大学院技術英語

佐藤 寛之
h.sato@uec.ac.jp

やること

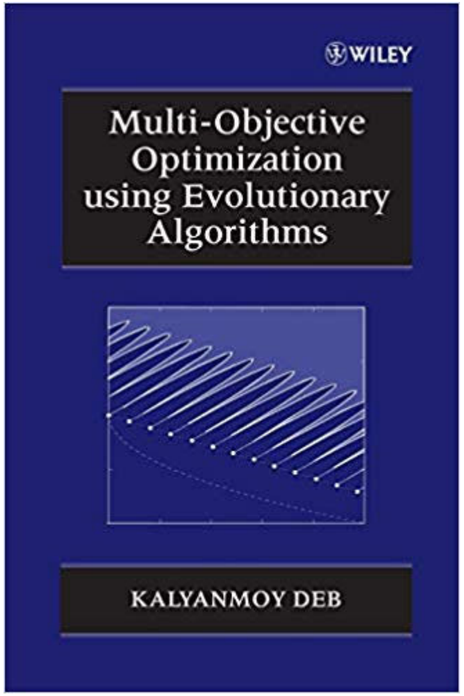
課題：配布資料の日本語訳
〆切：5/27講義開始時

5/13
- 前提知識の解説（約20分）
- 課題への取り組み（約60分）
- 課題に関するアドバイス（必要に応じて）

5/20
- 課題に関するアドバイスと質疑応答（約20分）
- 課題への取り組み（約60分）

配布資料

- Kalyanmoy Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*, John Wiley & Sons, Inc., New York, NY, USA, 2001.
  - Section 6.2 (NSGA-II)



- **進化計算による多目的最適化の教科書**
  - **本研究分野の発展に大きく寄与した良書**
  - **対象節は最も有名なアルゴリズムNSGA-IIの説明**

K. Deb, A. Pratap, S. Agarwal and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," in *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182-197, 2002.(**27,021 citations at Google scholar**)

