Recent Advances in Multi-objective and Many-objective Evolutionary Algorithms

Dr. Anupam Trivedi and Prof. Dipti Srinivasan

Department of Electrical and Computer Engineering
National University of Singapore

CEC 2017, San Sebastian, Spain

Outline

1. Introduction to Multi-objective Optimization
   - Multi-objective Optimization
   - Multi-objective Optimization Methods
   - Goals in Multi-objective Optimization
   - Evolutionary Multi-objective Optimization Frameworks
2. Multi-objective Evolutionary Algorithm based on Decomposition
   - Introduction to MOEA/D
   - Main Design Components of MOEA/D
   - Studies on Weight Vector Generation Methods
   - Studies on Decomposition Approaches
   - Studies on Computational Resource Allocation
   - Studies on Modifications in the Reproduction Operators
   - Studies on Mating Selection and Replacement Mechanism
   - Studies on Many-objective Optimization
3. Directions for Future Work

Outline

1 Introduction to Multi-objective Optimization
- Multi-objective Optimization
- Multi-objective Optimization Methods
- Goals in Multi-objective Optimization
- Evolutionary Multi-objective Optimization Frameworks

2 Multi-objective Evolutionary Algorithm based on Decomposition
- Introduction to MOEA/D
- Main Design Components of MOEA/D
- Studies on Weight Vector Generation Methods
- Studies on Decomposition Approaches
- Studies on Computational Resource Allocation
- Studies on Modifications in the Reproduction Operators
- Studies on Mating Selection and Replacement Mechanism
- Studies on Many-objective Optimization

3 Directions for Future Work

Introduction to Multi-objective Optimization

Real World Example Problem - Flight Ticket Booking

Introduction to Multi-objective Optimization

Real World Example Problem - Flight Ticket Booking

- Booking a flight ticket from Singapore to Spain
- Objectives are - 1) minimizing cost and 2) maximizing comfort
- Different airlines available (search space) : Singapore Airlines, British Airways, Air Asia, etc
- If cost is the only objective : Air Asia, Tiger Airways, Scoot
- If comfort is the only objective : Singapore Airlines, British Airways
- The above solutions represent extremes

Introduction to Multi-objective Optimization

Real World Example Problem - Flight Ticket Booking

Introduction to Multi-objective Optimization

Real World Example Problem - Flight Ticket Booking

- However, there are several intermediate solutions as well : Jet Airways, Silk Air
- Overall, there exists several trade-off solutions
- Airline booking portals (search engines) like expedia, zuji, etc. return all the trade-off solutions
- Decision Maker (Traveler) then selects a solution according to his/her budget and desired comfort level
- No single solution can be said to be optimal in terms of both the objectives
- All trade-off solutions are equally important before a solution is actually selected

Introduction to Multi-objective Optimization

Decision Making - Buying a Car

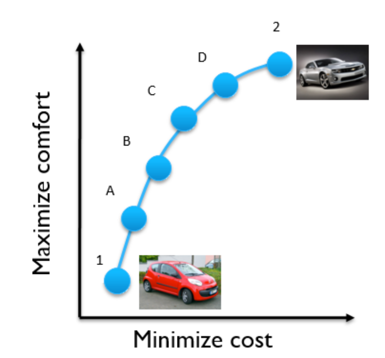


Figure: Trade-off Solutions.

Introduction to Multi-objective Optimization

Decision Making - Buying a Car

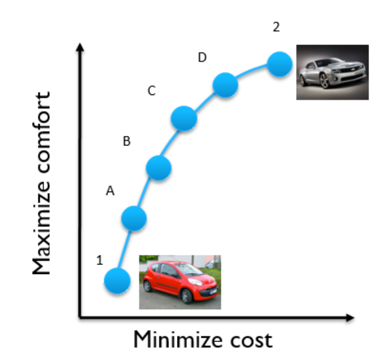


Figure: Trade-off Solutions.

Features of a MOP

- Two or three conflicting objectives
- There is no single optimum solution
- Multiple (trade-off) optimal solutions exist and all such optimal solutions are important
- Without any further information, no solution from the set of optimal solutions can be said to be better than other

Introduction to Multi-objective Optimization

Mathematical Formulation of MOP

A multi-objective optimization problem (MOP) is defined as follows:

Introduction to Multi-objective Optimization

Mathematical Formulation of MOP

A multi-objective optimization problem (MOP) is defined as follows:

$$minimize\ F(x) = (f_1(x), ..., f_m(x))^T\ subject\ to\ x \epsilon \Omega \quad (1)$$

where Ω is the n-dimensional search space and x is the decision variable and $F : \Omega \rightarrow \Re^m$ where m is the number of objectives and $\Re^m$ is the objective space

Introduction to Multi-objective Optimization

Mathematical Formulation of MOP

A multi-objective optimization problem (MOP) is defined as follows:

$$minimize\ F(x) = (f_1(x), ..., f_m(x))^T\ subject\ to\ x \epsilon \Omega \quad (1)$$

where Ω is the n-dimensional search space and x is the decision variable and $F : \Omega \rightarrow \Re^m$ where m is the number of objectives and $\Re^m$ is the objective space

$$g_j(x) \leq 0; \quad j = 1, 2, ..., J \quad (2)$$

$$h_k(x) \leq 0; \quad k = 1, 2, ..., K \quad (3)$$

$$(x_i)^L \leq x_i \leq (x_i)^U; \quad i = 1, 2, ..., n \quad (4)$$

Outline

1 Introduction to Multi-objective Optimization
- Multi-objective Optimization
- Multi-objective Optimization Methods
- Goals in Multi-objective Optimization
- Evolutionary Multi-objective Optimization Frameworks

2 Multi-objective Evolutionary Algorithm based on Decomposition
- Introduction to MOEA/D
- Main Design Components of MOEA/D
- Studies on Weight Vector Generation Methods
- Studies on Decomposition Approaches
- Studies on Computational Resource Allocation
- Studies on Modifications in the Reproduction Operators
- Studies on Mating Selection and Replacement Mechanism
- Studies on Many-objective Optimization

3 Directions for Future Work

Multi-objective Optimization Methods

Classical Multi-objective Optimization

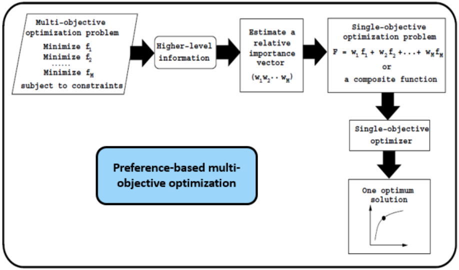


Figure: Classical Multi-objective Optimization.

Multi-objective Optimization Methods

Classical Multi-objective Optimization

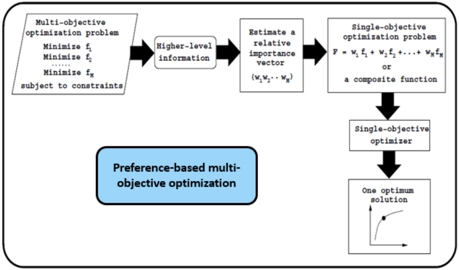


Figure: Classical Multi-objective Optimization.

Features of Classical MOO

- Advantage - Simple and adequate when a reliable relative preference vector is known

Multi-objective Optimization Methods

Classical Multi-objective Optimization

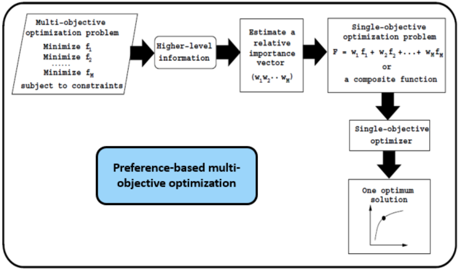


Figure: Classical Multi-objective Optimization.

Features of Classical MOO

- Advantage - Simple and adequate when a reliable relative preference vector is known
- Limitation 1 - Estimating a relative preference vector is difficult without any knowledge of possible consequences
- Limitation 2 - Trade-off solution obtained is largely sensitive to the relative preference vector
- Limitation 3 - Able to obtain only one solution at a time

Multi-objective Optimization Methods

Ideal Multi-objective Optimization

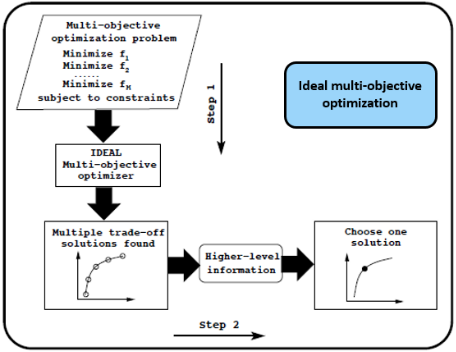


Figure: Ideal Multi-objective Optimization.

Multi-objective Optimization Methods

Ideal Multi-objective Optimization

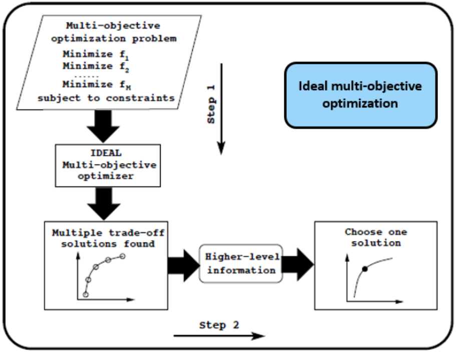


Figure: Ideal Multi-objective Optimization.

Features of Ideal MOO

- Advantage 1 - More methodical, practical and less subjective
- Advantage 2 - Higher-level information is used to evaluate and compare each of the obtained trade-off solutions to choose one solution
- Challenge - To obtain all the trade-off optimal solutions

Outline

1 Introduction to Multi-objective Optimization
- Multi-objective Optimization
- Multi-objective Optimization Methods
- Goals in Multi-objective Optimization
- Evolutionary Multi-objective Optimization Frameworks

2 Multi-objective Evolutionary Algorithm based on Decomposition
- Introduction to MOEA/D
- Main Design Components of MOEA/D
- Studies on Weight Vector Generation Methods
- Studies on Decomposition Approaches
- Studies on Computational Resource Allocation
- Studies on Modifications in the Reproduction Operators
- Studies on Mating Selection and Replacement Mechanism
- Studies on Many-objective Optimization

3 Directions for Future Work

Goals in Multi-objective Optimization

Cantilever Beam Design Problem

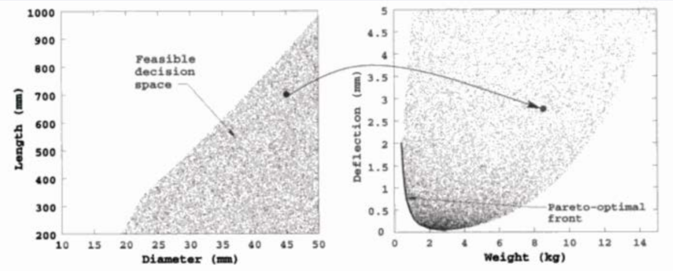


Figure: Decision variable space and corresponding objective space.

Goals in Multi-objective Optimization

Which solutions are optimal in MOP?

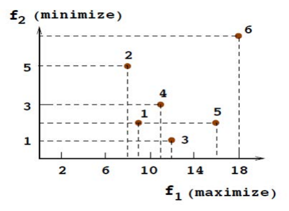


Figure: A set of six solutions.

Goals in Multi-objective Optimization

Which solutions are optimal in MOP?

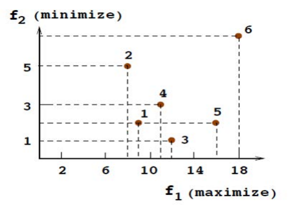


Figure: A set of six solutions.

Concept of Domination Principle

- Solution A dominates solution B if
- A is no worse than B in all objectives
- A is strictly better than B in at least one objective

Goals in Multi-objective Optimization

Which solutions are optimal in MOP?

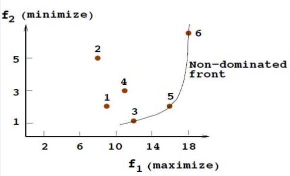


Figure: A set of six solutions.

Goals in Multi-objective Optimization

Which solutions are optimal in MOP?

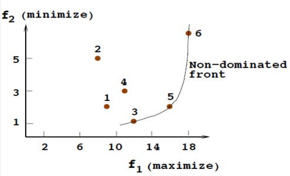


Figure: A set of six solutions.

- Non-dominated solutions when viewed together in objective space form a non-dominated front
- Global non-dominated front in the objective space is called Pareto-optimal front (PF)
- Corresponding set of solutions in the decision space is called Paretio-optimal set (PS)

Goals in Multi-objective Optimization

Pareto-Optimal Front

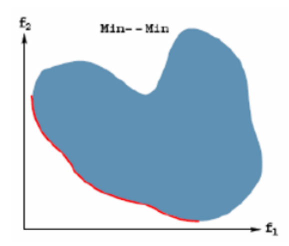


Figure: Pareto-Optimal Front.

Goals in Multi-objective Optimization

Pareto-Optimal Front

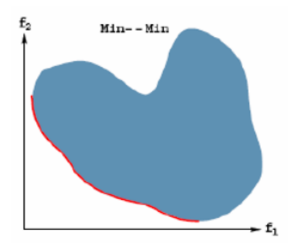


Figure: Pareto-Optimal Front.

Goals in MOO

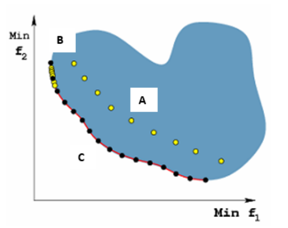


Figure: Goals in MOO.

Goals in Multi-objective Optimization

Pareto-Optimal Front

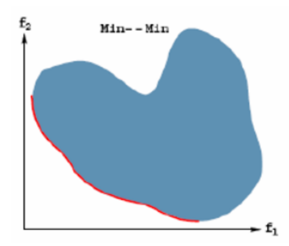


Figure: Pareto-Optimal Front.

Goals in MOO

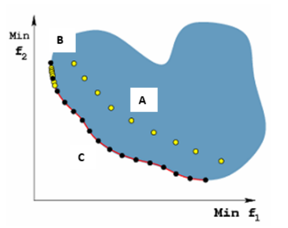


Figure: Goals in MOO.

Goals in Ideal MOO

Convergence, and Diversity

Multi-objective Evolutionary Algorithms

Why use Evolutionary Algorithms?

- Work with a population of solutions
- Search in multiple directions parallelly
- Can be modified to obtain PO solutions in a single run
- Different frameworks modify EAs differently to obtain PO solutions

Multi-objective Evolutionary Algorithms

Performance Comparison of MOEAs

- In EMO literature, different performance indicators have been proposed for the quantitative comparison of the performance of different algorithms
- Indicators such as **Generational Distance** measure only the *convergence*
- Indicators such as **Generalized Spread** measure only the *diversity*
- However, there are some indicators which measure both *convergence* and *diversity*
- **Inverted Generational Distance**
- **Hypervolume Indicator**

S. Jiang, Y. S. Ong, J. Zhang and L. Feng, Consistencies and Contradictions of Performance Metrics in Multiobjective Optimization, in IEEE Transactions on Cybernetics, 2014

Multi-objective Evolutionary Algorithms

Hypervolume Indicator

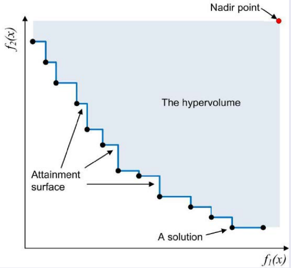


Figure: Volume of the objective space dominated by the approximation set.

S. Jiang, Y. S. Ong, J. Zhang and L. Feng, "Consistencies and Contradictions of Performance Metrics in Multiobjective Optimization", in IEEE Transactions on Cybernetics, 2014

Outline

1. Introduction to Multi-objective Optimization
   - Multi-objective Optimization
   - Multi-objective Optimization Methods
   - Goals in Multi-objective Optimization
   - Evolutionary Multi-objective Optimization Frameworks
2. Multi-objective Evolutionary Algorithm based on Decomposition
   - Introduction to MOEA/D
   - Main Design Components of MOEA/D
   - Studies on Weight Vector Generation Methods
   - Studies on Decomposition Approaches
   - Studies on Computational Resource Allocation
   - Studies on Modifications in the Reproduction Operators
   - Studies on Mating Selection and Replacement Mechanism
   - Studies on Many-objective Optimization
3. Directions for Future Work

Evolutionary Multi-objective Optimization Frameworks

Domination-based Framework

- A MOP is optimized by simultaneously optimizing all the objectives
- The assignment of fitness to solutions is based on Pareto-dominance principle
- An explicit diversity preservation scheme is necessary
- Example MOEAs - NSGA-II [1], SPEA2 [2]

Domination-based Framework

NSGA-II

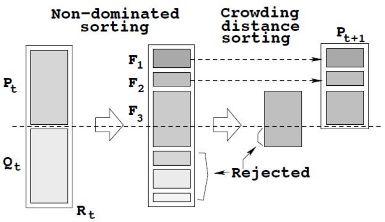


Figure: Replacement Procedure in NSGA-II.

Domination-based Framework

NSGA-II

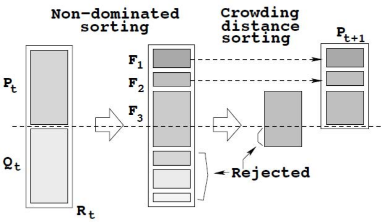


Figure: Replacement Procedure in NSGA-II.

