[GECCO'21 Tutorial]

Decomposition Multi-Objective Optimisation

Current Developments and Future Opportunities

Ke Li, Qingfu Zhang
Department of Computer Science, University of Exeter
Department of Computer Science, City University of Hong Kong
k.li@exeter.ac.uk

ACM ISBN 978-1-4503-8351-6/21/07…$15.00
https://doi.org/10.1145/3449726.3461395

Instructors

- **Ke Li** is a UKRI Future Leaders Fellow and a Senior Lecturer (Associate Professor) in Computer Science at the University of Exeter. His current research interests include the evolutionary multi-objective optimisation, automatic problem solving, machine learning and applications in search-based software engineering and water engineering.

- **Qingfu Zhang** is a Professor at the Department of Computer Science, City University of Hong Kong. His main research interests include evolutionary computation, optimization, neural networks, data analysis, and their applications. He is an Associate Editor of the IEEE Transactions on Evolutionary Computation and the IEEE Transactions Cybernetics. He was awarded the 2010 IEEE Transactions on Evolutionary Computation Outstanding Paper Award. He is on the list of the Thomson Reuters 2016 to 2019 highly cited researchers in computer science. He is an IEEE fellow.

2

Outline

- **Part I: Basics**
  - Basic Concepts
  - Ideas in MOEA/D
  - A Simple Variant
- **Part II: Advanced Topics**
  - Current Developments
    - Decomposition Methods
    - Search Methods
    - Collaboration Methods
  - Resources
  - Future Directions

3

Outline

- **Part I: Basics**
  - Basic Concepts
  - Ideas in MOEA/D
  - A Simple Variant
- **Part II: Advanced Topics**
  - Current Developments
    - Decomposition Methods
    - Search Methods
    - Collaboration Methods
  - Resources
  - Future Directions

4

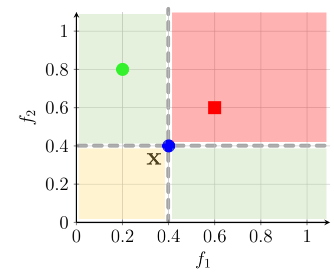
Multi-objective Optimisation Problem (MOP)

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

5

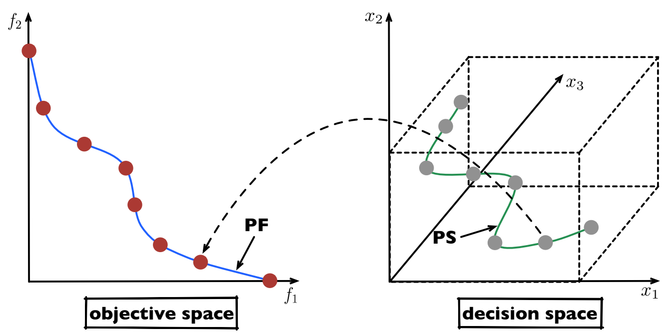
Dominance : How to compare two solutions

- Pareto dominance: $\mathbf{x}^1 \preceq \mathbf{x}^2$
  - $\mathbf{F}(\mathbf{x}^1)$ 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



6

Pareto-Optimal Solutions = Best Trade-off Candidates

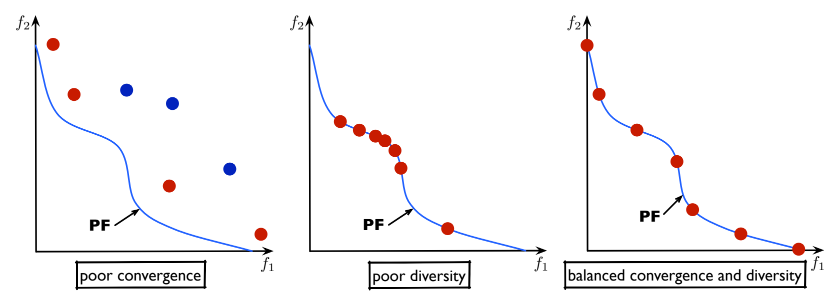


- $\boldsymbol{X}$ is Pareto-optimal iff no solution dominates it
- Pareto set (PS): all Pareto-optimal solutions in decision space
- Pareto front (PF): image of PS in the objective space

7

Convergence and Diversity in EMO

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

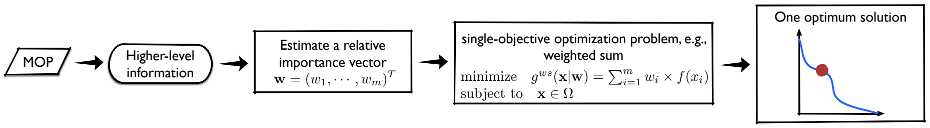


Achieving the balance between convergence and diversity is the key in evolutionary multi-objective optimisation

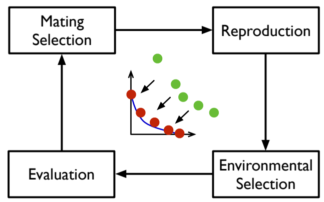
8

Classic Methods vs Evolutionary Approaches

- Classic multi-objective optimisation [3]



- Evolutionary multi-objective optimisation (EMO)
  - set-based method, approximate the PF at a time
  - Major EMO algorithms
    - Pareto dominance based: NSGA-II, SPEA2, PESA2, …
    - Performance indicator based: SMS-EMOA, HypE, …
    - Decomposition based



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

Outline

Decomposition in EMO

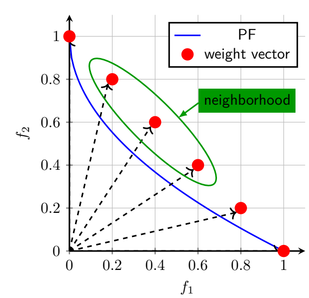
- Decomposition has been used to some extent in EMO area for many years.
- Examples includes:
  - MOGLS [Ishibuchi, et al, 1998, Jaszkiewicz, 2002]
  - MOSA [Ulungu ,et al, 1999]
  - MOTS [Hansen, 1997]
  - 2PLS [Paquete & Stutzle, 2003]
  - AWA [Jin et al, 2001]
  - MSOPS [Hughes, 2003]
  - MOTGA [Alves, 2007]
  - CMOGA [4] [Murata et al, 2001] → MOEA/D
  - ….

These algorithms use traditional aggregation approaches.

[4] T. Murata, H. Ishibuchi, M. Gen, "Specification of Genetic Search Directions in Cellular Multi-objective Genetic Algorithms", EMO'01: 82-95, 2001.

MOEA/D = Decomposition + Collaboration

- Basic idea
  - Decomposition (from traditional optimisation)
    - 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 (from EC)
    - Population-based technique: N agents for N subproblems.
    - Subproblems are related to each other while N agents solve these subproblems in a collaborative manner.

