Exact Methods for Multi-objective Combinatorial Optimisation

Matthias Ehrgott

Department of Management Science
Lancaster University Management School, United Kingdom

13th MCDM/A Summer School
Chania, Crete, Greece
23 July – 3 August 2018

Overview

Overview

Overview

Overview

Overview

Overview

Overview

Mathematical Formulation

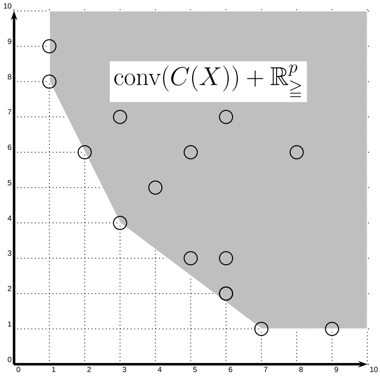
$$\begin{aligned} \min z(x) &= Cx \\ \text{subject to } Ax &= b \\ x &\in \{0,1\}^n \end{aligned}$$

$$\begin{aligned} x \in \{0,1\}^n &\longrightarrow n \text{ variables, } i = 1, \ldots, n \\ C \in \mathbb{Z}^{p \times n} &\longrightarrow p \text{ objective functions, } k = 1, \ldots, p \\ A \in \mathbb{Z}^{m \times n} &\longrightarrow m \text{ constraints, } j = 1, \ldots, m \end{aligned}$$

Combinatorial structure: paths, trees, flows, tours, etc.

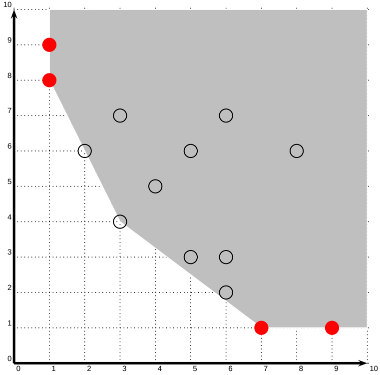
Feasible Sets

- $X = \{x \in \{0,1\}^n : Ax = b\}$ feasible set in decision space
- $Y = z(X) = \{Cx : x \in X\}$ feasible set in objective space
- $\mathrm{conv}(Y) + \mathbb{R}^p_{\geqq}$



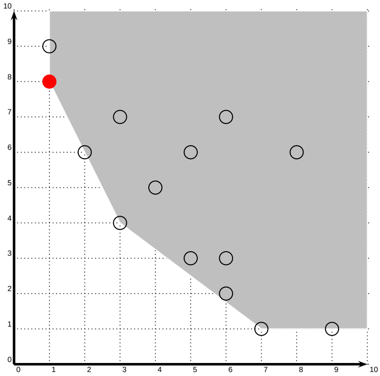
Lexicographic Optimality

- Individual minima
  $z_k(\hat{x}) \leqq z_k(x)$ for all $x \in X$
- Lexicographic optimality (1)
  $z(\hat{x}) \leqq_{lex} z(x)$ for all $x \in X$
- Lexicographic optimality (2)
  $z^\pi(\hat{x}) \leqq_{lex} z^\pi(x)$ for all $x \in X$ and some permutation z^π of $(z_1, \ldots, z_p)$



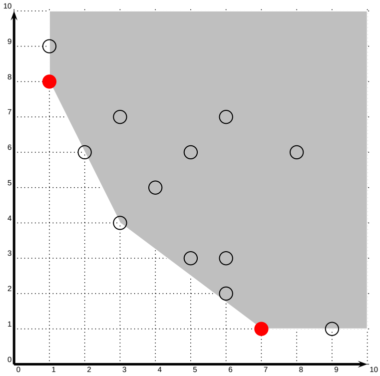
Lexicographic Optimality

- Individual minima
  $z_k(\hat{x}) \leqq z_k(x)$ for all $x \in X$
- Lexicographic optimality (1)
  $z(\hat{x}) \leqq_{lex} z(x)$ for all $x \in X$
- Lexicographic optimality (2)
  $z^\pi(\hat{x}) \leqq_{lex} z^\pi(x)$ for all $x \in X$
  and some permutation z^π of
  $(z_1, \ldots, z_p)$



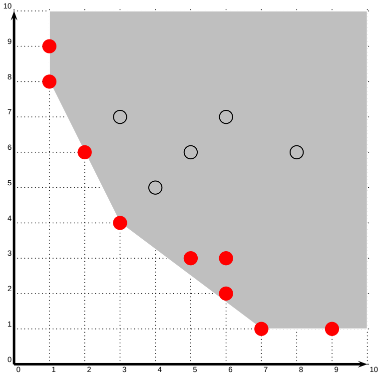
Lexicographic Optimality

- Individual minima
  $z_k(\hat{x}) \leqq z_k(x)$ for all $x \in X$
- Lexicographic optimality (1)
  $z(\hat{x}) \leqq_{lex} z(x)$ for all $x \in X$
- Lexicographic optimality (2)
  $z^\pi(\hat{x}) \leqq_{lex} z^\pi(x)$ for all $x \in X$ and some permutation z^π of $(z_1, \ldots, z_p)$



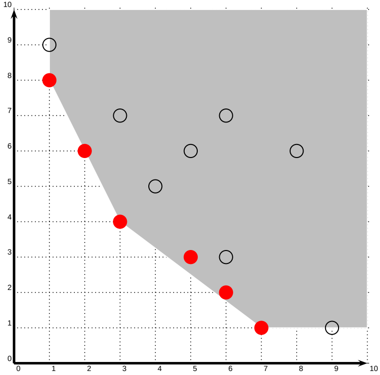
Efficient Solutions

- Weakly efficient solutions X_{wE}:
  there is no x with $z(x) < z(\hat{x})$
  $z(\hat{x})$ is weakly nondominated
  $Y_{wN} := z(X_{wN})$
- Efficient solutions X_E:
  there is no x with $z(x) \leq z(\hat{x})$
  $z(\hat{x})$ is nondominated
  $Y_N := z(X_E)$



Efficient Solutions

- Weakly efficient solutions X_{wE}:
  there is no x with $z(x) < z(\hat{x})$
  $z(\hat{x})$ is weakly nondominated
  $Y_{wN} := z(X_{wN})$
- Efficient solutions X_E:
  there is no x with $z(x) \leq z(\hat{x})$
  $z(\hat{x})$ is nondominated
  $Y_N := z(X_E)$

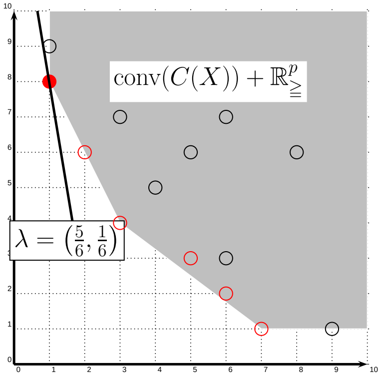


Efficient Solutions

- Supported efficient solutions X_{SE}: There is $\lambda > 0$ with $\lambda^T C\hat{x} \leqq \lambda^T Cx$ for all $x \in X$
  - $C\hat{x}$ is extreme point of $\text{conv}(Y) + \mathbb{R}^p_{\geqq} \to X_{SE1}$
  - $C\hat{x}$ is in relative interior of face of $\text{conv}(Y) + \mathbb{R}^p_{\geqq} \to X_{SE2}$
- Nonsupported efficient solutions X_{NE}: $C\hat{x}$ is in interior of $\text{conv}(Y) + \mathbb{R}^p_{\geqq}$

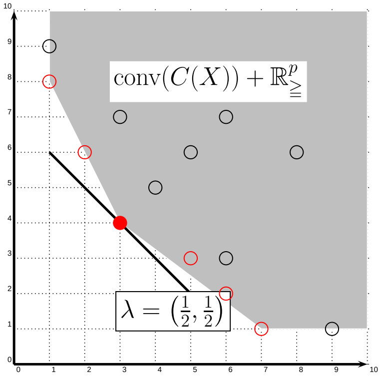
Efficient Solutions

- Supported efficient solutions X_{SE}: There is $\lambda > 0$ with $\lambda^T C\hat{x} \leqq \lambda^T Cx$ for all $x \in X$
  - $C\hat{x}$ is extreme point of $\text{conv}(Y) + \mathbb{R}^p_{\geqq} \to X_{SE1}$
  - $C\hat{x}$ is in relative interior of face of $\text{conv}(Y) + \mathbb{R}^p_{\geqq} \to X_{SE2}$
- Nonsupported efficient solutions X_{NE}: $C\hat{x}$ is in interior of $\text{conv}(Y) + \mathbb{R}^p_{\geqq}$