- Non-dominated solutions are emphasized for progressing towards the PO front
- Elites are preserved to provide a faster and reliable convergence near the PO front
- Solutions with larger crowding distance are emphasized for maintaining a good diversity among obtained solutions

Evolutionary Multi-objective Optimization Frameworks

Indicator-based Framework

- A MOP is optimized by simultaneously optimizing all the objectives
- A performance indicator, in particular, the hypervolume (HV) indicator is used to measure the fitness of a solution
- HV indicator is the only known indicator that is compliant with the concept of Pareto-dominance
- Whenever a set of solutions dominates another set, its hypervolume indicator value is higher than the one of the latte
- This is the reason why most of the indicator based algorithms are based on HV
- Example MOEAs - IBEA [3], SIBEA [4]

Evolutionary Multi-objective Optimization Frameworks

Decomposition-based Framework

- MOP is decomposed into several single-objective optimization subproblems using scalarizing functions such as the weighted Tchebycheff
- All the subproblems are optimized in a single run using an EA
- Decomposition-based MOEAs utilize aggregated fitness value of solutions in the selection
- Example MOEAs - C-MOGA [6], MOEA/D [5]
- This framework has attracted the most attention of researchers in the EMO community in the last decade

Q. Zhang and H. Li, MOEA/D: A multiobjective evolutionary algorithm based on decomposition, IEEE Trans. Evol. Comput, 2007

Outline

1. Introduction to Multi-objective Optimization
   - Multi-objective Optimization
   - Multi-objective Optimization Methods
   - Goals in Multi-objective Optimization
   - Evolutionary Multi-objective Optimization Frameworks
2. Multi-objective Evolutionary Algorithm based on Decomposition
   - Introduction to MOEA/D
   - Main Design Components of MOEA/D
   - Studies on Weight Vector Generation Methods
   - Studies on Decomposition Approaches
   - Studies on Computational Resource Allocation
   - Studies on Modifications in the Reproduction Operators
   - Studies on Mating Selection and Replacement Mechanism
   - Studies on Many-objective Optimization
3. Directions for Future Work

MOEA/D

MOEA/D

- The idea of decomposition for solving MOPs had been implemented to a certain extent in several metaheuristics
- But it became popular with the introduction of MOEA/D by Zhang and Li in 2007 [5]

MOEA/D

MOEA/D

- The idea of decomposition for solving MOPs had been implemented to a certain extent in several metaheuristics
- But it became popular with the introduction of MOEA/D by Zhang and Li in 2007 [5]

Characteristic Features of MOEA/D

- MOP is decomposed into several scalar optimization subproblems, which are formulated by decomposition approach such as the Tchebycheff using uniformly distributed weight vectors
- All the subproblems are solved simultaneously in a collaborative manner by employing an EA and evolving a population of solutions
- A neighborhood relation is defined among the subproblems based on the distance between their weight vectors
- Local mating as well as local replacement is implemented in a steady-state manner

MOEA/D

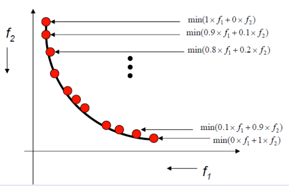


Figure: Illustration of Decomposition Approach.

MOEA/D

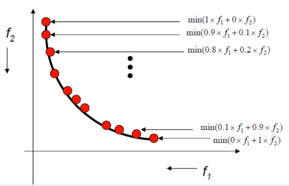


Figure: Illustration of Decomposition Approach.

$$
\begin{array}{ll}
\lambda^1 = (1, 0), & g(x, \lambda^1) = 1 \times f_1 + 0 \times f_2 \\
\lambda^2 = (0.9, 0.1) & g(x, \lambda^2) = 0.9 \times f_1 + 0.1 \times f_2 \\
\vdots & \\
\lambda^{11} = (0, 1) & g(x, \lambda^{11}) = 0 \times f_1 + 1 \times f_2
\end{array}
$$

Figure: Weight Vectors and Corresponding Subproblems.

MOEA/D

Why neighborhood relation is defined in MOEA/D?

- If weight vector λ_i and λ_j, we can say that their corresponding subproblems i.e., *min* $g(x, \lambda_i)$ and *min* $g(x, \lambda_j)$ are neighboring subproblems
- The neighboring subproblems should have similar solutions
- Thus, a neighborhood relation is defined among the subproblems based on the distance between their weight vectors
- Local mating as well as local replacement is implemented in a steady-state manner

Origin of MOEA/D

- The original MOEA/D framework [5] has its origins, particularly in the cellular multi-objective genetic algorithm (C-MOGA) presented by Murata and Gen in 2000 [6]

Origin of MOEA/D

- The original MOEA/D framework [5] has its origins, particularly in the cellular multi-objective genetic algorithm (C-MOGA) presented by Murata and Gen in 2000 [6]
- Similarities - Decomposition of MOP into several single-objective optimization subproblems, weight vector generation method, and the idea of neighborhood for mating selection

Origin of MOEA/D

- The original MOEA/D framework [5] has its origins, particularly in the cellular multi-objective genetic algorithm (C-MOGA) presented by Murata and Gen in 2000 [6]
- Similarities - Decomposition of MOP into several single-objective optimization subproblems, weight vector generation method, and the idea of neighborhood for mating selection
- Difference 1 - In C-MOGA, a newly generated offspring corresponding to an index is compared with only the current solution of the cell
- However, in MOEA/D, it is compared with its neighbors as well
- Thus, in MOEA/D, along with the mating neighborhood structure, there is a replacement neighborhood structure as well
- Difference 2 - In C-MOGA, only the WS scalarizing function was investigated while in MOEA/D, the WS, the TCH, and the PBI scalarizing functions were investigated

MOEA/D Framework

Input

- Taregt MOP
- N: the number of subproblems considered i.e., the population size;
- $\lambda_1, \lambda_2, \ldots, \lambda_N$: a set of N even distributed distributed weight vectors;
- T: the neighborhood size;

MOEA/D Framework

Input

- Taregt MOP
- N: the number of subproblems considered i.e., the population size;
- $\lambda_1, \lambda_2, \ldots, \lambda_N$: a set of N even distributed distributed weight vectors;
- T: the neighborhood size;

At each generation, MOEA/D maintains the following

- A population of N solutions $x_1, \ldots, x_N$, where x_i is the current solution to the ith subproblem, and $F(x_1), \ldots, F(x_N)$
- Reference point $z = (z_1, \ldots, z_m)$, where z_j is the best value found so far for objective $f_j \quad \forall j = 1, \ldots, m$

MOEA/D Framework

MOEA/D Framework

- **Step 1: Initialization**

MOEA/D Framework

- **Step 1: Initialization**
  - **Step 1.1 Neighborhood:** Determine the T closest weight vectors to each weight vector

MOEA/D Framework

- **Step 1: Initialization**
  - **Step 1.1** **Neighborhood:** Determine the T closest weight vectors to each weight vector
  - **Step 1.2** Initial Population - randomly generated
  - **Step 1.3** Function Evaluation
  - **Step 1.4** Reference point (z) - Initialization

MOEA/D Framework

- **Step 1: Initialization**
  - **Step 1.1 Neighborhood:** Determine the T closest weight vectors to each weight vector
  - **Step 1.2** Initial Population - randomly generated
  - **Step 1.3** Function Evaluation
  - **Step 1.4** Reference point (z) - Initialization
- **Step 2: Update**

MOEA/D Framework

- **Step 1: Initialization**
  - **Step 1.1 Neighborhood:** Determine the T closest weight vectors to each weight vector
  - **Step 1.2** Initial Population - randomly generated
  - **Step 1.3** Function Evaluation
  - **Step 1.4** Reference point (z) - Initialization
- **Step 2: Update**
  For $i = 1, \ldots, N$, **do**
  - **Step 2.1 Reproduction:** Offpsring generation via **local mating** using **genetic operators**

MOEA/D Framework

- **Step 1: Initialization**
  - **Step 1.1 Neighborhood:** Determine the T closest weight vectors to each weight vector
  - **Step 1.2** Initial Population - randomly generated
  - **Step 1.3** Function Evaluation
  - **Step 1.4** Reference point (z) - Initialization
- **Step 2: Update**
  For $i = 1, \ldots, N$, **do**
  - **Step 2.1 Reproduction:** Offpsring generation via **local mating** using **genetic operators**
  - **Step 2.2 Repair:** Offspring repair using heuristics

MOEA/D Framework

- **Step 1: Initialization**
  - **Step 1.1 Neighborhood:** Determine the T closest weight vectors to each weight vector
  - **Step 1.2** Initial Population - randomly generated
  - **Step 1.3** Function Evaluation
  - **Step 1.4** Reference point (z) - Initialization
- **Step 2: Update**
  For $i = 1, \ldots, N$, **do**
  - **Step 2.1 Reproduction:** Offpsring generation via **local mating** using **genetic operators**
  - **Step 2.2 Repair:** Offspring repair using heuristics
  - **Step 2.3 Function Evaluation:** Offspring

MOEA/D Framework

- **Step 1: Initialization**
  - **Step 1.1 Neighborhood:** Determine the T closest weight vectors to each weight vector
  - **Step 1.2** Initial Population - randomly generated
  - **Step 1.3** Function Evaluation
  - **Step 1.4** Reference point (z) - Initialization
- **Step 2: Update**
  For $i = 1, \ldots, N$, **do**
  - **Step 2.1 Reproduction:** Offpsring generation via **local mating** using **genetic operators**
  - **Step 2.2 Repair:** Offspring repair using heuristics
  - **Step 2.3 Function Evaluation:** Offspring
  - **Step 2.4 Update of Reference point (z):** Offspring is used to update z

MOEA/D Framework

- **Step 1: Initialization**
  - **Step 1.1 Neighborhood:** Determine the T closest weight vectors to each weight vector
  - **Step 1.2** Initial Population - randomly generated
  - **Step 1.3** Function Evaluation
  - **Step 1.4** Reference point (z) - Initialization
- **Step 2: Update**
  For $i = 1, \ldots, N$, **do**
  - **Step 2.1 Reproduction:** Offpsring generation via **local mating** using **genetic operators**
  - **Step 2.2 Repair:** Offspring repair using heuristics
  - **Step 2.3 Function Evaluation:** Offspring
  - **Step 2.4 Update of Reference point (z):** Offspring is used to update z
  - **Step 2.5 Replacement:** If offspring is a better solution that the existing solutions to neighboring subproblems, then they are replaced by the offspring

MOEA/D Framework

- **Step 1: Initialization**
  - **Step 1.1 Neighborhood:** Determine the T closest weight vectors to each weight vector
  - **Step 1.2** Initial Population - randomly generated
  - **Step 1.3** Function Evaluation
  - **Step 1.4** Reference point (z) - Initialization
- **Step 2: Update**
  For $i = 1, \ldots, N$, **do**
  - **Step 2.1 Reproduction:** Offpsring generation via **local mating** using **genetic operators**
  - **Step 2.2 Repair:** Offspring repair using heuristics
  - **Step 2.3 Function Evaluation:** Offspring
  - **Step 2.4 Update of Reference point (z):** Offspring is used to update z
  - **Step 2.5 Replacement:** If offspring is a better solution that the existing solutions to neighboring subproblems, then they are replaced by the offspring
- **Step 3: Stopping Criteria**
  If termination criterion is satisfied, then obtain approximation to PO the PF else go to **Step 2**

MOEA/D

MOEA/D essentially consists of:

- a set of evenly spaced weight vectors
- a decompositionmethod (scalarizing function)
- a neighbourhood of a fixed size defined in the weight vector space
- genetic operators for reproduction (crossover and mutation)
- neighborhood mating selection
- steady state neighborhood replacement

Outline

1 Introduction to Multi-objective Optimization
- Multi-objective Optimization
- Multi-objective Optimization Methods
- Goals in Multi-objective Optimization
- Evolutionary Multi-objective Optimization Frameworks

2 Multi-objective Evolutionary Algorithm based on Decomposition
- Introduction to MOEA/D
- Main Design Components of MOEA/D
- Studies on Weight Vector Generation Methods
- Studies on Decomposition Approaches
- Studies on Computational Resource Allocation
- Studies on Modifications in the Reproduction Operators
- Studies on Mating Selection and Replacement Mechanism
- Studies on Many-objective Optimization

3 Directions for Future Work

Main Design Components of MOEA/D

The main design components of MOEA/D are:

Main Design Components of MOEA/D

The main design components of MOEA/D are:

- Weight vector generation method
- Decomposition method
- Mating selection mechanism
- Reproduction operators
- Replacement procedure
- Computational resource allocation strategy

Different Lines of Research on Decomposition-based MOEAs

- Address the drawbacks in the design components of the original MOEA/D

Different Lines of Research on Decomposition-based MOEAs

- Address the drawbacks in the design components of the original MOEA/D
- Enhance the performance of MOEA/D

Different Lines of Research on Decomposition-based MOEAs

- Address the drawbacks in the design components of the original MOEA/D
- Enhance the performance of MOEA/D
- Present new decomposition-based MOEAs

Different Lines of Research on Decomposition-based MOEAs

- Address the drawbacks in the design components of the original MOEA/D
- Enhance the performance of MOEA/D
- Present new decomposition-based MOEAs
- Extend decomposition-based MOEAs to other category of problems such as
  - constrained optimization
  - many-objective optimization
  - preference incorporation
  - real-world optimization

A. Trivedi, D. Srinivasan, K. Sanyal and A. Ghosh, "A Survey of Multiobjective Evolutionary Algorithms Based on Decomposition," in IEEE Transactions on Evolutionary Computation, 2017

Outline

1 Introduction to Multi-objective Optimization
- Multi-objective Optimization
- Multi-objective Optimization Methods
- Goals in Multi-objective Optimization
- Evolutionary Multi-objective Optimization Frameworks

2 Multi-objective Evolutionary Algorithm based on Decomposition
- Introduction to MOEA/D
- Main Design Components of MOEA/D
- Studies on Weight Vector Generation Methods
- Studies on Decomposition Approaches
- Studies on Computational Resource Allocation
- Studies on Modifications in the Reproduction Operators
- Studies on Mating Selection and Replacement Mechanism
- Studies on Many-objective Optimization

3 Directions for Future Work

Studies on Weight Vector Generation Methods

Simplex Lattice Method

- Original MOEA/D and many of its subsequent variants use Das and Dennis's simplex-lattice design method
- In this method, weight vectors are systematically sampled from a simplex
- The population size N and weight vectors $\lambda_1, \lambda_2, \ldots, \lambda_N$ are controlled by m and an integer H
- $H \geq 0$ is the number of divisions along each co-ordinate
- $\lambda_1, \lambda_2, \ldots, \lambda_N$ are all the weight vectors in which each individual weight takes a value from $0/H, 1/H, \ldots, H/H$
- Therefore, the number of weight vectors generated are $N = \binom{H+m-1}{m-1}$

Studies on Weight Vector Generation Methods

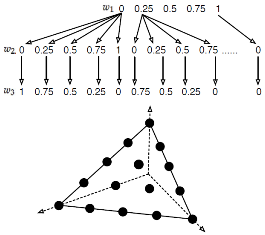


Figure: 15 weight vectors sampled on a simplex for $m = 3$ and $H = 4$

Studies on Weight Vector Generation Methods

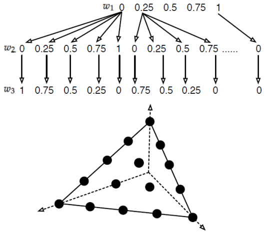


Figure: 15 weight vectors sampled on a simplex for $m = 3$ and $H = 4$

Limitations of Simplex Lattice Method

- Population size grows exponentially with the number of objectives
- For example, if H is set a constant 20, then population size will be 21, 231, 1771, and 10626 for 2-5 objectives
- Setting of population size is not flexible
- Distribution of weight vectors is not very uniform for objectives more than three

Studies on Weight Vector Generation Methods

UMOEA/D

- Tan *et al.* [7] proposed MOEA/D + uniform design, termed UMOEA/D for MaOPs
- In UMOEA/D, the goal is to find a set of weight vectors P - $\lambda_1, \lambda_2, \ldots, \lambda_N$ which are distributed uniformly
- Let $M(P)$ be a measure of non-uniformity of P, then the goal is to minimize $M(P)$ and obtain $P*$
- UMOEA/D utilizes a discrepancy measure and obtains the weight vectors which yield minimum discrepancy in their distribution

Tan *et al.*, "MOEA/D + uniform design: A new version of MOEA/D for optimization problems with many objectives," Computers & Operations Research, 2013

Studies on Weight Vector Generation Methods

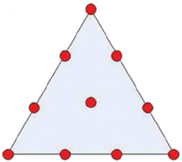


Figure: Distribution with Simplex Lattice Design

Studies on Weight Vector Generation Methods

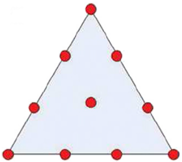


Figure: Distribution with Simplex Lattice Design

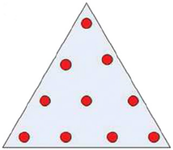


Figure: Distribution in UMOEA/D

Studies on Weight Vector Generation Methods

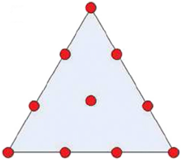


Figure: Distribution with Simplex Lattice Design

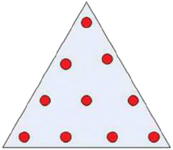


Figure: Distribution in UMOEA/D

UMOEA/D features

- The weight vectors are more uniformly distributed than the simplex lattice design
- Population size is decoupled with the number of objectives
- UMOEA/D was found to perform better than MOEA/D and NSGA-II
- DLTZ1-4 test problems with 3-5 objectives, knapsack problems with 2-4 objectives, and problems with complicated PS shapaes

Studies on Weight Vector Generation Methods

Generalized Decomposition

- Giagkiozis *et al.* [8] presented a novel method known as generalized decomposition (gD)
- The authors presented the gD method with respect to the TCH scalarizing function
- The gD method assumes that a reference PF exists, and then for the reference PO solutions, it obtains the optimal set of weight vectors
- A limitation of the gD method is that it requires a *priori* information about the PF geometry
- If a *priori* information about the PF geometry is unavailable, a reference PF with affine geometry can be assumed
- The gD method is remarkably better than the simplex lattice design and uniform random sampling for a wide range of PF geometries, and different number of objective functions in generating evenly distributed points

I. Giagkiozis, R. Purshouse, and P. Fleming, "Generalized decomposition," in Evolutionary Multi-Criterion Optimization. 7th International Conference, EMO 2013.

