Universität Konstanz

Computing enclosures for multi-objective mixed-integer nonconvex optimization problems with application to energy supply networks

Moritz Link and Stefan Volkwein (Universität Konstanz)

3SMINLP, IMUS – March 31st, 2023

1 Motivation – energy supply networks

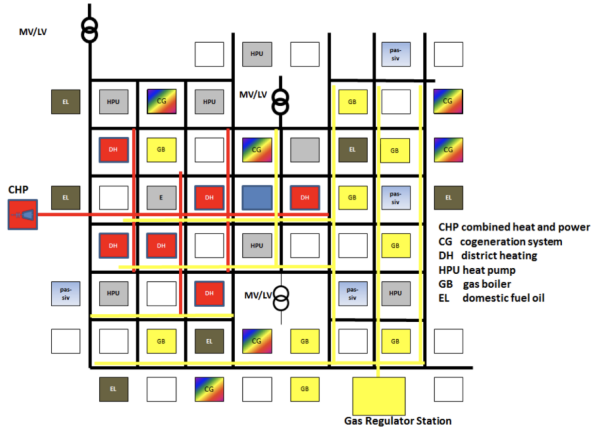
Fachbereich Mathematik und Statistik

Transferplattform "Energiewende vor Ort"

Universität Konstanz

Forschung REVISION StadtKlima Personen und Einrichtungen

Home of Math. & Stat. > Transferplattform "Energiewende vor Ort"



Transferplattform "Energiewende vor Ort – Optimale dezentrale Energieversorgung und Kommunikationsstrategien hinsichtlich politischer und sozialer Akzeptanz"

Forschungsprogramm und langfristiges Forschungsziel

REVISION StadtKlima

Reallabor für EnergieVersorgung: Innovative Strategien und Interdisziplinäre Optimierung mit Nebenbedingungen

Kontakt

Prof. Dr. Stefan Volkwein
Fachbereich Mathematik und Statistik

E-Mail schreiben

in cooperation with

Rechenzentrum für Versorgungsnetze Wehr GmbH

Energy supply network optimization – goal

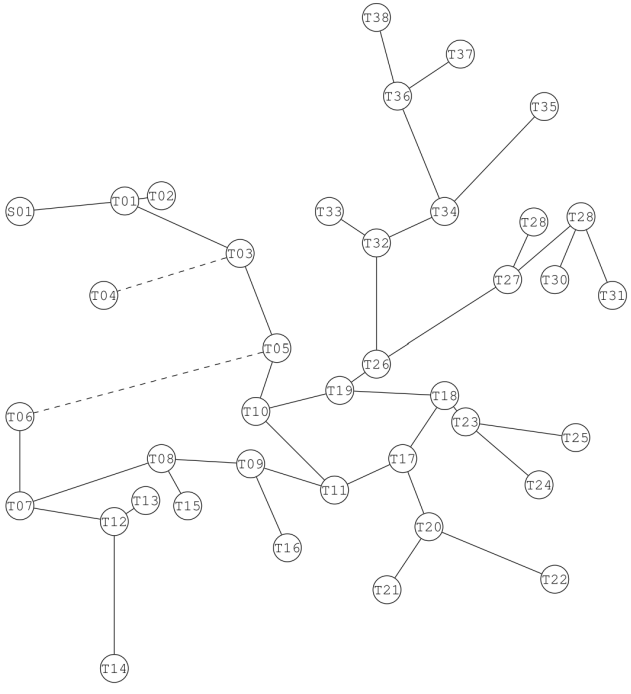


Figure: part of the district *Paradies, Konstanz*; [Lu, 2023, Figure 3.5]

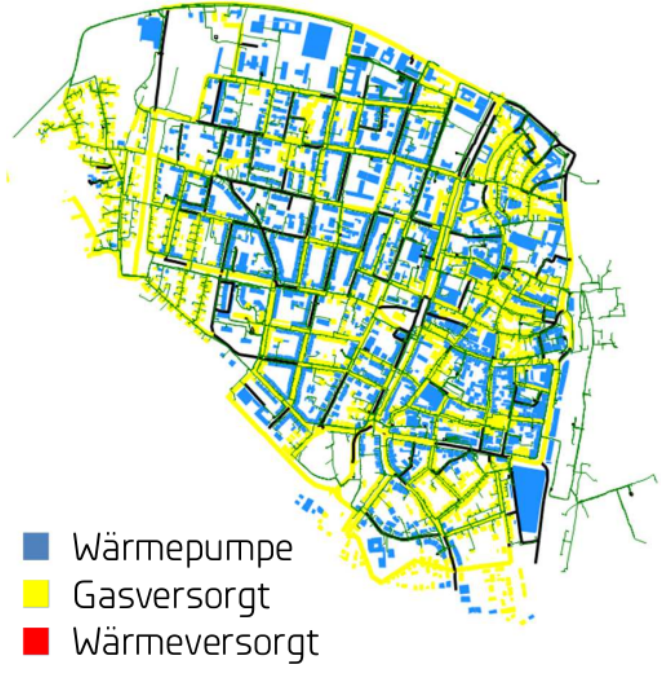


Figure: exemplary network plan for *Konstanz (left bank)*; computations and visualization by RZVN

Goal: find an energy supply network plan which is optimal with respect to

- ▷ operational and investment costs;
- ▷ CO_2-emissions.

⇒ we obtain a **multi-objective optimization problem** due to conflicting objectives

Lu, J., (2023). *Mixed-Integer Nonlinear Modeling and Optimization of Designing Decentralized Energy Supply Networks*, Universität Konstanz.

Energy supply network optimization – constraints

Allow: choices and decisions

- ▷ choose energy carrier individually;
- ▷ individual decisions for energy-efficient house renovations.

Require: technical viability

- ▷ covering of energy demands of all houses;
- ▷ respecting (stationary) physical flow equations.

