European Journal of Operational Research 208 (2011) 253–262

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Production, Manufacturing and Logistics

Heuristic algorithms for the 2-period balanced Travelling Salesman Problem in Euclidean graphs

Tatiana Bassetto, Francesco Mason *

Department of Applied Mathematics, University of Venice, Venice, Italy

ARTICLE INFO

Article history:
Received 27 January 2009
Accepted 4 September 2010
Available online 9 September 2010

Keywords:
Combinatorial optimization
Period routing problem
Period Travelling Salesman Problem
Logistics
Heuristic algorithms

ABSTRACT

In the 2-period Travelling Salesman Problem some nodes, called double nodes, are visited in both of two periods while the remaining ones, called single nodes, are visited in either one of the periods. In this paper we study the case in which a balance constraint is also introduced. We require that the difference between the number of visited nodes in the two periods must be below a fixed threshold. Moreover, we suppose that distances between nodes are Euclidean. The problem is NP-hard, and exact methods, now available, appear inadequate. Here, we propose three heuristics. Computational experiences and a comparison between the algorithms are also given.

1. Introduction

In the organization of picking up orders for commercial firms, it often happens that an agent has to visit his customers periodically but at different times depending on the number and the frequency of the orders. The simplest, but most frequent, case is when the set V of customers can be divided into two sets over a period of two days: customers visited daily (set D) and customers visited only once in the period (set S). So, two tours must be constructed in such a way that both contain all the elements in D, while the customer in S are partitioned between them.

Moreover, we require the agent to work every day for almost the same amount of time: in this way, we also impose some balancing constraints on the two tours.

In particular, in this study we require that the difference between the number of customers visited in each day, be below a given threshold.

In this paper we consider in particular the case in which the two tours either contain the same number of customers or differ by one: then, we generalise to the case in which the difference must be below a threshold more than 1.

The objective is to minimize the total distance travelled in the two days.

The (non-balanced) 2-period TSP may be viewed as a particular case of the Period Travelling Salesman Problem (PTSP) (see [10]). In PTSP every customer must be visited a specified number of times during p days. For each customer a set of feasible (allowable) combinations of visit days is also given. The aim is to build p routes (one route for each day) in order to minimize the total distance covered. Both PTSP and 2-period TSP are NP-hard problems.

The literature on periodic routing problems is not very extensive.

Christofides and Beasley (1984) in their seminal paper [11] introduce the PVRP (Period Vehicle Routing Problem), i.e. a problem with weighted nodes and a capacity constraint on the vehicles. They propose two heuristic algorithms which make use of the solution of other NP-hard problems: the Period Median Problem and the Travelling Salesman Problem.

Other heuristic approaches were subsequently proposed in 1992 by Paletta [22], and Chao et al. [10]. In 1997, Cordeau et al. [13] introduced a tabu search technique. In 2002 Paletta [23] presents a new heuristic algorithm for the PTSP, improved in Bertazzi et al. [8] in 2004. Other works are involved with PVRP ([14,7]), with asymmetric PTSP [24], with dynamic versions of multiperiod TSP ([2–4]) and multiperiod VRP ([1,16]). Periodic arc routing problems are also considered in literature: see for example [15].

In PTSP there are two interrelated sub-problems: an assignment problem, because for each customer a feasible combination of visit days must be chosen, and then a routing problem for each day. In the heuristic approaches, these sub-problems are often solved subsequently.

The 2-period TSP was introduced in 1997 by Butler et al. [9]: it is essentially a PTSP in which the planning period is $p = 2$ days. Two tours T^{1*} and T^{2*} must be built. The authors solve exactly a

* Corresponding author.
E-mail addresses: tbassetto@unive.it (T. Bassetto), fmason@unive.it (F. Mason).

0377-2217/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.ejor.2010.09.008

particular case study of 42 nodes applied to milk collection in Ireland. They suggest an exact branch and cut approach. However, this algorithm requires decisions 'on line' about the introduction of constraints. They do not give a fully automatic procedure.

Balancing constraints are never explicitly considered (both in PTSP and 2-period TSP). In [10], in the assignment phase, the Authors try to get a uniform distribution of customers to the visit days. However in the improvement steps, the solution can become unbalanced. In [11], a constraint on the maximum length of each route is given. In [9] the solution to the 42 nodes case consists of two tours, having respectively 24 and 31 nodes (including the depot: the nodes to be visited daily are 12).

In our experience, branch and cut takes too long a time also for moderate size instances. This way, approximate algorithms are useful.

In this paper we propose three heuristics suitable for Euclidean graphs.

The paper is organised as follows. In Section 2 the 2-period balanced TSP is formulated as an integer programming problem. In Section 3, three heuristic algorithms are described. In Section 4 we discuss some possible improvements to the solution obtained through the heuristics. Then, in Section 5, computational experiments and comparisons are provided.

2. An integer programming model for the 2B-TSP

Let $G = (V,E)$ be a complete graph of n nodes ($n > 1$) without loops. Let c_{ij} be the weight of the edge (i,j). The set V is partitioned into two (disjoint) subsets:

– the set of single-nodes $S = \{s_1, s_2, \ldots, s_k\}$, i.e. the ones to visit once over two days;
– the set of double-nodes $D = \{d_1, d_2, \ldots, d_h\}$, i.e. the ones to visit every day.