I. Giagkiozis, R. Purshouse, and P. Fleming, "Generalized decomposition and cross entropy methods formany-objective optimization," Information Sciences, 2014.

Studies on Weight Vector Generation Methods

- Giagkiozis *et al.* modified MOEA/D by incorporating gD and defining neighborhood relationship in objective space
- The resulting algorithm was named MAEA-gD

Studies on Weight Vector Generation Methods

- Giagkiozis *et al.* modified MOEA/D by incorporating gD and defining neighborhood relationship in objective space
- The resulting algorithm was named MAEA-gD

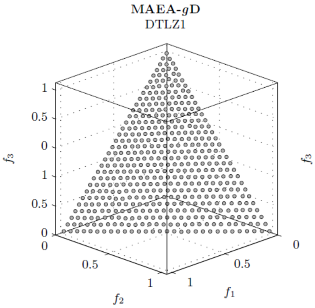


Figure: Attained PF

Studies on Weight Vector Generation Methods

- Giagkiozis *et al.* modified MOEA/D by incorporating gD and defining neighborhood relationship in objective space
- The resulting algorithm was named MAEA-gD

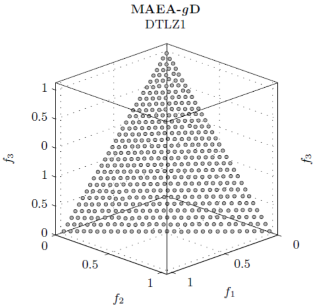


Figure: Attained PF

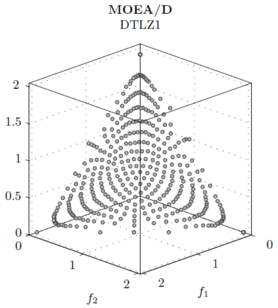


Figure: Attained PF

I. Giagkiozis, R. Purshouse, and P. Fleming, "Generalized decomposition," in Evolutionary Multi-Criterion Optimization. 7th International Conference, EMO 2013.

Studies on Weight Vector Generation Methods

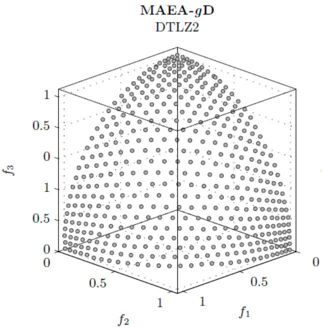


Figure: Attained PF

Studies on Weight Vector Generation Methods

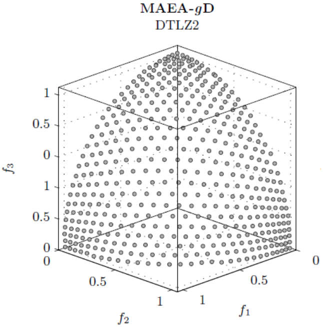


Figure: Attained PF

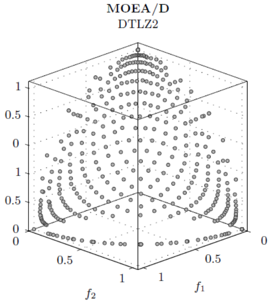


Figure: Attained PF

Studies on Weight Vector Generation Methods

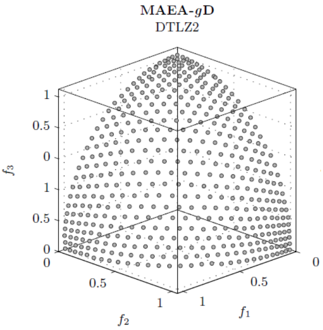


Figure: Attained PF

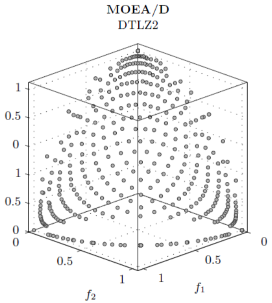


Figure: Attained PF

- There is a limitation in the comparative study
- For fair comparison, MOEA/D should have been compared with MOEA/D + gD

Studies on Weight Vector Generation Methods

Two Layer Weight Vector Generation

- With simplex lattice design, in order to have intermediate weight vectors within the simplex, the setting of $H \geq m$
- In MaOP, for $m = 7$, even $H = m$ will result in 1716 weight vectors
- If this issue is tackled by lowering H i.e., $H < m$, then it will make all weight vectors lie sparsely along the boundary of the simplex
- Li *et al.* [9] presented a two-layer weight vector generation method for MaOPs
- In this method, a set of N_1 and N_2 weight vectors are separately generated for boundary and inside layers, respectively, using the simplex-lattice design method (such that $N_1 + N_2 = N$)

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 Transactions on Evolutionary Computation, 2014.

Studies on Weight Vector Generation Methods

- The set of weight vectors for the boundary and the inside layer are then combined to represent the final weight vector set

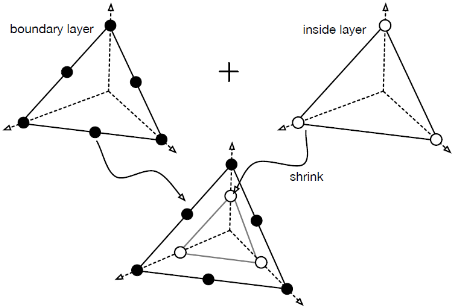


Figure: Two layer weight vector generation, with $m = 3$, $H_1 = 2$ for the boundary layer and $H_2 = 1$ for the inside layer

- This method overcomes the limitation of the simplex-lattice design method in generating relatively small number of evenly spread weight vectors for MaOPs

Studies on Weight Vector Generation Methods

Summary

- The distribution of PO solutions is highly dependent on distribution of the weight vectors
- The weight vector generation method in UMOEA/D, gD, and two-layer generation method decouple the population size with the number of objectives
- The gD method has been found to be significantly better than the the simplex lattice design and uniform random sampling on problems with 3 or more objectives
- The two-layer weight vector generation method [9] extends the simplex-lattice design method for many-objective optimization

Outline

1 Introduction to Multi-objective Optimization
- Multi-objective Optimization
- Multi-objective Optimization Methods
- Goals in Multi-objective Optimization
- Evolutionary Multi-objective Optimization Frameworks

2 Multi-objective Evolutionary Algorithm based on Decomposition
- Introduction to MOEA/D
- Main Design Components of MOEA/D
- Studies on Weight Vector Generation Methods
- Studies on Decomposition Approaches
- Studies on Computational Resource Allocation
- Studies on Modifications in the Reproduction Operators
- Studies on Mating Selection and Replacement Mechanism
- Studies on Many-objective Optimization

3 Directions for Future Work

Studies on Decomposition Approaches

Decomposition Frameworks in Original MOEA/D

- The original MOEA/D was tested with the WS, the TCH, and the PBI decomposition approaches
- WS - In this approach, the *i*th subproblem is defined in the form

$$minimize\ g^{ws}(x|\lambda_i) = \sum_{j=1}^{m} \lambda_i^j f_j(x) \tag{5}$$

Studies on Decomposition Approaches

Decomposition Frameworks in Original MOEA/D

- The original MOEA/D was tested with the WS, the TCH, and the PBI decomposition approaches
- WS - In this approach, the *i*th subproblem is defined in the form

$$minimize\ g^{ws}(x|\lambda_i) = \sum_{j=1}^{m} \lambda_i^j f_j(x) \quad (5)$$

- TCH - In this approach, the *i*th subproblem is defined in the form

$$minimize\ g^{te}(x|\lambda_i, z^*) = \max_{1 \leq j \leq m} \{\lambda_i^j |f_j(x) - z_j^*|\} \quad (6)$$

Studies on Decomposition Approaches

Decomposition Frameworks in Original MOEA/D

- The original MOEA/D was tested with the WS, the TCH, and the PBI decomposition approaches
- WS - In this approach, the *i*th subproblem is defined in the form

$$minimize\ g^{ws}(x|\lambda_i) = \sum_{j=1}^{m} \lambda_i^j f_j(x) \quad (5)$$

- TCH - In this approach, the *i*th subproblem is defined in the form

$$minimize\ g^{te}(x|\lambda_i, z^*) = \max_{1 \leq j \leq m} \{\lambda_i^j |f_j(x) - z_j^*|\} \quad (6)$$

where $z^* = (z_1^*, ..., z_m^*)^T$ is the ideal reference point with

$$z_j^* < \min\{f_j(x)|x \in \Omega\} \quad \text{for } j = 1, 2, ..., m. \quad (7)$$

Studies on Decomposition Approaches

Decomposition Frameworks in Original MOEA/D

- PBI - In this approach, the *i*th subproblem is defined in the form

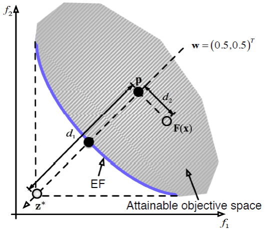
$$\begin{aligned} &\text{minimize } g^{pbi}(x|\lambda_i, z^*) = d_1 + \theta d_2 \\ &\text{where } d_1 = \frac{\|(F(x) - z^*)^T \lambda_i\|}{\|\lambda_i\|} \\ &d_2 = \|F(x) - (z^* - d_1 \frac{\lambda_i}{\|\lambda_i\|})\| \end{aligned} \tag{8}$$

Studies on Decomposition Approaches

Decomposition Frameworks in Original MOEA/D

- PBI - In this approach, the ith subproblem is defined in the form

$$\begin{aligned} \text{minimize } g^{pbi}(x|\lambda_i, z^*) &= d_1 + \theta d_2 \\ \text{where } d_1 &= \frac{\|(F(x) - z^*)^T \lambda_i\|}{\|\lambda_i\|} \\ d_2 &= \|F(x) - (z^* - d_1 \frac{\lambda_i}{\|\lambda_i\|})\| \end{aligned} \tag{8}$$



Studies on Decomposition Approaches

Limitations in the Traditional Decomposition Approaches

- Not very suitable for MaOPs [10]
- The improvement region corresponding to these methods may be too large in some problems, resulting in low population diversity [11]
- In the PBI approach, the parameter θ needs to be tuned

Studies on Decomposition Approaches

Limitations in the Traditional Decomposition Approaches

- Not very suitable for MaOPs [10]
- The improvement region corresponding to these methods may be too large in some problems, resulting in low population diversity [11]
- In the PBI approach, the parameter θ needs to be tuned

General Limitation of Decomposition-based MOEAs

- The performance of D-MOEAs on any particular problem is dependent on the decomposition method
- Thus, it is a challenge to determine an appropriate decomposition method for a particular problem

Studies on Improved Decomposition Approaches

Inverted PBI approach (IPBI)

Studies on Improved Decomposition Approaches

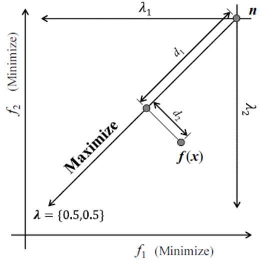
Inverted PBI approach (IPBI)

- In [12], Sato argued that the conventional decomposition approaches encounter difficulty in approximating widely spread PF in some problems like MOKPs
- Sato [12] extended the conventional PBI approach and proposed IPBI approach
- In conventional approaches like the TCH and the PBI, solutions are evolved towards the reference point z by minimizing the scalarizing function value
- In IPBI, solutions are evolved from the nadir point n by maximizing the scalarizing function value

Studies on Improved Decomposition Approaches

Inverted PBI approach (IPBI)

- In [12], Sato argued that the conventional decomposition approaches encounter difficulty in approximating widely spread PF in some problems like MOKPs
- Sato [12] extended the conventional PBI approach and proposed IPBI approach
- In conventional approaches like the TCH and the PBI, solutions are evolved towards the reference point z by minimizing the scalarizing function value
- In IPBI, solutions are evolved from the nadir point n by maximizing the scalarizing function value



Studies on Improved Decomposition Approaches

Inverted PBI approach (IPBI)

Studies on Improved Decomposition Approaches

Inverted PBI approach (IPBI)

- In IPBI, the *i*th subproblem is defined in the form

$$maximize\ g^{ipbi}(x|\lambda_i, n) = d_1 - \theta d_2 \tag{9}$$

Studies on Improved Decomposition Approaches

Inverted PBI approach (IPBI)

- In IPBI, the *i*th subproblem is defined in the form

$$maximize\ g^{ipbi}(x|\lambda_i, n) = d_1 - \theta d_2 \tag{9}$$

- A solution having large d_1 and small d_2 is considered a better solution
- The experimental study on MOKPs and WFG4 problem [13], with 2-8 objectives, illustrated that the IPBI approach can better approximate widely spread PF

H. Sato, "Analysis of inverted PBI and comparison with other scalarizing functions in decomposition based MOEAs," Journal of Heuristics, 2015.

Studies on Decomposition Approaches

- Sato [10] investigated the robustness of scalarizing functions on problems with different PF shapes
- Concave PF - conventional WFG4 problem
- Linear and Convex PF - transformed WFG4 problems
- Discontinuous PF - WFG2 problem

Studies on Decomposition Approaches

- Sato [10] investigated the robustness of scalarizing functions on problems with different PF shapes
- Concave PF - conventional WFG4 problem
- Linear and Convex PF - transformed WFG4 problems
- Discontinuous PF - WFG2 problem

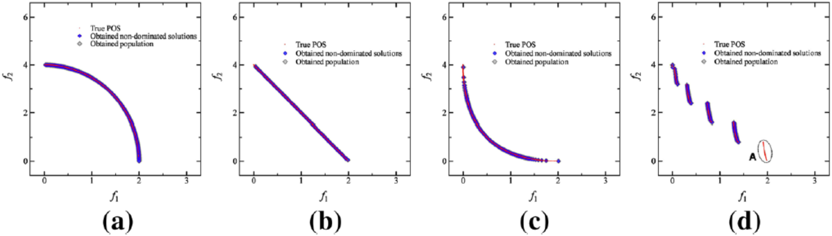


Figure: Solutions obtained using **TCH** in problems with four different shapes of Pareto front. a. Concave, b. Linear, c. Convex, d. Discontinuos

Studies on Decomposition Approaches

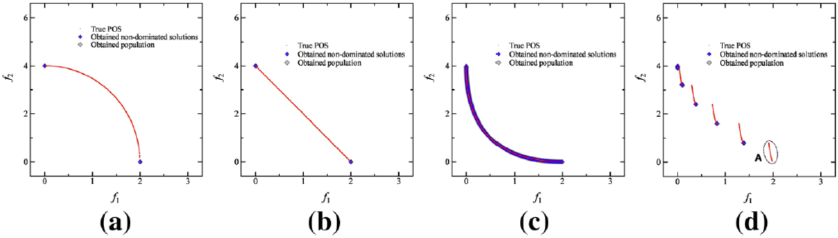


Figure: Solutions obtained by **WS** in problems with four different shapes of Pareto front. a. Concave, b. Linear, c. Convex, d. Discontinuous

Studies on Decomposition Approaches

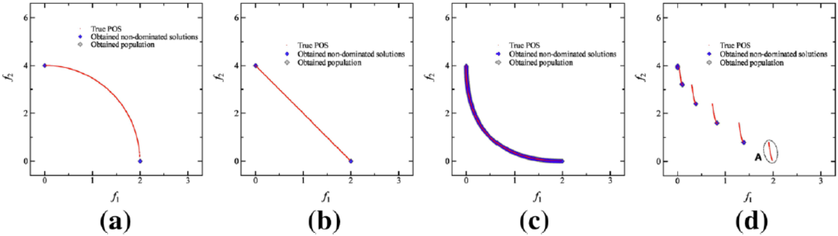


Figure: Solutions obtained by **WS** in problems with four different shapes of Pareto front. a. Concave, b. Linear, c. Convex, d. Discontinuous

- WS approach failed completely on problems with concave, linear, and discontinuous PF
- Performed very well on problem with convex PF

Studies on Decomposition Approaches

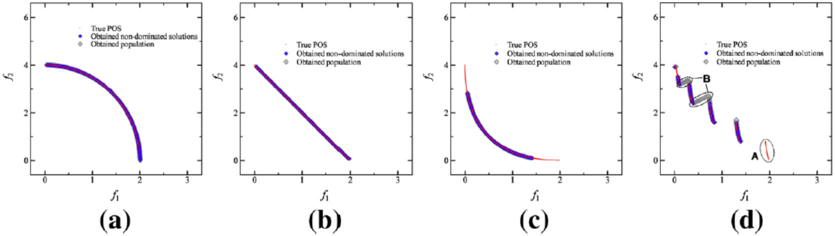


Figure: Solutions obtained by **PBI** in problems with four different shapes of Pareto front. a. Concave, b. Linear, c. Convex, d. Discontinuous

