Solving ILPs with three and more objectives

Recent Advances in Multi-Objective Optimization

Uni Wien

12 Sep 2014

Axel Werner

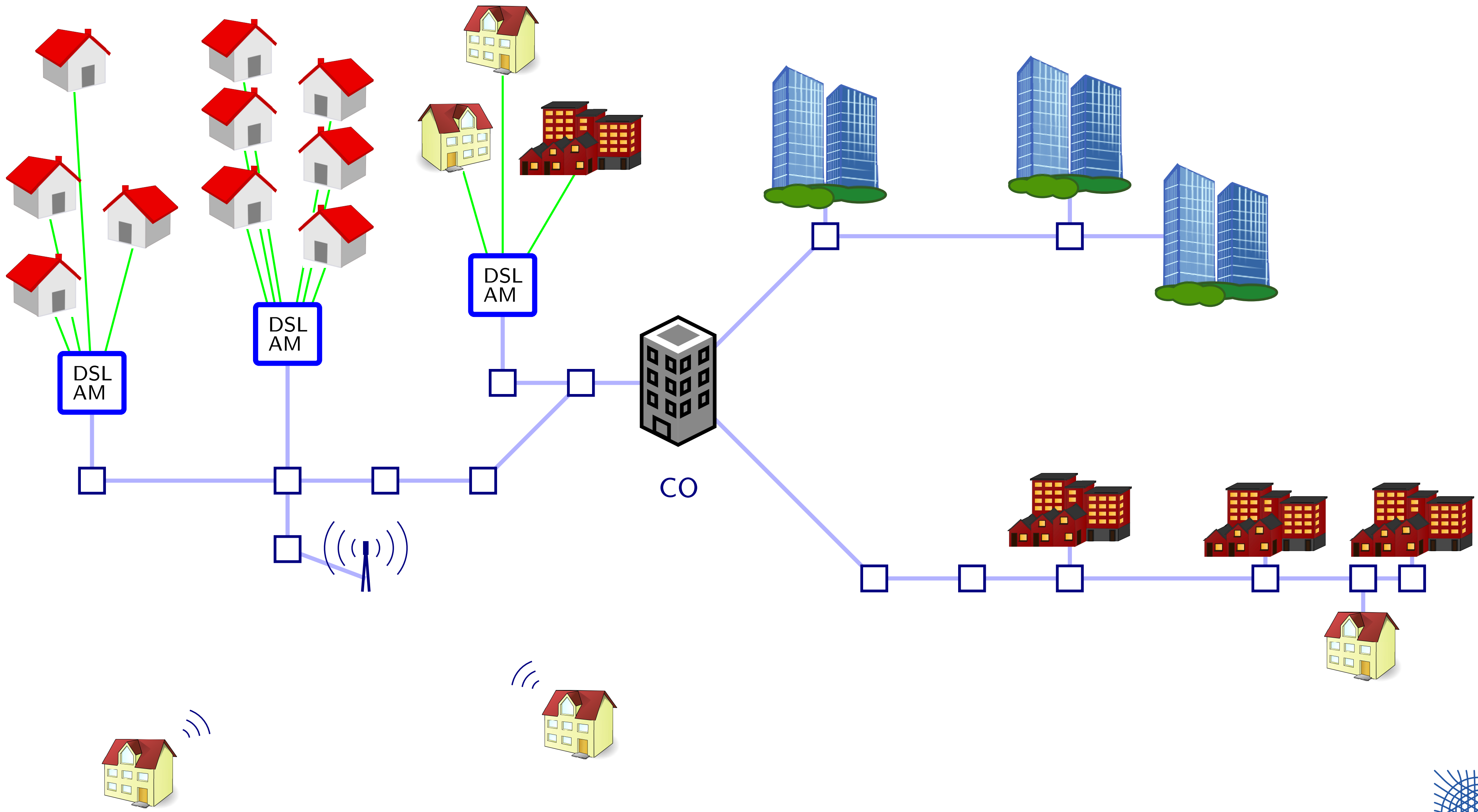
Zuse-Institut Berlin

Overview

- Original Problem
  - ➡ k-Architecture Connected Facility Location
  - ➡ Multiobjective Problem Formulation
- Multiobjective Framework for solving IP Problems with Objective Space Methods
- Solution Methods for 3 objectives
  - ➡ Epsilon Method
  - ➡ Box-Split Algorithm
- (Preliminary) Computations
- Outlook