This leads to constraints of the form

$$\sum_{t \in T} x_{i,j}^t = 1, \quad x_{i,j}^t \in \{0,1\},$$

$$s_{i,j}^{\mathsf{Esum}} = \sum_{t \in T} \mathsf{SUMELOAD}_{i,j}^t x_{i,j}^t$$

This leads to constraints of the form

$$R_{i,j}^{\mathsf{e}} f_{\mathsf{in},i,j}^{\mathsf{e}} = a^{\mathsf{e}}(u_j u_i - u_j^2)$$

$$R_{i,j}^{\mathsf{g}} \left(f_{i,j}^{\mathsf{g}}\right)^2 \leq \left(a^{\mathsf{g}}\right)^2 (\bar{p}_{\max} y_{i,j})$$

$\Rightarrow$ we obtain a **multi-objective mixed-integer nonconvex optimization problem** of the form:

$$\min f(x) \quad \text{s.t.} \quad e(x) = 0,\ g(x) \leq 0,\ x \in X = X_C \times X_I, \qquad \text{(MOP)}$$

where $f = (f_1, \ldots, f_k)^\top$ linear, $e = (e_1, \ldots, e_p)^\top$, $g = (g_1, \ldots, g_q)^\top$ polynomial and $X_C = [\ell_C, u_C] \subseteq \mathbb{R}^n$ and $X_I = [\ell_I, u_I] \cap \mathbb{Z}^m$ with compact feasible set S and nondominated set $\mathcal{N}$ w.r.t. $\mathcal{C} = \mathbb{R}_+^k$.

Why is it not straightforward?

- **problem size:** 1570 variables, thereof 360 binaries
- **off-the-shelf solver:** `pyscipopt`-package[1] with standard settings
- **single objective version:** introduce carbon emission **redution rate as constraint**

	reduction rates				
	0	0.2	0.25	0.3	0.4
CPU time (s)	>10800	208	92	82	59

- using different rates mimics the spirit of the **ε-constraint method**

[1] Maher, S., Miltenberger, M., Pedroso, J. P., Rehfeldt, D., Schwarz, R. and Serrano, F., (2016). *PySCIPOpt: Mathematical Programming in Python with the SCIP Optimization Suite*, Mathematical Software – ICMS, Springer International Publishing, p. 301-307.

2 Notions of multi-objective optimization

Efficiency and nondominance

For problems

$$\min\ f(x) \quad \text{s.t.} \quad x \in S \tag{MOP}$$

with $f = (f_1, \ldots, f_k)^T$ we cannot assume that there exists $x^\star \in S$ minimizing all f_i simultaneously.

Definitions

- For $y^1, y^2 \in \mathbb{R}^k$ we say that y^1 *dominates* y^2 if
$$y^1 \leq y^2 \text{ and } y^1 \neq y^2.$$
- $\bar{x} \in S$ is called *efficient solution* of (MOP) if there exists no $x \in S$ such that $f(x)$ dominates $f(\bar{x})$.
- We call
$$\mathcal{N} = \left\{ f(x) \in \mathbb{R}^k \mid x \in S \text{ is efficient} \right\}$$
the *set of nondominated points (or Pareto front) of* (MOP).

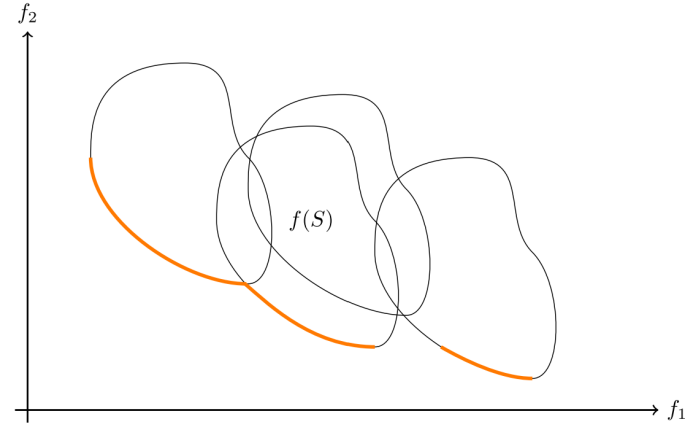


Figure: Pareto front of biobjective (MOP), cf. [Eichfelder and Warnow, 2021]

Eichfelder, G. and Warnow, L. (2021). *A hybrid path decomposition approach to compute an enclosure for multiobjective mixed-integer convex optimization problems.*

Enclosure of $\mathcal{N}$ and its width

Definitions (cf. [Eichfelder et al., 2021])

- Let $L, U \subset \mathbb{R}^k$ be finite with

$$\mathcal{N} \subseteq L + \mathbb{R}^k_+ \text{ and } \mathcal{N} \subseteq U - \mathbb{R}^k_+.$$

Then L is called **lower bound set**, U is called **upper bound set**, and

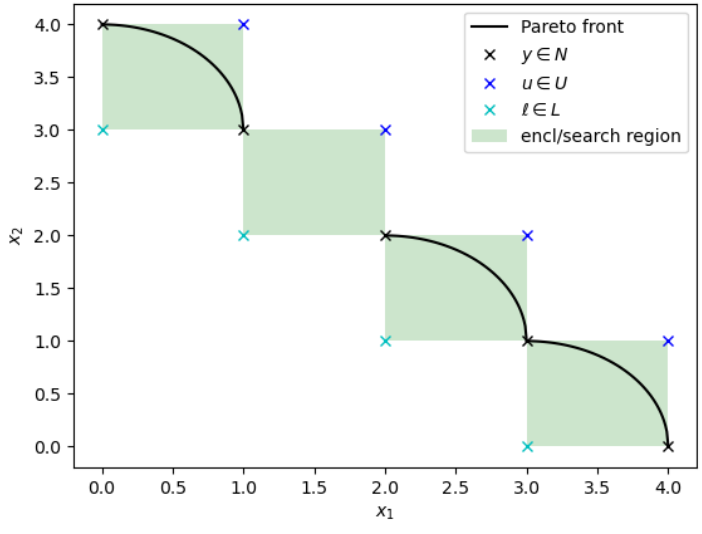
$$E = E(L, U) = \bigcup_{\ell \in L} \bigcup_{u \in U, \ell \leq u} [\ell, u] \supseteq \mathcal{N}$$

is called an **enclosure** of $\mathcal{N}$.