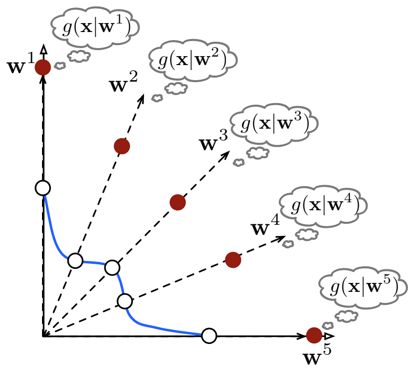


Subproblem Settings

- Subproblem formulation

multiple objectives — parameters — scalarizing function

$\mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), \cdots, f_m(\mathbf{x})) \in \mathbb{R}^m \longrightarrow$ transformation $\longrightarrow g(\mathbf{x}|\cdot)$

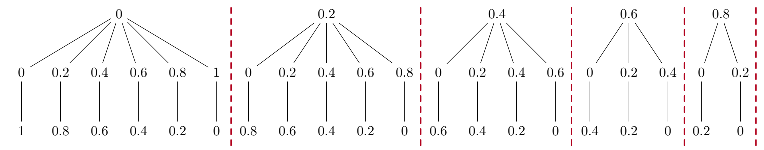


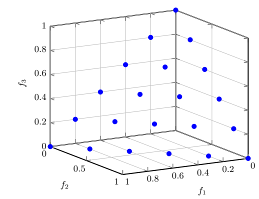
$$\begin{array}{ll} \text{minimize} & g^{ws}(\mathbf{x}|\mathbf{w}) = \sum_{i=1}^{m} w_i f_i(\mathbf{x}) \\ \text{subject to} & \mathbf{x} \in \mathbf{\Omega} \end{array}$$

☞ NOTE: It works for convex PF!

Subproblem Settings (cont.)

- Weight vector/Reference point Setting [1]
  - Sample a set of evenly distributed weight vectors from a unit simplex
  - $\mathbf{w} = (w_1, \cdots, w_m)^T$ where $\sum_{i=1}^{m} w_i = 1, \mathbf{w} \in \mathbb{R}^m$

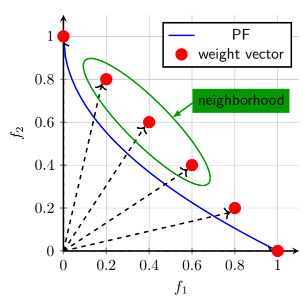




[5] I. Das et. al., "Normal-Boundary Intersection: A New Method for Generating the Pareto Surface in Nonlinear Multicriteria Optimization Problems", SIAM J. Optim, 8(3): 631-657, 1998.

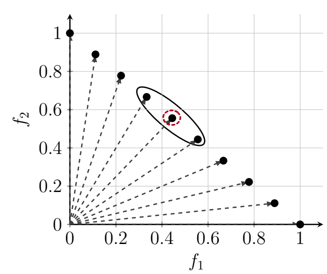
Decomposition-based EMO

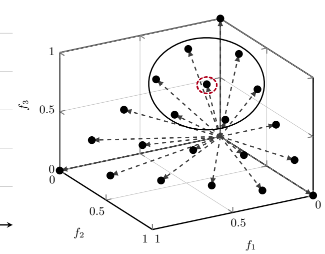
- Basic idea
  - Decomposition (from traditional optimisation)
  - Collaboration (from EC)

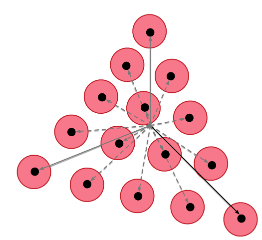


Subproblem Settings (cont.)

- Neighbourhood structure:
  - Two subproblems are neighbours if their weight vectors are close.
  - Neighbouring subproblems are more likely to have similar properties (e.g. similar objective function and/or optimal solution).

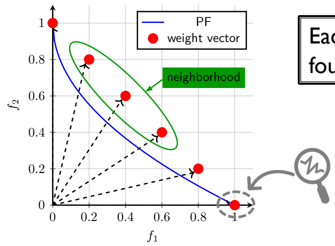






Many different ways for defining neighbourhood structure.

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

- **Part I: Basics**
  - Basic Concepts
  - Ideas in MOEA/D
  - A Simple Variant
- **Part II: Advanced Topics**
  - Current Developments
    - Decomposition Methods
    - Search Methods
    - Collaboration Methods
  - 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.

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

Experimental Results

- Continuous MOP test instances.
- Same population size and same crossover and mutation.
- Same number of objective function evaluations.

D-METRIC VALUES OF THE SOLUTIONS FOUND BY MOEA/D WITH THE TCHEBYCHEFF APPROACH AND NSGA-II. THE NUMBERS IN PARENTHESES REPRESENT THE STANDARD DEVIATION

		NSGA-II	MOEA/D
Instance	ZDT1	0.0050 (0.0002)	0.0055 (0.0039)
	ZDT2	0.0049 (0.0002)	0.0079 (0.0109)
	ZDT3	0.0065 (0.0054)	0.0143 (0.0091)
	ZDT4	0.0182 (0.0237)	0.0076 (0.0023)
	ZDT6	0.0169 (0.0028)	0.0042 (0.0003)
	DTLZ1	0.0648 (0.1015)	0.0317 (0.0005)
	DTLZ2	0.0417 (0.0013)	0.0389 (0.0001)

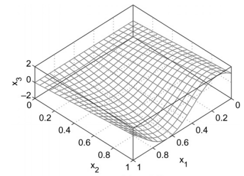
TABLE V
AVERAGE CPU TIME (IN SECONDS) USED BY NSGA-II AND MOEA/D WITH THE TCHEBYCHEFF APPROACH

		NSGA-II	MOEA/D
Instance	ZDT1	1.03	0.60
	ZDT2	1.00	0.47
	ZDT3	1.03	0.57
	ZDT4	0.77	0.33
	ZDT6	0.73	0.27
	DTLZ1	10.27	1.20
	DTLZ2	8.37	1.10

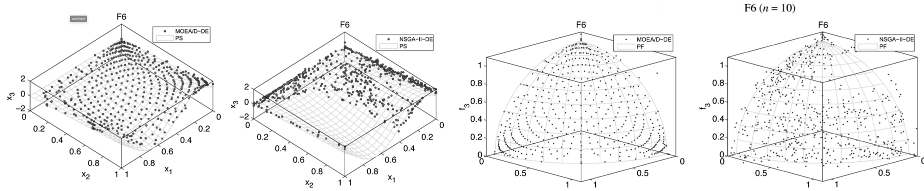
- Observation: it works.
  - Solution quality: MOEA/D ≈ NSGA-II
  - CPU time: MOEA/D is better.

