule in real time. This should exclude POIs already visited and recommend a tour for the remainder of the current day (starting from the user's current position) as well as the next days of stay at the destination.

5.5 Parallel computation

One of the most important objectives in the design of algorithmic methods for the TTDP is the real time response to user queries. Parallel computing is a promising approach for attaining this important objective. Considering all the solution methods for TTDP, heuristics and metaheuristics are most amenable to parallel computation since the huge solution space arising in this kind of problems enables a lot of variation in parallelizing solution searching. Specifically, according to [42] one could parallelize the local search for good neighboring solutions or partition the solution space in number of subspaces and run a heuristic in each of these subspaces, in parallel. Alternatively, a number of search threads could be created working on the same solution space, starting from different or the same initial solution and applying same or different heuristics. These threads could work independently or could cooperate periodically exchanging information about their progress and the good solutions they have found so far. An interesting aspect of these approaches is that they may as well provide new heuristic solutions with improved solution quality since they can search the solution space and combine solutions in such a way that it is very costly to simulate with a sequential implementation. Although, parallel heuristics has been proposed in the literature for the VRP and TSP [35], [39], [41], [86], [101] parallel solutions for TTDP are missing and the design of new parallel heuristics for TTDP may solve the problem of the fast derivation of the tourist itineraries.

Acknowledgement
This work was supported by the EU FP7/2007-2013 (DG CONNECT.H5-Smart Cities and Sustainability), under grant agreement no. 288094 (project eCOMPASS).

References

[1] City trip planner, http://www.citytripplanner.com/, Last accessed: March 2012.

[2] mtrip travel guides, http://www.mtrip.com/, Last accessed: March 2012.

[3] R. A. Abbaspour and F. Samadzadegan. Time-dependent personal tour planning and scheduling in metropolises. *Expert Syst. Appl.*, 38:12439–12452, September 2011.

[4] D. Aksen and N. Aras. Customer selection and profit maximization in vehicle routing problems. In *Operations Research Proceedings 2005*, Operations Research Proceedings. Springer Berlin Heidelberg.

[5] J. Aráoz, E. Fernández, and O. Meza. Solving the prize-collecting rural postman problem. *European Journal of Operational Research*, 196(3):886 – 896, 2009.

[6] J. Aráoz, E. Fernández, and C. Zoltan. Privatized rural postman problems. *Computers & Operations Research*, 33(12):3432–3449, 2006.

[7] C. Archetti, D. Feillet, A. Hertz, and M. G. Speranza. Optimal solutions for routing problems with profits.

[8] C. Archetti, D. Feillet, A. Hertz, and M. G. Speranza. The capacitated team orienteering and profitable tour problems. *J. Oper. Res. Soc.*, 60:831–842, 2009.

[9] C. Archetti, D. Feillet, A. Hertz, and M. G. Speranza. The undirected capacitated arc routing problem with profits. *Computers & Operations Research*, 37(11):1860 – 1869, 2010.

[10] C. Archetti, A. Hertz, and M. Speranza. Metaheuristics for the team orienteering problem. *Journal of Heuristics*, 13:49–76, 2007.

[11] E. M. Arkin, R. Hassin, and A. Levin. Approximations for minimum and min-max vehicle routing problems. *Journal of Algorithms*, 59(1):1 – 18, 2006.

[12] E. M. Arkin, J. S. B. Mitchell, and G. Narasimhan. Resource-constrained geometric network optimization. In *Proceedings of the fourteenth annual symposium on Computational geometry*, SCG '98, pages 307–316, New York, NY, USA, 1998. ACM.

[13] S. Arora. Polynomial time approximation schemes for euclidean traveling salesman and other geometric problems. *J. ACM*, 45(5):753–782, sep 1998.

[14] S. Arora and G. Karakostas. A $2+\epsilon$ approximation algorithm for the k-mst problem. *Mathematical Programming*, 107:491–504.

[15] A. Asadpour, M.X. Goemans, A. Madry, S.O. Gharan, and A. Saberi. An $o(\log n/\log\log n)$-approximation algorithm for the asymmetric traveling salesman problem. In *ACM-SIAM Symposium on Discrete Algorithms*, volume 2010, pages 379–389. 2010.