- Given an enclosure E of $\mathcal{N}$, we define its **width** $w(E)$ by

$$\max_{\ell, u} s(\ell, u) \quad \text{s.t.} \quad \ell \in L, u \in U, \ell \leq u,$$

where $s(\ell, u) = \min\{u_i - \ell_i \mid i \in \{1, \ldots, k\}\}$.



Eichfelder, G., Kirst, P., Meng, L., and Stein, O. (2021). *A general branch-and-bound framework for continuous global multiobjective optimization*. J. Global Optim., 80(1):195-227.

Why do we compute enclosures?

Definition

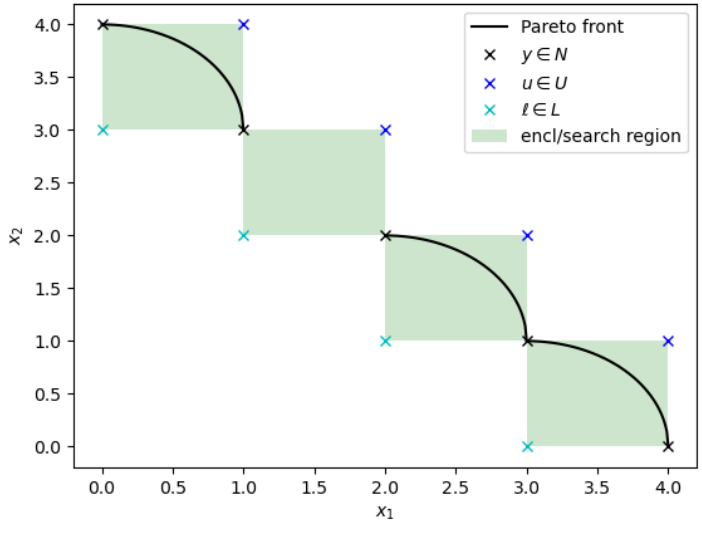
Let $\bar{x} \in S$. We call $f(\bar{x})$ an **ε-nondominated point** of (MOP), if there exists no $x \in S$ such that $f(x) + \varepsilon(1, \ldots, 1)^T$ dominates $f(\bar{x})$.

Remark

Similar to ε-optimality in the single-objective setting.

Lemma ([Eichfelder et al., 2021, Lemma 3.1])

Let E be an enclosure of $\mathcal{N}$ with $w(E) < \varepsilon$ for some $\varepsilon > 0$. Then for any $\bar{x} \in S$ with $f(\bar{x}) \in E$ we have that $f(\bar{x})$ is an ε-nondominated point of (MOP).



Eichfelder, G., Kirst, P., Meng, L., and Stein, O. (2021). *A general branch-and-bound framework for continuous global multiobjective optimization*. J. Global Optim., 80(1):195-227.

3 The toolbox

Basic main scheme

Similar as in [Klamroth et al., 2015] and [Eichfelder and Warnow, 2021]:

- **require:** termination tolerance $\varepsilon_{\text{encl}} > 0$ and a box $B = [z^\ell, z^u]$ with $f(S) \subseteq \text{int}(B)$
- initialize the enclosure as $E = (L, U)$ where $L = \{z^\ell\}$ and $U = \{z^u\}$
- **while** $w(E) \geq \varepsilon_{\text{encl}}$ **do:**
  - **for** all **search regions** **do:**
    - try to **improve** the search region

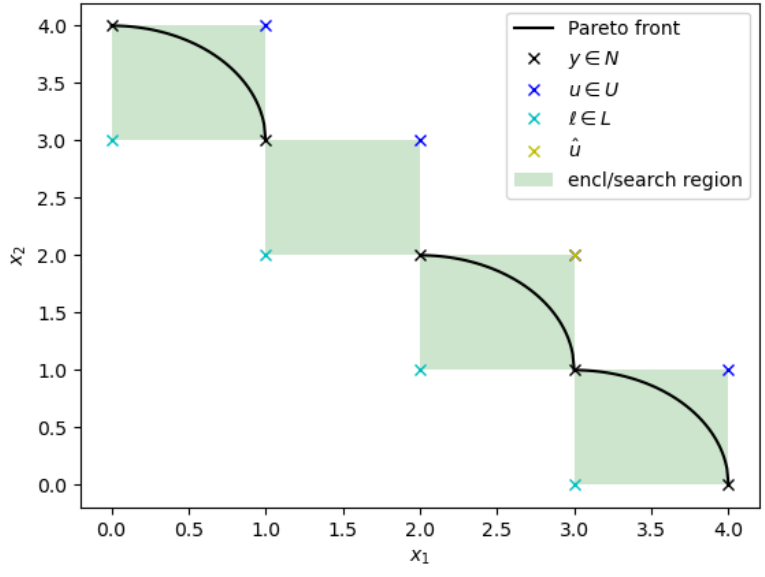
  $\Rightarrow$ decrease width of enclosure
- **return:** enclosure E with $w(E) < \varepsilon_{\text{encl}}$ together with a set of $\varepsilon_{\text{encl}}$-nondominated points N

Klamroth, K., Lacour, R., and Vanderpooten, D. (2015), *On the representation of the search region in multi-objective optimization*. European J. Oper. Res., 245(3):767-778.

Eichfelder, G. and Warnow, L., (2021), *An approximation algorithm for multi-objective optimization problems using a box-coverage*. J. Global Optim.

What is a search region?

- the set U consists of so-called **local upper bounds** [Klamroth et al., 2015]
- depends on a set $N \subseteq f(S)$ of images of feasible points
- the set L consists of so-called **local lower bounds** [Eichfelder and Warnow, 2021]
- depends on a set $\tilde{N}$ of points which are non-dominated by $\mathcal{N}$
- for all nondominated points $y \in \mathcal{N}$ there exist $\ell \in L(\tilde{N})$ and $u \in U(N)$ with $y \in [\ell, u]$



$\Rightarrow$ any local upper bound $u \in U(N)$ determines a **search region**

Klamroth, K., Lacour, R., and Vanderpooten, D. (2015), *On the representation of the search region in multi-objective optimization*. European J. Oper. Res., 245(3):767-778.