Test Instances with complicated PS shapes

F6	$f_1 = \cos(0.5x_1\pi)\cos(0.5x_2\pi) + \frac{2}{\lvert J_1\rvert}\sum_{j\in J_1}(x_j - 2x_2\sin(2\pi x_1 + \frac{j\pi}{n}))^2$ $f_2 = \cos(0.5x_1\pi)\sin(0.5x_2\pi) + \frac{2}{\lvert J_2\rvert}\sum_{j\in J_2}(x_j - 2x_2\sin(2\pi x_1 + \frac{j\pi}{n}))^2$ $f_3 = \sin(0.5x_1\pi) + \frac{2}{\lvert J_3\rvert}\sum_{j\in J_3}(x_j - 2x_2\sin(2\pi x_1 + \frac{j\pi}{n}))^2$ where $J_1 = \{j \mid 3 \le j \le n$, and $j-1$ is a multiplication of $3\}$, $J_2 = \{j \mid 3 \le j \le n$, and $j-2$ is a multiplication of $3\}$, $J_3 = \{j \mid 3 \le j \le n$, and j is a multiplication of $3\}$ Its PS is $x_j = 2x_2\sin(2\pi x_1 + \frac{j\pi}{n}), j = 3, \ldots, n$.	$[0,1]^2 \times [-2,2]^{n-2}$



F6 ($n = 10$)



- MOEA/D is better.
- Combination of MOEA/D and NSGA-II (Cai, et al): Champion in CEC 2017 competition.

Some Remarks

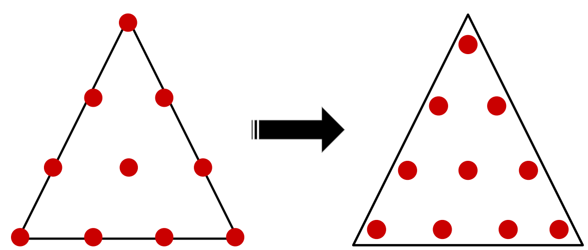
- Diversity among subproblems leads to diversity among solutions
- MOEA/D has a well-organised memory.
- It deals with a population of subtasks, related to recent proposed evolutionary multi-task optimisation (Y-S Ong et al, 2016).

Outline

- **Part I: Basics**
  - Basic Concepts
  - Ideas in MOEA/D
  - A Simple Variant
- **Part II: Advanced Topics**
  - Current Developments
    - Decomposition Methods
    - Search Methods
    - Collaboration Methods
  - Resources
  - Future Directions

Setting of Weight Vectors (cont.)

- Drawbacks of the Das and Dennis's method
  - Not very '*uniform*' [2]



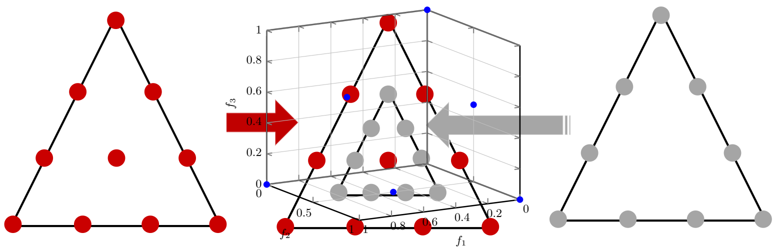
simplex-lattice design

uniform design [7]

[7] 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.

Setting of Weight Vectors (cont.)

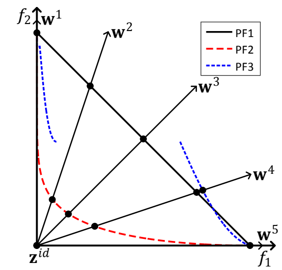
- Drawbacks of the Das and Dennis's method
  - …
  - Number of weights is restricted to $N = \binom{H+m-1}{m-1}$ [8, 9]
    - N increases non-linearly with m
    - If N is not large enough, all weight vectors will be at the boundary of the simplex



[8] K. Deb and H. Jain, "An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints", IEEE Trans. Evol. Comput., 18(4): 577-601, 2014.
[9] **K. Li**, K. Deb, 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.)

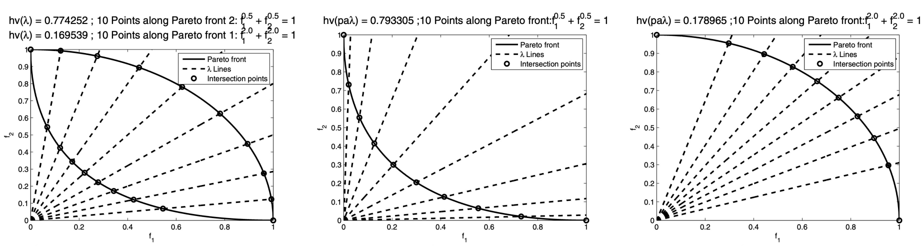
- Is even distribution really a good choice?
  - Do NOT always lead to evenly distributed solutions
  - Do NOT support all PF shapes
    - Disconnected PF
    - Inverted PF
    - …



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

Setting of Weight Vectors (cont.)

- Is even distribution really a good choice?
  - …



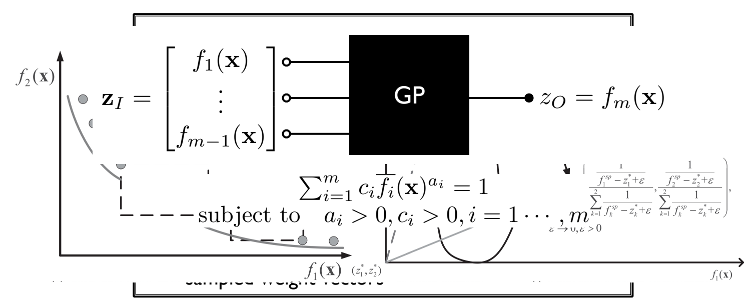
(a) λ (MOEA/D) for convex and concave PF (b) $pa\lambda$ for convex PF (c) $pa\lambda$ for concave PF

Assume PF as $\sum_{i=1}^{m} f_i^p = 1$, estimate p according to the number of non-dominated solutions [10]

[10] S. Jiang, et al., "Multiobjective Optimization by Decomposition with Pareto-adaptive Weight Vectors", ICNC'11, 1260-1264, 2011.

Setting of Weight Vectors (cont.)

