technische universität
dortmund

Computational Intelligence

Winter Term 2025/26

Prof. Dr. Günter Rudolph

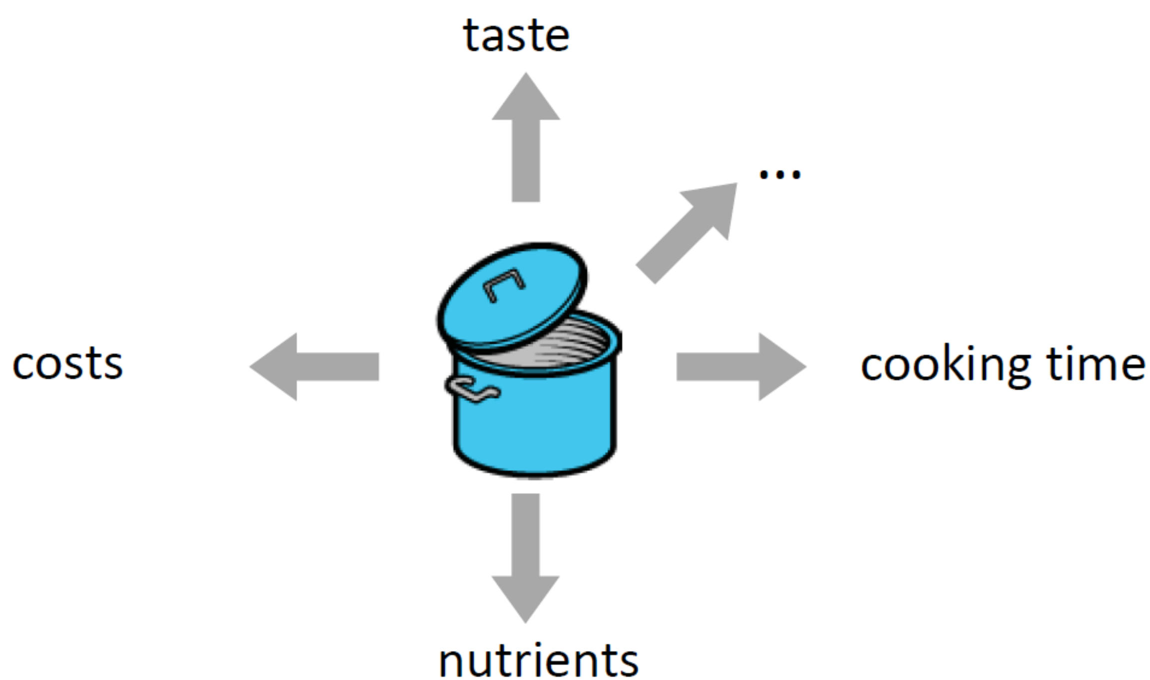
Computational Intelligence

Fakultät für Informatik

TU Dortmund

- Multiobjective Evolutionary Algorithms
  - Examples of Multiobjective Problems
  - Theoretical Basics
  - Contemporary MOEAs

Example from daily life: Cooking



Example: buying a used car

	VW	Opel	Ford	Toyota	
price [k€]	16	14	15	13	→ min!
fuel consumption [l/100km]	7,2	7,0	7,5	7,8	→ min!
power [kW]	65	55	58	55	→ max!

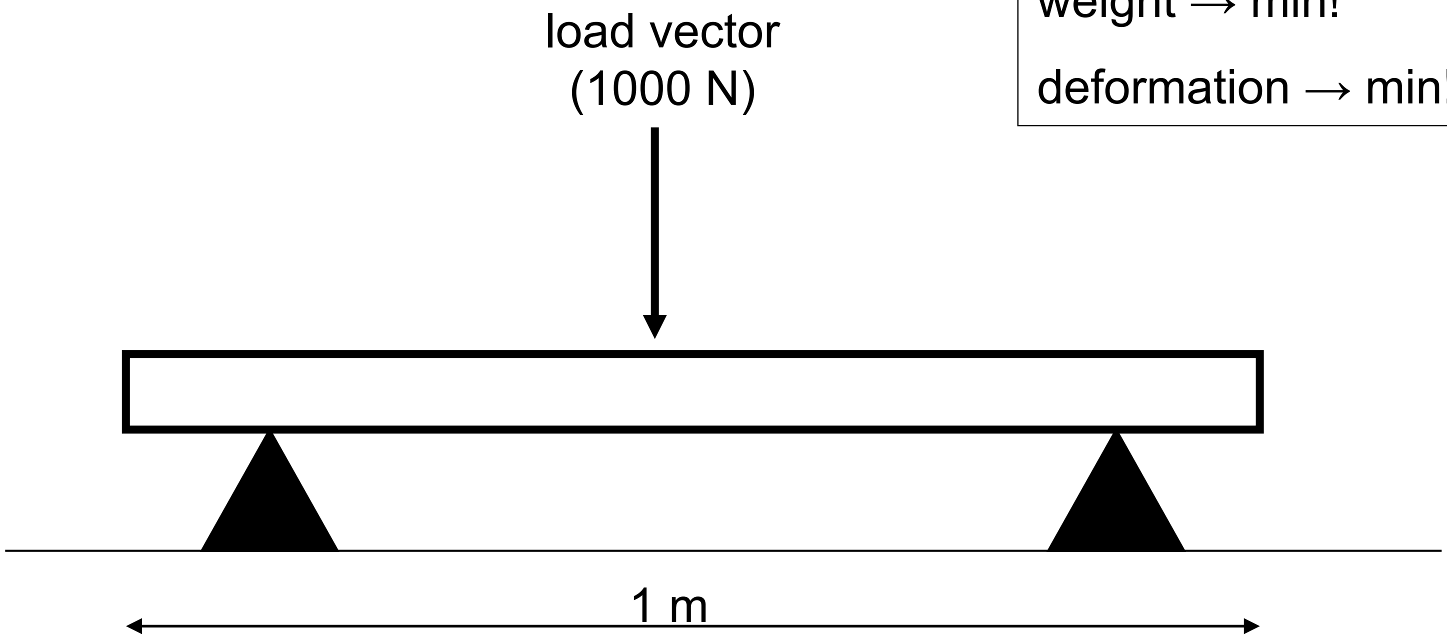
3 objectives, 4 alternatives → best alternative?

