GECCO '18 Companion, July 15–19, 2018, Kyoto, Japan

ACM ISBN 978-1-4503-5764-7/18/07.
https://doi.org/10.1145/3205651.3207856

Decomposition Multi-Objective Optimisation

Current Developments and Future Opportunities

Ke Li k.li@exeter.ac.uk
Department of Computer Science, University of Exeter
Qingfu Zhang qingfu.zhang@cityu.edu.hk
Department fo Computer Science, City University of Hong Kong

Outline

- Why Multi-Objective Optimisation Important?
- Basic Concepts
- Simple MOEA/D
- Current Developments
  - Decomposition methods
  - Search methods
  - Collaboration
- Resources
- Future Directions

2

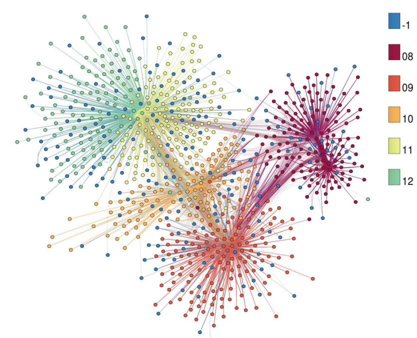
Outline

- Why Multi-Objective Optimisation Important?
- Basic Concepts
- Simple MOEA/D
- Current Developments
  - Decomposition methods
  - Search methods
  - Collaboration
    - Mating selection
    - Replacement
- Resources
- Future Directions

3

Why Multi-Objective Optimisation Important?

- Many real-world applications involve more than one objective



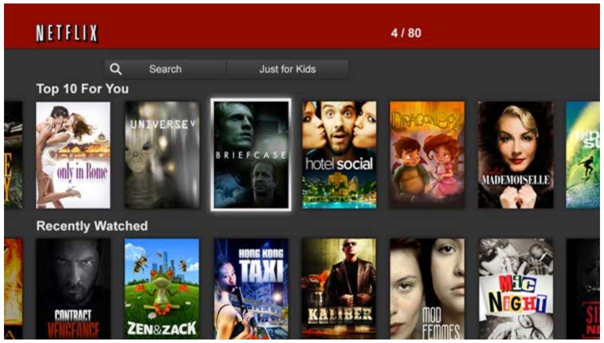
Discrepancy of the same community/cluster ⟶ minimize
Discrepancy of different communities/clusters ⟶ maximize

[1] M. Gong, et. al., "Complex Network Clustering by Multiobjective Discrete Particle Swarm Optimization Based on Decomposition", IEEE Trans. Evol. Comput., 18(1): 82-97, 2014.

4

Why Multi-Objective Optimisation Important?

- Many real-world applications involve more than one objective



accuracy diversity novelty

[2] M. Ribeiro, et. al., "Multi-Objective Pareto-Efficient Approaches for Recommender Systems", ACM Trans. Intelligent Systems and Technology, 5(4): 1-20, 2014.

Why Multi-Objective Optimisation Important?

- Many real-world applications involve more than one objective

Shinkansen N700, bullet train [3]

Exeter water distribution network [4]

[3] http://english.jr-central.co.jp/news/n20040616/index.html
[4] R. Farmani, et al. "Evolutionary multi-objective optimization in water distribution network design", Engineering Optimization, 37(2): 167-183, 2005

Outline

- Why Multi-Objective Optimisation Important?
- Basic Concepts
- Simple MOEA/D
- Current Developments
  - Decomposition methods
  - Search methods
  - Collaboration
    - Mating selection
    - Replacement
- Resources
- Future Directions

Multi-objective Optimisation Problem (MOP)

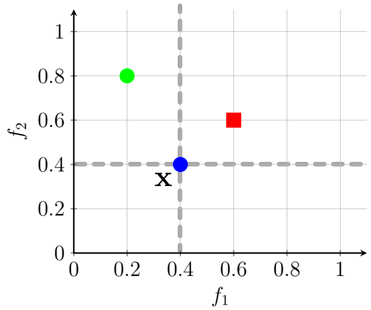
- Mathematical definition (continuous problem)

$$
\begin{array}{ll}
\text{minimize} & \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), \cdots, f_m(\mathbf{x}))^T \\
\text{subject to} & g_j(\mathbf{x}) \geq a_j, \quad j = 1, \cdots, q \\
& h_j(\mathbf{x}) = b_j, \quad j = q+1, \cdots, \ell \\
& \mathbf{x} \in \Omega
\end{array}
$$

- $\mathbf{x}$: decision variable
- $\mathbf{F}$: objective vector
- Ω: decision space
- $\Omega \to \mathbb{R}^m$: objective space

Which Solution is Better?

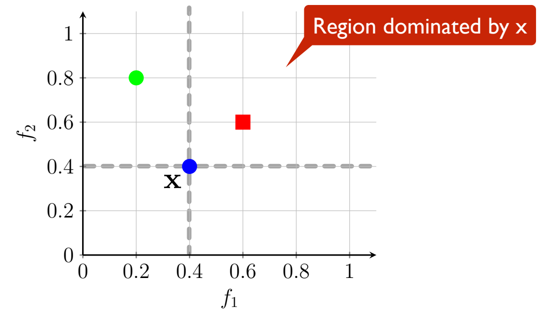
- Pareto domination: $\mathbf{x}^1 \preceq \mathbf{x}^2$
  - $\mathbf{F}(\mathbf{x}^1)$ is no worse than $\mathbf{F}(\mathbf{x}^2)$ in any objective, and
  - $\mathbf{F}(\mathbf{x}^1)$ is better than $\mathbf{F}(\mathbf{x}^2)$ in at least one objective



9

Which Solution is Better?

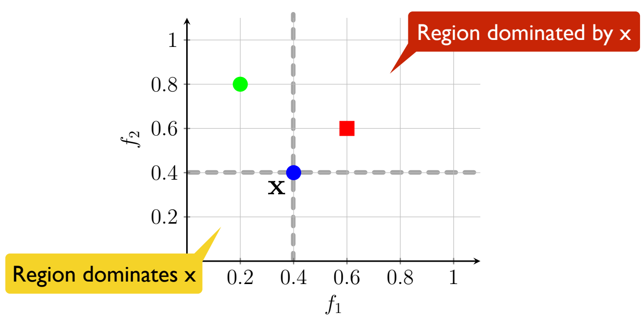
- Pareto domination: $\mathbf{x}^1 \preceq \mathbf{x}^2$
  - $\mathbf{F}(\mathbf{x}^1)$ is no worse than $\mathbf{F}(\mathbf{x}^2)$ in any objective, and
  - $\mathbf{F}(\mathbf{x}^1)$ is better than $\mathbf{F}(\mathbf{x}^2)$ in at least one objective



9

Which Solution is Better?

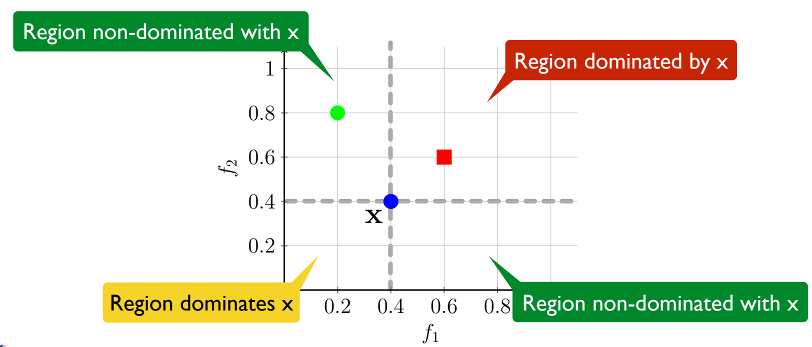
- Pareto domination: $\mathbf{x}^1 \preceq \mathbf{x}^2$
  - $\mathbf{F}(\mathbf{x}^1)$ is no worse than $\mathbf{F}(\mathbf{x}^2)$ in any objective, and
  - $\mathbf{F}(\mathbf{x}^1)$ is better than $\mathbf{F}(\mathbf{x}^2)$ in at least one objective



9

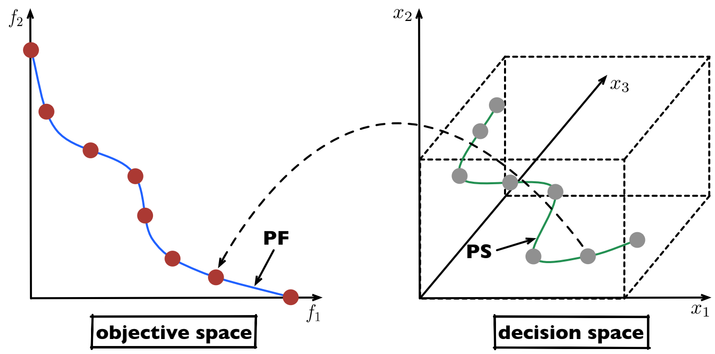
Which Solution is Better?

- Pareto domination: $\mathbf{x}^1 \preceq \mathbf{x}^2$
  - $\mathbf{F}(\mathbf{x}^1)$ is no worse than $\mathbf{F}(\mathbf{x}^2)$ in any objective, and
  - $\mathbf{F}(\mathbf{x}^1)$ is better than $\mathbf{F}(\mathbf{x}^2)$ in at least one objective



9

Pareto-optimal Solutions = Best Trade-off Candidates



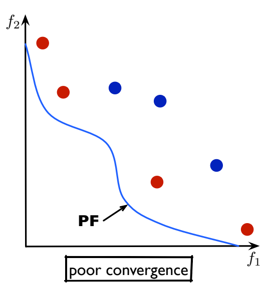
- x is Pareto-optimal iff no solution dominates it
- Pareto set (PS): all Pareto-optimal solutions in decision space
- Pareto front (PF): image of PS in the objective space

Convergence and Diversity in EMO

- Convergence: non-dominated, close to the PF
- Diversity: even distribution along the PF

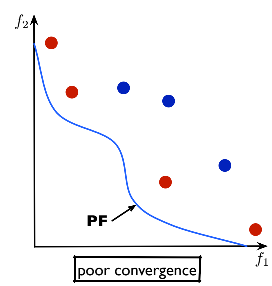
Convergence and Diversity in EMO

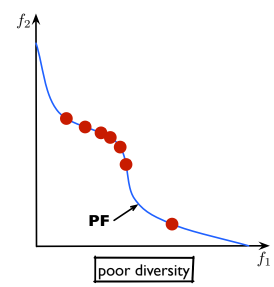
- Convergence: non-dominated, close to the PF
- Diversity: even distribution along the PF



Convergence and Diversity in EMO

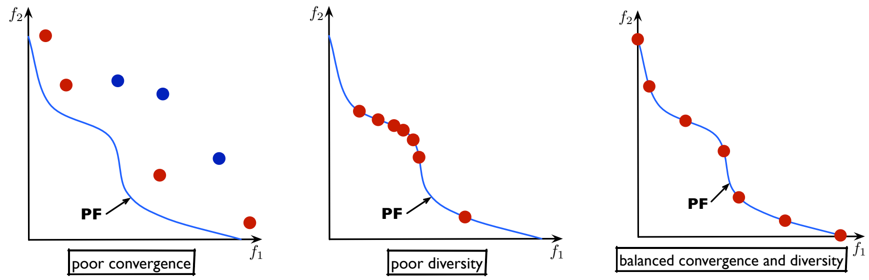
- Convergence: non-dominated, close to the PF
- Diversity: even distribution along the PF





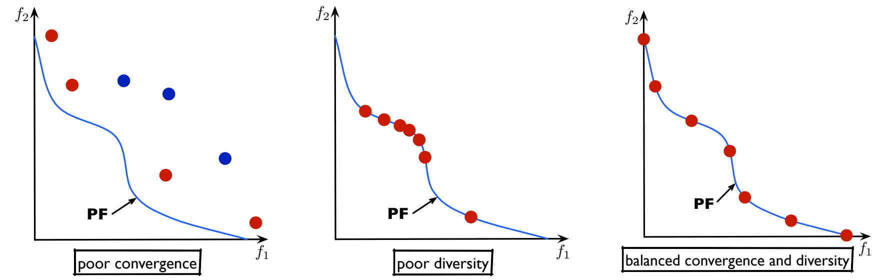
Convergence and Diversity in EMO

- Convergence: non-dominated, close to the PF
- Diversity: even distribution along the PF



Convergence and Diversity in EMO

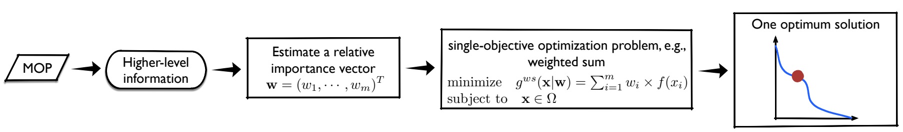
- Convergence: non-dominated, close to the PF
- Diversity: even distribution along the PF



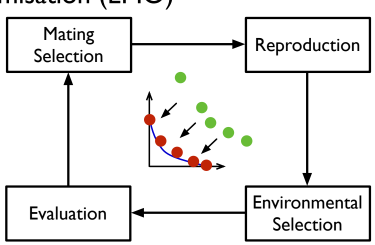
Balance between convergence and diversity is the corner stone

Classic Methods vs Evolutionary Approaches

- Classic multi-objective optimisation [4]



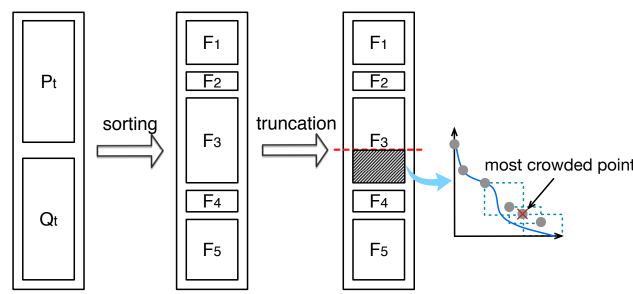
- Evolutionary multi-objective optimisation (EMO)
  - set-based method
  - approximate the PF at a time



[4] K. Deb, "Multi-Objective Optimization Using Evolutionary Algorithms", Wiley, 2009.

