The Multi-color Traveling Salesman Problem

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

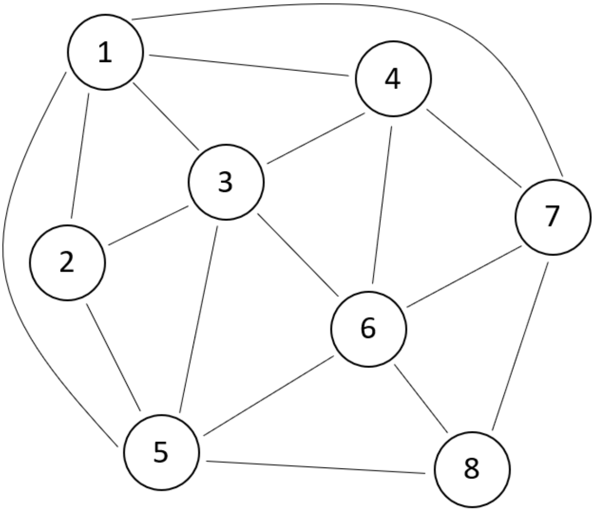
DI, Università degli Studi di Milano, Milano, Italy.

DMEIO, Universidad de La Laguna, Tenerife, Spain.

LIPN, Université Paris 13, Villetaneuse, France.

Table of Contents

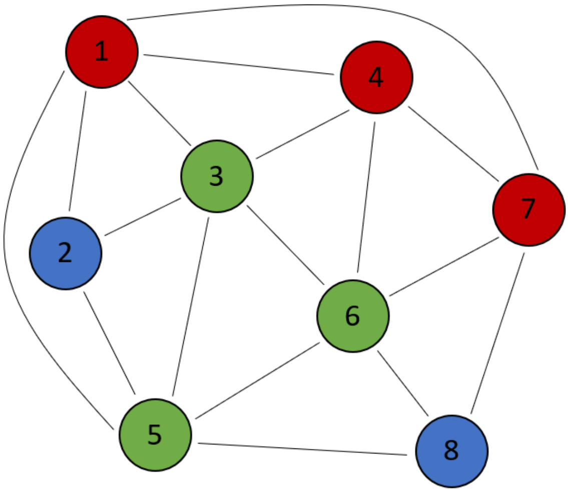
The Multi-color Traveling Salesman Problem



MCTSP is an extension of **TSP**.

The Multi-color Traveling Salesman Problem

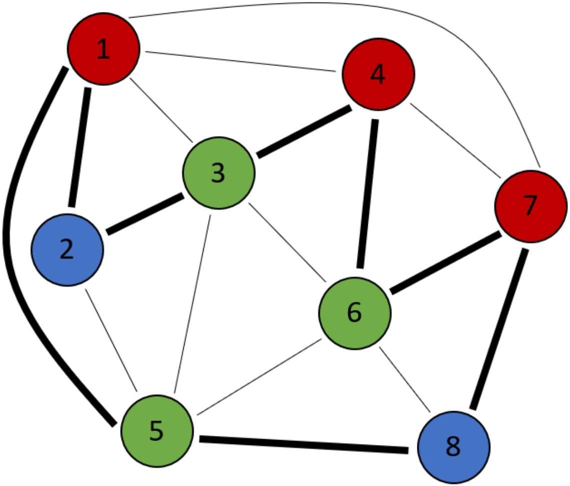
Color	Distance Min	Distance Max
Red	1	2
Green	1	2
Blue	2	4



Vertices divided in **clusters/colors** with min/max “distance” requirements.

The Multi-color Traveling Salesman Problem

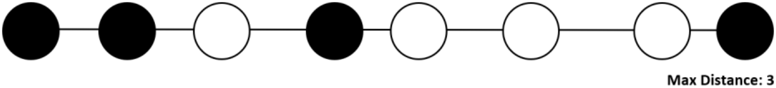
Color	Distance Min	Distance Max
Red	1	2
Green	1	2
Blue	2	4



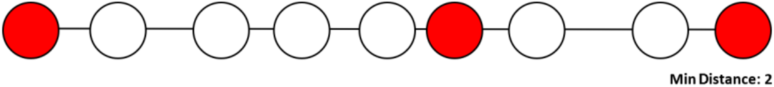
Find cheapest **Hamiltonian tour** satisfying min/max “distance” constraints.

Related Works

- **Black and White Travelling Salesman Problem**: the number of white vertices as well as the length of the tour between two consecutive black vertices are bounded above.



- **Travelling Salesman Problem with Separation Requirements**: a subset of vertices requiring minimum separation constraints is present.



