The Multi-color Traveling Salesman Problem

Emanuele Tresoldi[1], Juan José Salazar González[2], Roberto Wolfler Calvo[3]

[1] DI, Università degli Studi di Milano, Milano, Italy.
emanuele.tresoldi@unimi.it

[2] DMEIO, Universidad de La Laguna, Tenerife, Spain.

[3] LIPN, Université Paris 13, Villetaneuse, France.

Abstract

In this paper we investigate a new challenging variant of the classical Travelling Salesman Problem where the set of nodes is divided into clusters and a different color is associated with each cluster. The goal is to find a Minimum Cost Hamiltonian Cycle satisfying different separation constraints between nodes with the same color. We present a new effective mathematical formulation for the problem. Since it involves an exponential number of constraints we devise separations procedures to be used within a Branch-and-Cut framework. Promising preliminary results are obtained on a set of random instances.

Keywords : *TSP, Branch-and-Cut, Multistar inequalities.*

1 Problem Description

The Traveling Salesman Problem (TSP) does not require any introduction. It is, indeed, one of the most studied and influential problem in combinatorial optimization literature. There are thousands of papers and books eviscerating every aspect of the problem [1] and many different variants of the TSP with diverse objectives and requirements have been investigated in the literature [5]. In this paper we tackle a new challenging variant of the TSP called the Multi-color Traveling Salesman Problem (MCTSP).

The MCTSP is defined starting from an undirected graph $G = (V, E)$ and a set of colors C. The set of nodes V is partitioned in h clusters of nodes $C_1, C_2, \dots C_h$. Each cluster contains only nodes of the same color and for each color $h \in C$ two values α_h and β_h are given. The value α_h represents the minimum number of nodes not belonging to cluster C_h that must separate any pair of nodes of color h. While β_h is the maximum number of nodes not in C_h that can separate two consecutive nodes of color h. Finally, a cost c_e proportional to the length of the edges is associated with each edge $e \in E$.

We define the *Multi-color Travelling Salesman Tour* as a Hamiltonian tour such that each pair of consecutive nodes belonging to the same cluster C_h is separated by at least α_h nodes and by at most β_h nodes not belonging to cluster C_h (see Figure 1 for an example). The MCTSP requires to find the Multi-color Travelling Salesman Tour of minimal cost.

The MCTSP is an NP-hard problem since the TSP is polynomially reducible to MCTSP. Indeed, a TSP instance can be turned into a MCTSP instance by associating a different color h to each node and setting $\alpha_h = \beta_h = 1$ for all nodes.

In the remainder of this paper we present a Mixed Integer Linear Programming model for this problem, we describe a branch-and-cut procedure to solve it and we report preliminary results obtained on a set of random instances.

2 Mathematical Formulation

The model that we propose for the MCTSP requires the definition of only one set of binary variables: x_e $\forall e \in E$. They are equal to 1 if the edge e is used in the tour and are 0 otherwise.

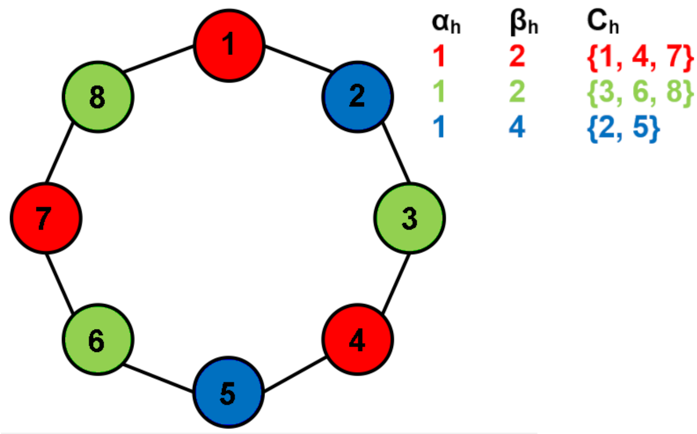

FIG. 1: Multi-color Travelling Salesman Tour

In the model we extensively use the notation $X(S : T)$, with $S \subseteq V$ and $T \subseteq V$, to identify the set of variables corresponding to edges with an endpoint in S and the other endpoint in T. Therefore, our mathematical formulation for the MCTSP reads as follows:

$$\max \sum_{e \in E} c_e x_e \tag{1}$$

$$s.t. \quad x(\{i\} : V \setminus \{i\}) = 2 \qquad \forall i \in V \tag{2}$$

$$x(S : V \setminus S) \geq 2 \qquad \forall S \subset V \tag{3}$$

$$x(S : S) + x(S : C_h) \leq |S| \qquad \forall h \in C, S \subset V \setminus C_h : |S| < \alpha_h \tag{4}$$

$$x(S : V \setminus S) \geq 2\frac{|S|}{\beta_h} \qquad \forall h \in C, S \subset V \setminus C_h \tag{5}$$

$$x_e \in \{0, 1\} \qquad \forall e \in E \tag{6}$$

The objective function (1) calls for the minimization of the total costs. Equalities (2) represents degree constraints stating that each node must be connected with two other nodes. Inequalities (3) are standard sub-tour elimination constraints. The requirement on the minimum number of nodes between two nodes of the same color is enforced by constraints (4). It states that, for each color h and for each subset of nodes $S \subset V \setminus C_h : |S| < \alpha_h$, either S represents a chain in the tour connected with C_h by at most one arc or S is not a chain. Finally, the maximum distance between two nodes of the same color is limited by inequalities (5). These constraints are satisfied if, for each color h and for each subset of nodes $S \subset V \setminus C_h$, either S is a chain not longer than β_h or S is not a chain.