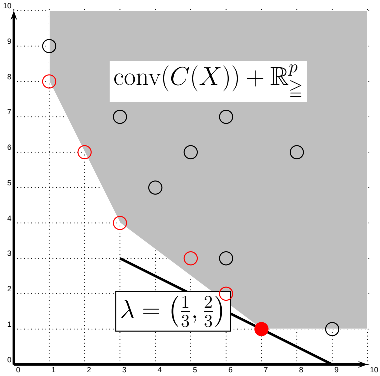
Efficient Solutions

- Supported efficient solutions X_{SE}: There is $\lambda > 0$ with $\lambda^T C\hat{x} \leqq \lambda^T Cx$ for all $x \in X$
  - $C\hat{x}$ is extreme point of $\text{conv}(Y) + \mathbb{R}^p_{\geqq} \to X_{SE1}$
  - $C\hat{x}$ is in relative interior of face of $\text{conv}(Y) + \mathbb{R}^p_{\geqq} \to X_{SE2}$
- Nonsupported efficient solutions X_{NE}: $C\hat{x}$ is in interior of $\text{conv}(Y) + \mathbb{R}^p_{\geqq}$



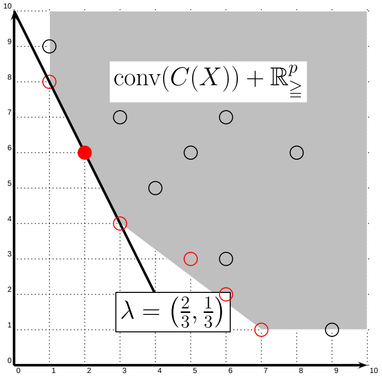
Efficient Solutions

- Supported efficient solutions X_{SE}: There is $\lambda > 0$ with $\lambda^T C\hat{x} \leqq \lambda^T Cx$ for all $x \in X$
  - $C\hat{x}$ is extreme point of $\text{conv}(Y) + \mathbb{R}^p_{\geqq} \rightarrow X_{SE1}$
  - $C\hat{x}$ is in relative interior of face of $\text{conv}(Y) + \mathbb{R}^p_{\geqq} \rightarrow X_{SE2}$
- Nonsupported efficient solutions X_{NE}: $C\hat{x}$ is in interior of $\text{conv}(Y) + \mathbb{R}^p_{\geqq}$



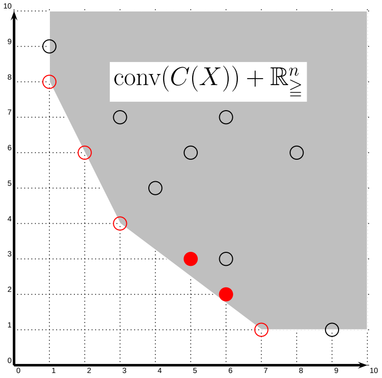
Efficient Solutions

- Supported efficient solutions X_{SE}: There is $\lambda > 0$ with $\lambda^T C\hat{x} \leqq \lambda^T Cx$ for all $x \in X$
  - $C\hat{x}$ is extreme point of $\text{conv}(Y) + \mathbb{R}^p_{\geqq} \to X_{SE1}$
  - $C\hat{x}$ is in relative interior of face of $\text{conv}(Y) + \mathbb{R}^p_{\geqq} \to X_{SE2}$
- Nonsupported efficient solutions X_{NE}: $C\hat{x}$ is in interior of $\text{conv}(Y) + \mathbb{R}^p_{\geqq}$



Efficient Solutions

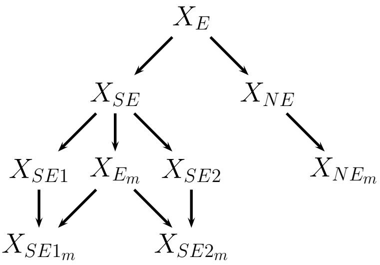
- Supported efficient solutions X_{SE}: There is $\lambda > 0$ with $\lambda^T C\hat{x} \leqq \lambda^T Cx$ for all $x \in X$
  - $C\hat{x}$ is extreme point of $\text{conv}(Y) + \mathbb{R}^p_{\geqq} \to X_{SE1}$
  - $C\hat{x}$ is in relative interior of face of $\text{conv}(Y) + \mathbb{R}^p_{\geqq} \to X_{SE2}$
- Nonsupported efficient solutions X_{NE}: $C\hat{x}$ is in interior of $\text{conv}(Y) + \mathbb{R}^p_{\geqq}$



Classification of Efficient Sets

Hansen 1979:

- $x^1, x^2 \in X_E$ are equivalent if $Cx^1 = Cx^2$
- Complete set: $\hat{X} \subset X_E$ such that for all $y \in Y_N$ there is $x \in \hat{X}$ with $z(x) = y$
- Minimal complete set contains no equivalent solutions
- Maximal complete set contains all equivalent solutions



MOCO Problems Are Hard

- Decision problem: Given $b \in \mathbb{Z}^p$: Does there exist $x \in X$ such that $Cx \leqq b$?
- Counting problem: Given $b \in \mathbb{Z}^p$: How many $x \in X$ satisfy $Cx \leqq b$?
- How many efficient solutions (nondominated points) do exist?
- KNAPSACK: Given $a^1, a^2 \in \mathbb{Z}^n$ and $b_1, b_2 \in \mathbb{Z}$, does there exist $x \in \{0,1\}^n$ such that $(a^1)^T x \leqq b_1$ and $(a^2)^T x \geqq b_2$?
- KNAPSACK is NP-complete and #P-complete

The Unconstrained MOCO Problem

Observation

Multiobjective combinatorial optimization problems are NP-hard, #P-complete, and intractable.

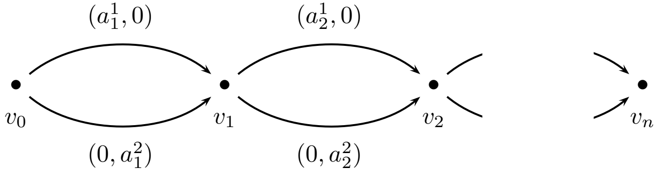
$$\begin{aligned} \min \sum_{i=1}^{n} c_i^k x_i & \qquad k = 1, \ldots, p \\ \text{subject to } x_i \in \{0, 1\} & \qquad i = 1, \ldots, n \end{aligned}$$

The Unconstrained MOCO Problem

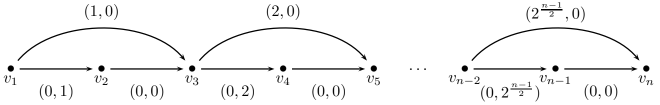
- Does there exist $x \in \{0,1\}^n$ such that $(c^1)^T x \leqq d_1$ and $(c^2)^T x \leqq d_2$?
- With an instance of KNAPSACK $c^1 := a^1$, $d_1 = b_1$, $c^2 := -a^2$, $d_2 := -b_2$ is a parsimonious transformation
- With $c_i^k := (-1)^k 2^{i-1}$ it holds $Y = Y_N$

Multiobjective Shortest Path Problem

- NP-hard: Hansen 1979



- Exponentially many efficient paths



Complexity results for MOCO problems

MOCO problem	Result	Reference
Bi-objective shortest path	NP-hard	[44]
	$\lvert Y_{SN} \rvert$ is exponential	[13]
Bi-objective integer minimum cost flow	$\lvert Y_{SN1} \rvert$ is exponential	[39]
Bi-objective minimum spanning tree	NP-hard	[3]
	$\lvert Y_{SN} \rvert$ is exponential	[12]
	$\lvert Y_{SN1} \rvert = O(\lvert \mathcal{E} \rvert^2)$	[41]
	$\lvert Y_{NN} \rvert$ is exponential	[40]
Bi-objective global minimum cut	$\lvert Y_N \rvert = O(\lvert \mathcal{V} \rvert^7)$	[1]
Bi-objective assignment	NP-hard	[44]
	#P-hard	[27]
Bi-objective search problem on a line	NP-hard	[30]
	$\lvert Y_N \rvert$ is exponential	[30]
Bi-objective uniform matroid	NP-complete	[8]

Number of Efficient Solutions

- Intractable: X_E, even Y_{SN}, can be exponential in the size of the instance
- Number of nondominated points for biobjective shortest path and assignment problems (Raith and Ehrgott 2009, Przybylski et al. 2008)

Shortest Path			Assignment	
Nodes	Edges	$\lvert Y_N \rvert$	n	$\lvert Y_N \rvert$
4,902	19,596	6	10	13
4,902	19,596	1,594	20	82
3,000	33,224	15	40	243
14,000	153,742	17	60	470
330,386	1,202,458	21	80	671
330,386	1,202,458	24	100	947

Number of Efficient Solutions

Empirically often

- $|X_{NE}|$ grows exponentially with instance size
- $|X_{SE}|$ grows polynomially with instance size
- but this depends on numerical values of C

Connectedness of Efficient Solutions

Define $\mathcal{EG} = (\mathcal{V}, \mathcal{E})$, where V are efficient solutions of a MOCO problem and $[x^1, x^2] \in E$ if and only if $x^2 \in N(x^1)$ and $x^1 \in N(x^2)$

Theorem

The adjacency graph of the set X_E of an instance of the bi-objective shortest path, bi-objective minimum spanning tree, and bi-objective integer minimum cost flow problems are not connected in general.