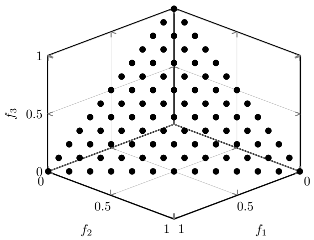
- Is even distribution really a good choice?
  - …



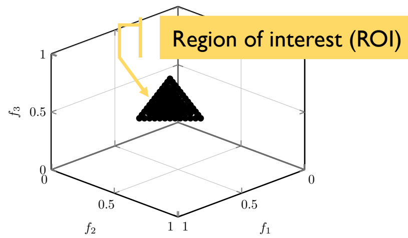
[11] Y. Qi, et al., "MOEA/D with Adaptive Weight Adjustment", Evol. Comput. 22(2): 231–264, 2014.
[12] M. Wu, et al., "Learning to Decompose: A Paradigm for Decomposition-Based Multiobjective Optimization", IEEE Trans. Evol. Comput., 23(3): 376-390, 2019.
[13] F. Gu, et al., "Self-Organizing Map-Based Weight Design for Decomposition-Based Many-Objective Evolutionary Algorithm", IEEE Trans. Evol. Comput., 22(2): 211-225, 2018.
[14] Y. Liu, et al., "Adapting Reference Vectors and Scalarizing Functions by Growing Neural Gas to Handle Irregular Pareto Fronts", IEEE Trans. Evol. Comput., accepted for publication, 2019.

Preference Incorporation

- Weight vectors represent the decision maker's preference information on the PF.



without any preference

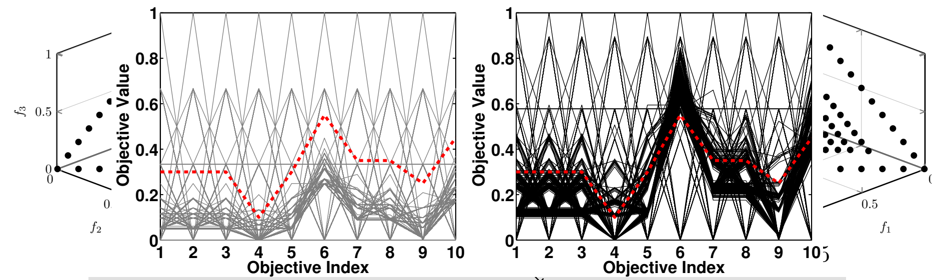


with preference

29

Preference Incorporation (cont.)

- Weight vectors represent the decision maker's preference information on the PF.
  - Shift weight vectors towards the decision maker supplied aspiration level vector [15]
    - Closed form
    - Size of ROI is controllable
    - Keep the boundary

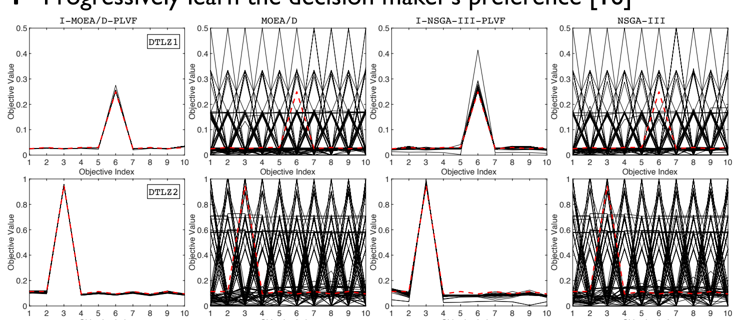


[15] K. Li, et al., "Integration of Preferences in Decomposition Multi-Objective Optimisation", IEEE Trans. Cyber., 2018.

30

Preference Incorporation (cont.)

- Weight vectors represent the decision maker's preference information on the PF.
  - Shift weight vectors towards the decision maker supplied aspiration level vector [15]
  - Progressively learn the decision maker's preference [16]

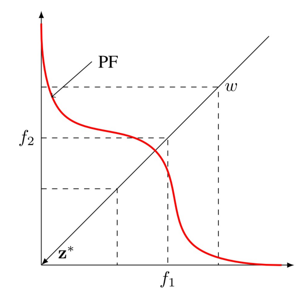


[16] K. Li, et al., "Interactive Decomposition Multiobjective Optimization Via Progressively Learned Value Functions", IEEE Trans. Fuzzy Systems, 2019.

31

Revisit Weighted Tchebycheff

- Weighted Tchebycheff



weighted Tchebycheff

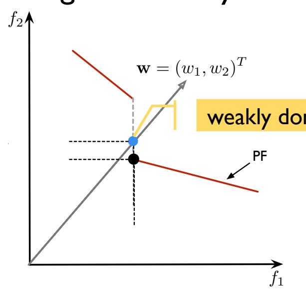
$$g(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = \max_{1\leq i\leq m} w_i|f_i(\mathbf{x} - z_i^*|$$

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

32

Revisit Weighted Tchebycheff

Weighted Tchebycheff



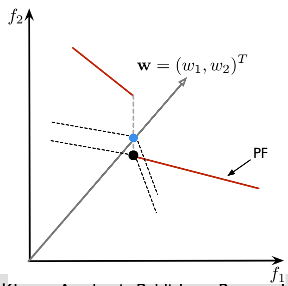
Drawbacks:
- non-smooth, weakly dominated solution
- evenly distributed weights do NOT lead to evenly distributed solutions
- might easily loose diversity

weighted Tchebycheff

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

augmented scalarizing function

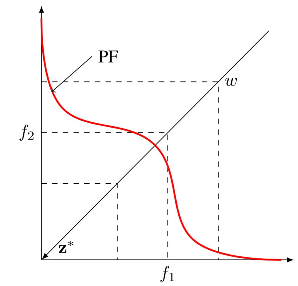
$$g^a(\mathbf{x}|\mathbf{w},\mathbf{z}^*) = \max_{1\leq i\leq m} (\frac{f_i(\mathbf{x}-z_i^*}{w_i}) + \rho\sum_{i=1}^{m}(\frac{f_i(\mathbf{x}-z_i^*}{w_i})$$



[17] K. Miettinen, "Nonlinear Multiobjective Optimization", Kluwer Academic Publishers, Boston, 1999.

