THE BI-OBJECTIVE MULTI-VEHICLE COVERING TOUR PROBLEM (BOMCTP): FORMULATION AND LOWER-BOUND COMPUTATION

B.M. SARPONG    C. ARTIGUES    N. JOZEFOWIEZ

LAAS-CNRS

13/04/2012
1. Introduction

2. Mathematical formulation of the BOMCTP

3. Column generation for a bi-objective integer problem

4. Lower bound for the BOMCTP

5. Conclusions and perspectives
Find a minimal-length tour on \( V' \subseteq V \) such that the nodes of \( W \) are covered by those of \( V' \).
Find a set of at most $m$ tours on $V' \subseteq V$, having minimum total length and such that the nodes of $W$ are covered by those of $V'$.

- The length of each route cannot exceed a preset value $p$.
- The number of vertices on each route cannot exceed a preset value $q$.

- May be visited
- MUST be visited : $T$
- MUST be covered : $W$
- Vehicle routes
- Cover distance
**Problem**

Given a graph $G = (V \cup W, E)$ with $T \subseteq V$, design a set of vehicle routes on $V' \subseteq V$.

**Objectives**

- Minimize the total length of the set of routes.
- Minimize the cover distance induced by the set of routes.

**Constraints**

- Each vertex of $T$ must belong to a vehicle route.
- Each vertex of $W$ must be covered.
- The length of each route cannot exceed a preset value $p$.
- The number of vertices on each route cannot exceed a preset value $q$. 
A set-covering model for the BOMCTP

**Variables**

- $\Omega$ : set of all feasible routes
- $r_k \in \Omega$ : feasible route $k$
- $c_k$ : cost of route $r_k$
- $\theta_k$ : 1 if route $r_k$ is selected in solution and 0 otherwise
- $z_{ij}$ : 1 if vertex $v_j \in V$ is used to cover vertex $w_i \in W$ and 0 otherwise
- $a_{ik}$ : 1 if $r_k$ uses vertex $v_i \in V$ and 0 otherwise
- $Cov_{\text{max}}$ : cover distance induced by a set of routes

**Objective functions**

$$\text{minimize} \sum_{r_k \in \Omega} c_k \theta_k$$

$$\text{minimize} \ Cov_{\text{max}}$$
A set-covering model for the BOMCTP

Constraints

\[- z_{ij} + \sum_{r_k \in \Omega} a_{jk} \theta_k \geq 0 \quad (w_i \in W, v_j \in V)\]

\[\sum_{r_k \in \Omega} a_{jk} \theta_k \geq 1 \quad (v_j \in T)\]

\[\text{Cov}_{\max} - c_{ij} z_{ij} \geq 0 \quad (w_i \in W, v_j \in V)\]

\[\sum_{v_j \in V} z_{ij} \geq 1 \quad (w_i \in W)\]

\[\text{Cov}_{\max} \geq 0\]

\[z_{ij} \in \{0, 1\} \quad (w_i \in W, v_j \in V)\]

\[\theta_k \in \{0, 1\} \quad (r_k \in \Omega)\]
Lower bound of a MOIP [Villarreal and Karwan, 1981]
Lower bound of a MOIP [Villarreal and Karwan, 1981]
Lower bound of a MOIP [Villarreal and Karwan, 1981]
Lower bound of a MOIP [Villarreal and Karwan, 1981]
Lower bound of a MOIP [Villarreal and Karwan, 1981]
Lower bound of a MOIP [Villarreal and Karwan, 1981]
Lower bound of a MOIP [Villarreal and Karwan, 1981]
Problem

minimize \((c_1x, c_2x)\)

\[Ax \geq b\]

\[x \geq 0 \text{ and integer}\]

Procedure

- Transform bi-objective problem into a single-objective one by means of \(\varepsilon\)-constraint scalarization.
- Solve the linear relaxation of the problem obtained for different values of \(\varepsilon\) by means of column generation.
**Master Problem**

minimize \( c_1 x \)

\[ Ax \geq b \]
\[ -c_2 x \geq -\varepsilon \]
\[ x \geq 0 \]

**Dual**

maximize \( by_1 - \varepsilon y_2 \)

\[ Ay_1 - c_2 y_2 \leq c_1 \]
\[ y_1, y_2 \geq 0 \]
**Approach 1: point-by-point search**
Approach 1: point-by-point search
APPROACH 1: POINT-BY-POINT SEARCH
APPRAOC 1: POINT-BY-POINT SEARCH
APPROACH 1: POINT-BY-POINT SEARCH
APPROACH 1: POINT-BY-POINT SEARCH
APPROACH 1: POINT-BY-POINT SEARCH
APPENDIX 1: point-by-point search
APPROACH 1: POINT-BY-POINT SEARCH
**Approach 1: Point-by-Point Search**