Overview

Labelling Algorithms for Shortest Path Problems

- Digraph $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ with arc costs $c_{ij}^k, k = 1, \ldots, p, (i,j) \in \mathcal{A}$
- Given origin $s \in \mathcal{V}$, destination $t \in \mathcal{V}$ find efficient paths from s to t:

$$\min_{P \in \mathcal{P}} \sum_{(i,j) \in P} c_{ij}$$

where $\mathcal{P}$ is set of all s-t paths

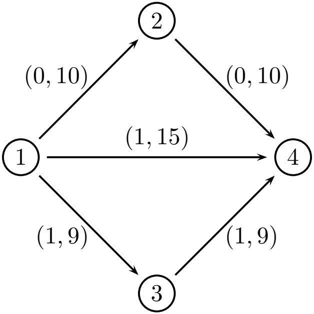
- Assume that all $c_{ij}^k \geqq 0$

Proposition

Let P_{st} be an efficient path from s to t. Then any subpath P_{uv} from u to v, where u and v are vertices on P_{st} is an efficient path from u to v.

Labelling Algorithms for Shortest Path Problem

Concatenations of efficient paths need not be efficient!



1-3 is efficient, 3-4 is efficient, 1-3-4 is not

Labelling Algorithms for Shortest Path Problems

For a labelling algorithm we need

- Sets of nondominated labels at each node
- A list of permanent and temporary labels
- Make sure that a permanent label defines an efficient path: Choose the lexicographically smallest label from temporary list

Lemma

If P_1 and P_2 are two paths between nodes s and t and $c(P_1) \leq c(P_2)$ then $c(P_1) <_{lex} c(P_2)$.

Multiobjective Label Correcting Algorithm

- Label setting fails if negative arc lengths are permitted
- Negative cycles C
  - Case 1: If $\sum_{a \in C} c_a^k < 0$ and $\sum_{a \in C} c_a^j > 0$ for $j \neq k$ there are infinitely many efficient paths
  - Case 2: If $\sum_{a \in C} c_a \leq 0$ there is no efficient path
- A label correcting algorithm is required

Greedy Algorithms and Spanning Trees

- Graph $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ with edge costs $c_{ij}^k, k = 1, \ldots, p; (i,j) \in \mathcal{E}$
- Find efficient spanning trees of $\mathcal{G}$:

$$\min_{T \in \mathcal{T}} \sum_{[i,j] \in T} c_{ij}$$

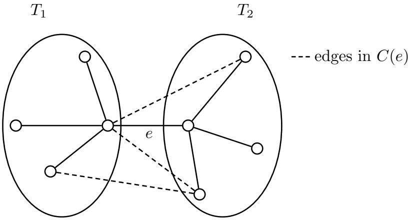
where $\mathcal{T}$ is set of all spanning trees of $\mathcal{G}$

The Multiobjective Spanning Tree Problem

Theorem (Hamacher and Ruhe 1994)

T efficient spanning tree of $\mathcal{G}$

1. *Let $e \in \mathcal{E}(T)$ be an edge of T. Let $(\mathcal{V}(T_1), \mathcal{E}(T_1))$ and $(\mathcal{V}(T_2), \mathcal{E}(T_2))$ be the two connected components of $\mathcal{G} \setminus \{e\}$. Let $C(e) := \{f = (v_i, v_j) \in \mathcal{E} : v_i \in \mathcal{V}(T_1), v_j \in \mathcal{V}(T_2)\}$ be* the cut *defined by deleting e. Then $c(e) \in \min\{c(f) : f \in C(e)\}$.*

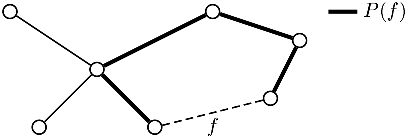


The Multiobjective Spanning Tree Problem

Theorem (Hamacher and Ruhe 1994)

T efficient spanning tree of $\mathcal{G}$

1. *Let $f \in \mathcal{E} \setminus \mathcal{E}(T)$ and let $P(f)$ be the unique path in T connecting the end nodes of f. Then $c(f) \leq c(e)$ does not hold for any $e \in P(f)$.*



This enables multiobjective Prim/Kruskal algorithms

Multi-objective Matroid Problems

Theorem (Serafini 1986)

Let B be an efficient matroid base. Then there exists a topological order of the elements of $\mathcal{E}$ such that the greedy algorithm applied to this order yields B.

Overview

Principle and Properties of Scalarization

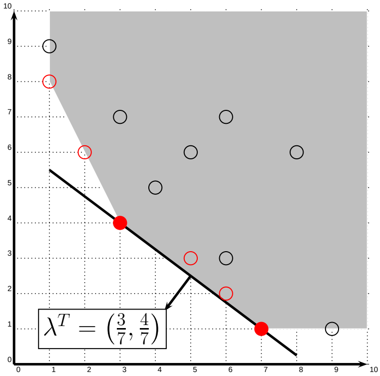
Convert multiobjective problem to (parameterized) single objective problem and solve repeatedly with different parameter values

Desirable properties of scalarizations: (Wierzbicki 1984)

- Correctness: Optimal solutions are (weakly) efficient
- Completeness: All efficient solutions can be found
- Computability: Scalarization is not harder than single objective version of problem (theory and practice)
- Linearity: Scalarization has linear formulation

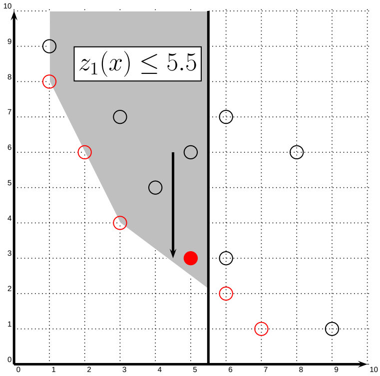
Scalarization Methods

- Weighted sum:
  $$\min_{x \in X} \left\{ \lambda^T z(x) \right\}$$
- ε-constraint:
  $$\min_{x \in X} \{ z_l(x) : z_k(x) \leq \varepsilon_k, k \neq l \}$$
- Weighted Chebychev:
  $$\min_{x \in X} \left\{ \max_{k=1,\ldots,p} \nu_k (z_k(x) - y_k^I) \right\}$$



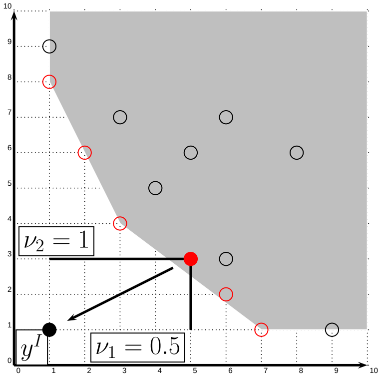
Scalarization Methods

- Weighted sum:
  $\min_{x \in X} \left\{ \lambda^T z(x) \right\}$
- ε-constraint:
  $\min_{x \in X} \{ z_l(x) : z_k(x) \leq \varepsilon_k, k \neq l \}$
- Weighted Chebychev:
  $\min_{x \in X} \left\{ \max_{k=1,\ldots,p} \nu_k (z_k(x) - y_k^I) \right\}$



Scalarization Methods

- Weighted sum:
  $$\min_{x \in X} \left\{ \lambda^T z(x) \right\}$$
- ε-constraint:
  $$\min_{x \in X} \{ z_l(x) : z_k(x) \leq \varepsilon_k, k \neq l \}$$
- Weighted Chebychev:
  $$\min_{x \in X} \left\{ \max_{k=1,\dots,p} \nu_k (z_k(x) - y_k^I) \right\}$$



General Formulation

$$\min_{x \in X} \left\{ \max_{k=1}^{p} [\nu_k(c_k x - \rho_k)] + \sum_{k=1}^{p} [\lambda_k(c_k x - \rho_k)] \right\}$$

$$\text{subject to} \quad c_k x \leq \varepsilon_k \qquad k = 1, \ldots, p$$

Includes	Correct	Complete	Computable	Linear
Weighted sum	+	-	+	+
ε-constraint	+	+	-	+
Chebychev	+	(+)	(-)	+

General Formulation

Theorem (Ehrgott 2005)

