Evolutionary Methods in Multi-Objective Optimization
- Why do they work ? -

Lothar Thiele

Computer Engineering and Networks Laboratory
Dept. of Information Technology and Electrical Engineering
Swiss Federal Institute of Technology (ETH) Zurich

Overview

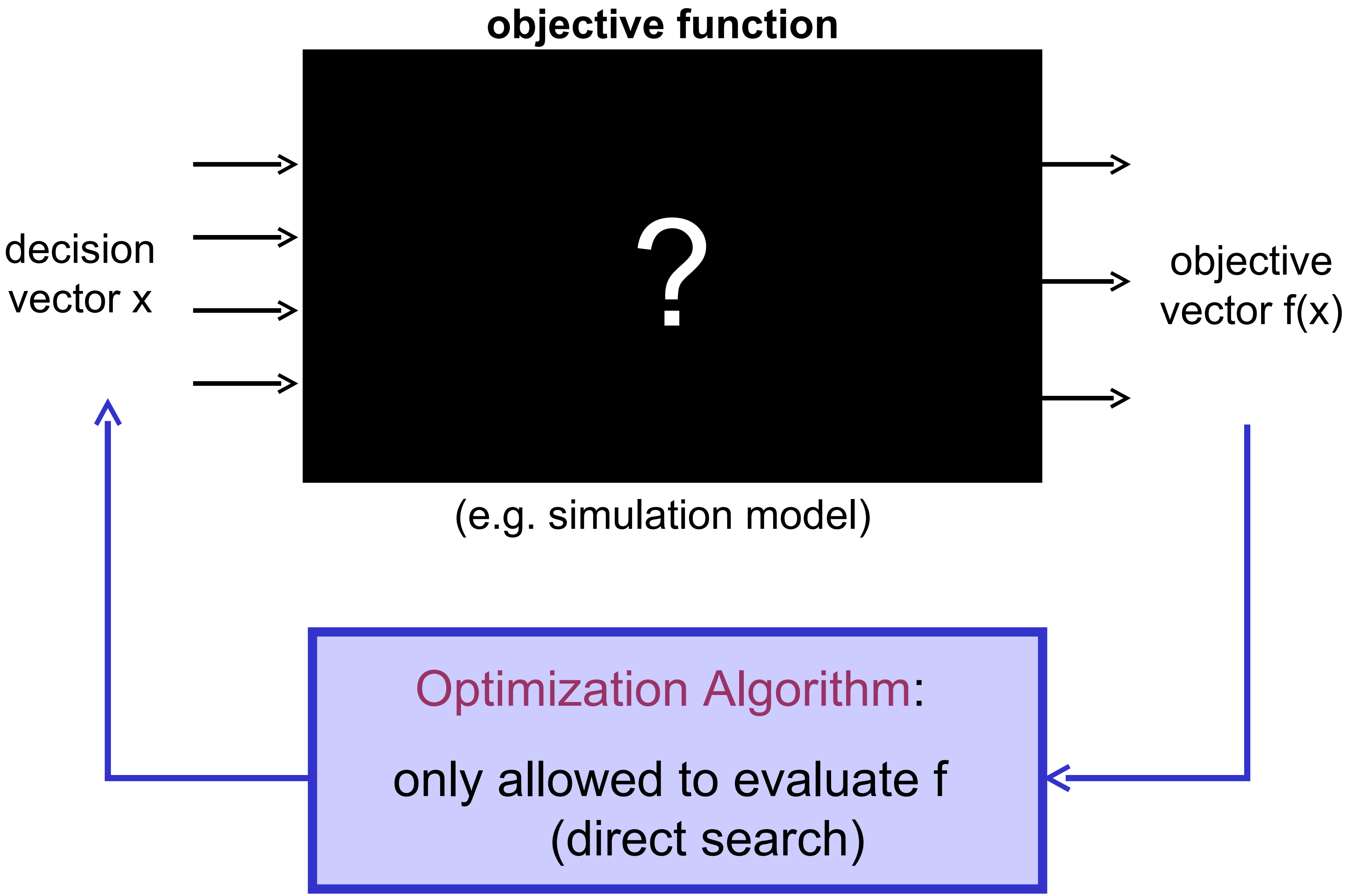
❶ **introduction**

❷ limit behavior

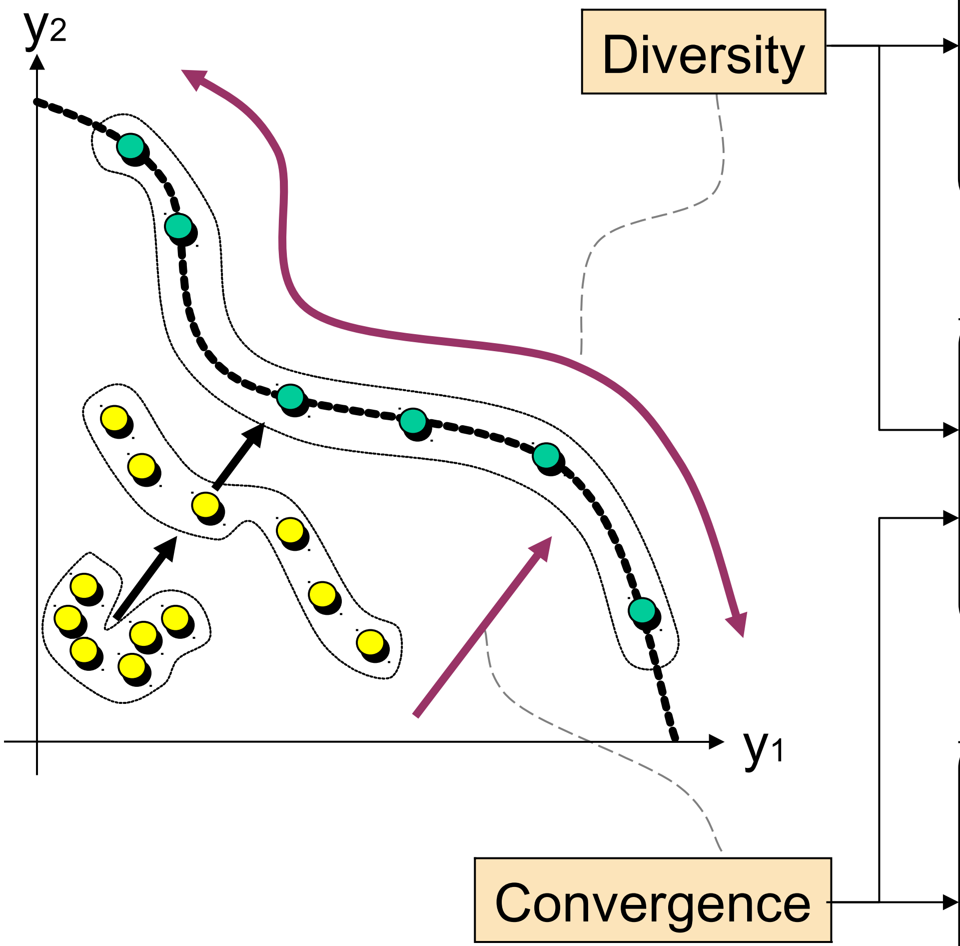
❸ run-time

❹ performance measures

Black-Box Optimization



Issues in EMO



- How to maintain a diverse Pareto set approximation?

  ❷ **density estimation**

- How to prevent nondominated solutions from being lost?

  ❸ **environmental selection**

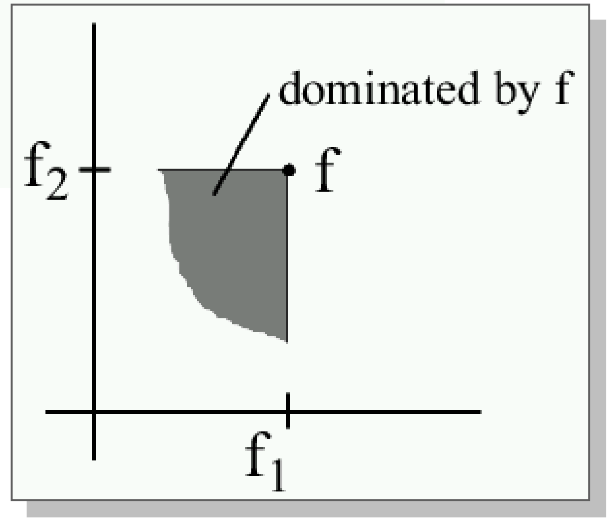
- How to guide the population towards the Pareto set?

  ❶ **fitness assignment**

Multi-objective Optimization

Definition 1 (Dominance relation)
Let $f, g \in \mathbb{R}^m$. Then f is said to dominate g, denoted as $f \succ g$, iff

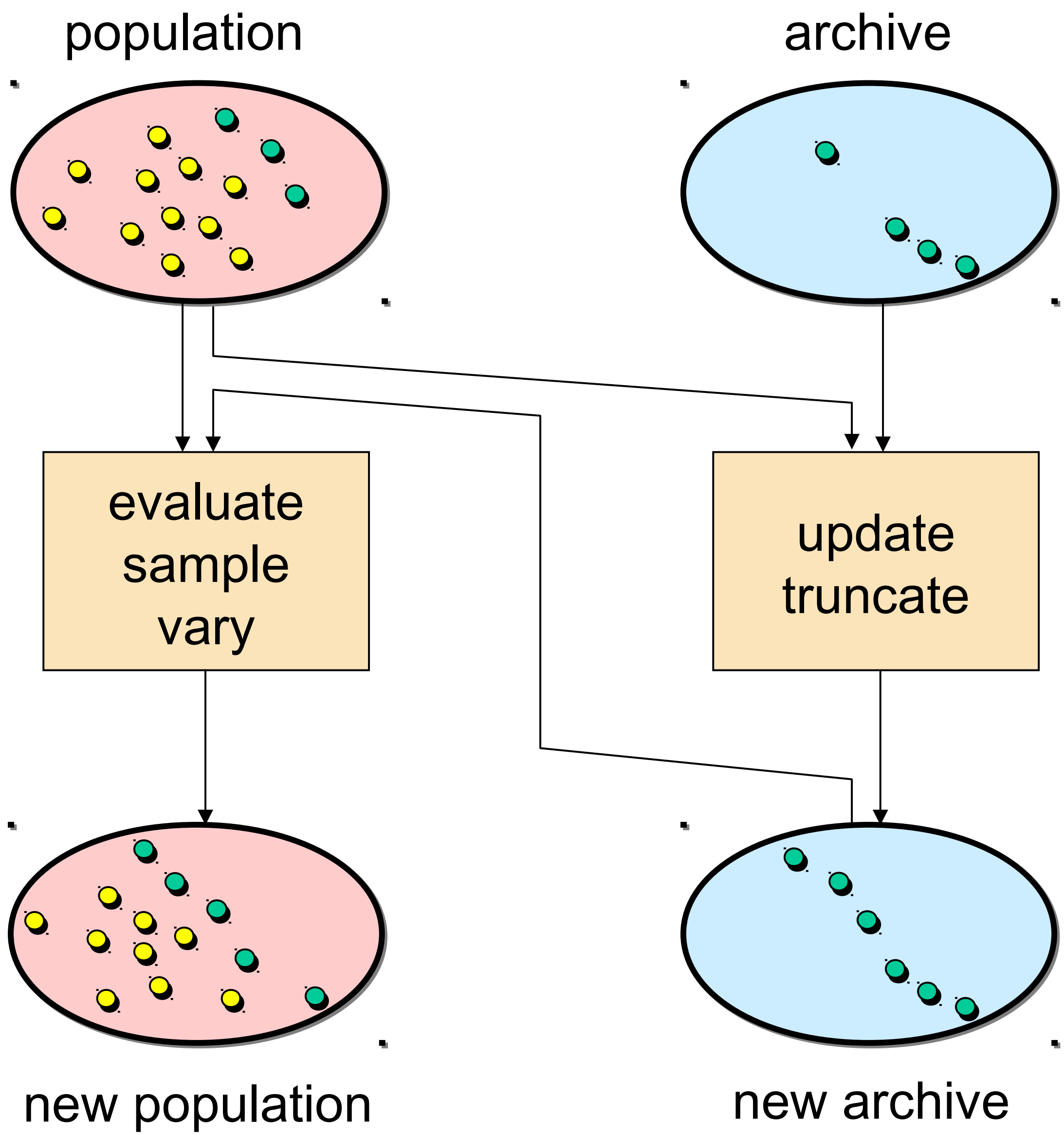
1. $\forall i \in \{1, \ldots, m\} : f_i \geq g_i$
2. $\exists j \in \{1, \ldots, m\} : f_j > g_j$



Definition 2 (Pareto set)
Let $F \subseteq \mathbb{R}^m$ be a set of vectors. Then the Pareto set $F^* \subseteq F$ is defined as follows: F^* contains all vectors $g \in F$ which are not dominated by any vector $f \in F$, i.e.