Pareto-based EMO Methods

- Two-step procedure
  - Rank the population by dominance principle
    - dominance level, dominance count, …
  - Refine the dominance-based ranking by density estimation
    - crowding distance, k-th nearest neighbour, …



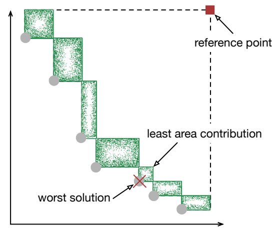
[5] K. Deb, et. al., "A fast and elitist multiobjective genetic algorithm: NSGA-II", IEEE Trans. Evol. Comput., 6(2): 182-197, 2002.

Indicator-based EMO Methods

- A (unary) quality indicator I is a function $I : \Psi = 2^X \mapsto \mathbb{R}$ that assigns a Pareto set approximation a real value.

Multi-objective Problem → **Indicator** → Single-objective Problem

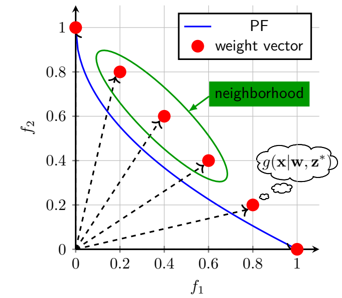
NOTE: performance indicator should be dominance preserving!



[6] N. Beume, et. al., "SMS-EMOA: Multiobjective selection based on dominated hypervolume", Eur J Oper Res. 181(3): 1653-1669, 2007.

General Framework of MOEA/D

- Basic idea
  - Decomposition
    - Decompose the task of approximating the PF into *N* subtasks, i.e. MOP to subproblems
    - Each subproblem can be either single objective or multi-objective
  - Collaboration
    - Population-based technique: *N* agents for *N* subproblems.
    - Subproblems are related to each other while *N* agents solve these subproblems in a collaborative manner.



Outline

- Why Multi-Objective Optimisation Important?
- Basic Concepts
- Simple MOEA/D
- Current Developments
  - Decomposition methods
  - Search methods
  - Collaboration
    - Mating selection
    - Replacement
- Resources
- Future Directions

Simple MOEA/D

- A simple MOEA/D works as follows:

Step 1: Initialize a population of solutions $P := \{\mathbf{x}^i\}_{i=1}^N$, a set of reference points $W := \{\mathbf{w}^i\}_{i=1}^N$ and their neighborhood structure. Randomly assign each solution to a reference point.

Step 2: For $i = 1, \cdots, N$, do

Step 2.1: Randomly selects a required number of mating parents from $\mathbf{w}^i$'s neighborhood.

Step 2.2: Use crossover and mutation to reproduce offspring $\mathbf{x}^c$.

Step 2.3: Update the subproblems within the neighborhood of $\mathbf{w}^i$ by $\mathbf{x}^c$.

Step 3: If the stopping criteria is met, then stop and output the population. Otherwise, go to Step 2.

[7] Q. Zhang et al., "MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition", IEEE Trans. Evol. Comput., 11(6): 712-731, 2007.

Simple MOEA/D

- A simple MOEA/D works as follows:

Step 1: Initialize a population of solutions $P := \{\mathbf{x}^i\}_{i=1}^N$, a set of reference points $W := \{\mathbf{w}^i\}_{i=1}^N$ and their neighborhood structure. Randomly assign each solution to a reference point.

Step 2: For $i = 1, \cdots, N$, do

Step 2.1: Randomly selects a required number of mating parents from $\mathbf{w}^i$'s neighborhood.

Step 2.2: Use crossover and mutation to reproduce offspring $\mathbf{x}^c$.

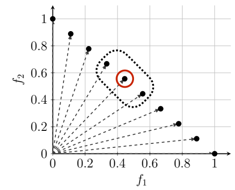
Step 2.3: Update the subproblems within the neighborhood of $\mathbf{w}^i$ by $\mathbf{x}^c$.

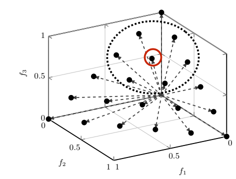
Step 3: If the stopping criteria is met, then stop and output the population. Otherwise, go to Step 2.

[7] Q. Zhang et al., "MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition", IEEE Trans. Evol. Comput., 11(6): 712-731, 2007.

Algorithm Settings

- Weight vector/Reference point Setting
  - Use Das and Dennis's method [8] to sample a set of uniformly distributed weight vectors from a unit simplex
  - $\mathbf{w} = (w_1, \cdots, w_m)^T$ where $\sum_{i=1}^{m} w_i = 1, \mathbf{w} \in \mathbb{R}^m$
  - Each weight vector set a direction line (starting from the utopian point)
- Neighbourhood structure:
  - Two subproblems are neighbours if their weight vectors are close.
  - Neighbouring subproblems are more likely assumed to have similar property (e.g. similar objective function and/or optimal solution).





[8] I. Das et. al., "Normal-Boundary Intersection: A New Method for Generating the Pareto Surface in Nonlinear Multicriteria Optimization Problems", SIAM J. Optim, 8(3): 631-657, 1998.

Algorithm Settings

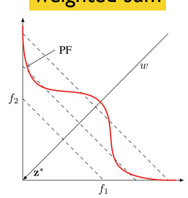
- Subproblem formulation

multiple objectives — parameters — single objective

$(f_1(\mathbf{x}), \cdots, f_m(\mathbf{x})) \Longrightarrow$ transformation $\Longrightarrow g(\mathbf{x}|\cdot)$

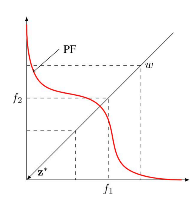
A scalarizing function $g : \mathbb{R}^m \to \mathbb{R}$ that maps each objective vector $\mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), \cdots, f_m(\mathbf{x})) \in \mathbb{R}^m$ to a real value $g(\mathbf{F}(\mathbf{x})) \in \mathbb{R}$

weighted sum



$$g(\mathbf{x}|\mathbf{w}) = \sum_{i=1}^{m} w_i \times f_i(\mathbf{x})$$

weighted Tchebycheff



$$g(\mathbf{x}|\mathbf{w}, \mathbf{z}^*) = \max_{1 \le i \le m} w_i |f_i(\mathbf{x} - z_i^*|$$

Simple MOEA/D (cont.)

- A simple MOEA/D works as follows:

Step 1: Initialize a population of solutions $P := \{\mathbf{x}^i\}_{i=1}^N$, a set of reference points $W := \{\mathbf{w}^i\}_{i=1}^N$ and their neighborhood structure. Randomly assign each solution to a reference point.

Step 2: For $i = 1, \cdots, N$, do

Step 2.1: Randomly selects a required number of mating parents from $\mathbf{w}^i$'s neighborhood.

Step 2.2: Use crossover and mutation to reproduce offspring $\mathbf{x}^c$.

Step 2.3: Update the subproblems within the neighborhood of $\mathbf{w}^i$ by $\mathbf{x}^c$.

Step 3: If the stopping criteria is met, then stop and output the population. Otherwise, go to Step 2.

Simple MOEA/D (cont.)

- A simple MOEA/D works as follows:

Step 1: Initialize a population of solutions $P := \{\mathbf{x}^i\}_{i=1}^N$, a set of reference points $W := \{\mathbf{w}^i\}_{i=1}^N$ and their neighborhood structure. Randomly assign each solution to a reference point.

Step 2: For $i = 1, \cdots, N$, do

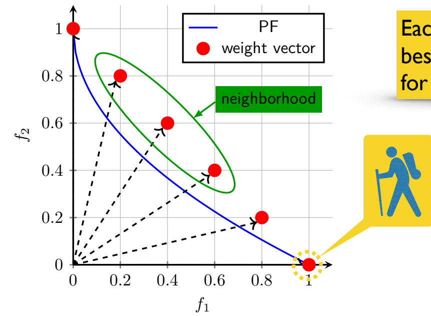
Step 2.1: Randomly selects a required number of mating parents from $\mathbf{w}^i$'s neighborhood.

Step 2.2: Use crossover and mutation to reproduce offspring $\mathbf{x}^c$.

Step 2.3: Update the subproblems within the neighborhood of $\mathbf{w}^i$ by $\mathbf{x}^c$.

Step 3: If the stopping criteria is met, then stop and output the population. Otherwise, go to Step 2.

Collaboration Among Different Agents



Each agent i records the best-so-far solution found for its subproblem (memory)

- At each iteration, each agent does the following:
  - Mating selection (local selection): borrows solutions from its neighbours.
  - Reproduction: reproduce a new solution by applying reproduction operators on its own solutions and borrowed solutions.
  - Replacement (local competition):
    - Pass the new solution among its neighbours (including itself).
    - Replace the old solution by the new one if the new one is better than old one for its objective.

Outline

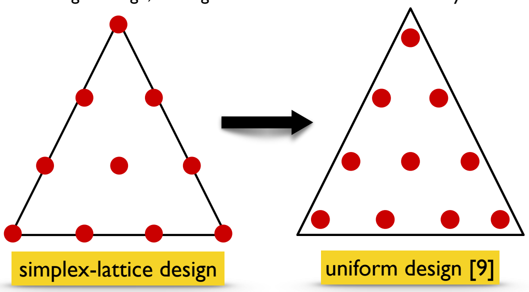
Setting of Weight Vectors

- Drawbacks of the Das and Dennis's method
  - Not very uniform [9]
  - Number of weights is restricted to $N = \binom{H+m-1}{m-1}$
    - N increases nonlinearly with m
    - If N is not large enough, all weight vectors will be at the boundary of the simplex

[9] Y-Y Tan, et al., "MOEA/D + uniform design: A new version of MOEA/D for optimization problems with many objectives", Comput & OR, 40: 1648-1660, 2013.
[10] K. Li, et al., "An Evolutionary Many-Objective Optimization Algorithm Based on Dominance and Decomposition", IEEE Trans. Evol. Comput., 19(5): 694-716, 2015.

Setting of Weight Vectors

- Drawbacks of the Das and Dennis's method
  - Not very uniform [9]
  - Number of weights is restricted to $N = \binom{H+m-1}{m-1}$
    - N increases nonlinearly with m
    - If N is not large enough, all weight vectors will be at the boundary of the simplex



[9] Y-Y Tan, et al., "MOEA/D + uniform design: A new version of MOEA/D for optimization problems with many objectives", Comput & OR, 40: 1648-1660, 2013.
[10] K. Li, et al., "An Evolutionary Many-Objective Optimization Algorithm Based on Dominance and Decomposition", IEEE Trans. Evol. Comput., 19(5): 694-716, 2015.

Setting of Weight Vectors

- Drawbacks of the Das and Dennis's method
  - Not very uniform [9]
  - Number of weights is restricted to $N = \binom{H+m-1}{m-1}$
    - N increases nonlinearly with m
    - If N is not large enough, all weight vectors will be at the boundary of the simplex

[9] Y-Y Tan, et al., "MOEA/D + uniform design: A new version of MOEA/D for optimization problems with many objectives", Comput & OR, 40: 1648-1660, 2013.
[10] K. Li, et al., "An Evolutionary Many-Objective Optimization Algorithm Based on Dominance and Decomposition", IEEE Trans. Evol. Comput., 19(5): 694-716, 2015.

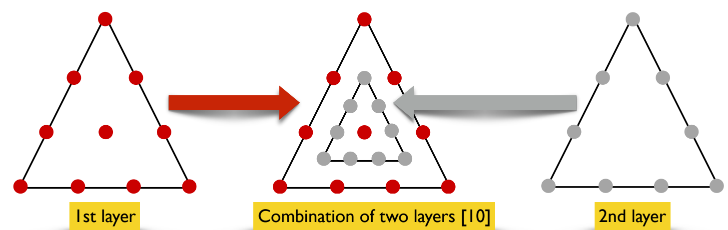
Setting of Weight Vectors

- Drawbacks of the Das and Dennis's method
  - Not very uniform [9]
  - Number of weights is restricted to $N = \binom{H+m-1}{m-1}$
    - N increases nonlinearly with m
    - If N is not large enough, all weight vectors will be at the boundary of the simplex

[9] Y-Y Tan, et al., "MOEA/D + uniform design: A new version of MOEA/D for optimization problems with many objectives", Comput & OR, 40: 1648-1660, 2013.
[10] K. Li, et al., "An Evolutionary Many-Objective Optimization Algorithm Based on Dominance and Decomposition", IEEE Trans. Evol. Comput., 19(5): 694-716, 2015.

Setting of Weight Vectors

- Drawbacks of the Das and Dennis's method
  - Not very uniform [9]
  - Number of weights is restricted to $N = \binom{H+m-1}{m-1}$
    - N increases nonlinearly with m
    - If N is not large enough, all weight vectors will be at the boundary of the simplex



[9] Y-Y Tan, et al., "MOEA/D + uniform design: A new version of MOEA/D for optimization problems with many objectives", Comput & OR, 40: 1648-1660, 2013.
[10] K. Li, et al., "An Evolutionary Many-Objective Optimization Algorithm Based on Dominance and Decomposition", IEEE Trans. Evol. Comput., 19(5): 694-716, 2015.

Setting of Weight Vectors (cont.)