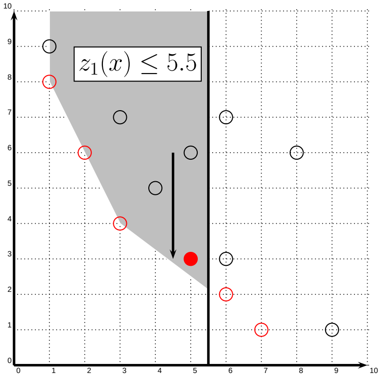
1. *The general scalarization is NP-hard.*
2. *An optimal solution of the Lagrangian dual of the linearized general scalarization is a supported efficient solution.*

Method of Elastic Constraints

$$\min_{x \in X} c_l x + \sum_{k \neq l} \mu_k w_k$$

$$\text{s.t.} \quad c_k x + v_k - w_k \leq \varepsilon_k \quad k \neq l$$

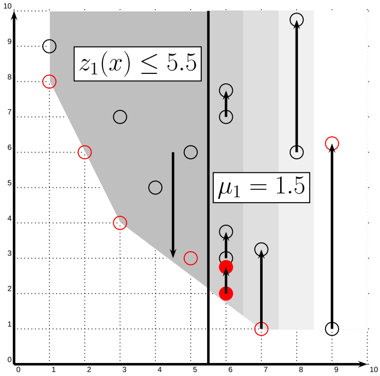
$$v_k, w_k \geq 0 \quad k \neq l$$



Method of Elastic Constraints

$$\min_{x \in X} c_l x + \sum_{k \neq l} \mu_k w_k$$

$$\begin{aligned} \text{s.t.} \quad c_k x + v_k - w_k &= \varepsilon_k \quad k \neq l \\ v_k, w_k &\geq 0 \quad k \neq l \end{aligned}$$



Method of Elastic Constraints

Theorem (Ehrgott and Ryan 2002)

The method of elastic constraints

- *is correct and complete,*
- *contains the weighted sum and ε-constraint method as special cases,*
- *is NP-hard.*

… but (often) solvable in practice because

- it "respects" problem structure
- it "limits damage" of ε-constraints

How many scalarised problems need to be solved?

Theorem

1. *In the case $p = 2$, the number of scalarised single objective problems to be solved in order to determine Y_N is bounded by $2|Y_N| - 1$ (Chalmet et al., 1986; Ralphs et al., 2006). In case the ε-constrained scalarisation is used, this bound is $|Y_N| + 1$ (Laumanns et al. 2006).*
2. *In the case $p = 3$ the bound is $3|Y_N| - 2$ and $2|Y_N| - 1$ for the ε-constraint scalarisation (Klamroth and Dächert 2015). For $p > 3$ the general bound is $O\left(|Y_N|^{\lfloor \frac{p}{2} \rfloor}\right)$ (Klamroth et al., 2015).*

Algorithms Based on Scalarisation

MOCO problem	Scalarisation	Reference
BO binary LP	Weighted sum with ε-constraints	[4]
BO knapsack, capacitated network routing	Weighted Chebychev	[37]
TO multidimensional knapsack	Lexicographic ε-constraint	[19]
Generic	Weighted sum with ε-constraints	[17]
TO three-dimensional knapsack	General scalarisation	[6]
BO integer minimum cost flow (*)	ε-constraint	[10]
BO knapsack	Lexicographic weighted Chebychev	[42]
BO multidimensional knapsack	Weighted sum with constraints	[47]
TO three-dimensional knapsack	Lexicographic ε-constraint	[28]
MO knapsack, shortest path, spanning tree	Lexicographic ε-constraint	[21]
MO three-dimensional knapsack, assignment	Lexicographic ε-constraint	[29]
MO TSP		
TO knapsack, assignment	Lexicographic ε-constraint	[15]
BO knapsack	Augmented weighted Chebychev	[5]
MO integer LP	Single objective with constraints	[18]
BO, TO multidimensional knapsack	Augmented ε-constraint	[24]
BO shortest path		
BO set partitioning (*)	Elastic constraint	[9], [48]
BO TSP with profits (*)	ε-constraint	[2]

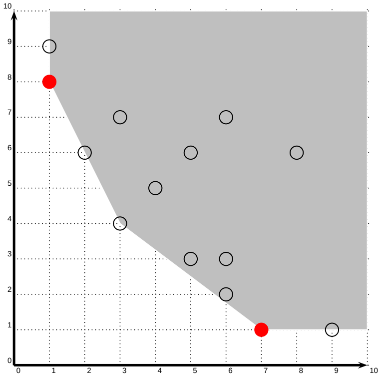
Overview

The Two Phase Method with 2 Objectives

- Phase 1: Compute $X_{SE(1)}$
  1. Find lexicographic solutions
  2. Recursively:
     Calculate λ
     Solve $\min_{x \in X} \lambda^T Cx$
- Phase 2: Compute X_{NE}
  1. Solve by triangle
  2. Use neighborhood (wrong)
  3. Use constraints (bad)
  4. Use variable fixing (possible)
  5. Use ranking (good)

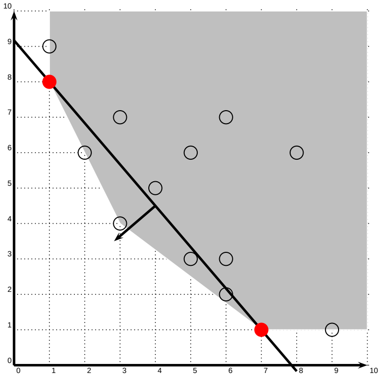
The Two Phase Method with 2 Objectives

- Phase 1: Compute $X_{SE(1)}$
  1. Find lexicographic solutions
  2. Recursively:
     Calculate λ
     Solve $\min_{x \in X} \lambda^T Cx$
- Phase 2: Compute X_{NE}
  1. Solve by triangle
  2. Use neighborhood (wrong)
  3. Use constraints (bad)
  4. Use variable fixing (possible)
  5. Use ranking (good)



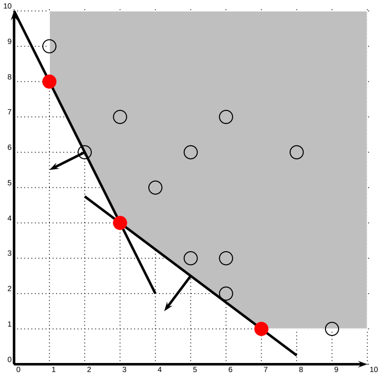
The Two Phase Method with 2 Objectives

- Phase 1: Compute $X_{SE(1)}$
  1. Find lexicographic solutions
  2. Recursively:
     Calculate λ
     Solve $\min_{x \in X} \lambda^T Cx$
- Phase 2: Compute X_{NE}
  1. Solve by triangle
  2. Use neighborhood (wrong)
  3. Use constraints (bad)
  4. Use variable fixing (possible)
  5. Use ranking (good)



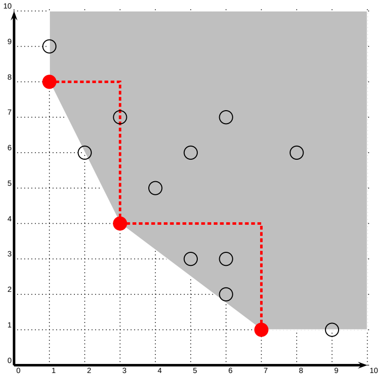
The Two Phase Method with 2 Objectives

- Phase 1: Compute $X_{SE(1)}$
  1. Find lexicographic solutions
  2. Recursively:
     Calculate λ
     Solve $\min_{x \in X} \lambda^T Cx$
- Phase 2: Compute X_{NE}
  1. Solve by triangle
  2. Use neighborhood (wrong)
  3. Use constraints (bad)
  4. Use variable fixing (possible)
  5. Use ranking (good)



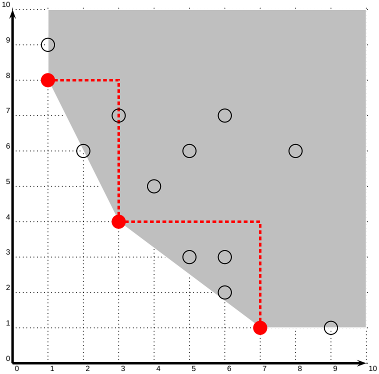
The Two Phase Method with 2 Objectives

- Phase 1: Compute $X_{SE(1)}$
  1. Find lexicographic solutions
  2. Recursively:
     Calculate λ
     Solve $\min_{x \in X} \lambda^T Cx$
- Phase 2: Compute X_{NE}
  1. Solve by triangle
  2. Use neighborhood (wrong)
  3. Use constraints (bad)
  4. Use variable fixing (possible)
  5. Use ranking (good)



