Column Generation for a Class of Bi-Objective Vehicle Routing Problems

Christian Artigues, Nicolas Jozefowiez, Boadu M. Sarpong

LAAS-CNRS, Université de Toulouse, France

Friday, 12th 2014

Outline

- Column generation
- Column generation for multi-objective optimization
- Application to the bi-objective multi-vehicle covering tour problem
- Computation results
- Conclusions

Column Generation

What is column generation ?

A method for solving linear programs (LPs) in which there is an exponential number of variables without having to enumerate all the variables *a priori*

- Based on the principles of LP decomposition
- Useful in computing dual bounds for integer LPs
- Useful as a heuristic in finding feasible solutions of Integer LPs

Where has it been applied ?

- Vehicle routing problems
- Multi-commodity flow problems
- Cutting stock problems
- Binary cutting stock problems
- Crew rostering
- etc

Some Definitions

IP Master Problem (IPM)

The original IP having an exponential number of variables (corresponding to the columns of the constraint matrix)

LP Master Problem (LPM)

The linear relaxation of IPM

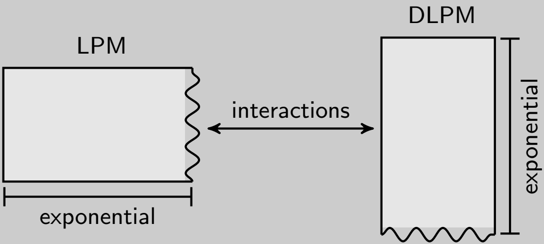
Restricted LP Master Problem (RLPM)

A copy of LPM in which only a subset of the original columns are present

Subproblem

A problem solved to determine which columns of the constraint matrix of LPM to introduce into an RLPM

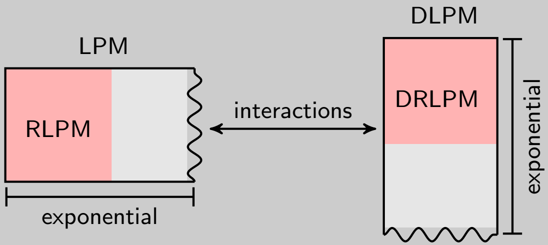
Main Idea and Convergence of Column Generation



Notes

- DLPM : Dual of LPM
- LPM has an exponential number of variables (columns)
- DLPM has an exponential number of constraints

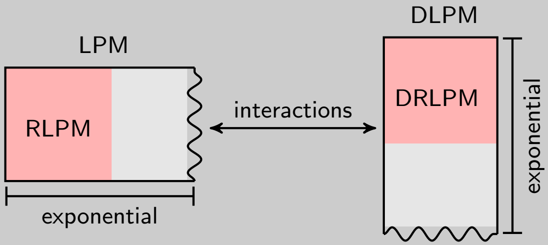
Main Idea and Convergence of Column Generation



Notes

- The feasible space of RLPM is a subset of the feasible space of LPM
- The feasible space of DLPM is a subset of the feasible space of DRLPM
- An optimal solution for DRLPM may not be feasible for DLPM

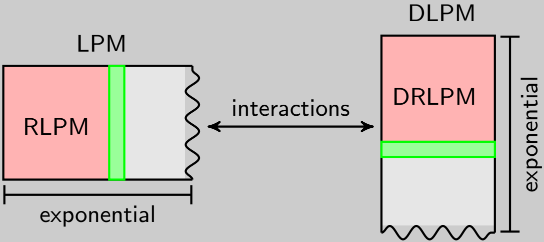
Main Idea and Convergence of Column Generation



Subproblem(s)

- Is the current objective value of RLPM optimal for LPM ?
- Are any constraints of DLPM violated in DRLPM (is DRLPM feasible) ?
- Depends on the particular problem and the formulation used

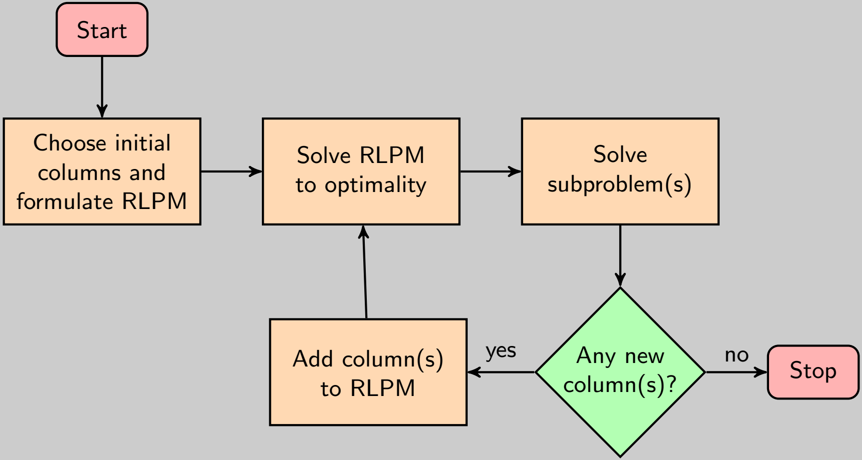
Main Idea and Convergence of Column Generation



Notes

- Adding a column to RLPM corresponds to adding a constraint to DRLPM
- Feasible space of RLPM enlarges and approaches that of LPM
- Feasible space of DRLPM shrinks and approaches that of DLPM

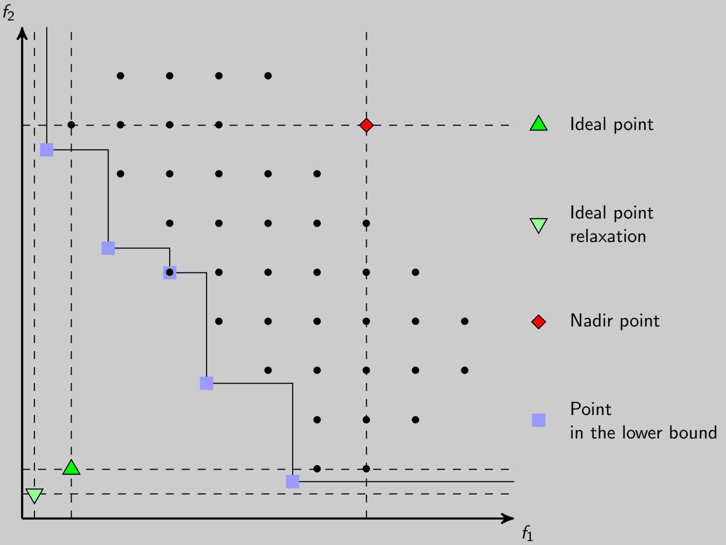
Flow Chart: A Column Generation Algorithm



Part I

Multi-objective column generation

Bounds



How to compute good quality lower bound ?