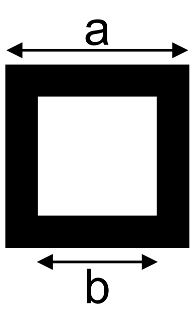
M. Ehrgott: Multicriteria Optimization, 2nd ed., Springer: Berlin 2005. (S. 1f.)

Example: Design of a Hollow Beam

weight → min!

deformation → min!





$a^2 - b^2 \rightarrow \text{min!}$

$1000 + [32 \times 10^8 \times (a^4 - b^4)]^{-1} \rightarrow \text{min!}$

$0 \leq b \leq b + 0{,}04 \leq a$ und $a \leq 0{,}1$

Multiobjective Optimization:

optimization under multiple objectives, where objectives are

- <u>in conflict</u> and
- <u>incommensurable</u> (= incomparable w.r.t. unit)

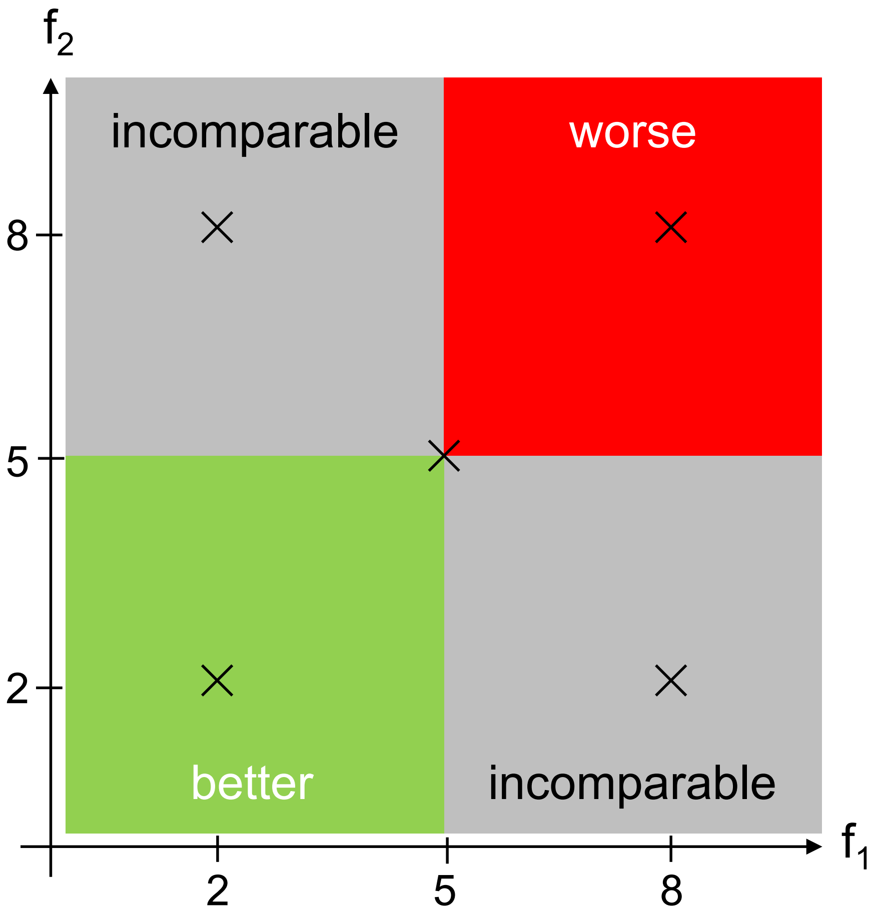
Example:

costs	[€]
weight	[kg]
pressure resistence	[hPa]
length	[m]
…	

→ concept of optimality?

→ concept of solution?