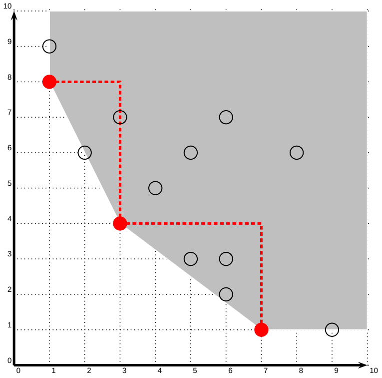
The Two Phase Method with 2 Objectives

- Phase 1: Compute $X_{SE(1)}$
  1. Find lexicographic solutions
  2. Recursively:
     Calculate λ
     Solve $\min_{x \in X} \lambda^T Cx$
- Phase 2: Compute X_{NE}
  1. Solve by triangle
  2. Use neighborhood (wrong)
  3. Use constraints (bad)
  4. Use variable fixing (possible)
  5. Use ranking (good)



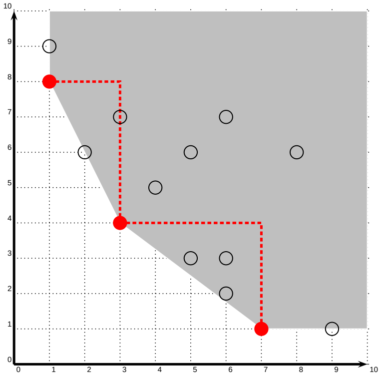
The Two Phase Method with 2 Objectives

- Phase 1: Compute $X_{SE(1)}$
  1. Find lexicographic solutions
  2. Recursively:
     Calculate λ
     Solve $\min_{x \in X} \lambda^T Cx$
- Phase 2: Compute X_{NE}
  1. Solve by triangle
  2. Use neighborhood (wrong)
  3. Use constraints (bad)
  4. Use variable fixing (possible)
  5. Use ranking (good)



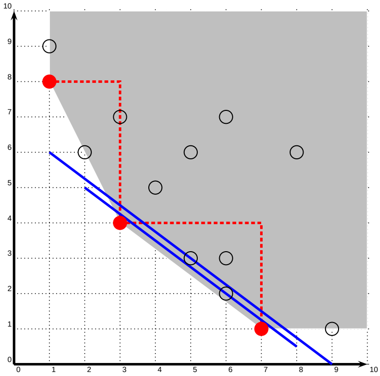
The Two Phase Method with 2 Objectives

- Phase 1: Compute $X_{SE(1)}$
  1. Find lexicographic solutions
  2. Recursively:
     Calculate λ
     Solve $\min_{x \in X} \lambda^T Cx$
- Phase 2: Compute X_{NE}
  1. Solve by triangle
  2. Use neighborhood (wrong)
  3. Use constraints (bad)
  4. Use variable fixing (possible)
  5. Use ranking (good)



The Two Phase Method with 2 Objectives

- Phase 1: Compute $X_{SE(1)}$
  1. Find lexicographic solutions
  2. Recursively:
     Calculate λ
     Solve $\min_{x \in X} \lambda^T Cx$
- Phase 2: Compute X_{NE}
  1. Solve by triangle
  2. Use neighborhood (wrong)
  3. Use constraints (bad)
  4. Use variable fixing (possible)
  5. Use ranking (good)

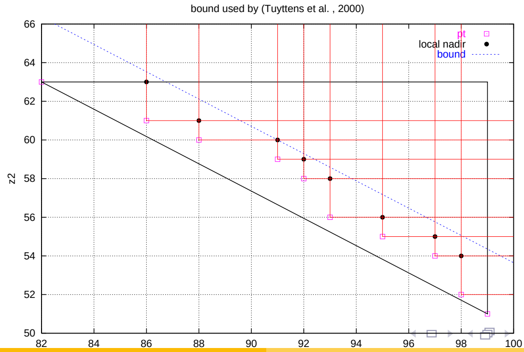


Bounds on $\lambda^T Cx$ in Phase Two

$\{x^i : 0 \leqq i \leqq q\}$ candidates for X_{NE} sorted by increasing z^1 in $\Delta(x^r, x^s)$

$$\gamma := \max_{i=0}^{q-1}\{\lambda_1 z_1(x^{i+1}) + \lambda_2 z_2(x^i)\}$$

$$\beta_0 := \max\left\{\gamma, \lambda^1 z^1(x^0) + \lambda^2 z^2(x^r), \lambda^1 z^1(x^s) + \lambda^2 z^2(x^q)\right\}$$



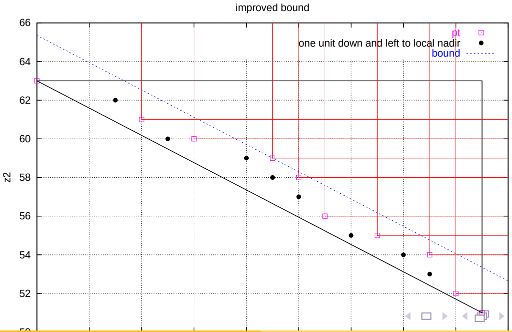
Bounds on $\lambda^T Cx$ in Phase Two

$\{x^i : 0 \leqq i \leqq q\}$ candidates for X_{NE} sorted by increasing z^1 in $\Delta(x^r, x^s)$

$$\delta_1 := \max_{i=0}^{q}\{\lambda^1 z^1(x^i) + \lambda^2 z^2(x^i)\}$$

$$\delta_2 := \max_{i=1}^{q}\{\lambda^1(z^1(x^i) - 1) + \lambda^2(z^2(x^{i-1}) - 1)\}$$

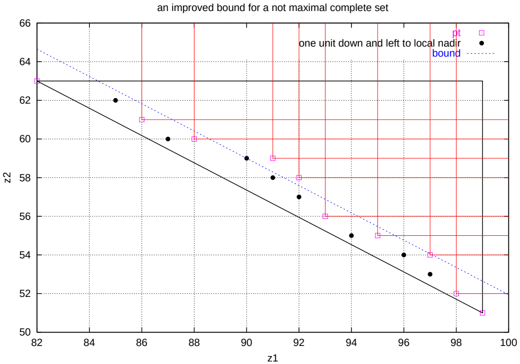
$$\beta_1 := \max\left\{\delta_1, \delta_2, \lambda^1(z^1(x^0) - 1) + \lambda^2(z^2(x^r) - 1), \lambda^1(z^1(x^s) - 1) + \lambda^2(z^2(x^q) - 1)\right\}$$



Bounds on $\lambda^T Cx$ in Phase Two

$\{x^i : 0 \leqq i \leqq q\}$ candidates for X_{NE} sorted by increasing z^1 in $\Delta(x^r, x^s)$

$$\beta_2 := \max\left\{\delta_2, \lambda^1(z^1(x^0) - 1) + \lambda^2(z^2(x^r) - 1), \lambda^1(z^1(x^s) - 1) + \lambda^2(z^2(x^n) - 1)\right\}$$



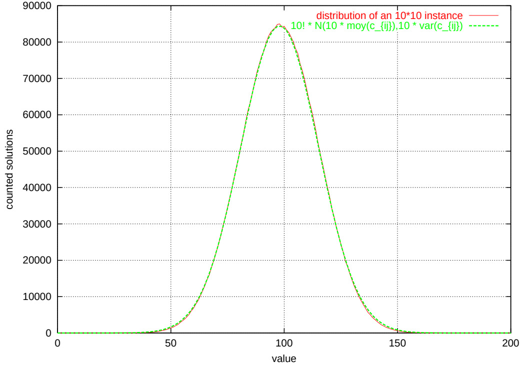
Two Phase Algorithm for Biobjective Assignment

Comparison with CPLEX 9.0 using constraints (3.4 GHz, 4 GB RAM)

Range of c_{ij}^k	CPLEX 9.0	Ranking
$[0, 20]$	200.63	85.58
$[0, 40]$	512.96	83.63
$[0, 60]$	1730.65	149.73
$[0, 80]$	3766.00	274.06
$[0, 100]$	4822.00	275.09

Two Phase Algorithm for Biobjective Assignment

- Objective values of an AP with $c_{ij} \in \{0, \ldots, r-1\}$



- Proof by Przybylski and Bourdon 2006:

$$\mu = \frac{n(r-1)}{2}, \sigma^2 = \frac{n(r^2-1)}{12}$$

The Two Phase Method with 3 Objectives

- Hyperplane defined by 3 points, possibly 6 lexicographically optimal points
- Which hyperplane to choose?
- Normal vector defined by 3 nondominated points may not be positive
- How to start the method?