Column Generation in Multi-Objective Optimization

Some References

- Ehrgott and Tind. Column Generation in Integer Programming with Applications in Multicriteria Optimization. *Technical Report of the Faculty of Engineering, University of Auckland, New Zealand*, 2007

- Khanafer et al. The Min-Conflict Packing Problem. *Computers and Operations Research 39*, 2012

- Peng et al. A new column generation based algorithm for VMAT treatment plan optimization. *Physics in Medicine and Biology, 57(14)*, 2012

- Salari and Unkelbach. A Column-Generation-Based Method for Multi-Criteria Direct Aperture Optimization. *Physics in Medicine and Biology, 58*, 2013

Master problem (MP)

$$
\begin{array}{llll}
\text{minimize} & \sum_{j\in J} c_j^r \theta_j & & (r = 1, 2) \\
\text{s.t.} & \sum_{j\in J} a_{ij} \theta_j \geq & b_i & (i \in I) \\
& \theta_j \in \mathbb{N} & & (j \in J)
\end{array}
$$

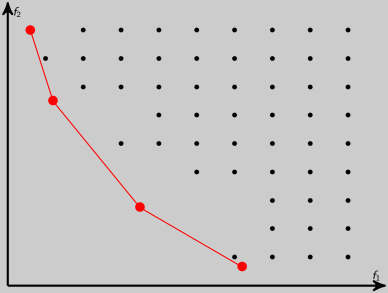
Linear relaxation of the MP (LMP)

$$\begin{array}{rlcll} \text{minimize} & \sum_{j \in J} c_j^r \theta_j & & & (r = 1, 2) \\ \text{s.t.} & \sum_{j \in J} a_{ij} \theta_j & \geq & b_i & (i \in I) \\ & \theta_j \in \mathbb{R}^+ & & & (j \in J) \end{array}$$

Linear relaxation of the MP (LMP)

$$\begin{array}{llll} \text{minimize} & \sum_{j \in J} c_j^r \theta_j & & (r = 1, 2) \\ \text{s.t.} & \sum_{j \in J} a_{ij} \theta_j \geq b_i & & (i \in I) \\ & \theta_j \in \mathbb{R}^+ & & (j \in J) \end{array}$$

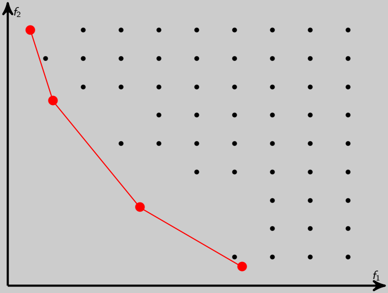
Weighted sum



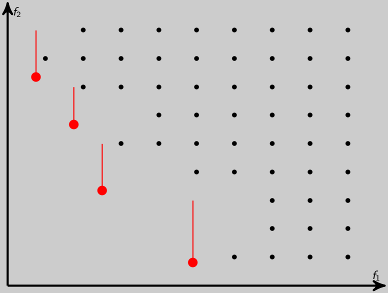
Linear relaxation of the MP (LMP)

$$\begin{aligned} \text{minimize} \quad & \textstyle\sum_{j\in J} c_j^r \theta_j & & (r = 1, 2) \\ \text{s.t.} \quad & \textstyle\sum_{j\in J} a_{ij}\theta_j \geq b_i & & (i \in I) \\ & \theta_j \in \mathbb{R}^+ & & (j \in J) \end{aligned}$$

Weighted sum



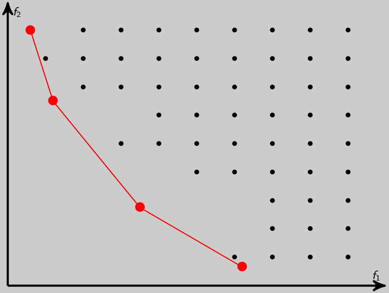
ϵ-constraint method



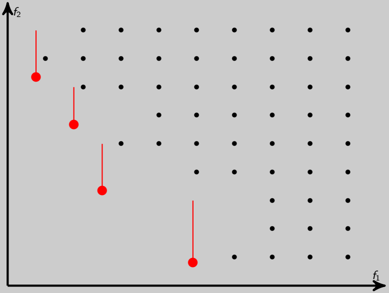
Linear relaxation of the MP (LMP)

$$\begin{aligned} \text{minimize} \quad & \sum_{j \in J} c_j^r \theta_j & & (r = 1, 2) \\ \text{s.t.} \quad & \sum_{j \in J} a_{ij} \theta_j \geq b_i & & (i \in I) \\ & \theta_j \in \mathbb{R}^+ & & (j \in J) \end{aligned}$$

Weighted sum



ϵ-constraint method

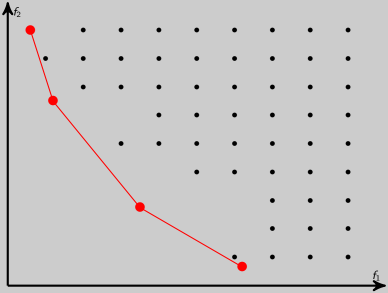


Restricted master problem (RMP) $J' \subset J, |J'| << |J|$

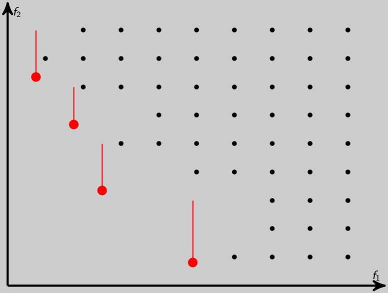
Linear relaxation of the MP (LMP)

$$\begin{array}{llll} \text{minimize} & \sum_{j \in J} c_j^r \theta_j & & (r = 1,2) \\ \text{s.t.} & \sum_{j \in J} a_{ij} \theta_j & \geq \; b_i & (i \in I) \\ & \theta_j \in \mathbb{R}^+ & & (j \in J) \end{array}$$

Weighted sum



ϵ-constraint method



Restricted master problem (RMP) $J' \subset J, |J'| << |J|$

$\rightarrow$ Generate columns for the ●

Using a Weighted Sum Method or an ε-Constraint Method

LPM(λ)

Given $\lambda = (\lambda_1, \lambda_2)$ with $\lambda_1, \lambda_2 \geq 0$ and $\lambda_1 + \lambda_2 = 1$

$$\begin{aligned} \text{Minimize} \quad & \lambda_1 (c^1)^T x + \lambda_2 (c^2)^T x \\ Ax &\geq b \\ x &\geq 0 \end{aligned}$$

LPM(ε)

Given $\varepsilon \in \mathbb{R}$

$$\begin{aligned} \text{Minimize} \quad & (c^1)^T x \\ Ax &\geq b \\ -(c^2)^T x &\geq -\varepsilon \\ x &\geq 0 \end{aligned}$$

Dual of LPM(λ)

$$\begin{aligned} \text{Maximize} \quad & b^T \pi \\ A^T \pi &\leq \lambda_1 c^1 + \lambda_2 c^2 \\ \pi &\geq 0 \end{aligned}$$

Dual of LPM(ε)

$$\begin{aligned} \text{Maximize} \quad & b^T \pi - \varepsilon \varphi \\ A^T \pi &\leq c^1 + \varphi c^2 \\ \pi, \varphi &\geq 0 \end{aligned}$$

Similar Subproblem Structure

Subproblem : $S(\lambda)$

Find a variable corresponding to a column of matrix A and which satisfy an inequality of the form :

$$\lambda_1 c^1 + \lambda_2 c^2 - A^T \pi < 0$$

Subproblem : $S(\varepsilon)$

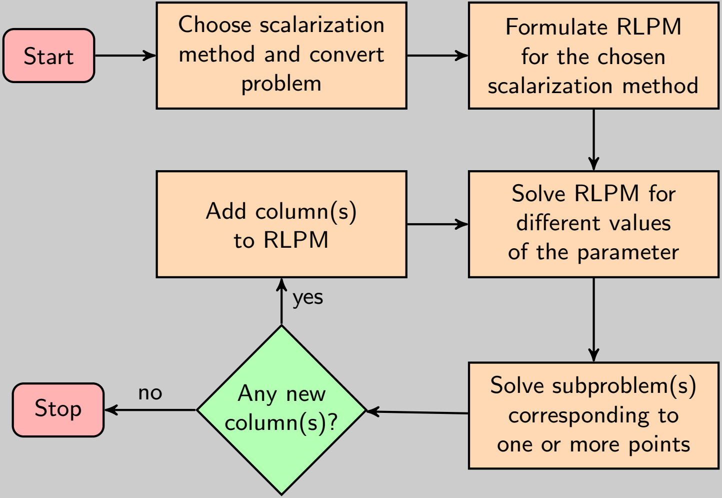
Find a variable corresponding to a column of matrix A and which satisfy an inequality of the form :

$$c^1 + \varphi c^2 - A^T \pi < 0$$

Implications and Consequences

- Subproblems have similar structure for both scalarization methods
- Strategies described for one scalarization method can be adopted for the other
- For any of the methods, it is possible to treat more than one subproblem at the same time when searching for columns