[16] B. Awerbuch, Y. Azar, A. Blum, and S. Vempala. New approximation guarantees for minimum-weight k-trees and prize-collecting salesmen. *SIAM J. Comput.*, 28(1):254–262, 1998.

[17] E. Balas. The prize collecting traveling salesman problem. *Networks*, 19(6):621–636, 1989.

[18] N. Bansal, A. Blum, S. Chawla, and A. Meyerson. Approximation algorithms for deadline-tsp and vehicle routing with time-windows. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, STOC '04, pages 166–174, New York, NY, USA, 2004. ACM.

[19] J.-F. Bérubé, M. Gendreau, and J.-Y. Potvin. An exact epsilon-constraint method for bi-objective combinatorial optimization problems: Application to the traveling salesman problem with profits. *European Journal of Operational Research*, 194(1):39–50, 2009.

[20] D. Bienstock, M. X. Goemans, D. Simchi-Levi, and D. Williamson. A note on the prize collecting traveling salesman problem. *Mathematical Programming*, 59:413–420.

[21] A. Blum, S. Chawla, D. R. Karger, T. Lane, A. Meyerson, and M. Minkoff. Approximation algorithms for orienteering and discounted-reward tsp. In *Foundations of Computer Science, 2003. Proceedings. 44th Annual IEEE Symposium on*, pages 46 –55, oct. 2003.

[22] A. Blum, S. Chawla, D. R. Karger, Lane T., A. Meyerson, and M. Minkoff. Approximation algorithms for orienteering and discounted-reward tsp. *SIAM J. Comput.*, 37(2):653–670, 2007.

[23] H. Bouly, D.-C. Dang, and A. Moukrim. A memetic algorithm for the team orienteering problem. *4OR: A Quarterly Journal of Operations Research*, 8:49–70.

[24] S. Boussier, D. Feillet, and M. Gendreau. An exact algorithm for team orienteering problems. *4OR: A Quarterly Journal of Operations Research*, 5:211–230.

[25] S. E. Butt and T. M. Cavalier. A heuristic for the multiple tour maximum collection problem. *Computers & Operations Research*, 21(1):101 – 111, 1994.

[26] S. E. Butt and D. M. Ryan. An optimal solution procedure for the multiple tour maximum collection problem using column generation. *Computers and Operations Research*, 26(4):427 – 441, 1999.

[27] A. Campbell, M. Gendreau, and B. Thomas. The orienteering problem with stochastic travel and service times. *Annals of Operations Research*, 186:61–81.

[28] I-M. Chao, B. L. Golden, and E. A. Wasil. A fast and effective heuristic for the orienteering problem. *European Journal of Operational Research*, 88(3):475 – 489, 1996.

[29] I-M. Chao, B. L. Golden, and E. A. Wasil. The team orienteering problem. *European Journal of Operational Research*, 88(3):464 – 474, 1996.

[30] K. Chaudhuri, B. Godfrey, S. Rao, and K. Talwar. Paths, trees, and minimum latency tours. In *Foundations of Computer Science, 2003. Proceedings. 44th Annual IEEE Symposium on*, pages 36 – 45, oct. 2003.

[31] C. Chekuri, N. Korula, and M. Pál. Improved algorithms for orienteering and related problems. In *Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms*, SODA '08, pages 661–670, Philadelphia, PA, USA, 2008. Society for Industrial and Applied Mathematics.

[32] C. Chekuri and A. Kumar. Maximum coverage problem with group budget constraints and applications. In *PROC. OF APPROX, SPRINGER LNCS, 722̌01383*, pages 72–83, 2004.

[33] C. Chekuri and M. Pal. A recursive greedy algorithm for walks in directed graphs. In *Foundations of Computer Science, 2005. FOCS 2005. 46th Annual IEEE Symposium on*, pages 245 – 253, oct. 2005.

[34] K. Chen and S. Har-Peled. The orienteering problem in the plane revisited. In *Proceedings of the twenty-second annual symposium on Computational geometry*, SCG '06, pages 247–254, New York, NY, USA, 2006. ACM.

[35] L. Chen, H.-Y. Sun, and S. Wang. A parallel ant colony algorithm on massively parallel processors and its convergence analysis for the travelling salesman problem. *Information Sciences*, 2012. in press.