Weight Space Decomposition

$$W^0 := \left\{\lambda : \lambda_1 > 0, \ldots, \lambda_p > 0, \lambda_p = 1 - \sum_{k=1}^{p-1} \lambda_k\right\}$$

$$W^0(y) := \{\lambda \in W^0 : \lambda^T y \leqq \lambda^t Y' \text{ for all } y' \in Y\}$$

Weight Space Decomposition

Proposition

1. *If y is a supported extreme point then $W^0(y)$ is a polytope of dim $p-1$.*
2. *Let y^1 and y^2 be two supported points and $W^0(y^1) \cap W^0(y^2) \neq \emptyset$ then $W^0(y^1)$ and $W^0(y^2)$ have a common face.*

Two supported extreme points y^1 and y^2 are adjacent if $W^0(y^1) \cap W^0(y^2)$ is a polytope of dimension $p-2$

Proposition

Let $\{y^1, \ldots, y^n\}$ be the set of supported extreme points, then $W^0 = \bigcup_{i=1}^n W^0(y^i)$.

Weight Space Decomposition

Proposition (Optimality Condition)

If S is a set of supported points then

$$Y_{SN1} \subseteq S \iff W^0 = \bigcup_{y \in S} W^0(y).$$

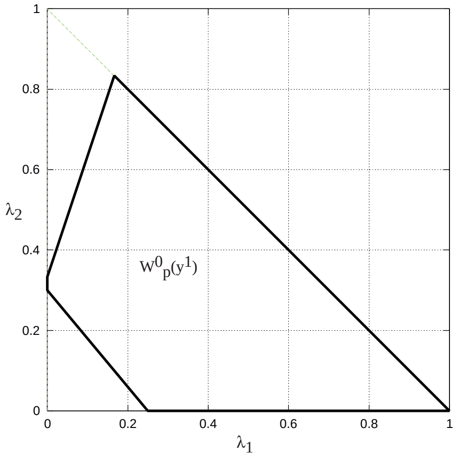
- Let S be a set of supported points
- Let $W^0_p(y) = \{\lambda \in W^0 : \langle \lambda, y \rangle \leq \langle \lambda, y^* \rangle \text{ for all } y^* \in S\}$
- $W^0(y) \subseteq W^0_p(y)$ for all $y \in S$
- $W^0 = \bigcup_{y \in S} W^0_p(y)$

An Example

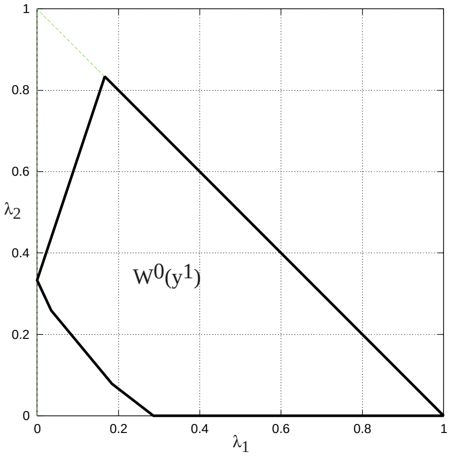
$$C^1 = \begin{pmatrix} 2 & 5 & 4 & 7 \\ 3 & 3 & 5 & 7 \\ 3 & 8 & 4 & 2 \\ 6 & 5 & 2 & 5 \end{pmatrix}, C^2 = \begin{pmatrix} 3 & 3 & 6 & 2 \\ 5 & 3 & 7 & 3 \\ 5 & 2 & 7 & 4 \\ 4 & 6 & 3 & 5 \end{pmatrix}, C^3 = \begin{pmatrix} 4 & 2 & 5 & 3 \\ 5 & 3 & 4 & 3 \\ 4 & 3 & 5 & 2 \\ 6 & 4 & 7 & 3 \end{pmatrix}$$

- Lexicographically optimal points: $y^1 = (9, 13, 16)$, $y^2 = (19, 11, 17)$, $y^3 = (18, 20, 13)$
- $S = \{y^1, y^2, y^3\}$

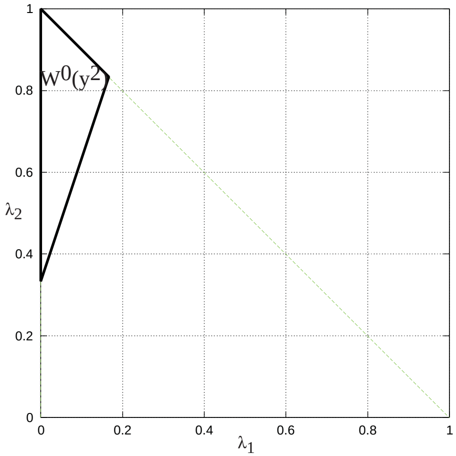
An Example



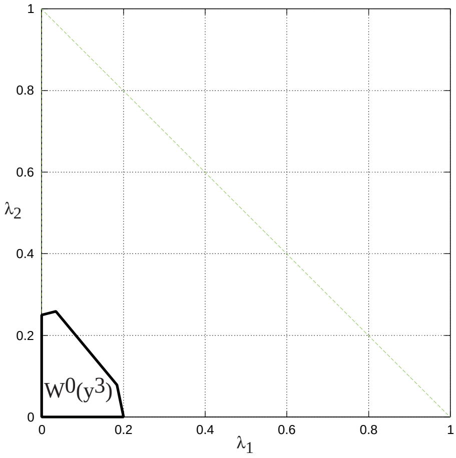
An Example



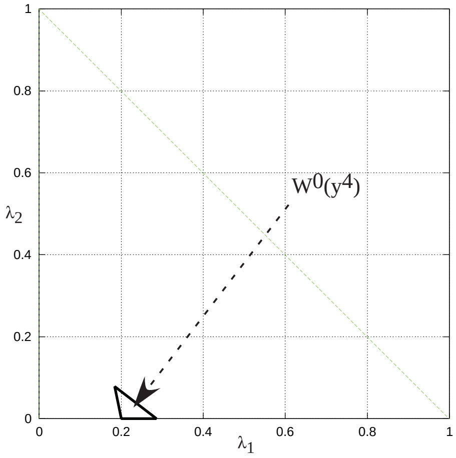
An Example



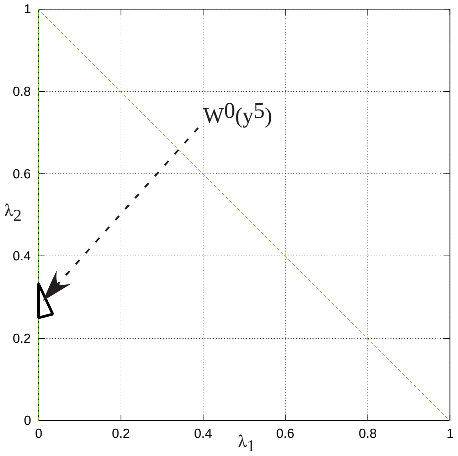
An Example



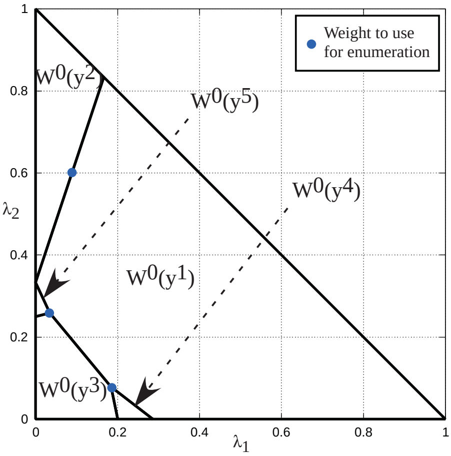
An Example



An Example



An Example


Weight to use for enumeration
W0(y2)
W0(y5)
W0(y4)
W0(y1)
W0(y3)
λ2
λ1
0
0.2
0.4
0.6
0.8
1

Phase 2 for 3 Objectives

- Intersection points between 3 weight sets are used for ranking solutions of weighted sum problems
- But areas for nonsupported nondominated points do not decompose as for $p = 2$
- Bounds can be generalised
- Recursive algorithm for $p > 3$ is possible

Numerical Results

Size	$\vert Y_N\vert$	SC 2004	T-P 2003	LZT 2005	2 Phase
5	12	0.15	0.04	0.15	0.00
10	221	99865.00	97.30	41.70	0.08
15	483	×	544.53	172.29	0.36
20	1942	×	×	1607.92	4.51
25	3750	×	×	5218.00	30.13
30	5195	×	×	15579.00	55.87
35	10498	×	×	101751.00	109.96
40	14733	×	×	×	229.05
45	23941	×	×	×	471.60
50	29193	×	×	×	802.68

