南京大学 人工智能学院
SCHOOL OF ARTIFICIAL INTELLIGENCE, NANJING UNIVERSITY

UNIVERSITY OF BIRMINGHAM

Stochastic Population Update Can Provably Be Helpful in Multi-Objective Evolutionary Algorithms*

Chao Bian, Yawen Zhou, Miqing Li, **Chao Qian**

http://www.lamda.nju.edu.cn/qianc/
Email: qianc@nju.edu.cn

LAMDA Group
Nanjing University, China

*Published at IJCAI'23

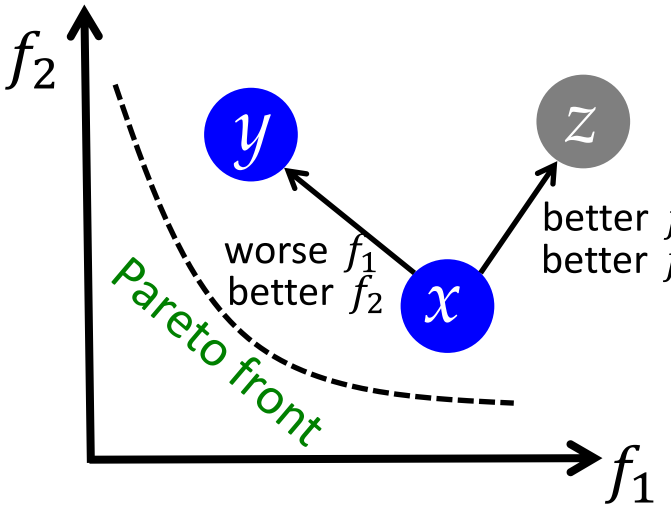
Multi-objective Optimization

Multi-objective optimization tries to optimize multiple objectives simultaneously

$$\min_{\boldsymbol{s}\in S}\ (f_1(\boldsymbol{s}), f_2(\boldsymbol{s}), \dots, f_m(\boldsymbol{s}))$$

x *dominates* ***z***: $f_1(\boldsymbol{x}) < f_1(\boldsymbol{z}) \wedge f_2(\boldsymbol{x}) < f_2(\boldsymbol{z})$

x *is incomparable with* ***y***: $f_1(\boldsymbol{x}) > f_1(\boldsymbol{y}) \wedge f_2(\boldsymbol{x}) < f_2(\boldsymbol{y})$



Pareto optimal solution: a solution that cannot be dominated by any other solution in S

Pareto front: the set of objective vectors of all the Pareto optimal solutions, which represents different optimal trade-offs between objectives

Goal: finding the Pareto front or its good approximation

Multi-objective Optimization

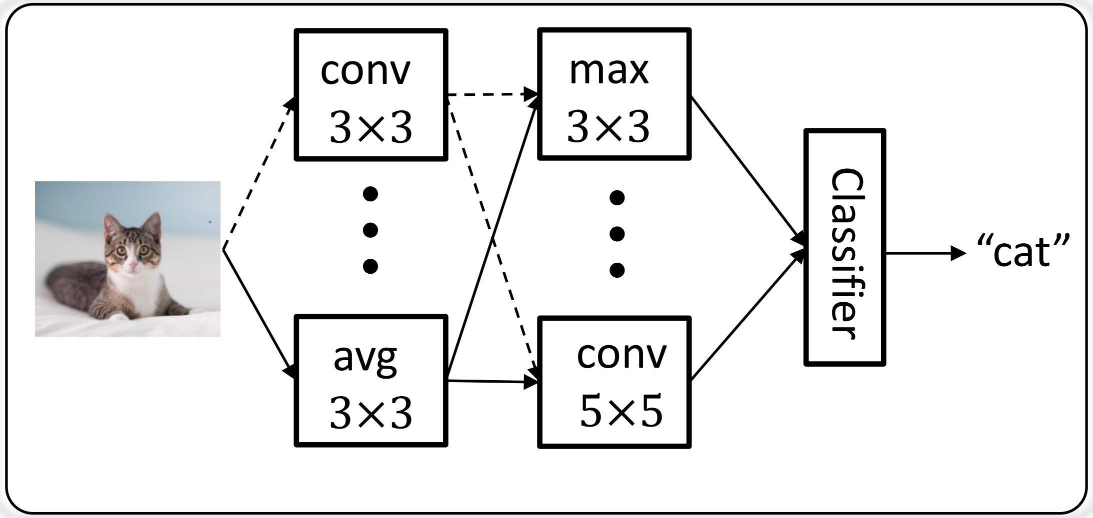
Multi-objective optimization has many applications:

Buy cars

- Max: performance
- Min: price

Search neural architectures

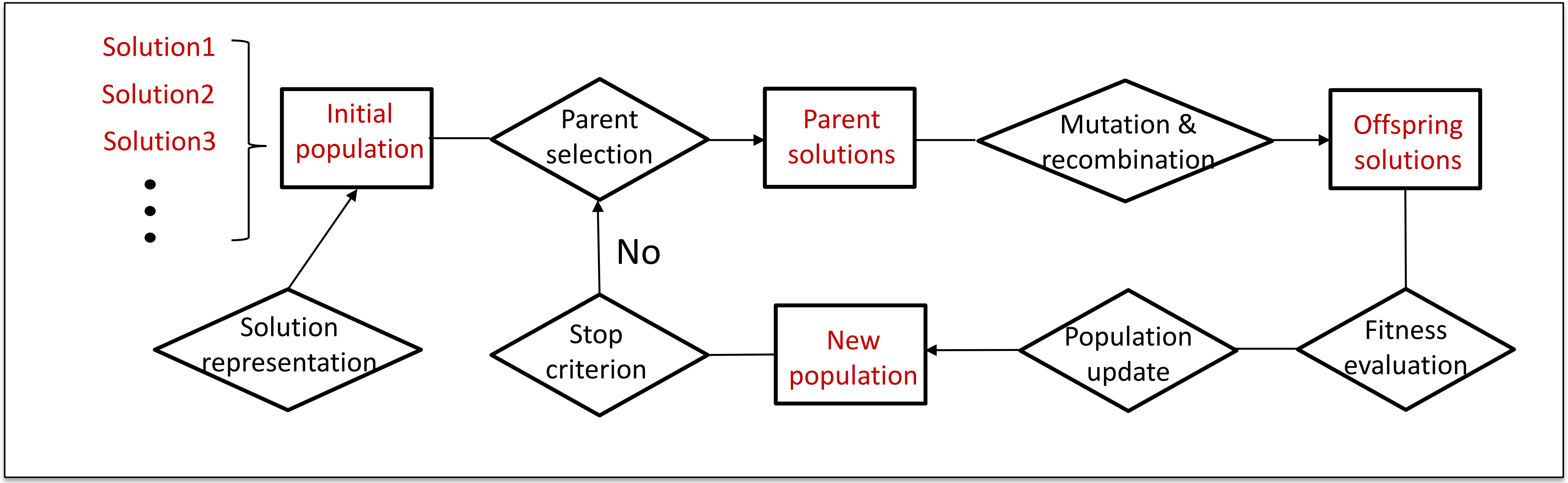
- Max: accuracy
- Min: network complexity



Evolutionary Algorithms (EAs)

LAMDA
Learning And Mining from DatA
http://www.lamda.nju.edu.cn

The general structure of EAs



The population-based nature makes EAs suitable for solving multi-objective optimization problems

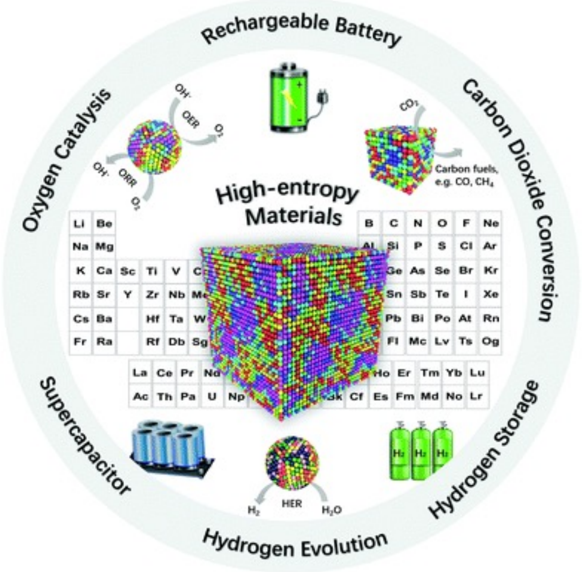
Applications of Multi-objective EAs (MOEAs)

MOEAs have been widely applied for solving real-world multi-objective tasks

High entropy alloy design

[Menou et al, Materials and Design'18]

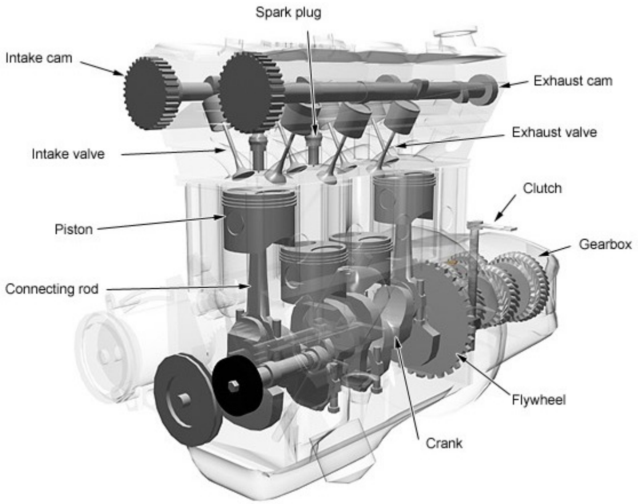
- Max: strength
- Min: density



Gasoline engine design

[Fujita et al, DETC'98]

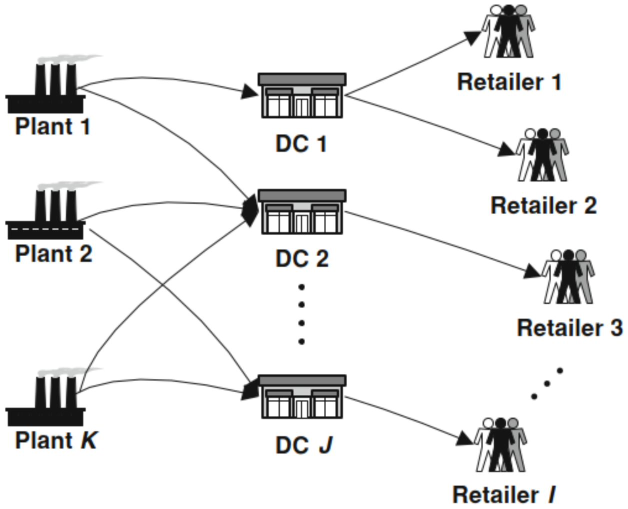
- Max: acceleration ability
- Min: fuel consumption



Supply chain design

[Benyoucef and Xie, Springer'11]

- Max: service quality
- Min: cost



Popular MOEAs

Pareto dominance based: NSGA-II, SPEA-II, …

K. Deb, A. Pratap, S. Agarwal and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 2002. (Google scholar citations: 47549)

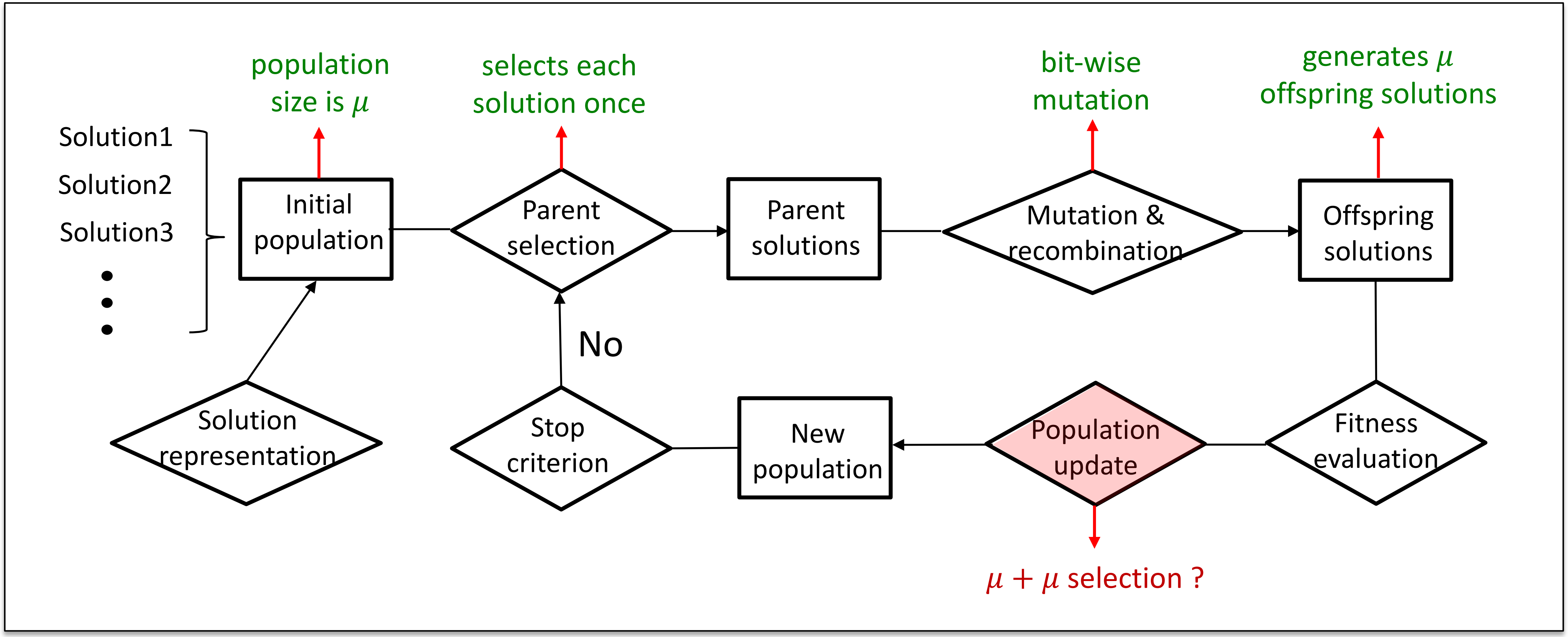
Performance indicator based: SMS-EMOA , HyPE, ….

N. Beume, B. Naujoks and M. Emmerich. SMS-EMOA: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research*, 2007. (Google scholar citations: 2009)

Decomposition based: MOEA/D, ….