本日

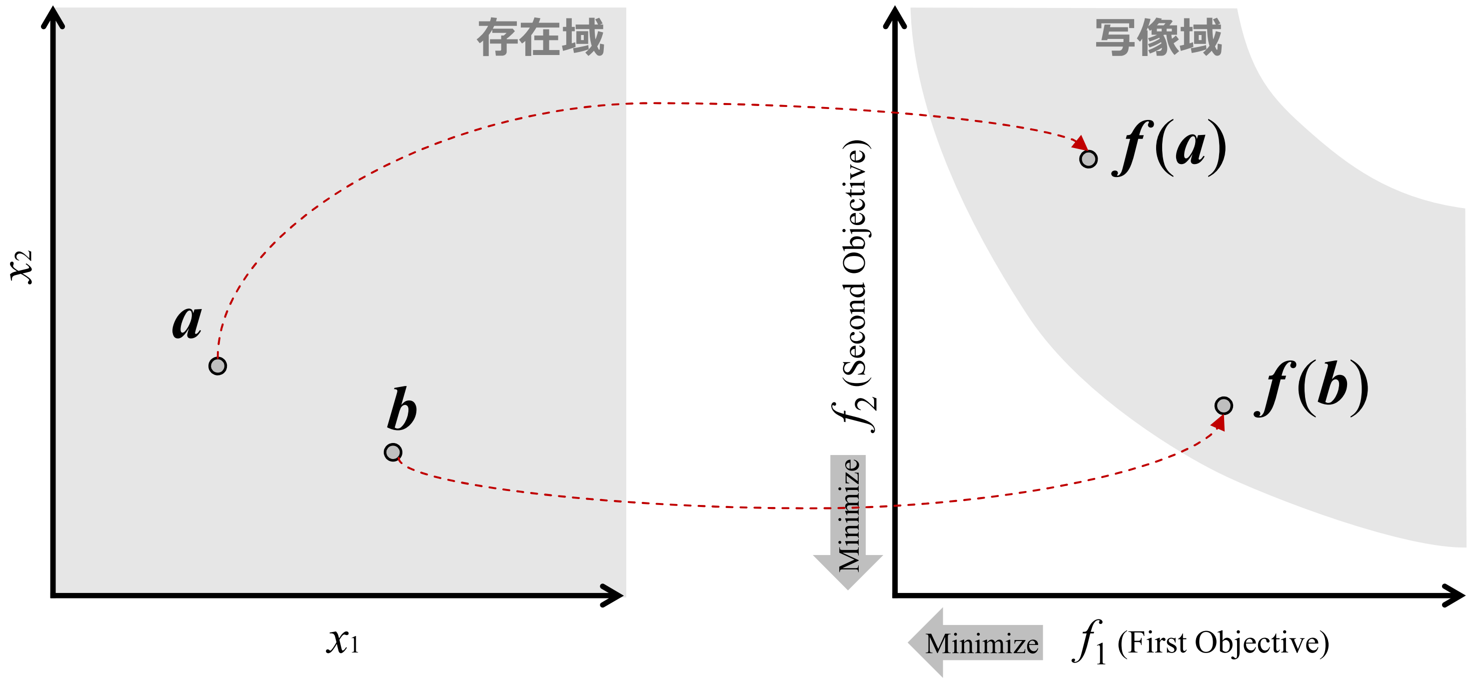
- **英語によるアルゴリズムの記述を知る**

  専門的だが手計算の記載があるため自分の理解を確認しやすい

多目的最適化問題

- 複数の目的関数$f_i\ (i = 1,2, \dots, M)$を最小化(最大化)する設計変数ベクトル$\boldsymbol{x} = (x_1, x_2, \dots, x_N)$を見出す問題

$$\text{Minimize/Maximize } f_i(\boldsymbol{x})(i = 1,2, \cdots, M)$$



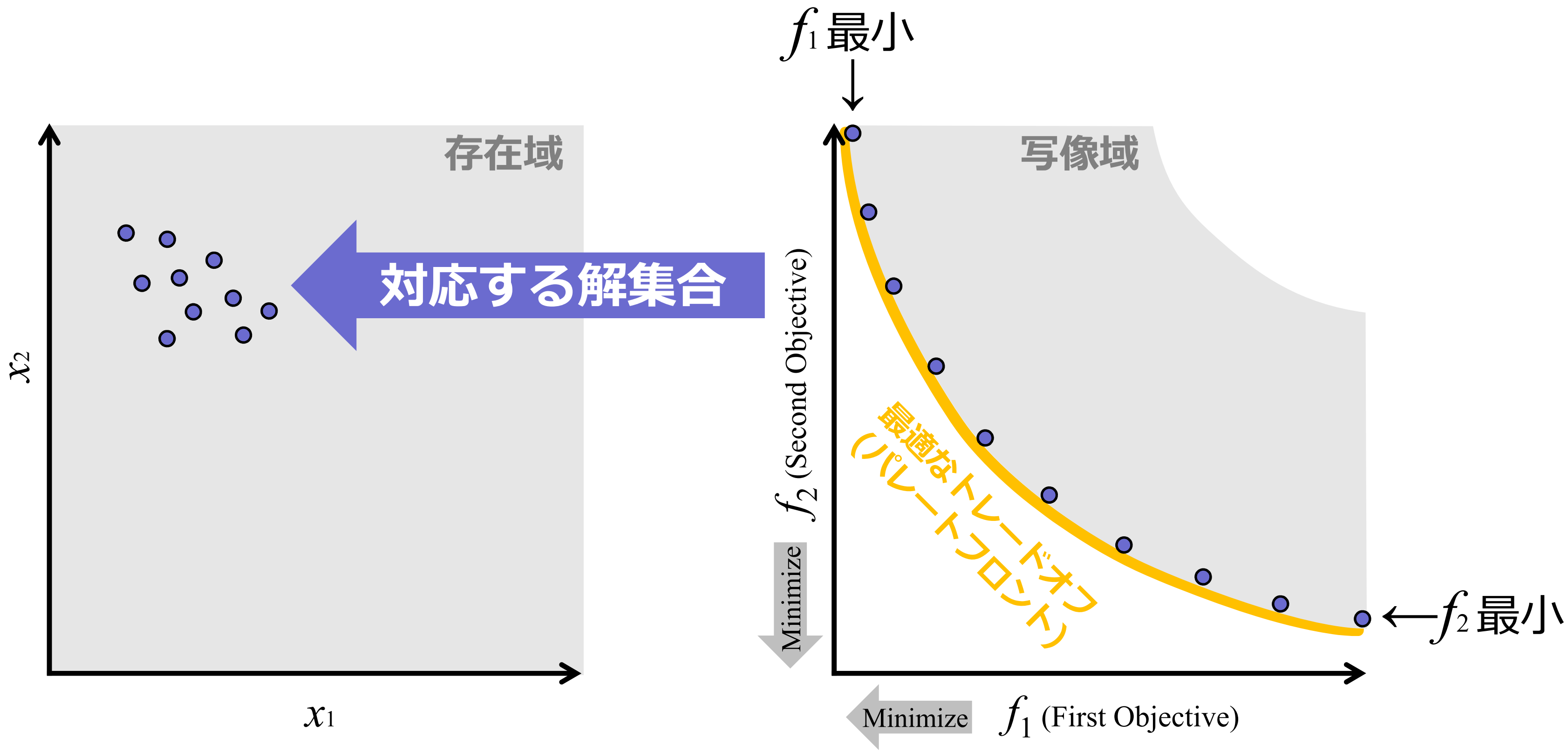
設計変数空間
(Design Variable Space)