Studies on Decomposition Approaches

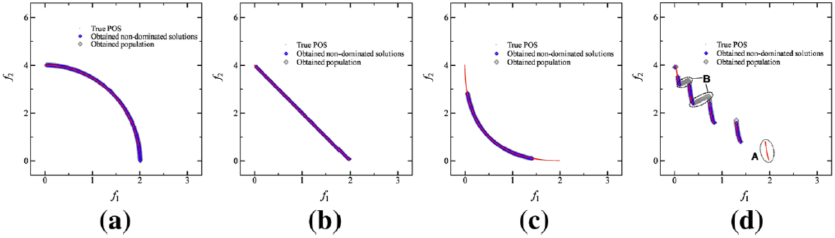


Figure: Solutions obtained by **PBI** in problems with four different shapes of Pareto front. a. Concave, b. Linear, c. Convex, d. Discontinuous

- PBI approach could not find solutions on extremes of PF for convex PF problem
- On discontinuous PF problem, it could not find some portions of PF and also returned several dominated solutions

Studies on Decomposition Approaches

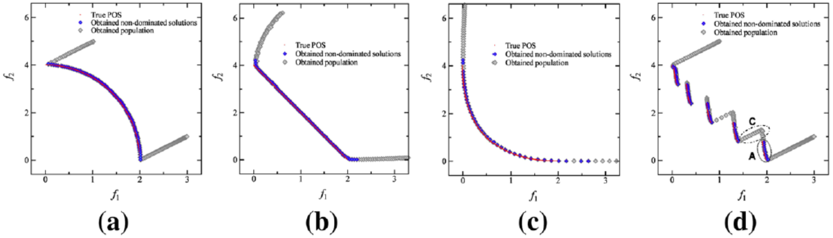


Figure: Solutions obtained by **IPBI** ($\theta = 10$) in problems with four different shapes of Pareto front. a. Concave, b. Linear, c. Convex, d. Discontinuous

Studies on Decomposition Approaches

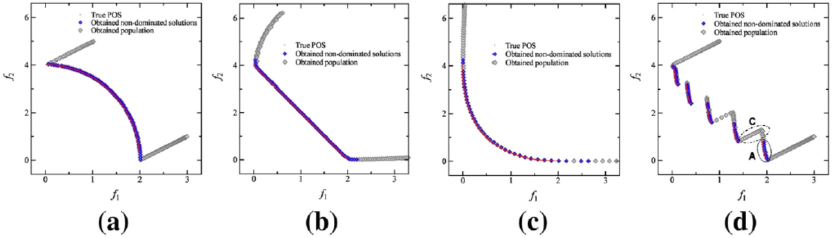


Figure: Solutions obtained by **IPBI** ($\theta = 10$) in problems with four different shapes of Pareto front. a. Concave, b. Linear, c. Convex, d. Discontinuous

- Although IPBI approach could cover the entire PF in all problems, it also returned many dominated solutions
- Another limitation is that it requires the tuning of the parameter θ

Studies on Improved Decomposition Approaches

Yang *et al*. [14] investigated the influence of the penalty parameter θ in PBI

Studies on Improved Decomposition Approaches

Yang *et al*. [14] investigated the influence of the penalty parameter θ in PBI

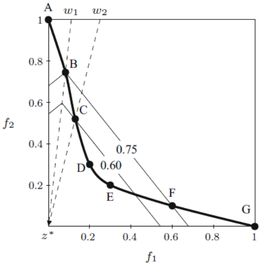


Figure: Illustration of insufficient penalty for weight vectors where $w_1 = (0.9, 0.1)$, $w_2 = (0.8, 0.2)$ and $\theta = 1$ in PBI

Studies on Improved Decomposition Approaches

Yang *et al.* [14] investigated the influence of the penalty parameter θ in PBI

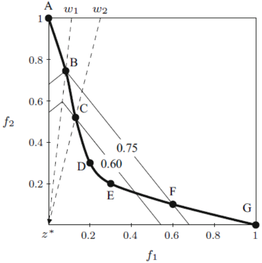


Figure: Illustration of insufficient penalty for weight vectors where $w_1 = (0.9, 0.1)$, $w_2 = (0.8, 0.2)$ and $\theta = 1$ in PBI

- B and C are ideal optimal points for subproblems associated with w_1 and w_2

Studies on Improved Decomposition Approaches

Yang *et al.* [14] investigated the influence of the penalty parameter θ in PBI

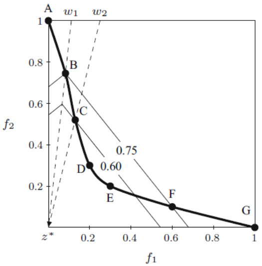


Figure: Illustration of insufficient penalty for weight vectors where $w_1 = (0.9, 0.1)$, $w_2 = (0.8, 0.2)$ and $\theta = 1$ in PBI

- B and C are ideal optimal points for subproblems associated with w_1 and w_2
- However, solution B of the subproblem associated with w_1 will be replaced by C

Studies on Improved Decomposition Approaches

Yang *et al.* [14] investigated the influence of the penalty parameter θ in PBI

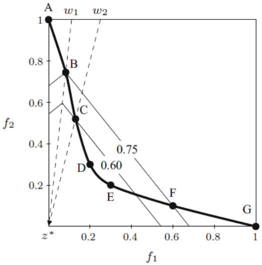


Figure: Illustration of insufficient penalty for weight vectors where $w_1 = (0.9, 0.1)$, $w_2 = (0.8, 0.2)$ and $\theta = 1$ in PBI

- B and C are ideal optimal points for subproblems associated with w_1 and w_2
- However, solution B of the subproblem associated with w_1 will be replaced by C
- $g^{pbi}(C|w_1, z^*) = 0.6$ is smaller than $g^{pbi}(B|w_1, z^*) = 0.75$

Studies on Improved Decomposition Approaches

Yang *et al.* [14] investigated the influence of the penalty parameter θ in PBI

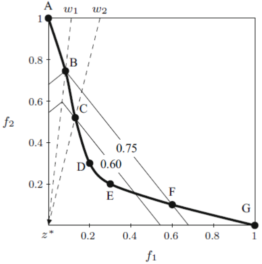


Figure: Illustration of insufficient penalty for weight vectors where $w_1 = (0.9, 0.1)$, $w_2 = (0.8, 0.2)$ and $\theta = 1$ in PBI

- B and C are ideal optimal points for subproblems associated with w_1 and w_2
- However, solution B of the subproblem associated with w_1 will be replaced by C
- $g^{pbi}(C|w_1, z^*) = 0.6$ is smaller than $g^{pbi}(B|w_1, z^*) = 0.75$
- Due to insufficient penalty, POF points far away from the obtained ideal point are replaced by those close to the ideal point

Studies on Improved Decomposition Approaches

Yang *et al.* [14] investigated the influence of the penalty parameter θ in PBI

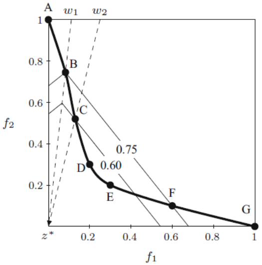


Figure: Illustration of insufficient penalty for weight vectors where $w_1 = (0.9, 0.1)$, $w_2 = (0.8, 0.2)$ and $\theta = 1$ in PBI

- B and C are ideal optimal points for subproblems associated with w_1 and w_2
- However, solution B of the subproblem associated with w_1 will be replaced by C
- $g^{pbi}(C|w_1, z^*) = 0.6$ is smaller than $g^{pbi}(B|w_1, z^*) = 0.75$
- Due to insufficient penalty, POF points far away from the obtained ideal point are replaced by those close to the ideal point
- The impact is even worse on extremely convex problems with sharp peak and low tail

S. Yang, S. Jiang, and Y. Jiang, "Improving the multiobjective evolutionary algorithm based on decomposition with new penalty schemes," Soft Computing, pp. 115, 2016.

Studies on Improved Decomposition Approaches

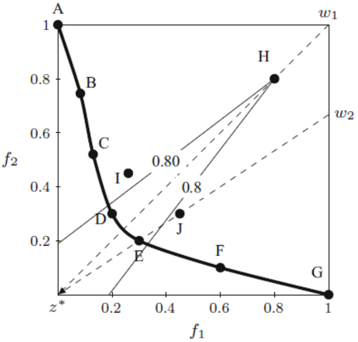


Figure: Illustration of excessive penalty for weight vectors where $w_1 = (0.5, 0.5)$, $w_2 = (0.4, 0.6)$ and $\theta = 5$ in PBI

Studies on Improved Decomposition Approaches

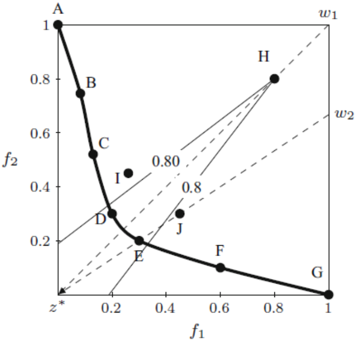


Figure: Illustration of excessive penalty for weight vectors where $w_1 = (0.5, 0.5)$, $w_2 = (0.4, 0.6)$ and $\theta = 5$ in PBI

- For w_1, PBI is expected to drive the search toward the PF segment between D and E

Studies on Improved Decomposition Approaches

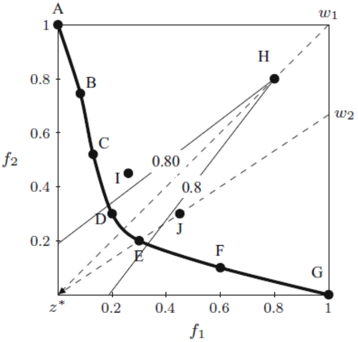


Figure: Illustration of excessive penalty for weight vectors where $w_1 = (0.5, 0.5)$, $w_2 = (0.4, 0.6)$ and $\theta = 5$ in PBI

- For w_1, PBI is expected to drive the search toward the PF segment between D and E
- At the early stage of search, the subproblems associated with w_1 and w_2 find their best solution H and J, respectively

Studies on Improved Decomposition Approaches

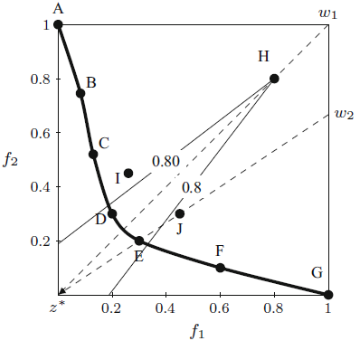


Figure: Illustration of excessive penalty for weight vectors where $w_1 = (0.5, 0.5)$, $w_2 = (0.4, 0.6)$ and $\theta = 5$ in PBI

- For w_1, PBI is expected to drive the search toward the PF segment between D and E
- At the early stage of search, the subproblems associated with w_1 and w_2 find their best solution H and J, respectively
- J is much closer than H to the expected PF segment

Studies on Improved Decomposition Approaches

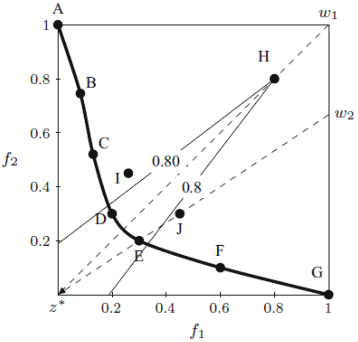


Figure: Illustration of excessive penalty for weight vectors where $w_1 = (0.5, 0.5)$, $w_2 = (0.4, 0.6)$ and $\theta = 5$ in PBI

- For w_1, PBI is expected to drive the search toward the PF segment between D and E
- At the early stage of search, the subproblems associated with w_1 and w_2 find their best solution H and J, respectively
- J is much closer than H to the expected PF segment
- However, for subproblem associated with w_1, its current solution H will not be replaced by J

Studies on Improved Decomposition Approaches

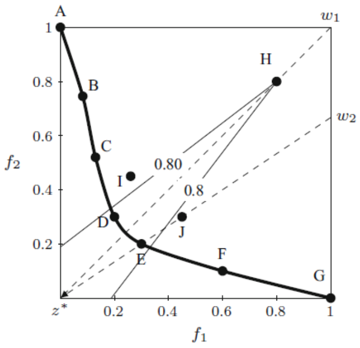


Figure: Illustration of excessive penalty for weight vectors where $w_1 = (0.5, 0.5)$, $w_2 = (0.4, 0.6)$ and $\theta = 5$ in PBI

- For w_1, PBI is expected to drive the search toward the PF segment between D and E
- At the early stage of search, the subproblems associated with w_1 and w_2 find their best solution H and J, respectively
- J is much closer than H to the expected PF segment
- However, for subproblem associated with w_1, its current solution H will not be replaced by J
- Due to excessive penalty, solutions close to weight vectors but far away from the OF may be preferred over those close to the PF but far away from the weight vectors

Studies on Improved Decomposition Approaches

Yang *et al*. [14] suggested two new penalty schemes to overcome the problem of tuning the parameter θ in PBI

Studies on Improved Decomposition Approaches

Yang *et al*. [14] suggested two new penalty schemes to overcome the problem of tuning the parameter θ in PBI

Adaptive penalty scheme (APS) and Subproblem-based penalty scheme (SPS)

Studies on Improved Decomposition Approaches

Yang *et al.* [14] suggested two new penalty schemes to overcome the problem of tuning the parameter θ in PBI

Adaptive penalty scheme (APS) and Subproblem-based penalty scheme (SPS)

- In APS, θ is linearly increased with the number of generations from $\theta_{min}(=1)$ to $\theta_{max}(=10)$
- A small value of θ emphasizes convergence in the initial stage while θ is gradually increased to emphasize diversity later

Studies on Improved Decomposition Approaches

Yang *et al.* [14] suggested two new penalty schemes to overcome the problem of tuning the parameter θ in PBI

Adaptive penalty scheme (APS) and Subproblem-based penalty scheme (SPS)

- In APS, θ is linearly increased with the number of generations from $\theta_{min}(= 1)$ to $\theta_{max}(= 10)$
- A small value of θ emphasizes convergence in the initial stage while θ is gradually increased to emphasize diversity later
- In SPS, each subproblem is assigned a different (but fixed) penalty value
- In particular, the extreme subproblems are assigned a larger i.e., stricter penalty value

Studies on Improved Decomposition Approaches

Yang *et al*. [14] suggested two new penalty schemes to overcome the problem of tuning the parameter θ in PBI

Adaptive penalty scheme (APS) and Subproblem-based penalty scheme (SPS)

- In APS, θ is linearly increased with the number of generations from $\theta_{min}(= 1)$ to $\theta_{max}(= 10)$
- A small value of θ emphasizes convergence in the initial stage while θ is gradually increased to emphasize diversity later
- In SPS, each subproblem is assigned a different (but fixed) penalty value
- In particular, the extreme subproblems are assigned a larger i.e., stricter penalty value
- Both the penalty schemes help to improve the performance of MOEA/D-PBI, particularly in terms of better coverage of the PF
- **Limitation** - The proposed schemes were tested on only few selected MOPs

Studies on Improved Decomposition Approaches

- Wang *et al*. [11] defined the ***improvement region*** of a current solution x_i for subproblem i as the region in the objective space, in which if a new solution y is produced then the current solution x_i can get replaced

Studies on Improved Decomposition Approaches

- Wang *et al.* [11] defined the ***improvement region*** of a current solution x_i for subproblem i as the region in the objective space, in which if a new solution y is produced then the current solution x_i can get replaced

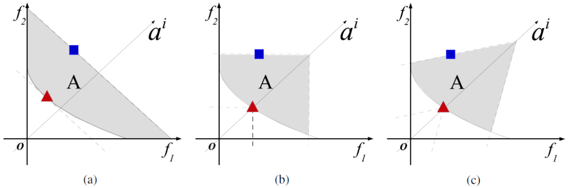


Figure: Illustrations of the improvement regions for : (a) WS, (b) TCH, and (c) PBI. In each sub-figure, region A is the improvement region. The square point is the current solution of subproblem i with the weight vector a^i, the triangle point is its optimal solution and the dash line is its contour

Studies on Improved Decomposition Approaches

- Wang *et al.* [11] defined the ***improvement region*** of a current solution x_i for subproblem i as the region in the objective space, in which if a new solution y is produced then the current solution x_i can get replaced

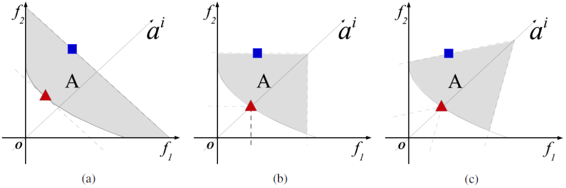


Figure: Illustrations of the improvement regions for : (a) WS, (b) TCH, and (c) PBI. In each sub-figure, region A is the improvement region. The square point is the current solution of subproblem i with the weight vector a^i, the triangle point is its optimal solution and the dash line is its contour

- The figure shows that the improvement regions corresponding to the conventional decomposition approaches may be too large for some problems

Studies on Improved Decomposition Approaches

- Wang *et al.* [11] defined the ***improvement region*** of a current solution x_i for subproblem i as the region in the objective space, in which if a new solution y is produced then the current solution x_i can get replaced

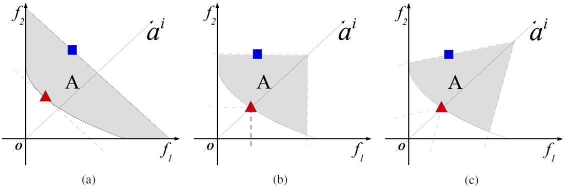


Figure: Illustrations of the improvement regions for : (a) WS, (b) TCH, and (c) PBI. In each sub-figure, region A is the improvement region. The square point is the current solution of subproblem i with the weight vector a^i, the triangle point is its optimal solution and the dash line is its contour

- The figure shows that the improvement regions corresponding to the conventional decomposition approaches may be too large for some problems
- **Limitation** - A single new good solution can lead to replacement of several old solutions, and result in deterioration of the population diversity

Studies on Improved Decomposition Approaches

- To overcome this limitation, Wang *et al.* [11] suggested to impose constraints on the subproblems to reduce the volumes of the improvement regions
- Constraint - The divergence of x_i to a_i should be less than $0.5 * \theta_i$
- θ^i is a control parameter for defining the improvement region corresponding to subproblem i

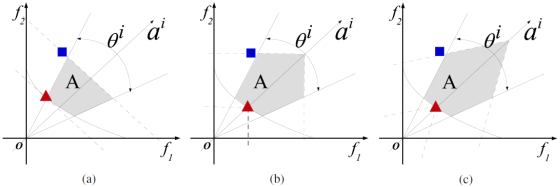


Figure: Illustrations of the improvement regions for constrained : (a) WS, (b) TCH, and (c) PBI. In each sub-figure, region A is the improvement region. The square point is the current solution of subproblem i with the weight vector a^i, the triangle point is its optimal solution and the dash line is its contour

L. Wang, Q. Zhang, A. Zhou, M. Gong, and L. Jiao, "Constrained subproblems in a decomposition-based multiobjective evolutionary algorithm," IEEE Transactions on Evolutionary Computation, 2016.