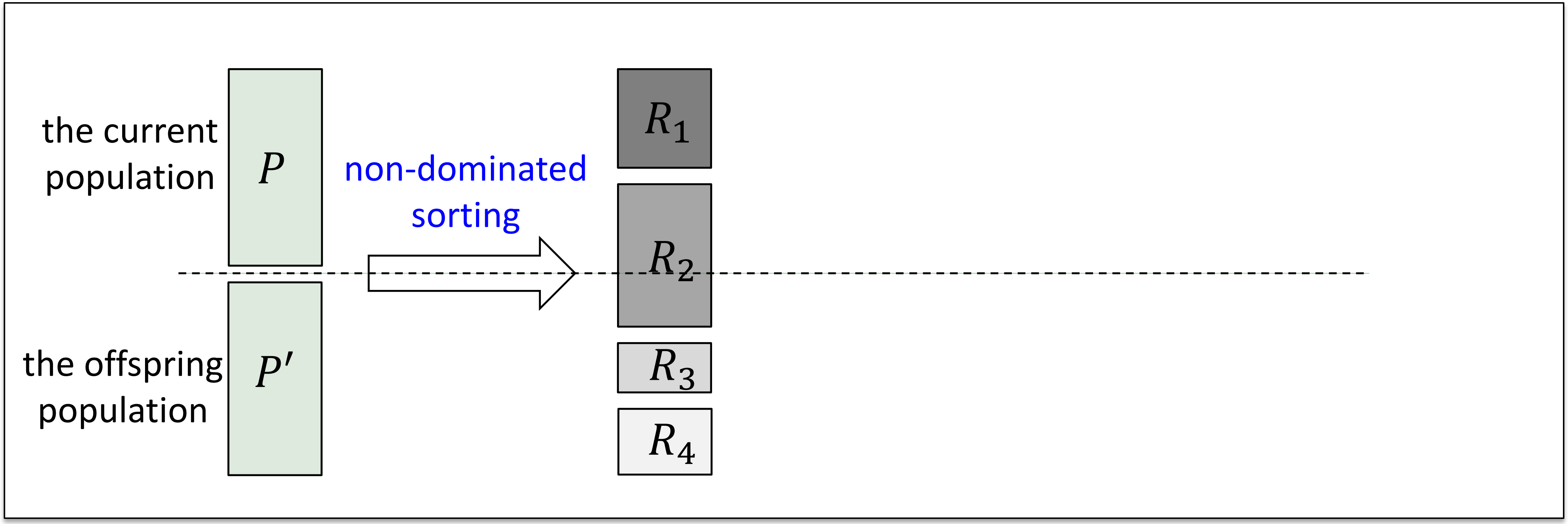
Q. Zhang and H. Li. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation*, 2007. (Google scholar citations: 8329)

Framework of NSGA-II:



Population Update of NSGA-II:

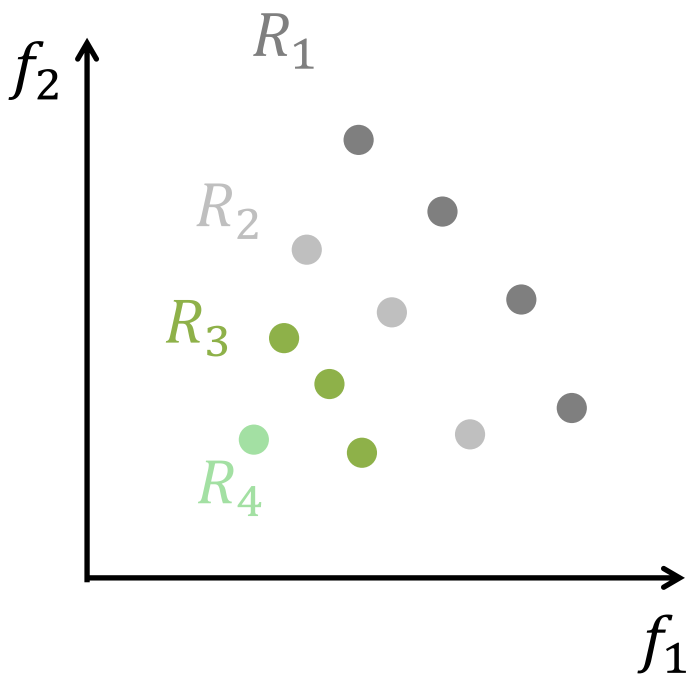
Use non-dominated sorting and crowding distance sorting to rank the solutions, and delete the worst ones



Non-dominated sorting

Partition the solutions in $P \cup P'$ into $R_1, R_2, \ldots, R_v$

- solutions in R_1 (has rank 1): cannot be dominated by any solution in $P \cup P'$
- solutions in R_2 (has rank 2): cannot be dominated by any solution in $(P \cup P') \backslash R_1$
- …

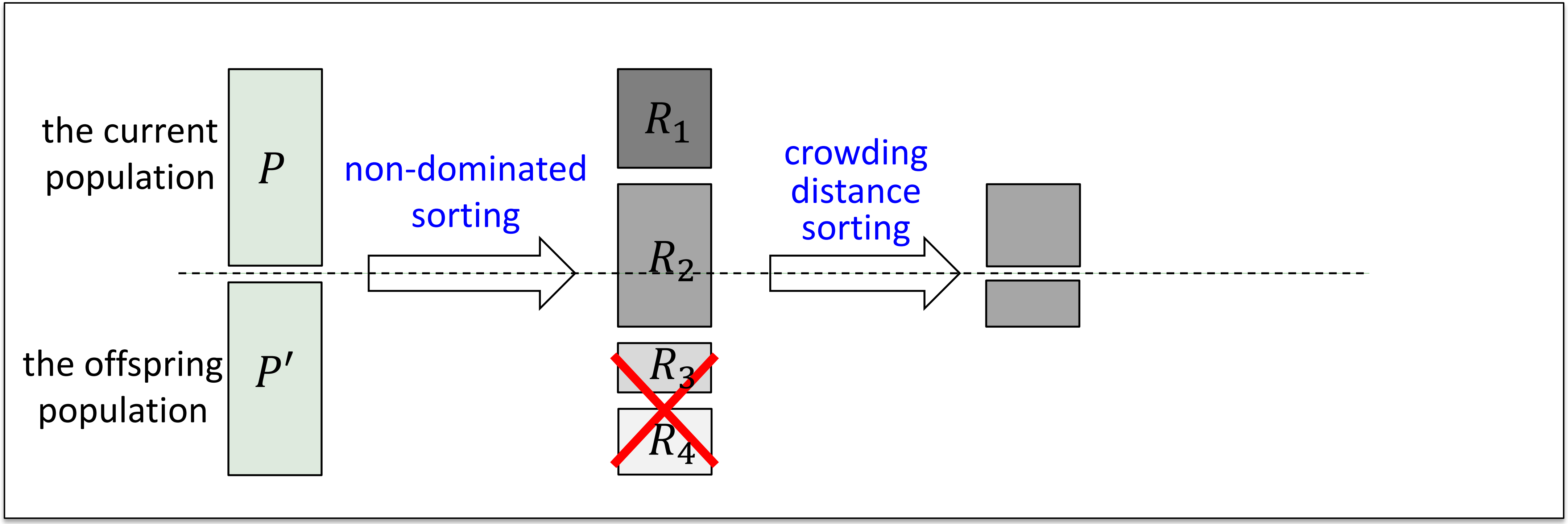


Rank reflects the convergence of a solution

Solutions with smaller rank are better

Population Update of NSGA-II:

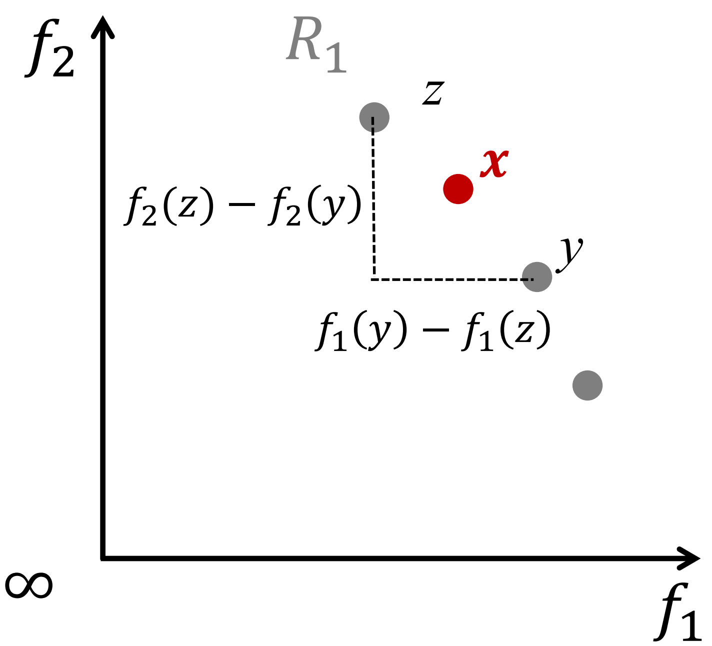
Use non-dominated sorting and crowding distance sorting to rank the solutions, and delete the worst ones



Crowding distance assignment

For each objective f_i:

- sort the solutions w.r.t. f_i (ascending)
- for each solution $\boldsymbol{x}$, compute the normalized distance w.r.t. f_i, and add the distance to the final crowding distance value of $\boldsymbol{x}$



normalized distance:

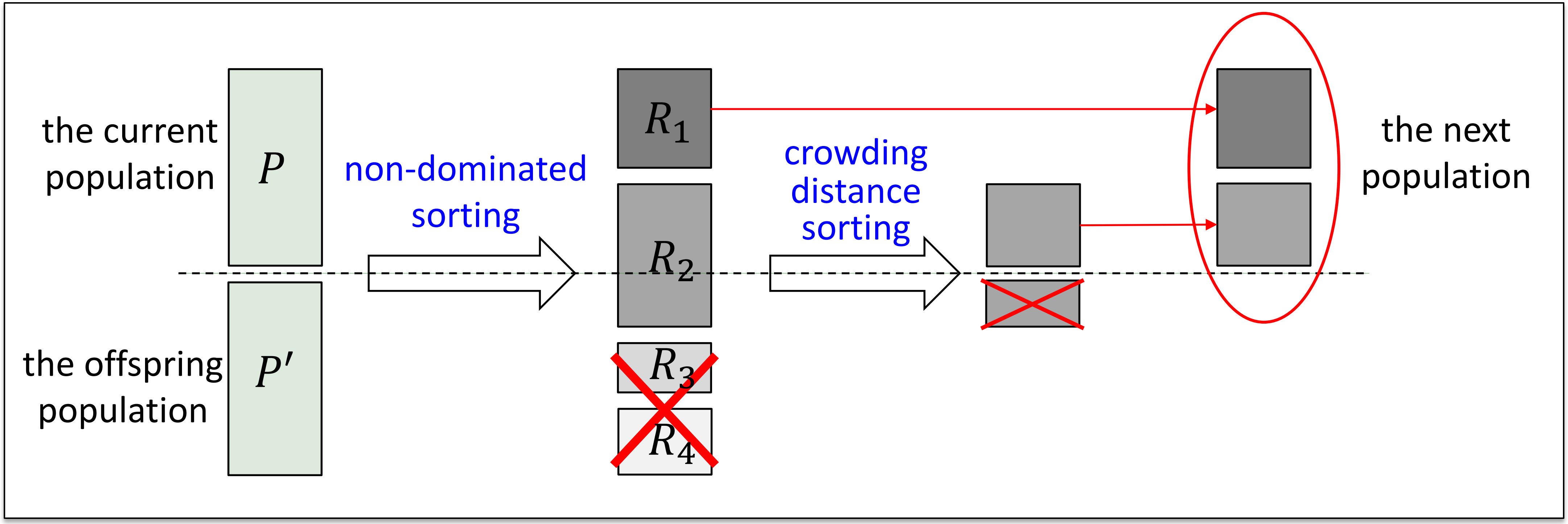
- if $\boldsymbol{x}$ is sorted in the first or last position, set the distance to ∞
- otherwise, set the distance to $\frac{f_i(\mathrm{succ}(\boldsymbol{x})) - f_i(\mathrm{prec}(\boldsymbol{x}))}{\max f_i - \min f_i}$ → Normalization

Crowding distance reflects the diversity of a solution

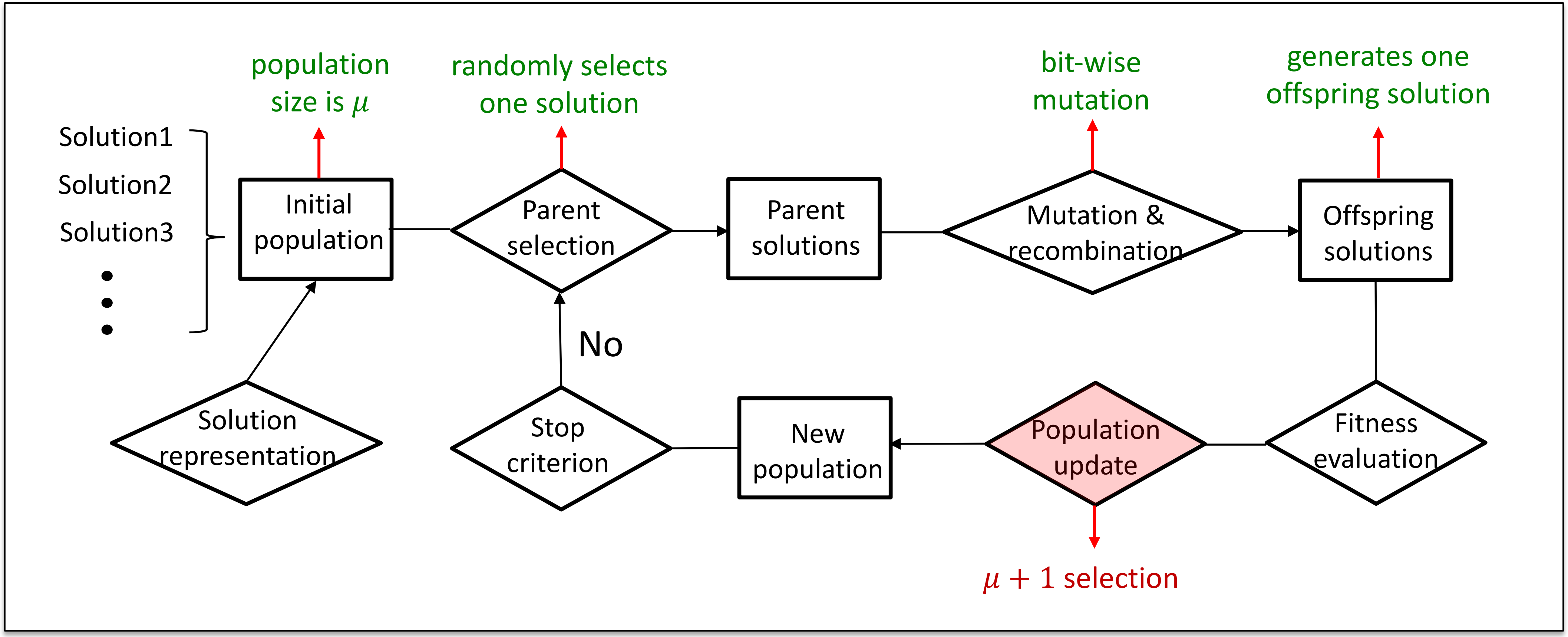
Solutions with larger crowding distance are better

Population Update of NSGA-II:

Use non-dominated sorting and crowding distance sorting to rank the solutions, and delete the worst ones