→ algorithmic approach?



- weak partial order

  $a \preceq b \quad \Leftrightarrow \quad \forall i \in [1..d] : a_i \leq b_i$

- partial order

  $a \prec b \quad \Leftrightarrow \quad a \preceq b$ and $a \neq b$

- a, b comparable $\Leftrightarrow$ $a \preceq b$ or $b \preceq a$

- a, b incomparable $\Leftrightarrow$ $a \parallel b$ $\Leftrightarrow$ neither $a \preceq b$ nor $b \preceq a$

$$\begin{pmatrix} 2 \\ 2 \end{pmatrix} \prec \begin{pmatrix} 5 \\ 5 \end{pmatrix} \prec \begin{pmatrix} 8 \\ 8 \end{pmatrix} \quad \text{but} \quad \begin{pmatrix} 2 \\ 8 \end{pmatrix} \parallel \begin{pmatrix} 5 \\ 5 \end{pmatrix} \parallel \begin{pmatrix} 8 \\ 2 \end{pmatrix}$$

antichain =
set of mutually
incomparable elements

Definition 1:

Let $S \subseteq \mathbb{R}^n$ and $f: S \to \mathbb{R}^d$, $d \geq 2$.

***multiobjective optimization problem* =**

$(f_1(x), f_2(x), \ldots, f_d(x))' \to \min!$

s.t. $x \in S$ ■

Definition 2:

If $f(x) \prec f(y)$, then: x ***dominates*** y, f(x) ***dominates*** f(y).

solution $x^* \in S$ is termed ***Pareto-optimal*** $\Leftrightarrow$ exists no $x \in S$ with $f(x) \prec f(x^*)$.

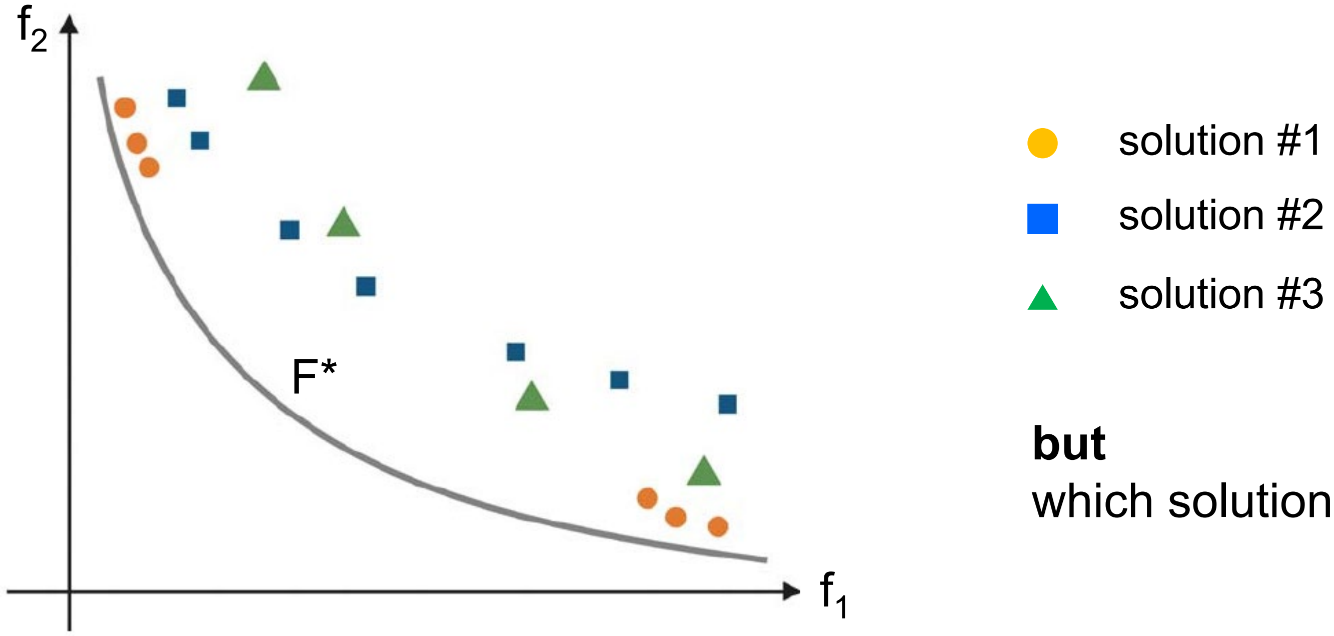
If x* Pareto-optimal, then f(x*) ***efficient***.

set of all Pareto-optimal elements: S* = ***Pareto set***

set of all efficient elements F* = ***efficient set*** or ***Pareto front***. ■

Remark: If X $\subseteq \mathbb{R}^n$ then size of F* may be innumerable
and locating exact solution on F* intractable
$\Rightarrow$ hopeless to find X* or F* completely elementwise

Remedy: Find finite approximation of F*



but
which solution is better?