Studies on Improved Decomposition Approaches

- Wang *et al.* [11] included the constrained decomposition approach in MOEA/D and modified the replacement rules
- In particular, the replacement rules are similar to constrained binary tournament selection method
- The authors proposed two variants in the study
  - MOEA/D-CD - which requires appropriate tuning of the θ value
  - MOEA/D-ACD - in which the θ^i value is adaptively adjusted
- Both were tested on several MOPs and found to outperform MOEA/D

L. Wang, Q. Zhang, A. Zhou, M. Gong, and L. Jiao, ”Constrained subproblems in a decomposition-based multiobjective evolutionary algorithm,” IEEE Transactions on Evolutionary Computation, 2016.

Studies on Combination/Adaptation of Scalarizing functions

- Ishibuchi *et al.* [15] compared the performance of MOEA/D-WS and MOEA/D-TCH on MOKPs with 2-6 objectives (convex PF)
- MOEA/D-WS performs remarkably better than MOEA/D-TCH on 4 and 6 objective problems
- The authors compared the approaches with respect to the number of objectives

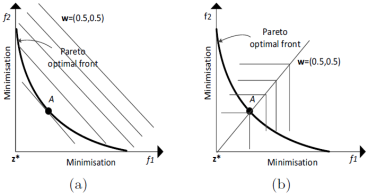


Figure: Contour lines of the WS (a) and weighted TCH (b) scalarising functions

H. Ishibuchi, Y. Sakane, N. Tsukamoto, and Y. Nojima, "Simultaneous use of different scalarizing functions in MOEA/D," GECCO 2010.

Studies on Combination/Adaptation of Scalarizing functions

- The objective space is divided into two sub-spaces by the contour line
- Solutions in one sub-space are better than solutions on the contour line while solutions in the other sub-space are worse
- The contour line of the WS approach is a line, and the contour line of the weighted TCH approach is a polygonal line (with vertical angle)
- For the WS approach, the size of a better region equals to half of the whole objective space regardless of the number of objectives

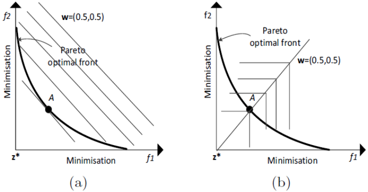


Figure: Contour lines of the WS (a) and weighted TCH (b) scalarising functions

Studies on Combination/Adaptation of Scalarizing functions

- The maximal probability of replacement of an existing solution by a newly generated solution i.e., 1/2, is not influenced by the number of objectives
- Thus, the search ability of the WS approach is not affected by an increase in the number of objectives
- For the TCH approach, the size of a better region equals to $(1/2)^m$ of m-dimensional objective space
- Thus, the maximal probability of replacement i.e., the search ability significantly decreases as the number of objective increases

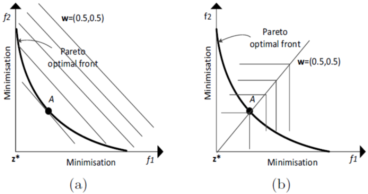


Figure: Contour lines of the WS (a) and weighted TCH (b) scalarising functions

Studies on Combination/Adaptation of Scalarizing functions

Adaptive scalarizing function strategy

- Further, Ishibuchi *et al*. [15] demonstrated that MOEA/D-TCH outperforms MOEA/D-WS on modified MOKP with non-convex PF
- Adaptive scalarizing function based approach for MOEA/D which automatically employs the WS and the TCH scalarizing function approach for subproblems along convex and non-convex regions of PF, respectively
- The experimental study on modified MOKPs with non-convex PFs demonstrated the effectiveness of the idea of adapting scalarizing functions in MOEA/D framework

H. Ishibuchi, Y. Sakane, N. Tsukamoto, and Y. Nojima, "Simultaneous use of different scalarizing functions in MOEA/D," GECCO 2010.

Studies on Combination/Adaptation of Scalarizing functions

Two phase strategy

- Jiang and Yang [16] highlighted that in convex PF problems with sharp peak and long tail, MOEA/D obtains dense solutions in the intermediate region of the PF
- And can hardly achieve well-distributed solutions in the extreme region of the PF
- Thus, the authors proposed a two-phase (TP) strategy for MOEA/D to tackle such problems
- In the first phase, M_r% of entire computing resources are dedicated with MOEA/D-TCH
- At the end of the first phase, the uniformity of solutions is evaluated using a crowding based method to determine if the problem is probably convex
- If the problem is convex, in the second phase, reverse TCH is used, in which solutions are evolved from the nadir point n by maximizing the scalarizing function value
- The experimental study on 2- and 3-objective problems with convex PFs having sharp peak and low tail confirmed the efficacy of the two-phase strategy

S. Jiang and S. Yang, An improved multiobjective optimization evolutionary algorithm based on decomposition for complex pareto fronts, IEEE Transactions on Cybernetics, 2016

Studies on Combination/Adaptation of Scalarizing functions

Pareto adaptive scalarizing functions (MOEA/D-par)

- Wang *et al.* investigated the property of the L_p weighted scalarizing functions
- It is noted that in the L_p weighted approaches, $p = 1$ and $\lim p \rightarrow \infty$ represent the WS and the TCH approach, respectively
- This study illustrated that there is a trade-off dependent on the p value between the search ability of the L_p weighted approach and its robustness on PF geometry
- Hence, the study recommended that an appropriate p value should be selected corresponding to each subproblem on the basis of PF geometry
- In MOEA/D-par, the p value is initialized as 1 for every subproblem at the beginning
- At every iteration, the PF geometry is estimated based on PF approximation using a family of reference curves [17] corresponding to every subproblem, and the p values associated with the subproblems are adapted accordingly

Studies on Combination/Adaptation of Scalarizing functions

Pareto adaptive scalarizing functions (MOEA/D-PaS)

- Wang *et al.* [18] extended their work on MOEA/D-par [19] and presented an enhanced online method, named Pareto-adaptive scalarizing approximation (*PaS*), to approximate the optimal p value in the L_p weighted approaches
- The advantage of *PaS* over the method presented earlier by the authors in [19] is that MOEA/D-PaS does not require PF estimation
- The PaS method in MOEA/D-PaS is simple and computationally efficient
- MOEA/D-PaS is found to be highly efficient on several difficult test problems with 2-, 4-, and 7-objectives, and different PF geometries

Summary

Summary

- In the L_p weighted approach, there is a trade-off dependent on the p value between the search ability of the approach and its robustness on PF geometry
- The Pareto-adaptive scalarizing approximation (PaS) has overcome the challenge of choosing an appropriate scalarizing function for a particular problem
- The MOEA/D variants based on constrained decomposition i.e., MOEA/D-CD and MOEA/D-ACD seem to be promising
- Alternate ways of decomposition (as in MOEA/D-M2M) provide new direction in which the objective space is partitioned into small subspaces using reference vectors, and good solutions are emphasized in each of the subspaces to maintain a balance between convergence and diversity

Outline

1 Introduction to Multi-objective Optimization
- Multi-objective Optimization
- Multi-objective Optimization Methods
- Goals in Multi-objective Optimization
- Evolutionary Multi-objective Optimization Frameworks

2 Multi-objective Evolutionary Algorithm based on Decomposition
- Introduction to MOEA/D
- Main Design Components of MOEA/D
- Studies on Weight Vector Generation Methods
- Studies on Decomposition Approaches
- Studies on Computational Resource Allocation
- Studies on Modifications in the Reproduction Operators
- Studies on Mating Selection and Replacement Mechanism
- Studies on Many-objective Optimization

3 Directions for Future Work

Studies on Computational Resource Allocation

Computational Resource Allocation in MOEA/D

In the original MOEA/D,

- A fixed set of weight vectors are utilized irrespective of the characteristics of the PF
- All the subproblems are allocated equal computational effort at each generation

Studies on Computational Resource Allocation

Computational Resource Allocation in MOEA/D

In the original MOEA/D,

- A fixed set of weight vectors are utilized irrespective of the characteristics of the PF
- All the subproblems are allocated equal computational effort at each generation

Limitations

- Some regions of the PF might be more difficult to approximate compared to others
- Fixed set of weight vectors may not hold good for all problems such as disconnected PFs
- Thus, uniform treatment of all the subproblems and fixed set of weight vectors may lead to wastage of computational resources

Studies on Computational Resource Allocation

The studies under this class can be divided into two categories:

- Computational resource allocation with **fixed weight vectors**
- Computational resource allocation with **weight vector adaptation**

Computational Resource Allocation with Fixed Weight Vectors

Dynamic resource allocation (MOEA/D-DRA)

- Zhang *et al.* [20] proposed **MOEA/D-DRA**, based on dynamic allocation of computational resources to different subproblems
- A utility function used to represent the relative improvement of the scalarizing function for each subproblem, and computed every 50 generations
- The subproblems for which the relative improvement in scaralizing function is higher, are assigned higher utility function value otherwise it is decreased
- Using a 10-tournament selection step, the subproblem with highest utility function value from 10 randomly selected subproblems enter the exploration phase
- Thus, the computational resources are dynamically allocated to those subproblems for which the utility function is higher
- MOEA/D-DRA is the winner of unconstrained MOEA competition of CEC 2009.

Q. Zhang, W. Liu, and H. Li, The performance of a new version of MOEA/D on CEC 2009 unconstrained mop test instances, CEC 2009

Computational Resource Allocation with Fixed Weight Vectors

Generalized Resource Allocation (MOEA/D-GRA)

- Zhou and Zhang [21] extended MOEA/D-DRA and presented **MOEA/D-GRA**
- Each subproblem is associated with a probability of improvement (PoI) vector
- At each generation, computational resources are assigned to some subproblems selected according to the PoI vector
- The authors introduced both offline (OFRA) and online (ONRA) resource allocation strategy in MOEA/D

A. Zhou and Q. Zhang, "Are all the subproblems equally important? Resource allocation in decomposition-based multiobjective evolutionary algorithms," IEEE Transactions on Evolutionary Computation, 2016

Computational Resource Allocation with Fixed Weight Vectors

Offline Resource Allocation Strategy

- To implement this strategy, a key task is to first define and measure the subproblem hardness
- The authors chose a problem $T1$ and decomposed it into 300 subproblems
- Executed **CoDE** [**?**] on each subproblem and recorded the average number of FEs for reducing the objective function value below a given error
- The average number of FEs is used to measure the hardness of each subproblem and define the PoI vector

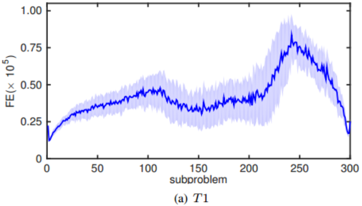


Figure: The FE values for different subproblems to achieve the goal that error $\leq 10^{-5}$ over 51 runs

Computational Resource Allocation with Fixed Weight Vectors

Online Resource Allocation Strategy

- To implement this strategy, a key issue is to measure subproblem hardness in an online manner
- As in MOEA/D-DRA [20], the authors defined utility function as the relative improvement in the last ΔT generations
- The PoI vector for subproblem i is defined as the ratio of its utility function to the maximum utility function across all the subproblems
- The experimental study demonstrated that ONRA strategy is superior to OFRA and NORA
- The experimental study on several MOPs demonstrated that MOEA/D-GRA significantly outperforms both MOEA/D-DE [22] and MOEA/D-DRA [20]

Computational Resource Allocation with Fixed Weight Vectors

External Archive Guided MOEA/D (MOEA/D-EAG)

- Cai *et al.* [23] proposed an external archive guided MOEA/D, termed EAG-MOEA/D
- EAG-MOEA/D evolves an internal (working) population using MOEA/D, and updates the external archive using non-dominated sorting and crowding distance principle of NSGA-II
- It records the number of successful solutions each subproblem contributes to the external archive over L previous generations
- The subproblems are probabilistically allocated computational resources depending upon their respective contribution to the external archive
- The external archive guides MOEA/D in allocating computation resources depending upon the historical convergence and diversity information
- EAG-MOEA/D significantly outperforms NSGA-II [1], MOEA/D [5], and MOEA/D-DRA [20] on multiobjective software next release problem (MNRP) and multiobjective traveling salesman problem (MTSP)

X. Cai, Y. Li, Z. Fan, and Q. Zhang, ”An external archive guided multiobjective evolutionary algorithm based on decomposition for combinatorial optimization,” IEEE Transactions on Evolutionary Computation, 2015

Computational Resource Allocation with Fixed Weight Vectors

External Archive Guided MOEA/D (MOEA/D-EAG)

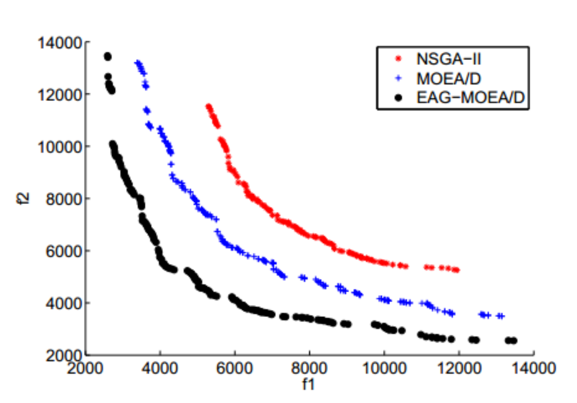


Figure: The final non-dominated solutions found by NSGA-II, MOEA/D and EAG-MOEA/D on an instance of 2-objective TSP

Computational Resource Allocation with Fixed Weight Vectors

External Archive Guided MOEA/D (MOEA/D-EAG)

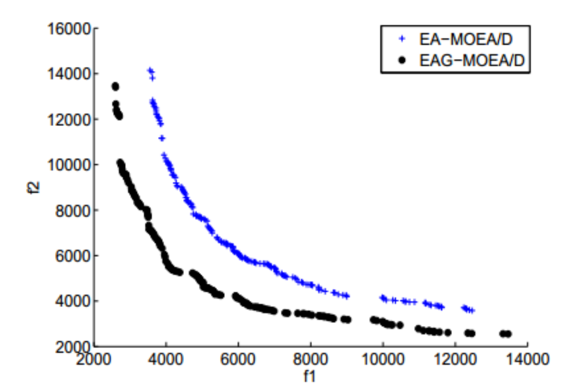


Figure: The final non-dominated solutions found by EA-MOEA/D and EAG-MOEA/D on an instance of 2-objective TSP

Computational Resource Allocation with Fixed Weight Vectors

MOEA/D with Support Vector Machine (MOEA/D-SVM)

- Lin *et al.* [24] presented a MOEA/D variant with classification based on SVM
- In MOEA/D-SVM, a classification model is built on the search space
- For the training set, the solutions in the current population are regarded as the promising solutions and the most recently discarded solutions for each subproblem as unpromising ones
- Thus, the model is designed to be accurate in the search area around the current population
- MOEA/D-SVM classifies all new solutions and performs function evaluation of all promising solutions
- Unpromising solutions are evaluated with small probability for exploration purpose
- The experimental study demonstrated that the classification approach can significantly improve the performance of MOEA/D

X. Lin, Q. Zhang, and S. Kwong, A decomposition based multiobjective evolutionary algorithm with classification, WCCI 2016

Computational Resource Allocation with Weight Vector Adaptation

Adaptive Weight Vector Adjustment (MOEA/D-AWA)

- Qi *et al.* [25] highlighted that in MOPs with discontinuous PF, several subproblems will have the same solution
- Thus, dealing with such problems simultaneously will lead to wastage of computational resources
- Further, for problems with sharp peak and low tail, MOEA/D cannot produce uniformly distributed solutions in extreme regions of PF
- To tackle such MOPs with complex PFs, the authors proposed an adaptive weight vector adjustment (AWA) strategy and integrated within MOEA/D-DRA
- It is based on a two-stage strategy where a set of predetermined weight vectors are employed until it converges to certain extent
- MOEA/D-AWA utilizes an external population to store the visited non-dominated solutions and to guide the algorithm in removal and addition of subproblems

Y. Qi et al., MOEA/D with adaptive weight adjustment, Evolutionary Computation, 2014

Computational Resource Allocation with Weight Vector Adaptation

Adaptive Weight Vector Adjustment (MOEA/D-AWA)

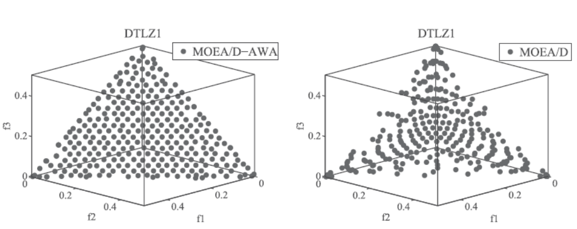


Figure: Performance comparison on DTLZ1

Computational Resource Allocation with Weight Vector Adaptation

Adaptive Weight Vector Adjustment (MOEA/D-AWA)