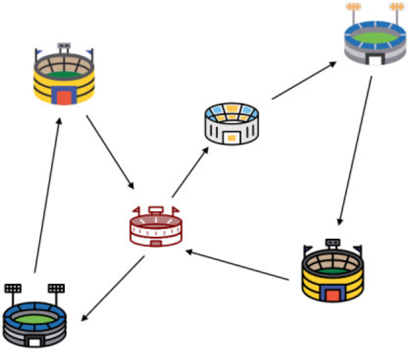
- *Travelling Salesman Problem with Time Windows*.
- *Travelling Salesman Problem with Precedence Constraints*.

Real World Applications

Overnight Security Service Problem (Wolfer Calvo and Cordone 2003): design routes respecting minimum and maximum time intervals between *two consecutive visits of the same location* (vertices having the same colour).

Real World Applications

Tournament Travelling Problem (Easton et al. 2001):
Major League Baseball teams go on *road trips* visiting a number of opposing teams before returning home.
It is required to balance the **travel time** and the **home/away pattern**.



The sequence of matches of a team is a solution of the MCTSP.

Mathematical Formulation - 1

Data

- Undirected graph $G = (V, E)$.
- C: set of colors.
- $C_h \subseteq V$ set of vertices with color h, $\forall h \in C$.
- t_e: cost for traversing edge $e \in E$.
- α_h: minimum number of vertices between two vertices of color h.
- β_h: maximum number of vertices between two vertices of color h.

Variables

- x_e $\forall e \in E$: 1 if the edge e is traversed in the tour and are 0 otherwise.

Mathematical Formulation - 2

$$\min \sum_{e \in E} t_e x_e \quad (1)$$

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

$$x(S : V \setminus S) \geq 2 \qquad \forall S \subset V \quad (3)$$

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

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

$$x_e \in \{0, 1\} \qquad \forall e \in E \quad (6)$$

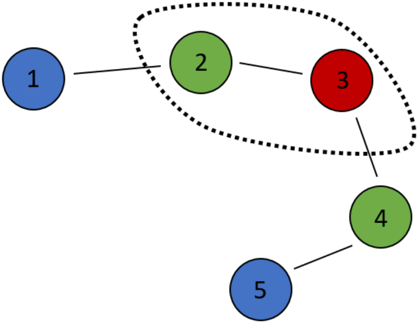
Notation: $x(S : T)$, with $S \subseteq V$ and $T \subseteq V$, is the set of variables corresponding to edges with an endpoint in S and the other endpoint in T.

Mathematical Formulation - 3

Min distance between two vertices of the same color.

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

Satisfied if for each color h and for each subset $S \subset V \setminus C_h : |S| < \alpha_h$:
- either S is a chain in the tour connected with C_h by at most one edge
- or S is not a chain.



$\alpha_h = 3$

$C_h = \{1, 5\}$ (*Blue*)

$S = \{2, 3\}$

$x(S:S) = 1$

$x(S:C_h) = 1$

Constraint: $1 + 1 \leq 2$

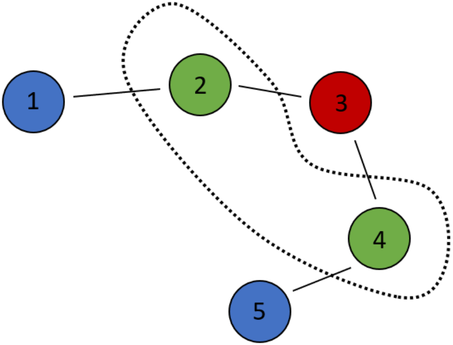
OK!

Mathematical Formulation - 3

Min distance between two vertices of the same color.

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

Satisfied if for each color h and for each subset $S \subset V \setminus C_h : |S| < \alpha_h$:
- either S is a chain in the tour connected with C_h by at most one edge
- or S is not a chain.



$\alpha_h = 3$

$C_h = \{1, 5\}$ (Blue)

$S = \{2, 4\}$

$x(S:S) = 0$

$x(S:C_h) = 2$

Constraint: $0 + 2 \leq 2$

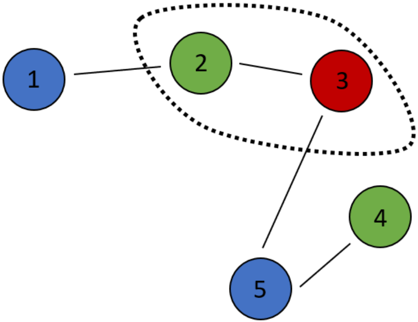
OK!