Isn't there an easier way? → **Scalarization**

⇒ merge vector-valued fitness function into a scalar-valued fitness function

frequently seen: weighted sum $f^S(x; w) = \sum_{i=1}^{d} w_i \, f_i(x) \rightarrow \min!$

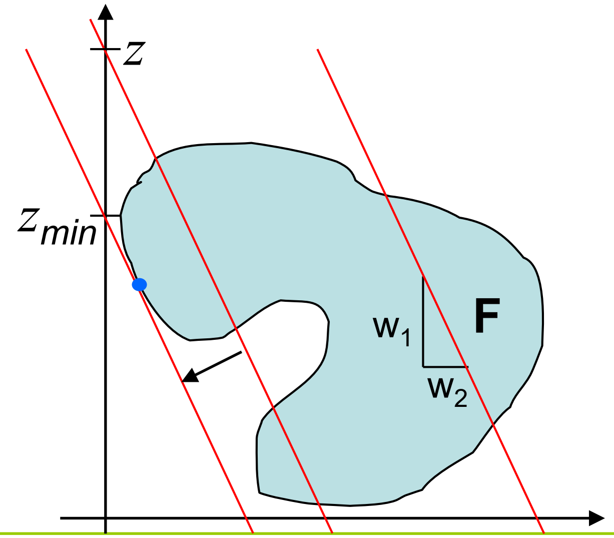
what happens?

$z = w_1 \, f_1(x) + w_2 \, f_2(x) = w_1 \, y_1 + w_2 \, y_2 \rightarrow$ min!

solve for y_2:

$$y_2 = -\frac{w_1}{w_2} y_1 + z$$

is minimized while optimizing the scalar problem

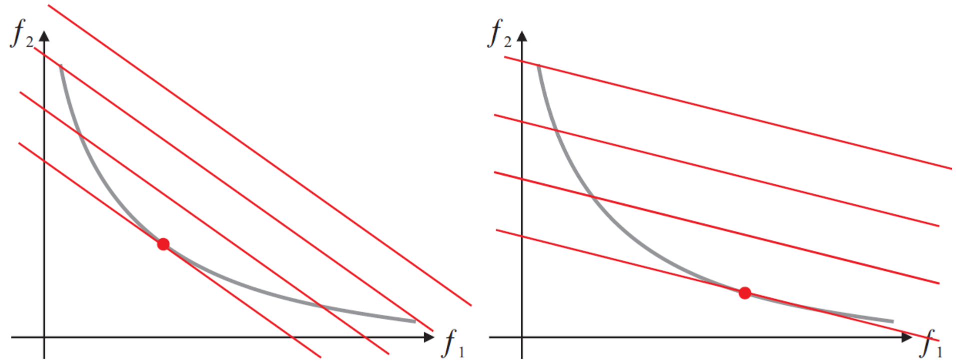


→ find straight line with minimal z, such that F is just touched

→ tangent point with F

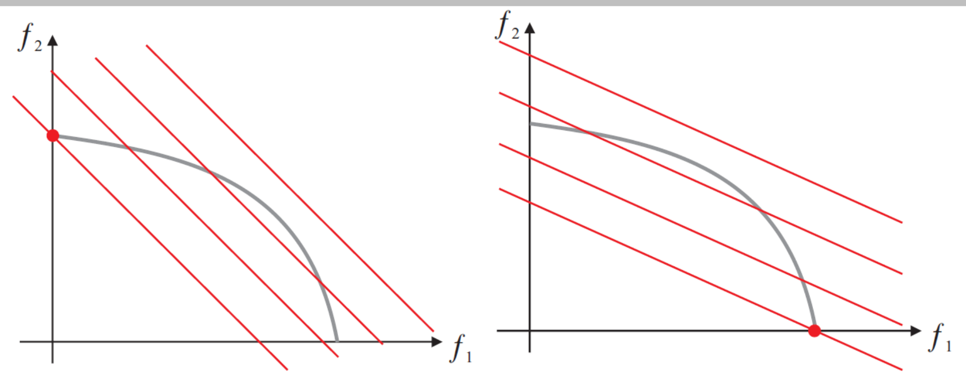
good news

every optimal solution found for the scalar problem =
optimal solution for the multiobjective problem



bad news

not all optimal solution of the multiobjective problem can be found this way!



