ISTITUTO DI ANALISI DEI SISTEMI ED INFORMATICA
"Antonio Ruberti"
CONSIGLIO NAZIONALE DELLE RICERCHE

C. Gentile, S. Mancini, G. Stecca, M. Suanno

SOLVING THE ORIENTEERING PROBLEM WITH TIME WINDOWS IN BIG EVENT MANAGEMENT WITH A MATHEURISTIC APPROACH

R. 04, 2017

Claudio Gentile – Istitute of Systems Analysis and Computer Science "A. Ruberti", CNR — Via dei Taurini 19, 00185, Roma, Italy (gentile@iasi.cnr.it).

Simona Mancini – University of Cagliari — Via Università 40, 09124 Cagliari, Italy (simona.mancini@unicag.it).

Giuseppe Stecca – Istitute of Systems Analysis and Computer Science "A. Ruberti", CNR — Via dei Taurini 19, 00185, Roma, Italy (giuseppe.stecca@iasi.cnr.it).

Maria Suanno – "Tor Vergata" University of Rome — Via Orazio Raimondo, 18, 00173 Roma, Italy (mariasuanno91@gmail.com).

ISSN: 1128–3378

Collana dei Rapporti dell'Istituto di Analisi dei Sistemi ed Informatica "Antonio Ruberti", CNR

via dei Taurini 19, 00185 ROMA, Italy

tel. ++39-06-49937101/02
fax ++39-06-49937106
email: iasi@iasi.cnr.it
URL: http://www.iasi.cnr.it

Abstract

In this work, we address, from a user perspective, the problem to define optimal itineraries for visiting scheduled events in an urban environment. The problem is modeled as an orienteering problem with time windows. Real-case instances are difficult to solve by means of commercial solvers. We propose a matheuristic approach which is compared against the exact solver by using a real-case instance derived by a big Italian furniture fair. Test results show clear effectiveness of the proposed approach.

Key words: orienteering problem; matheuristics; big events management; tourist planning.

1. Introduction

Intelligent mobility entails the integration of key enabling technologies, such as sensor networks and data mining with innovative organization model and decision support systems. In these settings, we study an application of intelligent mobility for supporting the visit of organized events. In the analyzed scenario, a big event, such a fair, is composed of micro-events scheduled during a day or a couple of days. The users, based on their preferences aim to visit most of the events "maximising" their utility. In the following we describe, the model, the solution approach, the results, and draw some conclusions.

2. Problem formulation

The problem is modeled as an orienteering problem with time windows [1] defined on a digraph $G = \{0 \cup N \cup n+1, A\}$ where nodes $1, 2, \ldots, n$ represent point of interests where the events are organized. Nodes 0 and $n+1$ coincide and represent the departure / arrival point for the user (e.g. his hotel).
Parameters:

- p_i profit or utility associated to the node (point of interest) i, $p_0 = p_{n+1} = 0$
- t_{ij} time to travel arc i, j
- s_i average time required to visit node i
- $[e_i, l_i]$ time window related to the duration of the event in node i
- T_{MAX} maximum arrival time at node $n+1$. The time window for the nodes 0 and $n+1$ is then $[0, T_{MAX}]$
- M a big constant

Variables:

- x_{ij} binary variable equal to 1 if arc i, j is travelled, 0 otherwise
- y_i binary variable equal to 1 if node i is visited, 0 otherwise
- a_i arrival time at node i

The model:

$$\max Z = \sum_{i \in N} p_i y_i \tag{1}$$

s.t.

$$\sum_{j \in N \cup n+1} x_{ij} = y_i \qquad i \in N \tag{2}$$

$$\sum_{i \in N \cup 0} x_{ij} - \sum_{i \in N \cup n+1} x_{ji} = 0 \qquad i \in N \tag{3}$$

$$\sum_{j \in N} x_0 j = \sum_{j \in N} x_{jn+1} = y_0 = y_{n+1} = 1 \tag{4}$$

$$a_i + t_{ij} + s_i - a_j \leq (1 - x_{ij})\mathrm{M} \qquad (i,j) \in A \tag{5}$$

$$e_i y_i \leq a_i \leq l_i y_i \qquad i \in N \cup 0 \cup n+1 \tag{6}$$

$$x_{ij} \in \{0,1\},\ y_i \in \{0,1\},\ a_i \in \mathbb{R}^+ \qquad i \in N,\ (i,j) \in A \tag{7}$$

Formula (1) represents the objective function. Constraint set (2) connects arc with node variables. (4) ensures flow conservation, while constraint set (4) imposes the departure and the arrival of the tour. Formulas (5) and (6) define the time window constraints, while (7) specifies the variables domains.

3. Solution approach

The aim of the proposed solution approach is to develop a tool able to be responsive and to provide near-optimal tours in short time. A similar approach has been applied in a real case of tourist cruise planning [2]. Following this scope, a matheuristic approach able to exploit the model in order to explore large neighborhoods has been developed. This approach is particularly suitable in settings where the set of potential visited points is big and only a limited subset can be visited during the available time. Following a Large Neighboorhood Search matheuristic scheme, the proposed approach tries to iteratively partially destroy and reconstruct a feasible solution, exploring the consequently created large neighborhood. In the case of the orienteering problem with large set of nodes, destroy operator can be developed around the nodes which can be forced to be out of the solution, in the solution, or be free to be selected or not.
Our proposed matheuristic models this choice by introducing two additional binary parameters to the model, u_i, and v_i, respectively, which, during the algorithm iterations, are set to the following values:

- $u_i = 0 \wedge v_i = 0 \rightarrow$ node i forced out of the solution
- $u_i = 1 \wedge v_i = 1 \rightarrow$ node i forced to be in solution
- $u_i = 0 \wedge v_i = 1 \rightarrow$ node i left free