Revisit Weighted Tchebycheff

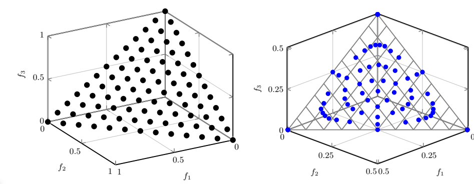
Weighted Tchebycheff



weighted Tchebycheff

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

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

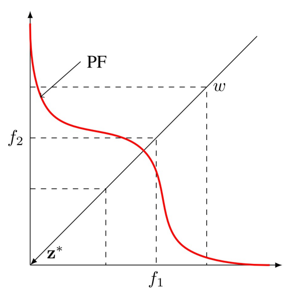


The search direction for $\mathbf{w} = (w_1, \cdots, w_m)^T$ is $\mathbf{w} = (\frac{1/w_1}{\sum_{i=1}^{m} 1/w_i}, \cdots, \frac{1/w_m}{\sum_{I=1}^{m} 1/w_i})^T$

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

Revisit Weighted Tchebycheff

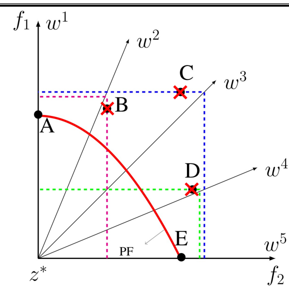
Weighted Tchebycheff



weighted Tchebycheff

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

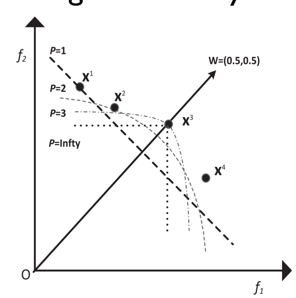
Drawbacks:
- non-smooth, weakly dominated solution
- evenly distributed weights do NOT lead to evenly distributed solutions
- might easily loose diversity



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

Revisit Weighted Tchebycheff

Weighted Tchebycheff



weighted L_p scalarizing [19]

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

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

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

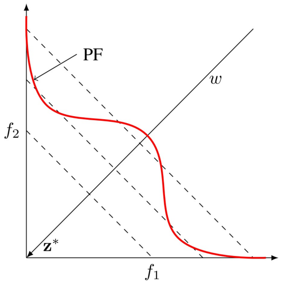
Pareto adaptive scalarizing to choose p

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

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

Revisit Weighted Sum

- Weighted Sum

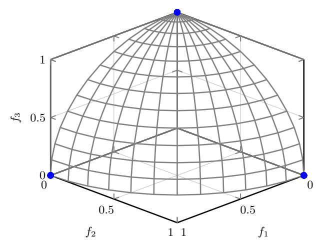


weighted sum

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

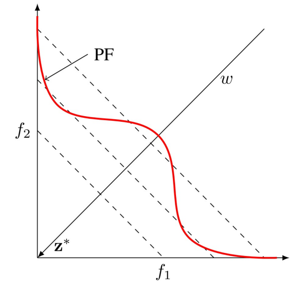
Drawbacks:

- 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



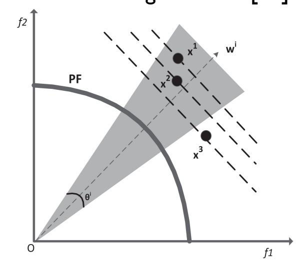
weighted sum

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

Is weighted sum really that bad?

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

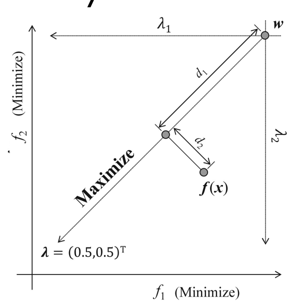
Localized weighted sum [20]



[20] 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]



Inverted PBI [23]

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

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

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

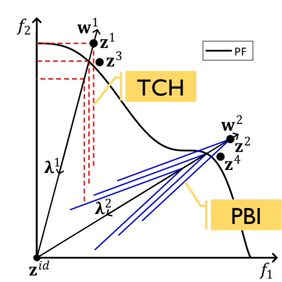
Characteristics:

- d_1 'measures' the convergence
  - can be replaced by other measure [7]
- d_2 'measures' the diversity
  - can be replaced by angle [21, 22]
- θ controls the contour and trade-offs

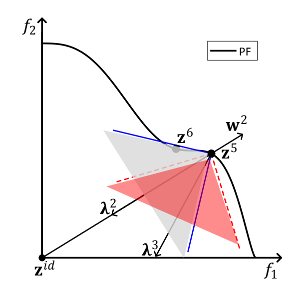
[7] Q. Zhang et al., "MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition", IEEE Trans. Evol. Comput., 11(6): 712-731, 2007.
[21] R. Cheng, et al., "A Reference Vector Guided Evolutionary Algorithm for Many-Objective Optimization", IEEE Trans. Evol. Comput., 20(5): 773-791, 2016.
[22]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

Learning to Decompose

- Search dynamics of MOEA/D largely depends on the contours induced by the subproblem formulation.
  - The shape of the contour might mislead the selection
  - The search direction might not be suitable



- $\boldsymbol{z}^3$ is better than $\boldsymbol{z}^1$; $\boldsymbol{z}^4$ is better than $\boldsymbol{z}^2$



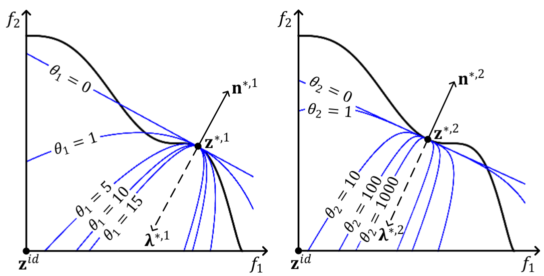
- $\boldsymbol{z}^5$ is better than $\boldsymbol{z}^6$ with respect to $\boldsymbol{w}^2$

[12] M. Wu, et al., "Learning to Decompose: A Paradigm for Decomposition-Based Multiobjective Optimization", IEEE Trans. Evol. Comput., 23(3): 376-390, 2019.

Learning to Decompose (cont.)

- Learned subproblem formulation

$$\text{minimize } y(\mathbf{x}|\mathbf{n}^*,\mathbf{z}^*) = h(\overline{\mathbf{F}}(\mathbf{x})|\mathbf{n}^*,\mathbf{z}^*) = d_1 + \theta_1 d_2^2 + \theta_2 d_2^4,$$
$$d_1 = (\overline{\mathbf{F}}(\mathbf{x}) - \mathbf{z}^*)^T\mathbf{n}^*$$
$$d_2 = ||\overline{\mathbf{F}}(\mathbf{x}) - \mathbf{z}^* - d_1\mathbf{n}^*||.$$



- A smaller/larger θ_1 or θ_2 leads to a wider/narrower opening.
  - Too narrow opening causes a strict selection of the better solutions.
- θ_1 controls the opening and the curvature of the contour at the vertex; θ_2 does not influence the curvature.

How to set θ_1 or θ_2?

[12] M. Wu, et al., "Learning to Decompose: A Paradigm for Decomposition-Based Multiobjective Optimization", IEEE Trans. Evol. Comput., 23(3): 376-390, 2019.

41

Learning to Decompose (cont.)