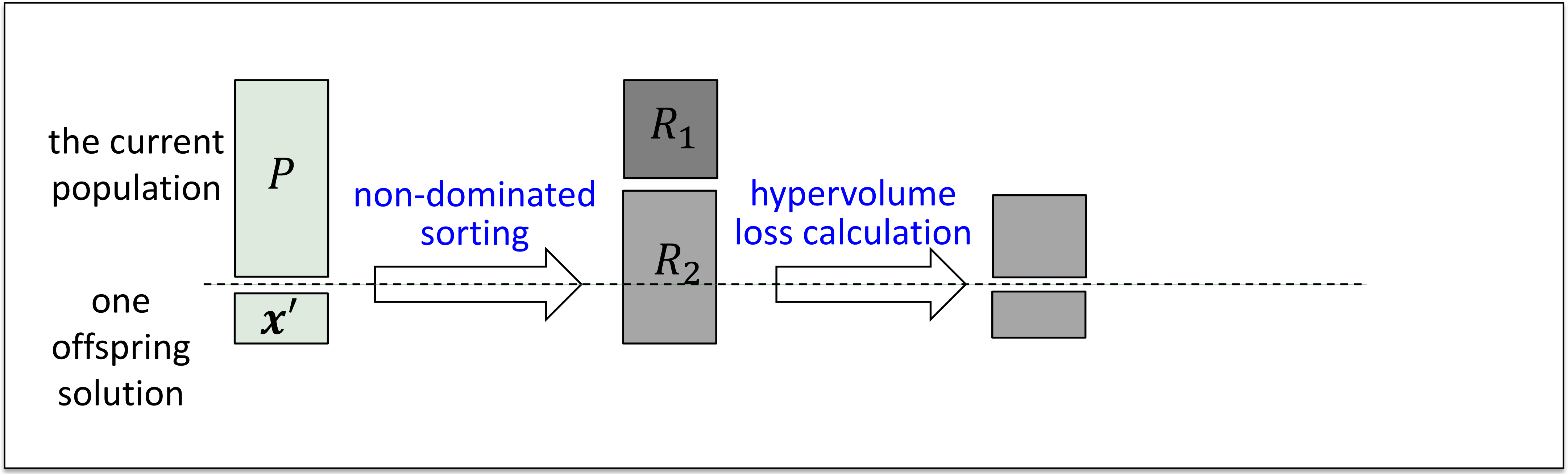


Framework of SMS-EMOA:



Population Update of SMS-EMOA:

Use non-dominated sorting and quality indicators (e.g., hypervolume) to rank the solutions, and delete the worst solution



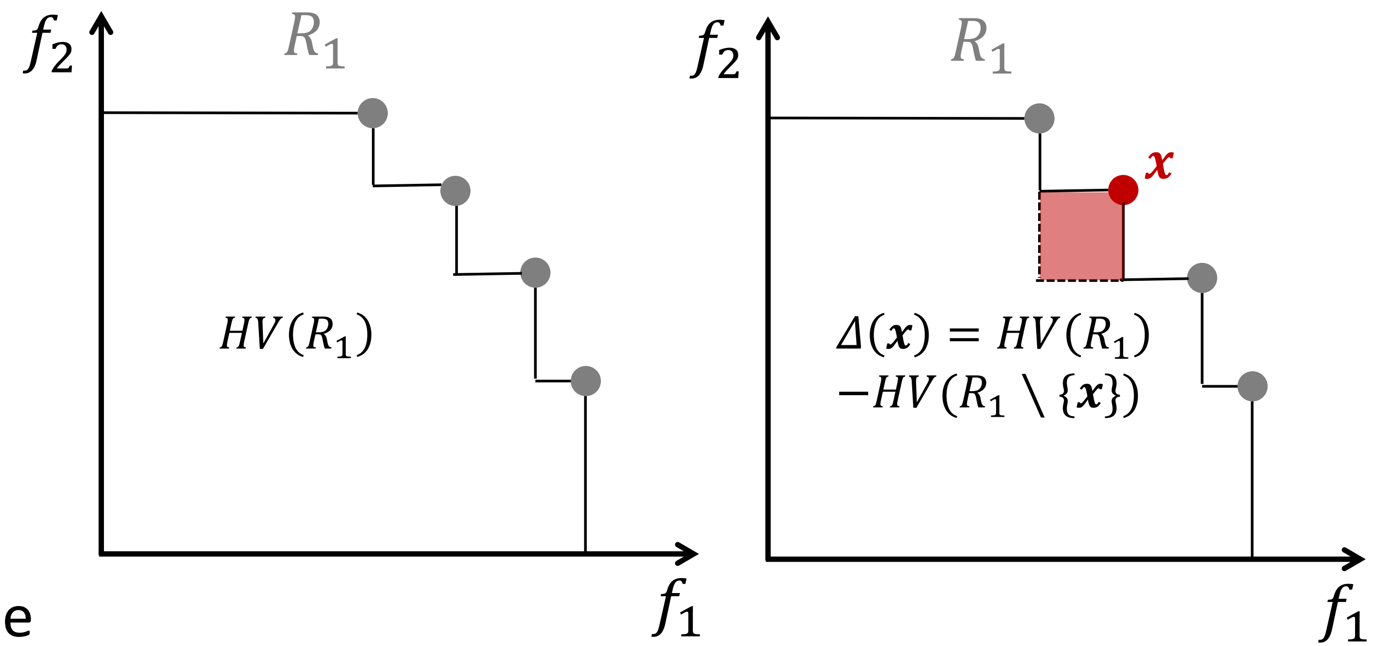
SMS-EMOA

Hypervolume: volume of the space dominated by a set of solutions, reflecting the convergence and diversity of the solutions

Hypervolume loss calculation

Hypervolume loss of $\boldsymbol{x}$:

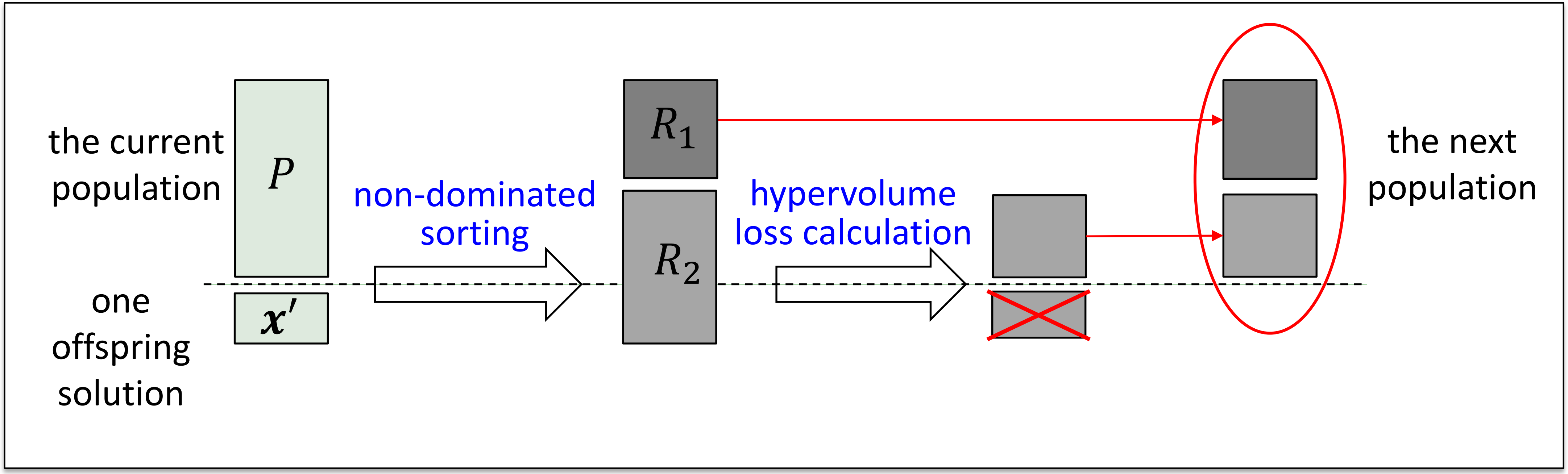
- decreased hypervolume value of the solution set when $\boldsymbol{x}$ is removed



Solutions with larger hypervolume loss are better

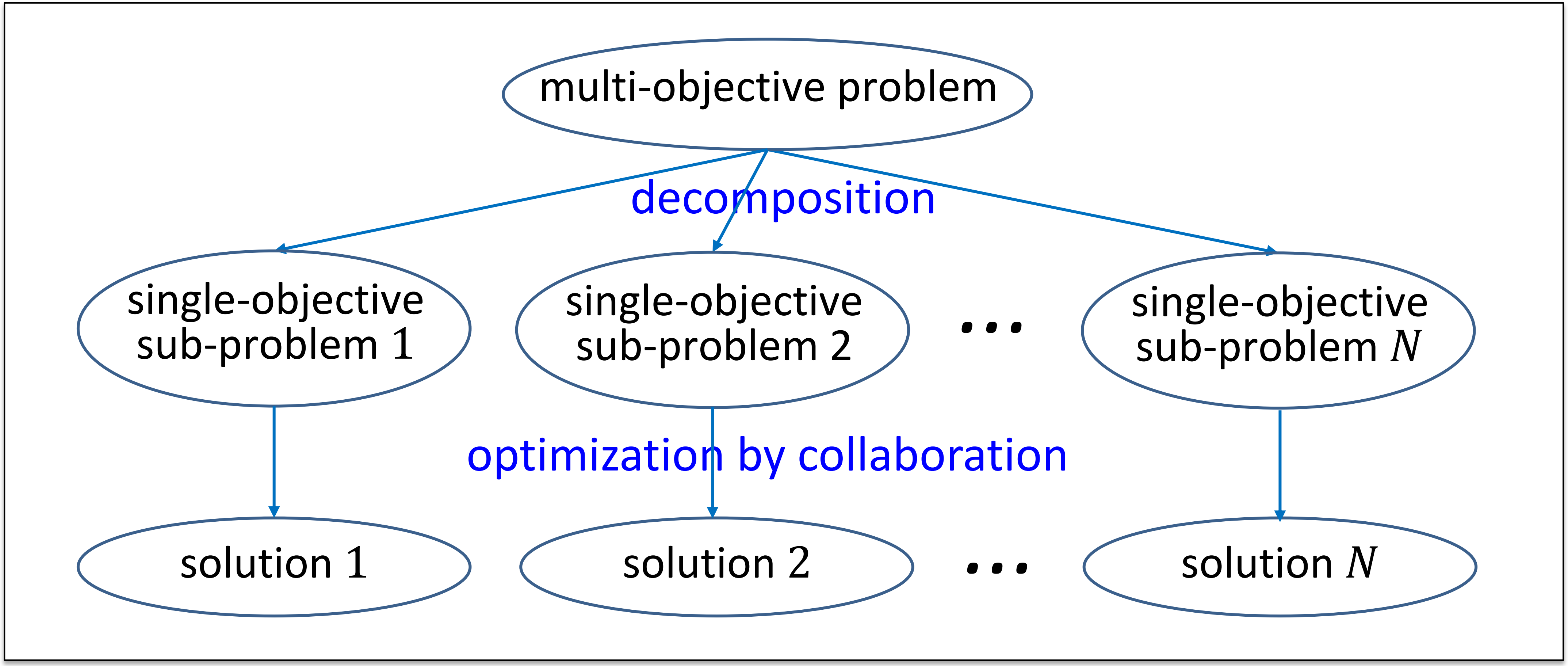
Population Update of SMS-EMOA:

Use non-dominated sorting and quality indicators (e.g., hypervolume) to rank the solutions, and delete the worst solution



Learning And Mining from DatA
http://www.lamda.nju.edu.cn

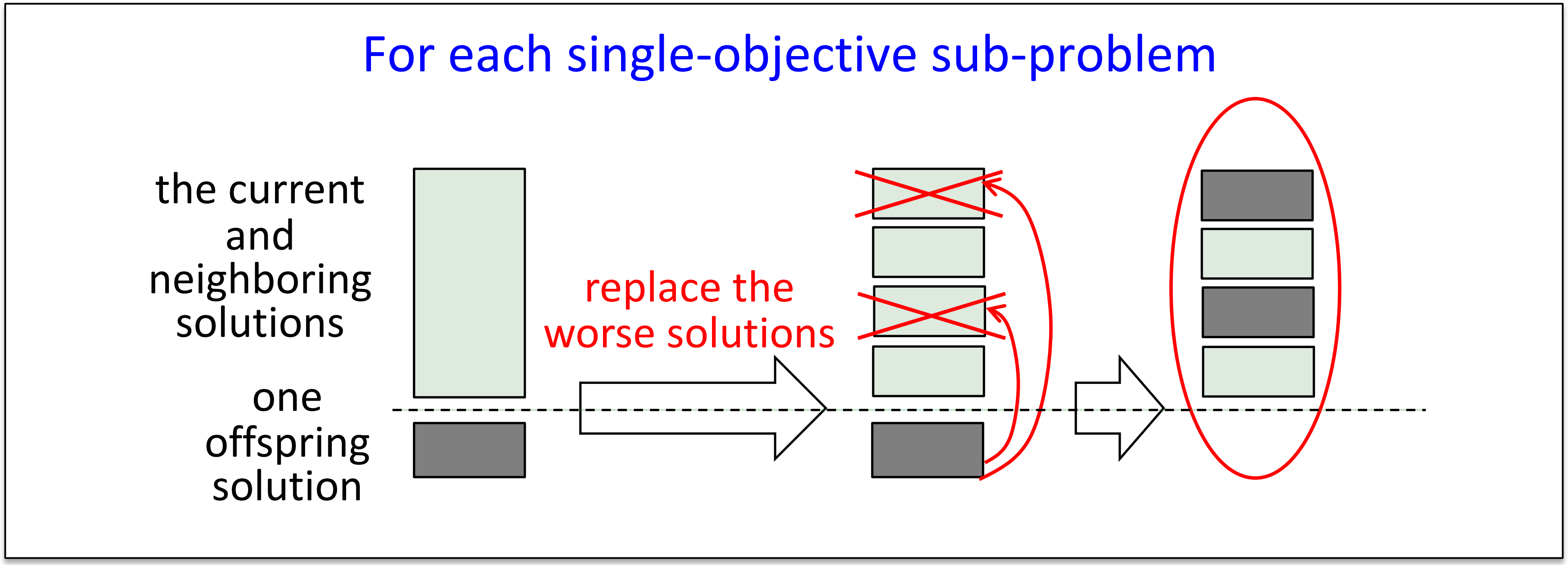
Framework of MOEA/D:



Learning And Mining from DatA
http://www.lamda.nju.edu.cn

Population Update of MOEA/D:

For each single-objective sub-problem, the newly generated solution will replace the worse solutions



Motivation

The prominent feature in population update of MOEAs: greedy and deterministic

- the next-generation population is formed by selecting the best-ranked solutions

K. Deb

*"One common aspect of these **first-generation multi-objective algorithms is that they did not use any elite-preservation operator, thereby compromising the performance** and was also contrary to Rudolph's asymptotic convergence proof which required the preservation of elites from one generation to the next."*

An Interview with Kalyanmoy Deb 2022 ACM Fellow

Is deterministic population update always better?

NO!