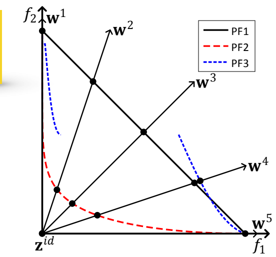
- Drawbacks of uniform distributed weight vectors
  - Do NOT always lead to evenly distributed solutions
  - Do NOT support all PF shapes
    - Disconnected PF
    - Inverted PF
    - …

[11] S. Jiang, et al., "Multiobjective Optimization by Decomposition with Pareto-adaptive Weight Vectors", ICNC'11, 1260-1264, 2011.
[12] Y. Qi, et al., "MOEA/D with Adaptive Weight Adjustment", Evol. Comput. 22(2): 231–264, 2014.
[13] M. Wu, et al., "Adaptive Weights Generation for Decomposition-Based Multi-Objective Optimization Using Gaussian Process Regression", GECCO'17, 641–648, 2017.
[14] F. Gu, et al., "Self-Organizing Map-Based Weight Design for Decomposition-Based Many-Objective Evolutionary Algorithm", IEEE Trans. Evol. Comput., 22(2): 211-225, 2018.

Setting of Weight Vectors (cont.)

- Drawbacks of uniform distributed weight vectors
  - Do NOT always lead to evenly distributed solutions
  - Do NOT support all PF shapes
    - Disconnected PF
    - Inverted PF
    - …

If the PF meets $\sum_{i=1}^{m} f_i = 1$, that's fine; otherwise …



[11] S. Jiang, et al., "Multiobjective Optimization by Decomposition with Pareto-adaptive Weight Vectors", ICNC'11, 1260-1264, 2011.
[12] Y. Qi, et al., "MOEA/D with Adaptive Weight Adjustment", Evol. Comput. 22(2): 231–264, 2014.
[13] M. Wu, et al., "Adaptive Weights Generation for Decomposition-Based Multi-Objective Optimization Using Gaussian Process Regression", GECCO'17, 641–648, 2017.
[14] F. Gu, et al., "Self-Organizing Map-Based Weight Design for Decomposition-Based Many-Objective Evolutionary Algorithm", IEEE Trans. Evol. Comput., 22(2): 211-225, 2018.

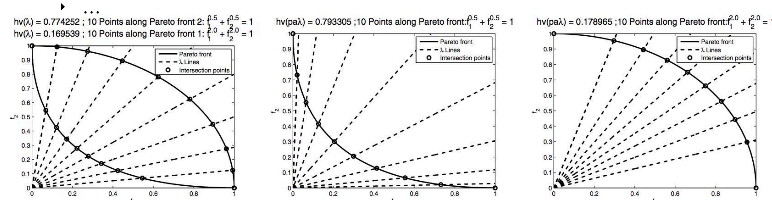
Setting of Weight Vectors (cont.)

- Drawbacks of uniform distributed weight vectors
  - Do NOT always lead to evenly distributed solutions
  - Do NOT support all PF shapes
    - Disconnected PF
    - Inverted PF
    - …

[11] S. Jiang, et al., "Multiobjective Optimization by Decomposition with Pareto-adaptive Weight Vectors", ICNC'11, 1260-1264, 2011.
[12] Y. Qi, et al., "MOEA/D with Adaptive Weight Adjustment", Evol. Comput. 22(2): 231–264, 2014.
[13] M. Wu, et al., "Adaptive Weights Generation for Decomposition-Based Multi-Objective Optimization Using Gaussian Process Regression", GECCO'17, 641–648, 2017.
[14] F. Gu, et al., "Self-Organizing Map-Based Weight Design for Decomposition-Based Many-Objective Evolutionary Algorithm", IEEE Trans. Evol. Comput., 22(2): 211-225, 2018.

Setting of Weight Vectors (cont.)

- Drawbacks of uniform distributed weight vectors
  - Do NOT always lead to evenly distributed solutions
  - Do NOT support all PF shapes
    - Disconnected PF
    - Inverted PF
    - …



(a) λ (MOEA/D) for convex and concave PF (b) $pa\lambda$ for convex PF (c) $pa\lambda$ for concave PF

Assume PF as $\sum_{i=1}^{m} f_i^p = 1$, estimate p according to the number of non-dominated solutions [11]

[12] Y. Qi, et al., "MOEA/D with Adaptive Weight Adjustment", Evol. Comput. 22(2): 231–264, 2014.
[13] M. Wu, et al., "Adaptive Weights Generation for Decomposition-Based Multi-Objective Optimization Using Gaussian Process Regression", GECCO'17, 641–648, 2017.
[14] F. Gu, et al., "Self-Organizing Map-Based Weight Design for Decomposition-Based Many-Objective Evolutionary Algorithm", IEEE Trans. Evol. Comput., 22(2): 211-225, 2018.

Setting of Weight Vectors (cont.)

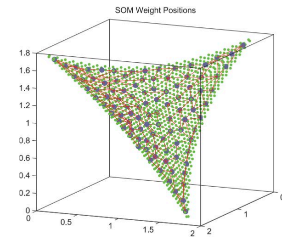
- Drawbacks of uniform distributed weight vectors
  - Do NOT always lead to evenly distributed solutions
  - Do NOT support all PF shapes
    - Disconnected PF
    - Inverted PF
    - …

[11] S. Jiang, et al.,"Multiobjective Optimization by Decomposition with Pareto-adaptive Weight Vectors", ICNC'11, 1260-1264, 2011.
[12] Y. Qi, et al.,"MOEA/D with Adaptive Weight Adjustment", Evol. Comput. 22(2): 231–264, 2014.
[13] M. Wu, et al., "Adaptive Weights Generation for Decomposition-Based Multi-Objective Optimization Using Gaussian Process Regression", GECCO'17, 641–648, 2017.
[14] F. Gu, et al., "Self-Organizing Map-Based Weight Design for Decomposition-Based Many-Objective Evolutionary Algorithm", IEEE Trans. Evol. Comput., 22(2): 211-225, 2018.

Setting of Weight Vectors (cont.)

- Drawbacks of uniform distributed weight vectors
  - Do NOT always lead to evenly distributed solutions
  - Do NOT support all PF shapes
    - Disconnected PF
    - Inverted PF
    - …

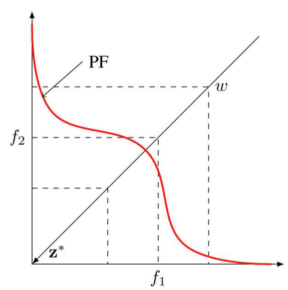
<u>Adaptive weight vectors adjustment</u>
- Estimate the PF shape progressively according to the current population
- Resample a set of weight vectors according to the estimated PF
  - ✓ Add new ones in feasible regions, and remove useless ones from infeasible regions [12]
  - ✓ Sampling from some estimated model, e.g. GP [13] and SOM [14]
- Construct new subproblems with respect to newly sampled weight vectors



[11] S. Jiang, et al.,"Multiobjective Optimization by Decomposition with Pareto-adaptive Weight Vectors", ICNC'11, 1260-1264, 2011.
[12] Y. Qi, et al.,"MOEA/D with Adaptive Weight Adjustment", Evol. Comput. 22(2): 231–264, 2014.
[13] M. Wu, et al., "Adaptive Weights Generation for Decomposition-Based Multi-Objective Optimization Using Gaussian Process Regression", GECCO'17, 641–648, 2017.
[14] F. Gu, et al., "Self-Organizing Map-Based Weight Design for Decomposition-Based Many-Objective Evolutionary Algorithm", IEEE Trans. Evol. Comput., 22(2): 211-225, 2018.

Revisit Weighted Tchebycheff

- Weighted Tchebycheff



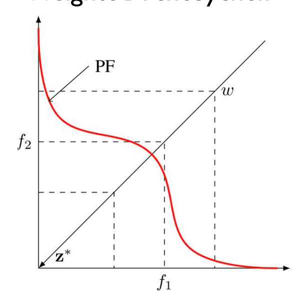
Drawback:
- non-smooth, weakly dominate solution
- evenly distributed weights evenly do NOT lead to distributed solutions
- might easily loose diversity

weighted Tchebycheff

$$g(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = \max_{1\le i\le m} w_i|f_i(\mathbf{x} - z_i^*|$$

Revisit Weighted Tchebycheff

- Weighted Tchebycheff



Drawback:
- non-smooth, weakly dominate solution [15]
- evenly distributed weights evenly do NOT lead to distributed solutions
- might easily loose diversity

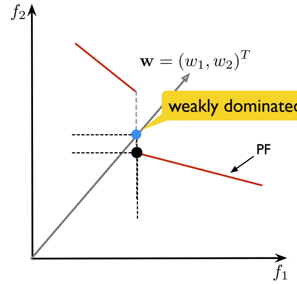
weighted Tchebycheff

$$g(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = \max_{1\le i\le m} w_i|f_i(\mathbf{x} - z_i^*|$$

[15] K. Miettinen, et al.,"Nonlinear Multiobjective Optimization", Kluwer Academic Publishers, Boston, 1999

Revisit Weighted Tchebycheff

- Weighted Tchebycheff



Drawback:
- non-smooth, weakly dominate solution [15]
- evenly distributed weights evenly do NOT lead to distributed solutions
- might easily loose diversity

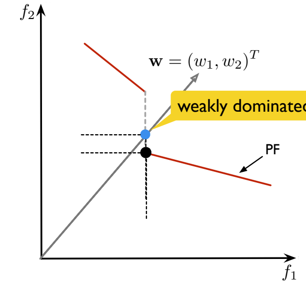
weighted Tchebycheff

$$g(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = \max_{1\le i\le m} w_i|f_i(\mathbf{x}-z_i^*|$$

[15] K. Miettinen, et al., "Nonlinear Multiobjective Optimization", Kluwer Academic Publishers, Boston, 1999

Revisit Weighted Tchebycheff

- Weighted Tchebycheff



Drawback:
- non-smooth, weakly dominate solution [15]
- evenly distributed weights evenly do NOT lead to distributed solutions
- might easily loose diversity

weighted Tchebycheff

$$g(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = \max_{1\le i\le m} w_i|f_i(\mathbf{x}-z_i^*|$$

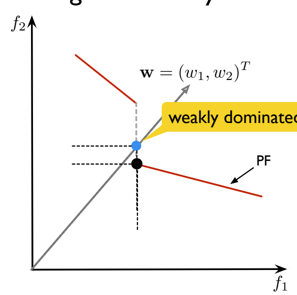
augmented scalarizing function

$$g^a(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = \max_{1\le i\le m}\left(\frac{f_i(\mathbf{x}-z_i^*}{w_i}\right) + \rho\sum_{i=1}^{m}\left(\frac{f_i(\mathbf{x}-z_i^*}{w_i}\right)$$

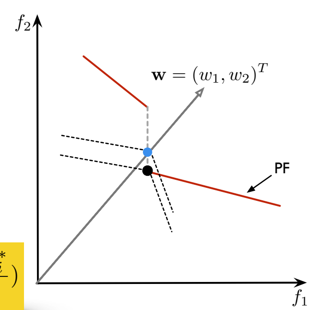
[15] K. Miettinen, et al., "Nonlinear Multiobjective Optimization", Kluwer Academic Publishers, Boston, 1999

Revisit Weighted Tchebycheff

- Weighted Tchebycheff



Drawback:
- non-smooth, weakly dominate solution [15]
- evenly distributed weights evenly do NOT lead to distributed solutions
- might easily loose diversity



weighted Tchebycheff

$$g(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = \max_{1\le i\le m} w_i|f_i(\mathbf{x}-z_i^*|$$

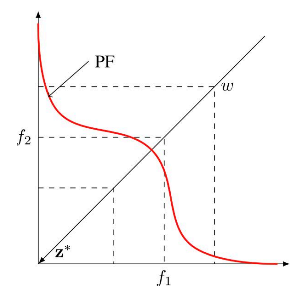
augmented scalarizing function

$$g^a(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = \max_{1\le i\le m}\left(\frac{f_i(\mathbf{x}-z_i^*}{w_i}\right) + \rho\sum_{i=1}^{m}\left(\frac{f_i(\mathbf{x}-z_i^*}{w_i}\right)$$

[15] K. Miettinen, et al., "Nonlinear Multiobjective Optimization", Kluwer Academic Publishers, Boston, 1999

Revisit Weighted Tchebycheff (cont.)

- Weighted Tchebycheff



Drawback:
- non-smooth, weakly dominate solution
- evenly distributed weights evenly do NOT lead to distributed solutions
- might easily loose diversity

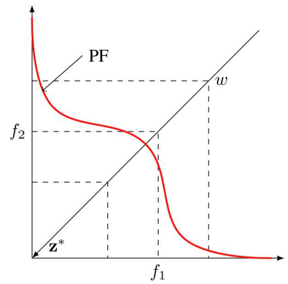
weighted Tchebycheff

$$g(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = \max_{1\le i\le m} w_i|f_i(\mathbf{x}-z_i^*|$$

[12] Y. Qi, et al., "MOEA/D with Adaptive Weight Adjustment", Evol. Comput. 22(2): 231–264, 2014.

Revisit Weighted Tchebycheff (cont.)

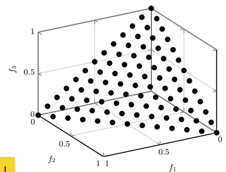
- Weighted Tchebycheff



weighted Tchebycheff

$$g(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = \max_{1\le i\le m} w_i|f_i(\mathbf{x} - z_i^*|$$

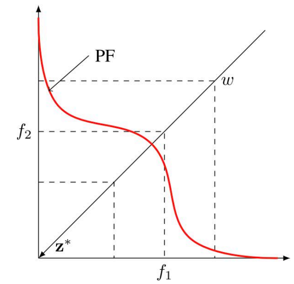
Drawback:
- non-smooth, weakly dominate solution
- evenly distributed weights evenly do NOT lead to distributed solutions
- might easily loose diversity



[12] Y. Qi, et al., "MOEA/D with Adaptive Weight Adjustment", Evol. Comput. 22(2): 231–264, 2014.

Revisit Weighted Tchebycheff (cont.)