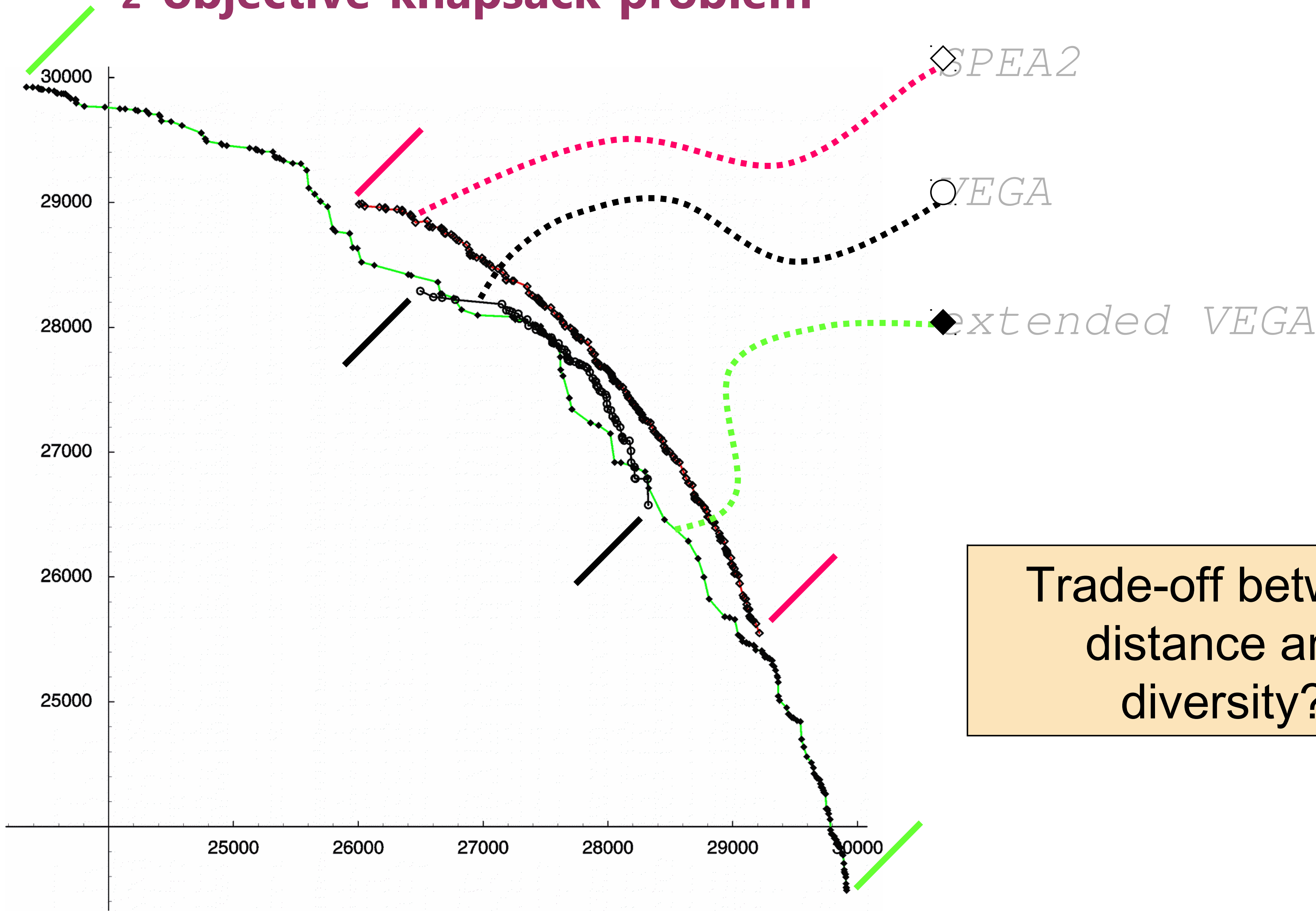
$$F^* := \{g \in F \mid \nexists f \in F : f \succ g\} \tag{1}$$

A Generic Multiobjective EA



Comparison of Three Implementations

2-objective knapsack problem



Trade-off between distance and diversity?

Performance Assessment: Approaches

Which technique is suited for which problem class?

❶ **Theoretically (by analysis):** difficult

- Limit behavior (unlimited run-time resources)
- Running time analysis

❷ **Empirically (by simulation):** standard

Problems: randomness, multiple objectives

Issues: quality measures, statistical testing, benchmark problems, visualization, …