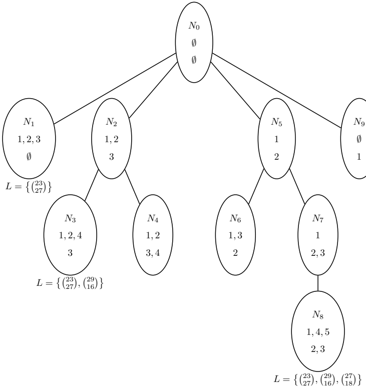
Two Phase Algorithms

MOCO problem	Phase 1 Approach	Phase 2 Approach	Reference
Bi-objective integer network flow	Parametric	Local search	[20]
Bi-objective integer network flow	Parametric	Local search	[43]
Bi-objective integer network flow	Parametric	Ranking	[36]
Bi-objective assignment	Dichotomic	Variable fixing	[51]
Bi-objective assignment	Dichotomic	Variable fixing	[49]
Bi-objective assignment	Dichotomic	Ranking	[32]
Three-objective assignment	Dichotomic	Ranking	[33, 34]
Bi-objective multimodal assignment	Dichotomic	Ranking	[31]
Bi-objective spanning tree	Dichotomic	Ranking, branch and bound	[46]
Bi-objective shortest path	Parametric	Label correcting	[26]
Bi-objective shortest path	Dichotomic	Label correcting, label setting	[35]
	Parametric	Ranking	
Bi-objective knapsack	Dichotomic	Branch and bound	[53]
Bi-objective knapsack	Dichotomic	Ranking	[14]
Three-objective knapsack	Dichotomic	Ranking	[14]

Overview

Multi-objective Branch and Bound

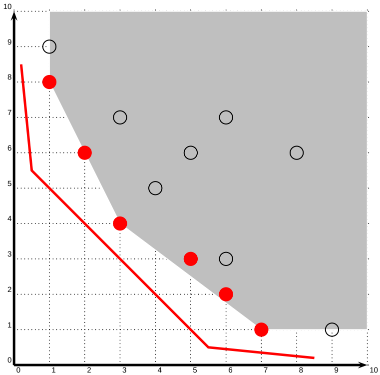
- Branching: As in single objective case
- Bounding: Ideal point of problem at node is dominated by efficient solution
- Branching may be very ineffective
- Use lower and upper bound sets



Bound Sets

Ehrgott and Gandibleux 2005:

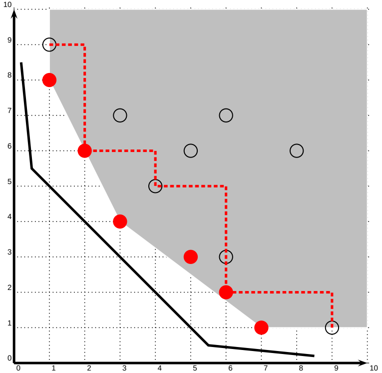
1. Lower bound set L
   - is $\mathbb{R}^p_{\geqq}$-closed
   - is $\mathbb{R}^p_{\geqq}$-bounded
   - $Y_N \subset L + \mathbb{R}^p_{\geqq}$
   - $L \subset \left(L + \mathbb{R}^p_{\geqq}\right)_N$
2. Upper bound set U
   - is $\mathbb{R}^p_{\geqq}$-closed
   - is $\mathbb{R}^p_{\geqq}$-bounded
   - $Y_N \in \text{cl}\left[\left(U + \mathbb{R}^p_{\geqq}\right)^c\right]$
   - $U \subset \left(U + \mathbb{R}^p_{\geqq}\right)_N$



Bound Sets

Ehrgott and Gandibleux 2005:

1. Lower bound set L
   - is $\mathbb{R}^p_{\geqq}$-closed
   - is $\mathbb{R}^p_{\geqq}$-bounded
   - $Y_N \subset L + \mathbb{R}^p_{\geqq}$
   - $L \subset \left(L + \mathbb{R}^p_{\geqq}\right)_N$
2. Upper bound set U
   - is $\mathbb{R}^p_{\geqq}$-closed
   - is $\mathbb{R}^p_{\geqq}$-bounded
   - $Y_N \in \mathrm{cl}\left[\left(U + \mathbb{R}^p_{\geqq}\right)^c\right]$
   - $U \subset \left(U + \mathbb{R}^p_{\geqq}\right)_N$



Ingredients of a MOBB Algorithm

- Branching and Node Selection: Branching is the same as single objective case, but branching rules often use objective function information (e.g. knapsack problems)
- Bounding and Fathoming Nodes: Terminate as soon as upper bound set $U = Y_N$ and fathom if
  1. The subproblem has an empty feasible set (infeasibility);
  2. The non-dominated set Y_N of the subproblem belongs to L (optimality);
  3. For every $l \in L$ there exists some $u \in U$ such that $u \leqq l$ (dominance).

Fathoming

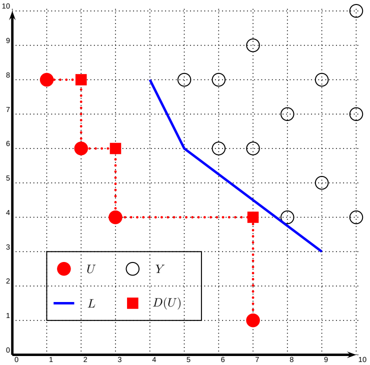


Figure: The node can be fathomed by dominance

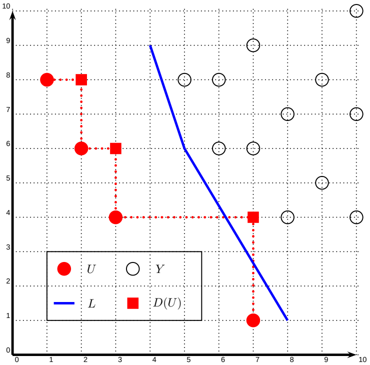


Figure: The node can be fathomed by dominance assuming $Y \subset \mathbb{Z}^p$

Multi-objective Branch and Bound Algorithms

MOCO problem	Upper bound	Lower bound	Branching	Node sel.	Feas. sol.	Reference
MO binary	Incumbent set	Utopia point	Variable fixing	Depth-first	Variable fixing	[16]
BO knapsack	Incumbent set	Utopia point	Variable fixing	Depth-first	Variable fixing	[50]
BO knapsack	Adaptation of [50] restricted to triangles					[53]
BO spanning tree	Incumbent set	Utopia point	Edge fixing	Depth-first	At leaves	[38]
BO spanning tree	Incumbent set	Convex relaxation	Edge fixing	Depth-first	Convex relaxation at nodes	[45]
MO knapsack	Incumbent set	Ideal point LP relaxation	Variable fixing	Depth-first	At leaves	[11]
TO knapsack	Incumbent set	Utopia point	Variable fixing	Depth-first	At leaves	[14]
TO knapsack	Incumbent set	Convex relaxation	Variable fixing	Lexicographic order of 5 criteria	Part of UB	[14]
BO assignment	Adaptation of [45] restricted to triangles					[7]
BO flow shop	No details	No details	No details	Depth-first	No details	[25]
BO mixed integer	Incumbent set	Ideal point of LP relaxation	Variable fixing	Depth-first	LP at leaves	[22, 23]
BO mixed integer	Extended incumbent set	Ideal point of LP relaxation	Variable fixing	Depth-first	LP at leaves	[52]

Overview

Conclusion

- Summarised main approaches used within exact methods for MOCO
- Challenge 1: Understand importance of objective function values
- Challenge 2: Polyhedral theory for MOCO scalarization
- Challenge 3: Toolbox to build exact algorithms for challenging problems

H. Aissi, R. Mahjoub, S. T. McCormick, and M. Queyranne.
A strongly polynomial time algorithm for multicriteria global minimum cuts.
In J. Lee and J. Vygen, editors, *Integer Programming and Combinatorial Optimization. 17th International Conference, IPCO 2014, Bonn, Germany, June 23-25, 2014, Proceedings*, volume 8494 of *Lecture Notes in Computer Science*, pages 25–36. Springer, 2014.

J.-F. Bérubé, F. Gendreau, and J.-Y. Potvin.
An exact ε-constraint method for bi-objective combinatorial optimization problems: Application to he travelling slaesman problem with profits.
European Journal of Operational Research, 194:39–50, 2009.

P.M. Camerini, G. Galbiati, and F. Maffioli.
The complexity of multi-constrained spanning tree problems.
In L. Lovasz, editor, *Theory of Algorithms*, pages 53 – 101. North-Holland, 1984.

L.G. Chalmet, L. Lemonidis, and D.J. Elzinga.
An algorithm for the bi-criterion integer programming problem.
European Journal of Operational Research, 25:292–300, 1986.

K. Dächert, J. Gorski, and K. Klamroth.
An augmented weighted tchebycheff method with adaptively chosen parameters for discrete bicriteria optimization problems.
Computers & Operations Research, 39:2929–2943, 2012.

K. Dächert and K. Klamroth.
A linear bound on the number of scalarizations needed to solve discrete tricriteria optimization problems.
Journal of Global Optimization, 61:643–676, 2015.

C. Delort.

Algorithmes d'énumération implicite pour l'optimisation multi-objectifs: Exploitation d'ensembles bornant et application aux problèmes de sac à dos et d'affectation.
PhD thesis, Université Pierre et Marie Curie (Paris VI), 2011.