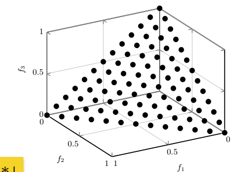
- Weighted Tchebycheff

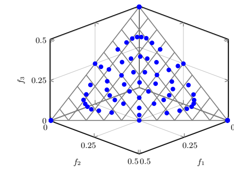


weighted Tchebycheff

$$g(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = \max_{1\le i\le m} w_i|f_i(\mathbf{x} - z_i^*|$$

Drawback:
- non-smooth, weakly dominate solution
- evenly distributed weights evenly do NOT lead to distributed solutions
- might easily loose diversity

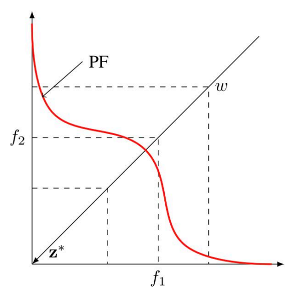




[12] Y. Qi, et al., "MOEA/D with Adaptive Weight Adjustment", Evol. Comput. 22(2): 231–264, 2014.

Revisit Weighted Tchebycheff (cont.)

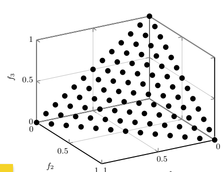
- Weighted Tchebycheff

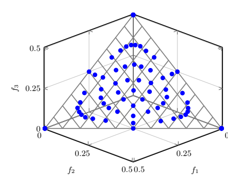


weighted Tchebycheff

$$g(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = \max_{1\le i\le m} w_i|f_i(\mathbf{x} - z_i^*|$$

Drawback:
- non-smooth, weakly dominate solution
- evenly distributed weights evenly do NOT lead to distributed solutions
- might easily loose diversity



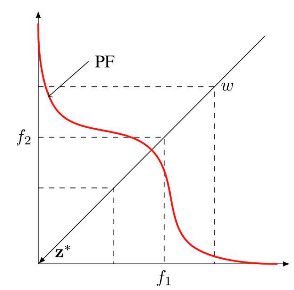


The search direction for $\mathbf{w} = (w_1, \cdots, w_m)^T$ is $\mathbf{w} = (\frac{1/w_1}{\sum_{i=1}^m 1/w_i}, \cdots, \frac{1/w_m}{\sum_{I=1}^m 1/w_i})^T$

[12] Y. Qi, et al., "MOEA/D with Adaptive Weight Adjustment", Evol. Comput. 22(2): 231–264, 2014.

Revisit Weighted Tchebycheff (cont.)

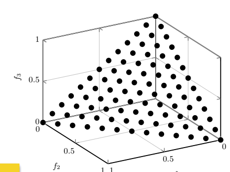
- Weighted Tchebycheff

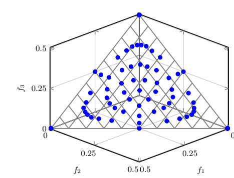


weighted Tchebycheff

$$g(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = \max_{1\le i\le m} \frac{1}{w_i}|f_i(\mathbf{x} - z_i^*|$$

Drawback:
- non-smooth, weakly dominate solution
- evenly distributed weights evenly do NOT lead to distributed solutions
- might easily loose diversity



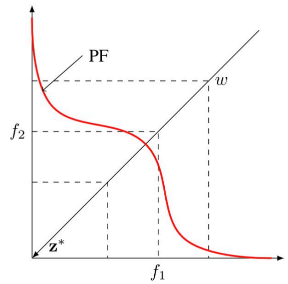


The search direction for $\mathbf{w} = (w_1, \cdots, w_m)^T$ is $\mathbf{w} = (\frac{1/w_1}{\sum_{i=1}^m 1/w_i}, \cdots, \frac{1/w_m}{\sum_{I=1}^m 1/w_i})^T$

[12] Y. Qi, et al., "MOEA/D with Adaptive Weight Adjustment", Evol. Comput. 22(2): 231–264, 2014.

Revisit Weighted Tchebycheff (cont.)

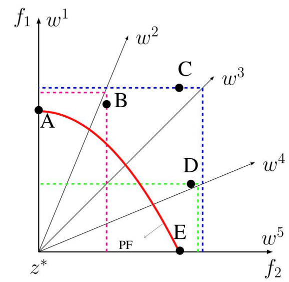
- Weighted Tchebycheff



weighted Tchebycheff

$$g(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = \max_{1\le i\le m} w_i|f_i(\mathbf{x}-z_i^*)|$$

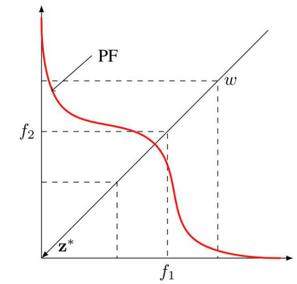
Drawback:
- non-smooth, weakly dominate solution
- evenly distributed weights evenly do NOT lead to distributed solutions
- might easily loose diversity [16]



[16] S. Jiang, et al., "Scalarizing Functions in Decomposition-Based Multiobjective Evolutionary Algorithms", IEEE Trans. Evol. Comput., 22(2): 296-313, 2018.

Revisit Weighted Tchebycheff (cont.)

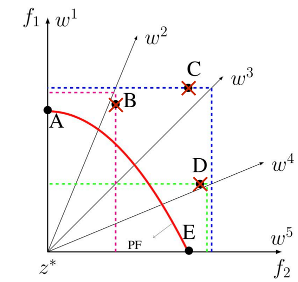
- Weighted Tchebycheff



weighted Tchebycheff

$$g(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = \max_{1\le i\le m} w_i|f_i(\mathbf{x}-z_i^*)|$$

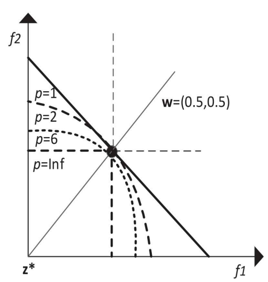
Drawback:
- non-smooth, weakly dominate solution
- evenly distributed weights evenly do NOT lead to distributed solutions
- might easily loose diversity [16]



[16] S. Jiang, et al., "Scalarizing Functions in Decomposition-Based Multiobjective Evolutionary Algorithms", IEEE Trans. Evol. Comput., 22(2): 296-313, 2018.

Revisit Weighted Tchebycheff (cont.)

- Weighted Tchebycheff



weighted Lp scalarizing [12]

$$g^{wd}(\mathbf{x}|\mathbf{w}) = (\sum_{I=1}^{m} \lambda_i (f_i(\mathbf{x}) - z_i^*)^p)^{\frac{1}{p}}$$

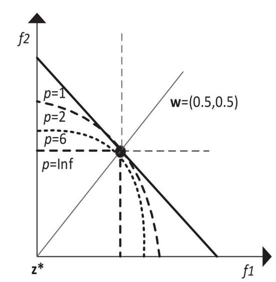
$$\lambda_i = (\frac{1}{w_i}), p \ge 1$$

Drawback:
- non-smooth, weakly dominate solution
- evenly distributed weights evenly do NOT lead to distributed solutions
- might easily loose diversity [17]

[17] R. Wang, et al., "Decomposition-Based Algorithms Using Pareto Adaptive Scalarizing Methods", IEEE Trans. Evol. Comput., 20(6): 821-837, 2016.

Revisit Weighted Tchebycheff (cont.)

- Weighted Tchebycheff



weighted Lp scalarizing [12]

$$g^{wd}(\mathbf{x}|\mathbf{w}) = (\sum_{I=1}^{m} \lambda_i (f_i(\mathbf{x}) - z_i^*)^p)^{\frac{1}{p}}$$

$$\lambda_i = (\frac{1}{w_i}), p \ge 1$$

Drawback:
- non-smooth, weakly dominate solution
- evenly distributed weights evenly do NOT lead to distributed solutions
- might easily loose diversity [17]

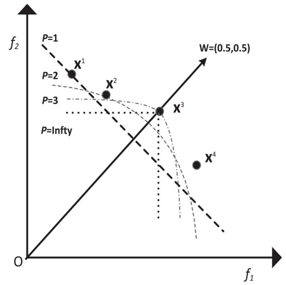
Pareto adaptive scalarizing to choose p

$$\begin{aligned}&\text{minimize} \quad p, \quad p \in P\\ &\text{subject to} \quad \forall \mathbf{x}^k : g^{wd}(\mathbf{x}^*|\mathbf{w},\mathbf{z}^*,p)\\ &\qquad\qquad \le g^{wd}(\mathbf{x}^k|\mathbf{w},\mathbf{z}^*,p)\end{aligned}$$

[17] R. Wang, et al., "Decomposition-Based Algorithms Using Pareto Adaptive Scalarizing Methods", IEEE Trans. Evol. Comput., 20(6): 821-837, 2016.

Revisit Weighted Tchebycheff (cont.)

- Weighted Tchebycheff



weighted Lp scalarizing [12]

$$g^{wd}(\mathbf{x}|\mathbf{w}) = (\sum_{I=1}^{m} \lambda_i (f_i(\mathbf{x}) - z_i^*)^p)^{\frac{1}{p}}$$

$$\lambda_i = (\frac{1}{w_i}), p \geq 1$$

Drawback:
- non-smooth, weakly dominate solution
- evenly distributed weights evenly do NOT lead to distributed solutions
- might easily loose diversity [17]

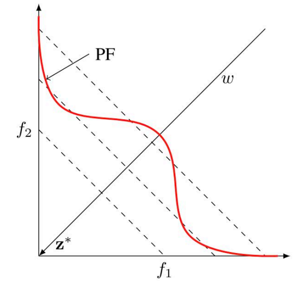
Pareto adaptive scalarizing to choose p

$$\begin{aligned} &\text{minimize} \quad p, \quad p \in P \\ &\text{subject to} \quad \forall \mathbf{x}^k : g^{wd}(\mathbf{x}^*|\mathbf{w}, \mathbf{z}^*, p) \\ &\qquad\qquad \leq g^{wd}(\mathbf{x}^k|\mathbf{w}, \mathbf{z}^*, p) \end{aligned}$$

[17] R. Wang, et al., "Decomposition-Based Algorithms Using Pareto Adaptive Scalarizing Methods", IEEE Trans. Evol. Comput., 20(6): 821-837, 2016.

Revisit Weighted Sum

- Weighted sum



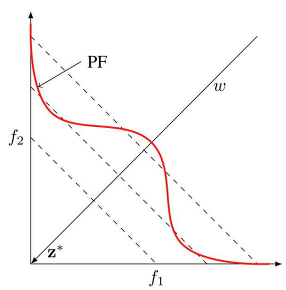
weighted sum

$$g(\mathbf{x}|\mathbf{w}) = \sum_{i=1}^{m} w_i \times f_i(\mathbf{x})$$

Drawback:
- only useful for convex PFs while not all Pareto-optimal solutions can be found if the PF is not convex.
- …

Revisit Weighted Sum

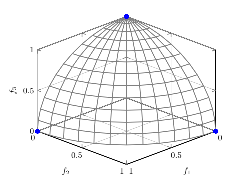
- Weighted sum



weighted sum

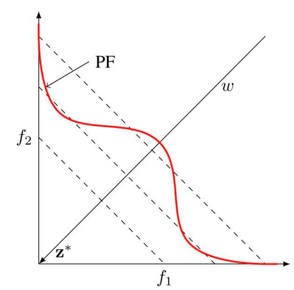
$$g(\mathbf{x}|\mathbf{w}) = \sum_{i=1}^{m} w_i \times f_i(\mathbf{x})$$

Drawback:
- only useful for convex PFs while not all Pareto-optimal solutions can be found if the PF is not convex.
- …



Revisit Weighted Sum

- Weighted sum



weighted sum

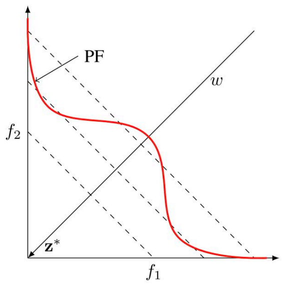
$$g(\mathbf{x}|\mathbf{w}) = \sum_{i=1}^{m} w_i \times f_i(\mathbf{x})$$

Is weighted sum really that bad?
- The superior region is constantly $1/2$, whereas it is $1/2^m$ for the Lp scalarizing
- MOEA/D with weighted sum have better convergence (given convex PF)

[18] R. Wang, et al., "Localized Weighted Sum Method for Many-Objective Optimization", IEEE Trans. Evol. Comput., 22(1): 3-18, 2018.

Revisit Weighted Sum

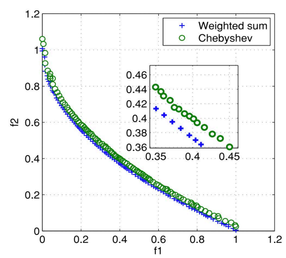
- Weighted sum



weighted sum

$$g(\mathbf{x}|\mathbf{w}) = \sum_{i=1}^{m} w_i \times f_i(\mathbf{x})$$

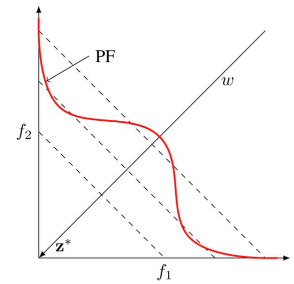
Is weighted sum really that bad?
- The superior region is constantly $1/2$, whereas it is $1/2^m$ for the Lp scalarizing
- MOEA/D with weighted sum have better convergence (given convex PF)



[18] R. Wang, et al., "Localized Weighted Sum Method for Many-Objective Optimization", IEEE Trans. Evol. Comput., 22(1): 3-18, 2018.