Mathematical Formulation - 3

Min distance between two vertices of the same color.

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

Satisfied if for each color h and for each subset $S \subset V \setminus C_h : |S| < \alpha_h$:
- either S is a chain in the tour connected with C_h by at most one edge
- or S is not a chain.



$\alpha_h = 3$

$C_h = \{1, 5\}$ (Blue)

$S = \{2, 3\}$

$x(S{:}S) = 1$

$x(S{:}C_h) = 2$

Constraint: $1 + 2 \leq 2$

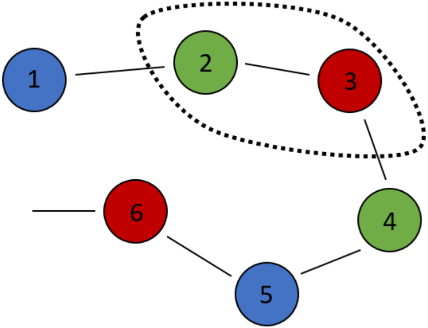
Violated!

Mathematical Formulation - 4

Limits the **max distance** between two vertices of the same color.

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

Satisfied if, for each color h and for each subset of vertices $S \subset V \setminus C_h$:
- either S is a chain not longer than β_h
- or S is not a chain.



$\beta_h = 3$

$C_h = \{1, 5\}$ (Blue)

$S = \{2, 3\}$

$V \setminus S = \{1, 4, 5, 6\}$

$x(S:V \setminus S) = 2$

Constraint: $2 \geq 4/3$

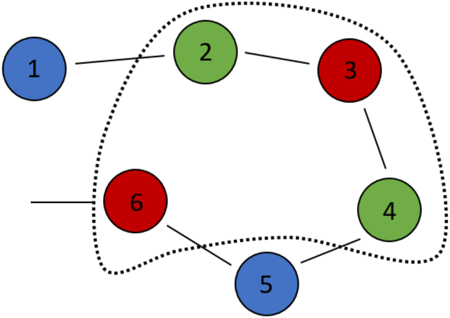
OK!

Mathematical Formulation - 4

Limits the **max distance** between two vertices of the same color.

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

Satisfied if, for each color h and for each subset of vertices $S \subset V \setminus C_h$:
- either S is a chain not longer than β_h
- or S is not a chain.



$\beta_h = 3$

$C_h = \{1, 5\}$ (Blue)

$S = \{2, 3, 4, 6\}$

$V \setminus S = \{1, 5\}$

$x(S:V \setminus S) = 4$

Constraint: $4 \geq 8/3$

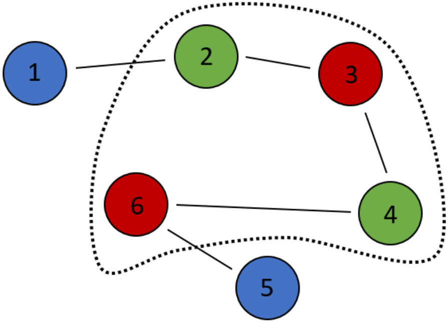
OK!

Mathematical Formulation - 4

Limits the **max distance** between two vertices of the same color.

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

Satisfied if, for each color h and for each subset of vertices $S \subset V \setminus C_h$:
- either S is a chain not longer than β_h
- or S is not a chain.



$\beta_h = 3$

$C_h = \{1, 5\}$ (Blue)

$S = \{2, 3, 4, 6\}$

$V \setminus S = \{1, 5\}$

$x(S{:}V \setminus S) = 2$

Constraint: $2 \geq 8/3$

Violated!

Branch-and-Cut

Our solution approach is based on the **branch-and-cut** paradigm.

1. Solve LP relaxation with
   - objective function (1)
   - cardinality constraints (2)
   - sub-tour elimination constraints (3) with $|S| = 2$
   - min distance (α) constraints (4) with $|S| \leq 2$.
2. Look for violated constraints
   - sub-tour elimination constraints (3) with $|S| > 2$
   - min distance (α) constraints (4) with $|S| > 2$
   - max distance (β) constraints (5).

   If violated constraint found add to LP and go to 1.
3. Stop loop when no violated constraints are identified.
4. If the solution found is not integer branch and repeat from 1.

Separation Procedures

Sub-tour elimination constraints (3)

Standard **max-flow-min-cut** based procedure.

Min distance (α) constraints (4)

Interpreted as **Enhanced Reverse Multistar Inequalities**:

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

$$\Updownarrow$$

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

Separation: polynomial time algorithm by *Gouveia and Salazar 2013*.