A Generalized Column Generation Algorithm for BOILPs



Point-by-point search (PPS)

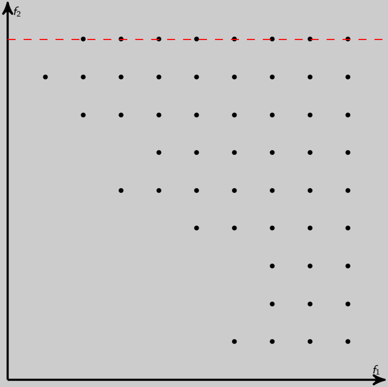
Point-by-point search (PPS)

Scalarization technique = ϵ-constraint method

Point-by-point search (PPS)

Scalarization technique = ϵ-constraint method

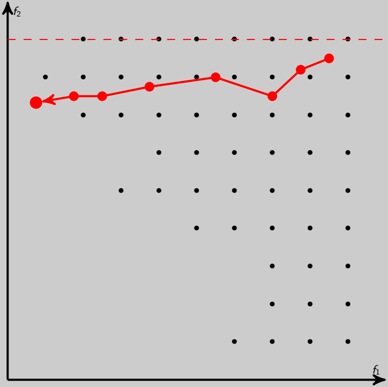
- Iterative ϵ-constraint method



Point-by-point search (PPS)

Scalarization technique = ϵ-constraint method

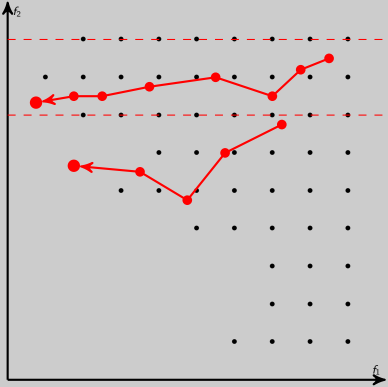
- Iterative ϵ-constraint method
- Full column generation algorithm at each iteration



Point-by-point search (PPS)

Scalarization technique = ϵ-constraint method

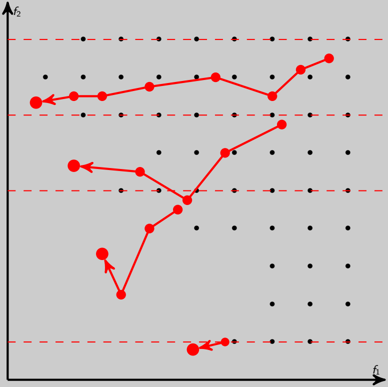
- Iterative ϵ-constraint method
- Full column generation algorithm at each iteration



Point-by-point search (PPS)

Scalarization technique = ϵ-constraint method

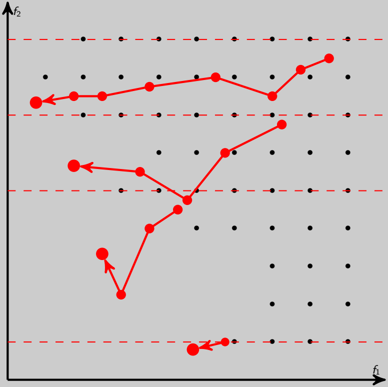
- Iterative ϵ-constraint method
- Full column generation algorithm at each iteration



Point-by-point search (PPS)

Scalarization technique = ϵ-constraint method

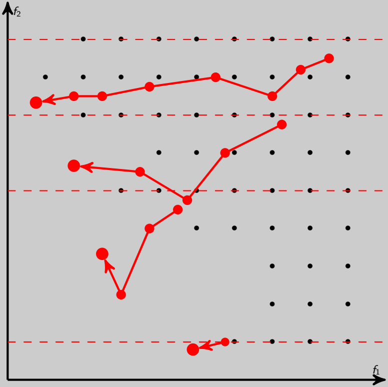
- Iterative ϵ-constraint method
- Full column generation algorithm at each iteration
- Possible improvements



Point-by-point search (PPS)

Scalarization technique = ϵ-constraint method

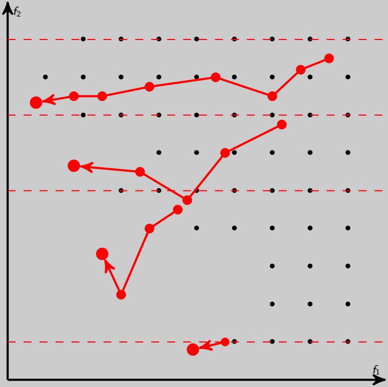
- Iterative ϵ-constraint method
- Full column generation algorithm at each iteration
- Possible improvements
  - Column storage



Point-by-point search (PPS)

Scalarization technique = ϵ-constraint method

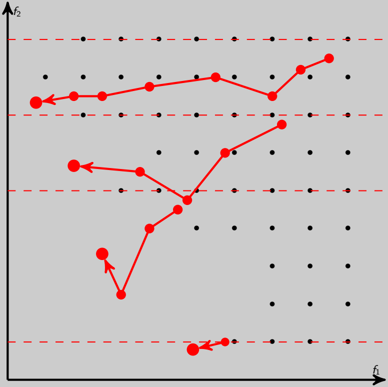
- Iterative ϵ-constraint method
- Full column generation algorithm at each iteration
- Possible improvements
  - Column storage
  - Blind ad-hoc heuristics (Improved PPS)
  - ...



Point-by-point search (PPS)

Scalarization technique = ϵ-constraint method

- Iterative ϵ-constraint method
- Full column generation algorithm at each iteration
- Possible improvements
  - Column storage
  - Blind ad-hoc heuristics (Improved PPS)
  - ...

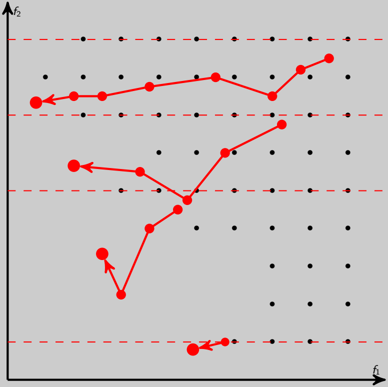


Problems: may be caught in a tailing effect, no uniform convergence, no factorization, not good as a heuristic ...

Point-by-point search (PPS)

Scalarization technique = ϵ-constraint method

- Iterative ϵ-constraint method
- Full column generation algorithm at each iteration
- Possible improvements
  - Column storage
  - Blind ad-hoc heuristics (Improved PPS)
  - ...



Problems: may be caught in a tailing effect, no uniform convergence, no factorization, not good as a heuristic ...