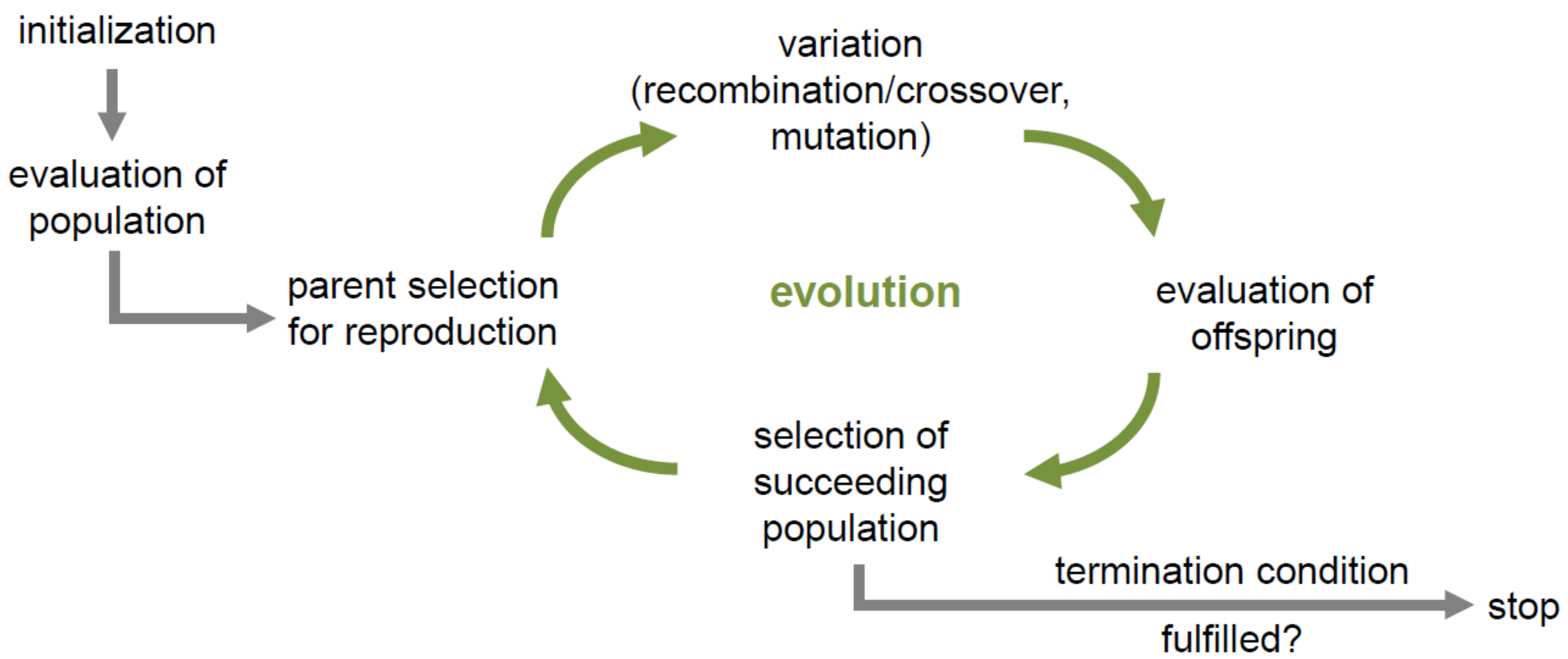
classification of methods

- ***a priori* approach**
  first specify preferences, then optimize
  more advanced scalarization techniques (e.g. Tschebyscheff) can find entire PF
  remaining difficulty:
  how to express your desires through parameter values!?

- ***a posteriori* approach**
  first optimize (approximate Pareto front), then choose solution

→ back to a-posteriori approach
→ state-of-the-art methods: evolutionary algorithms

Which changes are necessary to make EA working in multiobjective case?



⇒ selection operation must be able to cope with **partial order of fitness values**

⇒ no need to alter variation operators

Selection in EMOA / MOEA

Selection requires kind of sortable population to choose “best” individuals

But: How to sort d-dimensional objective vectors?

Possible two-stage approach:

Primary selection criterion:
use Pareto dominance relation to sort *comparable* individuals

Secondary selection criterion:
apply additional measure to *incomparable* individuals to enforce total order

auxiliary device: building a **hierarchy of antichains** (aka **nondominated sorting**)

→ foundation of many selection operators!

the dual result to theorem of Dilworth (1950):

Theorem: (Mirsky 1971)

Let $(F, \preccurlyeq)$ a partially ordered set of height h. Then there exists a partition $(F_1, F_2, \ldots, F_h)$ of F consisting of antichains $F_1, \ldots, F_h$ with the property

$$\forall\, y \in F_{i+1} : \exists\, x \in F_i : x \prec y \text{ for } i = 1, \ldots, h-1.$$