- $f_2$
- $f_1$
- $\varepsilon_0$
- $\varepsilon_1$
- $\varepsilon_2$
APPROACH 1: POINT-BY-POINT SEARCH
APPROACH 1: POINT-BY-POINT SEARCH
Approach 1: Point-by-Point Search
Approach 1: point-by-point search
Approach 1: point-by-point search
APPROACH 1: POINT-BY-POINT SEARCH

\[ f_2 \]
\[ f_1 \]
\[ \varepsilon_0 \]
\[ \varepsilon_1 \]
\[ \varepsilon_2 \]
\[ \varepsilon_{k-1} \]
Approach 1: point-by-point search
Approach 1: point-by-point search
Approach 2: Parallel Search 1
APPROACH 2: PARALLEL SEARCH 1
Approach 2: parallel search 1
Approach 2: Parallel Search 1
Approach 2: parallel search 1
Approach 2: parallel search

\[ f_2 \]
\[ f_1 \]
\[ \varepsilon_0 \]
\[ \varepsilon_1 \]
\[ \varepsilon_2 \]
\[ \varepsilon_3 \]
\[ \varepsilon_{k-1} \]
\[ \varepsilon_k \]

- Generate \( \frac{m}{k} \) columns for RMP
- Generate \( \frac{m}{k} \) columns for RMP
- Generate \( \frac{m}{k} \) columns for RMP
- Generate \( \frac{m}{k} \) columns for RMP
Approach 2: parallel search 1
APPROACH 2: PARALLEL SEARCH 1

\[ f_2 \]
\[ f_1 \]
\[ \varepsilon_0 \]
\[ \varepsilon_k \]
Approach 2: Parallel Search 1
Approach 2: parallel search 1
Approach 2: parallel search

\[ f_2 \]
\[ f_1 \]
\[ \varepsilon_0 \]
\[ \varepsilon_k \]
\[ \varepsilon_{k-1} \]
\[ \varepsilon_2 \]
\[ \varepsilon_1 \]
\[ \varepsilon_3 \]
\[ 1 \]
\[ 1 \]
\[ 1 \]
\[ 1 \]
Approach 3: parallel search 2
Approach 3: parallel search 2

\[ f_2 \]

\[ f_1 \]

\[ \varepsilon_0 \]

generate m columns for RMP
Approach 3: parallel search 2
Approach 3: parallel search 2
APPRAOCH 3: PARALLEL SEARCH 2

\[ f_2 \]

\[ f_1 \]

\[ \varepsilon_0 \]

\[ \varepsilon_1 \]

Generate \( m \) columns for RMP
Approach 3: parallel search 2

![Graph showing f1 and f2 axes with points ε0 and ε1]
Approach 3: parallel search 2

\[ f_2 \]
\[ f_1 \]
\[ \varepsilon_0 \]
\[ \varepsilon_1 \]
\[ \varepsilon_2 \]
Approach 3: parallel search 2

generate m columns for RMP
Approach 3: parallel search 2
Approach 3: parallel search 2
The Restricted Master Problem (RMP)

minimize \( \sum_{r_k \in \Omega_1} c_k \theta_k \)

Constraints

\[-z_{ij} + \sum_{r_k \in \Omega_1} a_{jk} \theta_k \geq 0 \quad (w_i \in W, v_j \in V)\]

\[\sum_{r_k \in \Omega_1} a_{jk} \theta_k \geq 1 \quad (v_j \in T)\]

\[\text{Cov}_{\text{max}} - c_{ij}z_{ij} \geq 0 \quad (w_i \in W, v_j \in V)\]

\[\sum_{v_j \in V} z_{ij} \geq 1 \quad (w_i \in W)\]

\[-\text{Cov}_{\text{max}} \geq -\varepsilon\]
The Restricted Master Problem (RMP)

\[
\text{minimize } \sum_{r_k \in \Omega_1} c_k \theta_k
\]

Constraints

\[-z_{ij} + \sum_{r_k \in \Omega_1} a_{jk} \theta_k \geq 0 \quad (w_i \in W, v_j \in V)\]

\[\sum_{r_k \in \Omega_1} a_{jk} \theta_k \geq 1 \quad (v_j \in T)\]

\[C_{ov_{\text{max}}} - c_{ij} z_{ij} \geq 0 \quad (w_i \in W, v_j \in V)\]

\[\sum_{v_j \in V} z_{ij} \geq 1 \quad (w_i \in W)\]

\[-C_{ov_{\text{max}}} \geq -\epsilon\]
**Dual of RMP**

maximize \(- \varepsilon \lambda + \sum_{w_i \in W} \beta_i + \sum_{v_j \in T} \varphi_j\)

subject to:

\[
\sum_{w_i \in W} a_{jk} \alpha_{ij} + \sum_{v_j \in T} a_{jk} \varphi_j \leq c_k \quad (r_k \in \Omega_1)
\]

\[-\lambda + \sum_{w_i \in W} \sum_{v_j \in V} \gamma_{ij} \leq 0
\]

\[-c_{ij} \gamma_{ij} + \beta_i - \alpha_{ij} \leq 0 \quad (w_i \in W, v_j \in V)\]
**Definition of sub-problem**