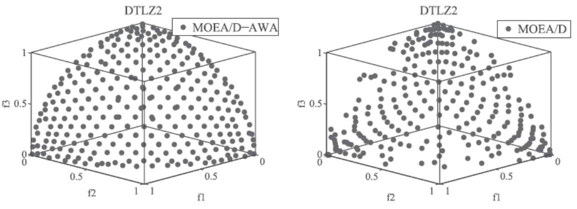


Figure: Performance comparison on DTLZ2

Computational Resource Allocation with Weight Vector Adaptation

Adaptive Weight Vector Adjustment (MOEA/D-AWA)

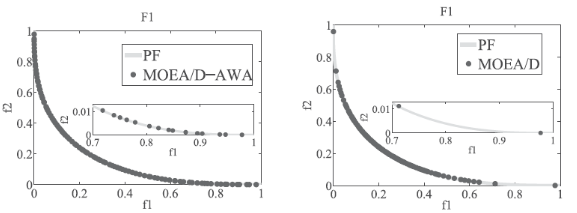


Figure: Performance comparison on F1

Computational Resource Allocation with Weight Vector Adaptation

Adaptive Weight Vector Adjustment (MOEA/D-AWA)

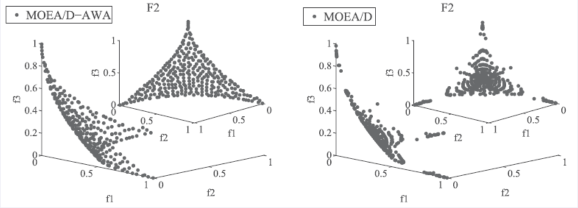


Figure: Performance comparison on F2

Studies on Computational Resource Allocation

Summary

- Depending upon the hardness of the problem, different subproblems may require different computational budget in order to be efficiently solved [21]
- Strategies based on dynamic computational resource allocation to different subproblems can improve the performance of MOEA/D [20], [21]
- Weight vector adaptation is essential for target MOP with complex or irregular PF (e.g. discontinuous PF, PF with sharp peak and low tail)
- Use of external archive to store non-dominated solutions and guide the internal working population of MOEA/D as in MOEA/D-AWA [25], EAG-MOEA/D [23] is highly promising

Outline

1. Introduction to Multi-objective Optimization
   - Multi-objective Optimization
   - Multi-objective Optimization Methods
   - Goals in Multi-objective Optimization
   - Evolutionary Multi-objective Optimization Frameworks
2. Multi-objective Evolutionary Algorithm based on Decomposition
   - Introduction to MOEA/D
   - Main Design Components of MOEA/D
   - Studies on Weight Vector Generation Methods
   - Studies on Decomposition Approaches
   - Studies on Computational Resource Allocation
   - Studies on Modifications in the Reproduction Operators
   - Studies on Mating Selection and Replacement Mechanism
   - Studies on Many-objective Optimization
3. Directions for Future Work

Studies on Modifications in the Reproduction Operators

Reproduction Operators in Original MOEA/D

- In original MOEA/D, the genetic operators - simulated binary crossover (SBX) and polynomial mutation operators have been used
- However, it is well known that there is no single EA which outperforms all other EAs across different problems
- Thus, several studies have aimed at modifying the reproduction operators in order to improve the performance of decomposition based MOEAs

Reproduction Operation based on DE

MOEA/D-DE

- Differential evolution (DE) is a very popular optimizer known for its simplicity and efficiency in solving real parameter optimization problems
- Li and Zhang [22] proposed an enhanced version of MOEA/D using differential evolution (DE) algorithm, termed **MOEA/D-DE**, to solve problems with complicated Pareto sets
- The study introduced nine test instances (F1-F9) with complicated PS shapes
- The experimental study demonstrated that MOEA/D-DE significantly outperforms NSGA-II-DE on all the test instances

H. Li and Q. Zhang, "Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II," IEEE Trans. Evol. Comput., 2009

Reproduction Operation based on DE

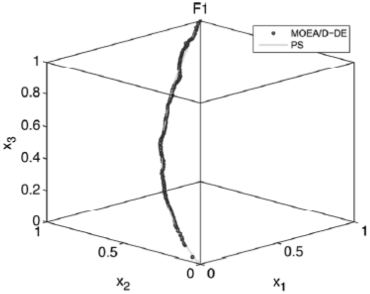


Figure: Performance of MOEA/D-DE.

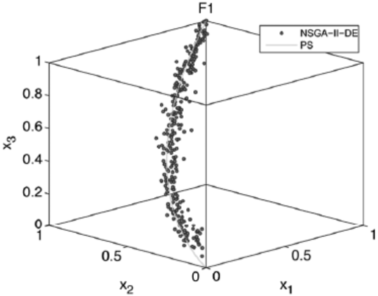


Figure: Performance of NSGA-II.

Reproduction Operation based on DE

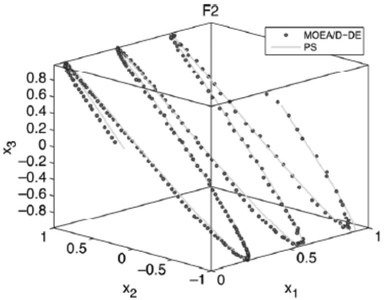


Figure: Performance of MOEA/D-DE.

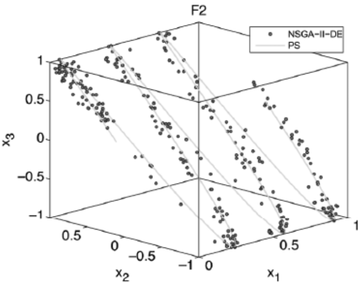


Figure: Performance of NSGA-II.

Reproduction Operation based on DE

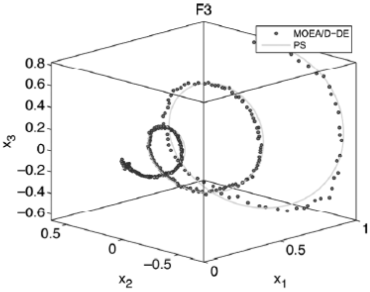


Figure: Performance of MOEA/D-DE.

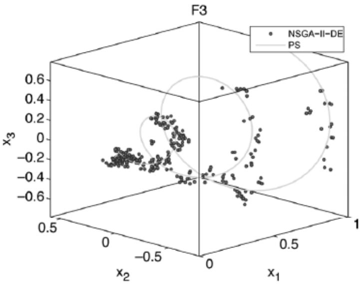


Figure: Performance of NSGA-II.

Reproduction Operation based on ACO

MOEA/D-ACO

- Ant colony optimization (ACO) is a very popular algorithm for solving combinatorial optimization problems
- Ke *et al.* [26] proposed a combination of ACO and MOEA/D, termed **MOEA/D-ACO**
- The experimental study comprehensively investigated the efficiency of MOEA/D-ACO on MOKP and MTSP

L. Ke, Q. Zhang, and R. Battiti, "MOEA/D-ACO: A multiobjective evolutionary algorithm using decomposition and antcolony," IEEE Trans. Cybern., 2013

Adaptive Operator Selection for Reproduction

Adaptive Operator Selection (AOS)

- A method that dynamically determines rate of application of different operators considering the performance history of operators
- AOS comprises of two tasks - **credit assignment (CA)** and **operator selection (OS)**
- The credit assignment task is rewarding an operator based on its recent performance
- The operator selection is choosing the operator to be applied next based on the reward information accumulated during the optimization process
- *Examples of CA:* average of the fitness improvements, rank based schemes like sum of ranks (SR), and fitness rate rank (FRR)
- *Examples of OS:* probabilistic methods such as probability matching (PM) and adaptive pursuit (AP), or multi-armed bandit (MAB) methods like Upper confidence bound (UCB)

Adaptive Operator Selection for Reproduction

MOEA/D with Fitness-rate-rank-based Multi-armed bandit (MOEA/D-FRRMAB)

- Li *et al.* [27] proposed **MOEA/D-FRRMAB** which utilizes FRR based credit assignment scheme and MAB based operator selection scheme
- The operator pool consists of four different DE variants namely *DE/rand/1*, *DE/rand/2*, *DE/current-to-rand/1* and *DE/current-to-rand/2*,
- The authors incorporated the proposed FRRMAB method within MOEA/D-DRA [20]
- MOEA/D-FRRMAB significantly outperforms MOEA/D-DE [22], MOEA/D-DRA [20] on CEC 2009 [28] MOPs

K. Li, A. Fialho, S. Kwong, and Q. Zhang, "Adaptive operator selection with bandits for a multiobjective evolutionary algorithm based on decomposition," IEEE Trans. Evol. Comput, 2014

Adaptive Operator Selection for Reproduction

MOEA/D with Fitness-rate-rank-based Multi-armed bandit (MOEA/D-FRRMAB)

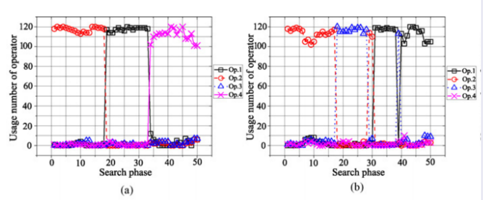


Figure: Operator adaptation trajectories of MOEA/D-FRRMAB on a) UF1, and b) UF2

- No single operator can dominate over the whole search process on different test instances
- FRRMAB can use different operators at different search stages and that it can efficiently switch from one operator to another
- For most of the instances, operators 2 and/or 3 (favor exploration) are preferred at early stages, while operators 1 and/or 4 (favor exploitation) are more frequently used at later stages

Studies on Modifications in the Reproduction Operators

Summary

- MOEA/D-DE is highly suitable for MOPs with complex PS shapes
- MOEA/D-ACO is a good algorithm for combinatorial MOPs
- Interesting work is being conducted on incorporating adaptive operator selection (AOS) in the MOEA/D framework

Outline

1 Introduction to Multi-objective Optimization
- Multi-objective Optimization
- Multi-objective Optimization Methods
- Goals in Multi-objective Optimization
- Evolutionary Multi-objective Optimization Frameworks

2 Multi-objective Evolutionary Algorithm based on Decomposition
- Introduction to MOEA/D
- Main Design Components of MOEA/D
- Studies on Weight Vector Generation Methods
- Studies on Decomposition Approaches
- Studies on Computational Resource Allocation
- Studies on Modifications in the Reproduction Operators
- Studies on Mating Selection and Replacement Mechanism
- Studies on Many-objective Optimization

3 Directions for Future Work

Studies on Mating Selection and Replacement Mechanism

Mating Selection and Replacement in Original MOEA/D

- The neighborhood structure as well as the neighborhood size (NS) play an important role in MOEA/D
- Because mating selection and update of neighboring solutions is dependent on neighborhood structure as well as NS
- In the original MOEA/D, the neighborhood relationship is defined in the weight vector space
- The neighborhood structure as well as the NS remain fixed throughout the evolutionary process
- A single good offspring solution can replace several inferior neighboring solutions and lead to deterioration of the population diversity

Studies on Mating Selection and Replacement Mechanism

Studies conducted on mating selection and replacement mechanism can be classified in three major categories:

- Studies on the mating selection
- Studies on the replacement mechanism
- Studies on both mating selection and replacement mechanism

Studies on Mating Selection and Replacement Mechanism

MOEA/D-DE

- Besides the introduction of DE operators in MOEA/D, Li and Zhang [22] refined the MOEA/D framework by introducing two extra measures
- The first measure allows parent solutions to be selected during reproduction with a low probability from the whole population (i.e., outside the neighborhood)
- The second measure puts an upper bound (n_r) on the maximal number of solutions that can be replaced by a child solution during the update of neighboring solutions
- The introduction of these extra measures help to maintain the population diversity

H. Li and Q. Zhang, "Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II," IEEE Trans. Evol. Comput., 2009

Studies on the Mating Selection

Niche-guided Mating Scheme

- Jiang and Yang [16] presented a niche-guided scheme for the setting of mating selection range
- In this scheme, each individual's niche count is computed over its T neighboring subproblems
- If the niche count of an individual is over a certain threshold, it means that the individual is similar to its T neighboring subproblems
- Thus the mating parents corresponding to the individual are selected from outside its neighborhood
- The strategy is particularly beneficial on 2-3 objective problems having disconnected PFs

S. Jiang and S. Yang, An improved multiobjective optimization evolutionary algorithm based on decomposition for complex Pareto fronts, IEEE Trans. Cybern., 2016

Studies on the Replacement Mechanism

Global Replacement Scheme based MOEA/D (MOEA/D-GR)

- Wang *et al.* [29] argue that the new solution x_i^{new} of subproblem i may not be the most suitable solution for its neighboring subproblems $B(i)$
- Thus, x_i^{new} will get discarded in the update stage unless a very big replacement neighborhood size is used

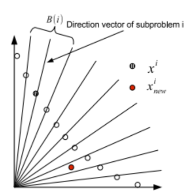


Figure: Example case

Studies on the Replacement Mechanism

Global Replacement Scheme based MOEA/D (MOEA/D-GR)

- Wang *et al.* [29] argue that the new solution x_i^{new} of subproblem i may not be the most suitable solution for its neighboring subproblems $B(i)$
- Thus, x_i^{new} will get discarded in the update stage unless a very big replacement neighborhood size is used

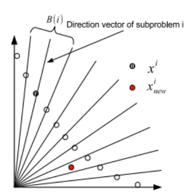


Figure: Example case

Z. Wang, Q. Zhang, M. Gong, and A. Zhou, "A replacement strategy for balancing convergence and diversity in MOEA/D," CEC 2014

Studies on the Replacement Mechanism

Global Replacement Scheme based MOEA/D (MOEA/D-GR)

- To overcome this shortcoming, the authors proposed a global replacement (GR) scheme for MOEA/D and named the resulting algorithm as **MOEA/D-GR**
- In this study, two different neighborhoods i.e., mating neighborhood (of size T_m) and replacement neighborhood (of size T_r), are considered for each subproblem i
- In the GR scheme, corresponding to a newly generated solution x_i^{new}, the most appropriate subproblem j is determined
- Thereafter, T_r closest subproblems to subproblem j are selected to form the replacement neighborhood i.e., $B_r(j)$
- Finally, the solutions of subproblems belonging to $B_r(j)$ are updated by the newly generated solution x_i^{new}

Z. Wang, Q. Zhang, M. Gong, and A. Zhou, "A replacement strategy for balancing convergence and diversity in MOEA/D," CEC 2014

Studies on the Replacement Mechanism

Adaptive Global Replacement Scheme based MOEA/D (MOEA/D-GR)

- Wang *et al.* [30] extended the GR scheme proposed in [29] and developed an adaptive GR scheme
- The authors argue that a small T_r is good for exploration at the beginning of the search process while a large T_r is good for exploitation towards the end of the search process
- The study investigated three different adaptive schemes for adjusting T_r, based on linear, exponential, and sigmoid functions
- Based on the adaptive replacement strategy, both a steady-state algorithm (named MOEA/D-AGR) and a generational algorithm (named gMOEA/D-AGR) are presented
- MOEA/D-AGR is found to be superior to several state-of-the-art MOEAs

Z. Wang, Q. Zhang, A. Zhou, M. Gong, and L. Jiao, Adaptive replacement strategies for MOEA/D, IEEE Trans. Cybern., 2016

Studies on Mating Selection and Replacement Mechanism

ENS-MOEA/D

- To overcome the problem of choosing a suitable NS for different problems, Zhao *et al.* [31] proposed an algorithm known as **ENS-MOEA/D**
- A pool of different NSs are used in the form of an ensemble
- The selection probabilities of NSs are dynamically adjusted based on their historical performances of generating promising solutions in certain fixed number of previous generations
- The experimental study demonstrated the superiority of ENS-MOEA/D against MOEA/D-DRA with fixed NSs on CEC 2009 [28] MOPs

S.-Z. Zhao, P. N. Suganthan, and Q. Zhang, ”Decomposition-based multiobjective evolutionary algorithm with an ensemble of neighborhood sizes,” IEEE Trans. Evol. Comput., 2012

Studies on Mating Selection and Replacement Mechanism

MACE-gD

- Giagkiozis *et al.* [8] presented an algorithm named **MACE-gD**
- In MACE-gD, the neighborhood structure of a subproblem is controlled by parameter ρ
- The parameter ρ indicates the percentage of top solutions in the current population with respect to a subproblem, which are used in building the probability model for CE method
- Thus, the neighborhood relationship is dynamically updated with respect to the objective space and is not static or defined with respect to weight vector space as in original MOEA/D
- In the replacement step, a new solution to subproblem *i* is only compared with the current solution to subproblem *i*
- Limitation - The study did not present an experiment to validate the performance of defining neighborhood relationship in objective space as compared to the weight vector space

I. Giagkiozis, R. C. Purshouse, and P. J. Fleming, Generalized decomposition and cross entropy methods for many-objective optimization, Inf. Sci., 2014

Studies on Mating Selection and Replacement Mechanism

Summary

- Extensive research has been conducted to improve the mating selection and the replacement mechanism in the MOEA/D framework
- The neighborhood relationship in the weight vector space, as defined in the original MOEA/D framework, can be deceptive to the algorithm [8]
- The neighborhood relationship should be rather defined in the objective space and should be adaptive such that solutions which participate in mating procedure are close in the objective space [8]
- MOEA/D based on adaptive global replacement scheme (MOEA/D-AGR) [30] seems to be highly promising