$\Rightarrow$ column search strategies

Solve once, generate for all (SOGA)

Scalarization technique = ϵ-constraint method

Solve once, generate for all (SOGA)

Scalarization technique = ϵ-constraint method

Main computational cost: solution of a subproblem

Solve once, generate for all (SOGA)

Scalarization technique = ϵ-constraint method

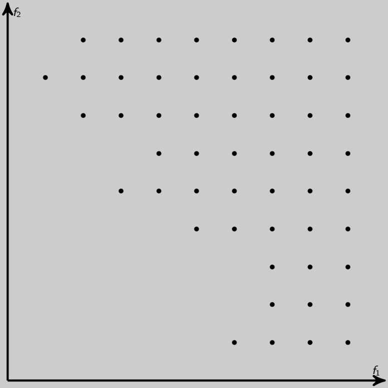
Main computational cost: solution of a subproblem
The subproblem is similar for several values of ϵ

Solve once, generate for all (SOGA)

Scalarization technique = ϵ-constraint method

Main computational cost: solution of a subproblem
The subproblem is similar for several values of ϵ

- At each iteration

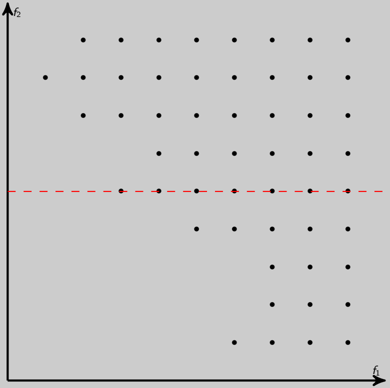


Solve once, generate for all (SOGA)

Scalarization technique = ϵ-constraint method

Main computational cost: solution of a subproblem
The subproblem is similar for several values of ϵ

- At each iteration
- Select a value ϵ_1

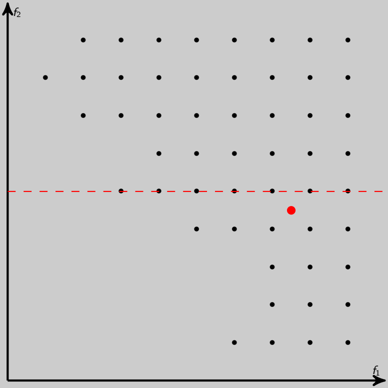


Solve once, generate for all (SOGA)

Scalarization technique = ϵ-constraint method

Main computational cost: solution of a subproblem
The subproblem is similar for several values of ϵ

- At each iteration
- Select a value ϵ_1
- Solve the LRMP for ϵ_1

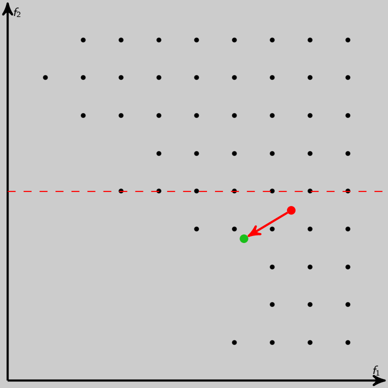


Solve once, generate for all (SOGA)

Scalarization technique = ϵ-constraint method

Main computational cost: solution of a subproblem
The subproblem is similar for several values of ϵ

- At each iteration
- Select a value ϵ_1
- Solve the LRMP for ϵ_1
- Search for a column set J^1

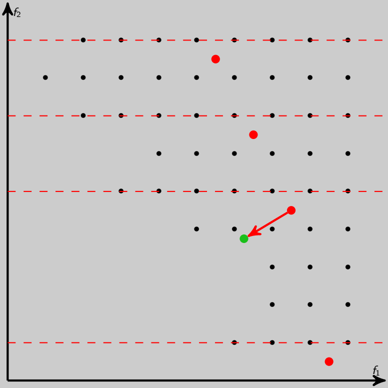


Solve once, generate for all (SOGA)

Scalarization technique = ϵ-constraint method

Main computational cost: solution of a subproblem
The subproblem is similar for several values of ϵ

- At each iteration
- Select a value ϵ_1
- Solve the LRMP for ϵ_1
- Search for a column set J^1
- For several ϵ_k, solve the LRMP $\rightarrow \pi_k^*$

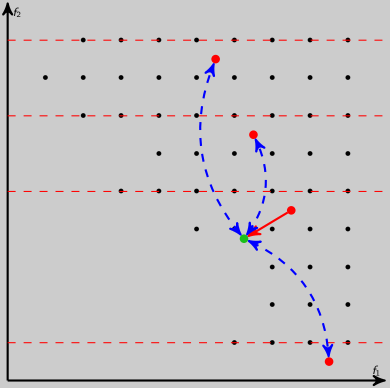


Solve once, generate for all (SOGA)

Scalarization technique = ϵ-constraint method

Main computational cost: solution of a subproblem
The subproblem is similar for several values of ϵ

- At each iteration
- Select a value ϵ_1
- Solve the LRMP for ϵ_1
- Search for a column set J^1
- For several ϵ_k, solve the LRMP $\rightarrow \pi_k^*$
- Heuristically built columns using J^1 and π_k^*

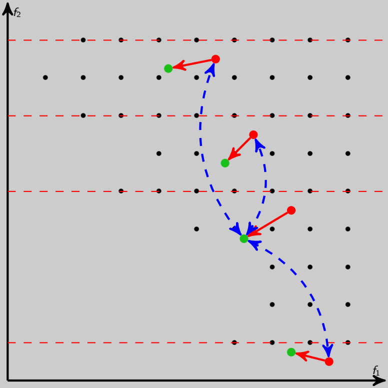


Solve once, generate for all (SOGA)

Scalarization technique = ϵ-constraint method

Main computational cost: solution of a subproblem
The subproblem is similar for several values of ϵ

- At each iteration
- Select a value ϵ_1
- Solve the LRMP for ϵ_1
- Search for a column set J^1
- For several ϵ_k, solve the LRMP $\rightarrow \pi_k^*$
- Heuristically built columns using J^1 and π_k^*



Part II

Application to a problem with a min-max objective

The multi-vehicle covering tour problem

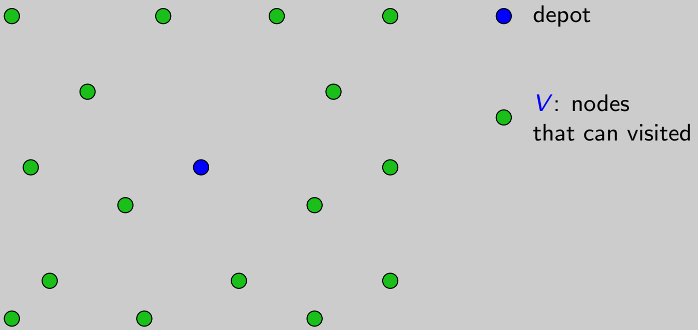
The multi-vehicle covering tour problem

Input: a valuated graph $G = (V \cup W, E, d)$, c, p

The multi-vehicle covering tour problem

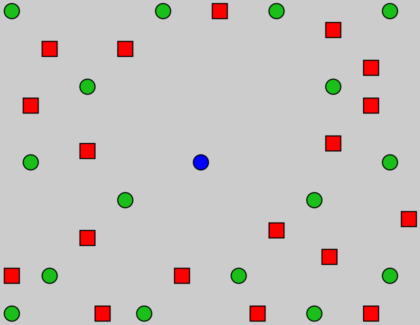
Input: a valuated graph $G = (V \cup W, E, d)$, c, p