Overview

❶ introduction

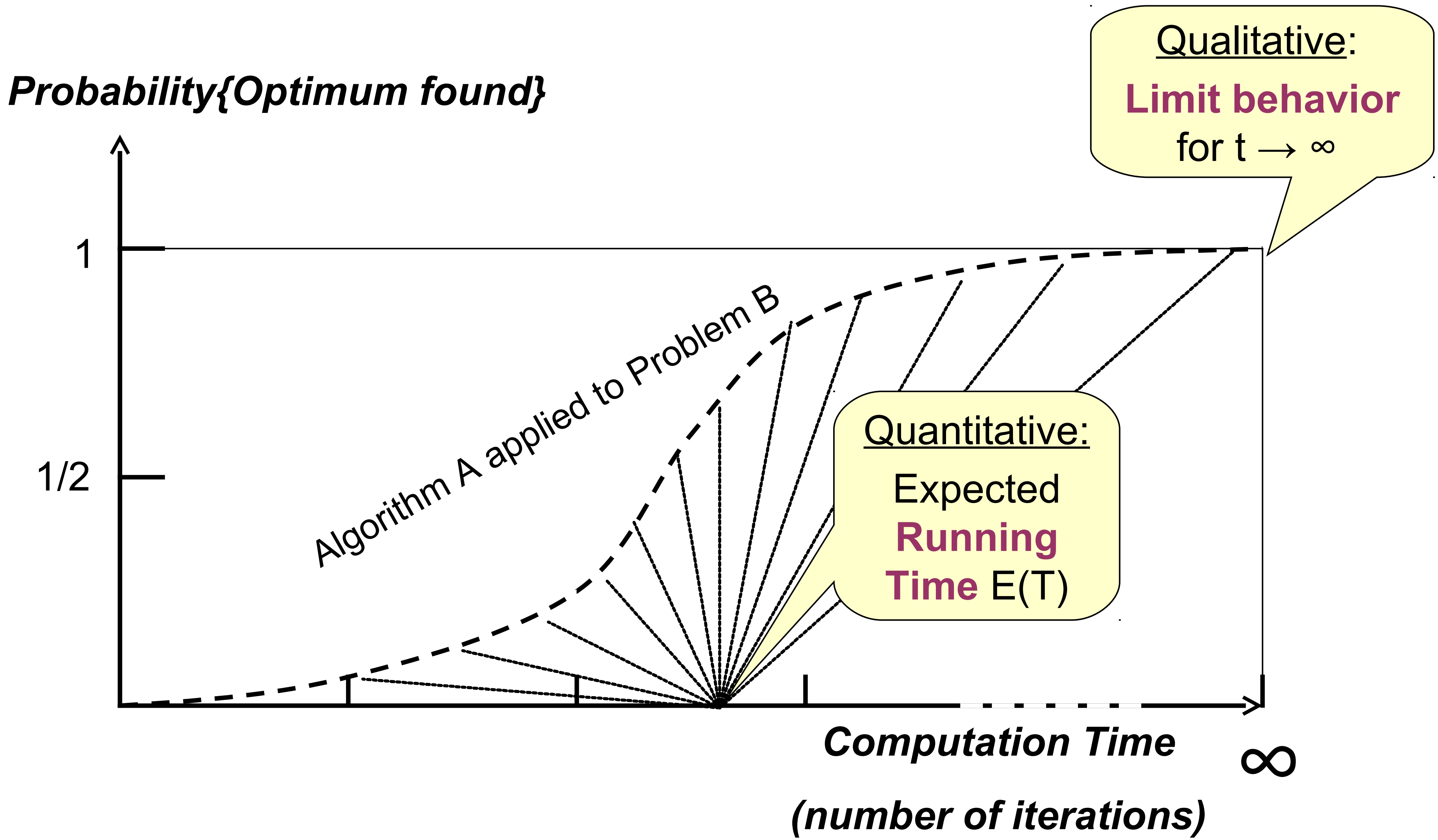
❷ limit behavior

❸ run-time

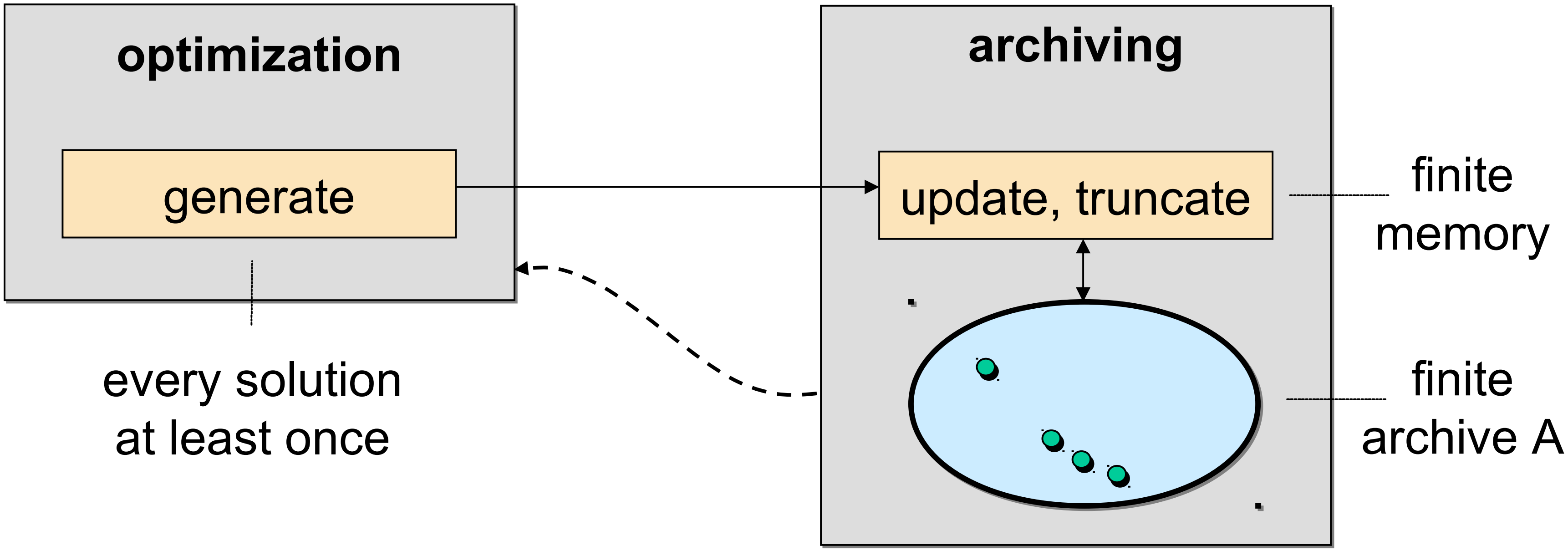
❹ performance measures

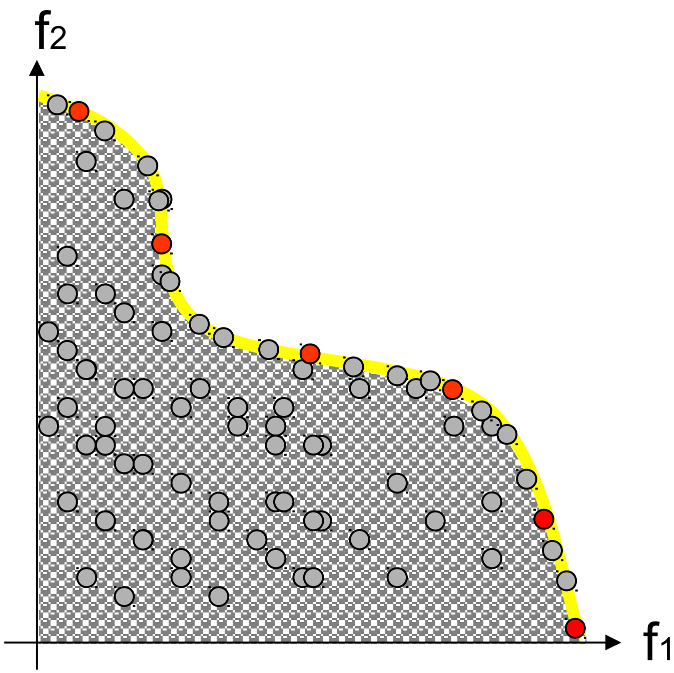
Analysis: Main Aspects

- Evolutionary algorithms are `random` search heuristics



Archiving

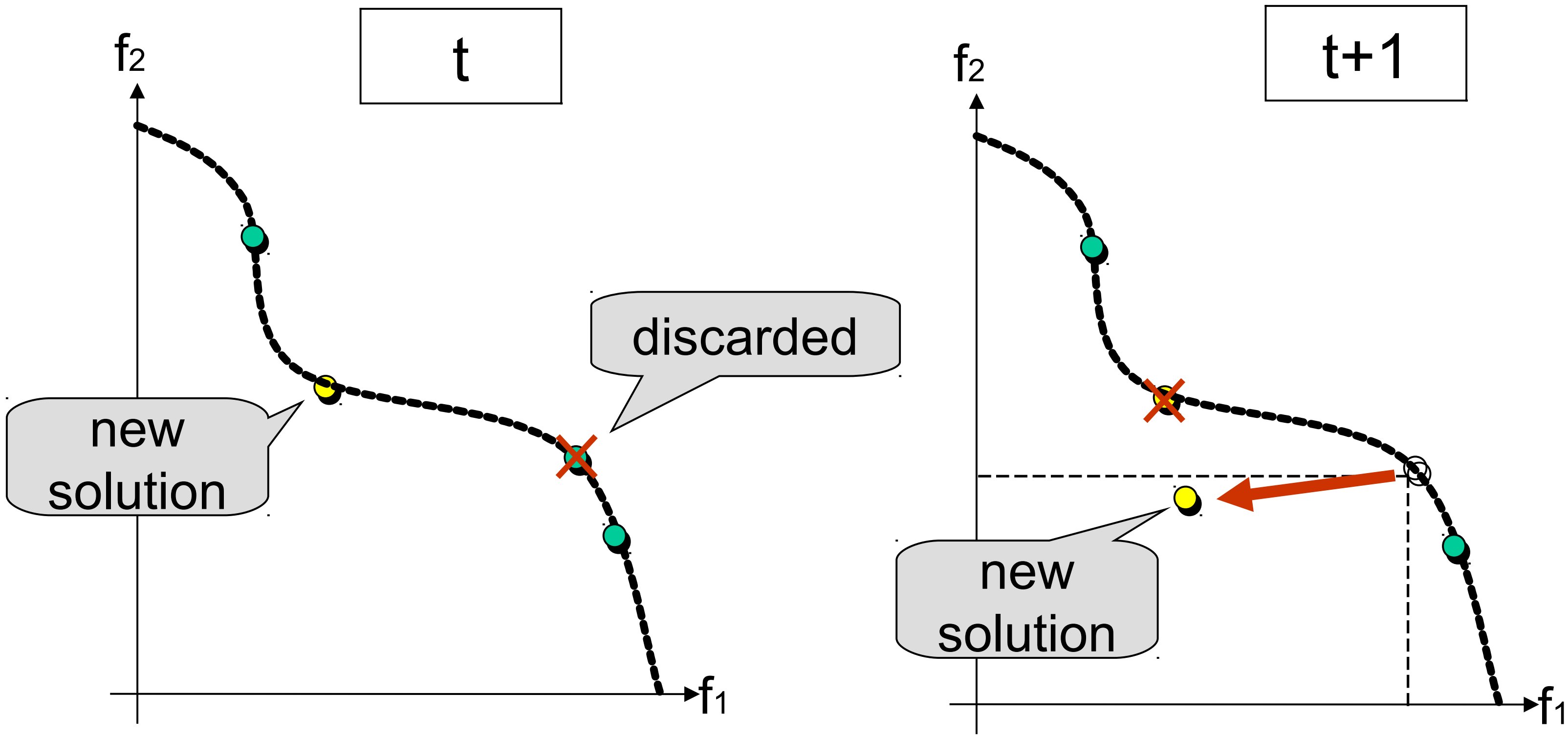




Requirements:

1. Convergence to Pareto front
2. Bounded size of archive
3. Diversity among the stored solutions

Problem: Deterioration

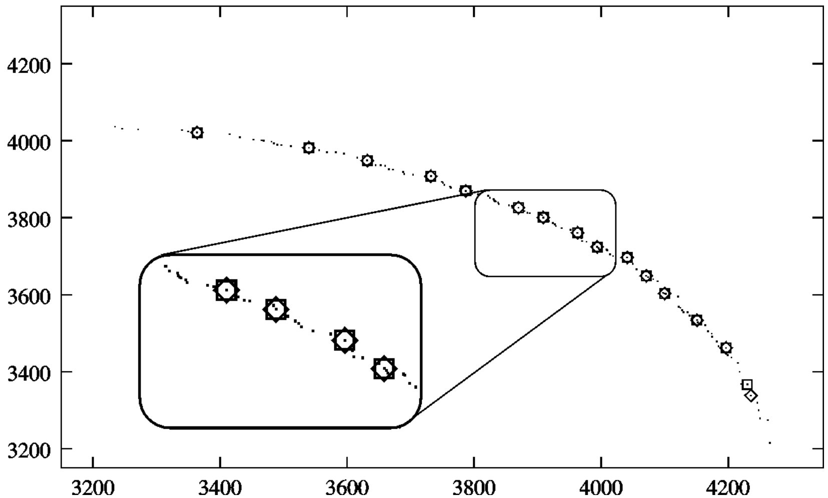


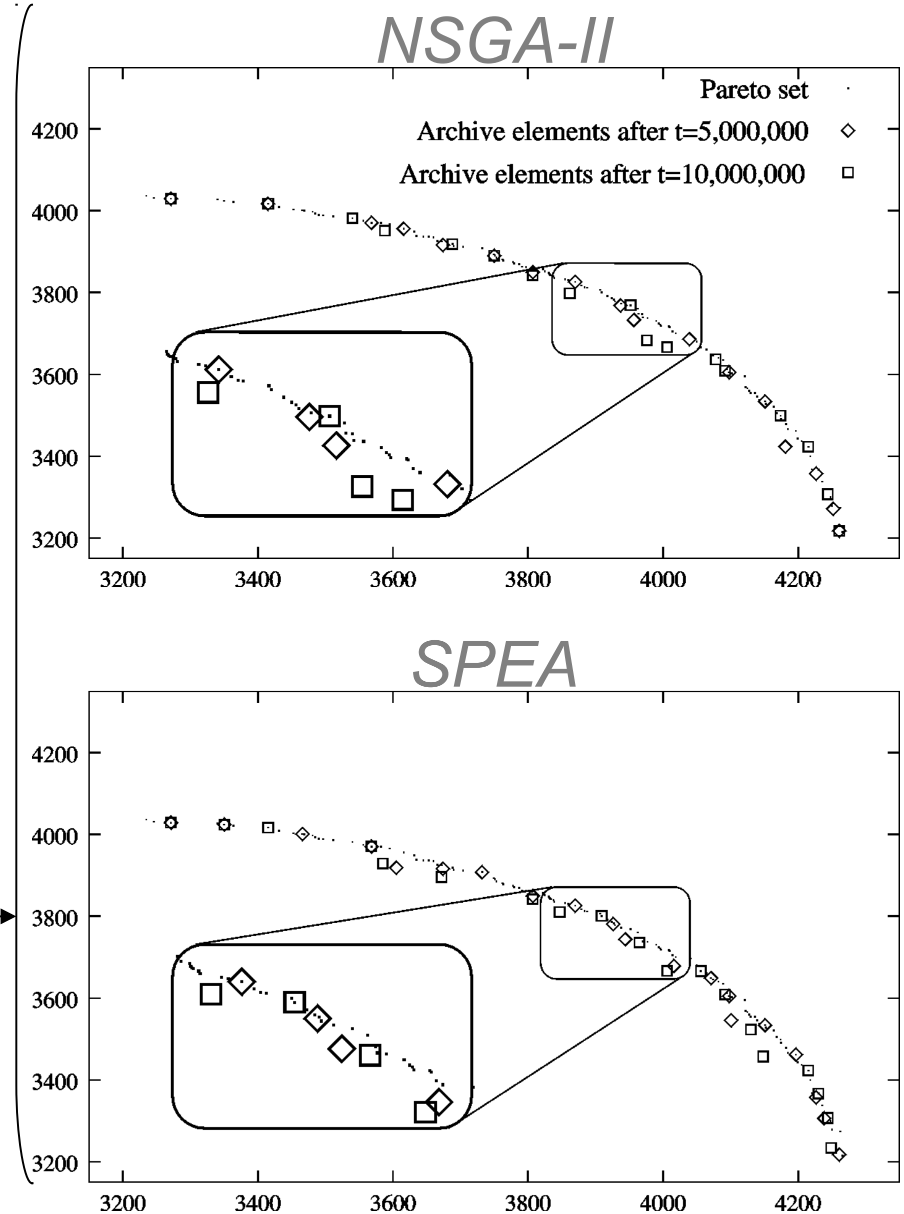
bounded archive (size 3)

- New solution accepted in t+1 is dominated by a solution found previously (and “lost” during the selection process)

Problem: Deterioration

Goal: Maintain “good” front
(distance + diversity)





But: Most archiving strategies may forget Pareto-optimal solutions…

Limit Behavior: Related Work

Requirements for archive:

1. Convergence
2. Diversity
3. Bounded Size

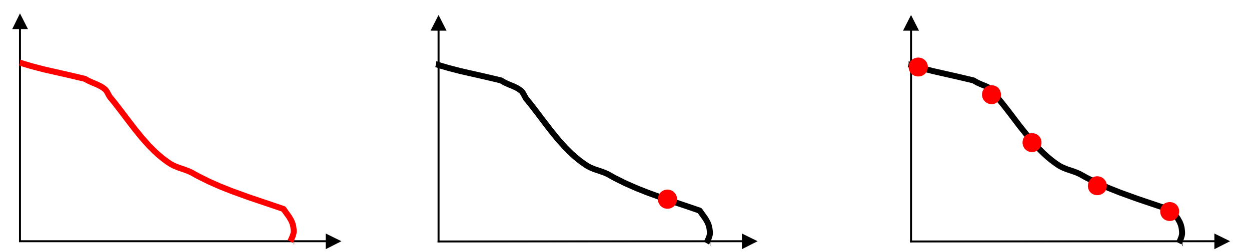
[Rudolph 98,00]
[Veldhuizen 99]