3 Branch-and-Cut

Formulation (1- 6) includes an exponential number of constraints, in particular constraints (3), (4) and (5) and it cannot be directly solved using standard MIP solvers such as CPLEX, GUROBI, etc. For this reason we propose a branch-and-cut procedure to efficiently address the exponential number of constraints and to achieve proved optimal solutions for MCTSP.

Following a classical branch-and-cut paradigm we start considering a linear program (LP) including only the objective function (1), constraints (2), all sub-tour elimination constraints (3) with $|S| = 2$ and all constraints (4) with $|S| \leq 2$. At each iteration, we solve the current LP, we look for constraints (3), (4) or (5) that are violated by the optimal LP solution and we add them to the current LP. This procedure stops when no violated constraints are identified. If the solution found is not integer we branch and we repeat the procedure.

In order to find violated constraints we run three different separation procedures, one for each family of constraints (3), (4) and (5). Given the optimal solution $\bar{x}$ of LP for the separation of sub-tour elimination constraints (3) we relied on the well known max-flow-min-cut based procedure presented in the literature (see for instance [3]).

Constraints (4) can be interpreted as Enhanced Reverse Multistar inequalities rewriting them in the following way:

$$\alpha_h x(C_h : S) \leq (\alpha_h - 2)x(S : V \setminus (S \cup C_h)) + 2|S| \qquad \forall h \in C, S \subset V \setminus C_h : |S| < \alpha_h$$

Then, violated constraints can be found using the polynomial time algorithm presented in [4].

Finally, for the separation of constraints (5) a max-flow-min-cut procedure using modified graphs can be employed. In detail, for each color $h \in C$ the modified graph is composed by all nodes in $V \setminus C_h$ plus a super-node obtained by merging all nodes in C_h. Edges between nodes in C_h are ignored while the capacity associated with the other edges $e \in E$ is equal to $\bar{x_e}$. On these graphs we compute the max-flow-min-cut from any node to the super-node. A violated inequality is found if the flow is lower than $2\frac{|S|}{\beta}$ where S is the set of nodes not including C_h identified by the min-cut procedure.

TAB. 1: Preliminary Results

$\|V\|$	$\|C\|$	α	β	$GR\%$	TR	$GF\%$	TF	$BBNode$	Cut (3)	Cut (4)	Cut (5)	Opt
10	3	2.00	3.00	0.00	0.02	0.00	0.02	0.00	4.00	0.00	0.00	4
10	4	2.38	5.13	0.00	0.03	0.00	0.04	0.00	7.00	0.00	5.00	4
10	5	6.00	7.00	0.00	0.05	0.00	0.07	0.00	4.75	0.00	6.25	4
10	6	6.00	7.13	1.80	0.09	0.00	0.12	8.50	16.25	2.50	20.00	4
10	7	6.13	7.50	0.49	0.07	0.00	0.09	2.25	13.25	0.50	9.25	4
10	8	6.25	8.00	0.00	0.04	0.00	0.04	0.00	6.25	0.25	6.25	4
20	3	2.00	3.00	0.00	0.02	0.00	0.03	0.00	4.25	0.00	0.00	4
20	4	2.00	6.00	1.45	0.18	0.00	0.23	41.25	44.25	0.00	36.25	4
20	5	2.25	9.50	0.45	0.14	0.00	0.20	49.75	55.75	0.00	56.25	4
20	6	3.63	10.50	0.75	0.22	0.00	0.29	19.50	71.50	7.25	72.00	4
20	7	5.38	12.00	1.74	0.22	0.00	0.48	426.75	154.50	51.00	119.50	4
20	8	7.00	12.25	4.16	0.24	0.00	0.90	760.00	299.75	199.00	255.00	4
50	3	2.00	3.00	0.00	0.07	0.00	0.07	0.00	13.00	0.00	0.00	4
50	4	2.00	7.75	0.00	0.48	0.00	0.49	0.00	57.50	0.00	10.00	4
50	5	2.00	10.50	2.49	1.05	0.00	16.58	2276.75	618.75	0.00	403.25	4
50	6	2.38	15.25	8.86	1.25	2.05	2692.94	75453.00	13757.75	20.75	14404.00	3
50	7	3.00	16.75	7.72	1.15	1.43	2167.97	72040.50	7563.75	2303.00	6808.50	3
50	8	2.75	19.38	4.08	0.95	1.73	1807.79	34838.50	9561.75	3119.75	9322.00	3
75	3	2.00	3.00	0.00	0.40	0.00	0.40	0.00	30.00	0.00	0.00	4
75	4	2.00	7.13	0.85	3.91	0.00	60.96	2385.50	575.00	0.00	394.75	4
75	5	2.00	10.63	2.78	1.93	0.00	762.59	31806.00	8029.25	0.00	7547.25	3
75	6	2.00	16.38	3.76	3.03	0.00	401.73	14497.50	2741.25	0.00	1817.25	4
75	7	2.38	15.75	3.44	3.40	0.00	767.85	27668.00	4362.50	0.00	3543.75	4
75	8	2.63	19.38	5.27	3.75	0.50	2947.55	77308.75	10937.50	485.00	8817.00	3
100	3	2.00	3.00	0.00	1.00	0.00	1.00	0.00	72.75	0.00	0.00	4
100	4	2.00	8.25	0.57	5.23	0.00	78.80	1343.75	551.25	0.00	167.75	4
100	5	2.00	12.38	1.78	6.00	0.00	484.63	8336.75	1602.00	0.00	839.50	4
100	6	2.00	16.38	6.87	15.31	4.78	5161.20	48264.75	18090.25	0.00	12833.25	2
100	7	2.13	19.25	8.14	10.68	5.95	6327.64	65120.50	19076.00	0.00	15347.50	2
100	8	2.38	25.25	14.53	6.97	11.85	5418.75	52877.75	21574.50	0.00	14073.00	2