Input: a valuated graph $G = (V \cup W, E, d)$, c, p

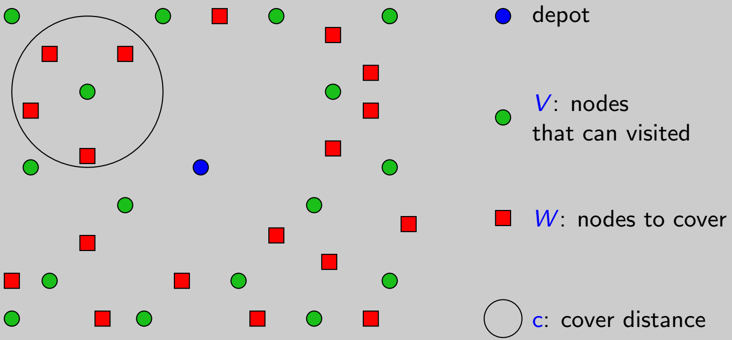


The multi-vehicle covering tour problem

Input: a valuated graph $G = (V \cup W, E, d)$, c, p

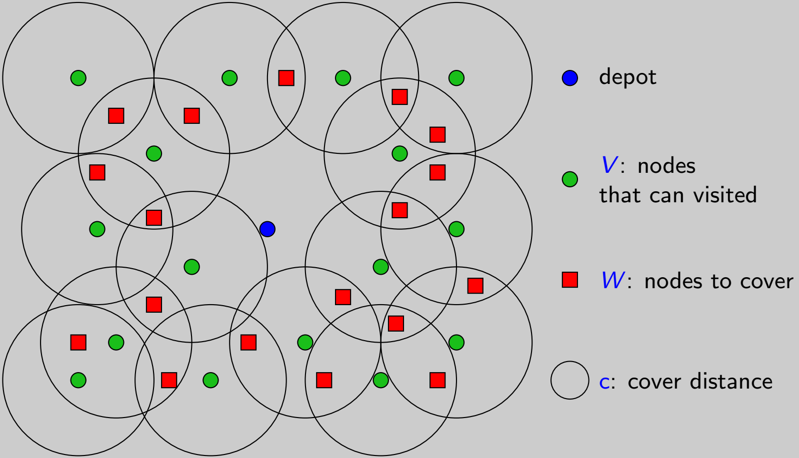


Input: a valuated graph $G = (V \cup W, E, d)$, c, p



THE MULTI-VEHICLE COVERING TOUR PROBLEM

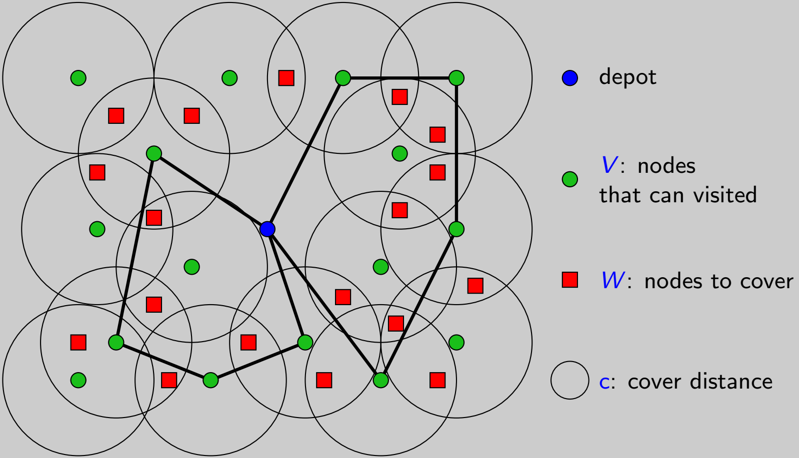
Input: a valuated graph $G = (V \cup W, E, d)$, c, p



The multi-vehicle covering tour problem

Input: a valuated graph $G = (V \cup W, E, d)$, c, p
Output: a minimal length set of routes on $V' \subseteq V$ s.t.

$$|V'| \leq p, \forall w_i \in W, \exists v_j \in V : d_{ij} \leq c$$



Bi-obj. multi-vehicle covering tour problem

Bi-obj. multi-vehicle covering tour problem

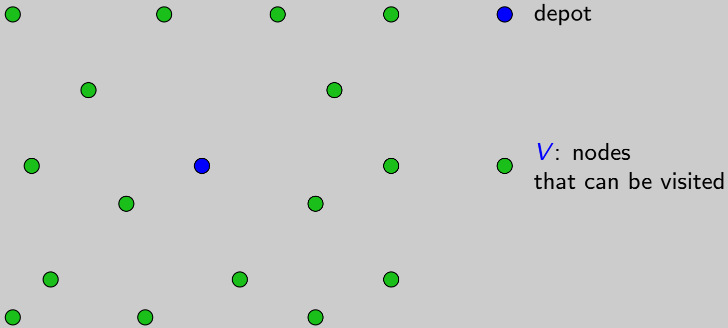
$G = (V \cup W, E, d)$

Bi-obj. multi-vehicle covering tour problem

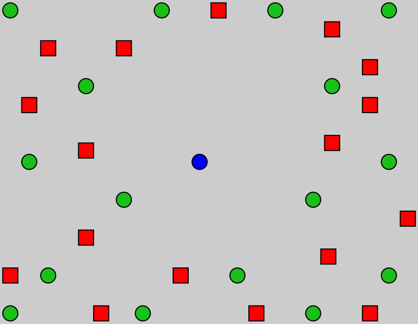
$G = (V \cup W, E, d)$

Bi-obj. multi-vehicle covering tour problem

$G = (V \cup W, E, d)$



$G = (V \cup W, E, d)$



depot

V: nodes that can be visited

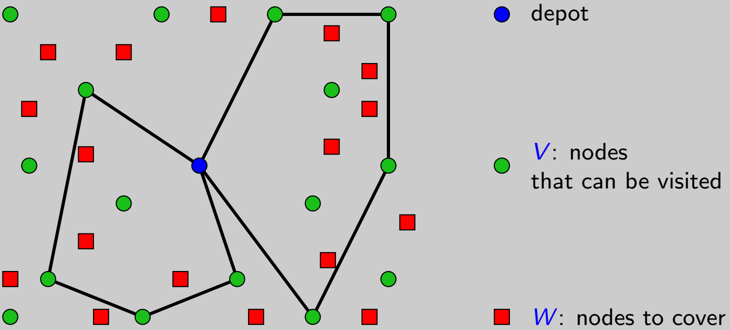
W: nodes to cover

Bi-obj. multi-vehicle covering tour problem

$G = (V \cup W, E, d)$, p: max # of nodes in a tour

A solution = a set of tours on $V' \subseteq V$

Objectives: i) minimize the total length

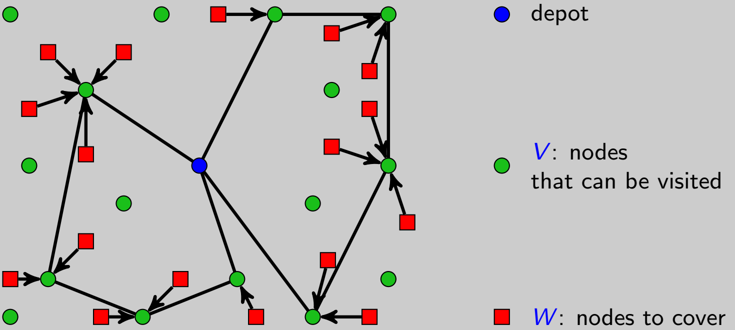


Bi-obj. multi-vehicle covering tour problem

$G = (V \cup W, E, d)$, p: max # of nodes in a tour

A solution = a set of tours on $V' \subseteq V$ + assignment of W to V'

Objectives: i) minimize the total length; ii) $\max_{w_i \in W} \min_{v_j \in V'} d_{ij}$



A FIRST MODEL

$$\text{minimize} \quad \sum_{r_k \in \Omega} c_k \theta_k \tag{1}$$

$$\text{minimize} \quad Cov_{\max} \tag{2}$$

$$Cov_{\max} - d_{ij} z_{ij} \geq 0 \quad (w_i \in W, v_j \in V \setminus \{v_0\}), \tag{3}$$

$$\sum_{v_j \in V \setminus \{v_0\}} z_{ij} \geq 1 \quad (w_i \in W), \tag{4}$$

$$\sum_{r_k \in \Omega} a_{jk} \theta_k - z_{ij} \geq 0 \quad (w_i \in W, v_j \in V \setminus \{v_0\}), \tag{5}$$

$$\sum_{r_k \in \Omega} a_{jk} \theta_k \geq 1 \quad (v_j \in T \setminus \{v_0\}), \tag{6}$$

$$Cov_{\max} \geq 0, \tag{7}$$

$$z_{ij} \in \{0, 1\} \quad (w_i \in W, v_j \in V \setminus \{v_0\}), \tag{8}$$

$$\theta_k \in \mathbb{N} \quad (r_k \in \Omega). \tag{9}$$

A second model for the BOMCTP

$$
\begin{aligned}
\text{minimize} \quad & \sum_{\omega_k \in R} c_k \theta_k \\
\text{minimize} \quad & \Gamma_{\max} \\
\text{s.t.} \quad & \sum_{\omega_k \in R} a_{ik} \theta_k \geq 1 \qquad (w_i \in W) \\
& \Gamma_{\max} \geq \rho_k \theta_k \quad (\omega_k \in R) \\
& \theta_k \in \{0, 1\} \qquad (\omega_k \in R)
\end{aligned}
$$