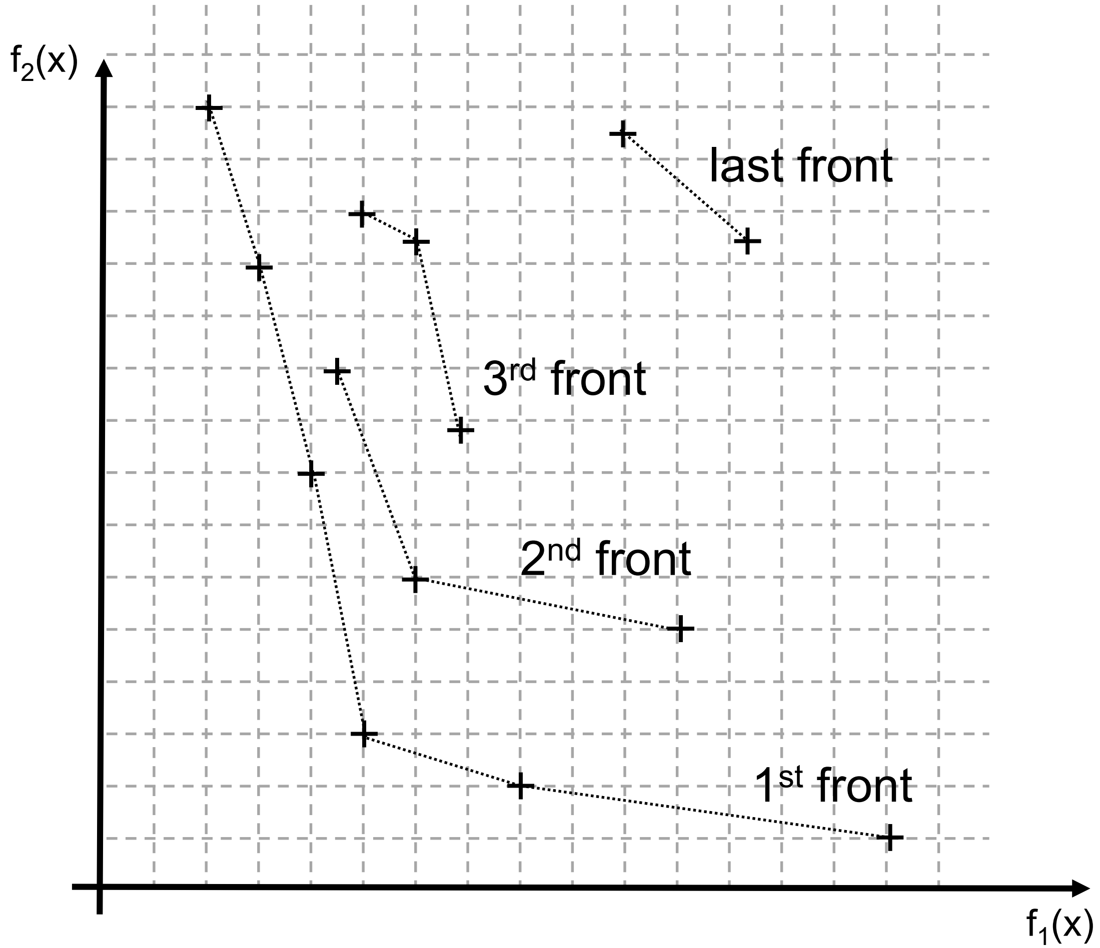
algorithmically:

let $F_1 = ND(F, \preccurlyeq)$, i.e., the set of nondominated elements of F.

set $F_i = ND(F \setminus (F_1 \cup F_2 \cup \ldots \cup F_{i-1}), \preccurlyeq)$ for $i = 2, \ldots, h$. ■

- L. Mirsky : A Dual of Dilworth's Decomposition Theorem. *The American Mathematical Monthly 78(8):876-877*, 1971.
- R. P. Dilworth: A Decomposition Theorem for Partially Ordered Sets. Annals of Mathematics 51(1):161-166, 1950.

example: nondominated sorting



NSGA-II

popular MOEA: **n**ondominated **s**orting **g**enetic **a**lgorithm (version **II**)

create μ parents $\in P(0)$ and μ offspring $\in Q(0)$; $t = 0$
repeat
  build hierarchy of antichains $(A_1, \ldots, A_h)$ from $A(t) = P(t) \cup Q(t)$
  $P(t+1) = \emptyset$; $i = 1$
  while card$(P(t+1) \cup A_i) \leq \mu$ **do**
    $P(t+1) = P(t+1) \cup A_i$; i++
  od
  if necessary do 'crowding-sort' on A_i; fill $P(t+1)$ from sorted A_i
  generate offspring $Q(t+1)$ from $P(t+1)$
until stopping criterion applies

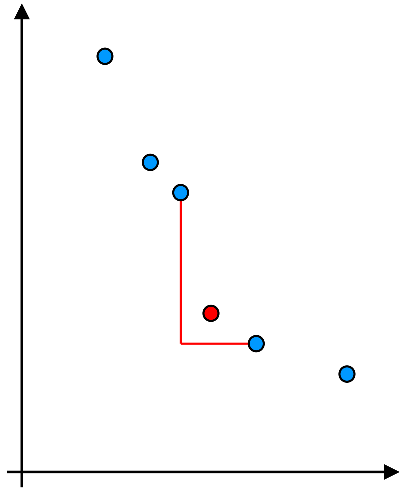
NSGA-II: crowding sort (secondary selection criterion)

crowding distance:

- half perimeter of empty bounding box around point
- value of infinity for boundary points
- large values good

crowding sort:

- sort w.r.t. crowding distance
- select those with largest value



NSGA-II: crowding selection (used for parent selection)

selection of parents used for recombinaton:

for each new parent: perform ‘crowded tournament selection‘

crowded tournament selection:

draw x and y uniformly at random from population

1. if rank(x) < rank(y) then select x
2. if rank(x) > rank(y) then select y
3. if $d_c(x) > d_c(y)$ then select x else y