[Rudolph 98,00]
[Hanne 99]

in this work

“store all”

“store one”



(impractical)

(not nice)

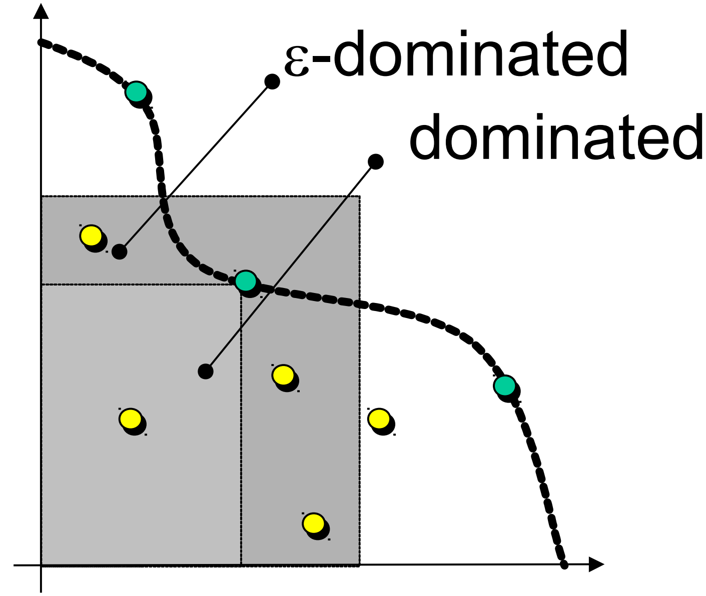
Solution Concept: Epsilon Dominance

Definition 1: ε-Dominance

A ε-dominates B iff

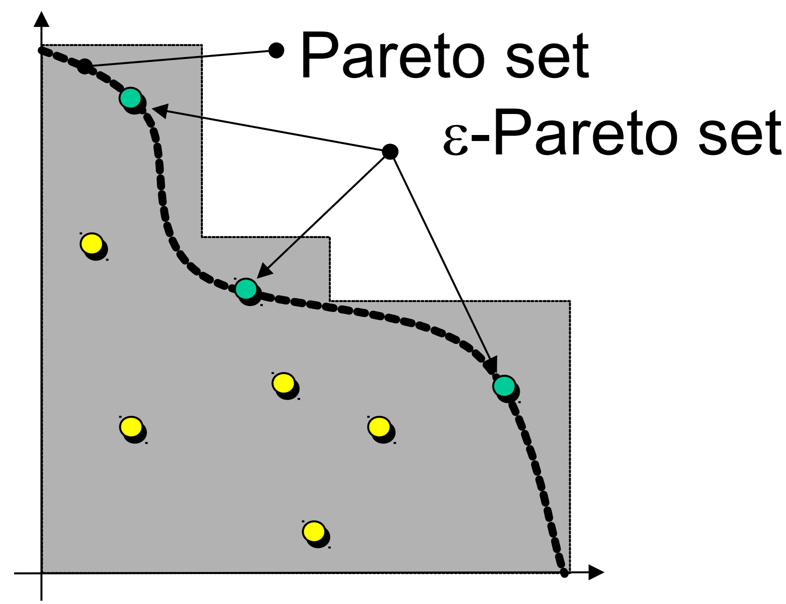
$$(1+\varepsilon)\cdot f(A) \geq f(B)$$

(known since 1987)



Definition 2: ε-Pareto set

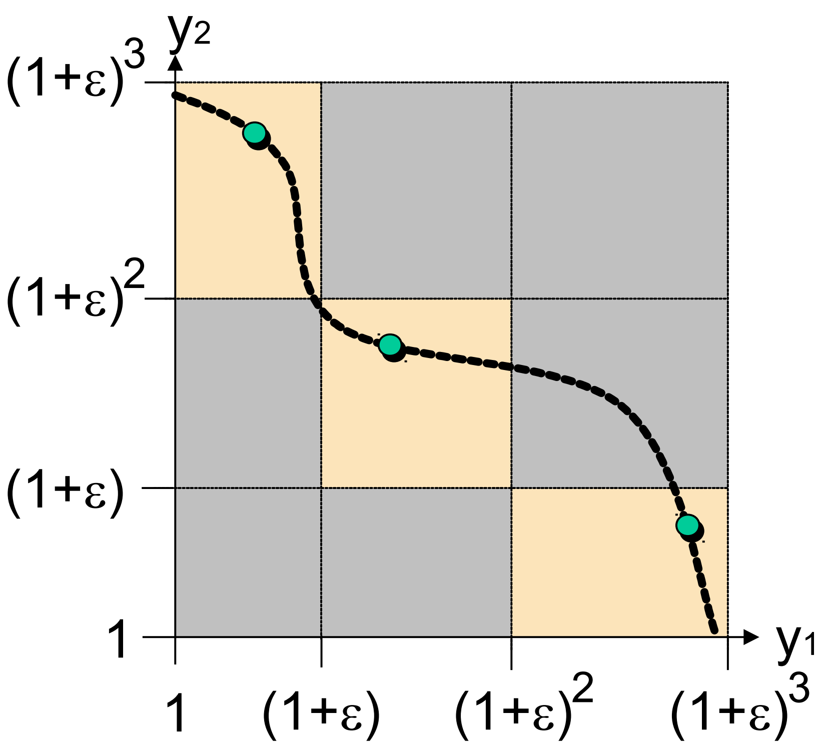
A subset of the Pareto-optimal set which ε-dominates all Pareto-optimal solutions



Keeping Convergence and Diversity

Goal: Maintain ε-Pareto set

Idea: ε-grid, i.e. maintain a set of non-dominated boxes (one solution per box)



Algorithm: (ε-update)

Accept a new solution f if

❶ the corresponding box is not dominated by any box represented in the archive A

AND

❷ any other archive member in the same box is dominated by the new solution

Correctness of Archiving Method

Theorem

Let $F = (f_1, f_2, f_3, \ldots)$ be an infinite sequence of objective vectors one by one passed to the ε-update algorithm, and F_t the union of the first t objective vectors of F.

Then for any $t > 0$, the following holds:

❶ the archive A at time t contains an ε-Pareto front of F_t

❷ the size of the archive A at time t is bounded by the term (K = "maximum objective value", m = "number of objectives")

$$\left(\frac{\log K}{\log(1+\varepsilon)}\right)^{m-1}$$

Correctness of Archiving Method

Sketch of Proof:

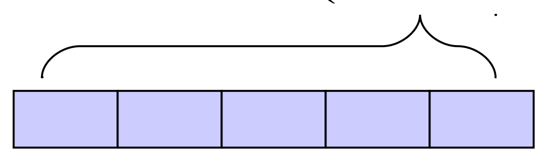
❶ 3 possible failures for A_t not being an ε-Pareto set of F_t (indirect proof)

- at time $k \leq t$ a necessary solution was missed
- at time $k \leq t$ a necessary solution was expelled
- A_t contains an $f \notin$ Pareto set of F_t

❷

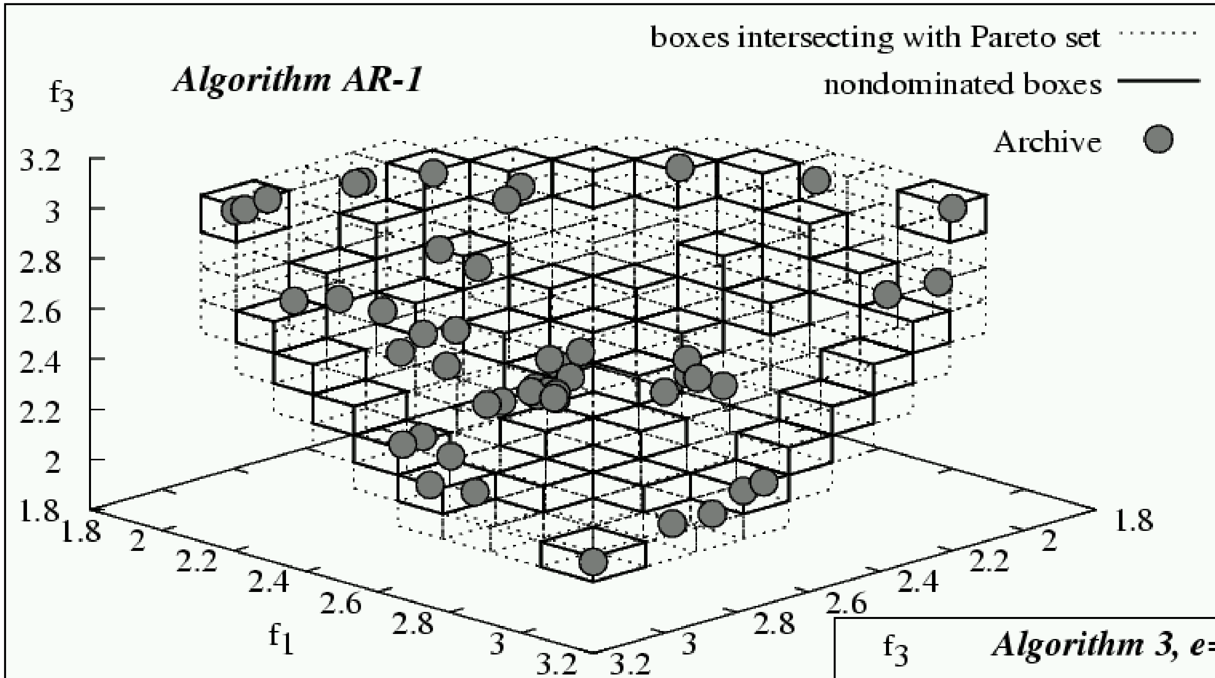
- Number of total boxes in objective space: $\left(\frac{\log K}{\log(1+\varepsilon)}\right)^m$
- Maximal one solution per box accepted
- Partition into $\left(\frac{\log K}{\log(1+\varepsilon)}\right)^{m-1}$ chains of boxes

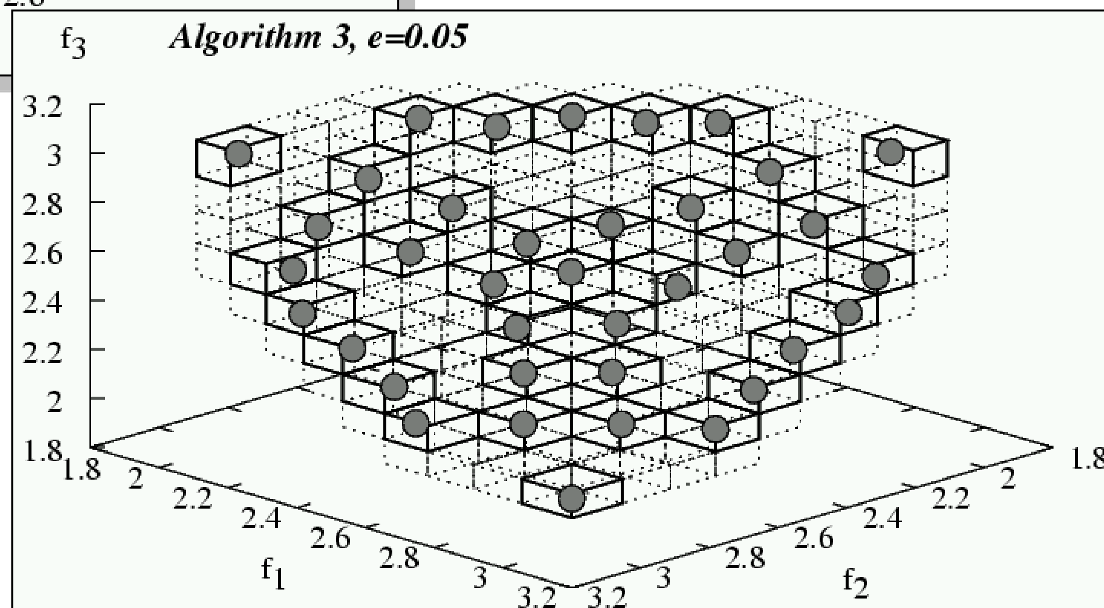
$$\left(\frac{\log K}{\log(1+\varepsilon)}\right)$$