目的関数空間
(Objective Space)

多目的最適化問題のゴール

- 複数の目的関数の間にはトレードオフ関係が存在

➡ 最適なトレードオフを近似する解集合を獲得

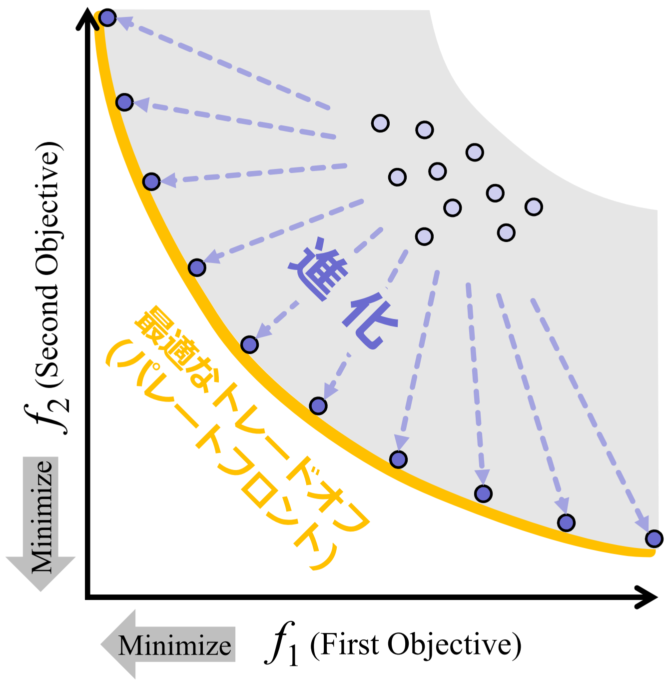


設計変数空間
(Design Variable Space)

目的関数空間
(Objective Space)