Our Main Results

Expected number of generations of SMS-EMOA and NSGA-II for solving the OneJumpZeroJump [Doerr and Zheng, AAAI'21] and bi-objective RealRoyalroad [Dang et al., AAAI'23] problems

	Population update	OneJumpZeroJump	Bi-objective RealRoyalroad
SMS-EMOA	Deterministic	$O(\mu n^k)$ $[\mu \geq n-2k+3]$	$O(\mu n^{n/5-2})$ $[\mu \geq 2n/5+1]$
		$\Omega(n^k)$ $[n-2k=\Omega(n) \wedge \mu = poly(n)]$	$\Omega(n^{n/5-1})$ $[\mu = poly(n)]$
	Stochastic	$O(\mu^2 n^k / 2^{k/4})$ $[\mu \geq 2(n-2k+4)]$	$O(\mu^2 n^{n/5} / 2^{n/20})$ $[\mu \geq 2(2n/5+2)]$
NSGA-II	Deterministic	$\Omega(n^k/\mu)$ $[n-2k=\Omega(n) \wedge \mu = poly(n)]$	$\Omega(n^{n/5-1}/\mu)$ $[\mu = poly(n)]$
	Stochastic	$O(k(n/2)^k)$ $[\mu \geq 8(n-2k+3)]$	$O(n(20e^2)^{n/5})$ $[\mu \geq 8(2n/5+1)]$

Green color: results of our IJCAI'23 work; Yellow color: extended results

Our Main Results

	Population update	OneJumpZeroJump	Bi-objective RealRoyalroad
SMS-EMOA	Deterministic	$O(\mu n^k)$ $[\mu \ge n-2k+3]$	$O(\mu n^{n/5-2})$ $[\mu \ge 2n/5+1]$
		$\Omega(n^k)$ $[n-2k=\Omega(n) \wedge \mu = poly(n)]$	$\Omega(n^{n/5-1})$ $[\mu = poly(n)]$
	Stochastic	$O(\mu^2 n^k/2^{k/4})$ $[\mu \ge 2(n-2k+4)]$	$O(\mu^2 n^{n/5}/2^{n/20})$ $[\mu \ge 2(2n/5+2)]$
NSGA-II	Deterministic	$\Omega(n^k/\mu)$ $[n-2k=\Omega(n) \wedge \mu = poly(n)]$	$\Omega(n^{n/5-1}/\mu)$ $[\mu = poly(n)]$
	Stochastic	$O(k(n/2)^k)$ $[\mu \ge 8(n-2k+3)]$	$O(n(20e^2)^{n/5})$ $[\mu \ge 8(2n/5+1)]$

For example, for SMS-EMOA solving the OneJumpZeroJump problem

Expected running time $\Omega(n^k)$ $\xrightarrow[\text{accelerated by } 2^{k/4}/\mu^2]{\text{Stochastic}}$ $O(\mu^2 n^k/2^{k/4})$

exponential acceleration if
$k = \Omega(n) \wedge k = n/2 - \Omega(n)$
$2(n-2k+4) \le \mu = poly(n)$

OneJumpZeroJump Problem

Definition of OneJumpZeroJump:

$$f_1(\boldsymbol{x}) = \begin{cases} k + |\boldsymbol{x}|_1, & \textit{if } |\boldsymbol{x}|_1 \le n - k \textit{ or } \boldsymbol{x} = 1^n \\ n - |\boldsymbol{x}|_1, & \textit{else} \end{cases}$$

$$f_2(\boldsymbol{x}) = \begin{cases} k + |\boldsymbol{x}|_0, & \textit{if } |\boldsymbol{x}|_0 \le n - k \textit{ or } \boldsymbol{x} = 0^n \\ n - |\boldsymbol{x}|_0, & \textit{else} \end{cases}$$

- Pareto set: $\{\boldsymbol{x} \mid |\boldsymbol{x}|_1 \in [k..n-k] \cup \{0, n\}\}$
- Pareto front: $\{(a, n + 2k - a) \mid a \in [2k..n] \cup \{k, n + k\}\}$

Characterize a class of problems where some adjacent Pareto optimal solutions in the objective space locate far away in the decision space

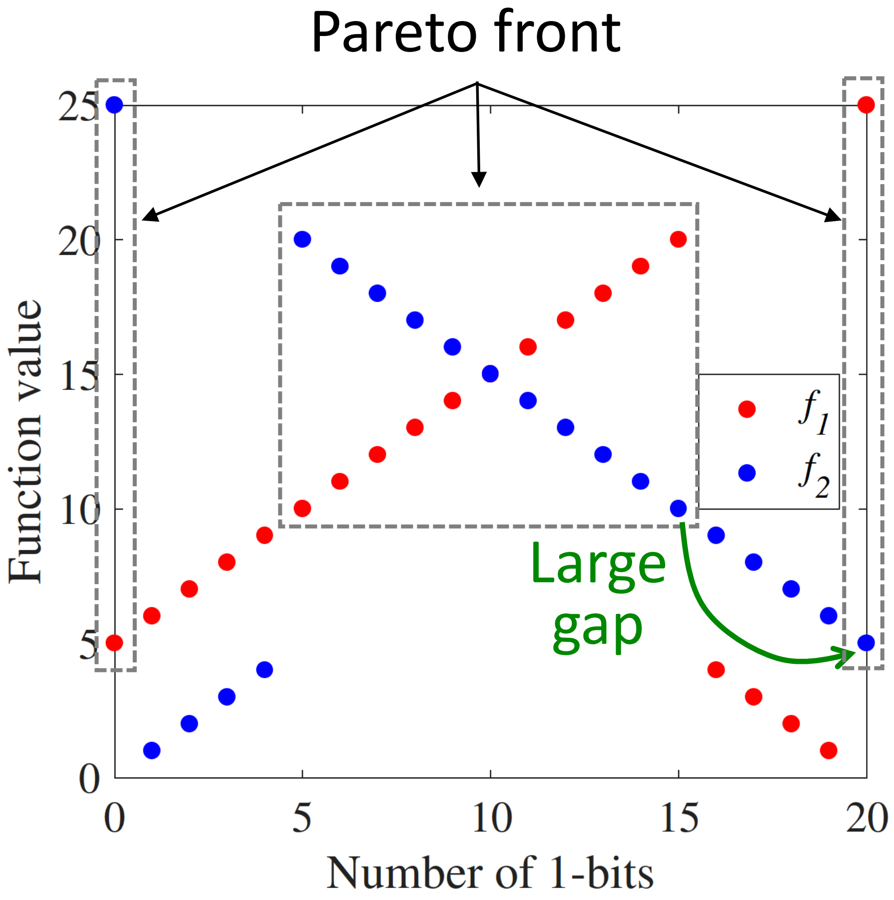
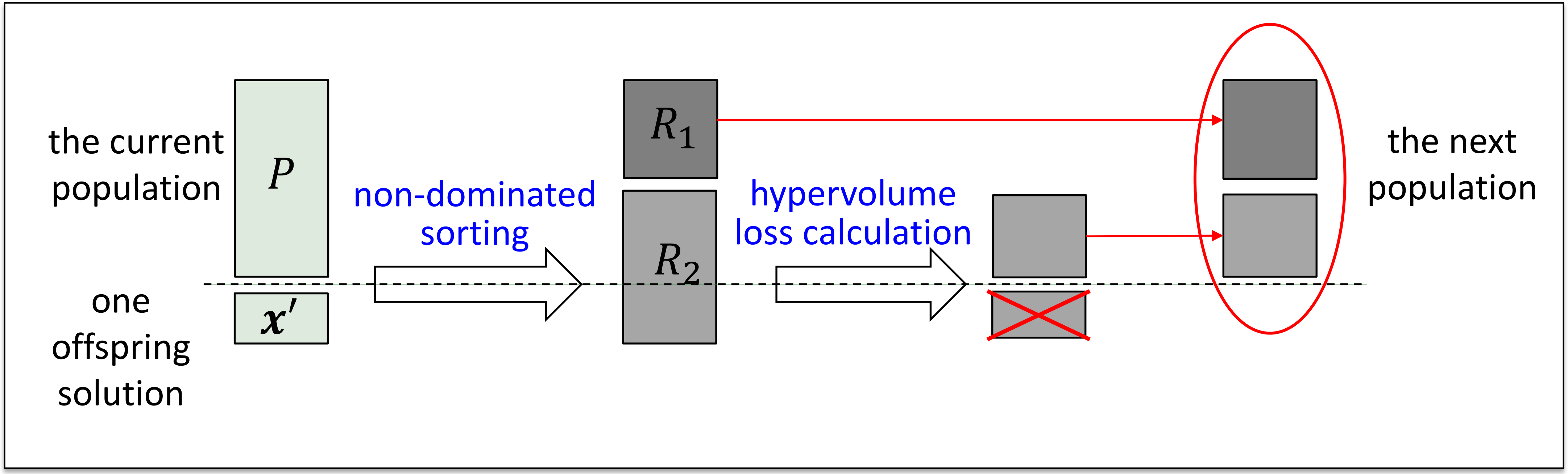


Illustration of function values when $n = 20$ and $k = 5$

Population Update of SMS-EMOA:

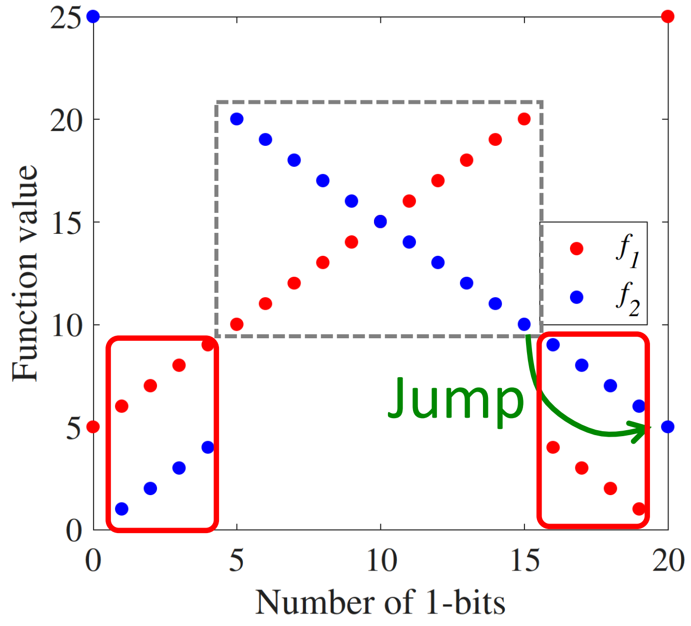
Use non-dominated sorting and quality indicators (e.g., hypervolume) to rank the solutions, and delete the worst solution



Theorem. For SMS-EMOA solving OneJumpZeroJump with $n - 2k = \Theta(n)$, if using a population size μ such that $\mu = poly\ (n)$, then the expected number of generations for finding the Pareto front is $\Omega(n^k)$.

Proof sketch:

- all the solutions in the initial population belong to the inner part of the Pareto front with probability $1 - o(1)$
- the solution with number of 1-bits in $[1, k-1] \cup [n-k+1, n-1]$ cannot be maintained
- the extreme solution 1^n (and 0^n) can only be generated by flipping k bits of a solution simultaneously (whose probability is at most $1/n^k$)



LAMDA
Learning And Mining from DatA
http://www.lamda.nju.edu.cn

SMS-EMOA Using Stochastic Population Update

Procedure of stochastic population update

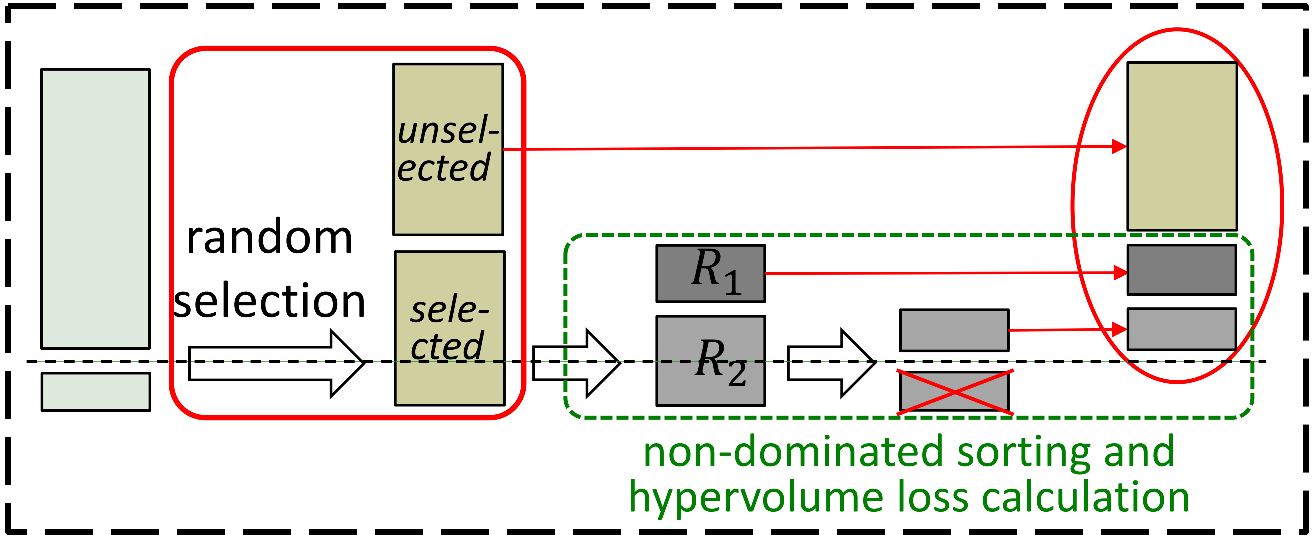
1. $Q' \leftarrow \lfloor |Q|/2 \rfloor$ solutions uniformly and randomly selected from Q without replacement
2. partition Q' into non-dominated sets $R_1, R_2, \ldots, R_v$
3. let $\boldsymbol{z} = \arg\min_{\boldsymbol{x} \in R_v} \Delta_{\boldsymbol{r}}(\boldsymbol{x}, R_v)$
4. return $Q \setminus \{\boldsymbol{z}\}$

Union of the current population and the offspring solution

Preselection: select a subset Q' of Q

Non-dominated sorting for Q'

Hypervolume loss calculation



The difference:
the removed solution is selected from a subset Q' of Q, instead of the entire set Q

Analysis of SMS-EMOA Using Stochastic Population Update

Theorem. For SMS-EMOA solving OneJumpZeroJump, if using stochastic population update, and a population size μ such that $\mu \geq 2(n-2k+4)$, then the expected number of generations for finding the Pareto front is $O(\mu^2 n^k / 2^{k/4})$.