Simulation Example

Rudolph and Agapie, 2000





Epsilon- Archive

Overview

❶ introduction

❷ limit behavior

❸ **run-time**

❹ performance measures

Running Time Analysis: Related Work

	problem domain	*type of results*	
Single-objective EAs	**discrete search spaces**	• expected RT (bounds) • RT with high probability (bounds)	*[Mühlenbein 92]* *[Rudolph 97]* *[Droste, Jansen, Wegener 98,02]* *[Garnier, Kallel, Schoenauer 99,00]* *[He, Yao 01,02]*
	continuous search spaces	• asymptotic convergence rates • exact convergence rates	*[Beyer 95,96,…]* *[Rudolph 97]* *[Jagerskupper 03]*
Multiobjective EAs		[none]	

Methodology

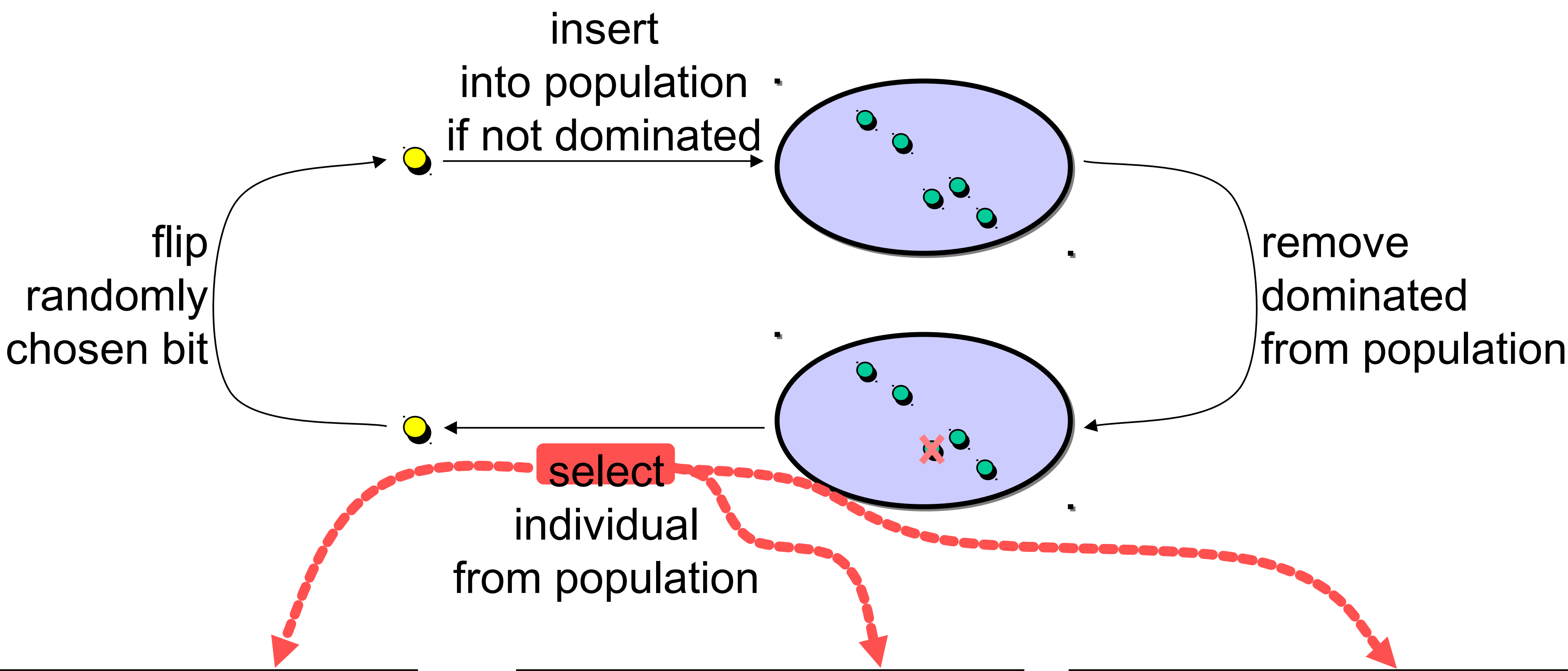
Typical "ingredients" of a Running Time Analysis:

Here:

- Simple algorithms ⇒ SEMO, FEMO, GEMO ("simple", "fair", "greedy")
- Simple problems ⇒ mLOTZ, mCOCZ (m-objective Pseudo-Boolean problems)
- Analytical methods & tools ⇒ General upper bound technique & Graph search process

1. Rigorous results for *specific* algorithm(s) on *specific* problem(s)
2. *General* tools & techniques
3. *General* insights (e.g., is a population beneficial at all?)

Three Simple Multiobjective EAs



Variant 1: SEMO

Each individual in the population is selected with the same probability (uniform selection)

Variant 2: FEMO

Select individual with minimum number of mutation trials (fair selection)

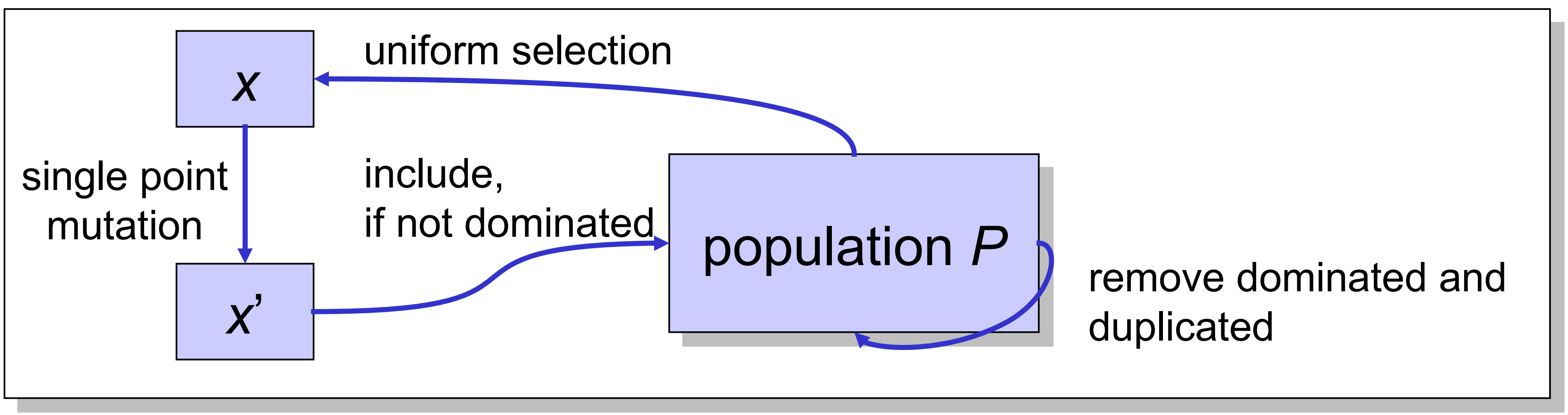
Variant 3: GEMO

Priority of convergence if there is progress (greedy selection)

SEMO

Algorithm 1 Simple Evolutionary Multi-Objective Optimizer (SEMO)

Choose an initial individual x uniformly from $X = \{0,1\}^n$
$P \leftarrow \{x\}$
loop
  Select one element x out of P uniformly.
  Create offspring x' by flipping a randomly chosen bit.
  $P \leftarrow P \setminus \{z \in P | x' \succ z\}$
  if $\nexists z \in P$ such that $(z \succ x' \vee f(x') = f(z))$ **then**
    $P \leftarrow P \cup \{x'\}$
  end if
end loop



Example Algorithm: SEMO

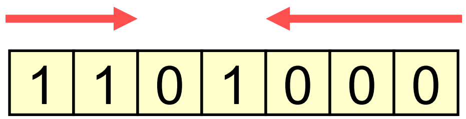
Simple Evolutionary Multiobjective Optimizer

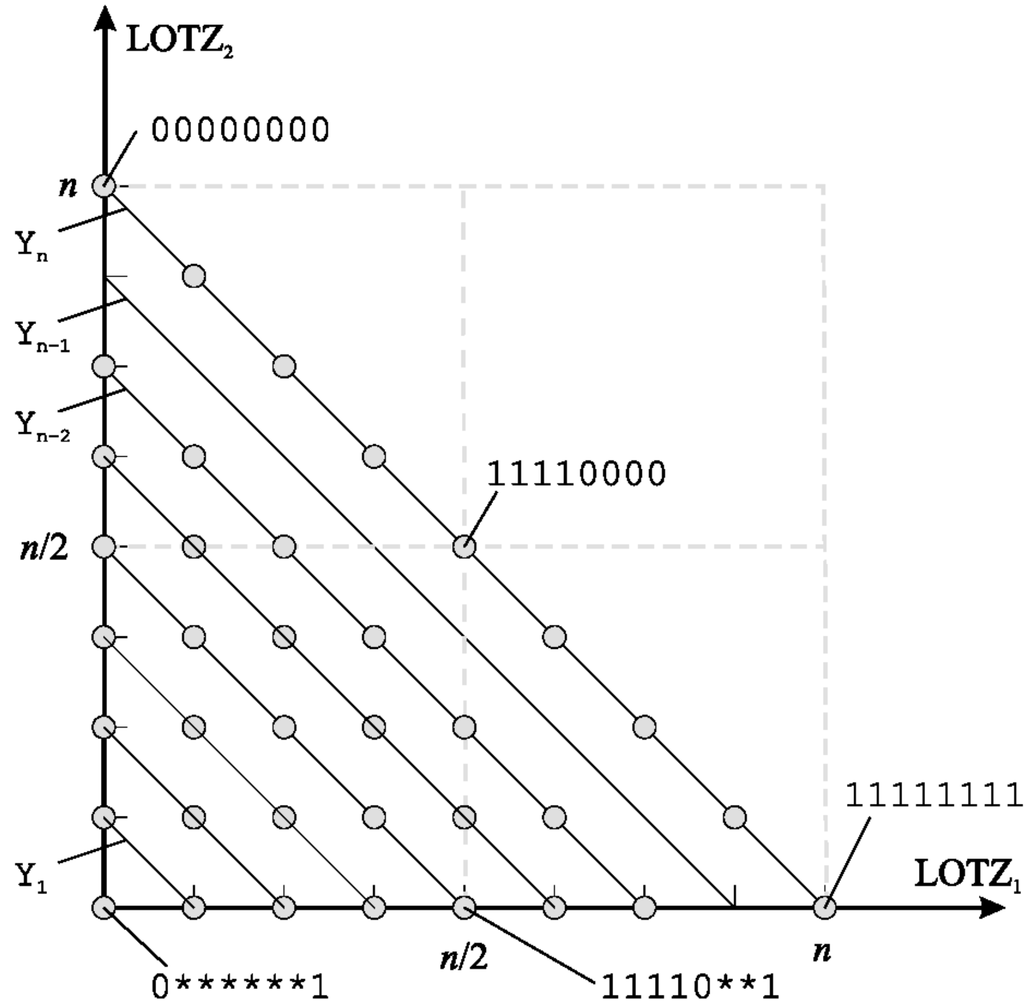
1. Start with a random solution
2. Choose parent randomly (uniform)
3. Produce child by variating parent
4. If child is not dominated then
   - add to population
   - discard all dominated
5. Goto 2

Run-Time Analysis: Scenario

Problem:

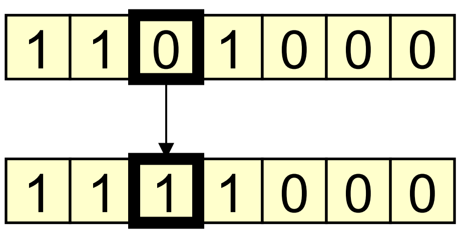
leading ones, trailing zeros (LOTZ)