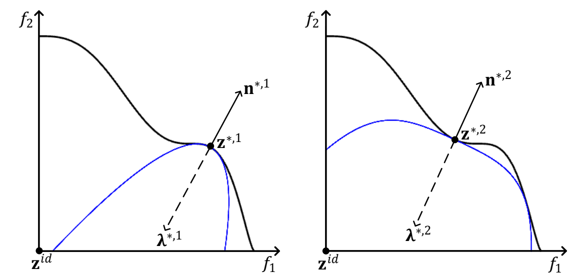
- Learned subproblem formulation

$$\text{minimize } y(\mathbf{x}|\mathbf{n}^*,\mathbf{z}^*) = h(\overline{\mathbf{F}}(\mathbf{x})|\mathbf{n}^*,\mathbf{z}^*) = d_1 + \theta_1 d_2^2 + \theta_2 d_2^4,$$
$$d_1 = (\overline{\mathbf{F}}(\mathbf{x}) - \mathbf{z}^*)^T\mathbf{n}^*$$
$$d_2 = ||\overline{\mathbf{F}}(\mathbf{x}) - \mathbf{z}^* - d_1\mathbf{n}^*||.$$

$$\theta_1 = \max(\frac{\kappa^*}{2}, 0) + 0.1$$
$$\theta_2 = \max(\min\{\theta_2 | h(\mathbf{z}|\mathbf{n}^*,\mathbf{z}^*) > 0, \forall \mathbf{z} \in Z^* \backslash \mathbf{z}^*\}, 0) + 0.1$$

Let the curvature of the contour

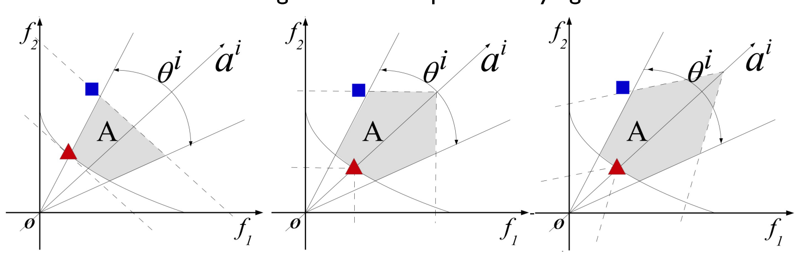
All other samples on the estimated PF have worse function values than $\mathbf{z}^*$ on $h(\mathbf{z}|\mathbf{n}^*,\mathbf{z}^*)$.



42

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 to reduce the improvement region [24]

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

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

43

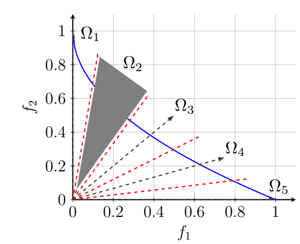
Subproblem Can Be Multi-Objective …

- MOP to MOP (M2M)
  - Decompose a MOP into K ($K > 1$) constrained MOPs [25].

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

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

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



[25] H. Liu, F. Gu and **Q. Zhang**, "Decomposition of a Multiobjective Optimization Problem Into a Number of Simple Multiobjective Subproblems", IEEE Trans. Evol. Comput., 18(3): 450-455, 2014.

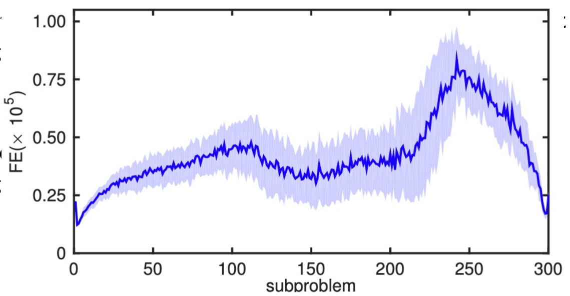
44

Dynamic Resource Allocation

- Are all subproblems equally important?
  - Some r
  - Differe ... tional resour
- Dynami ... /D [26]
  - Utili ... :
    - e.g
  - Allo
    - e.g

$$f_1(\mathbf{x}) = x1 + \frac{2}{|J_1|}\sum_{j\in J_1}(x_j - \sin(6\pi x_1 + \frac{j}{n}\pi))$$

$$f_2(\mathbf{x}) = 1 - \sqrt{x1} + \frac{2}{|J_2|}\sum_{j\in J_2}(x_j - \sin(6\pi x_1 + \frac{j}{n}\pi))$$



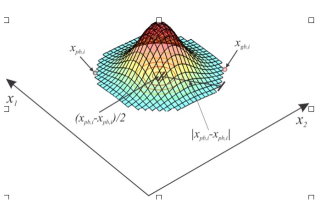
[26] A. Zhou and **Q. Zhang**, "Are All the Subproblems Equally Important? Resource Allocation in Decomposition-Based Multiobjective Evolutionary Algorithms", IEEE TEVC, 20(1): 52-64, 2016.

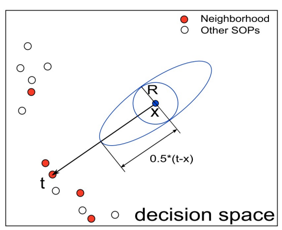
Outline

- **Part I: Basics**
  - Basic Concepts
  - Ideas in MOEA/D
  - A Simple Variant
- **Part II: Advanced Topics**
  - Current Developments
    - Decomposition Methods
    - Search Methods
    - Collaboration Methods
  - Resources
  - Future Directions

Search Methods

- Offspring reproduction in MOEA/D
  - Neighbourhood defines where to find mating parents
  - Any genetic operator can be used
    - GA [6], DE [27], PSO [28], guided mutation [29], …

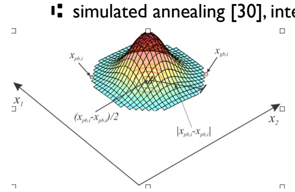


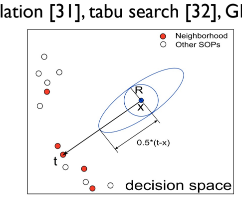


[6] **Q. Zhang** and H. Li, "MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition", IEEE Trans. Evol. Comput., 11(6): 712-731, 2007.
[27] H. Li and **Q. Zhang**, "Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II", IEEE Trans. Evol. Comput., 13(2): 284-302, 2009.
[28] S. Martínez, et al., "A multi-objective PSO based on decomposition", in GECCO 2011.
[29] 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 [30], interpolation [31], tabu search [32], GRASP [33], Nelder-Mead [34], …

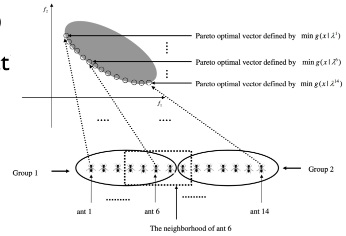




[30] H. Li, et al., "An adaptive evolutionary multi-objective approach based on simulated annealing", Evol. Comput. 19(4): 561-595, 2011.
[31] K. Sindhya, "A new hybrid mutation operator for multiobjective optimization with differential evolution", Soft Comput., 15:2041–2055, 2011.
[32] A. Alhindi and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