M. Ehrgott.
On matroids with multiple objectives.
Optimization, 38:73–84, 1996.

M. Ehrgott and D.M. Ryan.
Constructing robust crew schedules with bicriteria optimization.
Journal of Multi-Criteria Decision Analysis, 11:139–150, 2002.

A. Eusébio and J.R. Figuiera.
Finding non-dominated solutions in bi-objective integer network flow problems.
Computers & Operations Research, 36:2554–2564, 2009.

K. Florios, G. Mavrotas, and D. Diakoulaki.
Solving multiobjective, multiconstraint knapsack problems using mathematical programming and evolutionary algorithms.
European Journal of Operational Research, 203:14–21, 2010.

H.W. Hamacher and G. Ruhe.
On spanning tree problems with multiple objectives.
Annals of Operations Research, 52:209–230, 1994.

P. Hansen.
Bicriterion path problems.
In G. Fandel and T. Gal, editors, *Multiple Criteria Decision Making Theory and Application*, volume 177 of *Lecture Notes in Economics and Mathematical Systems*, pages 109–127. Springer, 1979.

J. Jorge.
Nouvelles propositions pour la résolution exacte du sac à dos multi-objectif unidimensionnel en variables binaires.
PhD thesis, Université de Nantes, 2010.

G. Kirlik and S. Sayin.
A new algorithm for generating all nondominated solutions of multiobjective discrete optimization problems.
European Journal of Operational Research, 232:479–488, 2014.

G. Kiziltan and E. Yucaoglu.
An algorithm for multiobjective zero-one linear programming.
Management Science, 29:1444–1453, 1983.

K. Klamroth, R. Lacour, and D. Vanderpooten.
On the representation of the search region in multi-objective optimization.
European Journal of Operational Research, 2015.

D. Klein and E. Hannan.
An algorithm for the multiple objective integer linear programming problem.
European Journal of Operational Research, 9:378–385, 1982.

M. Laumanns, L. Thiele, and Zitzler E.
An efficient, adaptive parameter variation scheme for metaheuristics based on the epsilon-constraint method.
European Journal of Operational Research, 169:932–942, 2006.

H. Lee and P.S. Pulat.
Bicriteria network flow problems: Integer case.
European Journal of Operational Research, 66:148–157, 1993.

M. Lokman, B.and Köksalan.
Finding all nondominated points of multi-objective integer programs.
Journal of Global Optimization, 57:347–365, 2013.

G. Mavrotas and D. Diakoulaki.
A branch and bound algorithm for mixed zero-one multiple objective linear programming.
European Journal of Operational Research, 107(3):530–541, 1998.

G. Mavrotas and D. Diakoulaki.
Multi-criteria branch and bound: A vector maximization algorithm for mixed 0-1 multiple objective linear programming.
Applied Mathematics and computation, 171(3):53–71, 2005.

K. Mavrotas, G.and Florios.
An improved version of the augmented ε-constraint method (augmecon2) for finding the exact pareto set in multi-objective integer programming problems.
Applied Mathematics and Computation, 219:9652–9669, 2013.

M. Mezmaz, N. Melab, and E.G. Talbi.
A grid-based parallel approach of the multi-objective branch and bound.
In *15th EUROMICRO International Conference on Parallel, Distributed and Network-Based Processing (PDP07)*, 2007.

J. Mote, I. Murthy, and D.L. Olson.
A parametric approach to solving bicriterion shortest path problems.
European Journal of Operational Research, 53:81–92, 1991.

P. Neumayer.
Complexity of optimization on vectorweighted graphs.

In A. Bachem, U. Derigs, M. Jünger, and R. Schrader, editors, *Operations Research 93*, pages 359–361. Physica Verlag, Heidelberg, 1994.

M. Özlen and M. Azizoglu.
Multi-objective integer programming: A general approach for generating all non-dominated solutions.
European Journal of Operational Research, 199:25–35, 2009.

M. Ozlen, B.A. Burton, and C.A.G. MacRae.
Multi-objective integer programming: An improved recursive algorithm.
Journal of Optimization Theory and Applications, 160:470–482, 2014.

L. Paquete, M. Jaschob, K. Klamroth, and J. Gorksi.
On a biobjective search problem in a line: Formulations and algorithms.
Theoretical Computer Science, 507:61–71, 2013.

C. R. Pedersen, L. R. Nielsen, and K. A. Andersen.
The bicriterion multi modal assignment problem: Introduction, analysis, and experimental results.
INFORMS Journal on Computing, 20:400–411, 2008.

A. Przybylski, X. Gandibleux, and M. Ehrgott.
Two phase algorithms for the bi-objective assignment problem.
European Journal of Operational Research, 185:509–533, 2008.

A. Przybylski, X. Gandibleux, and M. Ehrgott.
A recursive algorithms for finding all nondominated extreme points in the outcome set of a multiobjective integer program.
INFORMS Journal on Computing, 22:371–386, 2010.

A. Przybylski, X. Gandibleux, and M. Ehrgott.

A two phase method for multi-objective integer programming and its application to the assignment problem with three objectives.
Discrete Optimization, 7:149–165, 2010.

A. Raith and M. Ehrgott.
A comparison of solution strategies for biobjective shortest path problems.
Computers & Operations Research, 36:1299–1331, 2009.

A. Raith and M. Ehrgott.
A two-phase algorithm for the biobjective integer minimum cost flow problem.
Computers & Operations Research, 36:1945–1954, 2009.

T.K. Ralphs, M.J. Saltzmann, and M.M. Wiecek.
An improved algorithm for solving biobjective integer programs.
Annals of Operations Research, 147:43–70, 3006.

R.M. Ramos, S. Alonso, J. Sicilia, and C. González.
The problem of the optimal biobjective spanning tree.
European Journal of Operational Research, 111:617–628, 1998.

G. Ruhe.
Complexity results for multicriteria and parametric network flows using a pathological graph of Zadeh.
Zeitschrift für Operations Research, 32:9–27, 1988.

S. Ruzika.
On Multipe Objective Combinatorial Optimization.
PhD thesis, Department of Mathematics, Technical University of Kaiserslautern, 2007.

S. Ruzika and H. W. Hamacher.

A survey on multiple objective minimum spanning tree problems.
In D. Lerner, D. Wagner, and K. A. Zweig, editors, *Algorithmics*, volume 5515 of *Lecture Motes in Computer Science*, pages 104–116. Springer, 2009.

S. Sayin and P. Kouvelis.
The multiobjective discrete optimization problem: A weighted min-max two-stage optimization approach and a bicriteria algorithm.
Management Science, 51:1572–1581, 2005.

A. Sedeño-Noda and C. González-Martín.
An algorithm for the biobjective integer minimum cost flow problem.
Computers & Operations Research, 28:139–156, 2001.

P. Serafini.
Some considerations about computational complexity for multi objective combinatorial problems.
In J. Jahn and W. Krabs, editors, *Recent Advances and Historical Development of Vector Optimization*, volume 294 of *Lecture Notes in Economics and Mathematical Systems*, pages 222–232. Springer, 1986.

F. Sourd and O. Spanjaard.
A multiobjective branch-and-bound framework: Application to the biobjective spanning tree problem.
INFORMS Journal on Computing, 20:472–484, 2008.

S. Steiner and T. Radzik.
Computing all efficient solutions of the biobjective minimum spanning tree problem.
Computers & Operations Research, 35:198–211, 2008.

J. Sylva and Crema A.
A method for finding the set of non-dominated vectors for multiple objective integer linear programs.
European Journal of Operational Research, 158:46–55, 2004.

B. Tam, D. Ryan, and M. Ehrgott.
Multi-objective approaches to the unit crewing problem in airline crew scheduling.
Journal of Multi-criteria Decision Analysis, 21:257–277, 2014.

D. Tuyttens, J. Teghem, P. Fortemps, and K. Van Nieuwenhuyse.
Performance of the MOSA method for the bicriteria assignment problem.
Journal of Heuristics, 6:295–310, 2000.

E. Ulungu and J. Teghem.
Solving multi-objective knapsack problem by a branch-and-bound procedure.
In J. Climaco, editor, *Multicriteria Analysis*, pages 269–278. Springer, 1997.

E.L. Ulungu and J. Teghem.
The two phases method: An efficient procedure to solve bi-objective combinatorial optimization problems.
Foundations of Computing and Decision Sciences, 20:149–165, 1995.

T. Vincent, F. Seipp, S. Ruzika, A. Przybylski, and X. Gandibleux.
Multiple objective branch and bound for mixed 0-1 linear programming: Corrections and improvements for the biobjective case.
Computers & Operations Research, 40:498–509, 2013.

M. Visée, J. Teghem, M. Pirlot, and E.L. Ulungu.
Two-phases method and branch and bound procedures to solve the bi-obective knapsack problem.
Journal of Global Optimization, 12:139–155, 1998.