The depot is included in the set D of double-nodes. $D \neq \emptyset$.

We denote a node belonging to the set or tour X as X-node. Given a set A, we denote with $|A|$ its cardinality and with $\lfloor x \rfloor$ the floor function.

We want to build two tours, T^1 and T^2, which satisfy a balance constraint and minimize the total distance travelled. Both tours visit all the D-nodes, while every single-node, i.e. every S-node, must be inserted only in one of the two tours, T^1 or T^2.

Then, in every feasible solution, S is partitioned into two subsets, S^1 and S^2, the first made up of nodes visited on the first (or odd) day and the second on the second (or even) day.

T^1-nodes form the set $D \cup S^1$, while T^2-nodes are $D \cup S^2$.

Let $k_1 = |S^1|$ and $k_2 = |S^2|$, so that $k_1 + k_2 = k$.

Balance constraints can be formulated in several ways: we use a parameter g^* which represents the maximum allowed difference between the number of customers to be visited on the first and on the second day. Thus, we have:

$$|k_1 - k_2| \leqslant g^*.$$

Letting $g^* = 1$, then in a feasible solution the number of customers visited must be equal in the two days (if k is even), or it can differ by one unit (when k is odd).

The 2-period Balanced TSP can be formulated as an integer linear programming problem. Let be:

– $x_{ijq} = 1$, if and only if the customer j is visited immediately after i on the q-day and 0 otherwise, ($q = 1$ or 2);
– $y_{iq} = 1$, iff the customer i is visited on the q-day and 0 otherwise (for every node $s_i \in S$ and $q = 1$ or 2).

Then, the 2B-TSP is:

$$\text{Min} \quad \sum_{(i,j)\in E} \sum_{q=1}^{2} c_{ij}\, x_{ijq} \tag{1}$$

$$\text{s.t.} \quad \sum_{j\in V} x_{ijq} = 1 \quad \forall i \in D; \forall q, \tag{2}$$

$$\sum_{j\in V} x_{jiq} = 1 \quad \forall i \in D; \forall q, \tag{3}$$

$$y_{i1} + y_{i2} = 1 \quad \forall i \in S, \tag{4}$$

$$\sum_{j\in V} x_{ij1} = \sum_{j\in V} x_{ji1} = y_{i1} \quad \forall i \in S, \tag{5}$$

$$\sum_{j\in V} x_{ij2} = \sum_{j\in V} x_{ji2} = y_{i2} \quad \forall i \in S, \tag{6}$$

$$\sum_{i,j\in Z} x_{ijq} \leqslant |Z| - 1 \quad Z \subset S \cup D; \forall q, \tag{7}$$

$$\left| \sum_{i\in S} y_{i1} - \sum_{i\in S} y_{i2} \right| \leqslant g^*, \tag{8}$$

$$y_{iq} \in \{0,1\} \quad \forall i \in S; \forall q, \tag{9}$$

$$x_{ijq} \in \{0,1\} \quad \forall (i,j) \in E; \forall q. \tag{10}$$

The objective function minimizes the total costs.

Constraints (2) and (3) impose that all double nodes $i \in D$ be visited every day, both for odd ($q = 1$) and for even ($q = 2$) days.

Constraints (4) guarantee that every single-node $i \in S$ is visited only once, either on even or on odd days; constraints (5) and (6) compel the existence of only one outside edge and one inside edge for any node $i \in S$.

Constraints (7) are classical sub-tour elimination constraints of the TSP, Z being a proper subset of nodes of the graph G (see [17,18]).

Inequalities (8) represent the 'balance constraints': they impose an upper bound to the difference between the number of nodes (customers) that can be visited by a vehicle in each day.

There are two extreme cases: when D contains only the depot, the problem becomes a VRP in which customers have unit demand and there are two vehicles, each of them associated with a day, with capacity $\lfloor (k + g^*)/2 \rfloor$; on the other hand, if $S = \emptyset$ (so that every node has to be visited every day) the problem becomes a TSP.

As mentioned above, obviously, the 2-period balanced TSP is a NP-hard problem (see [11,13,18]).

3. Three heuristic techniques for 2B-TSP in Euclidean graphs

3.1. Premises

The core problem in a two-step heuristic for the 2-period balanced TSP is how to partition optimally the customers in S into the two sub-sets S^1 and S^2, to be attributed, respectively, to the two days of the period. This appears to be the crucial point, because software now available (see e.g. Concorde [5]) allows us to solve the subsequent Travelling Salesman Problems, in $D \cup S^1$ and $D \cup S^2$, in an exact way, at least for instances with some hundreds of nodes. This suggests the use of exact procedures which solve TSP as a step in the achievement of an approximate solution of the 2-balanced period TSP.

All the three algorithms we present start by using a (not feasible) tour, called General Tour, which visits all the nodes (both double and single ones).

The algorithm A1 builds the two tours which form the solution, firstly (odd days) deleting from GT a certain number of consecutive single nodes and then (even days) deleting the complementary (single node) set: in both the remaining sets of nodes a new TSP is then solved.