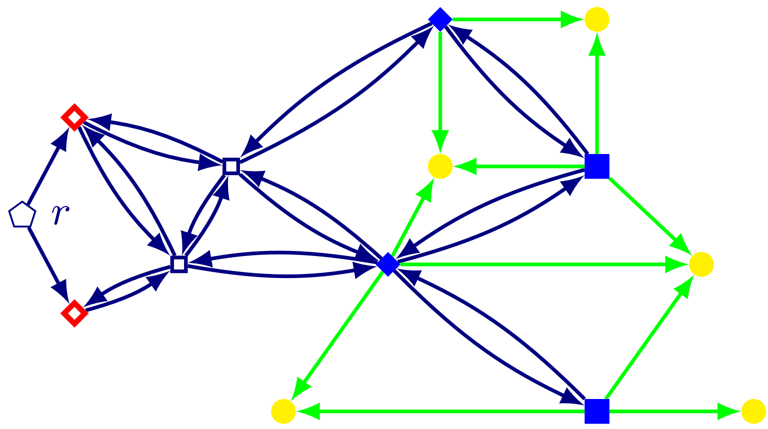
Joint work with Ivana Ljubić, Markus Leitner, Markus Sinnl

▷ Planning an access network using mixed technologies: fiber, copper & wireless



▷ Given

- directed graph (V, A) with root r, potential COs, potential facilities $F^1, \ldots, F^k$, Steiner nodes, and customers C, core arcs (for laying out fibers), and assignment arcs

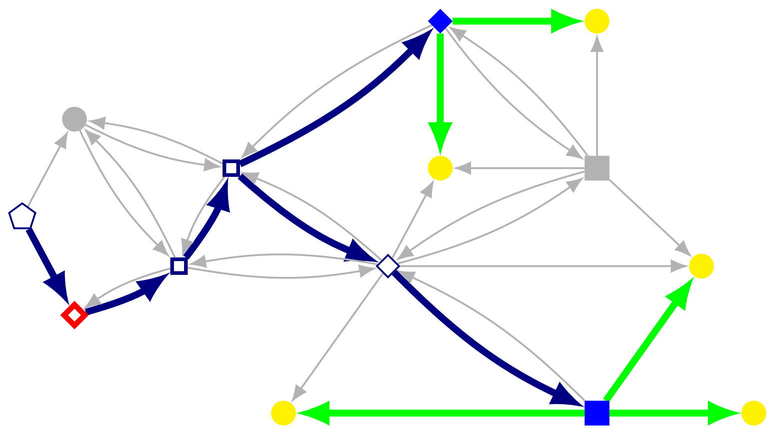


- costs for trenching edges, opening COs and facilities, and connecting customers architecture l), and c^l_{ij} for connecting customer j to facility i (using architecture l)

- demand values $d_j \ \forall j \in C$, with $D := \sum_{j \in C} d_j$

▷ Find

- an assignment of customers to open facilities
- an arborescence connecting open facilities to the root



variables		sets	
x_a	trenching on core arc a	$1,\ldots,k$	architectures
y_i^l	facility i used for architecture l	C	customers
z_j^l	customer j connected via architecture l	F^l	facilities for architecture l
x_{ij}^l	assignment of customer j to facility i	A_c	core arcs

$$\sum_{l=1}^{k} z_j^l \le 1 \qquad \forall j \in C$$

$$\sum_{i \in F_j^l} x_{ij}^l = z_j^l \qquad \forall j \in C,\ l = 1,\ldots,k$$

$$x_{ij}^l \le y_i^l \qquad \forall j \in C,\ i \in F_j^l,\ l = 1,\ldots,k$$

$$x(\delta^-(W)) \ge y_i^l \qquad \forall W \subseteq V \setminus C,\ i \in F^l \cap W,\ l = 1,\ldots,k$$

$$x_a, y_i^l, z_j^l \in \{0,1\} \qquad \forall a \in A_c,\ i \in F^l,\ j \in C,\ l = 1,\ldots,k$$

$$x_{ij}^l \in \{0,1\} \qquad \forall i \in F_j^l,\ j \in C,\ l = 1,\ldots,k$$

$$\min \sum_{(i,j)\in A_{rc}} c_{ij}x_{ij} + \sum_{l=1}^{k} \sum_{(i,j)\in A^l} c^l_{ij}x^l_{ij} + \sum_{l=1}^{k}\sum_{i\in F^l} c^l_i y^l_i \quad =: f_1 \qquad \text{(cost)}$$