進化計算による多目的最適化

- **解集団をパレートフロントへ進化させる**
  - パレートフロントに近い解の設計変数ベクトルを手掛かりに，新しい解を生成することを繰り返す



進化計算による多目的最適化のイメージ

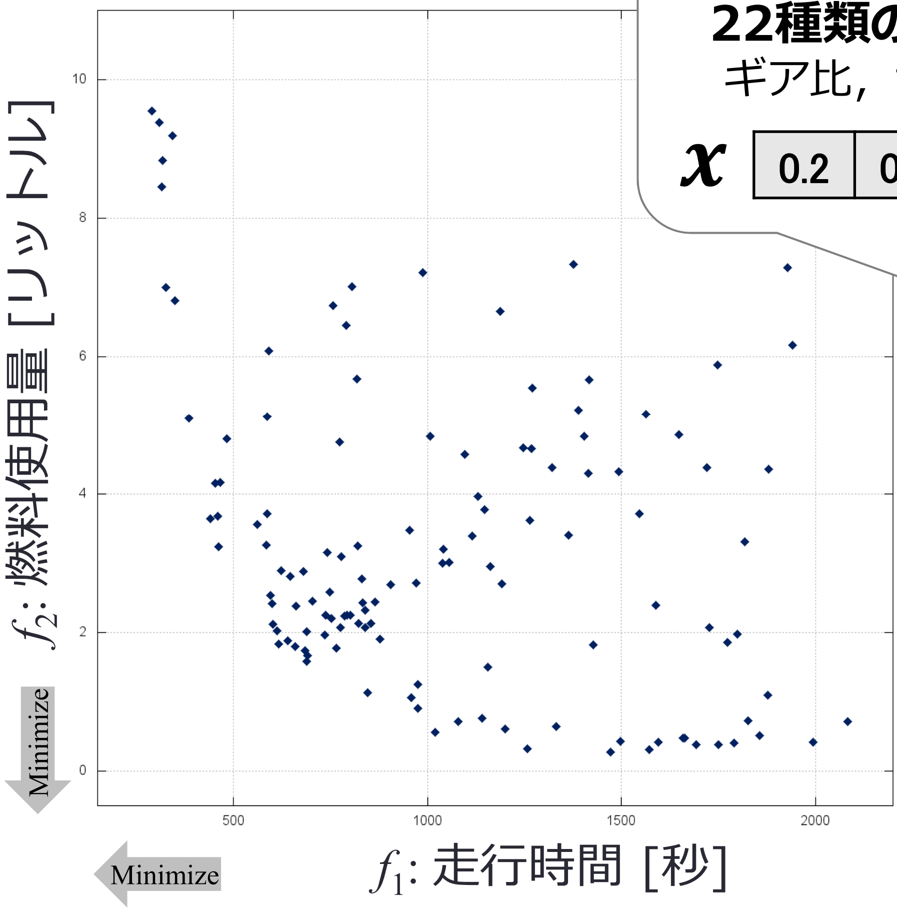
例：多目的自動車設計最適化

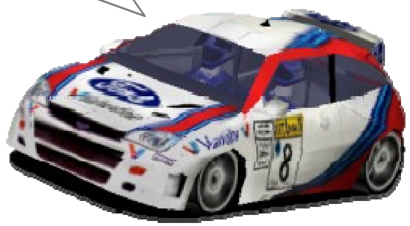
<u>遺伝子 (設計図)</u>

22種類の実数値：

ギア比，サスペンション強度など

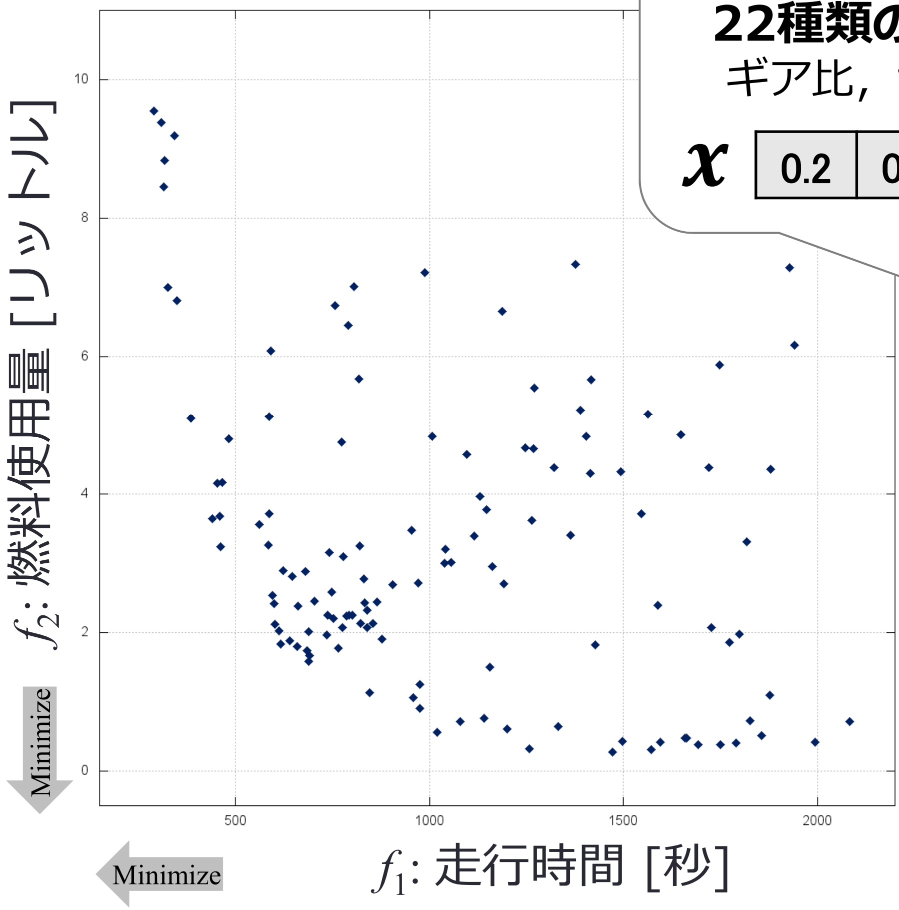
x						
	0.2	0.5	0.9	0.1	…	1.0





進化計算による多目的最適化のイメージ

例：多目的自動車設計最適化



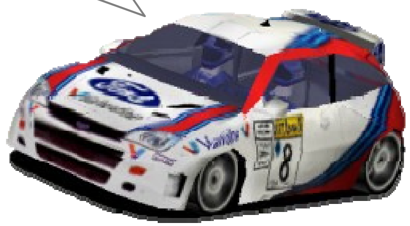