4 Computational Experiments

In order to assess the efficiency of our approach we generated a set of 120 random instances. In particular, this test-bed includes instances with a number of nodes $|V|$ equal to 10, 20, 50, 75 and 100 and a number of colors $|C|$ ranging from 3 to 8. For every combination of $|V|$ and $|C|$, four different instances were created. Those instances must be carefully crafted to guarantee the existence of a feasible solution. We proceeded as follows. Given set of $|V|$ randomly generated points in the $[1; 1000]^2$ square, a random TSP solution is built. Then colors are randomly assigned to the nodes according to this sequence ensuring that there is at least one node per color and that two adjacent nodes always receive different colors. Then, the values α_h and β_h are set consequently, with $\alpha_h \geq 1\ \forall h \in C$.

Our branch-and-cut procedure has been implemented in `C++` using CPLEX 12.10 Concert Technology. The max-flow-min-cut algorithm used in all three separation procedures is the Open-Source implementation of the Boykov-Kolmogorov MAXFLOW v.3.04 algorithm presented in [2]. All the experiments were carried out on a 64-bit Windows machine, equipped with processor Intel i7-6700K, 4.00 GHz and 16 GB of RAM. CPLEX is set to its default configuration. We set a time-limit of 7200 seconds on all instances.

Preliminary results obtained by our approach are shown in TAB. 1. For each pair $|V|$ and $|C|$ we report average values over all four instances of the same category. In detail, columns α and β shows the average values of α_h and β_h for each category of instances, columns $GR\%$ and TR contains the percentage MIP gap at root node (wrt the best known solution) and solution time (in seconds) at root node, while columns $GF\%$ and TF report the final gap and solution time. The total number of nodes explored in the branch-and-bound tree is shown in column $BBNode$. The average number of violated constraints found by each separation procedure is detailed in columns Cut (3), Cut (4) and Cut (5); note that these values do not include constraints (3) with $|S| = 2$ and (4) with $|S| = 2$ that are embedded in the model. Finally, the total number of instances per category solved to optimality is reported in column Opt.

Our branch-and-cut algorithm has been able to find the optimal solution on 110 instances over 120. In detail, all instances with five color or less are solved to proven optimality in, on average, less than 10 minutes. In particular, instances with three color are all solved at the root node. However, in ten instances our approach is not able to find the optimal solution and the average final gap is about 4% with two of the largest instances reporting final gaps greater than 10%. All these instances are characterize by more than five colors, larger than average values for both α_h and β_h and a gap at root node $\geq 5\%$.

In conclusion, preliminary experiments demonstrate the viability of our algorithm for the exact solution the MCTSP. Further experiments on instances taken from the literature are required to prove the general efficiency of the proposed approach.

References

[1] D. L. Applegate, R. E. Bixby, V. Chvatál, and W. J. Cook. *The Traveling Salesman Problem: A Computational Study.* Princeton University Press, 2006.

[2] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max- flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124–1137, 2004.

[3] Harlan Crowder and Manfred W. Padberg. Solving large-scale symmetric travelling salesman problems to optimality. *Management Science*, 26(5):495–509, 1980.

[4] Luis Gouveia and Juan-José Salazar-González. Polynomial-time separation of enhanced reverse multistar inequalities. *Operations Research Letters*, 41(3):294 – 297, 2013.

[5] G. Gutin and A. P. Punnen, editors. *The Traveling Salesman Problem and Its Variations.* Springer, Boston, MA, 2002.