$$\min \quad D - \sum_{j\in C} d_j z^1_j \quad =: f_2 \qquad \text{(total demand } \textit{not} \text{ covered by architecture 1)}$$

$$\min \quad D - \sum_{l=1}^{k}\sum_{j\in C} d_j z^l_j \quad =: f_3 \qquad \text{(total demand } \textit{not} \text{ covered by } \textit{any} \text{ architecture)}$$

$$\sum_{l=1}^{k} z^l_j \le 1 \qquad \forall j \in C$$

$$\sum_{i\in F^l_j} x^l_{ij} = z^l_j \qquad \forall j \in C,\ l = 1,\dots,k$$

$$x^l_{ij} \le y^l_i \qquad \forall j \in C,\ i \in F^l_j,\ l = 1,\dots,k$$

$$x(\delta^-(W)) \ge y^l_i \qquad \forall W \subseteq V \setminus C,\ i \in F^l \cap W,\ l = 1,\dots,k$$

$$x_a, y^l_i, z^l_j \in \{0,1\} \qquad \forall a \in A_c,\ i \in F^l,\ j \in C,\ l = 1,\dots,k$$

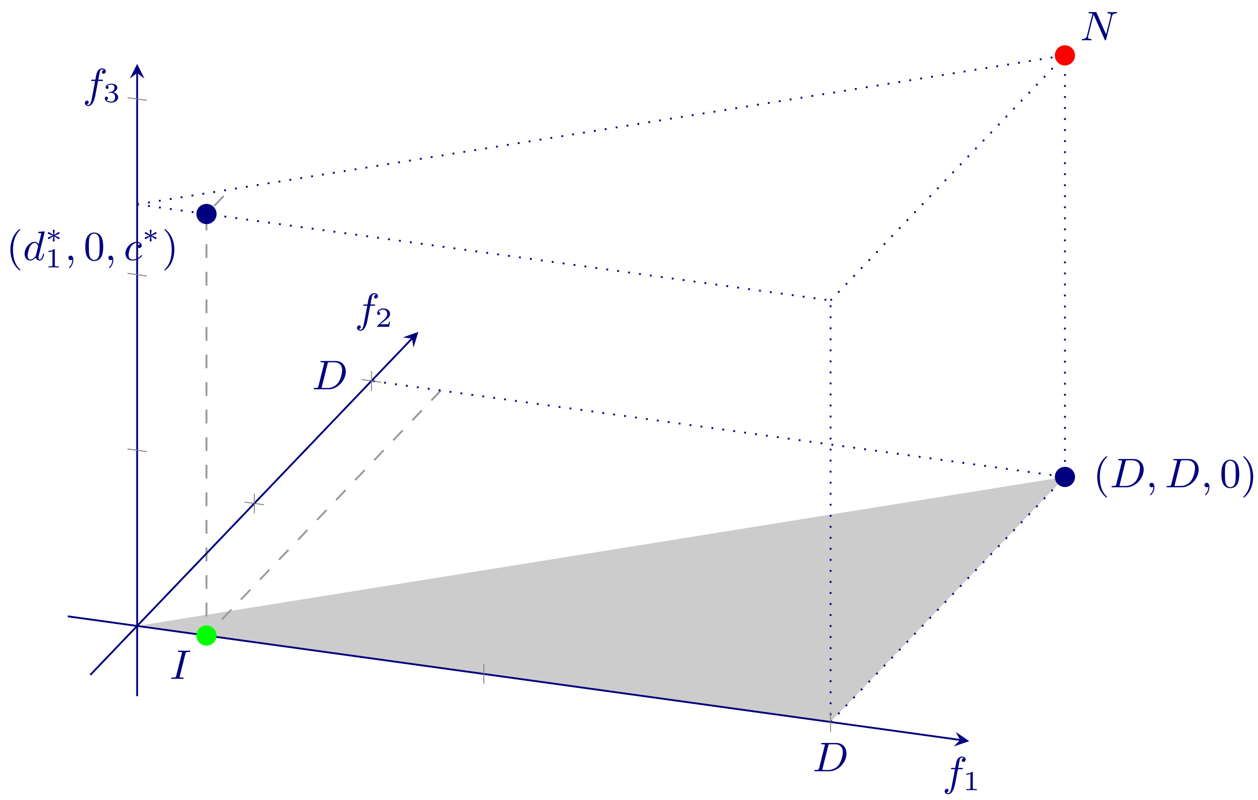
$$x^l_{ij} \in \{0,1\} \qquad \forall i \in F^l_j,\ j \in C,\ l = 1,\dots,k$$