[36] K. Cheverst, K. Mitchell, and N. Davies. The role of adaptive hypermedia in a context-aware tourist guide. *Commun. ACM*, 45(5):47–51, May 2002.

[37] M. De Choudhury, M. Feldman, S. Amer-Yahia, N. Golbandi, R. Lempel, and C. Yu. Automatic construction of travel itineraries using social breadcrumbs. In *Proceedings of the 21st ACM conference on Hypertext and hypermedia*, HT '10, pages 35–44, New York, NY, USA, 2010. ACM.

[38] N. Christofides, A. Mingozzi, and P. Toth. State-space relaxation procedures for the computation of bounds to routing problems. *Networks*, 11(2):145–164, 1981.

[39] J.-F. Cordeau and M. Maischberger. A parallel iterated tabu search heuristic for vehicle routing problems. *Computers & Operations Research*, 39:2033–2050, 2012.

[40] J.F. Cordeau, M. Gendreau, and G. Laporte. A tabu search heuristic for periodic and multi-depot vehicle routing. *Networks*, 30:105–119, 1997.

[41] T. G. Crainic. Parallel solution methods for vehicle routing problems. In B. Golden, editor, *The Vehicle Routing Problem.* Springer Science+Business Media.

[42] T. G. Crainic and M. Toulouse. Parallel meta-heuristics. In M. Gendreau and J.-Y. Potvin, editors, *Handbook of Metaheuristics*, International Series in Operations Research & Management Science. Springer Science+Business Media.

[43] M. Dell'Amico, F. Maffioli, and P. Värbrand. On prize-collecting tours and the asymmetric travelling salesman problem. *International Transactions in Operational Research*, 2(3):297–308, 1995.

[44] K. Doerner, W. Gutjahr, R. Hartl, C. Strauss, and C. Stummer. Pareto ant colony optimization: A metaheuristic approach to multiobjective portfolio selection. *Annals of Operations Research*, 131:79–99.

[45] D. Feillet, P. Dejax, and M. Gendreau. The profitable arc tour problem: Solution with a branch-and-price algorithm. *Transportation Science*, 39(4):539–552, 2005.

[46] D. Feillet, P. Dejax, and M. Gendreau. Traveling salesman problems with profits. *Transportation Science*, 39(2):188–205, 2005.

[47] T. A. Feo and M. G. C. Resende. A probabilistic heuristic for a computationally difficult set covering problem. 8:67–71, 1989.

[48] M. Fischetti, J. J. S. González, and P. Toth. Solving the orienteering problem through branch-and-cut. *INFORMS Journal on Computing*, 10(2):133–148, 1998.

[49] F. V. Fomin and A. Lingas. Approximation algorithms for time-dependent orienteering. *Information Processing Letters*, 83(2):57 – 62, 2002.

[50] G. Frederickson and B. Wittman. Approximation algorithms for the traveling repairman and speeding deliveryman problems. *Algorithmica*, 62:1198–1221.

[51] A. Garcia, M. Linaza, O. Arbelaitz, and P. Vansteenwegen. Intelligent routing system for a personalised electronic tourist guide. In Wolfram Hpken, Ulrike Gretzel, and Rob Law, editors, *Information and Communication Technologies in Tourism 2009*, pages 185–197. Springer Vienna.

[52] A. Garcia, P. Vansteenwegen, O. Arbelaitz, W. Souffriau, and M. Linaza. Integrating public transportation in personalised electronic tourist guides. *Computers & Operations Research*, (0):–, 2011.

[53] D. Gavalas, M. Kenteris, C. Konstantopoulos, and Pantziou. A web application for recommending personalized mobile tourist routes. *IET Software*, 6:313–322, 2012.

[54] M. Gendreau, G. Laporte, and F. Semet. A branch-and-cut algorithm for the undirected selective traveling salesman problem. *Networks*, 32(4):263–273, 1998.

[55] M. Gendreau, G. Laporte, and F. Semet. A tabu search heuristic for the undirected selective travelling salesman problem. *European Journal of Operational Research*, 106(2-3):539 – 545, 1998.

[56] M. X. Goemans and D. P. Williamson. A general approximation technique for constrained forest problems. *SIAM J. Comput.*, 24(2):296–317, 1995.