Lemma. For SMS-EMOA solving OneJumpZeroJump, if using stochastic population update, and a population size μ such that $\mu \geq 2(n-2k+4)$, then

- an objective vector $\boldsymbol{f}^*$ in the Pareto front will always be maintained once it has been found
- any solution in $P \cup \{\boldsymbol{x}'\}$ can be maintained in the next population with probability at least $1/2$

The current population

The offspring solution produced in the current generation

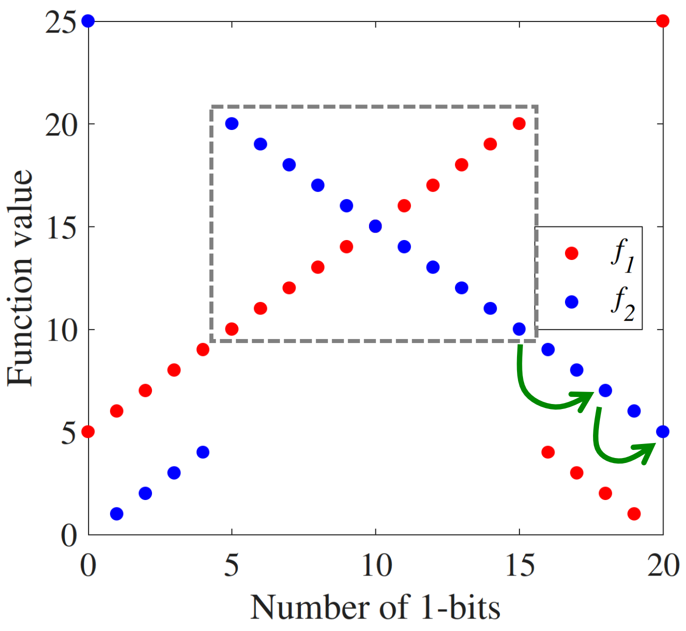
Analysis of SMS-EMOA Using Stochastic Population Update

Theorem. For SMS-EMOA solving OneJumpZeroJump, if using stochastic population update, and a population size μ such that $\mu \geq 2(n-2k+4)$, then the expected number of generations for finding the Pareto front is $O(\mu^2 n^k/2^{k/4})$.

Proof sketch:

- find a solution in the inner part of the Pareto set: $O(\mu k^k)$
- find the whole inner part of the Pareto front: $O(\mu n \log n)$
- the extreme solution 1^n (or 0^n) can be generated by gradually flipping the 0-bits (or 1-bits)

Use additive drift [He and Yao, AIJ'01]



Analysis of SMS-EMOA Using Stochastic Population Update

Theorem. For SMS-EMOA solving OneJumpZeroJump, if using stochastic population update, and a population size μ such that $\mu \geq 2(n-2k+4)$, then the expected number of generations for finding the Pareto front is $O(\mu^2 n^k / 2^{k/4})$.

Proof sketch:

Use additive drift [He and Yao, AIJ'01]: consider the change of $\max_{\boldsymbol{x} \in P} |\boldsymbol{x}|_1$

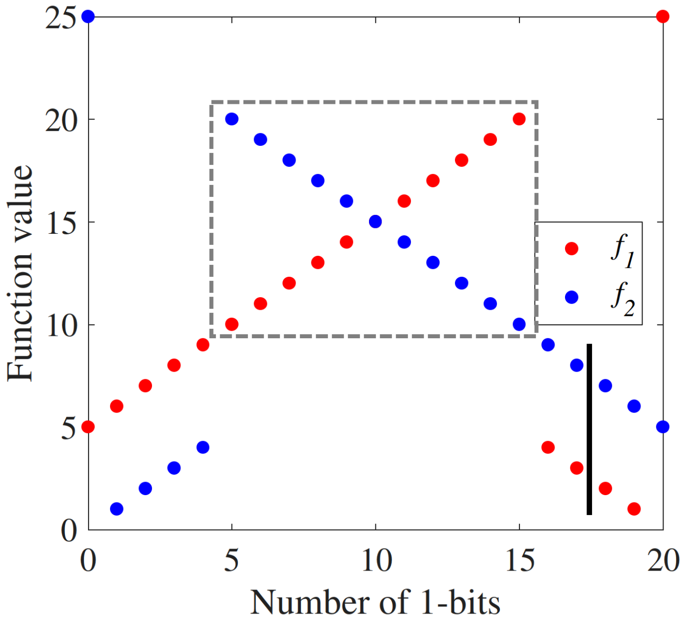
- the distance function is defined as

$$V(P) = \begin{cases} 0 & \text{if } \max_{x \in P} |x|_1 = n, \\ e\mu n^{k/2} & \text{if } n - k/2 \leq \max_{x \in P} |x|_1 \leq n-1, \\ e\mu n^{k/2} + 1 & \text{if } n - k \leq \max_{x \in P} |x|_1 < n - k/2. \end{cases}$$

Target state

The probability of jumping to the target state is small, thus the distance to the target state is set to be a large value

The probability of jumping to the better state is large, thus the distance to the better state is set to be a small value



Analysis of SMS-EMOA Using Stochastic Population Update

Theorem. For SMS-EMOA solving OneJumpZeroJump, if using stochastic population update, and a population size μ such that $\mu \geq 2(n-2k+4)$, then the expected number of generations for finding the Pareto front is $O(\mu^2 n^k / 2^{k/4})$.

Proof sketch:

Use additive drift [He and Yao, AIJ'01]: consider the change of $\max_{\boldsymbol{x}\in P}|\boldsymbol{x}|_1$

if $n - k/2 \leq q \coloneqq \max_{x\in P}|x|_1 \leq n-1$

- expected decrease of V: $(1/(e\mu n^{n-q})) \cdot e\mu n^{k/2} \geq 1$

  Prob. of generating 1^n

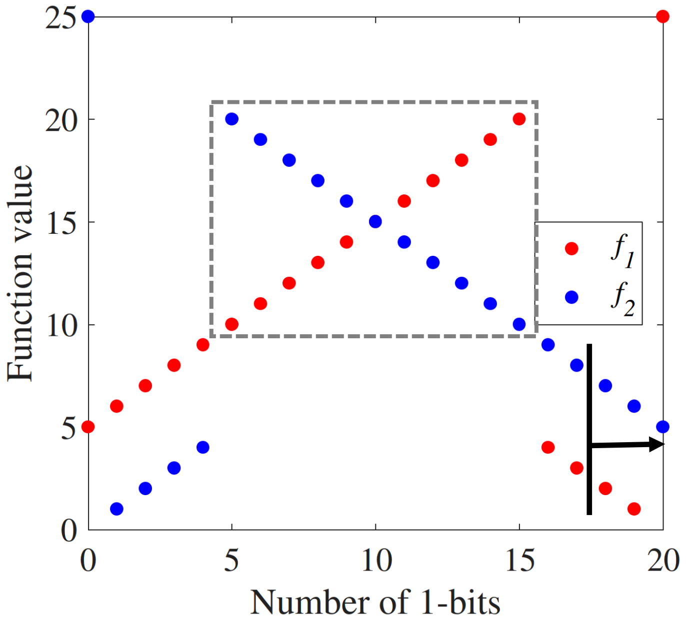
  Change of V

- expected increase of V: $1/2 \cdot 1$

  Change of V

  Prob. of losing the solution with q 1-bits

- expected change of V: $1/2$



Analysis of SMS-EMOA Using Stochastic Population Update

Theorem. For SMS-EMOA solving OneJumpZeroJump, if using stochastic population update, and a population size μ such that $\mu \geq 2(n-2k+4)$, then the expected number of generations for finding the Pareto front is $O(\mu^2 n^k / 2^{k/4})$.

Proof sketch:

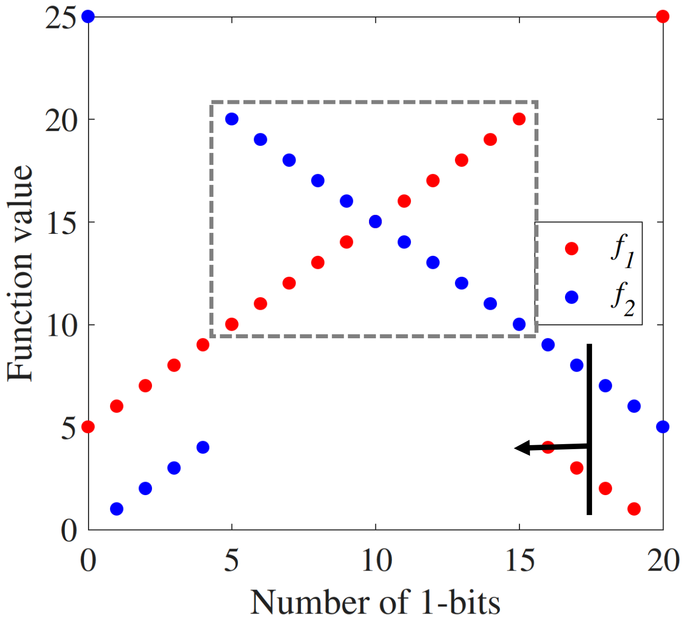
Use additive drift [He and Yao, AIJ'01]: consider the change of $\max_{\boldsymbol{x}\in P} |\boldsymbol{x}|_1$

if $n-k \leq q := \max_{x\in P} |x|_1 < n-k/2$

- expected decrease of V: $\binom{k/2}{k/4} / (2e\mu n^{k/2}) \cdot 1$

  Prob. of decreasing V Change of V

- V cannot increase because the solution with $(n-k)$ 1-bits will always be maintained
- expected change of V: $\binom{k/2}{k/4} / (2e\mu n^{k/2}) \geq 2^{k/4} / (2e\mu n^{k/2})$



Analysis of SMS-EMOA Using Stochastic Population Update

Theorem. For SMS-EMOA solving OneJumpZeroJump, if using stochastic population update, and a population size μ such that $\mu \geq 2(n-2k+4)$, then the expected number of generations for finding the Pareto front is $O(\mu^2 n^k / 2^{k/4})$.

Proof sketch:

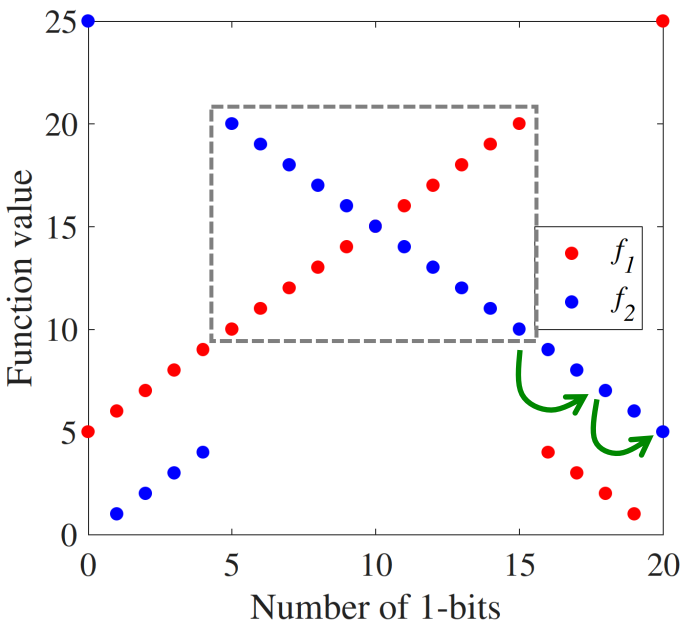
Use additive drift [He and Yao, AIJ'01]: consider the change of $\max_{\boldsymbol{x} \in P} |\boldsymbol{x}|_1$

Combining the analysis of the two cases

- expected change of V: $2^{k/4}/(2e\mu n^{k/2})$
- $V(P) \leq e\mu n^{k/2} + 1$

$\Downarrow$

- expected number of generations for finding 1^n: $O(\mu^2 n^k / 2^{k/4})$



Our Main Results

	Population update	OneJumpZeroJump	Bi-objective RealRoyalroad
SMS-EMOA	Deterministic	$O(\mu n^k)$ $[\mu \geq n-2k+3]$	$O(\mu n^{n/5-2})$ $[\mu \geq 2n/5+1]$
		$\Omega(n^k)$ $[n-2k=\Omega(n) \wedge \mu = poly(n)]$	$\Omega(n^{n/5-1})$ $[\mu = poly(n)]$
	Stochastic	$O(\mu^2 n^k/2^{k/4})$ $[\mu \geq 2(n-2k+4)]$	$O(\mu^2 n^{n/5}/2^{n/20})$ $[\mu \geq 2(2n/5+2)]$
NSGA-II	Deterministic	$\Omega(n^k/\mu)$ $[n-2k=\Omega(n) \wedge \mu = poly(n)]$	$\Omega(n^{n/5-1}/\mu)$ $[\mu = poly(n)]$
	Stochastic	$O(k(n/2)^k)$ $[\mu \geq 8(n-2k+3)]$	$O(n(20e^2)^{n/5})$ $[\mu \geq 8(2n/5+1)]$

For SMS-EMOA solving the OneJumpZeroJump problem

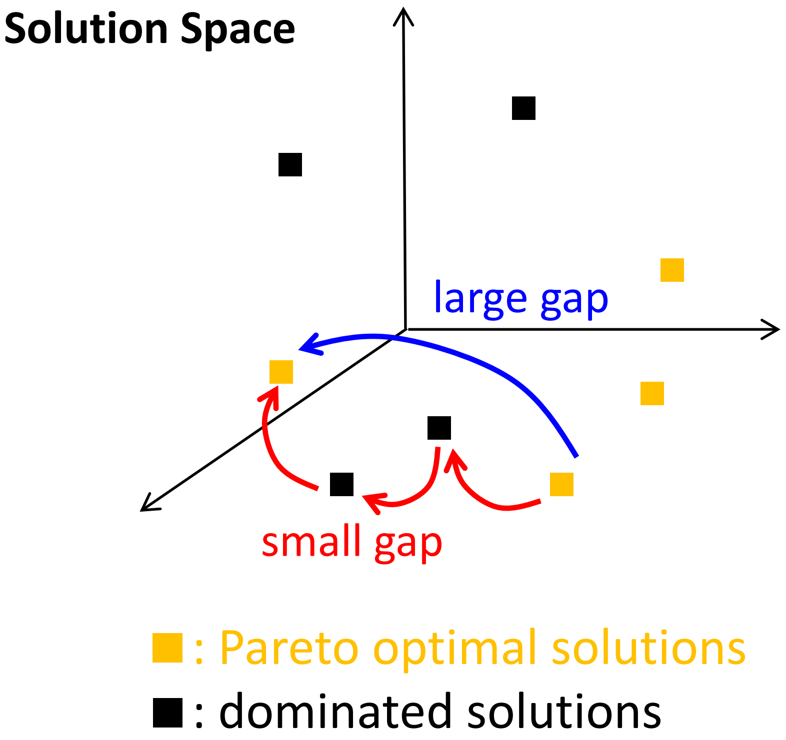
Expected running time $\Omega(n^k)$ $\xrightarrow[\text{accelerated by } 2^{k/4}/\mu^2]{\text{Stochastic}}$ $O(\mu^2 n^k/2^{k/4})$

exponential acceleration if
$k = \Omega(n) \wedge k = n/2 - \Omega(n)$
$2(n-2k+4) \leq \mu = poly(n)$

Intuitive Explanation

By introducing randomness into population update, MOEAs can go across inferior regions between different Pareto optimal solutions more easily

- **Deterministic**
  - prefers non-dominated solutions
  - if objective vectors in the Pareto front are far away in the solution space, easy to get trapped
- **Stochastic**
  - allows dominated solutions to participate in the evolutionary process
  - follows an easier path in the solution space to find adjacent points in the Pareto front