A second model for the BOMCTP

$$\begin{aligned}
\text{minimize} \quad & \sum_{\omega_k \in R} c_k \theta_k \\
\text{minimize} \quad & \Gamma_{\max} \\
\text{s.t.} \quad & \sum_{\omega_k \in R} a_{ik} \theta_k \geq 1 \qquad (w_i \in W) \\
& \Gamma_{\max} \geq \rho_k \theta_k \quad (\omega_k \in R) \\
& \theta_k \in \{0, 1\} \qquad (\omega_k \in R)
\end{aligned}$$

- $\omega_k \in R$: a tour on $V' \subseteq V + W' \subseteq W$

A second model for the BOMCTP

$$
\begin{aligned}
\text{minimize} \quad & \sum_{\omega_k \in R} c_k \theta_k \\
\text{minimize} \quad & \Gamma_{\max} \\
\text{s.t.} \quad & \sum_{\omega_k \in R} a_{ik} \theta_k \geq 1 \qquad (w_i \in W) \\
& \Gamma_{\max} \geq \rho_k \theta_k \quad (\omega_k \in R) \\
& \theta_k \in \{0,1\} \qquad (\omega_k \in R)
\end{aligned}
$$

- $\omega_k \in R$: a tour on $V' \subseteq V + W' \subseteq W$
- c_k: the tour length

A second model for the BOMCTP

$$
\begin{aligned}
\text{minimize} \quad & \sum_{\omega_k \in R} c_k \theta_k \\
\text{minimize} \quad & \Gamma_{\max} \\
\text{s.t.} \quad & \sum_{\omega_k \in R} a_{ik} \theta_k \geq 1 \quad (w_i \in W) \\
& \Gamma_{\max} \geq \rho_k \theta_k \quad (\omega_k \in R) \\
& \theta_k \in \{0, 1\} \quad (\omega_k \in R)
\end{aligned}
$$

- $\omega_k \in R$: a tour on $V' \subseteq V + W' \subseteq W$
- c_k: the tour length
- $a_{ik} = 1$ if $w_i \in W'$, 0 otherwise.

A second model for the BOMCTP

$$
\begin{aligned}
\text{minimize} \quad & \sum_{\omega_k \in R} c_k \theta_k \\
\text{minimize} \quad & \Gamma_{\max} \\
\text{s.t.} \quad & \sum_{\omega_k \in R} a_{ik} \theta_k \geq 1 \quad (w_i \in W) \\
& \Gamma_{\max} \geq \rho_k \theta_k \quad (\omega_k \in R) \\
& \theta_k \in \{0, 1\} \quad (\omega_k \in R)
\end{aligned}
$$

- $\omega_k \in R$: a tour on $V' \subseteq V$ + $W' \subseteq W$
- c_k: the tour length
- $a_{ik} = 1$ if $w_i \in W'$, 0 otherwise.
- $\rho_k = \max_{w_i \in W'} \min_{v_j \in V'} d_{ij}$

Reformulation

$$
\begin{aligned}
\text{minimize} \quad & \sum_{\omega_k \in R} c_k \theta_k & & \\
\text{s.t.} \quad & \sum_{\omega_k \in R} a_{ik} \theta_k \quad \geq \quad 1 & & (w_i \in W) \\
& \theta_k \in \{0, 1\} & & (\omega_k \in R)
\end{aligned}
$$

Reformulation

$$\begin{aligned} \text{minimize} \quad & \sum_{\omega_k \in R} c_k \theta_k \\ \text{s.t.} \quad & \sum_{\omega_k \in R} a_{ik} \theta_k \geq 1 \quad (w_i \in W) \\ & \theta_k \in \{0,1\} \quad (\omega_k \in R) \end{aligned}$$

- The master problem is a single objective problem

Reformulation

$$
\begin{aligned}
\text{minimize} \quad & \sum_{\omega_k \in R} c_k \theta_k \\
\text{s.t.} \quad & \sum_{\omega_k \in R} a_{ik} \theta_k \geq 1 \quad (w_i \in W) \\
& \theta_k \in \{0, 1\} \quad (\omega_k \in R)
\end{aligned}
$$

- The master problem is a single objective problem
- The subproblem is a single objective problem

$$
\begin{aligned}
\text{minimize} \quad & \sum_{\omega_k \in R} c_k \theta_k \\
\text{s.t.} \quad & \sum_{\omega_k \in R} a_{ik} \theta_k \geq 1 \quad (w_i \in W) \\
& \theta_k \in \{0, 1\} \quad (\omega_k \in R)
\end{aligned}
$$

- The master problem is a single objective problem
- The subproblem is a single objective problem
- Well-suited for an ϵ-constraint method [Bérubé et al., 2009]

Reformulation

$$
\begin{aligned}
\text{minimize} \quad & \sum_{\omega_k \in R} c_k \theta_k \\
\text{s.t.} \quad & \sum_{\omega_k \in R} a_{ik} \theta_k \geq 1 && (w_i \in W) \\
& \theta_k \in \{0,1\} && (\omega_k \in R)
\end{aligned}
$$

- The master problem is a single objective problem
- The subproblem is a single objective problem
- Well-suited for an ϵ-constraint method [Bérubé et al., 2009]
- $R^{\epsilon} = \{\omega_k \in R : \rho_k \leq \epsilon\}$

Reformulation

$$\begin{aligned} \text{minimize} \quad & \sum_{\omega_k \in R} c_k \theta_k \\ \text{s.t.} \quad & \sum_{\omega_k \in R} a_{ik} \theta_k \geq 1 \quad (w_i \in W) \\ & \theta_k \in \{0,1\} \quad (\omega_k \in R) \end{aligned}$$

- The master problem is a single objective problem
- The subproblem is a single objective problem
- Well-suited for an ϵ-constraint method [Bérubé et al., 2009]
- $R^{\epsilon} = \{\omega_k \in R : \rho_k \leq \epsilon\}$
- No weakening of the linear relaxation for a given ϵ value

Reformulation

$$
\begin{aligned}
\text{minimize} \quad & \sum_{\omega_k \in R} c_k \theta_k \\
\text{s.t.} \quad & \sum_{\omega_k \in R} a_{ik} \theta_k \geq 1 \quad (w_i \in W) \\
& \theta_k \in \{0,1\} \quad (\omega_k \in R)
\end{aligned}
$$

- The master problem is a single objective problem
- The subproblem is a single objective problem
- Well-suited for an ϵ-constraint method [Bérubé et al., 2009]
- $R^{\epsilon} = \{\omega_k \in R : \rho_k \leq \epsilon\}$
- No weakening of the linear relaxation for a given ϵ value
- Difference with the mono-objective model: a_{ik} is to be decided

Reformulation

$$
\begin{aligned}
\text{minimize} \quad & \sum_{\omega_k \in R} c_k \theta_k \\
\text{s.t.} \quad & \sum_{\omega_k \in R} a_{ik} \theta_k \quad \geq \quad 1 \quad (w_i \in W) \\
& \theta_k \in \{0, 1\} \qquad\qquad (\omega_k \in R)
\end{aligned}
$$

- The master problem is a single objective problem
- The subproblem is a single objective problem
- Well-suited for an ϵ-constraint method [Bérubé et al., 2009]
- $R^\epsilon = \{\omega_k \in R : \rho_k \leq \epsilon\}$
- No weakening of the linear relaxation for a given ϵ value
- Difference with the mono-objective model: a_{ik} is to be decided
- Large variety of problems