Search Methods

- Offspring reproduction in MOEA/D
  - Neighbourhood defines where to find mat
  - 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 [35], EDA [36], cross entropy [37], graphical model [38], CMA-ES [39], based on memory of itself and its neighbour

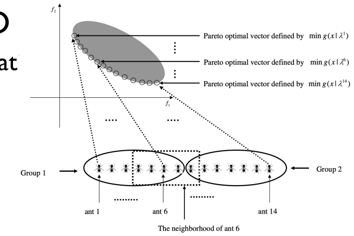


[35] L. Ke, **Q. Zhang**, et al., "MOEA/D-ACO: A Multiobjective Evolutionary Algorithm Using Decomposition and Ant Colony", IEEE Trans. Cybern., 43(6): 1845-1859, 2013.
[36] A. Zhou, **Q. Zhang**, et al., "A Decomposition based Estimation of Distribution Algorithm for Multiobjective Traveling Salesman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

49

Search Methods

- Offspring reproduction in MOEA/D
  - Neighbourhood defines where to find mat
  - 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 [28], EDA [29], cross entropy [30], graphical model [31], CMA-ES [32], 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

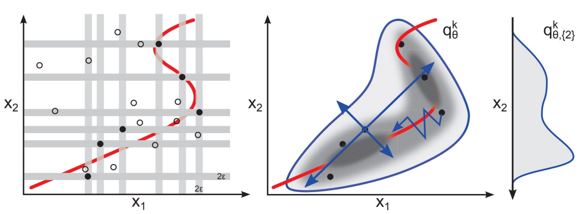


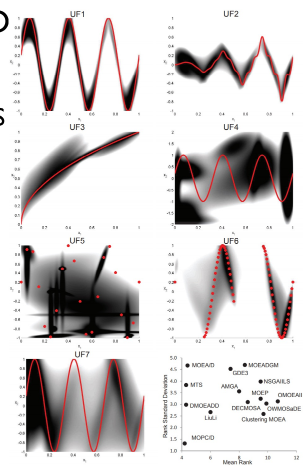
[35] L. Ke, **Q. Zhang**, et al., "MOEA/D-ACO: A Multiobjective Evolutionary Algorithm Using Decomposition and Ant Colony", IEEE Trans. Cybern., 43(6): 1845-1859, 2013.
[36] A. Zhou, **Q. Zhang**, et al., "A Decomposition based Estimation of Distribution Algorithm for Multiobjective Traveling Salesman Problems", Computers & Mathematics with Applications, 66(10): 1857-1868, 2013.
…

50

Search Methods

- 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





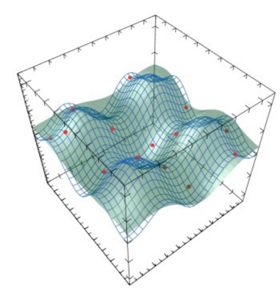
- Fit a Gaussian mixture model using solutions associated with each subproblem
- Search is based one sampling or local search upon the fitted model

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

51

Search Methods

- Expensive optimisation
  - Building surrogate model for expensive objective function
    - e.g. Gaussian processes (Kriging) [38, 39], RBF [40], …

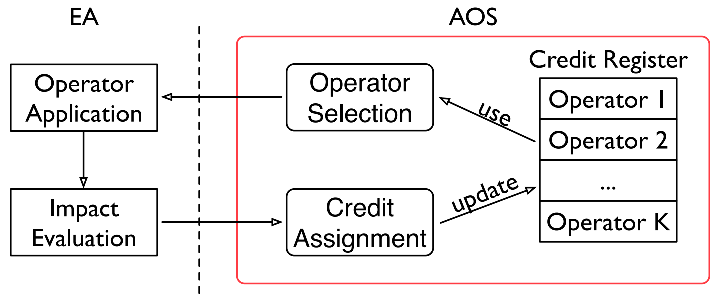


[41] **Q. Zhang**, et al., "Expensive Multiobjective Optimization by MOEA/D with Gaussian Process Model", IEEE Trans. Evol. Comput., 14(3): 456-474, 2010.
[42] T. Chugh, et al., "A Surrogate-Assisted Reference Vector Guided Evolutionary Algorithm for Computationally Expensive Many-Objective Optimization",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

52

Search Methods

- Adaptive operator selection as a bandit problem [37]
  - Strike the balance between the exploration and exploitation
    - Exploration: acquire new information (diversity)
    - Exploitation: capitalise on the available knowledge (convergence)

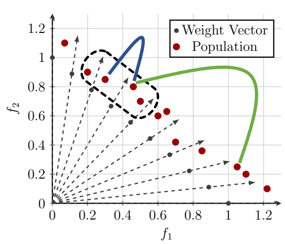


[44] **K. Li**, et al., "Adaptive operator selection with bandits for multiobjective evolutionary algorithm based on decomposition", IEEE Trans. Evol. Comput., 18(1): 114-130, 2014.

Outline

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)

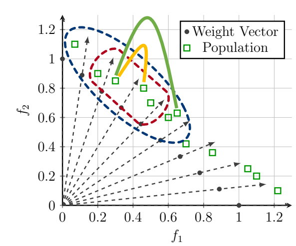


- Focusing on the neighbourhood is too much exploited
- Give some chance to explore in the whole population [27]

[27] 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)



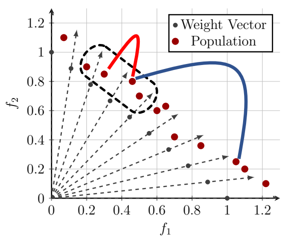
Effects of neighbourhood size (NS)
- Large neighbourhood makes the search globally
- Small neighbourhood encourages local search

Build an ensemble of neighbourhood sizes and chooses the appropriate one based on their historical performance. [45]

[45] S. Zhao, **Q. Zhang**, 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)

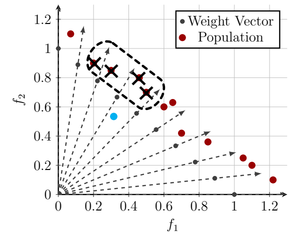


Take crowdedness into consideration [46]
- Compute the niche count of each solution within agent i's neighbour
- Select mating parents from outside of the neighbour if solutions are overly crowded

[46] 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.