Our Main Results

	Population update	OneJumpZeroJump	Bi-objective RealRoyalroad
SMS-EMOA	Deterministic	$O(\mu n^k)$ $[\mu \geq n - 2k + 3]$	$O(\mu n^{n/5-2})$ $[\mu \geq 2n/5 + 1]$
		$\Omega(n^k)$ $[n - 2k = \Omega(n) \wedge \mu = poly(n)]$	$\Omega(n^{n/5-1})$ $[\mu = poly(n)]$
	Stochastic	$O(\mu^2 n^k / 2^{k/4})$ $[\mu \geq 2(n - 2k + 4)]$	$O(\mu^2 n^{n/5} / 2^{n/20})$ $[\mu \geq 2(2n/5 + 2)]$
NSGA-II	Deterministic	$\Omega(n^k/\mu)$ $[n - 2k = \Omega(n) \wedge \mu = poly(n)]$	$\Omega(n^{n/5-1}/\mu)$ $[\mu = poly(n)]$
	Stochastic	$O(k(n/2)^k)$ $[\mu \geq 8(n - 2k + 3)]$	$O(n(20e^2)^{n/5})$ $[\mu \geq 8(2n/5 + 1)]$

For NSGA-II solving the bi-objective RealRoyalroad problem

Expected running time $\Omega(n^{n/5-1}/\mu)$ $\xrightarrow{\text{Stochastic}}$ $O(n(20e^2)^{n/5})$

Stochastic

accelerated by $(n/(20e^2))^{n/5}/(\mu n^2)$

Bi-objective RealRoyalroad Problem

Definition of bi-objective RealRoyalroad:

#trailing 0-bits #leading 0-bits

$$\boldsymbol{f}(\boldsymbol{x}) = \begin{cases} (n|\boldsymbol{x}|_1 + \mathrm{TZ}(\boldsymbol{x}), n|\boldsymbol{x}|_1 + \mathrm{LZ}(\boldsymbol{x})), & \text{if } \boldsymbol{x} \in H \cup G; \\ (0,0), & \text{else,} \end{cases}$$

where $H = \{\boldsymbol{x} \mid |\boldsymbol{x}|_1 = 4n/5 \wedge \mathrm{TZ}(\boldsymbol{x}) + \mathrm{LZ}(\boldsymbol{x}) = n/5\}$
and $G = \{\boldsymbol{x} \mid |\boldsymbol{x}|_1 \leq 3n/5\}$ e.g., H ={11110, 01111} for $n = 5$

- Pareto set: H
- Pareto front: $\{(4n^2/5 + a, 4n^2/5 + n/5 - a) \mid a \in [0..n/5]\}$

Characterize a class of problems where a large gap exists between the Pareto front and the second front in the decision space

$\{\boldsymbol{x} \mid |\boldsymbol{x}|_1 = 3n/5 \wedge \mathrm{TZ}(\boldsymbol{x}) + \mathrm{LZ}(\boldsymbol{x}) = 2n/5\}$

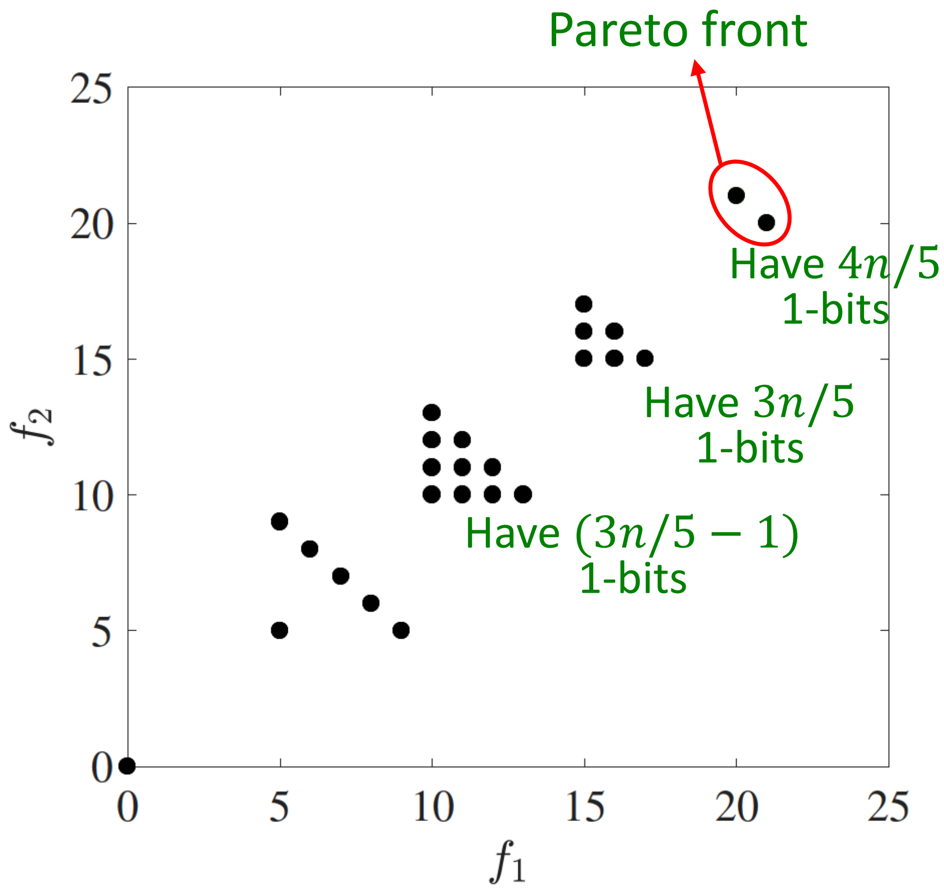
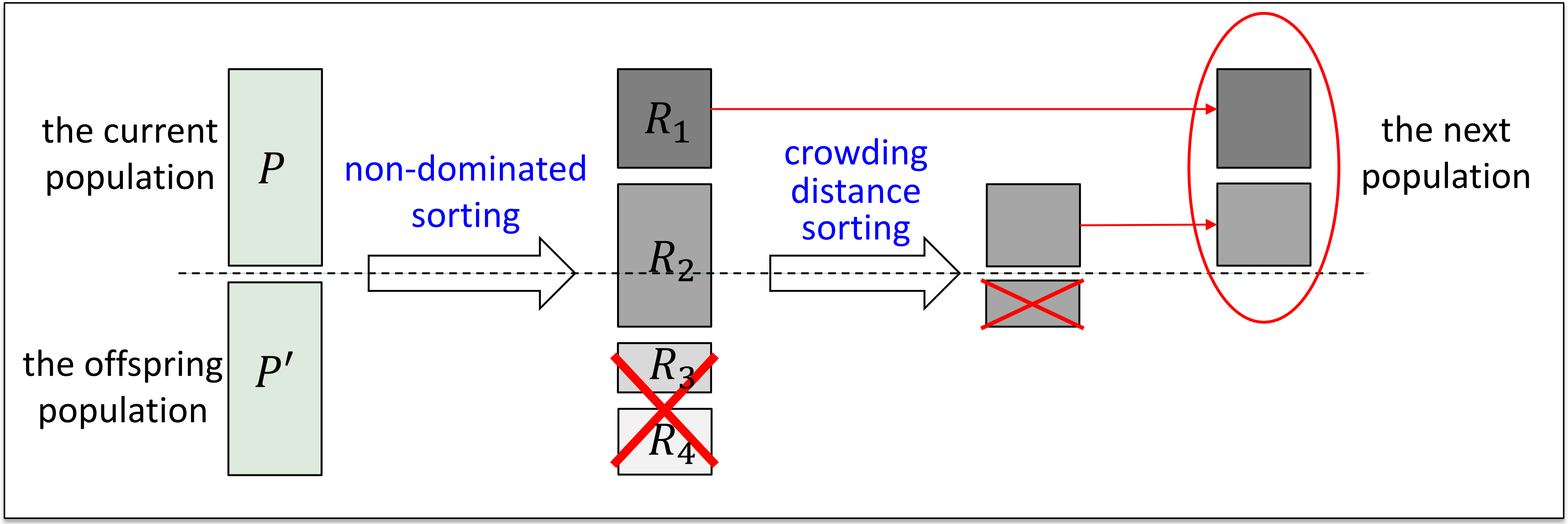


Illustration of function values when $n = 5$

Population Update of NSGA-II:

Use non-dominated sorting and crowding distance sorting to rank the solutions, and delete the worst ones

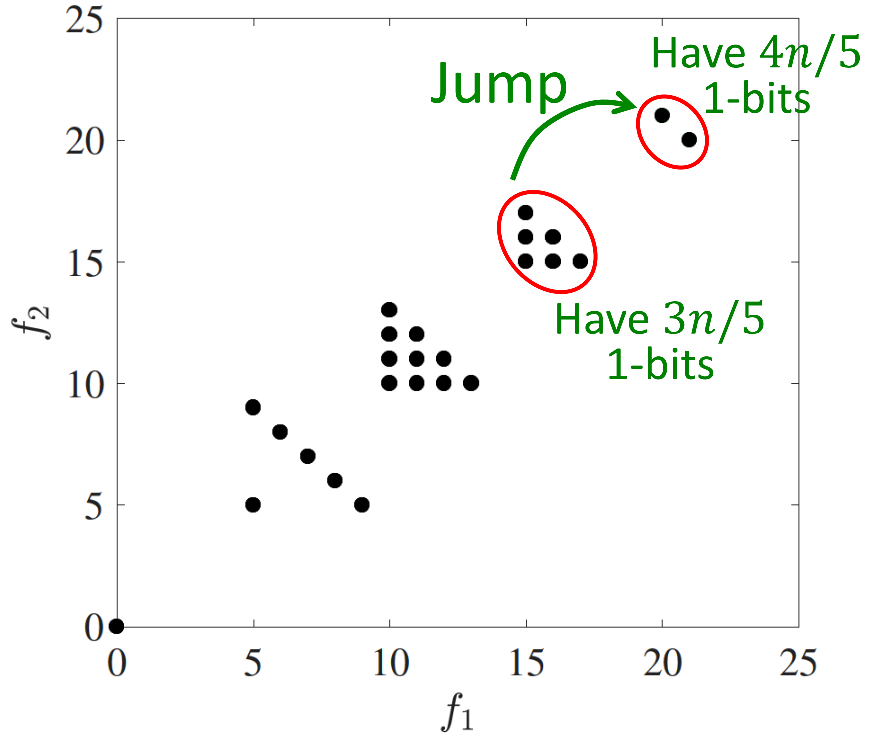


Analysis of NSGA-II

Theorem. For NSGA-II solving bi-objective RealRoyalroad, if using a population size μ such that $\mu = poly\ (n)$, then the expected number of generations for finding the Pareto front is $\Omega\left(n^{n/5-1}/\mu\right)$.

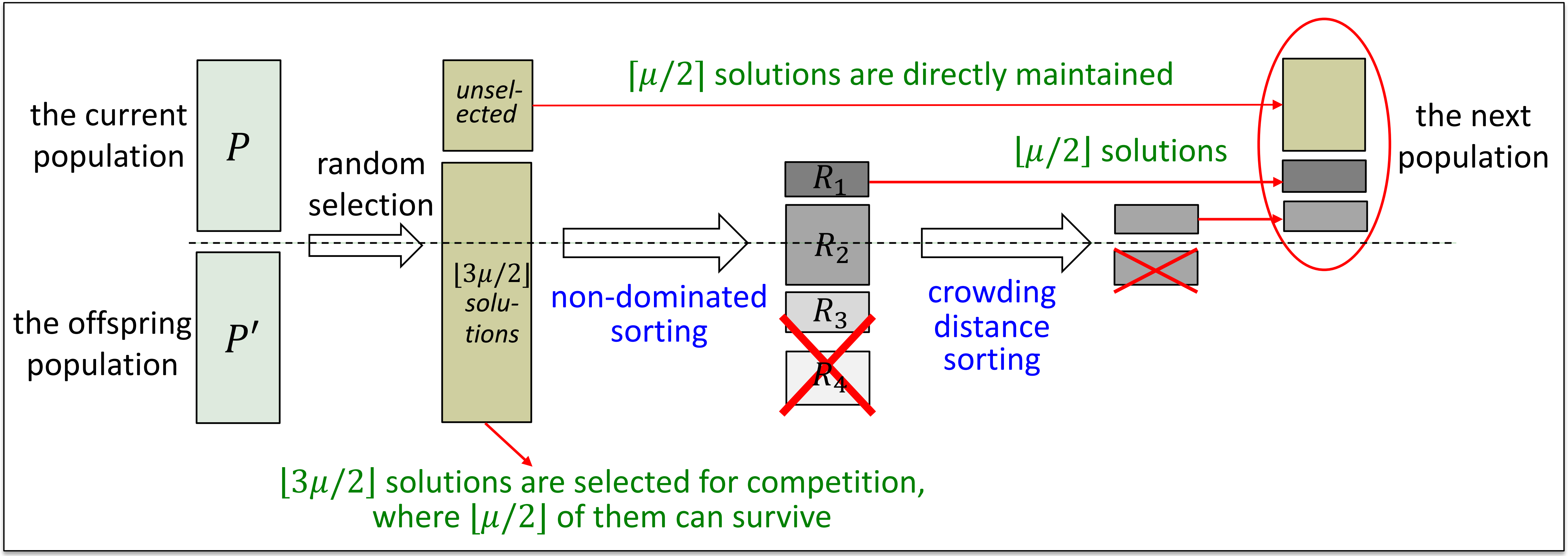
Proof sketch:

- all the solutions in the initial population have at most $3n/5$ 1-bits with probability $1 - o(1)$
- the solution with more than $3n/5$ 1-bits (not in H) has the objective vector (0,0), and cannot be maintained
- the solutions in H can only be generated by flipping at least $n/5$ bits of a solution simultaneously



NSGA-II Using Stochastic Population Update

The removed solutions are selected from a random subset of $P \cup P'$, instead of the entire set



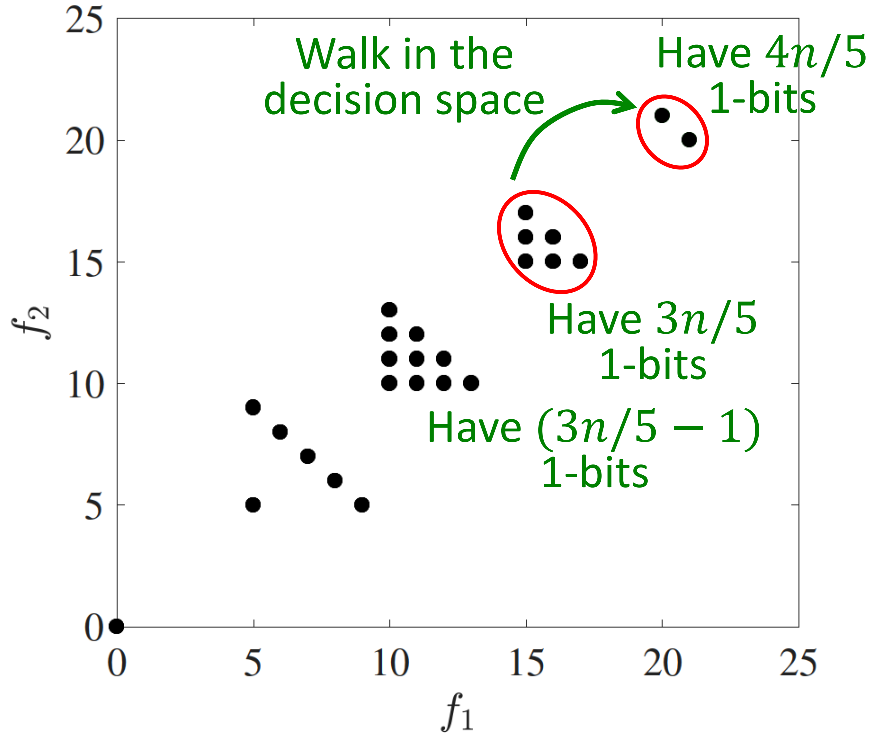
Analysis of NSGA-II Using Stochastic Population Update

Theorem. For NSGA-II solving bi-objective RealRoyalroad, if using stochastic population update and a population size μ such that $\mu \geq 8(2n/5+1)$, then the expected number of generations for finding the Pareto front is $O\left(n(20e^2)^{n/5}\right)$.