  A global objective on the complete solution

  An objective on the components → minimizing the worst case

Part III

Computational results

Experiments for the BOMCTP

Instances

- $|V| + |W|$ random points in the $[0, 100] \times [0, 100]$ square
- Depot is restricted to lie in the $[25, 75] \times [25, 75]$ square
- Set V taken as first $|V|$ points; Set W takes remaining points

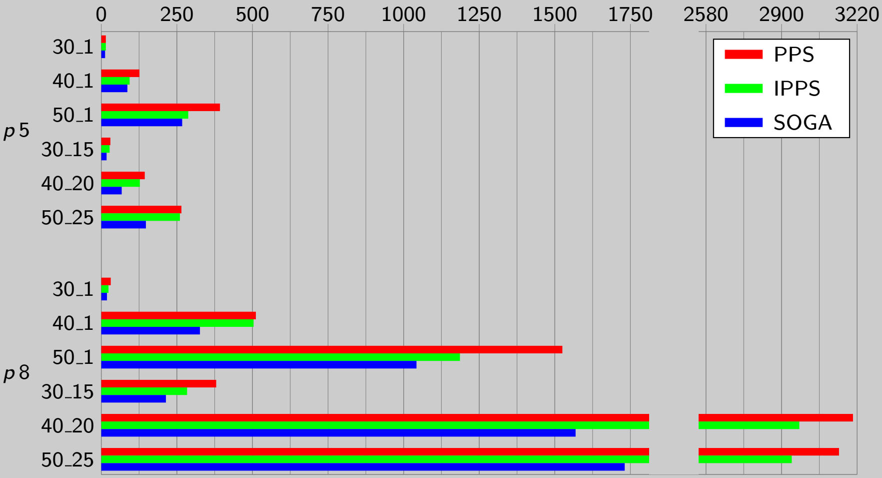
Algorithms and Coding

- All codes written in C/C++
- RLPM solved with CPLEX 12.4
- Subproblem solved by DSSR algorithm [*Boland et al. (2006), Righini and Salani (2008)*]
- SOGA/IPPS heuristics: "simple" greedy heuristics

Computer Specifications

- Intel Core 2 Duo, 2.93 GHz, 2 GB RAM

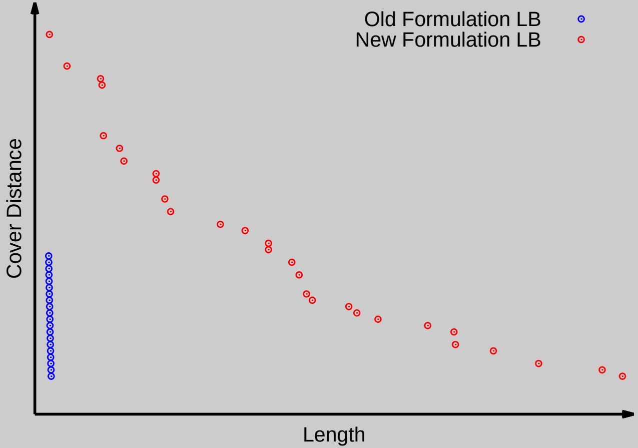
Computational Times (cpu seconds)



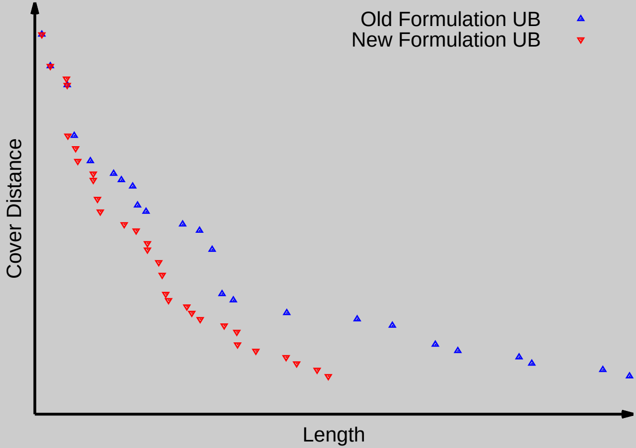
Number of Subproblems Solved

				PPS	IPPS	SOGA
p	$\|T\|$	$\|V\|$	$\|W\|$	# Subproblems	# Subproblems	# Subproblems
5	1	30	90	163	120	106
5	1	40	120	330	201	174
5	1	50	150	486	247	224
5	15	30	90	141	114	51
5	20	40	120	236	198	67
5	25	50	150	185	171	65
8	1	30	90	215	142	122
8	1	40	120	481	293	243
8	1	50	150	672	384	306
8	15	30	90	288	209	102
8	20	40	120	564	455	149
8	25	50	150	374	342	130

Instance type: $|V| = 50$, $|W| = 150$, $p = 5$, $q = \infty$.

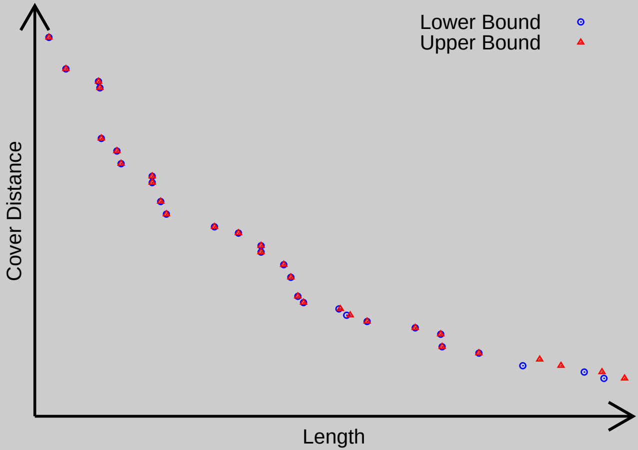


Instance type: $|V| = 50, \; |W| = 150, \; p = 5, \; q = \infty.$



Upper bound

Instance type: $|V| = 50, \ |W| = 150, \ p = 5, \ q = \infty.$



Part IV

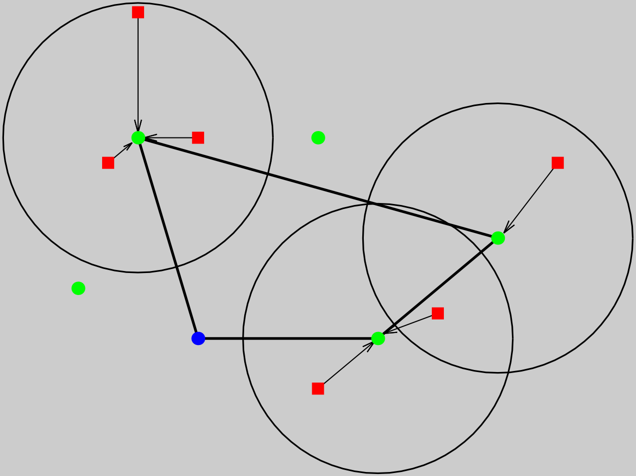
Conclusions

Conclusions

- Methods and models for computing lower bounds are needed in multi-objective optimization
- Application of column generation to multi-objective problems seems to have been overlooked
- Column generation techniques and strategies for single objective problems can easily be extended to bi-objective problems
- Good lower bounds for bi-objective problems can be obtained by column generation in reasonable times if columns are efficiently managed

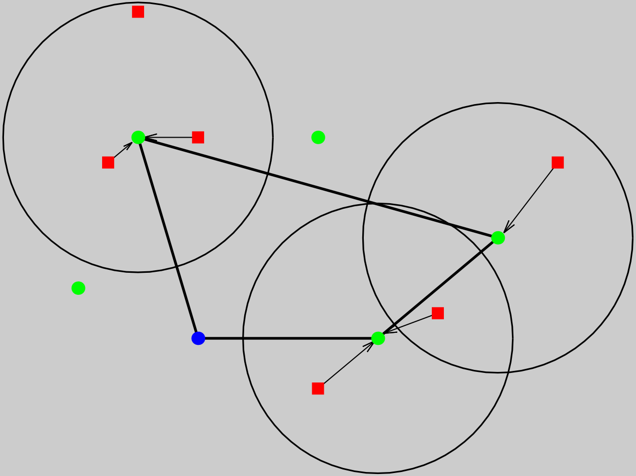
IPPS heuristics for the BOMCTP

At each column generation iteration, use heuristics to generate more columns from those returned by DSSR.