Outline

1. Introduction to Multi-objective Optimization
   - Multi-objective Optimization
   - Multi-objective Optimization Methods
   - Goals in Multi-objective Optimization
   - Evolutionary Multi-objective Optimization Frameworks
2. Multi-objective Evolutionary Algorithm based on Decomposition
   - Introduction to MOEA/D
   - Main Design Components of MOEA/D
   - Studies on Weight Vector Generation Methods
   - Studies on Decomposition Approaches
   - Studies on Computational Resource Allocation
   - Studies on Modifications in the Reproduction Operators
   - Studies on Mating Selection and Replacement Mechanism
   - Studies on Many-objective Optimization
3. Directions for Future Work

Studies on Many-objective Optimization

Many-objective Optimization

- Many-objective optimization problems (MaOPs) refer to the class of MOPs with four or more number of objectives
- MaOPs present several challenges to MOEAs such as
- With the increase in size of the objective space, balancing convergence and diversity becomes much more difficult
- Due to the computational efficiency consideration, the population size cannot be arbitrarily large. Thus, PF in high-dimensional objective space has to be approximated with limited number of solutions
- In Pareto-dominance based MOEAs, the selection pressure severely deteriorates
- In HV indicator-based MOEAs, the computational complexity to evaluate the HV indicator grows exponentially with the increasing number of objectives

Studies on Many-objective Optimization

NSGA-III

- Deb and Jain presented a reference-point-based many-objective evolutionary algorithm, termed **NSGA-III** [32]

Studies on Many-objective Optimization

NSGA-III

- Deb and Jain presented a reference-point-based many-objective evolutionary algorithm, termed **NSGA-III** [32]
- The basic framework of NSGA-III is similar to NSGA-II [1] with significant modifications in the replacement step

Studies on Many-objective Optimization

NSGA-III

- Deb and Jain presented a reference-point-based many-objective evolutionary algorithm, termed **NSGA-III** [32]
- The basic framework of NSGA-III is similar to NSGA-II [1] with significant modifications in the replacement step
- Further, unlike NSGA-II, the maintenance of diversity among population members in NSGA-III is aided by supplying well spread reference points

Studies on Many-objective Optimization

NSGA-III

- Deb and Jain presented a reference-point-based many-objective evolutionary algorithm, termed **NSGA-III** [32]
- The basic framework of NSGA-III is similar to NSGA-II [1] with significant modifications in the replacement step
- Further, unlike NSGA-II, the maintenance of diversity among population members in NSGA-III is aided by supplying well spread reference points
- Like weight vector generation in MOEA/D, NSGA-III also utilizes a set of reference points that spread over the objective space

Studies on Many-objective Optimization

NSGA-III

- Deb and Jain presented a reference-point-based many-objective evolutionary algorithm, termed **NSGA-III** [32]
- The basic framework of NSGA-III is similar to NSGA-II [1] with significant modifications in the replacement step
- Further, unlike NSGA-II, the maintenance of diversity among population members in NSGA-III is aided by supplying well spread reference points
- Like weight vector generation in MOEA/D, NSGA-III also utilizes a set of reference points that spread over the objective space
- However, NSGA-III does not decompose the MOP explicitly into single-objective subproblems like MOEA/D

Studies on Many-objective Optimization

NSGA-III

- Deb and Jain presented a reference-point-based many-objective evolutionary algorithm, termed **NSGA-III** [32]
- The basic framework of NSGA-III is similar to NSGA-II [1] with significant modifications in the replacement step
- Further, unlike NSGA-II, the maintenance of diversity among population members in NSGA-III is aided by supplying well spread reference points
- Like weight vector generation in MOEA/D, NSGA-III also utilizes a set of reference points that spread over the objective space
- However, NSGA-III does not decompose the MOP explicitly into single-objective subproblems like MOEA/D
- NSGA-III uses a generational replacement scheme like NSGA-II

Studies on Many-objective Optimization

NSGA-III

- Deb and Jain presented a reference-point-based many-objective evolutionary algorithm, termed **NSGA-III** [32]
- The basic framework of NSGA-III is similar to NSGA-II [1] with significant modifications in the replacement step
- Further, unlike NSGA-II, the maintenance of diversity among population members in NSGA-III is aided by supplying well spread reference points
- Like weight vector generation in MOEA/D, NSGA-III also utilizes a set of reference points that spread over the objective space
- However, NSGA-III does not decompose the MOP explicitly into single-objective subproblems like MOEA/D
- NSGA-III uses a generational replacement scheme like NSGA-II
- Further, at each generation, NSGA-III performs normalization of population members in the objective space, and associate each population member with a reference point

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., 2014

Studies on Many-objective Optimization

NSGA-III

- To select solutions from the last acceptable front, a niche-preservation operator is used, in which solutions associated with less crowded reference point have a higher chance to be selected

Studies on Many-objective Optimization

NSGA-III

- To select solutions from the last acceptable front, a niche-preservation operator is used, in which solutions associated with less crowded reference point have a higher chance to be selected
- The experimental study compared NSGA-III with MOEA/D-TCH, MOEA/D-PBI, and MOEA/D-DE [22] on several MaOPs ranging from 3-15 objectives

Studies on Many-objective Optimization

NSGA-III

- To select solutions from the last acceptable front, a niche-preservation operator is used, in which solutions associated with less crowded reference point have a higher chance to be selected
- The experimental study compared NSGA-III with MOEA/D-TCH, MOEA/D-PBI, and MOEA/D-DE [22] on several MaOPs ranging from 3-15 objectives
- The experimental results demonstrated that MOEA/D-TCH and MOEA/D-DE perform quite poorly on most of the problems

Studies on Many-objective Optimization

NSGA-III

- To select solutions from the last acceptable front, a niche-preservation operator is used, in which solutions associated with less crowded reference point have a higher chance to be selected
- The experimental study compared NSGA-III with MOEA/D-TCH, MOEA/D-PBI, and MOEA/D-DE [22] on several MaOPs ranging from 3-15 objectives
- The experimental results demonstrated that MOEA/D-TCH and MOEA/D-DE perform quite poorly on most of the problems
- NSGA-III performs better on some test problems while MOEA/D-PBI performs better on some other problems

Studies on Many-objective Optimization

NSGA-III

- To select solutions from the last acceptable front, a niche-preservation operator is used, in which solutions associated with less crowded reference point have a higher chance to be selected
- The experimental study compared NSGA-III with MOEA/D-TCH, MOEA/D-PBI, and MOEA/D-DE [22] on several MaOPs ranging from 3-15 objectives
- The experimental results demonstrated that MOEA/D-TCH and MOEA/D-DE perform quite poorly on most of the problems
- NSGA-III performs better on some test problems while MOEA/D-PBI performs better on some other problems
- The efficacy of NSGA-III is also demonstrated on two real-world problems - 3-objective crash worthiness vehicle design and 9-objective car design problem

Studies on Many-objective Optimization

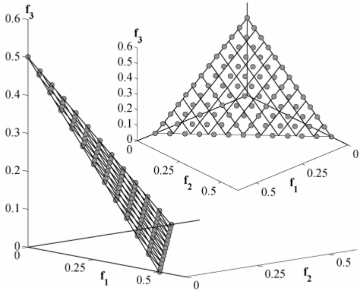


Figure: Performance of NSGA-III on DTLZ1.

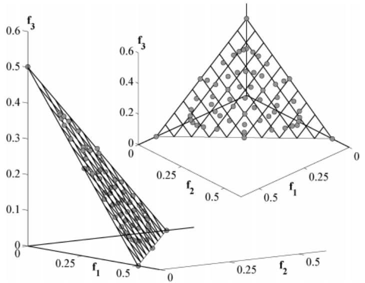


Figure: Performance of MOEA/D-TCH on DTLZ1.

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., 2014

Studies on Many-objective Optimization

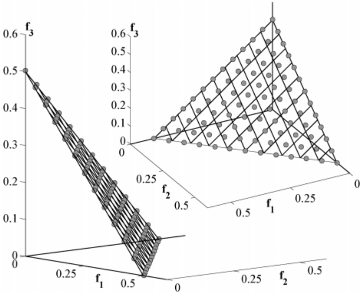


Figure: Performance of MOEA/D-PBI on DTLZ1.

Studies on Many-objective Optimization

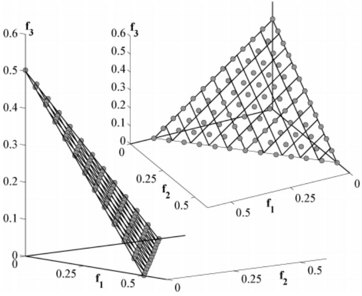


Figure: Performance of MOEA/D-PBI on DTLZ1.

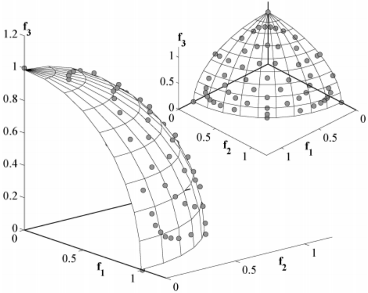


Figure: Performance of MOEA/D-TCH on DTLZ2.

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., 2014

Studies on Many-objective Optimization

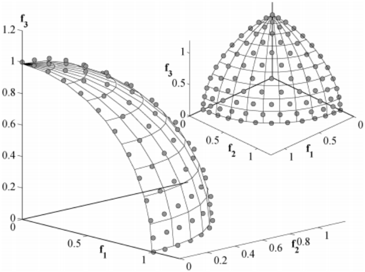


Figure: Performance of NSGA-III on DTLZ2.

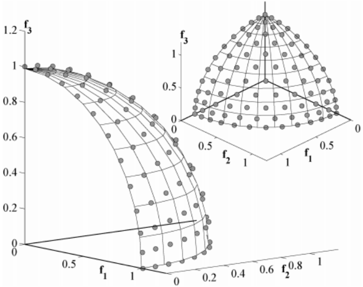


Figure: Performance of MOEA/D-PBI on DTLZ2.

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., 2014

Studies on Many-objective Optimization

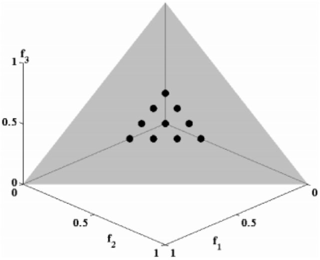


Figure: Reference points on normalized hyperplane.

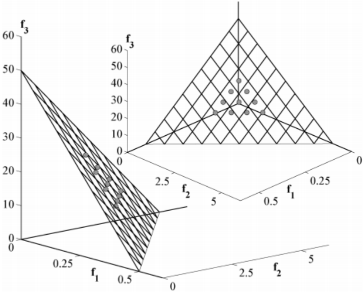


Figure: Preferred solutions obtained by NSGA-III DTLZ1.

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., 2014

Studies on Many-objective Optimization

NSGA-III

- Jain and Deb [33] extended NSGA-III [32] to tackle constrained optimization problems
- In **constrained NSGA-III**, the constraint binary tournament selection operator of NSGA-II is used in selecting parents for mating
- Further, in the elitist selection operator, the constraint-domination principle of NSGA-II is adopted to classify the combined parent offspring population into nondomination levels
- In the same study [33], Jain and Deb also proposed a constrained MOEA/D
- The constraint handling mechanism in **C-MOEA/D** is based on the modification of the replacement step
- When comparing offspring solution with a neighboring solution, rules similar to constrained binary tournament selection are applied

H. Jain and K. Deb, An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part II: Handling constraints and extending to an adaptive approach, IEEE Trans. Evol. Comput, 2014

Studies on Many-objective Optimization

- In [33], Jain and Deb introduced three different type of constrained optimization problems which are scalable with respect to objectives as well as number of decision variables
- These problems were designed to introduce different types of difficulties
- In Type-1 constrained problems, the original PF is still optimal, but there is an infeasible barrier in approaching the PF

Studies on Many-objective Optimization

- In [33], Jain and Deb introduced three different type of constrained optimization problems which are scalable with respect to objectives as well as number of decision variables
- These problems were designed to introduce different types of difficulties
- In Type-1 constrained problems, the original PF is still optimal, but there is an infeasible barrier in approaching the PF

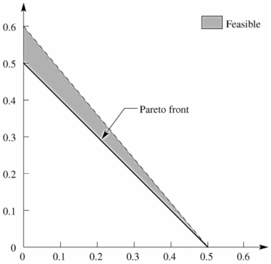


Figure: Two-objective version of the C1-DTLZ1 problem.

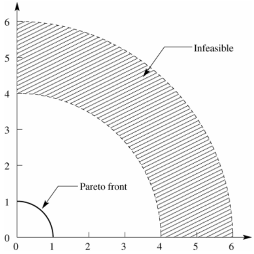


Figure: Two-objective version of the C1-DTLZ2 problem.

Studies on Many-objective Optimization

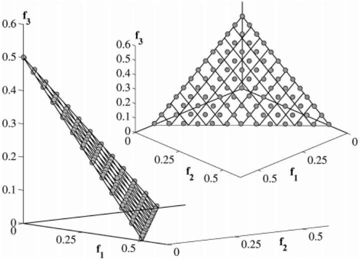


Figure: Performance of NSGA-III on C1-DTLZ1 problem.

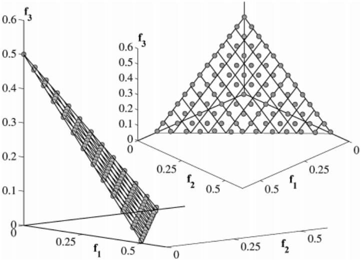


Figure: Performance of C-MOEAD on C1-DTLZ1 problem.

Studies on Many-objective Optimization

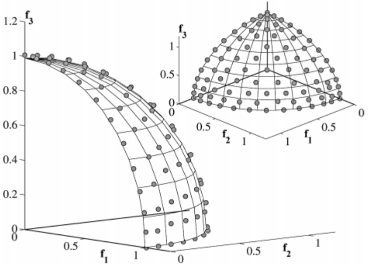


Figure: Performance of NSGA-III on C1-DTLZ3 problem.

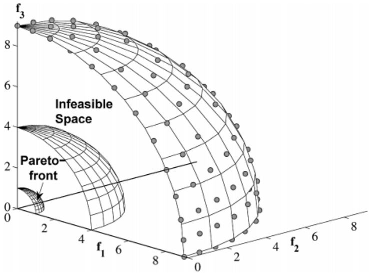


Figure: Performance of C-MOEAD on C1-DTLZ3 problem.

Studies on Many-objective Optimization

- In [33], Jain and Deb argued that in many constrained or even unconstrained problems, there will be some reference points with no PO solution associated with them
- On the other hand, there will be some reference points which have more than one PO solution associated with them
- Hence, NSGA-III may not be able to generate uniformly distributed PO solutions over then entire PF for such problems

Studies on Many-objective Optimization

- In [33], Jain and Deb argued that in many constrained or even unconstrained problems, there will be some reference points with no PO solution associated with them
- On the other hand, there will be some reference points which have more than one PO solution associated with them
- Hence, NSGA-III may not be able to generate uniformly distributed PO solutions over then entire PF for such problems

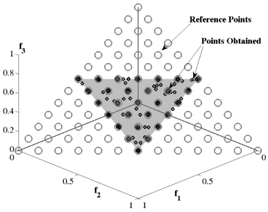


Figure: Example case for Inverted DTLZ1 problem

Studies on Many-objective Optimization

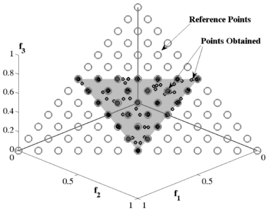


Figure: Example case for Inverted DTLZ1 problem

Studies on Many-objective Optimization

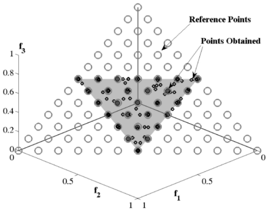


Figure: Example case for Inverted DTLZ1 problem

- NSGA-III with 91 reference points was run on the problem
- PO solutions (big solid circles) were found from 28 different useful reference points, but the remaining 63 reference points could not associate a PO solution
- The solutions marked in small open circles are additional solutions to the 28 useful reference points
- Since locations of these additional solutions are not used in any careful manner in the algorithm, their distribution is somewhat random
- Not only is the final population non-uniform, there is a wastage of computational efforts in processing these solutions

Studies on Many-objective Optimization

Adaptive NSGA-III

- To overcome this difficulty, the authors presented an adaptive NSGA-III, named **A-NSGA-III**
- A-NSGA-III always preserves the original reference points
- Adaptively adds new reference points around a crowded reference point which has more than one population member associated with it
- Adaptively deletes reference points which have no population member associated with them

H. Jain and K. Deb, An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part II: Handling constraints and extending to an adaptive approach, IEEE Trans. Evol. Comput, 2014

Studies on Many-objective Optimization

NSGA-III Performance

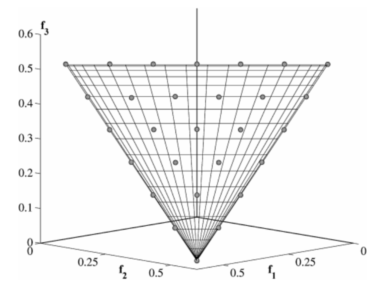


Figure: Obtained solutions using NSGA-III on the inverted DTLZ1 problem (only the closest solution for every useful reference point is shown).

Studies on Many-objective Optimization

NSGA-III Performance

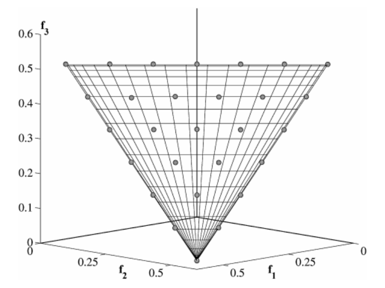


Figure: Obtained solutions using NSGA-III on the inverted DTLZ1 problem (only the closest solution for every useful reference point is shown).

