Computers and Operations Research 88 (2017) 280–289

Contents lists available at ScienceDirect

Computers and Operations Research

journal homepage: www.elsevier.com/locate/cor

An integer programming approach for the time-dependent traveling salesman problem with time windows

Agustín Montero[a,c,*], Isabel Méndez-Díaz[a,d], Juan José Miranda-Bront[a,b,c,*]

[a] Departamento de Computación, FCEyN, Universidad de Buenos Aires, Pabellón I, Ciudad Universitaria, C1428EGA, CABA, Argentina
[b] Universidad Torcuato Di Tella, Av. Figueroa Alcorta 7350, C1428BCW, CABA, Argentina
[c] Consejo Nacional de Investigaciones Científicas y Técnicas, Argentina
[d] CONICET-Universidad de Buenos Aires. Instituto de Investigación en Ciencias de la Computación (ICC). Buenos Aires, Argentina

ARTICLE INFO

Article history:
Received 15 June 2016
Revised 9 May 2017
Accepted 26 June 2017
Available online 4 July 2017

Keywords:
Time-dependent TSP
Time windows
Integer linear programming
Branch-and-Cut

ABSTRACT

Congestion in large cities and populated areas is one of the major challenges in urban logistics, and should be addressed at different planning and operational levels. The Time Dependent Travelling Salesman Problem (TDTSP) is a generalization of the well known Traveling Salesman Problem (TSP) where the travel times are not assumed to be constant along the day. The motivation to consider the time dependency factor is that it enables to have better approximations to many problems arising from practice. In this paper, we consider the Time-Dependent Traveling Salesman Problem with Time Windows (TDTSP-TW), where the time dependence is captured by considering variable average travel speeds. We propose an Integer Linear Programming model for the problem and develop an exact algorithm, which is compared on benchmark instances with another approach from the related literature. The results show that the approach is able to solve instances with up to 40 customers.

1. Introduction and literature review

The use of the transportation infrastructure and the impact of congestion have become one of the major issues in city planning and urban logistics. Projections indicate that this effect is expected to worsen in the medium and long term. Therefore, the current traffic situation as well as the projected traffic scenarios are likely to have, if not addressed correctly, a negative impact from a social, economic and a environmental standpoint.

Most of the research related to the Vehicle Routing Problem (VRP) considers that the travel time between two locations are fixed along the time horizon. An updated description of variants and methods can be found in Toth and Vigo (2014). In the last few years, there has been a trend to enrich these models by incorporating more complex travel time functions, aiming to obtain solutions that are closer to real-world operations. These models are particularly useful for urban logistics, where congestion may produce significant variations in travel times during different moments of the day. For instance, last mile deliveries, which are estimated to account of an important percentage of the total delivery costs, could be significantly improved by more realistic approaches, translating into a better service and a more efficient use of the resources.

Time-Dependent Vehicle Routing Problems (TDVRPs) is the name given to a family of problems that generalize the classical VRPs by considering more complex travel time and cost functions, generally by incorporating some variability depending on the moment of the day an arc is traversed. A recent survey on TDVRP variants is available in Gendreau et al. (2015), covering exact and heuristic algorithms. Commercial applications including traffic information are, to the best of our knowledge, quite scarce in practice. Google Maps and Waze provide detailed directions including traffic information, but limited to the *quickest path* between two points.

One of the variants that received some attention in the last decade is the so-called *Time-Dependent Traveling Salesman Problem* (TDTSP), which considers only one vehicle with infinite capacity. Therefore, the problem reduces to find a Hamiltonian tour at minimum total cost, while accounting for some particular travel time function. In this context, the name TDTSP has been used to refer to problems considering different travel time functions. The simplest generalization is the variant of the TDTSP considered in Picard and Queyranne (1978), which has applications within scheduling contexts and generalizes the well-known Traveling Deliveryman Problem (see, e.g., Fischetti et al., 1993; Lucena, 1990; Méndez-Díaz et al., 2008). The improvement with respect to the traditional TSP

* Corresponding authors.
E-mail addresses: aimontero@dc.uba.ar (A. Montero), imendez@dc.uba.ar (I. Méndez-Díaz), jmiranda@dc.uba.ar, jmiranda@utdt.edu (J.J. Miranda-Bront).

http://dx.doi.org/10.1016/j.cor.2017.06.026
0305-0548/

is that it considers that the travel cost function between two cities depends not only on the distance, but also on the position of the arc in the tour. Exact approaches for this problem can be found in Gouveia and Voß (1995), Abeledo et al. (2012), Miranda-Bront et al. (2013) and Godinho et al. (2014), where instances with up to 100 customers can be solved within reasonable computing times. To the best of our knowledge, the best exact approach in the literature is the Branch-Price and Cut (BPC) proposed in Abeledo et al. (2012).

A different approach is proposed in Malandraki and Daskin (1992), where the travel time between two cities depends on the moment of the day in which the arc is traversed. For this purpose, the authors proposed partitioning the time horizon in different *time periods* and the travel time is defined as a step function over these periods. This allows the model to capture, at least partially, the effect of congestion in different moments of the day. Exact approaches for variants of this problem with minor modifications (i.e., different objective functions and operational constraints) can be found in Stecco et al. (2008), Albiach et al. (2008), Méndez-Dıaz et al. (2011), Miranda-Bront (2012) and Melgarejo et al. (2015). Regarding applications, Furini et al. (2015) formulate a TDTSP to model an aircraft sequencing problem.