[57] B. Golden, Q. Wang, and L. Liu. A multifaceted heuristic for the orienteering problem. *Naval Research Logistics*, 35.

[58] B. L. Golden, L. Levy, and R. Vohra. The orienteering problem. *Naval Research Logistics (NRL)*, 34(3):307–318, 1987.

[59] A. Gupta, R. Krishnaswamy, V. Nagarajan, and R. Ravi. Approximation algorithms for stochastic orienteering. In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '12, pages 1522–1538. SIAM, 2012.

[60] P. Hansen and N. Mladenovic. An introduction to variable neighborhood search. In *Metaheuristics, Advances and Trends in Local Search Paradigms for Optimization.*

[61] T. Ilhan, S. M. R. Iravani, and M. S. Daskin. The orienteering problem with stochastic profits. *IIE Transactions*, 40(4):406–421, 2008.

[62] N. Jozefowiez, F. Glover, and M. Laguna. Multi-objective meta-heuristics for the traveling salesman problem with profits. *Journal of Mathematical Modelling and Algorithms*, 7:177–195.

[63] M. G. Kantor and M. B. Rosenwein. The orienteering problem with time windows. *The Journal of the Operational Research Society*, 43(6):pp. 629–635, 1992.

[64] S. Kataoka and S. Morito. An algorithm for single constraint maximum collection problem. *Journal of the Operations Research Society of Japan*, 31(4):515–530, 1988.

[65] L. Ke, C. Archetti, and Z. Feng. Ants can solve the team orienteering problem. *Comput. Ind. Eng.*, 54(3):648–665, April 2008.

[66] C. P. Keller and M. F. Goodchild. The multiobjective vending problem: a generalization of the travelling salesman problem. *Environment and Planning B: Planning and Design*, 15:447–460, 1988.

[67] M. Kenteris, D. Gavalas, and D. Economou. Electronic mobile guides: a survey. *Personal and Ubiquitous Computing*, 15:97–111.

[68] M. Kenteris, D. Gavalas, and D. Economou. An innovative mobile electronic tourist guide application. *Personal and Ubiquitous Computing*, 13:103–118.

[69] N. J. Korula. *Approximation algorithms for network design and orienteering*. PhD thesis, University of Illinois at Urbana-Champaign, 2010.

[70] N. Labadi, R. Mansini, J. Melechovský, and R. Wolfler Calvo. The team orienteering problem with time windows: An lp-based granular variable neighborhood search. *European Journal of Operational Research*, 220(1):15 – 27, 2012.

[71] N. Labadi, J. Melechovský, and R. Calvo. An effective hybrid evolutionary local search for orienteering and team orienteering problems with time windows. In Robert Schaefer, Carlos Cotta, Joanna Kolodziej, and Gnter Rudolph, editors, *Parallel Problem Solving from Nature PPSN XI*, volume 6239 of *Lecture Notes in Computer Science*, pages 219–228. Springer Berlin / Heidelberg.

[72] N. Labadi, J. Melechovský, and R. Wolfler Calvo. Hybridized evolutionary local search algorithm for the team orienteering problem with time windows. *Journal of Heuristics*, 17:729–753.

[73] G. Laporte, M. Desrochers, and Y. Nobert. Two exact algorithms for the distance-constrained vehicle routing problem. *Networks*, 14(1):161–172, 1984.

[74] G. Laporte and S. Martello. The selective travelling salesman problem. *Discrete Applied Mathematics*, 26(2-3):193 – 207, 1990.

[75] C.-L. Li, D. Simchi-Levi, and M. Desrochers. On the distance constrained vehicle routing problem. *Operations Research*, 40(4):pp. 790–799, 1992.

[76] J. Li. Model and algorithm for time-dependent team orienteering problem. In Song Lin and Xiong Huang, editors, *Advanced Research on Computer Education, Simulation and Modeling*, volume 175 of *Communications in Computer and Information Science*, pages 1–7. Springer Berlin Heidelberg, 2011.

[77] J. Li, Q. Wu, X. Li, and D. Zhu. Study on the time-dependent orienteering problem. In *E-Product E-Service and E-Entertainment (ICEEE), 2010 International Conference on*, pages 1–4, nov. 2010.