Variation:

single point mutation

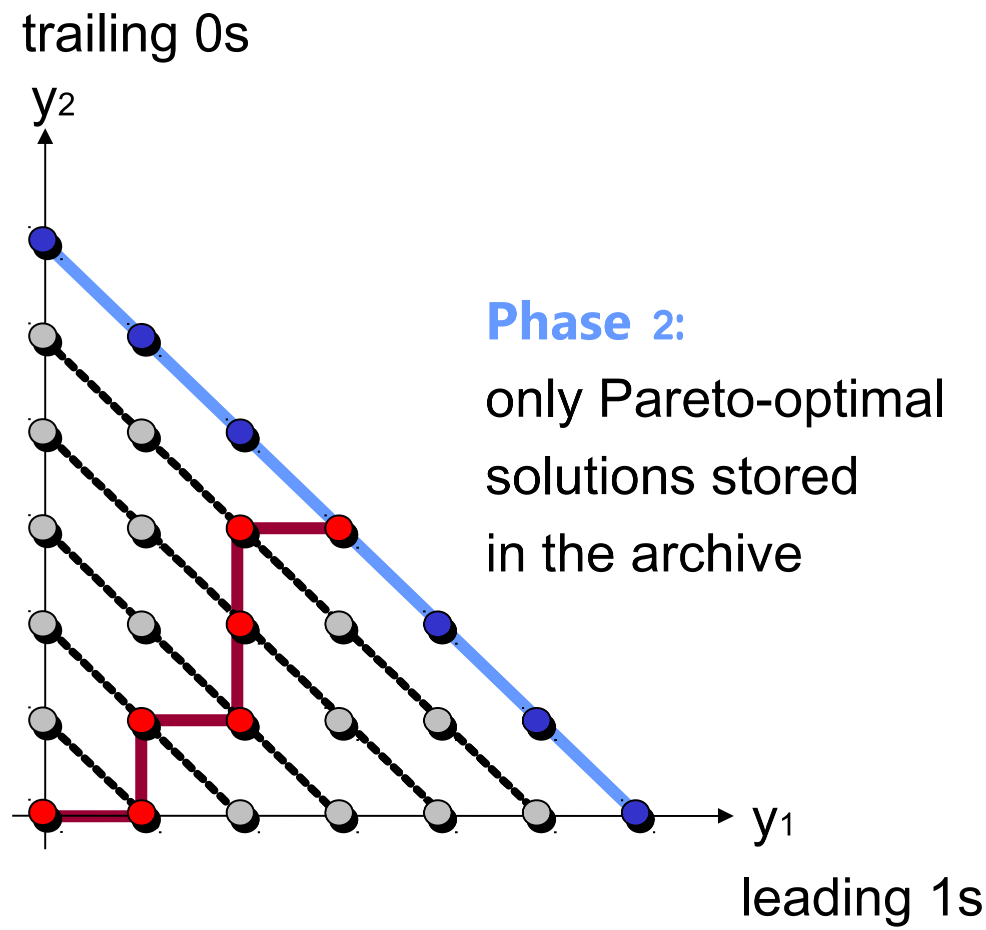


one bit per individual

The Good News

SEMO behaves like a single-objective EA until the Pareto set has been reached...

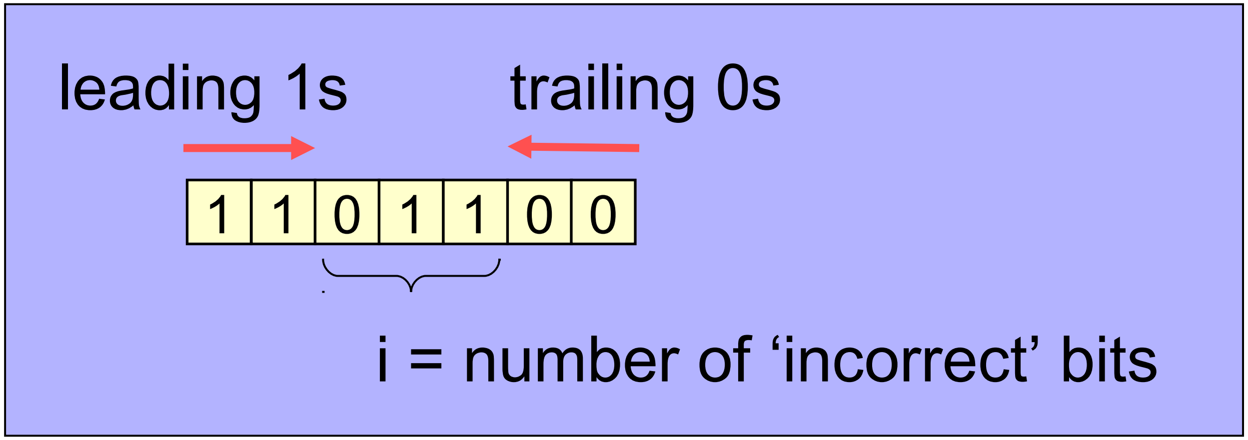
Phase 1:
only one
solution stored
in the archive



Phase 2:
only Pareto-optimal
solutions stored
in the archive

SEMO: Sketch of the Analysis I

Phase 1: until first Pareto-optimal solution has been found

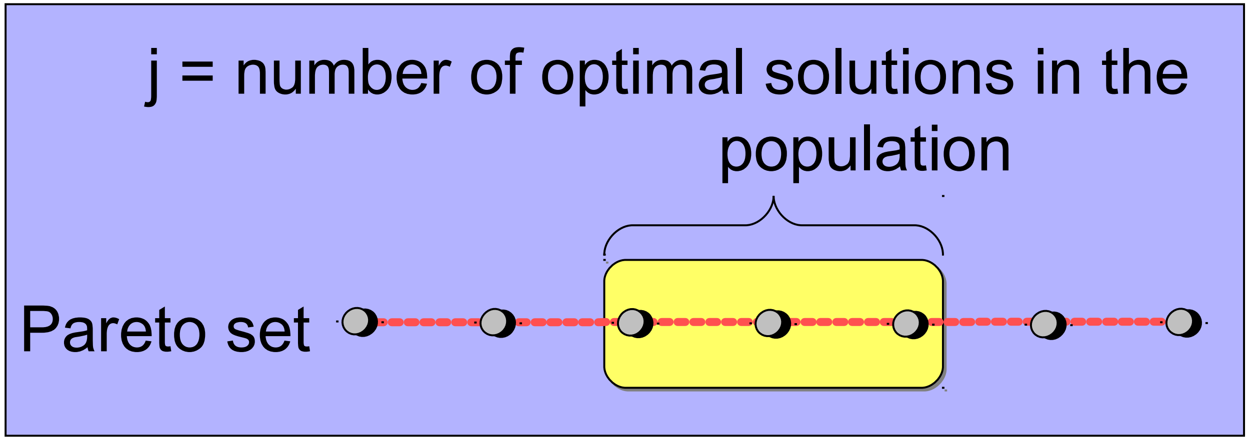


$i \rightarrow i\text{-}1$: probability of a successful mutation $\geq 1/n$
expected number of mutations = n

$i\text{=}n \rightarrow i\text{=}0$: at maximum n-1 steps (i=1 not possible)
expected overall number of mutations = $O(n^2)$

SEMO: Sketch of the Analysis II

Phase 2: from the first to all Pareto-optimal solutions



$j \rightarrow j+1$: probability of choosing an outer solution $\geq 1/j$, $\leq 2/j$
probability of a successful mutation $\geq 1/n$, $\leq 2/n$
expected number T_j of trials (mutations) $\geq nj/4$, $\leq nj$

$j=1 \rightarrow j=n$: at maximum n steps $\Rightarrow n^3/8 + n^2/8 \leq \sum T_j \leq n^3/2 + n^2/2$
expected overall number of mutations = $\Theta(n^3)$

SEMO on LOTZ

Lemma 1 (Expected running time for phase 1)
The expected running time of Alg. 1 until the first Pareto-optimal point is found is $O(n^2)$.

Lemma 2 (Expected running time for phase 2)
After the first Pareto-optimal point is found, the expected running time of Alg. 1 until all Pareto-optimal points are found is $\Theta(n^3)$

Corollary 1 (Expected running time Alg. 1)
The expected running time of Alg. 1 until all Pareto-optimal points are found is $\Theta(n^3)$

Can we do even better ?

- Our problem is the exploration of the Pareto-front.
- Uniform sampling is unfair as it samples early found Pareto-points more frequently.

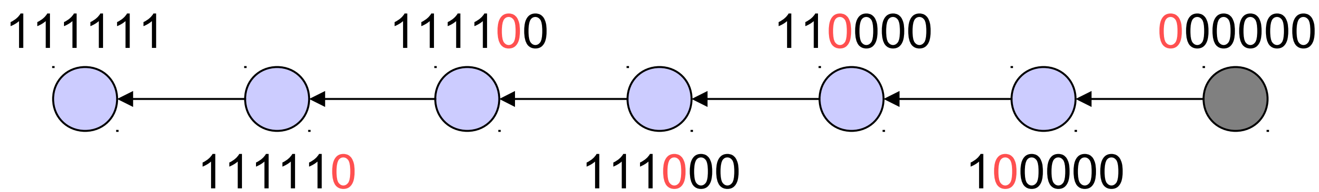
Algorithm 2 Fair Evolutionary Multi-Objective Optimizer (FEMO)

Choose an initial individual x uniformly from $X = \{0,1\}^n$
$w(x) \leftarrow 0$
$P \leftarrow \{x\}$
loop
  Select one element x out of $\{y \in P | w(y) \leq w(z) \forall z \in P\}$ uniformly.
  $w(x) \leftarrow w(x) + 1$
  Create offspring x' by flipping a randomly chosen bit.
  $P \leftarrow P \setminus \{z \in P | x' \succ z\}$
  if $\nexists z \in P$ such that $(z \succ x' \vee f(x') = f(z))$ **then**
    $P \leftarrow P \cup \{x'\}$
  end if
end loop

FEMO on LOTZ

Let the population P of FEMO applied to LOTZ contain exactly one Pareto-optimal solution and let $c > 0$ be an arbitrary constant. With probability at least $1 - n^{1-c}$, it takes at most $c \cdot n \log n$ mutation trials per solution to generate all remaining n Pareto-optimal solutions.