Eichfelder, G. and Warnow, L., (2021), *An approximation algorithm for multi-objective optimization problems using a box-coverage*. J. Global Optim.

How to improve a search region?

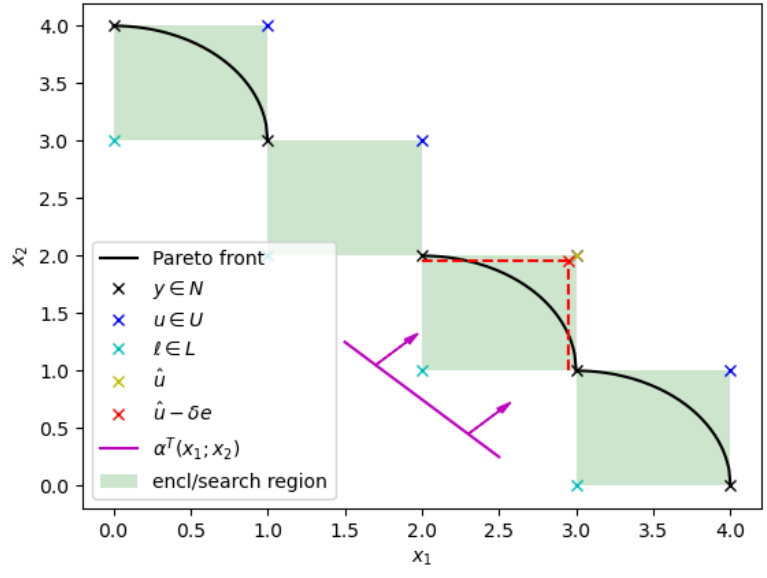
- given a local upper bound $\hat{u} \in U(N)$ we search for a point $x \in S$ with

$$f(x) \leq \hat{u} - \delta e$$

- we use the **restricted Weighted Sum Method**

$$\min \alpha^T f(x) \quad \text{s.t.} \quad x \in S, f(x) \leq \hat{u} - \delta e$$

- if there exists such $x \in S$, we update $N, \tilde{N}, L(\tilde{N})$ and $U(N)$
- if there exists no such $x \in S$, we update $\tilde{N}$ and $L(\tilde{N})$



How to improve a search region?

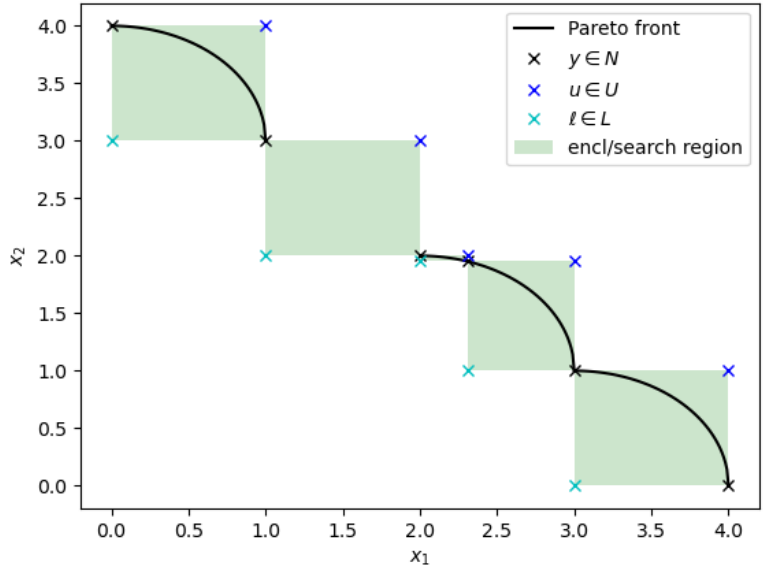
- given a local upper bound $\hat{u} \in U(N)$ we search for a point $x \in S$ with

$$f(x) \leq \hat{u} - \delta e$$

- we use the **restricted Weighted Sum Method**

$$\min \alpha^T f(x) \quad \text{s.t.} \quad x \in S, f(x) \leq \hat{u} - \delta e$$

- if there exists such $x \in S$, we update $N, \tilde{N}, L(\tilde{N})$ and $U(N)$
- if there exists no such $x \in S$, we update $\tilde{N}$ and $L(\tilde{N})$



How to improve a search region?

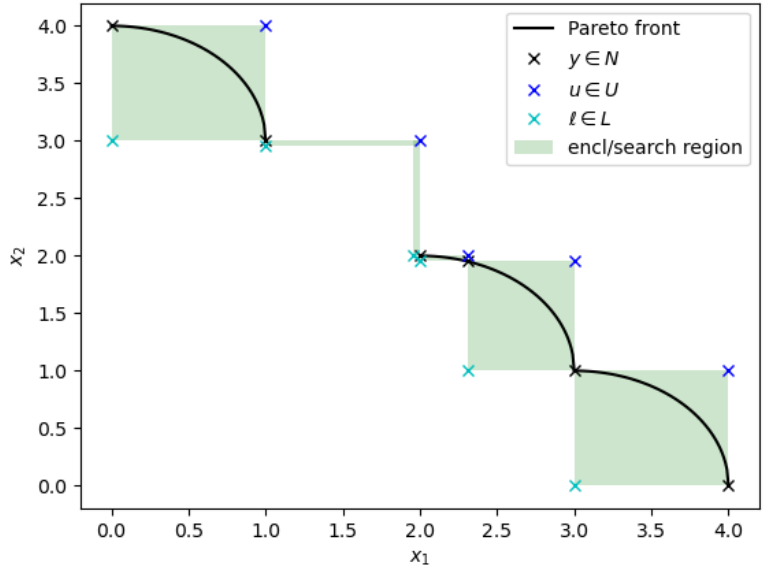
- given a local upper bound $\hat{u} \in U(N)$ we search for a point $x \in S$ with

$$f(x) \leq \hat{u} - \delta e$$

- we use the **restricted Weighted Sum Method**

$$\min \alpha^T f(x) \quad \text{s.t.} \quad x \in S, f(x) \leq \hat{u} - \delta e$$

- if there exists such $x \in S$, we update $N, \tilde{N}, L(\tilde{N})$ and $U(N)$
- if there exists no such $x \in S$, we update $\tilde{N}$ and $L(\tilde{N})$



How to improve a search region?