Find routes such that \( c_k - \sum_{w_i \in W} \sum_{v_j \in V} a_{jk} \alpha_{ij} - \sum_{v_j \in T} a_{jk} \varphi_j < 0. \)

- Let \( \alpha^*_{hj} = \alpha_{hj} \) if \( v_j \in V, w_h \in W \) and 0 otherwise.
- Let \( \varphi^*_j = \varphi_j \) if \( v_j \in T \) and 0 otherwise.
- Let \( A \) be the set of arcs formed between two nodes of \( V \).
- Let \( x_{ijk} = 1 \) if route \( r_k \) uses arc \( (v_i, v_j) \) and 0 otherwise.

**Note:** \( c_k = \sum_{(v_i, v_j) \in A} x_{ijk} c_{ij} \) and \( a_{jk} = \sum_{v_i \in V | (v_i, v_j) \in A} x_{ijk} \)

So \( \sum_{(v_i, v_j) \in A} c_{ij} x_{ijk} - \sum_{(v_i, v_j) \in A} \sum_{v_h \in W} \alpha^*_{hj} x_{ijk} - \sum_{(v_i, v_j) \in A} \varphi^*_j x_{ijk} < 0. \)
**Definition of sub-problem**

\[
\sum_{(v_i, v_j) \in A} \left( c_{ij} - \varphi_j^* - \sum_{v_h \in W} \alpha_{hj}^* \right) x_{ijk} < 0.
\]

**Sub-problem**

Find elementary paths from the depot to the depot with a negative cost, satisfying the constraints of length and maximum number of vertices on a path. Costs are set to

\[
c_{ij} - \varphi_j^* - \sum_{v_h \in W} \alpha_{hj}^*.
\]

- An elementary shortest path problem with resource constraints
- Solved by the Decremental State Space Relaxation (DSSR) algorithm [Righini and Salani, 2008].
Computational results

- **Instances:**
  - 120 random points generated in the \([0, 100] \times [0, 100]\) square
  - Set \(V\) taken as first \(|V|\) points; \(W\) taken as remaining points
- **RMP coded in C++ and solved with CPLEX 12.2**
- **Computer:** Intel Core 2 Duo, 2.93 GHz, 2 GB RAM

**Table:** Averages over 10 random instances for \(|T| = 1, \text{ and } q = +\infty\).

| \(|V|\) | \(p\) | Point-by-point | Parallel 1 | Parallel 2 |
|-------|------|--------------|------------|------------|
|       |      | time (sec)   | Nb. cols Gen | Nb. solved master | time (sec) | Nb. cols Gen | Nb. solved master | time (sec) | Nb. cols Gen | Nb. solved master |
| 40    | 6    | 5.93         | 16.7        | 34.5        | 46.41      | 18.7        | 917.8        | 4.35       | 16.8        | 35.0 |
|       | 8    | 11.76        | 18.0        | 34.6        | 48.61      | 20.1        | 925.5        | 5.49       | 18.3        | 35.1 |
|       | 12   | 25.01        | 20.7        | 34.9        | 57.21      | 23.3        | 947.0        | 7.12       | 20.6        | 35.4 |
| 50    | 6    | 10.94        | 18.2        | 38.9        | 60.87      | 20.5        | 1099.7       | 8.19       | 22.1        | 40.6 |
|       | 8    | 22.01        | 19.6        | 38.4        | 64.74      | 24.0        | 1115.2       | 8.99       | 19.7        | 38.9 |
|       | 12   | 41.74        | 22.1        | 37.1        | 75.85      | 26.9        | 1107.8       | 11.93      | 22.6        | 37.5 |
| 60    | 6    | 19.13        | 19.9        | 42.7        | 40.22      | 20.9        | 1017.4       | 9.45       | 20.4        | 43.1 |
|       | 8    | 44.56        | 19.3        | 42.3        | 52.78      | 22.5        | 1240.1       | 14.02      | 21.1        | 43.4 |
|       | 12   | 128.98       | 23.4        | 42.6        | 68.66      | 26.9        | 1258.1       | 22.55      | 25.8        | 44.0 |
Conclusions and perspectives

Conclusions

- Possible to have several (and efficient) ways of applying column generation to bi-objective integer problems.
- Model for BOMCTP has a weak linear relaxation.

Work in progress

- Investigate other intelligent ways of generating columns for a bi-objective integer problem.
- Test developed approaches on different problems (including another model for the BOMCTP with a stronger linear relaxation).
- Efficiently solve the BOMCTP by a multi-objective branch-and-price algorithm.
THANK YOU FOR YOUR ATTENTION.