▷ Assumptions:

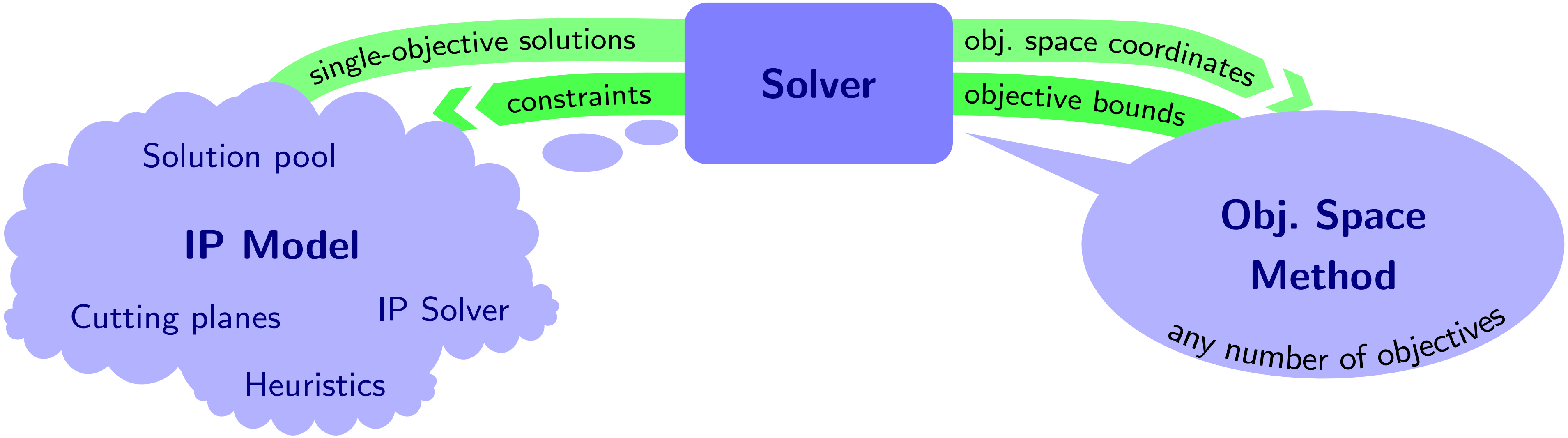
- Every customer can be reached by some architecture
- All assignment costs are strictly positive

➡ Non-dominating solutions always have $f_1 \geq f_2$

➡ Ideal and nadir point can be computed by trivial solution and fixing some variables in a suitably adapted model