遺伝子 (設計図)

22種類の実数値：

ギア比，サスペンション強度など

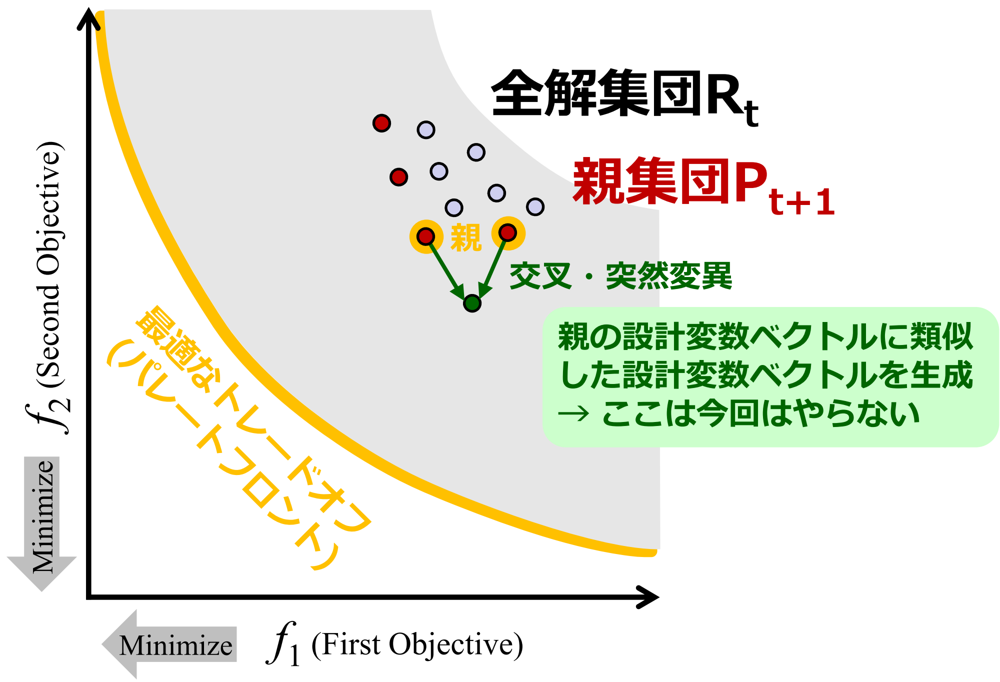
$\boldsymbol{x}$

0.2	0.5	0.9	0.1	…	1.0



アルゴリズムの概要

Step 1: 全解集団R_tをランダムに生成する.
Step 2: **全解集団R_tからパレートフロントの近似に適した親集団P_{t+1}を選択する.**
Step 3: **親集団P_{t+1}から親を選択**，**交叉・突然変異を施して子を生成することを繰り返して，子集団Q_{t+1}をつくる.**
Step 4: 親集団P_{t+1}と子集団Q_{t+1}を合わせた全解集団R_{t+1}をつくる.
Step 5: t=t+1して，**Step 2**へ戻る.