- given a local upper bound $\hat{u} \in U(N)$ we search for a point $x \in S$ with

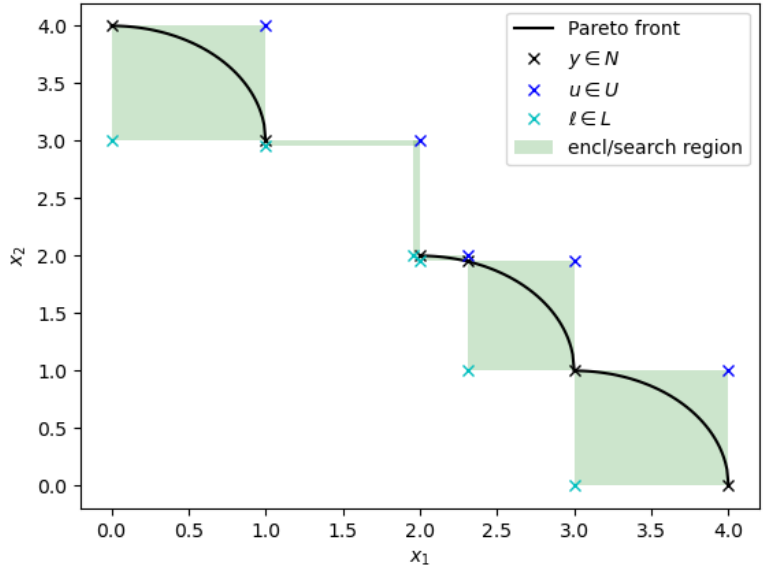
$$f(x) \leq \hat{u} - \delta e$$

- we use the **restricted Weighted Sum Method**

Alert! Complex MINLP!

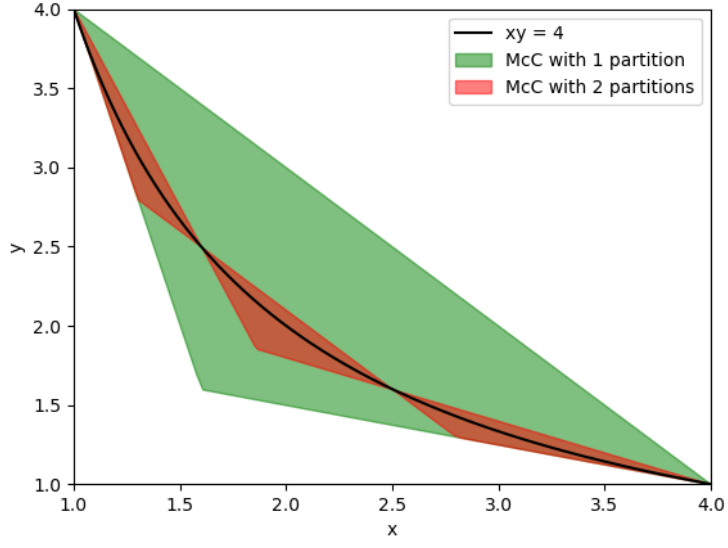
$$\min \alpha^T f(x) \quad \text{s.t.} \quad x \in S, f(x) \leq \hat{u} - \delta e$$

- if there exists such $x \in S$, we update $N, \tilde{N}, L(\tilde{N})$ and $U(N)$
- if there exists no such $x \in S$, we update $L(\tilde{N})$



How to relax?

- in order to avoid solving MINLPs we relax any nonlinear term $h\colon D_h \to \mathbb{R}$ in (MOP)
- we use **uniform McCormick piecewise linear envelopes** $h_u, h_o\colon D_h \to \mathbb{R}$ with $h_u(x) \leq h(x)$ and $h(x) \leq h_o(x)$ for all $x \in D_h$

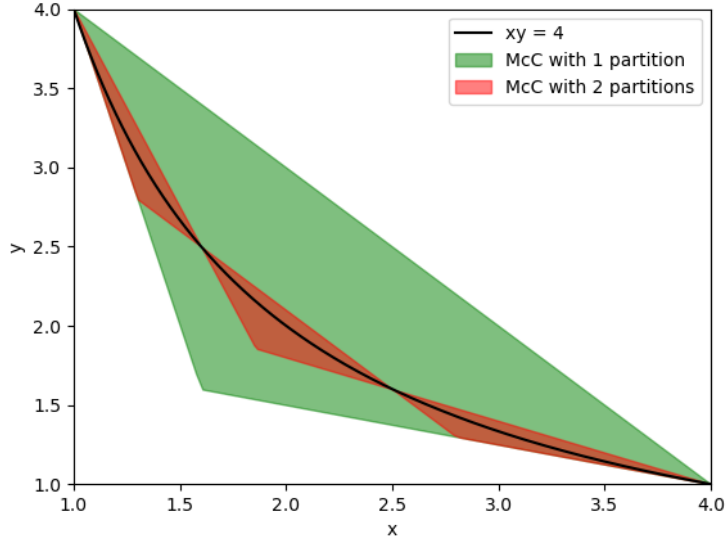


- partition the admissible interval $[x_i^\ell, x_i^u]$ of occuring variables x_i into $r \in \mathbb{N}$ uniform segments $\left[x_i^{\ell_j}, x_i^{u_j}\right], j \in [r]$
- introduce a binary variable for any such segment

McCormick, G. P., (1976), *Computability of global solutions to factorable nonconvex programs. I. Convex underestimating problems.* Math. Programming, 10(2):147-175.

How to relax?

- in order to avoid solving MINLPs we relax any nonlinear term $h\colon D_h \to \mathbb{R}$ in (MOP)
- we use **uniform McCormick piecewise linear envelopes** $h_u, h_o\colon D_h \to \mathbb{R}$ with $h_u(x) \leq h(x)$ and $h(x) \leq h_o(x)$ for all $x \in D_h$



- partition the admissible interval $[x_i^\ell, x_i^u]$ of occuring variables x_i into $r \in \mathbb{N}$ uniform segments $\left[x_i^{\ell_j}, x_i^{u_j}\right], j \in [r]$
- introduce a binary variable for any such segment

Alert! Complexity incoming!