[78] Z. Li and X. Hu. The team orienteering problem with capacity constraint and time window. *The Tenth International Symposium on Operations Research and Its Applications (ISORA 2011)*, pages 157–163, August 2011.

[79] S.-W. Lin and V. F. Yu. A simulated annealing heuristic for the team orienteering problem with time windows. *European Journal of Operational Research*, 217(1):94 – 107, 2012.

[80] R. Malaka and A. Zipf. Deep map - challenging it research in the framework of a tourist information system. In *Proceedings of the International Conference on Information and Communication Technologies in Tourism (ENTER 2000)*, pages 15–27, 2000.

[81] J. S. B. Mitchell. Guillotine subdivisions approximate polygonal subdivisions: A simple polynomial-time approximation scheme for geometric tsp, k-mst, and related problems. *SIAM J. Comput.*, 28(4):1298–1309, 1999.

[82] R. Montemanni and L. M. Gambardella. An ant colony system for team orienteering problems with time windows. *Foundations of Computing and Decision Sciences*, 34(4):287–306, 2009.

[83] S. Muthuswamy and S. Lam. Discrete particle swarm optimization for the team orienteering problem. *Memetic Computing*, 3:287–303.

[84] V. Nagarajan and R. Ravi. The directed orienteering problem. *Algorithmica*, 60:1017–1030, August 2011.

[85] V. Nagarajan and R. Ravi. Approximation algorithms for distance constrained vehicle routing problems. *Networks*, 59(2):209–214, 2012.

[86] K.-W. Pang. An adaptive parallel route construction heuristic for the vehicle routing problem with time windows constraints,. *Expert Systems with Applications*, 38:11939–11946, 2011.

[87] R. Ramesh and K. M. Brown. An efficient four-phase heuristic for the generalized orienteering problem. *Computers & Operations Research*, 18(2):151 – 165, 1991.

[88] R. Ramesh, Y.-S. Yoon, and M. H. Karwan. An optimal algorithm for the orienteering tour problem. *ORSA Journal on Computing*, 4(2):155–165, Spring 1992.

[89] G. Reinelt. Tsplib - a traveling salesman problem library. *ORSA Journal on Computing*, 3(4):376–384, 1991.

[90] G. Righini and M. Salani. New dynamic programming algorithms for the resource constrained elementary shortest path problem. *Networks*, 51(3):155–170, 2008.

[91] G. Righini and M. Salani. Decremental state space relaxation strategies and initialization heuristics for solving the orienteering problem with time windows with dynamic programming. *Computers & Operations Research*, 36(4):1191 – 1203, 2009.

[92] W. J. Savitch. Relationships between nondeterministic and deterministic tape complexities. *J. Comput. Syst. Sci.*, 4(2):177–192, 1970.

[93] M. Schilde, K. Doerner, R. Hartl, and G. Kiechle. Metaheuristics for the bi-objective orienteering problem. *Swarm Intelligence*, 3:179–201.

[94] J. Silberholz and B. Golden. The effective application of a new approach to the generalized orienteering problem. *Journal of Heuristics*, 16:393–415.

[95] F. Solomon. Algorithms for the vehicle routing and scheduling problem with time windows constraints. *Operations Research*, 35:254–265, 1987.

[96] W. Souffriau and P. Vansteenwegen. Tourist trip planning functionalities: State-oftheart and future. In F. Daniel and F. Facca, editors, *Current Trends in Web Engineering*, volume 6385 of *Lecture Notes in Computer Science*, pages 474–485. Springer Berlin / Heidelberg.

[97] W. Souffriau, P. Vansteenwegen, G. Vanden Berghe, and D. Van Oudheusden. A greedy randomised adaptive search procedure for the team orienteering problem. In *EU/MEeting 2008 on metaheuristics for logistics and vehicle routing*, October 2008.

[98] W. Souffriau, P. Vansteenwegen, G. Vanden Berghe, and D. Van Oudheusden. The multiconstraint team orienteering problem with multiple time windows. *Transportation Science*, 2011. in press.

[99] W. Souffriau, P. Vansteenwegen, G. Vanden Berghe, and G. Van Oudheusden. A path relinking approach for the team orienteering problem. *Computers and Operations Research*, 37:1853–1859, 2010.

[100] F. C. R. Spieksma. On the approximability of an interval scheduling problem. *Journal of Scheduling*, 2:215–227, 1999.