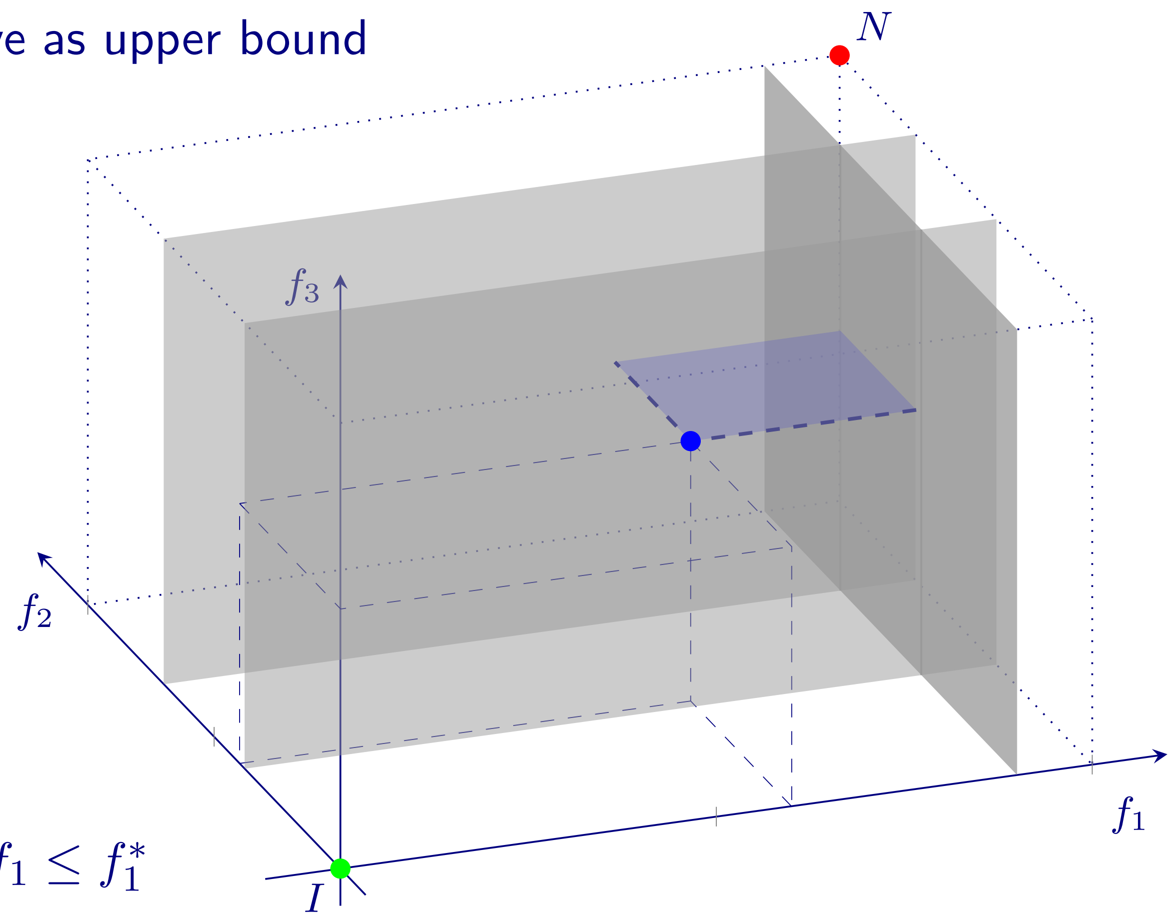


- Objective Space Methods: find the Pareto front by repeatedly restricting objective space
- Solution Framework