どんな解集合を得るべきか

最適なトレードオフを高精度に近似する解集合の特性

①収束性が高い

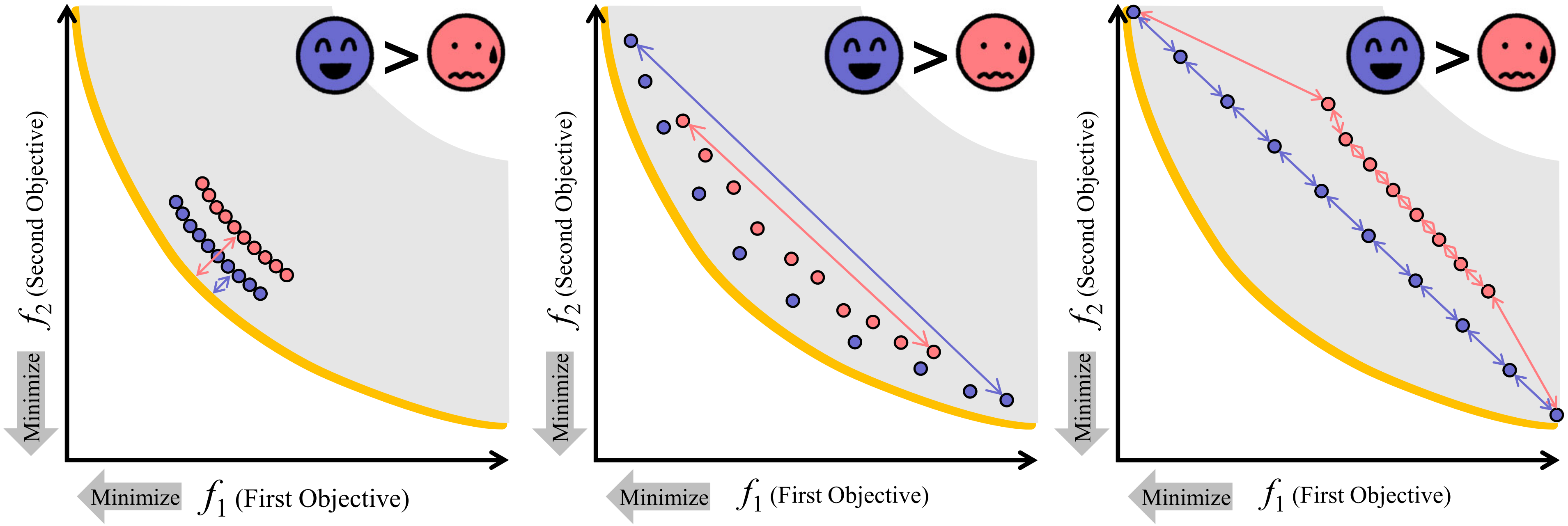
パレートフロントに近い
→ 解集合の最適性が良い

②多様性が高い

広く分布している
→ パレートフロントを広域に表現

③均一性が高い

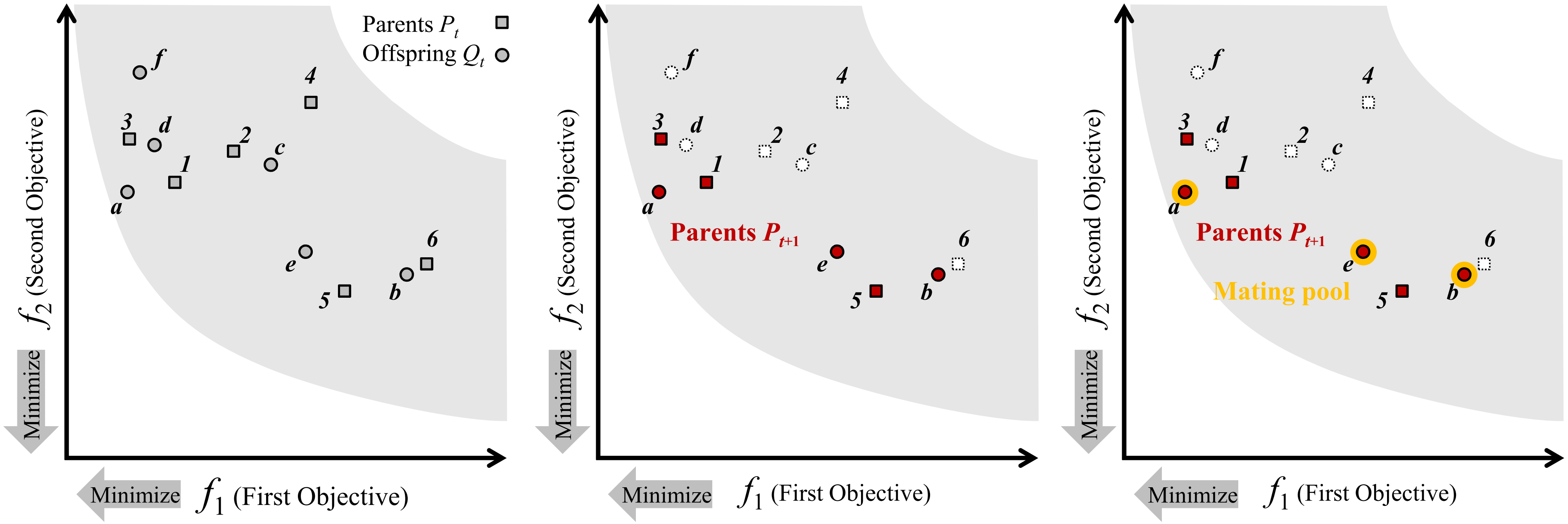
解と解の距離のばらつきが小さい
→ パレートフロントを均一に表現



今回，明らかにするアルゴリズム

どうやって選ぶか?　　どうやって選ぶか?

全解集団R_t ➡ 親集団P_{t+1}の選択 ➡ 親の選択



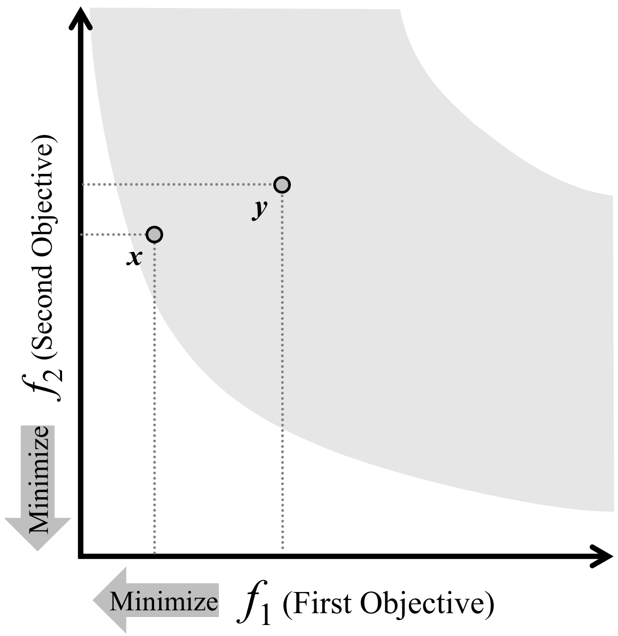