[101] A. Subramanian, L. M. A. Drummonda, C. Bentes, L. S. Ochi, and R. Farias. A parallel heuristic for the vehicle routing problem with simultaneous pickup and delivery. *Computers & Operations Research*, 37:1899–1911, 2010.

[102] H. Tang and E. Miller-Hooks. A tabu search heuristic for the team orienteering problem. *Computers & Operations Research*, 32(6):1379 – 1407, 2005.

[103] L. Tang and X. Wang. Iterated local search algorithm based on very large-scale neighborhood for prize-collecting vehicle routing problem. *The International Journal of Advanced Manufacturing Technology*, 29:1246–1258.

[104] F. Tricoire, M. Romauch, K. F. Doerner, and R. F. Hartl. Heuristics for the multi-period orienteering problem with multiple time windows. *Computers & Operations Research*, 37(2):351 – 367, 2010.

[105] T. Tsiligirides. Heuristic methods applied to orienteering. *The Journal of the Operational Research Society*, 35(9):pp. 797–809, 1984.

[106] J. N. Tsitsiklis. Special cases of traveling salesman and repairman problems with time windows. *Networks*, 22:263–282, 1992.

[107] P. Vansteenwegen. *Planning in Tourism and Public Transportation - Attraction Selection by Means of a Personalised Electronic Tourist Guide and Train Transfer Scheduling.* PhD thesis, Katholieke Universiteit Leuven, 2008.

[108] P. Vansteenwegen, W. Souffriau, and K. Sorensen. The travelling salesperson problem with hotel selection. *Journal of the Operational Research Society*, 63:207–217, 2012.

[109] P. Vansteenwegen, W. Souffriau, and D. Van Oudheusden. The orienteering problem: A survey. *European Journal of Operational Research*, 209(1):1 – 10, 2011.

[110] P. Vansteenwegen, W. Souffriau, G. Vanden Berghe, and D. Van Oudheusden. A guided local search metaheuristic for the team orienteering problem. *European Journal of Operational Research*, 196(1):118 – 127, 2009.

[111] P. Vansteenwegen, W. Souffriau, G. Vanden Berghe, and D. Van Oudheusden. Iterated local search for the team orienteering problem with time windows. *Comput. Oper. Res.*, 36:3281–3290, December 2009.

[112] P. Vansteenwegen, W. Souffriau, G. Vanden Berghe, and D. Van Oudheusden. Metaheuristics for tourist trip planning. In M. Geiger, W. Habenicht, M. Sevaux, and K. Sorensen, editors, *Metaheuristics in the Service Industry*, volume 624 of *Lecture Notes in Economics and Mathematical Systems*, pages 15–31. Springer-Verlag, 2009.

[113] P. Vansteenwegen, W. Souffriau, G. Vanden Berghe, and D. Van Oudheusden. The city trip planner: An expert system for tourists. *Expert Systems with Applications*, 38(6):6540 – 6546, 2011.

[114] P. Vansteenwegen and D. Van Oudheusden. The mobile tourist guide: An or opportunity. *Operational Research Insight*, 20(3):21–27, 2007.

[115] C. Voudouris and E. Tsang. Guided local search and its application to the traveling salesman problem. *European Journal of Operational Research*, 113(2):469 – 499, 1999.

[116] Q. Wang, X. Sun, B. L. Golden, and J. Jia. Using artificial neural networks to solve the orienteering problem. *Annals of Operations Research*, 61:111–120.

[117] X. Wang, B. L. Golden, and E. A. Wasil. Using a genetic algorithm to solve the generalized orienteering problem. In *The Vehicle Routing Problem: Latest Advances and New Challenges*, volume 43 of *Operations Research/Computer Science Interfaces Series*, pages 263–274. Springer US, 2008.

[118] R. Xu and D. Wunsch. Survey of clustering algorithms. *Neural Networks, IEEE Transactions on*, 16(3):645 –678, may 2005.

[119] B. Zenker and B. Ludwig. Rose: assisting pedestrians to find preferred events and comfortable public transport connections. In *Proceedings of the 6th International Conference on Mobile Technology, Application & Systems*, Mobility '09, pages 16:1–16:5, New York, NY, USA, 2009. ACM.