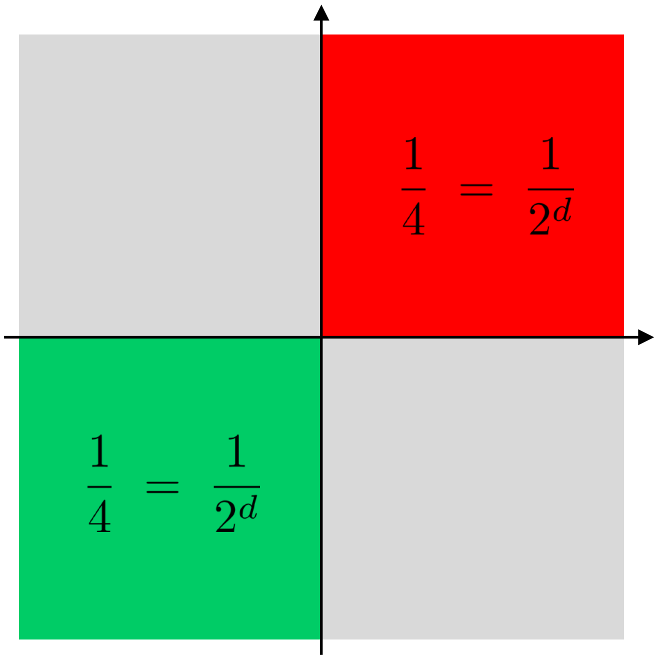
$d_c(x)$ = crowding distance

difficulties of selection

if d = #objectives large,
then most objective vectors incomparable:

share: $1 - 2 \cdot \frac{1}{2^d}$ (if uniformly distributed)

$\Rightarrow$ almost all solutions in 1st front!
$\Rightarrow$ selection in 1st stage with no effect
$\Rightarrow$ selection in 2nd stage must drive to Pareto front!



typical case: all individuals incomparable
$\Rightarrow$ mainly secondary selection criterion in operation

drawback of crowding distance:
rewards spreading of points, does not reward approaching the Pareto front
$\Rightarrow$ NSGA-II diverges for large *d*, difficulties already for *d* = 3

difficulties of selection

observation:

secondary selection criterion has to be meaningful!

desired: choose best subset of size μ from individuals

how to compare sets of partially incomparable points?

$\Rightarrow$ use quality indicators for sets

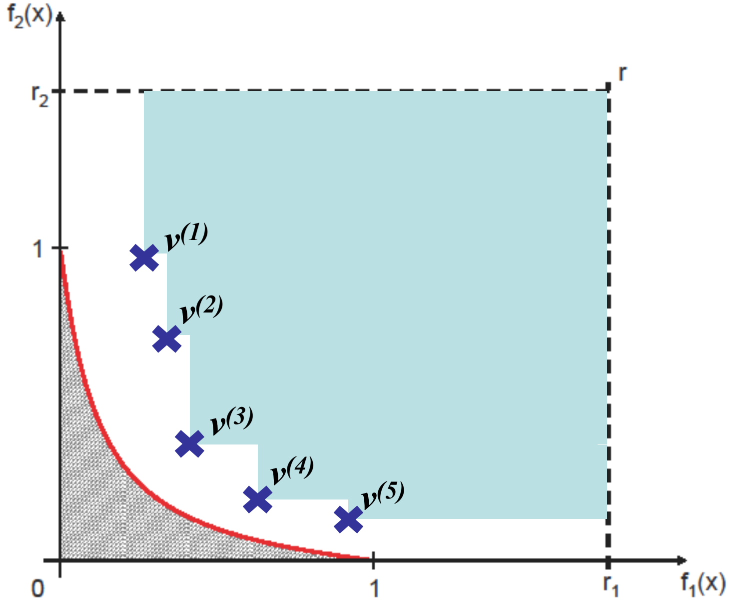
possible approach for selection:

$\Rightarrow$ for each point: determine contribution to quality value of set

$\Rightarrow$ sort points according to contribution

quality indicator: dominated hypervolume (aka S-metric)

- given antichain $v^{(1)}, v^{(2)}, \ldots v^{(\mu)} \in \mathbb{R}^2$ in lexicographic order
- given reference point $r \in \mathbb{R}^2$: $v^{(i)} \prec r$ for all $i = 1, \ldots, \mu$.



general case:

$$H(v^{(1)}, \ldots, v^{(\mu)}; r) = \text{vol}\left(\bigcup_{i=1}^{\mu} [\, v^{(i)}, r \,]\right)$$

dominated hypervolume w.r.t. r

$$H(v^{(1)}, \ldots, v^{(\mu)}; r) = \left[r_1 - v_1^{(1)}\right] \cdot \left[r_2 - v_2^{(1)}\right] + \sum_{i=2}^{\mu} \left[r_1 - v_1^{(i)}\right] \cdot \left[v_2^{(i-1)} - v_2^{(i)}\right]$$