解の①収束性，②多様性，③均一性が基準になって親が選ばれるので詳しくアルゴリズムを紐解いてみてください.

収束性関連：パレート支配

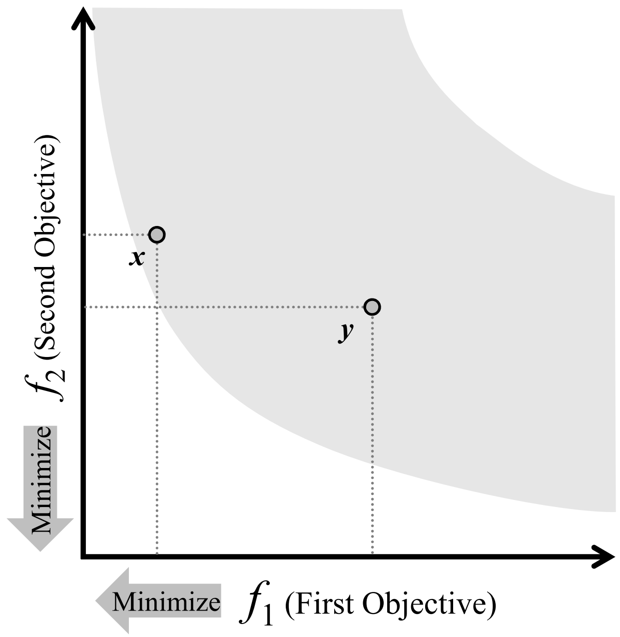
解xとyが次式を満たすとき，xはyを支配する(xがyより良い)

$$\forall i \in \{1,2,\ldots,M\}: f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{y}) \wedge \forall i \in \{1,2,\ldots,M\}: f_i(\boldsymbol{x}) < f_i(\boldsymbol{y})$$