McCormick, G. P., (1976), *Computability of global solutions to factorable nonconvex programs. I. Convex underestimating problems.* Math. Programming, 10(2):147-175.

4 Key ideas for mixing it together

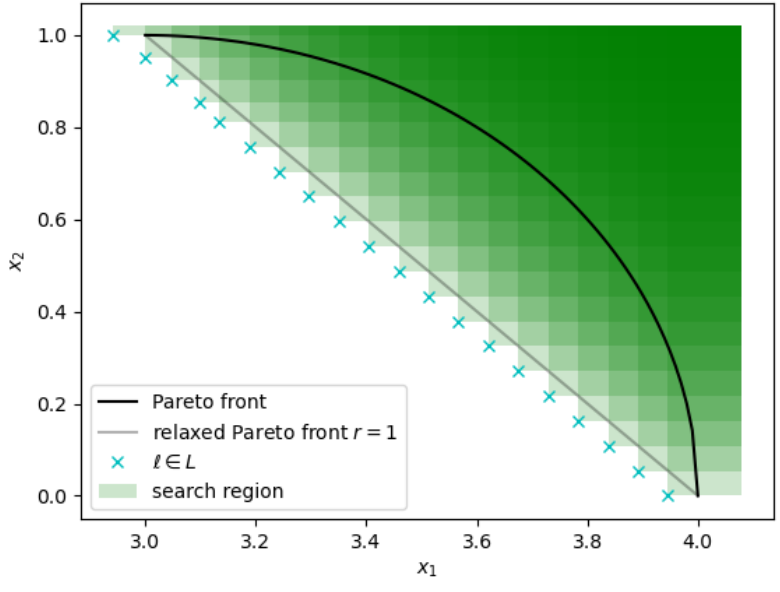
- use simplifed problems whenever possible
- refine relaxations only when necessary

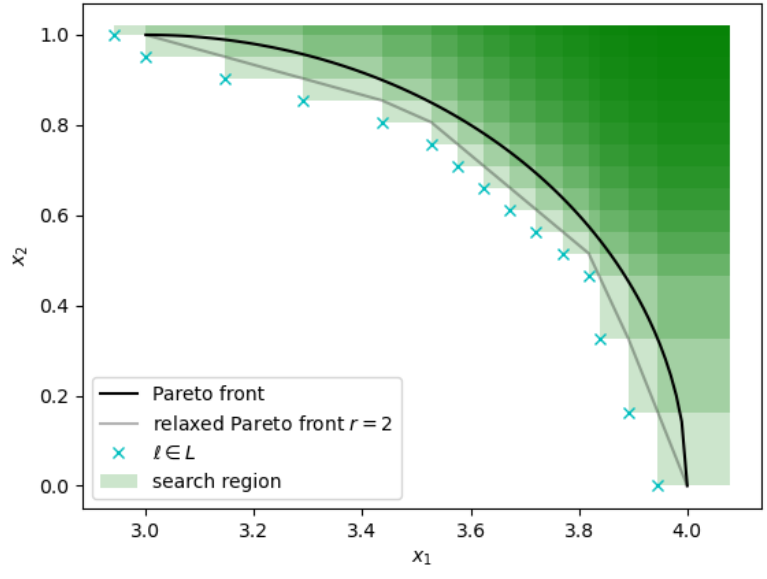
Key ideas – lower bounding

Idea 1: use refined relaxations to improve the lower bound set

Observation

Let $\mathcal{R}$ be a relaxation of (MOP) and $\tilde{N}$ an approximation of $\mathcal{N}_\mathcal{R}$. Then $L(\tilde{N})$ is a lower bound set of $\mathcal{N}$.





$\Rightarrow$ solve relaxed scalarized problem to obtain $\mathcal{N}_\mathcal{R} \ni f(\tilde{x}) \leq \hat{u} - \delta e$ and update $\tilde{N}$ and $L(\tilde{N})$

$\Rightarrow$ solve **only MILPs** for lower bounding

Key ideas – upper bounding

Idea 2: use information of relaxed solutions to compute feasible solutions of (MOP) for upper bounding part

- ▷ given $\tilde{x} = (\tilde{x}_C, \tilde{x}_I)$ efficient for some relaxation $\mathcal{R}$ of (MOP) with $f(\tilde{x}) \leq \hat{u} - \delta e$
- ▷ solve the reduced RWSM problem:

$$\begin{aligned} \min \alpha^\top f(x) \quad \text{s.t.} \quad & x \in S, \\ & f(x) \leq \hat{u} - \delta e, \\ & x_I = \tilde{x}_I. \end{aligned} \tag{redRWSM}$$

$\Rightarrow$ use solution to update N and $U(N)$

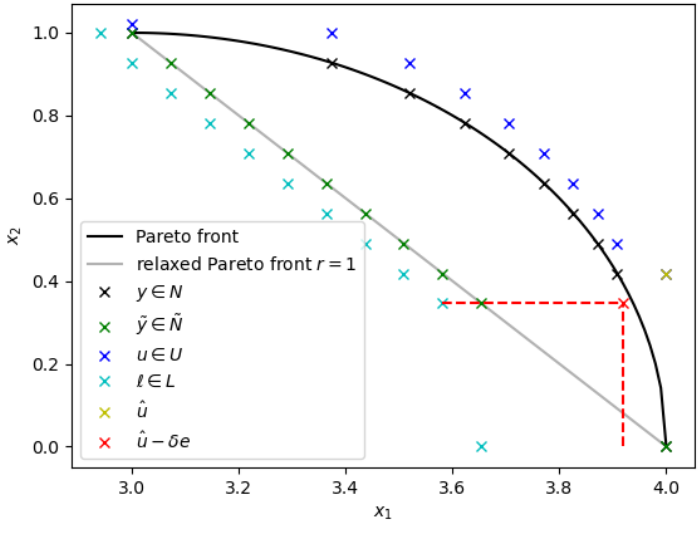
Positive effects:

- ▷ solve **only NLPs** for upper bounding
- ▷ no global optimality needed – **feasibility is enough**
- ▷ for any new local upper bound u we can relate a relaxation $\mathcal{R}^u$
  - $\Rightarrow$ choose relaxation $\mathcal{R}' \subseteq \mathcal{R}^{\hat{u}}$ for computations in search region determined by $\hat{u}$
  - $\Rightarrow$ allows **different relaxations in different parts** of the image space

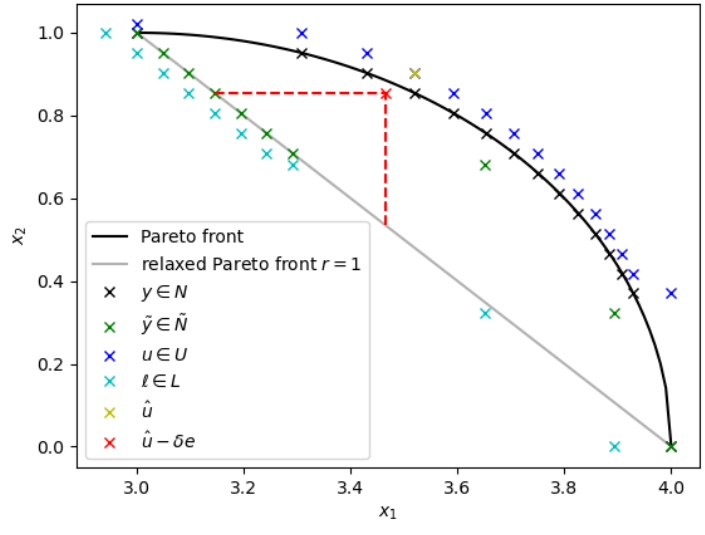
Key ideas – when to refine

Idea 3: refine relaxation **only if current relaxation does not help us anymore**, i.e. if it

- ▷ wrongly suggested that there exists $x \in S$ with $f(x) \leq \hat{u} - \delta e$



- ▷ did not improve $\tilde{N}$, i.e. if $f(\tilde{x})$ dominates $\tilde{N}$ or $f(\tilde{x}) \in \tilde{N}$ already



⇒ together with the relation of relaxations to local upper bounds this gives the possibility of a **dynamic refinement decision scheme**

Key ideas – when to stop refining

Similar as in [Burlacu et al., 2020]:

Idea 4: feasible points of **fine enough relaxations** are considered as **feasible** for (MOP)

- compute **relative relaxation error**:

$$\varepsilon_{\text{rel}}^{(h,\mathcal{R}_h)} = \frac{\max_{x \in D_h} \{h(x) - h_u(x), h_o(x) - h(x)\}}{\max_{x \in D_h} |h(x)|}$$

- given a relaxation $\mathcal{R}$ of (MOP), we denote the feasible set by $S_{\mathcal{R}}$
- for relaxations $\mathcal{R}^1$ and $\mathcal{R}^2$ of (MOP), we write $\mathcal{R}^1 \subsetneq \mathcal{R}^2$ if $S_{\mathcal{R}^1} \subseteq S_{\mathcal{R}^2}$ and $\varepsilon_{\text{rel}}^{\mathcal{R}^1} < \varepsilon_{\text{rel}}^{\mathcal{R}^2}$

Observation

Let $\mathcal{R}^1 \supsetneq \mathcal{R}^2 \supsetneq \dots$ be a chain of strictly finer relaxations. Then for any $\varepsilon_{\text{rel}} > 0$ there exists $s \in \mathbb{N}$ with $\varepsilon_{\text{rel}}^{\mathcal{R}^s} < \varepsilon_{\text{rel}}$.

Positive effect: guaranteed **improvement** of search region after **finitely many** refinement steps

Burlacu, R., Geißler, B., Schewe, L. (2020). *Solving mixed-integer nonlinear programmes using adaptively refined mixed-integer linear programmes*. Optimization Methods and Software, 35:1, 37-64.

Correct termination

Theorem (cf. [L. and Volkwein 2022, Theorem 5.8])

Let $\varepsilon_{\text{encl}} > \delta > 0, \varepsilon_{\text{rel}} > 0$ and $z^\ell, z^u \in \mathbb{R}^k$. We define

$$\Delta := \left\| z^u - z^\ell \right\|_\infty \quad \text{and} \quad \kappa := k \left\lceil \frac{\Delta - \varepsilon_{\text{encl}}}{\delta} \right\rceil + 1.$$

Then the number of outer `while`-loops is bounded by $\max\{1, \kappa\}$ and the procedure terminates after finitely many steps.

Proof:

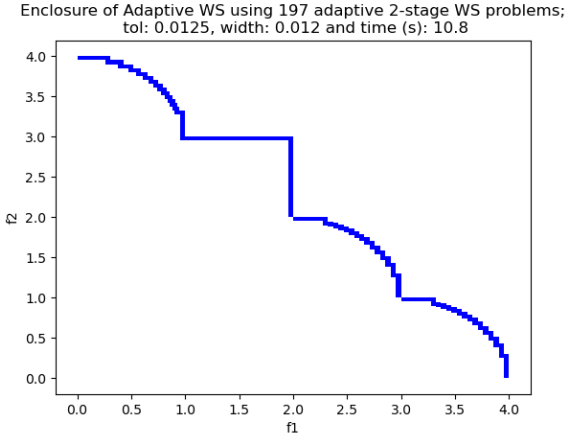
- only **finitely many relaxation refinement steps** are needed
- $U(N)$ is always finite
- **decrease guarantee**: after every iteration of the outer `while`-loop and for any new box $[\ell^n, u^n]$ there exists an old box $[\ell^o, u^o] \supseteq [\ell^n, u^n]$ and $j \in \{1, \ldots, k\}$ such that