Revisit Weighted Sum

- Weighted sum

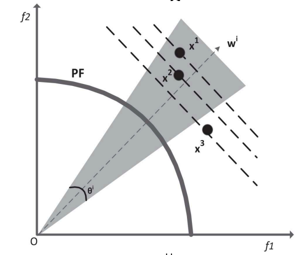


weighted sum

$$g(\mathbf{x}|\mathbf{w}) = \sum_{i=1}^{m} w_i \times f_i(\mathbf{x})$$

Is weighted sum really that bad?
- The superior region is constantly $1/2$, whereas it is $1/2^m$ for the Lp scalarizing
- MOEA/D with weighted sum have better convergence (given convex PF)

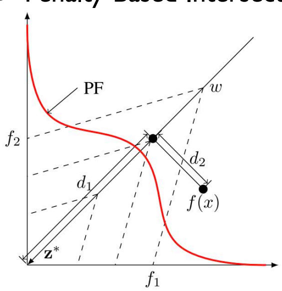
Localised weighted sum



[18] R. Wang, et al., "Localized Weighted Sum Method for Many-Objective Optimization", IEEE Trans. Evol. Comput., 22(1): 3-18, 2018.

Boundary Intersection

- Penalty-Based Intersection (PBI) [7]



$$g(\mathbf{x}|\mathbf{w}, \mathbf{z}^*) = d_1 + \theta d_2$$

$$d_1 = \frac{\|(\mathbf{F}(\mathbf{x}) - \mathbf{z}^*)^T \mathbf{w}\|}{\|\mathbf{w}\|}$$

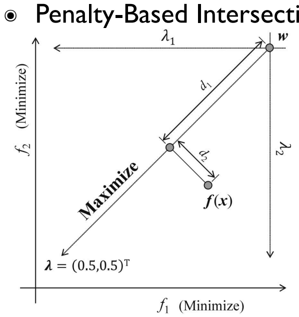
$$d_2 = \|\mathbf{F}(\mathbf{x}) - (\mathbf{z}^* + d_1 \frac{\mathbf{w}}{\|\mathbf{w}\|})\|$$

Characteristics:
- d_1 'measures' the <u>convergence</u>
  ➡can be replaced by other measure [19]
- d_2 'measures' the <u>diversity</u>
  ➡can be replaced by angle [19,20]
- θ controls the contour and trade-offs

[7] Q. Zhang and H. Li, "MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition", IEEE Trans. Evol. Comput., 11(6): 712-731, 2007.
[19] R. Cheng, et al., "A Reference Vector Guided Evolutionary Algorithm for Many-Objective Optimization", IEEE Trans. Evol. Comput., 20(5): 773-791, 2016.
[20] Y. Xiang, et al., "A Vector Angle-Based Evolutionary Algorithm for Unconstrained Many-Objective Optimization", IEEE Trans. Evol. Comput., 21(1): 131-152, 2017.
[21] H. Sato, "Analysis of inverted PBI and comparison with other scalarizing functions in decomposition based MOEAs", J. Heuristics, 21: 819-849, 2015.

Boundary Intersection

- Penalty-Based Intersection (PBI) [7]



Inverted PBI [21]

$$g(\mathbf{x}|\mathbf{w}, \mathbf{z}^*) = d_1 + \theta d_2$$

$$d_1 = \frac{\|(\mathbf{F}(\mathbf{x}) - \mathbf{z}^*)^T \mathbf{w}\|}{\|\mathbf{w}\|}$$

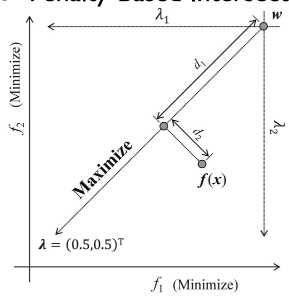
$$d_2 = \|\mathbf{F}(\mathbf{x}) - (\mathbf{z}^* + d_1 \frac{\mathbf{w}}{\|\mathbf{w}\|})\|$$

Characteristics:
- d_1 'measures' the <u>convergence</u>
  ➡can be replaced by other measure [19]
- d_2 'measures' the <u>diversity</u>
  ➡can be replaced by angle [19,20]
- θ controls the contour and trade-offs

[7] Q. Zhang and H. Li, "MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition", IEEE Trans. Evol. Comput., 11(6): 712-731, 2007.
[19] R. Cheng, et al., "A Reference Vector Guided Evolutionary Algorithm for Many-Objective Optimization", IEEE Trans. Evol. Comput., 20(5): 773-791, 2016.
[20] Y. Xiang, et al., "A Vector Angle-Based Evolutionary Algorithm for Unconstrained Many-Objective Optimization", IEEE Trans. Evol. Comput., 21(1): 131-152, 2017.
[21] H. Sato, "Analysis of inverted PBI and comparison with other scalarizing functions in decomposition based MOEAs", J. Heuristics, 21: 819-849, 2015.

Boundary Intersection

- Penalty-Based Intersection (PBI) [7]



$$g(\mathbf{x}|\mathbf{w},\mathbf{z}^{nad}) = d_1 - \theta d_2$$

$$d_1 = \frac{\|(\mathbf{F}(\mathbf{x}) - \mathbf{z}^{nad})^T\mathbf{w}\|}{\|\mathbf{w}\|}$$

$$d_2 = \|\mathbf{F}(\mathbf{x}) - (\mathbf{z}^{nad} + d_1\frac{\mathbf{w}}{\|\mathbf{w}\|})\|$$

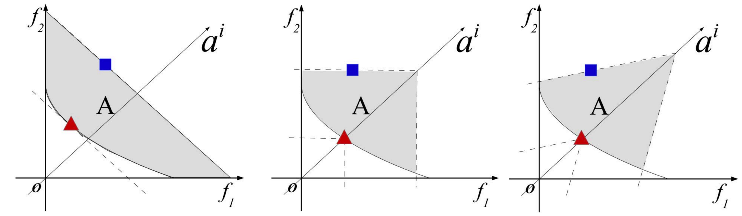
Inverted PBI [21]

Characteristics:
- d_1 'measures' the convergence
  ➡can be replaced by other measure [19]
- d_2 'measures' the diversity
  ➡can be replaced by angle [19,20]
- θ controls the contour and trade-offs

[7] Q. Zhang and H. Li, "MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition", IEEE Trans. Evol. Comput., 11(6): 712-731, 2007.
[19] R. Cheng, et al., "A Reference Vector Guided Evolutionary Algorithm for Many-Objective Optimization", IEEE Trans. Evol. Comput., 20(5): 773-791, 2016.
[20] Y. Xiang, et al., "A Vector Angle-Based Evolutionary Algorithm for Unconstrained Many-Objective Optimization", IEEE Trans. Evol. Comput., 21(1): 131-152, 2017.
[21] H. Sato, "Analysis of inverted PBI and comparison with other scalarizing functions in decomposition based MOEAs", J. Heuristics, 21: 819-849, 2015.

Constrained Decomposition

- The improvement region of WS, TCH and PBI is too large
  - Gives a solution large chance to update many agents: hazard to diversity



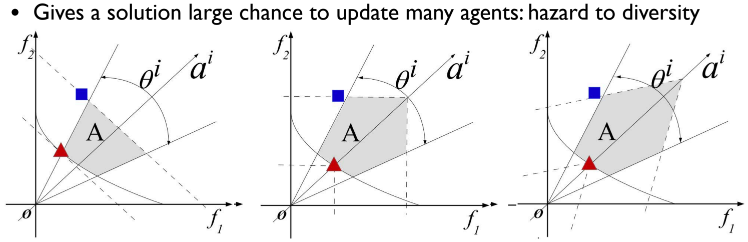
  - Add a constraint to the subproblem to reduce the improvement region [22]

$$\begin{array}{ll} \text{minimize} & g(\mathbf{x}|\mathbf{w},\mathbf{z}^*) \\ \text{subject to} & \langle \mathbf{a}^i, \mathbf{F}(\mathbf{x}) - \mathbf{z}^* \rangle \leq 0.5\theta^i \end{array}$$

[22] L. Wang, et al., "Constrained Subproblems in a Decomposition-Based Multiobjective Evolutionary Algorithm", IEEE Trans. Evol. Comput., 20(3): 475-480, 2016.

Constrained Decomposition

- The improvement region of WS, TCH and PBI is too large
  - Gives a solution large chance to update many agents: hazard to diversity



  - Add a constraint to the subproblem to reduce the improvement region [22]

$$\begin{array}{ll} \text{minimize} & g(\mathbf{x}|\mathbf{w},\mathbf{z}^*) \\ \text{subject to} & \langle \mathbf{a}^i, \mathbf{F}(\mathbf{x}) - \mathbf{z}^* \rangle \leq 0.5\theta^i \end{array}$$

[22] L. Wang, et al., "Constrained Subproblems in a Decomposition-Based Multiobjective Evolutionary Algorithm", IEEE Trans. Evol. Comput., 20(3): 475-480, 2016.

Subproblem Can Be Multi-Objective …

- MOP to MOP (M2M)
  - Decompose a MOP into K ($K > 1$) constrained MOPs [23].

$$\begin{array}{ll} \text{minimize} & \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), \cdots, f_m(\mathbf{x}))^T \\ \text{subject to} & \mathbf{x} \in \Omega \end{array}$$

[23] H. Liu, et al., "Decomposition of a Multiobjective Optimization Problem Into a Number of Simple Multiobjective Subproblems", IEEE Trans. Evol. Comput., 18(3): 450-455, 2014.

Subproblem Can Be Multi-Objective …

- MOP to MOP (M2M)
  - Decompose a MOP into K $(K > 1)$ constrained MOPs [23].

$$\begin{array}{ll} \text{minimize} & \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), \cdots, f_m(\mathbf{x}))^T \\ \text{subject to} & \mathbf{x} \in \Omega \end{array} \Rightarrow \begin{array}{ll} \text{minimize} & \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), \cdots, f_m(\mathbf{x}))^T \\ \text{subject to} & \mathbf{x} \in \Omega \\ & \mathbf{F}(\mathbf{x}) \in \Omega_k \end{array}$$

[23] H. Liu, et al., "Decomposition of a Multiobjective Optimization Problem Into a Number of Simple Multiobjective Subproblems", IEEE Trans. Evol. Comput., 18(3): 450-455, 2014.

Subproblem Can Be Multi-Objective …

- MOP to MOP (M2M)
  - Decompose a MOP into K $(K > 1)$ constrained MOPs [23].

$$\begin{array}{ll} \text{minimize} & \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), \cdots, f_m(\mathbf{x}))^T \\ \text{subject to} & \mathbf{x} \in \Omega \end{array} \Rightarrow \begin{array}{ll} \text{minimize} & \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), \cdots, f_m(\mathbf{x}))^T \\ \text{subject to} & \mathbf{x} \in \Omega \\ & \mathbf{F}(\mathbf{x}) \in \Omega_k \end{array}$$

$$\Omega_k = \{\mathbf{F}(\mathbf{x}) \in \mathbb{R}^m | \langle \mathbf{F}(\mathbf{x}), \mathbf{w}^i \rangle \leq \langle \mathbf{F}(\mathbf{x}), \mathbf{w}^j \rangle \text{ for any } j = 1, \cdots, K\}$$

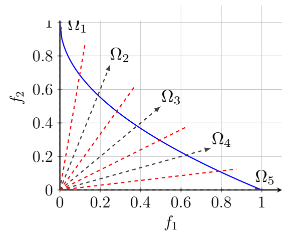
[23] H. Liu, et al., "Decomposition of a Multiobjective Optimization Problem Into a Number of Simple Multiobjective Subproblems", IEEE Trans. Evol. Comput., 18(3): 450-455, 2014.

Subproblem Can Be Multi-Objective …

- MOP to MOP (M2M)
  - Decompose a MOP into K $(K > 1)$ constrained MOPs [23].

$$\begin{array}{ll} \text{minimize} & \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), \cdots, f_m(\mathbf{x}))^T \\ \text{subject to} & \mathbf{x} \in \Omega \end{array} \Rightarrow \begin{array}{ll} \text{minimize} & \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), \cdots, f_m(\mathbf{x}))^T \\ \text{subject to} & \mathbf{x} \in \Omega \\ & \mathbf{F}(\mathbf{x}) \in \Omega_k \end{array}$$

$$\Omega_k = \{\mathbf{F}(\mathbf{x}) \in \mathbb{R}^m | \langle \mathbf{F}(\mathbf{x}), \mathbf{w}^i \rangle \leq \langle \mathbf{F}(\mathbf{x}), \mathbf{w}^j \rangle \text{ for any } j = 1, \cdots, K\}$$



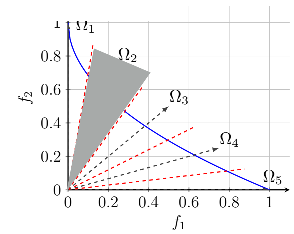
[23] H. Liu, et al., "Decomposition of a Multiobjective Optimization Problem Into a Number of Simple Multiobjective Subproblems", IEEE Trans. Evol. Comput., 18(3): 450-455, 2014.

Subproblem Can Be Multi-Objective …

- MOP to MOP (M2M)
  - Decompose a MOP into K $(K > 1)$ constrained MOPs [23].

$$\begin{array}{ll} \text{minimize} & \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), \cdots, f_m(\mathbf{x}))^T \\ \text{subject to} & \mathbf{x} \in \Omega \end{array} \Rightarrow \begin{array}{ll} \text{minimize} & \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), \cdots, f_m(\mathbf{x}))^T \\ \text{subject to} & \mathbf{x} \in \Omega \\ & \mathbf{F}(\mathbf{x}) \in \Omega_k \end{array}$$

$$\Omega_k = \{\mathbf{F}(\mathbf{x}) \in \mathbb{R}^m | \langle \mathbf{F}(\mathbf{x}), \mathbf{w}^i \rangle \leq \langle \mathbf{F}(\mathbf{x}), \mathbf{w}^j \rangle \text{ for any } j = 1, \cdots, K\}$$



[23] H. Liu, et al., "Decomposition of a Multiobjective Optimization Problem Into a Number of Simple Multiobjective Subproblems", IEEE Trans. Evol. Comput., 18(3): 450-455, 2014.