<u>ケース１</u>
xがyを支配する



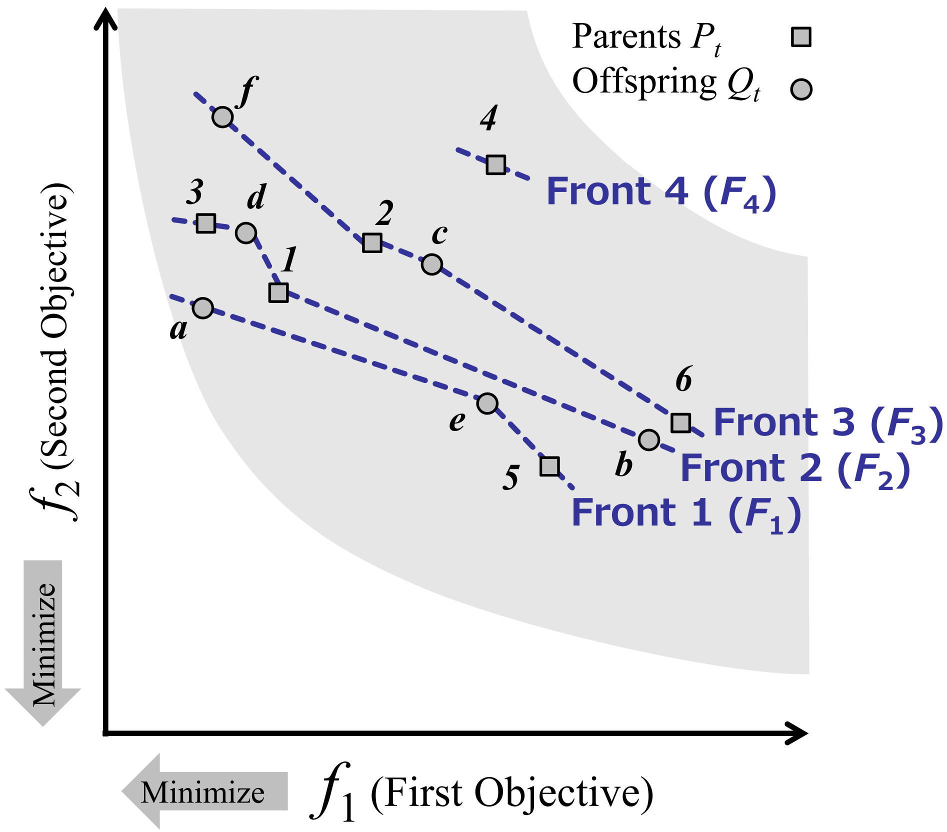
<u>ケース２</u>
xがyを支配しない



収束性関連：非支配ソーティング Non-dominated Sorting

支配されないレベルで解をグループ(フロント)化

– 全解集団から支配されない解集合を順次取り除き，フロントにグループ化する



手順例

全解集団R_t={***1,2,3,4,5,6,a,b,c,d,e,f***}

1. 解集団R_tから支配されないF_1={***a,e,5***}を取り出す． $R_t=R_t \backslash F_1$={***1,2,3,4,6,b,c,d,f***}になる．
2. 解集団R_tから支配されないF_2={***1,3,b,d***}を取り出す． $R_t=R_t \backslash F_2$={***2,4,6,c,f***}になる．
3. 解集団R_tから支配されないF_3= {***2,6,c,f***}を取り出す． $R_t=R_t \backslash F_3$={***4***}になる．
4. 解集団R_tから支配されないF_4={***4***}を取り出す． $R_t=R_t \backslash F_3=\emptyset$になる．

**フロント番号が小さい解ほど，
パレートフロントに対する収束性が高い
➡親に選ばれやすい**

参考資料

- Solution: 解
- Population: 解集団
- Recombination: 交叉
- Crossover: 交叉
- Mutation: 突然変異

- M: 目的数
- N: 集団サイズ
- O(・):ビックオー記法