In the model the following constraints are added:

$$y_i \geq u_i \qquad i \in N \tag{8}$$

$$y_i \leq v_i \qquad i \in N \tag{9}$$

The algorithm contains the following steps:

1. Find a feasible solution s^0 by running with GUROBI^{TM} the orienteering model with a short timelimit $tlim1$
2. Create the sets B, A, D such that:
   - S is the set of nodes $i \in N$ for which $y_i = 1$ in the current solution
   - B is the set of nodes belonging to N but not the current solution, $B = N \setminus S$
   - A a subset of S, $B \subset S$. Nodes are selected from S and inserted in A following a pseudo-random procedure detailed in the following;
   - $D = S \setminus A$
3. Nodes in A are forced to be in solution and all the others are leaved free or forced to be out of the solution.
4. The modified model is run with a timelimit $tlim2$ in order to find a feasible solution
5. The procedure is iterated for nit iterations.

The pseudo-random procedure is built using a ratio r_i for each node i equal to $r_i = \frac{p_i^2}{\Delta_{i-1,i+1}}$, where $\Delta_{i-1,i+1}$ is the traveling time added to the tour required to visit node i. This means that the probability to select a node is equal to r_i divided by the sum of all r_i. The ratio is inspired by the work of [3]. The use of two parameters u_i and v_i for each node allowed us to implement a more general matheuristic in which we have the availability to "cut off" some nodes or force some other nodes "in solution".

4. Results

4.1. Test Instance

The mathematical formulation against standard benchmark test instances (`https://www.mech.kuleuven.be/en/cib/op#section-6`). Computational tests have been carried out on the aforementioned test instances and on a real case instance difficult to solve exactly. The current version of the algorithm is not competitive in the tests against the best results found in literature but demonstrated good performances in solving the real test case. In particular, the real case instance is made up of 30 points of interest gathered by the Milan Furniture fair with the time windows of the events taken by the schedule of the April 4, 2017. Traveling times have been computed through Google Map API; service times have been roughly estimated by using presence data taken by the MIE research project; Profits of the points of interest have been set from 1 to 10 and its value is meant to be proportional to the positive feedback of the users. Figure 1 show the position of the points of interest.

4.2. Settings and results

The algorithm has been coded with Python 3 with the model developed with the GUROBI - Python Interface. The computation machine is a an Intel Core^{TM} model i3-5005U CPU with 2.00 GHZ, RAM 4 GB. In these settings, the exact model produced a Lower Bound of the problem of value 55 in the imposed time limit of 8 hours with GAP of 151%. Several experiments for tuning the parameters of the matheuristics have been performed. The experiments investigated

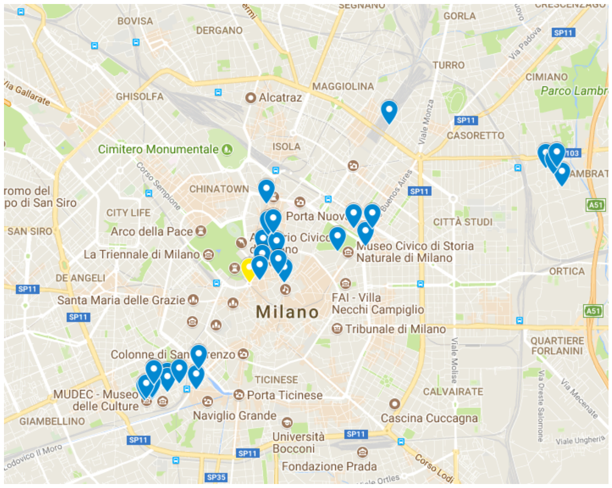

Figure 1: Map of the scheduled events to be visited

the different composition of the sets A, B, and D; different random procedures used to select nodes; different size of set A. The most promising settings found to guarantee short execution time and good solutions for the real case instance are:

- $A : \{u_i = 1, v_i = 1\}$, $D : \{u_i = 0,\ v_i = 0\}$, $B : \{u_i = 0, v_i = 1\}$
- $tilim1 = 3\text{s}$, $tilim2 = 10\text{s}$
- nit=10 (number of macro iterations of the matheuristic; at each microiteration thousands of micromoves are implicitly evaluated)
- $r_i = \dfrac{p_i^2}{\Delta_{i-1,i+1}}$ h
- size of A equal to 80% of the size of S

The algorithm has been run 10 times for each setting and in the described setting it is was able to find a best solution of 58 with a mean in the 10 runs of 52 and a minimum value of 46 (equal to the starting feasible solution).

5. Conclusion

In this work we have presented preliminary results of a matheuristic algorithm used to solve a real case instance of a problem modeled as orienteering with time windows. The problem considers the visit of point of interests and it is designed to be used in an intelligent mobility scenario where the user can optimally plan his visit to an organized big event. The preliminary results showed promising results for the algorithm. Future works aimed at improving the algorithm performances by using more effective diversification / intensification moves.

6. Acknowledgments

This work has been partially supported by CLUSTER progam of the Ministry of Instruction, University, and Research (MIUR) through the project MIE - Mobilità Intelligiente Ecosostenibile [grant number CTN01_00034_59412].

References

[1] M.G. Kantor and M.B. Rosenwein The orienteering problem with time windows. *Journal of the Operational Research Society*, 629–635, 1992.

[2] S. Mancini and G. Stecca A matheuristics approach to solve a liner network design problem. *Proceedings of the 18th free workshop on metaheuristics for a better world*, Rome, April 3–4, 2017.

[3] Aldy Gunuwan, Hoong Chuin Lau, Kun.S. Lu. An iterated local search algorithm for solving the orienteering problem with time windows. *In: European Conference on Evolutionary Computation in Combinatorial Optimization. Springer*, 61–73, 2015.