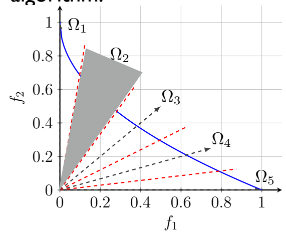
Subproblem Can Be Multi-Objective …

- MOP to MOP (M2M)
  - Decompose a MOP into K ($K > 1$) constrained MOPs [23].

$$\begin{array}{ll}\text{minimize} & \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), \cdots, f_m(\mathbf{x}))^T \\ \text{subject to} & \mathbf{x} \in \Omega\end{array} \Rightarrow \begin{array}{ll}\text{minimize} & \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), \cdots, f_m(\mathbf{x}))^T \\ \text{subject to} & \mathbf{x} \in \Omega \\ & \mathbf{F}(\mathbf{x}) \in \Omega_k\end{array}$$

$$\Omega_k = \{\mathbf{F}(\mathbf{x}) \in \mathbb{R}^m | \langle \mathbf{F}(\mathbf{x}), \mathbf{w}^i \rangle \leq \langle \mathbf{F}(\mathbf{x}), \mathbf{w}^j \rangle \text{ for any } j = 1, \cdots, K\}$$

  - Each agent is an EMO algorithm.



[23] H. Liu, et al., "Decomposition of a Multiobjective Optimization Problem Into a Number of Simple Multiobjective Subproblems", IEEE Trans. Evol. Comput., 18(3): 450-455, 2014.

Dynamic Resource Allocation

- Are all subproblems equally important?
  - Some regions in the PF/PS are easier than the others.
  - Different agents require different amounts of computational resources.
- Dynamic resource allocation (DRA) in MOEA/D [24]
  - Utility function to measure the likelihood of improvement
    - e.g. fitness improvement over ΔT

$$u^I = \frac{g^i(\mathbf{x}^{\mathbf{i}}_{\mathbf{t}-\mathbf{\Delta T}}) - g^I(\mathbf{x}^i_t)}{g^i(\mathbf{x}^{\mathbf{i}}_{\mathbf{t}-\mathbf{\Delta T}})}$$

  - Allocation mechanism
    - e.g. probability of improvement

$$p^i = \frac{u^i + \epsilon}{\max_{j=1,\cdots,N}\{u^j\} + \epsilon}$$

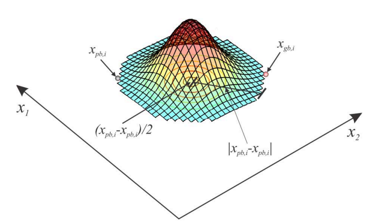
[24] A. Zhou, et al., "Are All the Subproblems Equally Important? Resource Allocation in Decomposition-Based Multiobjective Evolutionary Algorithms", IEEE TEVC, 20(1): 52-64, 2016.

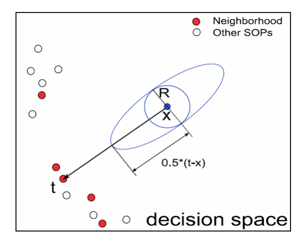
Outline

- Why Multi-Objective Optimisation Important?
- Basic Concepts
- Oracle MOEA/D
- Current Developments
  - Decomposition methods
  - Search methods
  - Collaboration
    - Mating selection
    - Replacement
- Resources
- Future Directions

Search Methods

- Offspring reproduction in MOEA/D
  - Neighbourhood defines where to find mating parents
  - Any genetic operator can be used
    - GA [7], DE [25], PSO [26], guided mutation [27], …

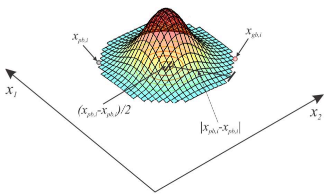


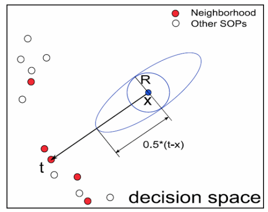


[7] Q. Zhang et al., "MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition", IEEE Trans. Evol. Comput., 11(6): 712-731, 2007.
[25] H. Li and Q. Zhang, "Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II", IEEE Trans. Evol. Comput., 13(2): 284-302, 2009.
[26] S. Martínez, et al., "A multi-objective PSO based on decomposition, in GECCO 2011.
[27] C. Chen, et al., "Enhancing MOEA/D with guided mutation and priority update for multi-objective optimization", CEC 2009

Search Methods

- Offspring reproduction in MOEA/D
  - Neighbourhood defines where to find mating parents
  - Any genetic operator can be used
  - Any local search can be used
    - simulated annealing [28], interpolation [29], tabu search [30], GRASP [31], Nelder-Mead [32], …



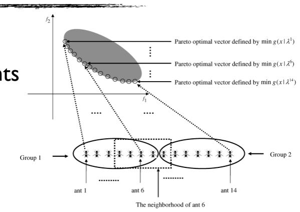


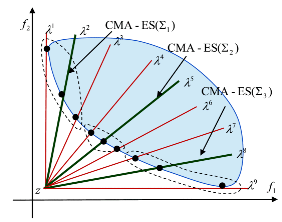
[28] H. Li, et al., "An adaptive evolutionary multi-objective approach based on simulated annealing", Evol. Comput. 19(4): 561-595, 2011.
[29] K. Sindhya, "A new hybrid mutation operator for multiobjective optimization with differential evolution", Soft Comput., 15:2041–2055, 2011.
[30] A. Alhindi, et al., "Hybridisation of decomposition and GRASP for combinatorial multiobjective optimisation", UKCI 2014.
[31] A. Alhindi, et al., "MOEA/D with Tabu Search for multiobjective permutation flow shop scheduling problems", CEC 2014.
[32] H. Zhang, et al., "Accelerating MOEA/D by Nelder-Mead method", CEC 2017.

38

Search Methods

- Offspring reproduction in MOEA/D
  - Neighbourhood defines where to find mating parents
  - Any genetic operator can be used
  - Any local search can be used
  - Probabilistic model can be used
    - Memory
      - Each agent records historical information, i.e. elites
    - Model building and solution construction
      - Each agent can build 'local model', e.g. ACO [33], EDA [34], cross entropy [35], graphical model [36], CMA-ES [37], based on memory of itself and its neighbour
      - New solutions are sampled from these models
      - NOTE: too many models may be too expensive
    - Memory update
      - Offspring update each agent's and its neighbour's memory

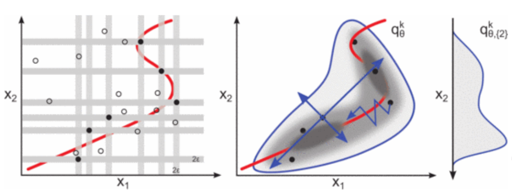


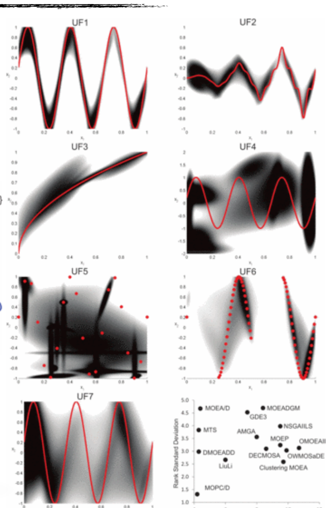


[33] L. Ke, et al., "MOEA/D-ACO: A Multiobjective Evolutionary Algorithm Using Decomposition and Ant Colony", IEEE Trans. Cybern., 43(6): 1845-1859, 2013.
[34] A. Zhou, et al., "A Decomposition based Estimation of Distribution Algorithm for Multiobjective Traveling Salesman Problems", Computers & Mathematics with Applications, 66(10): 1857-1868, 2013.
[35] I. Giagkiozis, et al., "Generalized decomposition and cross entropy methods for many-objective optimization", Inf. Sci., 282: 363-387, 2014.
[36] M. de Souza, et al., "MOEA/D-GM: Using probabilistic graphical models in MOEA/D for solving combinatorial optimization problems", arXiv:1511.05625, 2015.
[37] H. Li, et al., "Biased Multiobjective Optimization and Decomposition Algorithm", IEEE Trans. Cybern., 47(1): 52-66, 2016.

Search Methods (cont.)

- Using Probability Collective in MOEA/D
  - Instead of a point-based search, probability collective aims to fit a probability distribution highly peaked around the neighbourhood of PS





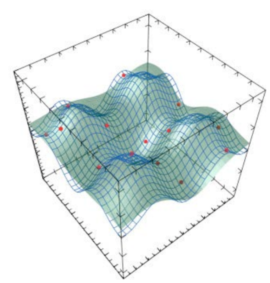
- Fit a Gaussian mixture model using solutions associated with each subproblem
- Search is based one sampling or local search upon the fitted model

[38] D. Morgan, et al., "MOPC/D: A new probability collectives algorithm for multiobjective optimisation", MCDM'13, 17-24, 2013

39

Search Methods (cont.)

- Expensive optimisation
  - Building surrogate model for expensive objective function
    - e.g. Gaussian process (Kriging) [39, 40], RBF [41], …

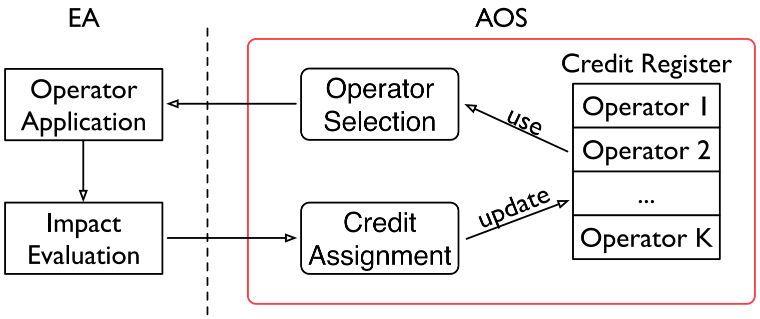


[39] Q. Zhang, et al., "Expensive Multiobjective Optimization by MOEA/D with Gaussian Process Model", IEEE Trans. Evol. Comput., 14(3): 456-474, 2010.
[40] T. Chugh, et al., "A Surrogate-Assisted Reference Vector Guided Evolutionary Algorithm for Computationally Expensive Many-Objective Optimization", 22(1): 129-142, 2018.
[41] S. Martínez, et al., "MOEA/D assisted by RBF Networks for Expensive Multi-Objective Optimization Problems", GECCO 2013.

41

Search Methods (cont.)

- Adaptive operator selection as a multi-armed bandits [39]
  - Strike the balance between the exploration and exploitation
    - Exploration: acquire new information (diversity)
    - Exploitation: capitalise on the available knowledge (convergence)



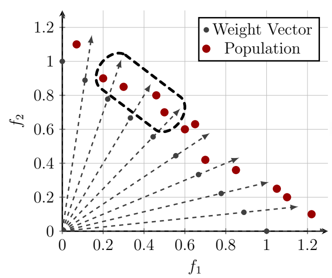
[39] K. Li, et al., "Adaptive operator selection with bandits for multiobjective evolutionary algorithm based on decomposition", IEEE Trans. Evol. Comput., 18(1): 114-130, 2014.

Outline

- Why Multi-Objective Optimisation Important?
- Basic Concepts
- Oracle MOEA/D
- Current Developments
  - Decomposition methods
  - Search methods
  - Collaboration
    - Mating selection
    - Replacement
- Resources
- Future Directions

Mating Selection

- Mating selection: how to select parents for offspring reproduction?
  - Tournament selection, genotype neighbours, …
  - MOEA/Ds leverage the neighbourhood structure of weight vectors
    - Assumption: neighbouring subproblems have similar structure
    - Select mating parents purely from neighbouring agents (simple MOEA/D)

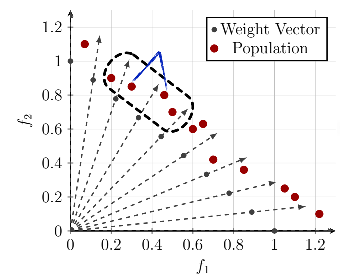


- Focusing on the neighbourhood is too much exploited
- Give some chance to explore in the whole population [25]

[25] H. Li and Q. Zhang, "Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II", IEEE Trans. Evol. Comput., 13(2): 284-302, 2009.

Mating Selection

- Mating selection: how to select parents for offspring reproduction?
  - Tournament selection, genotype neighbours, …
  - MOEA/Ds leverage the neighbourhood structure of weight vectors
    - Assumption: neighbouring subproblems have similar structure
    - Select mating parents purely from neighbouring agents (simple MOEA/D)

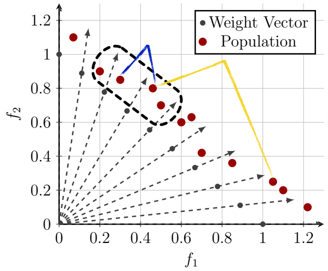


- Focusing on the neighbourhood is too much exploited
- Give some chance to explore in the whole population [25]

[25] H. Li and Q. Zhang, "Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II", IEEE Trans. Evol. Comput., 13(2): 284-302, 2009.

Mating Selection

- Mating selection: how to select parents for offspring reproduction?
  - Tournament selection, genotype neighbours, …
  - MOEA/Ds leverage the neighbourhood structure of weight vectors
    - Assumption: neighbouring subproblems have similar structure
    - Select mating parents purely from neighbouring agents (simple MOEA/D)

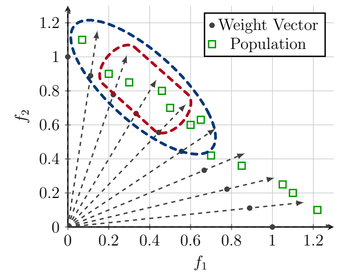


- Focusing on the neighbourhood is too much exploited
- Give some chance to explore in the whole population [25]

[25] H. Li and Q. Zhang, "Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II", IEEE Trans. Evol. Comput., 13(2): 284-302, 2009.

Mating Selection (cont.)