Sketch of Proof



Probability for each individual, that parent did not generate it with $c/p \log n$ trials:

$$(1-p)^t = (1 - \frac{1}{1/p})^{c/p \log n} = (1 - \frac{1}{1/p})^{1/p^{c \log n}} \leq \left(\frac{1}{e}\right)^{c \log n} = \frac{1}{n^c}$$

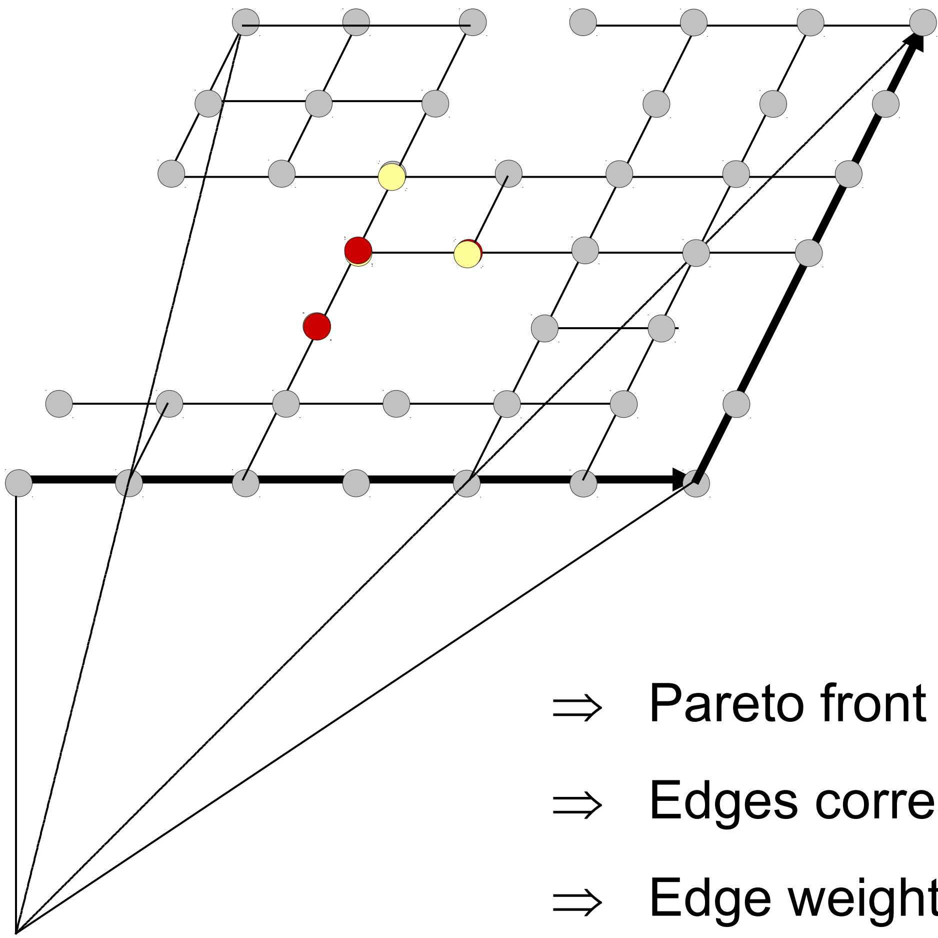
n individuals must be produced. The probability that one needs more than $c/p \log n$ trials is bounded by n^{1-c}.

FEMO on LOTZ

With probability at least $1 - O(1/n)$, the running time FEMO needs from the discovery of the first two Pareto-optimal objective vectors of LOTZ until the whole Pareto set has been found lies in the interval $[1/4 \cdot 1/p \cdot n \log n, 2 \cdot 1/p \cdot n \log n]$. Hence, $\mathsf{P}\{T = \Theta(1/p \cdot n \log n)\} = 1 - O(1/n)$. Furthermore, $\mathsf{E}[T] = O(1/p \cdot n \log n)$.

- Single objective (1+1) EA with multistart strategy (epsilon-constraint method) has running time $\Theta(n^3)$.
- EMO algorithm with fair sampling has running time $\Theta(n^2 \log(n))$.

Generalization: Randomized Graph Search



⇒ Pareto front can be modeled as a graph

⇒ Edges correspond to mutations

⇒ Edge weights are mutation probabilities

How long does it take to explore the whole graph?

How should we select the “parents”?

Randomized Graph Search

Algorithm 9 Randomized Graph Search

$w(v) \leftarrow 0$
$V \leftarrow \{v_1\}$; $E \leftarrow \emptyset$
loop
    Select a node v out of $\{v' \in V | w(v) \leq w(v') \; \forall v' \in V\}$ uniformly.
    $w(v) \leftarrow w(v) + 1$
    $v' \leftarrow \mathsf{jump}(v)$
    if $v' \notin V$ **then**
        $w(v) \leftarrow 0$
        $V \leftarrow V \cup \{v'\}$; $E \leftarrow E \cup \{(v, v')\}$
    end if
end loop

With probability at least $1 - |E|^{-c}$, Algorithm 9 explores all nodes and edges of G using not more that $(c+1)\frac{|V|}{p} \log |E|$ calls to the function jump*. The expected number of calls to* jump *is bounded by $O(\frac{|V|}{p} \log |E|)$.*

Running Time Analysis: Comparison

Algorithms

	(1+1)-EMO	SEMO	FEMO	GEMO
LOTZ	$\boxed{\Theta(n^3)}$	$\Theta(n^3)$	$\Theta(n^2 \log n)$	$\boxed{\Theta(n^2 \log n)}$
COCZ	$\boxed{\Theta(n^2 \log n)}$	$O(n^2 \log n)$	$O(n^2 \log n)$	$\boxed{\Theta(n^2)}$
mLOTZ	$\boxed{\Theta(n^{m/2} n^2)}$	$O(n^{m+1})$	$O(n^{m+1})$	$\boxed{O(n^{m/2} n \log n)}$
mCOCZ	$\Theta(n^{m/2} n \log n)$	$O(n^{m+1})$	$O(n^{m+1})$	$O(n^{m/2} n \log n)$

Problems

Population-based approach can be more efficient than multistart of single membered strategy

Overview

❶ introduction

❷ limit behavior

❸ run-time

❹ **performance measures**

Performance Assessment: Approaches

Which technique is suited for which problem class?

❶ **Theoretically (by analysis):** difficult

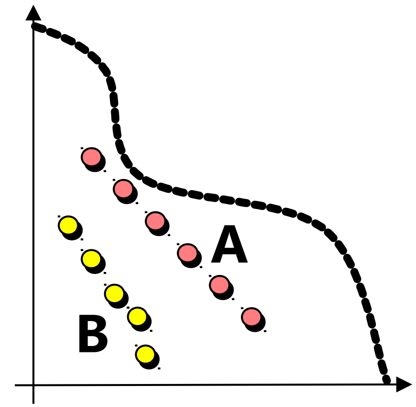
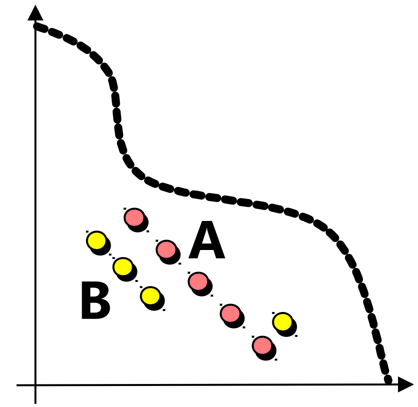
- Limit behavior (unlimited run-time resources)
- Running time analysis

❷ **Empirically (by simulation):** standard

Problems: randomness, multiple objectives

Issues: quality measures, statistical testing, benchmark problems, visualization, …

The Need for Quality Measures

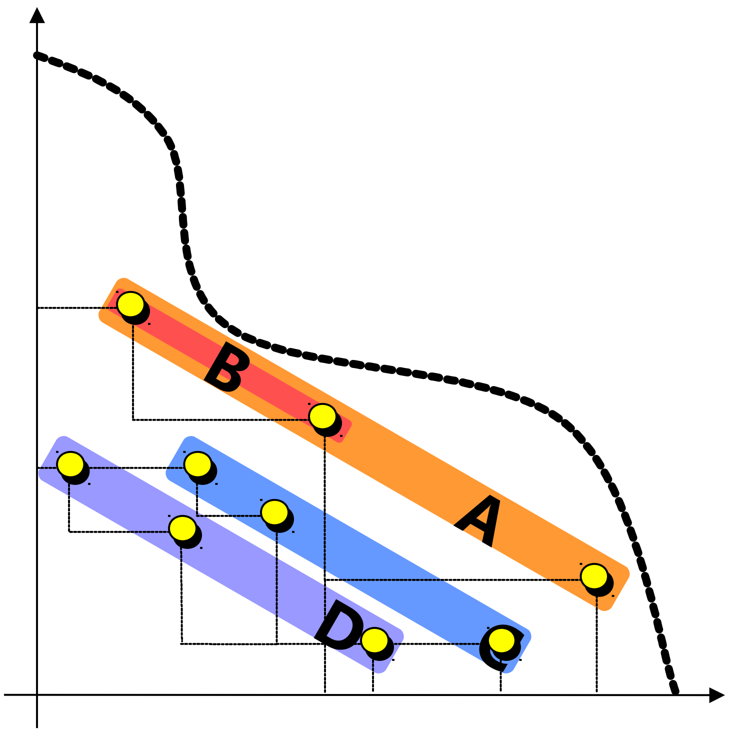
		
	Is A better than B?	
independent of user preferences	Yes (strictly)	No
dependent on user preferences	How much?	In what aspe

Ideal: quality measures allow to make both type of statements

Independent of User Preferences

Pareto set approximation (algorithm outcome) = set of incomparable solutions

Ω = set of all Pareto set approximations



A **weakly dominates** **B**
= not worse in all objectives
sets not equal

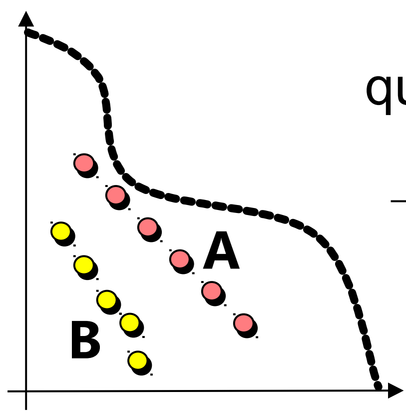
C **dominates** **D**
= better in at least one objective

A **strictly dominates** **C**
= better in all objectives

B is **incomparable** to **C**
= neither set weakly better

Dependent on User Preferences

Goal: Quality measures compare two Pareto set approximations A and B.



application of quality measures (here: unary)

	A	B
hypervolume	*432.34*	*420.13*
distance	*0.3308*	
diversity	*0.3637*	*0.3463*
spread	*0.3622*	*0.3601*
cardinality	*6*	*5*

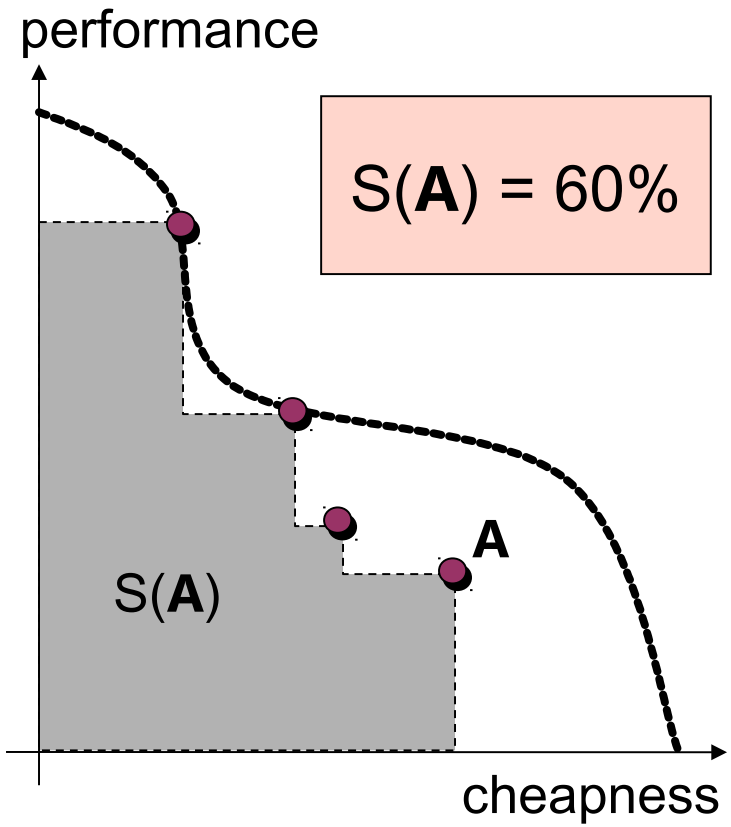
comparison and interpretation of quality values

“**A** better”