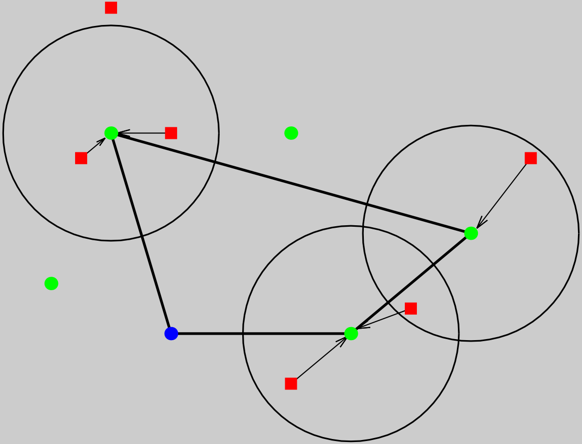
IPPS heuristics for the BOMCTP

At each column generation iteration, use heuristics to generate more columns from those returned by DSSR.



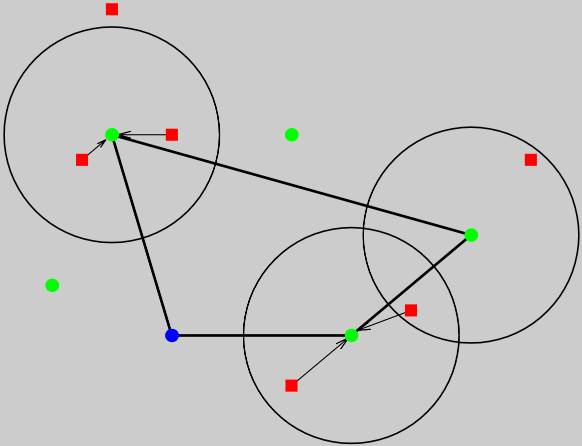
IPPS heuristics for the BOMCTP

At each column generation iteration, use heuristics to generate more columns from those returned by DSSR.



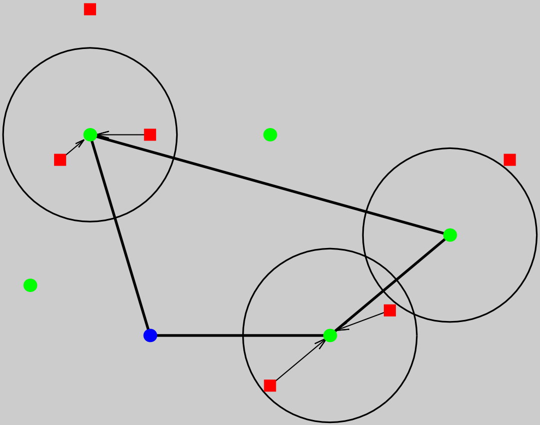
IPPS heuristics for the BOMCTP

At each column generation iteration, use heuristics to generate more columns from those returned by DSSR.



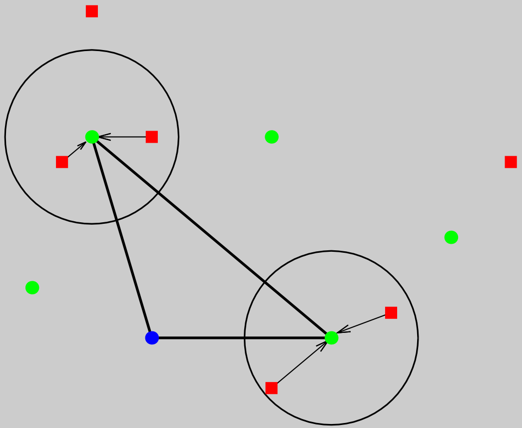
IPPS heuristics for the BOMCTP

At each column generation iteration, use heuristics to generate more columns from those returned by DSSR.



IPPS heuristics for the BOMCTP

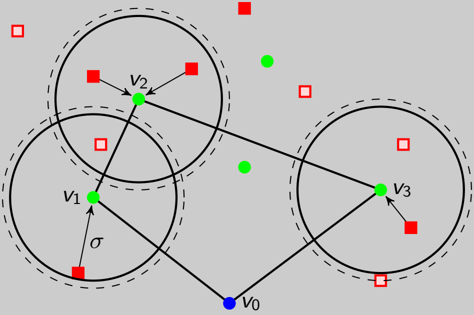
At each column generation iteration, use heuristics to generate more columns from those returned by DSSR.



SOGA heuristics for the BOMCTP

Principle

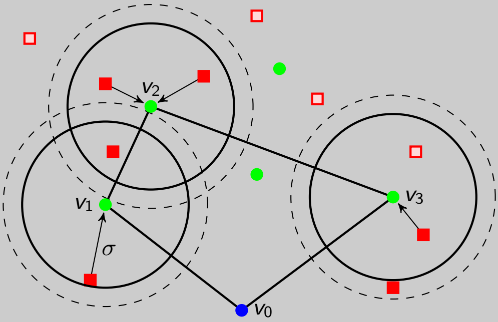
- Reconstruct the set of nodes to cover ($\Psi_k \subseteq W$)
- A different vector of dual values is used for each modification



SOGA heuristics for the BOMCTP

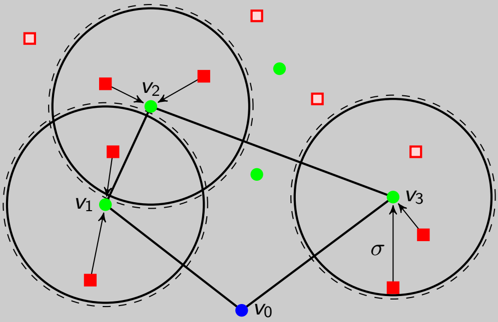
Principle

- Reconstruct the set of nodes to cover ($\Psi_k \subseteq W$)
- A different vector of dual values is used for each modification



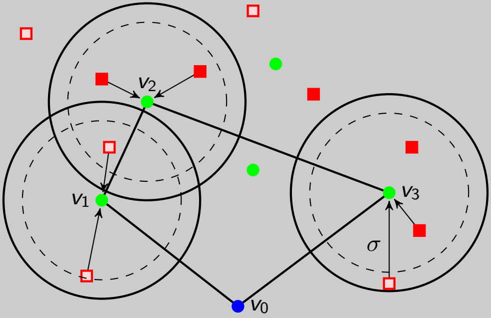
Principle

- Reconstruct the set of nodes to cover ($\Psi_k \subseteq W$)
- A different vector of dual values is used for each modification



Principle

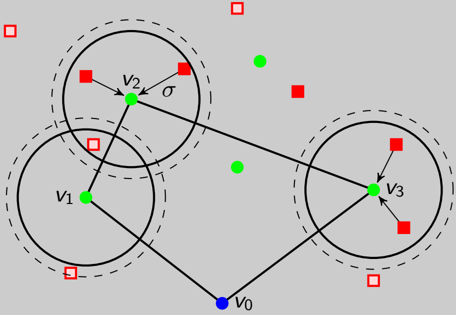
- Reconstruct the set of nodes to cover ($\Psi_k \subseteq W$)
- A different vector of dual values is used for each modification



SOGA heuristics for the BOMCTP

Principle

- Reconstruct the set of nodes to cover ($\Psi_k \subseteq W$)
- A different vector of dual values is used for each modification



Principle

- Reconstruct the set of nodes to cover ($\Psi_k \subseteq W$)
- A different vector of dual values is used for each modification