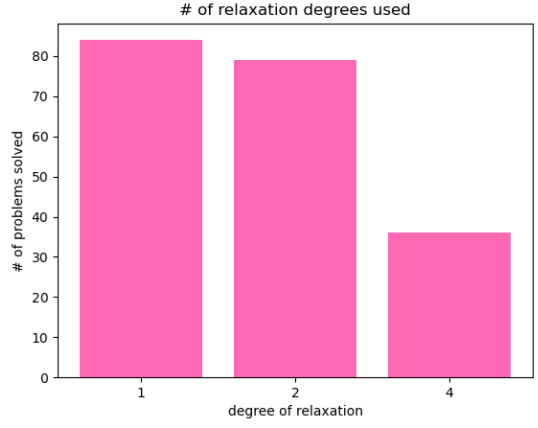
$$(u^n - \ell^n)_j < \max\left\{(u^o - \ell^o)_j - \delta, \varepsilon_{\text{encl}}\right\}$$

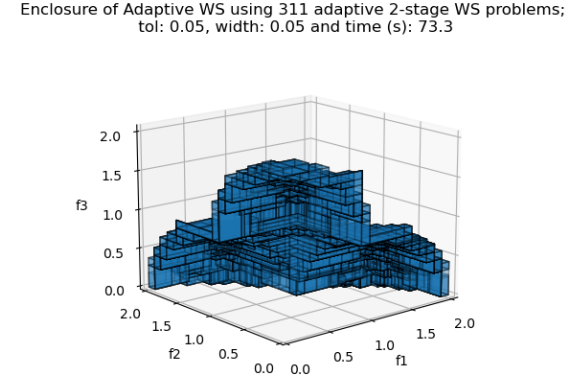
L., M. and Volkwein, S. (2022), *Computing an enclosure for multiobjective mixed-integer nonconvex optimization problems using piecewise linear relaxations.*

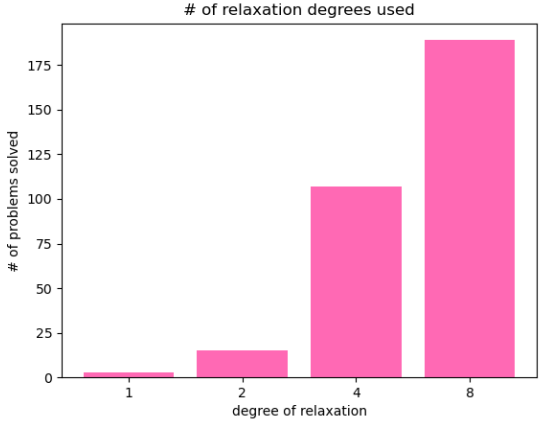
5 Numerical results

Numerical experiments and drawbacks









- ▷ at least 5 partitions for $\varepsilon_{\text{rel}} = 0.01$

comparison to direct method:

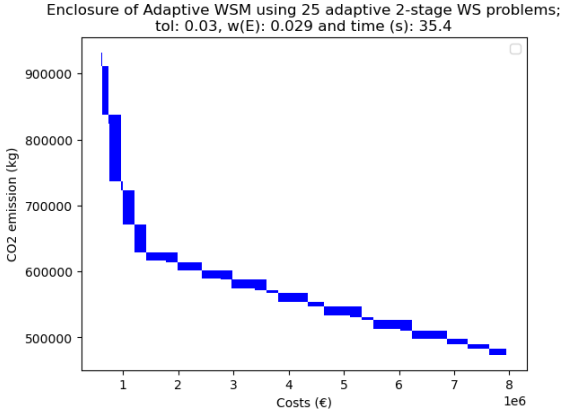
- ▷ **2 dim case:**
  - # 49 problems
  - time: 1.4s
- ▷ **3 dim case:**
  - # 210 problems
  - time: 27s

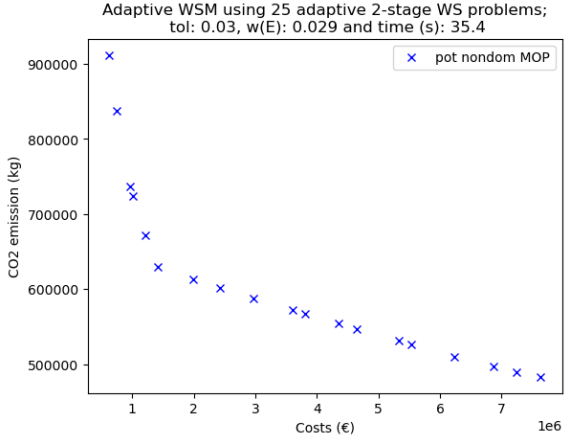
$\Rightarrow$ more problems need to be solved

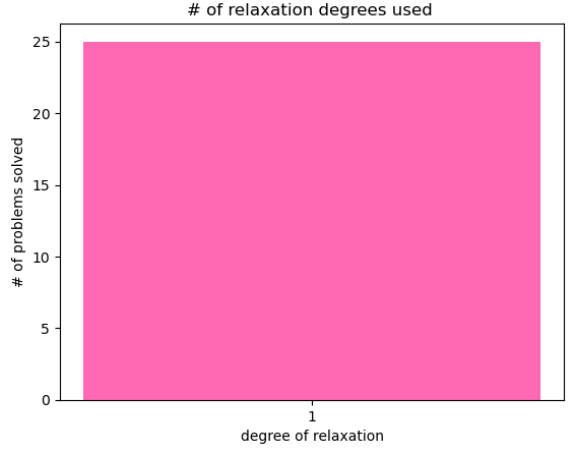
$\Rightarrow$ increase in computational time

Numerical experiments on the test network

On the same network instance as in the beginning we obtain:







- ▷ to reach a relative relaxation error tolerance of $\varepsilon_{\text{rel}} = 0.03$ (or 0.01) one has to use at least $r = 6$ ($r = 10$, resp.) partitions

⇒ it is **not really an option** to choose a fine enough relaxation and then use algorithms for multi-objective mixed-integer linear problems

⇒ **dynamic refinement decision procedure** works very well for that problem

Summary

- compute enclosures for multi-objective mixed-integer polynomially constrained problems
- avoid solving MINLPs
- use dynamically refined piecewise linear relaxations
- criterion space guides decision for refinement

If you are interested

L., M. and Volkwein, S. (2022), *Computing an enclosure for multiobjective mixed-integer nonconvex optimization problems using piecewise linear relaxations.*

Thanks for your attention!