One of the major objections to the above model is that the travel times do not necessarily satisfy the FIFO (*First In, First Out*) condition, which is usually a desired property for the network from a vehicle routing perspective. To overcome this difficulty, Ichoua et al. (2003) builds upon the model proposed by Hill and Benton (1992) and propose a similar setting as in Malandraki and Daskin (1992) but for the average travel speed within each period. The resulting travel times are computed based on the departure instant from the origin customer, assuming that the distance of the trip is traveled at the average speed and that when crossing the boundaries between consecutive time periods, the average speed is adjusted. This model is able to capture the time dependency while satisfying the FIFO condition on the travel times.

The model proposed by Ichoua et al. (2003) has recently caught the attention of many researchers. Cordeau et al. (2012) tackles the TDTSP with the objective to minimize the makespan. They study some of the properties of the travel time function, including the computation of a lower bound obtained by solving an auxiliary TSP with constant travel times. They show that the bound is tight depending on some parameters related to the travel speed definitions and that, under some particular settings, the solution of the auxiliary TSP is indeed optimal. They also propose a Branch and Cut (BC) algorithm and are able to solve instances with up to 40 vertices. Ghiani and Guerriero (2014) further exploit some of the properties of the travel time function, and study its generality. In a follow up paper, Arigliano et al. (2015) extend the ideas proposed in Cordeau et al. (2012) to the TDTSP with Time Windows (TDTSP-TW). However, the results obtained are not as good as for the TDTSP. A BC algorithm is evaluated on instances with up to 40 clients, obtaining mixed results.

Multi-vehicle versions of the TDVRP have also been tackled by exact algorithms that consider the model proposed in Ichoua et al. (2003). Dabia et al. (2013) consider the TDVRP with time windows with the objective of minimizing the overall duration instead of the makespan. They propose a set partitioning model and develop a Branch and Price (BP) algorithm, where the column generation subproblem is tackled by means of a tailored labeling algorithm. The authors conduct experiments on instances of different sizes, showing that the approach is able to solve consistently instances with 25 vertices and some of the ones having 100 customers. Related to this research is the work by Sun et al. (2015), where a profitable TDTSP with time windows and precedence constraints are considered. Indeed, this particular variant arises as the column generation subproblem of a TDVRP with time-windows and precedence constraints. They propose an Integer Linear Programming (ILP) model for the problem, which is not studied in detail due to its performance in standard commercial solvers, and resort to dynamic programming techniques.

In this paper we tackle the version TDTSP-TW considered also in Arigliano et al. (2015). The contribution of this paper is two fold. Firstly, we propose an alternative approach for the TDTSP-TW that builds on the ILP formulation proposed by Sun et al. (2015). This model is used to develop an exact algorithm following a Branch-and-Cut scheme (BC. We included several initial heuristics, preprocessing rules and incorporate several families of valid inequalities, which are used as cuts, in order to improve the overall computational times of the algorithm. Secondly, we evaluate our approach on benchmark instances and compare our results with two sets of instances proposed by Arigliano et al. (2015). To the best of our knowledge, this is the first comparison of two exact approaches for the TDTSP-TW, establishing a baseline for future approaches for the TDTSP-TW and related problems and opening the discussion regarding formulations, algorithms and benchmark instances.

The rest of the paper is organized as follows. In Section 2 we introduce the notation used throughout the paper and provide the detailed definition of the problem. In Section 3 we describe with more details some of the developments proposed for the TDTSP and TDTSP-TW with time-dependent travel speeds, and present a new formulation for the TDTSP-TW using the ideas proposed in Sun et al. (2015). Section 4 describes the details of the BC algorithm based on this formulation. Computational results are shown in Section 5 and finally we conclude and state some future research directions in Section 6.

2. Problem definition

In this section we present the definitions and the basic properties of the TDTSP-TW with the travel time model proposed in Ichoua et al. (2003).

For the definition of the network, consider a digraph $D = (V, A)$, with $V = \{0, 1, \dots, n, n+1\}$ the set of vertices and A the set of arcs. Vertices 0 and $n+1$ represent the depot, for which we do not consider the incoming and outgoing arcs, respectively. We denote by $V_0 = V \backslash \{n+1\}$ and $V_{n+1} = V \backslash \{0\}$. There is a time horizon $[0, T]$ (typically a single day) in which vehicles move along the network. For each vertex $i \in V$, we denote by p_i to its processing time and $W_i = [r_i, d_i]$ the corresponding (hard) time window, where r_i and d_i are the release and deadline times, respectively. In particular, we set $W_0 = W_{n+1} = [0, T]$. We allow waiting times when arriving at a vertex before its release time r_i, but the vehicle must wait until r_i before starting to process it. In addition, each arc $(i, j) \in A$ has an associated travel distance L_{ij}. Without loss of generality, $d_i + p_i \leq T$ for all $i \in V$. In addition, to simplify the notation in the manuscript, we slightly modify the standard definition and assume that $p_i = 0$ for $i \in V$. However, the models and formulae present in this paper can be easily adapted to consider processing times.

The time dependency is modeled as follows. The planning horizon is partitioned into M intervals $[T_h, T_{h+1}]$, $h = 0, \dots, M-1$. We assume that, for each arc $(i, j) \in A$, the average value of the travel speed during the time interval $[T_h, T_{h+1}]$, denoted by v_{ijh}, for $h = 0, \dots, M-1$, is known. This partition with its corresponding travel speeds are referred as *speed profiles*. It is important to remark that the speed profiles may differ among arcs. Based on this definition, the main idea behind the speed model is to compute the travel times using the information of the distance to be traveled, i.e. L_{ij}, combined with the travel speeds v_{ijh} defined for the arc. However, it is not assumed that the travel speed remains fixed during the trip and it may change whenever the boundaries of an interval are crossed. We denote by $\tau_{ij}(t)$ to the time-dependent travel time value on arc $(i, j) \in A$ if departing from i at time $t \in [0, T]$, and it