Proof sketch:

- a solution with $3n/5$ 1-bits can be found in $O(n\log n)$
- a solution in $G' = \{0^i 1^{3n/5} 0^{2n/5-i} \mid 0 \leq i \leq 2n/5\}$ can be found in $O(n^3)$
- all the solutions in G'can be found in $O(n^3)$
- $\boxed{\text{a Pareto optimal solution can be found in } O\left(n(20e^2)^{n/5}\right)}$
- all the Pareto optimal solutions can be found in $O(n^3)$

Use "lucky way" argument [Doerr, TCS'21]



Analysis of NSGA-II Using Stochastic Population Update

Theorem. For NSGA-II solving bi-objective RealRoyalroad, if using stochastic population update and a population size μ such that $\mu \geq 8(2n/5+1)$, then the expected number of generations for finding the Pareto front is $O\big(n(20e^2)^{n/5}\big)$.

Proof sketch:

Use "lucky way" argument [Doerr, TCS'21]

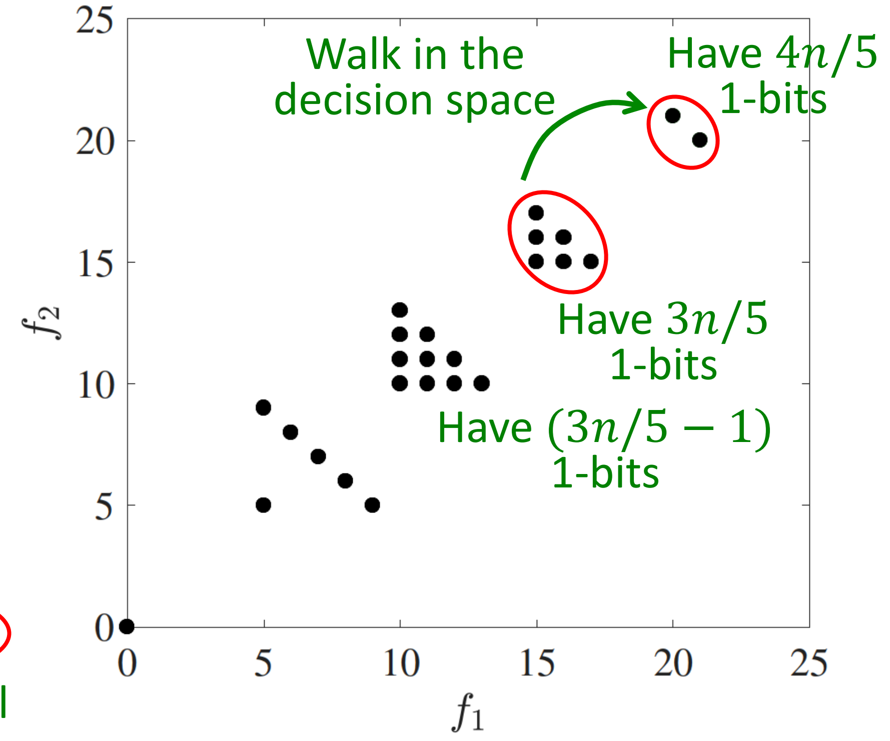
- consider a phase of consecutive $n/5$ generations

$1^{3n/5}0^{2n/5}$ $\xrightarrow[(n/5)/(en)\cdot(1/4)]{\text{with prob. at least}}$ find a solution $\boldsymbol{x}$ with $|\boldsymbol{x}|_1 = 3n/5+1$ and $\mathrm{TZ}(\boldsymbol{x}) \geq n/5$

any solution can survive with prob. at least $1/4$

$\xrightarrow[(n/5-1)/(en)\cdot(1/4)]{\text{with prob. at least}}$ find a solution $\boldsymbol{x}$ with $|\boldsymbol{x}|_1 = 3n/5+2$ and $\mathrm{TZ}(\boldsymbol{x}) \geq n/5$ $\cdots \longrightarrow$ $1^{4n/5}0^{n/5}$

Pareto optimal



Analysis of NSGA-II Using Stochastic Population Update

LAMDA Learning And Mining from DatA http://www.lamda.nju.edu.cn

Theorem. For NSGA-II solving bi-objective RealRoyalroad, if using stochastic population update and a population size μ such that $\mu \geq 8(2n/5+1)$, then the expected number of generations for finding the Pareto front is $O\left(n(20e^2)^{n/5}\right)$.

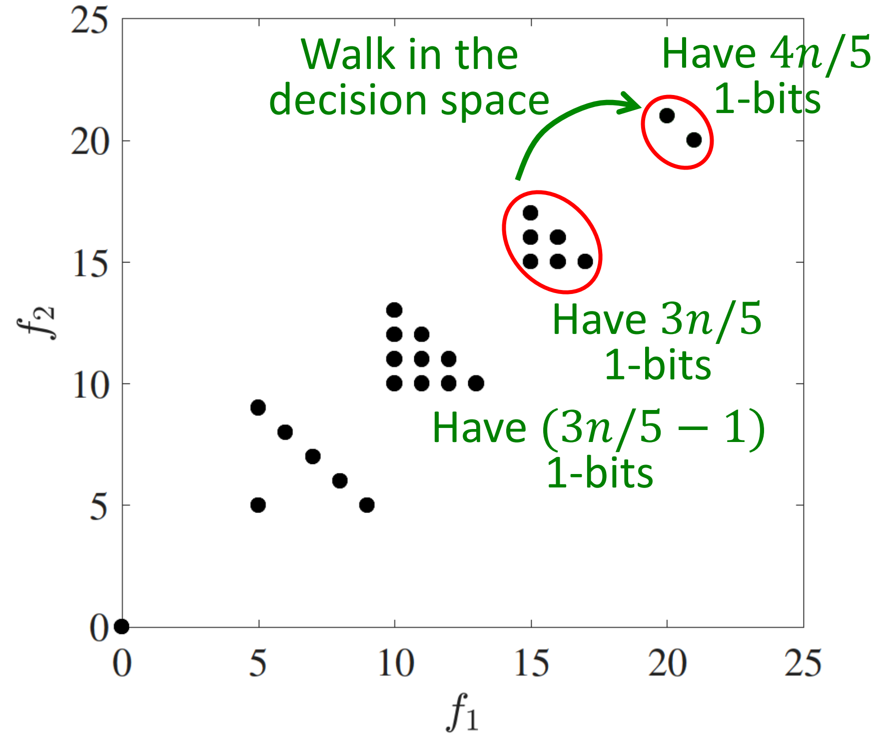
Proof sketch:

Use "lucky way" argument [Doerr, TCS'21]

- consider a phase of consecutive $n/5$ generations

$$1^{3n/5}0^{2n/5} \longrightarrow \cdots \longrightarrow 1^{4n/5}0^{n/5}$$ Pareto optimal

- the probability of the above event is at least $\prod_{i=1}^{n/5}(n/5-i+1)/(4en) \geq 2/(20e^2)^{n/5}$
- the expected number of generations of finding a Pareto optimal solution: $(20e^2)^{n/5}/2 \cdot (n/5)$



Our Main Results

	Population update	OneJumpZeroJump	Bi-objective RealRoyalroad
SMS-EMOA	Deterministic	$O(\mu n^k)$ $[\mu \geq n-2k+3]$	$O(\mu n^{n/5-2})$ $[\mu \geq 2n/5+1]$
		$\Omega(n^k)$ $[n-2k=\Omega(n) \wedge \mu = poly(n)]$	$\Omega(n^{n/5-1})$ $[\mu = poly(n)]$
	Stochastic	$O(\mu^2 n^k/2^{k/4})$ $[\mu \geq 2(n-2k+4)]$	$O(\mu^2 n^{n/5}/2^{n/20})$ $[\mu \geq 2(2n/5+2)]$
NSGA-II	Deterministic	$\Omega(n^k/\mu)$ $[n-2k=\Omega(n) \wedge \mu = poly(n)]$	$\Omega(n^{n/5-1}/\mu)$ $[\mu = poly(n)]$
	Stochastic	$O(k(n/2)^k)$ $[\mu \geq 8(n-2k+3)]$	$O(n(20e^2)^{n/5})$ $[\mu \geq 8(2n/5+1)]$

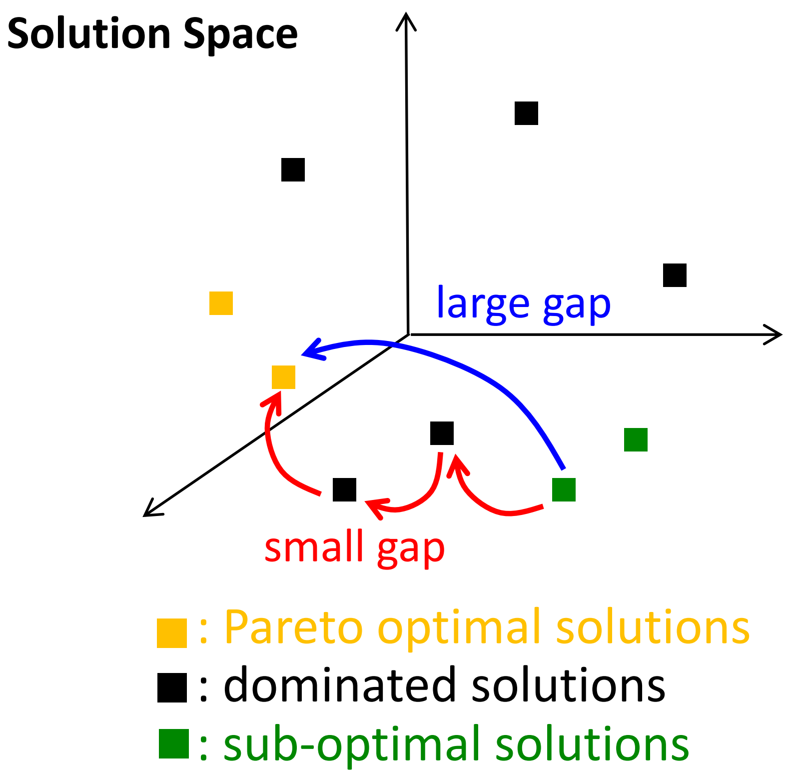
For NSGA-II solving the bi-objective RealRoyalroad problem

Expected running time $\Omega(n^{n/5-1}/\mu)$ $\xrightarrow[\text{accelerated by } (n/(20e^2))^{n/5}/(\mu n^2)]{\text{Stochastic}}$ $O(n(20e^2)^{n/5})$

Intuitive Explanation

By introducing randomness into population update, MOEAs can go across inferior regions between Pareto optimal solutions and sub-optimal solutions

- **Deterministic**
  - prefers non-dominated solutions
  - if Pareto optimal solutions are far away from sub-optimal solutions in the solution space, easy to get trapped
- **Stochastic**
  - allows dominated solutions to participate in the evolutionary process
  - follows an easier path in the solution space to find Pareto optimal solutions from sub-optimal solutions



Our Main Results

Expected number of generations of SMS-EMOA and NSGA-II for solving the OneJumpZeroJump [Doerr and Zheng, AAAI'21] and bi-objective RealRoyalroad [Dang et al., AAAI'23] problems

	Population update	OneJumpZeroJump	Bi-objective RealRoyalroad
SMS-EMOA	Deterministic	$O(\mu n^k)$ $[\mu \geq n - 2k + 3]$	$O(\mu n^{n/5-2})$ $[\mu \geq 2n/5 + 1]$
		$\Omega(n^k)$ $[n - 2k = \Omega(n) \wedge \mu = poly(n)]$	$\Omega(n^{n/5-1})$ $[\mu = poly(n)]$
	Stochastic	$O(\mu^2 n^k / 2^{k/4})$ $[\mu \geq 2(n - 2k + 4)]$	$O(\mu^2 n^{n/5} / 2^{n/20})$ $[\mu \geq 2(2n/5 + 2)]$
NSGA-II	Deterministic	$\Omega(n^k/\mu)$ $[n - 2k = \Omega(n) \wedge \mu = poly(n)]$	$\Omega(n^{n/5-1}/\mu)$ $[\mu = poly(n)]$
	Stochastic	$O(k(n/2)^k)$ $[\mu \geq 8(n - 2k + 3)]$	$O(n(20e^2)^{n/5})$ $[\mu \geq 8(2n/5 + 1)]$

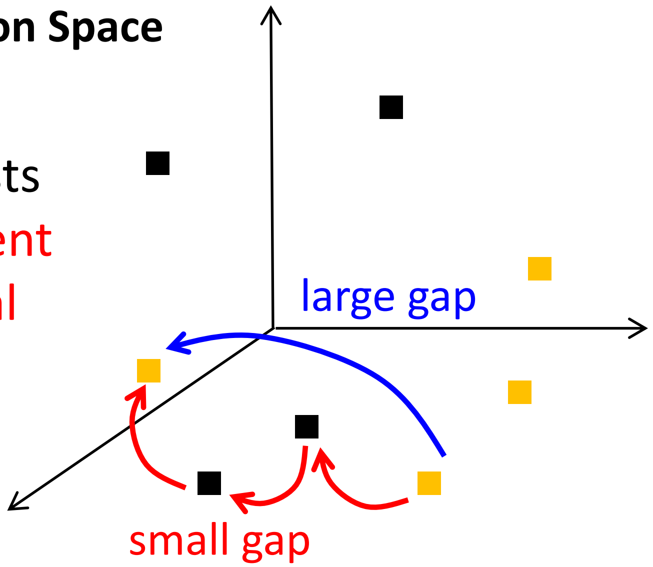
Green color: results of our IJCAI'23 work; Yellow color: extended results

Intuitive Explanation

By introducing randomness into population update, MOEAs can go across inferior regions around the Pareto optimal solutions more easily, thus facilitating to find the whole Pareto front