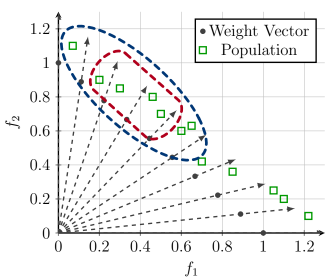
- Mating selection: how to select parents for offspring reproduction?
  - Tournament selection, genotype neighbours, …
  - MOEA/Ds leverage the neighbourhood structure of weight vectors
    - Assumption: neighbouring subproblems have similar structure
    - Select mating parents purely from neighbouring agents (simple MOEA/D)



[27] S. Zhao, et al., "Decomposition-Based Multiobjective Evolutionary Algorithm With an Ensemble of Neighborhood Sizes", IEEE Trans. Evol. Comput., 16(3): 442-446, 2013.

Mating Selection (cont.)

- Mating selection: how to select parents for offspring reproduction?
  - Tournament selection, genotype neighbours, …
  - MOEA/Ds leverage the neighbourhood structure of weight vectors
    - Assumption: neighbouring subproblems have similar structure
    - Select mating parents purely from neighbouring agents (simple MOEA/D)

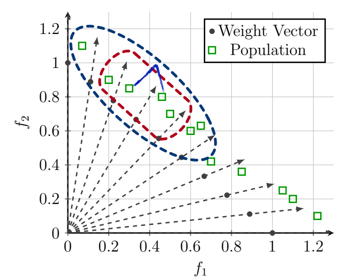


Effects of neighbourhood size (NS)
- Large neighbourhood makes the search globally
- Small neighbourhood encourages local search

[27] S. Zhao, et al., "Decomposition-Based Multiobjective Evolutionary Algorithm With an Ensemble of Neighborhood Sizes", IEEE Trans. Evol. Comput., 16(3): 442-446, 2013.

Mating Selection (cont.)

- Mating selection: how to select parents for offspring reproduction?
  - Tournament selection, genotype neighbours, …
  - MOEA/Ds leverage the neighbourhood structure of weight vectors
    - Assumption: neighbouring subproblems have similar structure
    - Select mating parents purely from neighbouring agents (simple MOEA/D)

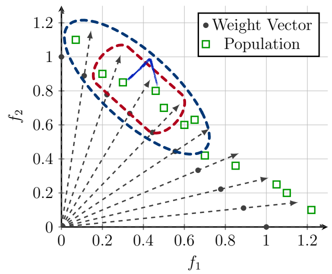


Effects of neighbourhood size (NS)
- Large neighbourhood makes the search globally
- Small neighbourhood encourages local search

[27] S. Zhao, et al., "Decomposition-Based Multiobjective Evolutionary Algorithm With an Ensemble of Neighborhood Sizes", IEEE Trans. Evol. Comput., 16(3): 442-446, 2013.

Mating Selection (cont.)

- Mating selection: how to select parents for offspring reproduction?
  - Tournament selection, genotype neighbours, …
  - MOEA/Ds leverage the neighbourhood structure of weight vectors
    - Assumption: neighbouring subproblems have similar structure
    - Select mating parents purely from neighbouring agents (simple MOEA/D)



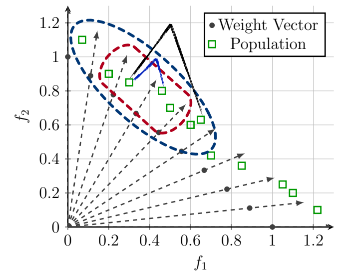
Effects of neighbourhood size (NS)
- Large neighbourhood makes the search globally
- Small neighbourhood encourages local search

Build an ensemble of neighbourhood sizes and chooses the appropriate one based on their historical performance. [27]

[27] S. Zhao, et al., "Decomposition-Based Multiobjective Evolutionary Algorithm With an Ensemble of Neighborhood Sizes", IEEE Trans. Evol. Comput., 16(3): 442-446, 2013.

Mating Selection (cont.)

- Mating selection: how to select parents for offspring reproduction?
  - Tournament selection, genotype neighbours, …
  - MOEA/Ds leverage the neighbourhood structure of weight vectors
    - Assumption: neighbouring subproblems have similar structure
    - Select mating parents purely from neighbouring agents (simple MOEA/D)



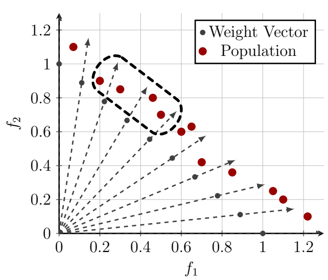
Effects of neighbourhood size (NS)
- Large neighbourhood makes the search globally
- Small neighbourhood encourages local search

Build an ensemble of neighbourhood sizes and chooses the appropriate one based on their historical performance. [27]

[27] S. Zhao, et al., "Decomposition-Based Multiobjective Evolutionary Algorithm With an Ensemble of Neighborhood Sizes", IEEE Trans. Evol. Comput., 16(3): 442-446, 2013.

Mating Selection (cont.)

- Mating selection: how to select parents for offspring reproduction?
  - Tournament selection, genotype neighbours, …
  - MOEA/Ds leverage the neighbourhood structure of weight vectors
    - Assumption: neighbouring subproblems have similar structure
    - Select mating parents purely from neighbouring agents (simple MOEA/D)

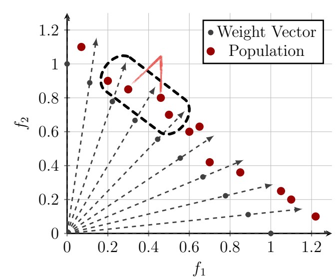


Take crowdedness into consideration [28]
- Compute the niche count of each solution within agent i's neighbour
- Select mating parents from outside of the neighbour if solutions are overly crowded

[28] S. Jiang, et al., "An improved multiobjective optimization evolutionary algorithm based on decomposition for complex Pareto fronts", IEEE Trans. Cybern, 46(2): 421-437, 2016.

Mating Selection (cont.)

- Mating selection: how to select parents for offspring reproduction?
  - Tournament selection, genotype neighbours, …
  - MOEA/Ds leverage the neighbourhood structure of weight vectors
    - Assumption: neighbouring subproblems have similar structure
    - Select mating parents purely from neighbouring agents (simple MOEA/D)

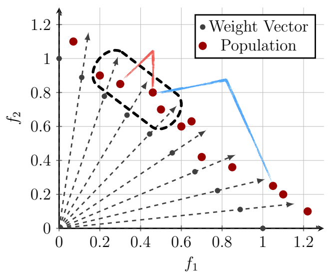


Take crowdedness into consideration [28]
- Compute the niche count of each solution within agent i's neighbour
- Select mating parents from outside of the neighbour if solutions are overly crowded

[28] S. Jiang, et al., "An improved multiobjective optimization evolutionary algorithm based on decomposition for complex Pareto fronts", IEEE Trans. Cybern, 46(2): 421-437, 2016.

Mating Selection (cont.)

- Mating selection: how to select parents for offspring reproduction?
  - Tournament selection, genotype neighbours, …
  - MOEA/Ds leverage the neighbourhood structure of weight vectors
    - Assumption: neighbouring subproblems have similar structure
    - Select mating parents purely from neighbouring agents (simple MOEA/D)



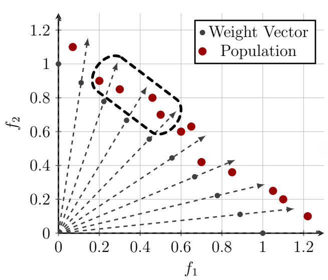
Take crowdedness into consideration [28]
- Compute the niche count of each solution within agent i's neighbour
- Select mating parents from outside of the neighbour if solutions are overly crowded

[28] S. Jiang, et al., "An improved multiobjective optimization evolutionary algorithm based on decomposition for complex Pareto fronts", IEEE Trans. Cybern, 46(2): 421-437, 2016.

Outline

Replacement

- Replacement: update the parent population
  - Steady-state evolution model (oracle MOEA/D)
  - Update as many neighbouring subproblems as it can (oracle MOEA/D)

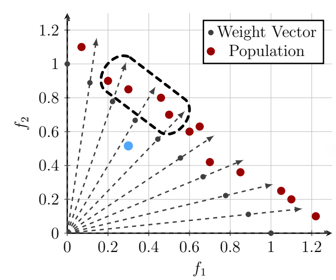


- The replacement strategy of the oracle MOEA/D is too greedy
- Offspring is only allowed to replace a limited number of parents [26]
  - Pros: Good for diversity
  - Cons: convergence may be slow

[26] H. Li and Q. Zhang, "Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II", IEEE Trans. Evol. Comput., 13(2): 284-302, 2009.

Replacement

- Replacement: update the parent population
  - Steady-state evolution model (oracle MOEA/D)
  - Update as many neighbouring subproblems as it can (oracle MOEA/D)



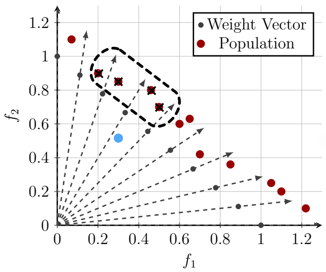
- The replacement strategy of the oracle MOEA/D is too greedy
- Offspring is only allowed to replace a limited number of parents [26]
  - Pros: Good for diversity
  - Cons: convergence may be slow

[26] H. Li and Q. Zhang, "Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II", IEEE Trans. Evol. Comput., 13(2): 284-302, 2009.

48

Replacement

- Replacement: update the parent population
  - Steady-state evolution model (oracle MOEA/D)
  - Update as many neighbouring subproblems as it can (oracle MOEA/D)

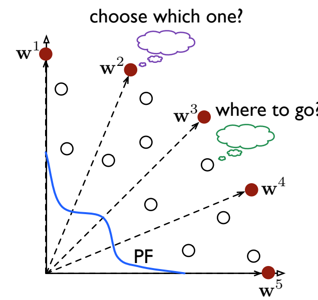


- The replacement strategy of the oracle MOEA/D is too greedy
- Offspring is only allowed to replace a limited number of parents [26]
  - Pros: Good for diversity
  - Cons: convergence may be slow

[26] H. Li and Q. Zhang, "Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II", IEEE Trans. Evol. Comput., 13(2): 284-302, 2009.

Replacement (cont.)

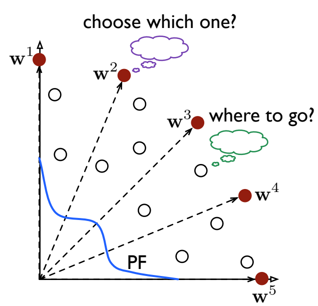
- Matching-based selection [29,30]
  - Subproblems and solutions are two sets of agents
  - Subproblems 'prefer' convergence, solutions 'prefer' diversity



[29] K. Li, et al., "Stable Matching Based Selection in Evolutionary Multiobjective Optimization", IEEE Trans. Evol. Comput., 18(6): 909–923, 2014.
[30] M. Wu, et al., "Matching-Based Selection with Incomplete Lists for Decomposition Multi-Objective Optimization", IEEE Trans. Evol. Comput., 21(4): 554–568, 2017.

Replacement (cont.)

- Matching-based selection [29,30]
  - Subproblems and solutions are two sets of agents
  - Subproblems 'prefer' convergence, solutions 'prefer' diversity

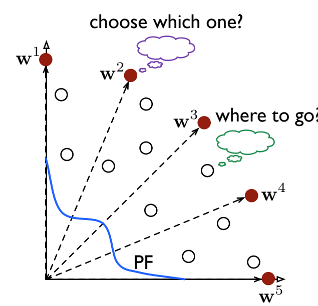


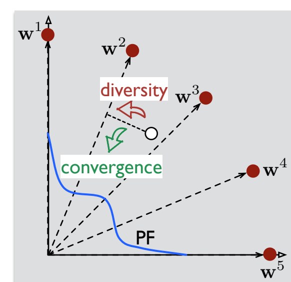
selection — matching

[29] K. Li, et al., "Stable Matching Based Selection in Evolutionary Multiobjective Optimization", IEEE Trans. Evol. Comput., 18(6): 909–923, 2014.
[30] M. Wu, et al., "Matching-Based Selection with Incomplete Lists for Decomposition Multi-Objective Optimization", IEEE Trans. Evol. Comput., 21(4): 554–568, 2017.

Replacement (cont.)

- Matching-based selection [29,30]
  - Subproblems and solutions are two sets of agents
  - Subproblems 'prefer' convergence, solutions 'prefer' diversity



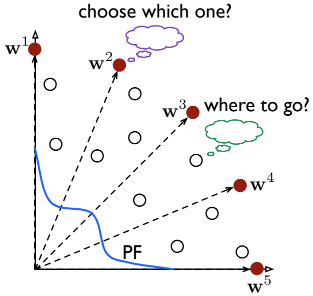


selection — matching

[29] K. Li, et al., "Stable Matching Based Selection in Evolutionary Multiobjective Optimization", IEEE Trans. Evol. Comput., 18(6): 909–923, 2014.
[30] M. Wu, et al., "Matching-Based Selection with Incomplete Lists for Decomposition Multi-Objective Optimization", IEEE Trans. Evol. Comput., 21(4): 554–568, 2017.

Replacement (cont.)

- Matching-based selection [29,30]
  - Subproblems and solutions are two sets of agents
  - Subproblems 'prefer' convergence, solutions 'prefer' diversity

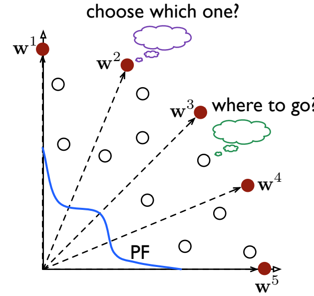


selection — matching

[29] K. Li, et al., "Stable Matching Based Selection in Evolutionary Multiobjective Optimization", IEEE Trans. Evol. Comput., 18(6): 909–923, 2014.
[30] M. Wu, et al., "Matching-Based Selection with Incomplete Lists for Decomposition Multi-Objective Optimization", IEEE Trans. Evol. Comput., 21(4): 554–568, 2017.