Quality Measures: Examples

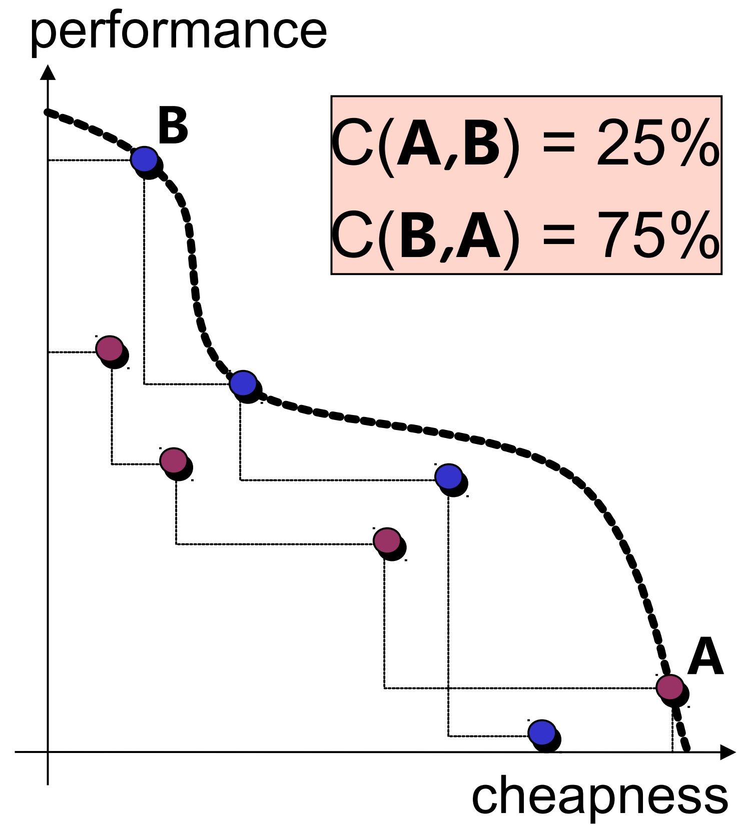
Unary

Hypervolume measure



Binary

Coverage measure



[Zitzler, Thiele: 1999]

Previous Work on Multiobjective Quality Measures

Status:

- Visual comparisons common until recently
- Numerous quality measures have been proposed since the mid-1990s
  [Schott: 1995][Zitzler, Thiele: 1998][Hansen, Jaszkiewicz: 1998][Zitzler: 1999]
  [Van Veldhuizen, Lamont: 2000][Knowles, Corne , Oates: 2000][Deb et al.: 2000] [Sayin: 2000][Tan, Lee, Khor: 2001][Wu, Azarm: 2001]…
- Most popular: unary quality measures (diversity + distance)
- No common agreement which measures should be used

Open questions:

- What kind of statements do quality measures allow?
- Can quality measures detect *whether* or *that* a Pareto set approximation is better than another?

[Zitzler, Thiele, Laumanns, Fonseca, Grunert da Fonseca: 2003]

Comparison Methods and Dominance Relations

Compatibility of a comparison method C:

C yields `true` $\Rightarrow$ **A** is (weakly, strictly) better than **B**

(C detects **that** **A** is (weakly, strictly) better than **B**)

Completeness of a comparison method C:

A is (weakly, strictly) better than **B** $\Rightarrow$ C yields `true`

Ideal: compatibility and completeness, i.e.,

A is (weakly, strictly) better than **B** $\Leftrightarrow$ C yields `true`

(C detects **whether** **A** is (weakly, strictly) better than **B**)

Limitations of Unary Quality Measures

Theorem:

There exists no unary quality measure that is able to detect *whether* **A** is better than **B**.

This statement even holds, if we consider a finite combination of unary quality measures.

There exists no combination of unary measures applicable to any problem.

Power of Unary Quality Indicators

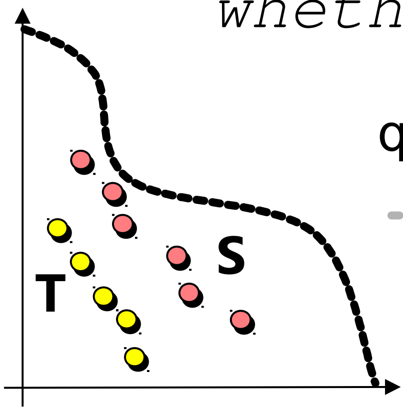
indicator	name / reference	Boolean function	compatibility	completeness
I_{HC}	enclosing hypercube indicator / Section III-B.1	$I_2^{HC}(A) < I_1^{HC}(B)$	$\succ\succ$	-
I_O	objective vector indicator / Section III-B.1	$I_i^{O}(A) < I_i^{O}(B)$	$\succ\succ$	-
I_H	hypervolume indicator / [7]	$I_H(A) > I_H(B)$	$\not\trianglerighteq$	$\triangleright$
I_W	average best weight combination / [19]	$I_W(A) < I_W(B)$	$\not\trianglerighteq$	$\succ\succ$
I_D	distance from reference set / [20]	$I_D(A) < I_D(B)$	$\not\trianglerighteq$	$\succ\succ$
$I_{\epsilon 1}$	unary ϵ-indicator / Section III-B.2	$I_{\epsilon 1}(A) < I_{\epsilon 1}(B)$	$\not\trianglerighteq$	$\succ\succ$
I_{PF}	fraction of Pareto-optimal front covered / [22]	$I_{PF}(A) > I_{PF}(B)$	$\not\trianglerighteq$	-
I_P	number of Pareto points contained / Section III-B.2	$I_P(A) > I_P(B)$	$\not\trianglerighteq$	-
I_{ER}	error ratio / [13]	$I_{ER}(A) > 0$	$\not\succ$	-
I_{CD}	chi-square-like deviation indicator / [14]	$I_{CD}(A) < I_{CD}(B)$	-	-
I_S	spacing / [23]	$I_S(A) < I_S(B)$	-	-
I_{ONVG}	overall nondominated vector generation / [13]	$I_{ONVG}(A) > I_{ONVG}(B)$	-	-
I_{GD}	generational distance / [13]	$I_{GD}(A) < I_{GD}(B)$	-	-
I_{ME}	maximum Pareto front error / [13]	$I_{ME}(A) < I_{ME}(B)$	-	-
I_{MS}	maximum spread / [21]	$I_{MS}(A) > I_{MS}(B)$	-	-
I_{MD}	minimum distance between two solutions / [24]	$I_{MD}(A) > I_{MD}(B)$	-	-
I_{CE}	coverage error / [24]	$I_{CE}(A) < I_{CE}(B)$	-	-
I_{DU}	deviation from uniform distribution / [25]	$I_{DU}(A) < I_{DU}(B)$	-	-
I_{OS}	Pareto spread / [26]	$I_{OS}(A) > I_{OS}(B)$	-	-
I_A	accuracy / [26]	$I_A(A) > I_A(B)$	-	-
I_{NDC}	number of distinct choices / [26]	$I_{NDC}(A) > I_{NDC}(B)$	-	-
I_{CL}	cluster / [26]	$I_{CL}(A) < I_{CL}(B)$	-	-

rictly dominates doesn’t weakly dominate doesn’t dominate weakly dominates

Quality Measures: Results

Basic question: Can we say on the basis of the quality measures

whether or *that* an algorithm outperforms another?



application of quality measures

	S	T
hypervolume	432.34	420.13
distance	0.3308	
diversity	0.3637	0.3463
spread	0.3622	0.3601
cardinality	6	5

There is no combination of unary quality measures such that
S is better than **T** in all measures is equivalent to **S**

Unary quality measures usually do not tell that
S dominates **T**; at maximum that **S** does not dominate **T**

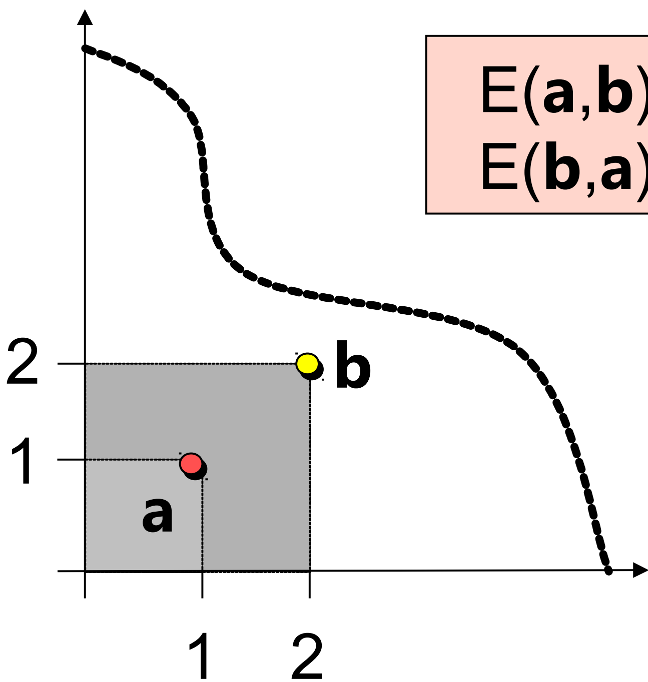
[Zitzler et al.: 2002]

A New Measure: ε-Quality Measure

Two solutions:

$E(\mathbf{a},\mathbf{b}) =$

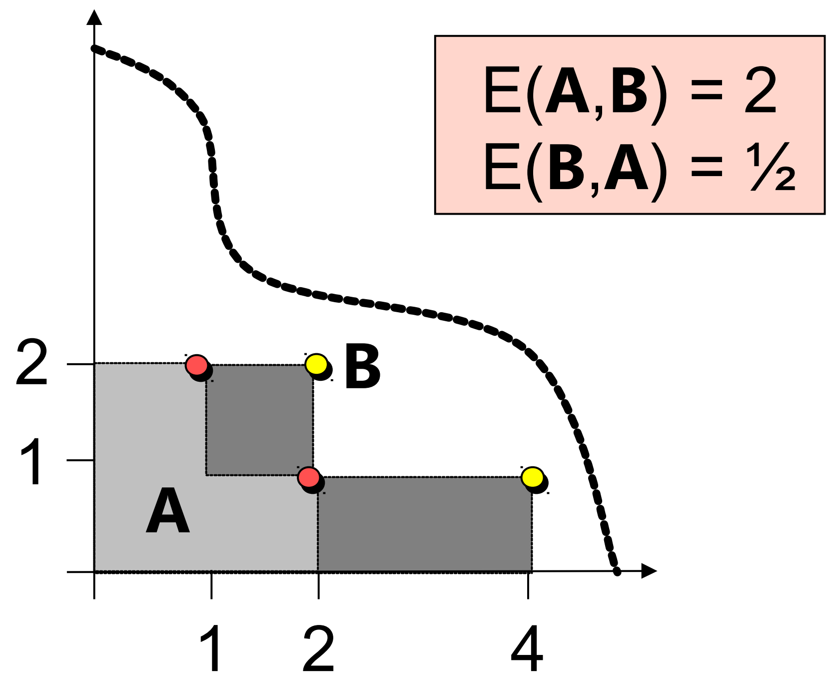
$\max_{1 \le i \le n} \min_{\varepsilon} \varepsilon \cdot f_i(\mathbf{a}) \ge f_i(\mathbf{b})$



Two approximations:

$E(\mathbf{A},\mathbf{B}) =$

$\max_{\mathbf{b} \in \mathbf{B}} \min_{\mathbf{a} \in \mathbf{A}} E(\mathbf{a},\mathbf{b})$



Advantages: allows all kinds of statements (complete **and** compatible)

Selected Contributions

How to apply (evolutionary) optimization algorithms to large-scale multiobjective optimization problems?

Algorithms:

- Improved techniques *[Zitzler, Thiele: IEEE TEC1999]* *[Zitzler, Teich, Bhattacharyya: CEC2000]* *[Zitzler, Laumanns, Thiele: EUROGEN2001]*
- Unified model *[Laumanns, Zitzler, Thiele: CEC2000]* *[Laumanns, Zitzler, Thiele: EMO2001]*
- Test problems *[Zitzler, Thiele: PPSN 1998, IEEE TEC 1999]* *[Deb, Thiele, Laumanns, Zitzler: GECCO2002]*

Theory:

- Convergence/diversity *[Laumanns, Thiele, Deb, Zitzler:*