Replacement

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

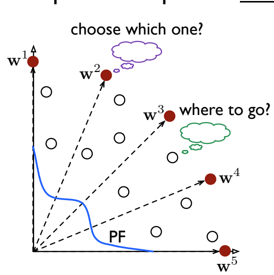


- The replacement strategy of the oracle MOEA/D is too greedy
- Offspring is only allowed to replace a limited number of parents [27]
  - Pros: Good for diversity
  - Cons: convergence may be slow

[27] 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 [47, 48]
  - Subproblems and solutions are two sets of agents
  - Subproblems 'prefer' convergence, solutions 'prefer' diversity



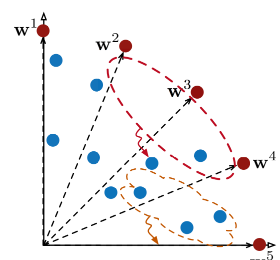
selection — matching

- 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, …

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

Replacement (cont.)

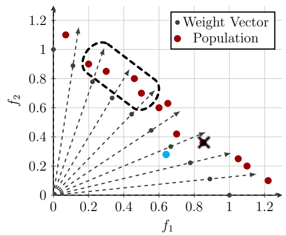
- Matching-based selection (extension) [49]
  - Identify the inter-relationship between subproblems and solutions
    - Find the related subproblems to each solution (e.g. fitness)
    - Find the related solutions for each subproblem (e.g. closeness)
  - Selection mechanism: each subproblem chooses its favourite solution



[49] **K. Li**, **Q. Zhang**, 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 [50]
    - 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 [51]
    - Update the newly generated offspring's 'nearest' subproblems



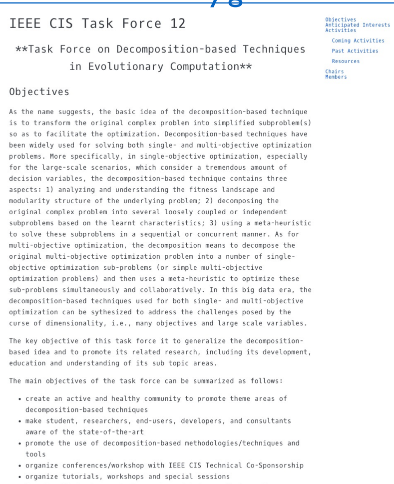
[50] Z. Wang, **Q. Zhang**, et al., "Adaptive Replacement Strategies for MOEA/D", IEEE Trans. Cybern., 46(2): 474-486, 2016.

[51] Y. Yuan, et al., "Balancing Convergence and Diversity in Decomposition-Based Many-Objective Optimizers", IEEE Trans. Evol. Comput., 20(2): 180-198, 2016.

Outline

Resources

- IEEE CIS task force on decomposition-based techniques in EC: https://cola-laboratory.github.io/docs/dtec/



IEEE CIS Task Force 12

Task Force on Decomposition-based Techniques in Evolutionary Computation

Objectives
Anticipated Interests
Activities
Coming Activities
Past Activities
Resources
Chairs
Members

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.

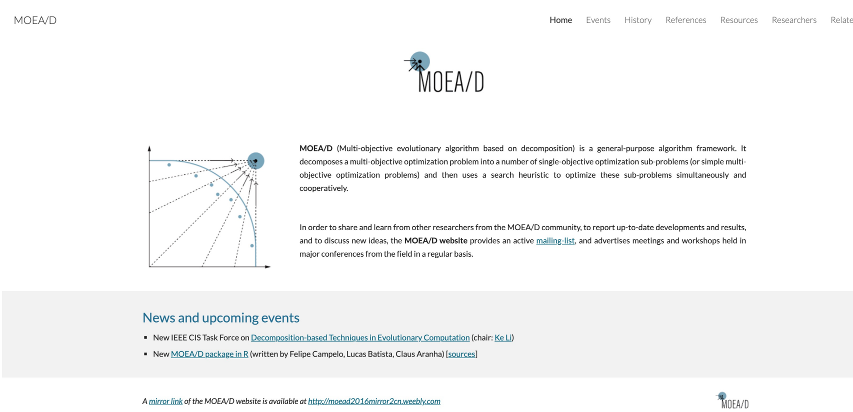
The key objective of this task force it to generalize the decomposition-based idea and to promote its related research, including its development, education and understanding of its sub topic areas.

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

Resources (cont.)

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



MOEA/D

Home Events History References Resources Researchers Related

MOEA/D

MOEA/D (Multi-objective evolutionary algorithm based on decomposition) is a general-purpose algorithm framework. It decomposes a multi-objective optimization problem into a number of single-objective optimization sub-problems (or simple multi-objective optimization problems) and then uses a search heuristic to optimize these sub-problems simultaneously and cooperatively.

In order to share and learn from other researchers from the MOEA/D community, to report up-to-date developments and results, and to discuss new ideas, the **MOEA/D website** provides an active mailing-list, and advertises meetings and workshops held in major conferences from the field in a regular basis.

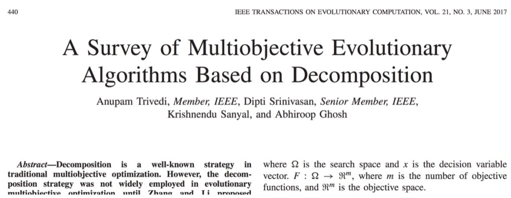
News and upcoming events

- New IEEE CIS Task Force on Decomposition-based Techniques in Evolutionary Computation (chair: Ke Li)
- New MOEA/D package in R (written by Felipe Campelo, Lucas Batista, Claus Aranha) [sources]

A mirror link of the MOEA/D website is available at http://moead2016mirror2cn.weebly.com

MOEA/D

Resources (cont.)



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

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

A Survey of Decomposition Methods for Multi-objective Optimization

Alejandro Santiago, Héctor Joaquín Fraire Huacuja, Bernabé Dorronsoro, Johnatan E. Pecero, Claudia Gómez Santillan, Juan Javier González Barbosa and José Carlos Soto Monterrubio

65

Resources (cont.)

- Workshop on decomposition techniques in evolutionary optimisation (DTEO)

66

Resources (cont.)

- Special Session on Advances in Decomposition-based Evolutionary Multi-objective Optimization (ADEMO)

67

Resources (cont.)

- **EMO 2021**: 11th International Conference on Evolutionary Multi-Criterion Optimization

68

Outline

- **Part I: Basics**
  - Basic Concepts
  - Ideas in MOEA/D
  - A Simple Variant
- **Part II: Advanced Topics**
  - Current Developments
    - Decomposition Methods
    - Search Methods
    - Collaboration Methods
  - Resources
  - Future Directions

69

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?
  - …

70

Future Directions (cont.)

- How to make the collaboration more effective?
  - "*In case of two agents for one problem, collaboration is useful*" [52]
  - 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?
  - …

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

71

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?
  - …

72