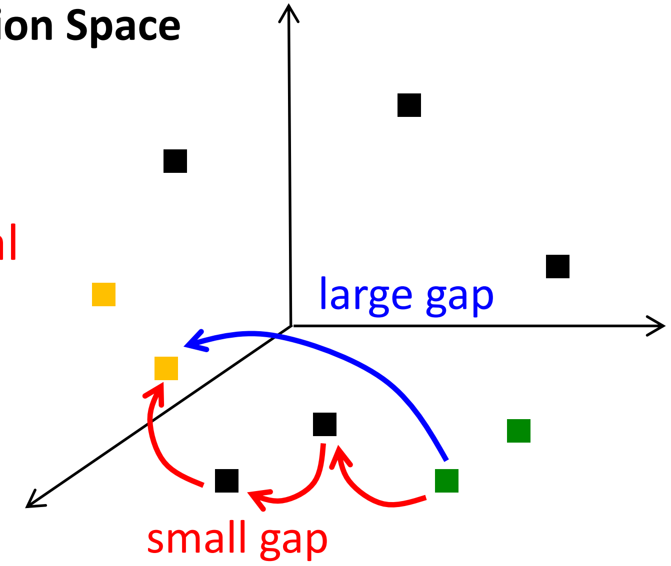
Solution Space

a large gap exists between different Pareto optimal solutions



Solution Space

a large gap exists between Pareto optimal solutions and sub-optimal solutions



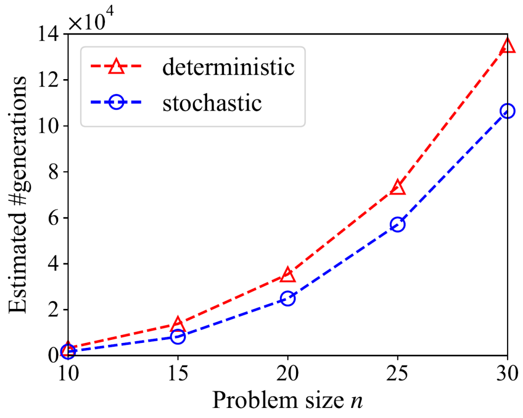
■ : Pareto optimal solutions ■ : dominated solutions ■ : sub-optimal solutions

Experiments

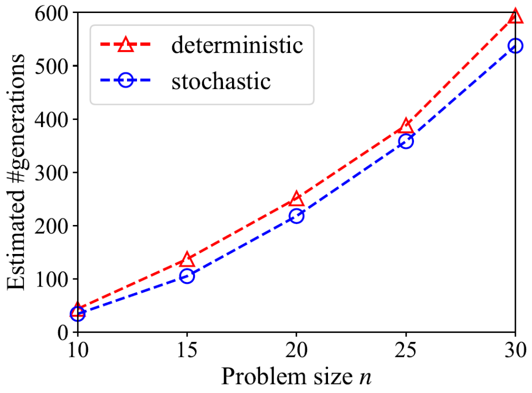
Estimated number of generations (average of 1000 independent runs) of SMS-EMOA and NSGA-II for solving the OneJumpZeroJump and bi-objective RealRoyalroad problems

OneJumpZeroJump with $k = 2$

SMS-EMOA



NSGA-II



Bi-objective RealRoyalroad

		$n = 5$	$n = 10$	$n = 15$	$n = 20$	$n = 25$
SMS-EMOA	Deterministic	43	704	6572	202558	10792477
	Stochastic	46	702	5747	144222	5797043
NSGA-II	Deterministic	2	27	143	1858	73001
	Stochastic	2	25	121	724	10757

Stochastic population update can bring a clear acceleration

Summary

- Prove that stochastic population update can significantly decrease the expected running time

	Population update	OneJumpZeroJump [Doerr and Zheng, AAAI'21]	Bi-objective RealRoyalroad [Dang et al., AAAI'23]
SMS-EMOA	Deterministic	$O(\mu n^k)$ $[\mu \geq n - 2k + 3]$	$O(\mu n^{n/5-2})$ $[\mu \geq 2n/5 + 1]$
		$\Omega(n^k)$ $[n - 2k = \Omega(n) \wedge \mu = poly(n)]$	$\Omega(n^{n/5-1})$ $[\mu = poly(n)]$
	Stochastic	$O(\mu^2 n^k / 2^{k/4})$ $[\mu \geq 2(n - 2k + 4)]$	$O(\mu^2 n^{n/5} / 2^{n/20})$ $[\mu \geq 2(2n/5 + 2)]$
NSGA-II	Deterministic	$\Omega(n^k / \mu)$ $[n - 2k = \Omega(n) \wedge \mu = poly(n)]$	$\Omega(n^{n/5-1} / \mu)$ $[\mu = poly(n)]$
	Stochastic	$O(k(n/2)^k)$ $[\mu \geq 8(n - 2k + 3)]$	$O(n(20e^2)^{n/5})$ $[\mu \geq 8(2n/5 + 1)]$

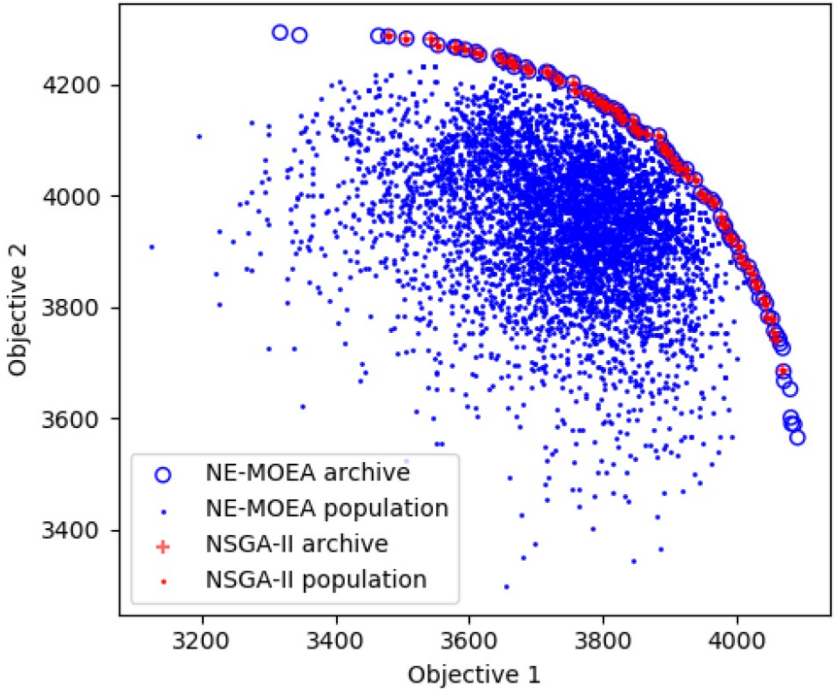
Popular MOEAs

- Challenge the common practice of MOEAs, i.e., greedy and deterministic population update
- Encourage the exploration of developing new MOEAs in the area

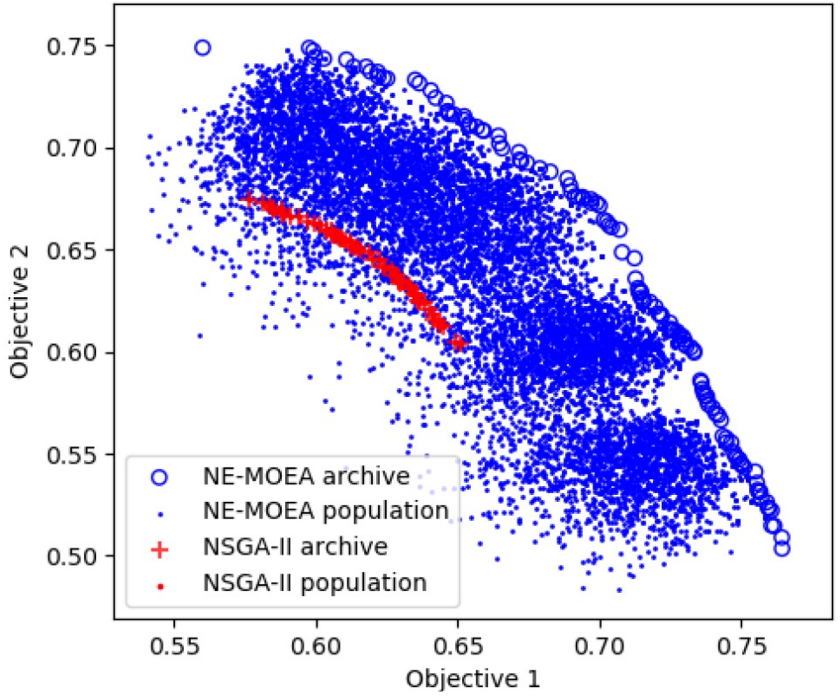
Similar observation

The recent empirical study [Liang, Li and Lehre, GECCO'23] shows that non-elitist population update (the generated offspring solutions form the next population directly) can be helpful

NSGA-II vs. Non-elitist MOEA (NE-MOEA)



On knapsack with 100 items



On NK-Landscape with $n = 200$ and $k = 10$

Similar observation

A subsequent theoretical study [Zheng and Doerr, AAAI'24] considers SMS-EMOA solving the many-objective mOneJumpZeroJump problem, and confirms the same advantage of using stochastic population update

A direct extension of OneJumpZeroJump, which has m objectives

Expected number of generations for $\mu = \Theta(M)$:

Size of the Pareto front, i.e., $(2n/m - 2k + 3)^{m/2}$

deterministic population update
$O(M^2 n^k)$

accelerated by $2^{k/2}/(Mk^{1/2})$

stochastic population update
$O((Mk^{1/2}/2^{k/2}) \cdot M^2 n^k)$

exponential if m is not too large (e.g., constant), and k is large (e.g., $\Theta(n)$)

Finally, I will give a brief review of running time analysis of MOEAs

GSEMO Solving Multi-objective Synthetic Problems

Learning And Mining from DatA
http://www.lamda.nju.edu.cn

GSEMO: a simple MOEA which employs bit-wise mutation to generate an offspring solution in each iteration and keeps the non-dominated solutions in the population

SEMO: a counterpart of GSEMO which employs one-bit mutation

Summary of GSEMO and SEMO solving multi-objective synthetic problems:

	GSEMO	SEMO
LeadingOnesTrailingZeroes	$O(n^3)$ [Giel, CEC'03] $\Omega(n^2/p)$ [Doerr et al., CEC'13]	$\Theta(n^3)$ [Laumanns et al, TEC'04]
CountOnesCountZeroes	$O(n^2 \log n)$ [Qian et al., AIJ'13]	$O(n^2 \log n)$ [Laumanns et al, TEC'04]
OneMinMax	$O(n^2 \log n)$ [Giel and Lehre, ECJ'10]	$O(n^2 \log n)$ [Giel and Lehre, ECJ'10]
OneJumpZeroJump	$O((n-2k)n^k)$ [Zheng and Doerr, ECJ'23]	cannot find the Pareto front [Zheng and Doerr, ECJ'23]

GSEMO Solving Multi-objective Combinatorial Problems

GSEMO: a simple MOEA which employs bit-wise mutation to generate an offspring solution in each iteration and keeps the non-dominated solutions in the population

GSEMO
- multi-objective minimum spanning tree [Neumann, EJOR'07; Qian et al., AIJ'13]

DEMO (diversity evolutionary multiobjective optimizer)
- multi-objective shortest path [Horoba, FOGA'09; Neumann and Theile, PPSN'10]

GSEMO
- multi-objective knapsack [Laumanns et al, NC'04]

Effectiveness of Some Strategies for MOEAs

Brief review:

- greedy selection [Laumanns et al., TEC’04]
- crossover [Qian et al., AIJ’13; Dang et al., AAAI’23]
- diversity-based parent selection [Osuna et al., 2020]
- Selection hyper-heuristics [Qian et al., PPSN’16]
- diversity [Friedrich et al., TCS’10]
- fairness [Laumanns et al., TEC’04; Friedrich et al., 2011]

For example, recombination can accelerate the filling of the Pareto front by recombining diverse Pareto optimal solutions [Qian et al., AIJ’13]

Running Time Analysis of Practical MOEAs

[Zheng, Liu and Doerr, AAAI'22] analyzed the expected running time of NSGA-II (without crossover) for the first time

	Upper bound	Lower bound
LeadingOnesTrailingZeroes	$O(\mu n^2)$, if $\mu \geq 4(n+1)$	-------------------- population size
OneMinMax	$O(\mu n \log n)$, if $\mu \geq 4(n+1)$	exponential, if $\mu = n+1$

[Bian and Qian, PPSN'22] analyzed the standard NSGA-II which uses binary tournament selection, one-point crossover and bit-wise mutation

- solves LeadingOnesTrailingZeroes in $O(\mu n^2)$
- expected running time can be improved to $O(\mu n)$ if using stochastic tournament selection strategy

Running Time Analysis of Practical MOEAs

Other results of NSGA-II:

- OneMinMax: $\Omega(\mu n \log n)$, if $\mu = c(n+1)$ for some $c \geq 4$ [Doerr and Qu, AAAI'23]
- OneJumpZeroJump:
  - $\Theta(Nn^k)$, where $\mu \geq 4(n-2k+3)$ [Doerr and Qu, TEC'23; Doerr and Qu, AAAI'23]
  - bit-wise mutation and uniform crossover: $O\left(\frac{N^2(Cn)^k}{(k-1)!}\right)$,
    where $\mu = c(n-2k+3)$ and $c > 4$, $C = \left(\frac{4c}{c-4}\right)^2$ [Doerr and Qu, AAAI'23]

NSGA-II solving bi-objective MST [Cerf et al., IJCAI'23]: $O(m^2 n w_{\max} \log n w_{\max})$

NSGA-III solving 3OneMinMax [Wietheger and Doerr, IJCAI'23]: $O(n \log n)$

Running Time Analysis of Practical MOEAs

Analysis of SIBEA (a simple indicator based MOEA):

- LeadingOnesTrailingZeroes: $O(\mu n^2)$ [Brockhoff et al., PPSN'08]
- OneMinMax: $O(\mu n \log n)$ [Nguyen et al., TCS'15]

SMS-EMOA solving OneJumpZeroJump: $O\left(\mu n^k\right) \wedge \Omega(n^k)$ [Bian et al., IJCAI'23]

Our work contributes to running time analysis of a major type of MOEAs, i.e., combining non-dominated sorting and quality indicators, for the first time

Analysis of MOEA/D:

- MOEA/D with Tchbycheff decomposition [Li et al., TEC'16]
  - CountOnesCountZeroes: $O(n^2 \, log \, n)$
  - LPTNO (an extension of LeadingOnesTrailingZeroes): $O(n^3)$
- comparison of different decomposition methods [Huang et al., IJCAI'21]

Running Time Analysis of MOEAs under noise

Posterior noise [Dang et al., GECCO'23]:

- noise model: (δ, p)-Bernoulli noise model ($\delta > f_{\max} - f_{\min}$)

Noisy objective vector

$$\tilde{\boldsymbol{f}}(\boldsymbol{x}) = \begin{cases} \boldsymbol{f}(\boldsymbol{x}) + \delta \cdot \mathbf{1}, & \text{with probability } p, \\ \boldsymbol{f}(\boldsymbol{x}), & \text{otherwise} \end{cases}$$

- result: the expected running time of GSEMO and NSGA-II solving LeadingOnesTrailingZeroes is exponential and $O(\mu n^2)$, respectively

Prior noise [Dinot et al., IJCAI'23] :

- one-bit noise: flips a uniformly chosen bit of a solution with prob. p before evaluation
- result: the expected running time of SEMO without reevaluation solving OneMinMax is $O(n^2 \log n)$, better than that using reevaluation

Conclusion

Previous running time analysis of MOEAs mainly focuses on simple MOEAs, while recently, researchers have started to analyze practical MOEAs

- (G)SEMO on synthetic problems
- (G)SEMO on combinatorial problems
- Effectiveness of some strategies based on (G)SEMO
- Analysis of practical MOEAs
- Analysis of MOEAs under noise

Many theoretical works can be done in MOEAs

Thank you