Replacement (cont.)

- Matching-based selection [29,30]
  - Subproblems and solutions are two sets of agents
  - Subproblems 'prefer' convergence, solutions 'prefer' diversity



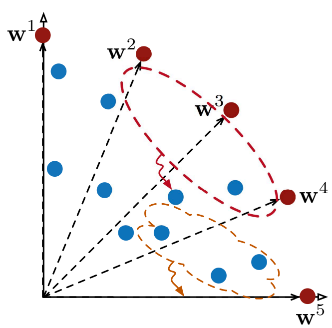
- A unified perspective to look at selection
- A generational evolution model for MOEA/D
  - ✓ What is convergence?
    - ➡ Aggregation function, …
  - ✓ What is diversity?
    - ➡ Perpendicular distance, angle …
  - ✓ Mechanism to match
    - ➡ Stable matching, …

selection — matching

[29] K. Li, et al., "Stable Matching Based Selection in Evolutionary Multiobjective Optimization", IEEE Trans. Evol. Comput., 18(6): 909–923, 2014.
[30] M. Wu, et al., "Matching-Based Selection with Incomplete Lists for Decomposition Multi-Objective Optimization", IEEE Trans. Evol. Comput., 21(4): 554–568, 2017.

Replacement (cont.)

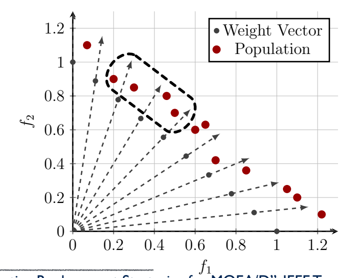
- Matching-based selection (extension) [31]
  - Identify the inter-relationship between subproblems and solutions
    - Find the related subproblems to each solution (e.g. fitness)
    - Find the related solutions for each subproblem (e.g. closeness)
  - Selection mechanism: each subproblem chooses its favourite solution



[31] K. Li, et al., "Interrelationship-based selection for decomposition multiobjective optimization", IEEE Trans. Cybern. 45(10): 2076–2088, 2015.

Replacement (cont.)

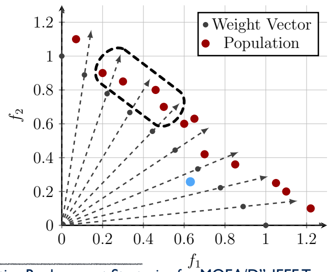
- Matching-based selection (extension):
  - Global replacement [32]
    - If the newly generated offspring is way beyond the current neighbourhood …
    - Find the 'best agent' (i.e. subproblem) for the newly generated offspring
    - Compete with solutions associated with this 'best agent'
  - MOEA/D-DU [33]
    - Update the newly generated offspring's 'nearest' subproblems



[32] Z. Wang, et al., "Adaptive Replacement Strategies for MOEA/D", IEEE Trans. Cybern., 46(2): 474-486, 2016.
[33] Y. Yuan, et al., "Balancing Convergence and Diversity in Decomposition-Based Many-Objective Optimizers", IEEE Trans. Evol. Comput., 20(2): 180-198, 2016.

Replacement (cont.)

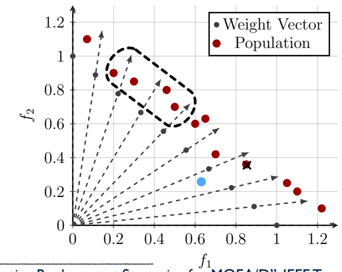
- Matching-based selection (extension):
  - Global replacement [32]
    - If the newly generated offspring is way beyond the current neighbourhood …
    - Find the 'best agent' (i.e. subproblem) for the newly generated offspring
    - Compete with solutions associated with this 'best agent'
  - MOEA/D-DU [33]
    - Update the newly generated offspring's 'nearest' subproblems



[32] Z. Wang, et al., "Adaptive Replacement Strategies for MOEA/D", IEEE Trans. Cybern., 46(2): 474-486, 2016.
[33] Y. Yuan, et al., "Balancing Convergence and Diversity in Decomposition-Based Many-Objective Optimizers", IEEE Trans. Evol. Comput., 20(2): 180-198, 2016.

Replacement (cont.)

- Matching-based selection (extension):
  - Global replacement [32]
    - If the newly generated offspring is way beyond the current neighbourhood …
    - Find the 'best agent' (i.e. subproblem) for the newly generated offspring
    - Compete with solutions associated with this 'best agent'
  - MOEA/D-DU [33]
    - Update the newly generated offspring's 'nearest' subproblems



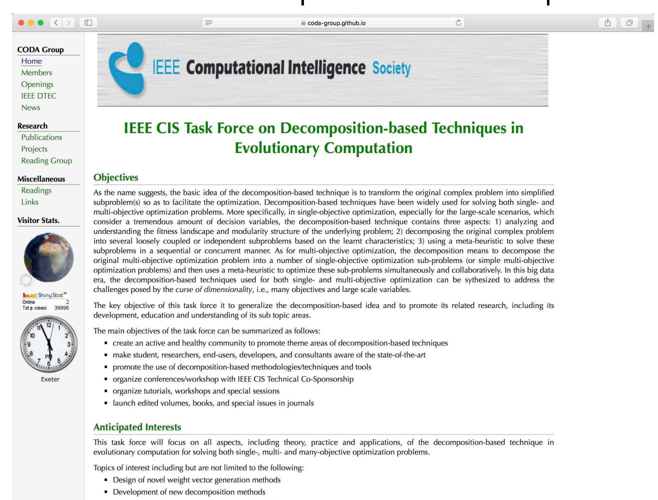
[32] Z. Wang, et al., "Adaptive Replacement Strategies for MOEA/D", IEEE Trans. Cybern., 46(2): 474-486, 2016.
[33] Y. Yuan, et al., "Balancing Convergence and Diversity in Decomposition-Based Many-Objective Optimizers", IEEE Trans. Evol. Comput., 20(2): 180-198, 2016.

Outline

- Why Multi-Objective Optimisation Important?
- Basic Concepts
- Simple MOEA/D
- Current Developments
  - Decomposition methods
  - Search methods
  - Collaboration
    - Mating selection
    - Replacement
- Resources
- Future Directions

Resources

- IEEE CIS task force on decomposition-based techniques in EC



coda-group.github.io

CODA Group
Home
Members
Openings
IEEE DTEC
News

Research
Publications
Projects
Reading Group

Miscellaneous
Readings
Links

Visitor Stats.

ShinyStat™
Online 2
Tot p.views 39996

Exeter

IEEE Computational Intelligence Society

IEEE CIS Task Force on Decomposition-based Techniques in Evolutionary Computation

Objectives

As the name suggests, the basic idea of the decomposition-based technique is to transform the original complex problem into simplified subproblem(s) so as to facilitate the optimization. Decomposition-based techniques have been widely used for solving both single- and multi-objective optimization problems. More specifically, in single-objective optimization, especially for the large-scale scenarios, which consider a tremendous amount of decision variables, the decomposition-based technique contains three aspects: 1) analyzing and understanding the fitness landscape and modularity structure of the underlying problem; 2) decomposing the original complex problem into several loosely coupled or independent subproblems based on the learnt characteristics; 3) using a meta-heuristic to solve these subproblems in a sequential or concurrent manner. As for multi-objective optimization, the decomposition means to decompose the original multi-objective optimization problem into a number of single-objective optimization sub-problems (or simple multi-objective optimization problems) and then uses a meta-heuristic to optimize these sub-problems simultaneously and collaboratively. In this big data era, the decomposition-based techniques used for both single- and multi-objective optimization can be sythesized to address the challenges posed by the *curse of dimensionality*, i.e., many objectives and large scale variables.

The key objective of this task force it to generalize the decomposition-based idea and to promote its related research, including its development, education and understanding of its sub topic areas.

The main objectives of the task force can be summarized as follows:

- create an active and healthy community to promote theme areas of decomposition-based techniques
- make student, researchers, end-users, developers, and consultants aware of the state-of-the-art
- promote the use of decomposition-based methodologies/techniques and tools
- organize conferences/workshop with IEEE CIS Technical Co-Sponsorship
- organize tutorials, workshops and special sessions
- launch edited volumes, books, and special issues in journals

Anticipated Interests

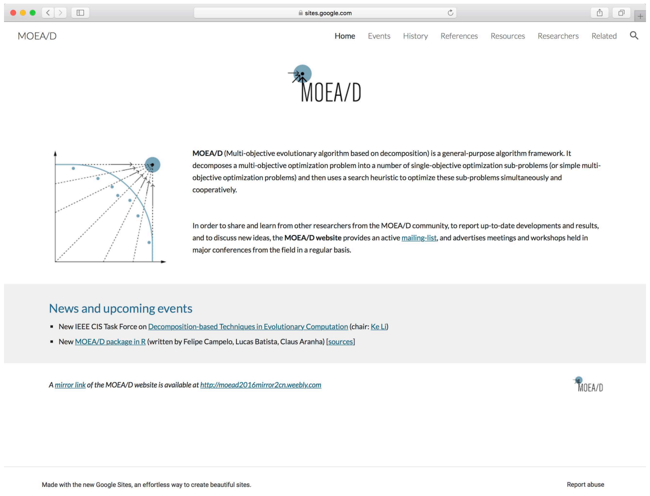
This task force will focus on all aspects, including theory, practice and applications, of the decomposition-based technique in evolutionary computation for solving both single-, multi- and many-objective optimization problems.

Topics of interest including but are not limited to the following:

- Design of novel weight vector generation methods
- Development of new decomposition methods
- Design of novel computational resource allocation strategies

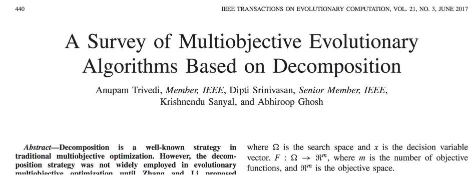
Resources (cont.)

- Website of MOEA/D: https://sites.google.com/view/moead/home



Resources (cont.)

- Three survey papers


440 IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, VOL. 21, NO. 3, JUNE 2017

A Survey of Multiobjective Evolutionary Algorithms Based on Decomposition

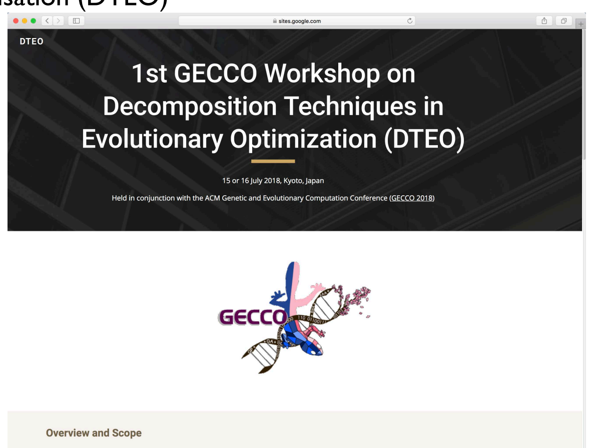
Anupam Trivedi, *Member, IEEE*, Dipti Srinivasan, *Senior Member, IEEE*, Krishnendu Sanyal, and Abhiroop Ghosh

Abstract—Decomposition is a well-known strategy in traditional multiobjective optimization. However, the decomposition strategy was not widely employed in evolutionary multiobjective optimization until Zhang and Li proposed

where Ω is the search space and x is the decision variable vector. $F : \Omega \to \Re^m$, where m is the number of objective functions, and $\Re^m$ is the objective space.

Events

- Workshop on decomposition techniques in evolutionary optimisation (DTEO)



Events

- Workshop on Computational Intelligence for Massive Optimisation (CIMO)

Outline

- Why Multi-Objective Optimisation Important?
- Basic Concepts
- Simple MOEA/D
- Current Developments
  - Decomposition methods
  - Search methods
  - Collaboration
    - Mating selection
    - Replacement
- Resources
- Future Directions

Future Directions

- Big optimisation
  - Many objectives
    - Is approximating the high-dimensional PF doable?
    - Problem reformulation (dimensionality reduction)
    - Visualisation
    - …
  - Many variables (large-scale)
    - Decomposition from decision space (divide-and-conquer): dependency structure analysis
    - What is the relationship between the decomposed variable and subproblem?
    - Sensitivity analysis for identifying important variables
    - …
  - Distributed and parallel computing platform
- EMO + MCDM: Human computer interaction perspective
  - Subproblem is another way to represent decision maker's preference
    - e.g. weighted scalarizing function, simplified MOP
  - How to help decision maker understand the solutions and inject appropriate preference information?
  - How to use preference information effectively?
  - …

Future Directions (cont.)

- How to make the collaboration more effective?
  - "*In case of two agents for one problem, collaboration is useful*" [34]
  - How about a multi-agent system and cooperative game?
- Automatic problem solving: meta-optimisation/learning perspective
  - Is the current MOEA/D the perfect algorithm structure?
  - Use artificial intelligence to design algorithm autonomously
  - Landscape analysis and problem feature engineering
  - Algorithm portfolio: choose the right algorithm structure for the right problem
  - …
- Data-driven optimisation
  - Build and maintain a surrogate for each subproblem
  - Subproblem has knowledge, e.g. solution history, knowledge can be shared among neighbourhood: transfer learning or multi-tasking?
  - …

[34] B. Huberman, et. al., "An Economics Approach to Hard Computational Problems", Science, 275(5296): 51-54, 1997.

Future Directions (cont.)

- Theoretical studies
  - Convergence analysis
  - Stopping condition
  - From an equilibrium perspective?
  - …
- Applications
  - Engineering, e.g. water, manufacturing, renewable energy, healthcare …
  - Search-based software engineering
  - …
- Any suggestions?
  - …

[22] B. Huberman, et. al., "An Economics Approach to Hard Computational Problems", Science, 275(5296): 51-54, 1997.


Thank you for your participation and any questions?