SMS-EMOA (S-metric selection EMOA)

initialize population of μ individuals
`repeat`
  draw two individuals uniformly at random
  recombine them and mutate resulting offspring
  determine antichain hierarchy $A_1, \ldots, A_h$
  replace individual from A_h with least S-metric contribution
`until` stopping criterion applies

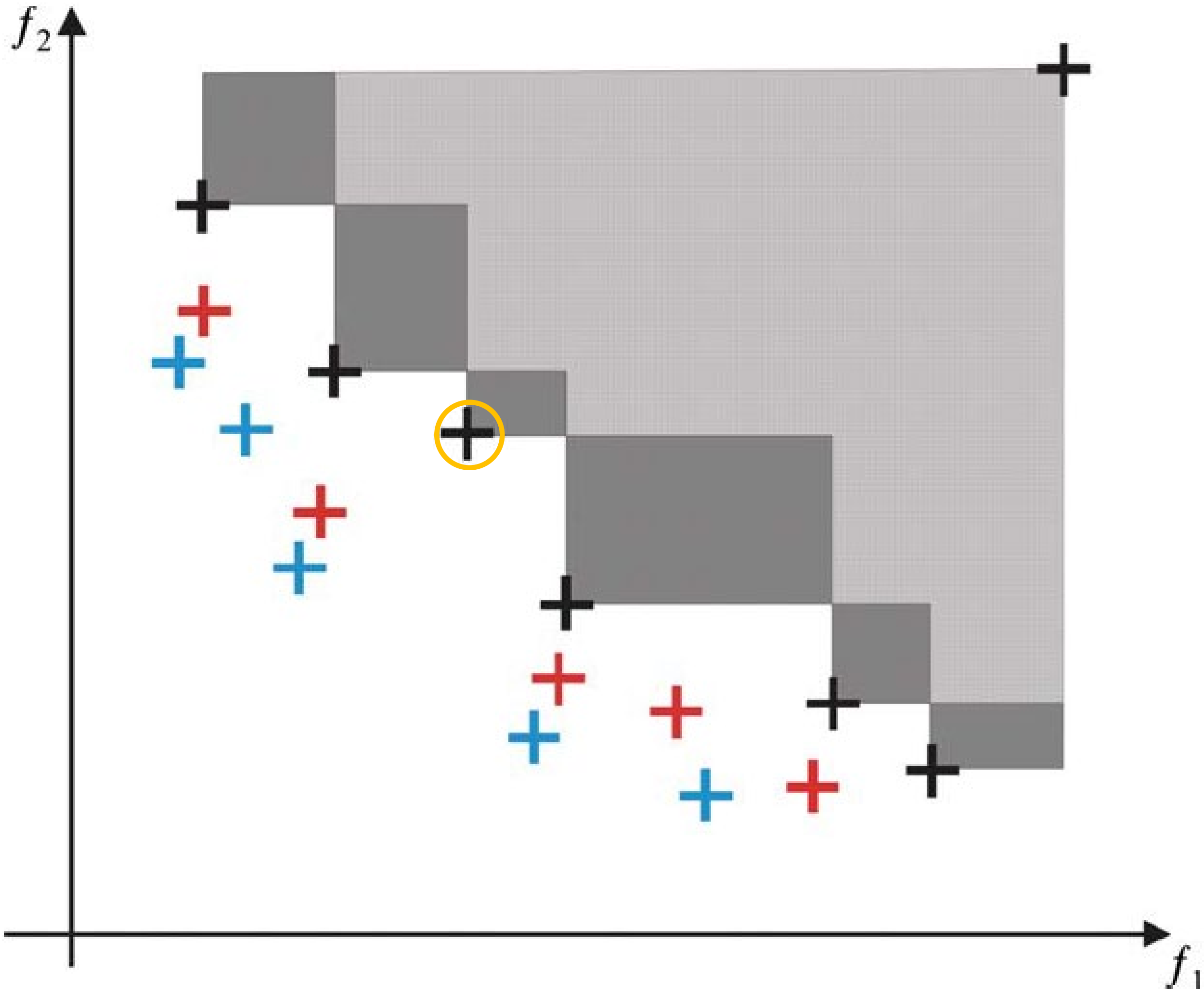
computational complexity:

S-metrik must be computed μ times for μ individuals $\Rightarrow$ naive: $O(\mu^{d+1})$

$\Rightarrow O(\mu^{d/2+1} \log \mu)$ via Overmars/Yap (Beume, März 2006)

$\Rightarrow O(\mu^{d/3} \log^k \mu)$ via Chen (2013)

example: S-metric selection in d=2



summary

- real-world problems are often multiobjective
- Pareto dominance only a partial order
- *a priori*: parameterization difficult
- *a posteriori*: choose solution after knowing possible compromises
- state-of-the-art a posteriori methods: EMOA, MOEA
- EMOA require sortable population for selection
- use quality measures as secondary selection criterion
- hypervolume: excellent quality measure, but computationally intensive
- use state-of-the-art EMOA, other may fail completely