Separation Procedures

Max distance (β) constraints (5)

Max-flow-min-cut procedure on a modified graphs.

1. For each color $h \in C$ the modified graph is composed by all vertices in $V \setminus C_h$ plus a super-node obtained by merging all vertices in C_h.
2. Edges between vertices in C_h are ignored while the capacity associated with the other edges $e \in E$ is equal to the value of the associated variable $\bar{x}_e$ in the LP solution.
3. Compute the max-flow-min-cut from any node to the super-node.
4. A violated inequality is found if the flow is lower than $2\frac{|S|}{\beta_h}$ where S is the set of vertices not including C_h identified by the min-cut procedure.

Computational Experiments - Setup

Data-set

- 120 random instances.
- $|V| = \{10, 20, 50, 75, 100\}$.
- $|C| = \{3 \ldots 8\}$.
- Colors assigned starting from a random TSP solution to ensure feasibility.

Implementation

- C++.
- CPLEX 12.10 (default parameters).
- Max-flow-min-cut algorithm: Boykov-Kolmogorov MAXFLOW v.3.04.
- Time-limit of 7200 seconds.

Computational Experiments - Preliminary Results 1

$\|V\|$	$\|C\|$	α	β	*Gap%*	*Time*	*BBNode*	*Cut* (3)	*Cut* (4)	*Cut* (5)	*Opt*
10	3	2.00	3.00	0.00	0.02	0.00	4.00	0.00	0.00	4
10	4	2.38	5.13	0.00	0.04	0.00	7.00	0.00	5.00	4
10	5	6.00	7.00	0.00	0.07	0.00	4.75	0.00	6.25	4
10	6	6.00	7.13	0.00	0.12	8.50	16.25	2.50	20.00	4
10	7	6.13	7.50	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	6.25	0.25	6.25	4
20	3	2.00	3.00	0.00	0.03	0.00	4.25	0.00	0.00	4
20	4	2.00	6.00	0.00	0.23	41.25	44.25	0.00	36.25	4
20	5	2.25	9.50	0.00	0.20	49.75	55.75	0.00	56.25	4
20	6	3.63	10.50	0.00	0.29	19.50	71.50	7.25	72.00	4
20	7	5.38	12.00	0.00	0.48	426.75	154.50	51.00	119.50	4
20	8	7.00	12.25	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	13.00	0.00	0.00	4
50	4	2.00	7.75	0.00	0.49	0.00	57.50	0.00	10.00	4
50	5	2.00	10.50	0.00	16.58	2276.75	618.75	0.00	403.25	4

Computational Experiments - Preliminary Results 2

$\|V\|$	$\|C\|$	α	β	*Gap%*	*Time*	*BBNode*	*Cut* (3)	*Cut* (4)	*Cut* (5)	*Opt*
50	6	2.38	15.25	2.05	2692.94	75453.00	13757.75	20.75	14404.00	3
50	7	3.00	16.75	1.43	2167.97	72040.50	7563.75	2303.00	6808.50	3
50	8	2.75	19.38	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	30.00	0.00	0.00	4
75	4	2.00	7.13	0.00	60.96	2385.50	575.00	0.00	394.75	4
75	5	2.00	10.63	0.00	762.59	31806.00	8029.25	0.00	7547.25	3
75	6	2.00	16.38	0.00	401.73	14497.50	2741.25	0.00	1817.25	4
75	7	2.38	15.75	0.00	767.85	27668.00	4362.50	0.00	3543.75	4
75	8	2.63	19.38	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	72.75	0.00	0.00	4
100	4	2.00	8.25	0.00	78.80	1343.75	551.25	0.00	167.75	4
100	5	2.00	12.38	0.00	484.63	8336.75	1602.00	0.00	839.50	4
100	6	2.00	16.38	4.78	5161.20	48264.75	18090.25	0.00	12833.25	2
100	7	2.13	19.25	5.95	6327.64	65120.50	19076.00	0.00	15347.50	2
100	8	2.38	25.25	11.85	5418.75	52877.75	21574.50	0.00	14073.00	2

Future Developments

1. Primal solutions found after several nodes and slow improvement:
   - Primal heuristic.
   - Constraint Programming Model.
2. Slow convergence on large instances:
   - More efficient implementation of separation procedures.
   - Additional cuts.
3. Data-set limited to random instances with max 100 vertices:
   - New instances built on TSPLib instances.
   - New instances with up to 400 vertices.
   - New real-world instances.

That's all Folks!

Thank you for your attention