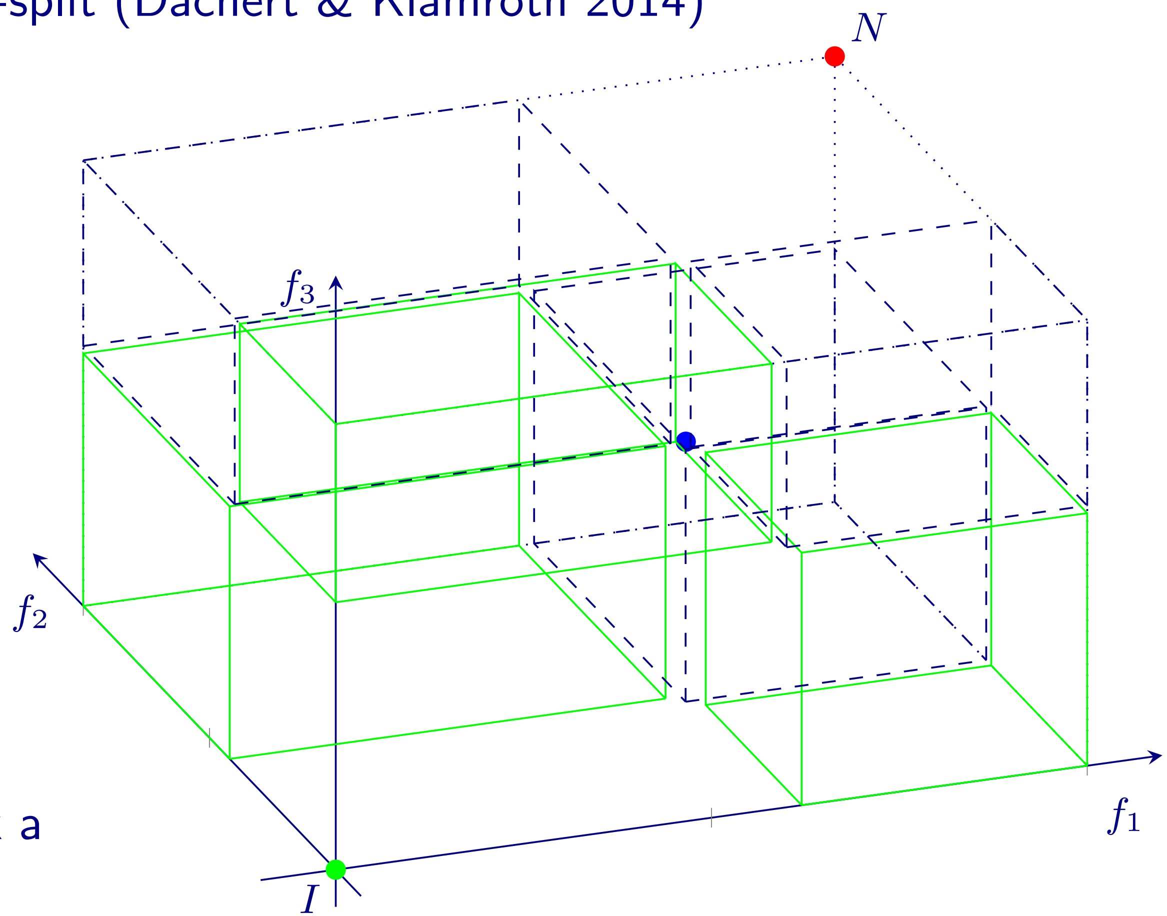


- kArchConFL | ε-constraint method | box-splitting algorithm
- Similar methods for 3-objective problems: Laumanns 2004, Özlen & Azizoğlu 2009, Dhaenens et al 2010, Boland et al 2014, Dächert & Klamroth 2014, ...

- ε-constraint method for 2 objectives (Haimes et al 1971)
  - ➡ iteratively solve with one objective as upper bound
- Generalization (for 3 objectives)
  - ➡ maintain upper bounds
    $\bar{f}_1$ (decreasing by ε_1 throughout the algorithm)
    $\bar{f}_2$ (decreasing by ε_2 within each f_1-iteration)
  - ➡ lexmin solve in each iteration:
    (i) $f_3^* = \min f_3$
    (ii) $f_1^* = \min f_1$, with $f_3 \leq f_3^*$
    (iii) $f_2^* = \min f_2$, with $f_3 \leq f_3^*, f_1 \leq f_1^*$
    - ➡ solution at point (f_1^*, f_2^*, f_3^*) with validity range $(\bar{f}_1, \bar{f}_2)$
  - ➡ skip solve if there is a solution from a previous iteration with validity range covering the current upper bounds



- ▷ Similar 3-objective approach: Full m-split (Dächert & Klamroth 2014)



- ▷ In each iteration consider a box
  $[f_1^-, f_1^+] \times [f_2^-, f_2^+] \times [f_3^-, f_3^+]$
  - ➡ weighted sum solve with positive weights
  - ➡ found solution splits the box into (max.) 3 high-priority boxes, (max.) 3 low-priority boxes
- ▷ Only high-priority boxes are searched
  - ➡ if no high-priority box is left, pick a low-priority box B and:
    - dismiss B, if dominated by a previously found solution;
    - slice B into (max. 3) new boxes, if partly dominated by a previously found solution;
    - else make B high-priority

▷ Instances, solutions and runtimes (sec.):

$\|V\|$	$\|A\|$	$\|C\|$	$\|F\|$	non-dom. solutions	ε-constraint	box-splitting
14	30	4	7	21	1.26	0.08
34	100	10	13	95	5.97	13.8
27	51	12	13	91	0.76	0.75
116	369	36	44	3285	323819	365682
39	77	16	18	233	10.8	15.0
84	260	26	31	1564	12675	19899
53	109	21	23	431	89.6	92.0
96	295	31	37	2211	47003	73976
48	103	16	19	362	85.9	77.3
424	1208	39	55	10463	1372050	...

- Extend methods to more than 3 objectives
  - ➡ relatively straightforward for ε-constraint method
  - ➡ more tricky for box-splitting algorithm: splitting yields $2^d - 2$ new boxes
- Incorporate more of IP solving repertoire
  - ➡ re-use feasible solutions
  - ➡ cut pool, branching priorities
- Plug in other problems that are solvable by IP
  - ➡ Prize-collecting Steiner Tree
- Include hypervolume computation (cf. Boland et al 2013, Zitzler et al 2003)
  - ➡ approximation of the Pareto front
  - ➡ adapt the multiobjective methods accordingly