Adaptive NSGA-III Performance

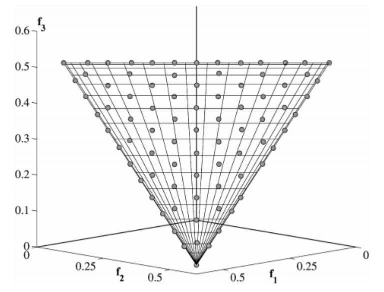


Figure: Obtained solutions using adaptive NSGA-III on the inverted DTLZ1 problem (only the closest solution for every useful reference point is shown).

Studies on Many-objective Optimization

- Li *et al.* [9] proposed a unified paradigm, termed **MOEA/DD**, based on combination of dominance- and decomposition-based approaches

Studies on Many-objective Optimization

- Li *et al.* [9] proposed a unified paradigm, termed **MOEA/DD**, based on combination of dominance- and decomposition-based approaches
- Each weight vector in MOEA/DD along with defining a subproblem, also specifies a unique subregion in the objective space

Studies on Many-objective Optimization

- Li *et al.* [9] proposed a unified paradigm, termed **MOEA/DD**, based on combination of dominance- and decomposition-based approaches
- Each weight vector in MOEA/DD along with defining a subproblem, also specifies a unique subregion in the objective space
- Thus, each solution is always associated with a unique subregion and the problem of diversity management is addressed by local density estimation of the subregions

Studies on Many-objective Optimization

- Li *et al.* [9] proposed a unified paradigm, termed **MOEA/DD**, based on combination of dominance- and decomposition-based approaches
- Each weight vector in MOEA/DD along with defining a subproblem, also specifies a unique subregion in the objective space
- Thus, each solution is always associated with a unique subregion and the problem of diversity management is addressed by local density estimation of the subregions
- The characteristic feature of MOEA/DD is its update procedure which is carried out in a steady-state hierarchical manner, depending on Pareto dominance, local density estimation, and scalarizing functions, sequentially

Studies on Many-objective Optimization

- Li *et al.* [9] proposed a unified paradigm, termed **MOEA/DD**, based on combination of dominance- and decomposition-based approaches
- Each weight vector in MOEA/DD along with defining a subproblem, also specifies a unique subregion in the objective space
- Thus, each solution is always associated with a unique subregion and the problem of diversity management is addressed by local density estimation of the subregions
- The characteristic feature of MOEA/DD is its update procedure which is carried out in a steady-state hierarchical manner, depending on Pareto dominance, local density estimation, and scalarizing functions, sequentially
- For example, when an offspring solution is added to the parent population, and the non-domination level is 1

Studies on Many-objective Optimization

- Li *et al.* [9] proposed a unified paradigm, termed **MOEA/DD**, based on combination of dominance- and decomposition-based approaches
- Each weight vector in MOEA/DD along with defining a subproblem, also specifies a unique subregion in the objective space
- Thus, each solution is always associated with a unique subregion and the problem of diversity management is addressed by local density estimation of the subregions
- The characteristic feature of MOEA/DD is its update procedure which is carried out in a steady-state hierarchical manner, depending on Pareto dominance, local density estimation, and scalarizing functions, sequentially
- For example, when an offspring solution is added to the parent population, and the non-domination level is 1
- The density (or niche count) of each subregion is estimated by counting the number of solutions associated with it

Studies on Many-objective Optimization

- Li *et al.* [9] proposed a unified paradigm, termed **MOEA/DD**, based on combination of dominance- and decomposition-based approaches
- Each weight vector in MOEA/DD along with defining a subproblem, also specifies a unique subregion in the objective space
- Thus, each solution is always associated with a unique subregion and the problem of diversity management is addressed by local density estimation of the subregions
- The characteristic feature of MOEA/DD is its update procedure which is carried out in a steady-state hierarchical manner, depending on Pareto dominance, local density estimation, and scalarizing functions, sequentially
- For example, when an offspring solution is added to the parent population, and the non-domination level is 1
- The density (or niche count) of each subregion is estimated by counting the number of solutions associated with it
- Thereafter, from the most crowded subregion, the solution will the largest PBI metric is eliminated

Studies on Many-objective Optimization

- Li *et al.* [9] proposed a unified paradigm, termed **MOEA/DD**, based on combination of dominance- and decomposition-based approaches
- Each weight vector in MOEA/DD along with defining a subproblem, also specifies a unique subregion in the objective space
- Thus, each solution is always associated with a unique subregion and the problem of diversity management is addressed by local density estimation of the subregions
- The characteristic feature of MOEA/DD is its update procedure which is carried out in a steady-state hierarchical manner, depending on Pareto dominance, local density estimation, and scalarizing functions, sequentially
- For example, when an offspring solution is added to the parent population, and the non-domination level is 1
- The density (or niche count) of each subregion is estimated by counting the number of solutions associated with it
- Thereafter, from the most crowded subregion, the solution will the largest PBI metric is eliminated
- MOEA/DD was found to outperform NSGA-III, MOEA/D, on several MOPs with 3-15 objectives

K. Li, K. Deb, Q. Zhang, and S. Kwong, An evolutionary manyobjective optimization algorithm based on dominance and decomposition,IEEE Trans. Evol. Comput, 2015

Studies on Many-objective Optimization

Sensitivity to PF Shapes

- Ishibuchi *et al.* [34] conducted an exhaustive performance analysis of decomposition-based many-objective algorithms to demonstrate their sensitivity to PF shapes

Studies on Many-objective Optimization

Sensitivity to PF Shapes

- Ishibuchi *et al.* [34] conducted an exhaustive performance analysis of decomposition-based many-objective algorithms to demonstrate their sensitivity to PF shapes
- The authors slightly changed the DTLZ and WFG problem formulations by multiplying (-1) to each objective of DTLZ and WFG problems

Studies on Many-objective Optimization

Sensitivity to PF Shapes

- Ishibuchi *et al.* [34] conducted an exhaustive performance analysis of decomposition-based many-objective algorithms to demonstrate their sensitivity to PF shapes
- The authors slightly changed the DTLZ and WFG problem formulations by multiplying (-1) to each objective of DTLZ and WFG problems
- In comparison to the original test problems, the performance of MOEA/DD, NSGA-III, etc. significantly deteriorated

Studies on Many-objective Optimization

Sensitivity to PF Shapes

- Ishibuchi *et al.* [34] conducted an exhaustive performance analysis of decomposition-based many-objective algorithms to demonstrate their sensitivity to PF shapes
- The authors slightly changed the DTLZ and WFG problem formulations by multiplying (-1) to each objective of DTLZ and WFG problems
- In comparison to the original test problems, the performance of MOEA/DD, NSGA-III, etc. significantly deteriorated
- The authors highlighted this to the reason that DTLZ1-4 and WFG4-9 have triangular shape PFs

Studies on Many-objective Optimization

Sensitivity to PF Shapes

- Ishibuchi *et al*. [34] conducted an exhaustive performance analysis of decomposition-based many-objective algorithms to demonstrate their sensitivity to PF shapes
- The authors slightly changed the DTLZ and WFG problem formulations by multiplying (-1) to each objective of DTLZ and WFG problems
- In comparison to the original test problems, the performance of MOEA/DD, NSGA-III, etc. significantly deteriorated
- The authors highlighted this to the reason that DTLZ1-4 and WFG4-9 have triangular shape PFs
- Thus their is a similarity between the shape of distribution of the weight vectors and the shape of the PFs of these test problems

Studies on Many-objective Optimization

Sensitivity to PF Shapes

- Ishibuchi *et al.* [34] conducted an exhaustive performance analysis of decomposition-based many-objective algorithms to demonstrate their sensitivity to PF shapes
- The authors slightly changed the DTLZ and WFG problem formulations by multiplying (-1) to each objective of DTLZ and WFG problems
- In comparison to the original test problems, the performance of MOEA/DD, NSGA-III, etc. significantly deteriorated
- The authors highlighted this to the reason that DTLZ1-4 and WFG4-9 have triangular shape PFs
- Thus their is a similarity between the shape of distribution of the weight vectors and the shape of the PFs of these test problems
- DTLZ1-4^{-1} and WFG4-9^{-1} have rotated triangular shape PFs

Studies on Many-objective Optimization

Sensitivity to PF Shapes

- Ishibuchi *et al*. [34] conducted an exhaustive performance analysis of decomposition-based many-objective algorithms to demonstrate their sensitivity to PF shapes
- The authors slightly changed the DTLZ and WFG problem formulations by multiplying (-1) to each objective of DTLZ and WFG problems
- In comparison to the original test problems, the performance of MOEA/DD, NSGA-III, etc. significantly deteriorated
- The authors highlighted this to the reason that DTLZ1-4 and WFG4-9 have triangular shape PFs
- Thus their is a similarity between the shape of distribution of the weight vectors and the shape of the PFs of these test problems
- DTLZ1-4^{-1} and WFG4-9^{-1} have rotated triangular shape PFs
- The study recommended - use of a wide variety of test problems with various PF shapes, adaptive mechanisms for weight vectors and scalarizing functions

H. Ishibuchi, Y. Setoguchi, H. Masuda, and Y. Nojima, "Performance of decomposition-based many-objective algorithms strongly depends on Pareto front shapes," IEEE Trans. Evol. Comput., to be published

Studies on Many-objective Optimization

Summary

- On MaOPs, MOEA/D-TCH and MOEA/D-DE do not perform well while the performance of MOEA/D-PBI is quite well on a range of test problems [32]
- The studies - NSGA-III [32], MOEA/DD [9] provide new direction in which the high-dimensional objective space in MaOPs can be partitioned into small subspaces using reference vectors, and sophisticated update procedures can be employed to preserve diversity in all subspaces
- Efficiently combining dominance- and decomposition-based approaches can result in high performance many-objective optimizers (e.g., NSGA-III [32], MOEA/DD [9], etc.)

Future Work

Decomposition Approaches

- Currently, only one study [14] has investigated into methods for adaptive tuning of the parameter θ in the PBI approach but that too has been tested only on few MOPs
- Thus, introduction of new methods to adaptively control parameter θ in PBI approach can be interesting future direction
- Combination/adaptation of scalarizing functions should be further designed so as to avoid the efforts of choosing a particular scalarizing function

Computational Resource Allocation

- Only a few decomposition-based MOEAs such as MOEA/D-AWA [25], adaptive NSGA-III [33] have been proposed with weight vector adaptation
- Thus, more studies should be undertaken to develop weight vector adaptation strategies for decomposition-based framework

Future Work

Many-objective Optimization

- Genetic operators have been used in most of the successful many-objective optimizers (e.g. NSGA-III [32], MOEA/DD [9], etc
- However, on MOPs, AOS has been found to be really beneficial [27], [35]. Thus, investigating AOS on MaOPs is an interesting future direction
- The role of neighborhood structure for mating and replacement in the decomposition-based MOEAs can be further studied for MaOPs. This is because I-DBEA [36] has been found to be considerably successful on MaOPs without involving the concept of neighborhood
- New decomposition-based MOEAs can be developed to efficiently tackle MaOPs
- To develop such many-objective optimizers, use of reference vectors to decompose the objective space into small multiple subspaces [32], [9], [37] and combining decomposition-based approach with dominance- or indicator-based approach can be a highly important component

References I

- K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *Evolutionary Computation, IEEE Transactions on*, vol. 6, no. 2, pp. 182–197, Apr 2002.
- E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization," in *Evolutionary Methods for Design, Optimisation, and Control*. CIMNE, Barcelona, Spain, 2002, pp. 95–100.
- E. Zitzler and S. Künzli, *Indicator-Based Selection in Multiobjective Search*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 832–842.
- D. Brockhoff and E. Zitzler, "Improving hypervolume-based multiobjective evolutionary algorithms by using objective reduction methods," Singapore, 2007, pp. 2086 – 2093.
- Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *Evolutionary Computation, IEEE Transactions on*, vol. 11, no. 6, pp. 712–731, Dec 2007.
- T. Murata and M. Gen, "Cellular genetic algorithm for multi-objective optimization," in *Proc. of the Fourth Asian Fuzzy System Symposium*, 2000, pp. 538 – 542.

References II

- Y. yan Tan, Y. chang Jiao, H. Li, and X. kuan Wang, "MOEA/D + uniform design: A new version of MOEA/D for optimization problems with many objectives," *Computers & Operations Research*, vol. 40, no. 6, pp. 1648 – 1660, 2013.
- I. Giagkiozis, R. Purshouse, and P. Fleming, "Generalized decomposition and cross entropy methods for many-objective optimization," *Information Sciences*, vol. 282, pp. 363 – 87, 2014.
- K. Li, K. Deb, Q. Zhang, and S. Kwong, "An evolutionary many-objective optimization algorithm based on dominance and decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 5, pp. 694–716, Oct 2015.
- H. Sato, "Inverted PBI in MOEA/D and its impact on the search performance on multi and many-objective optimization," in *Proceedings of the 2014 Conference on Genetic and Evolutionary Computation*, ser. GECCO '14. New York, NY, USA: ACM, 2014, pp. 645–652.
- L. Wang, Q. Zhang, A. Zhou, M. Gong, and L. Jiao, "Constrained subproblems in a decomposition-based multiobjective evolutionary algorithm," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 3, pp. 475–480, June 2016.

References III

- H. Sato, “Analysis of inverted PBI and comparison with other scalarizing functions in decomposition based MOEAs,” *Journal of Heuristics*, vol. 21, no. 6, pp. 819 – 849, 2015.
- S. Huband, P. Hingston, L. Barone, and L. While, “A review of multiobjective test problems and a scalable test problem toolkit,” *Evolutionary Computation, IEEE Transactions on*, vol. 10, no. 5, pp. 477–506, Oct 2006.
- S. Yang, S. Jiang, and Y. Jiang, “Improving the multiobjective evolutionary algorithm based on decomposition with new penalty schemes,” *Soft Computing*, pp. 1–15, 2016.
- H. Ishibuchi, Y. Sakane, N. Tsukamoto, and Y. Nojima, “Adaptation of scalarizing functions in MOEA/D: an adaptive scalarizing function-based multiobjective evolutionary algorithm,” Berlin, Germany, 2009, pp. 438 – 52.
- S. Jiang and S. Yang, “An improved multiobjective optimization evolutionary algorithm based on decomposition for complex pareto fronts,” *IEEE Transactions on Cybernetics*, vol. 46, no. 2, pp. 421–437, Feb 2016.

References IV

- S. Jiang, Z. Cai, J. Zhang, and Y.-S. Ong, "Multiobjective optimization by decomposition with pareto-adaptive weight vectors," in *Natural Computation (ICNC), 2011 Seventh International Conference on*, vol. 3, July 2011, pp. 1260–1264.
- R. Wang, Q. Zhang, and T. Zhang, "Decomposition based algorithms using pareto adaptive scalarizing methods," *Evolutionary Computation, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2016.
- ——, "Pareto adaptive scalarising functions for decomposition based algorithms," vol. Part I, Cham, Switzerland, 2015, pp. 248 – 62.
- Q. Zhang, W. Liu, and H. Li, "The performance of a new version of MOEA/D on cec09 unconstrained mop test instances," in *Evolutionary Computation, 2009. CEC '09. IEEE Congress on*, May 2009, pp. 203–208.
- A. Zhou and Q. Zhang, "Are all the subproblems equally important? Resource allocation in decomposition-based multiobjective evolutionary algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 1, pp. 52–64, Feb 2016.
- H. Li and Q. Zhang, "Multiobjective optimization problems with complicated pareto sets, MOEA/D and NSGA-II," *Evolutionary Computation, IEEE Transactions on*, vol. 13, no. 2, pp. 284–302, April 2009.

References V

- X. Cai, Y. Li, Z. Fan, and Q. Zhang, "An external archive guided multiobjective evolutionary algorithm based on decomposition for combinatorial optimization," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 4, pp. 508–523, Aug 2015.
- X. Lin, Q. Zhang, and S. Kwong, "A decomposition based multiobjective evolutionary algorithm with classification," in *World Congress on Computational Intelligence (WCCI) 2016*, July 2016.
- Y. Qi, X. Ma, F. Liu, L. Jiao, J. Sun, and J. Wu, "MOEA/D with adaptive weight adjustment," *Evol. Comput.*, vol. 22, no. 2, pp. 231–264, Jun. 2014.
- L. Ke, Q. Zhang, and R. Battiti, "MOEA/D-ACO: A multiobjective evolutionary algorithm using decomposition and antcolony," *Cybernetics, IEEE Transactions on*, vol. 43, no. 6, pp. 1845–1859, Dec 2013.
- K. Li, A. Fialho, S. Kwong, and Q. Zhang, "Adaptive operator selection with bandits for a multiobjective evolutionary algorithm based on decomposition," *Evolutionary Computation, IEEE Transactions on*, vol. 18, no. 1, pp. 114–130, Feb 2014.

References VI

- Q. Zhang, A. Zhou, S. Z. Zhao, P. N. Suganthan, W. Liu, and S. Tiwari, "Multiobjective optimization test instances for the CEC 2009 special session and competition," University of Essex and Nanyang Technological University, Essex, U.K./Singapore, Tech. Rep. CES-887, 2008.
- Z. Wang, Q. Zhang, M. Gong, and A. Zhou, "A replacement strategy for balancing convergence and diversity in MOEA/D," in *Evolutionary Computation (CEC), 2014 IEEE Congress on*, July 2014, pp. 2132–2139.
- Z. Wang, Q. Zhang, A. Zhou, M. Gong, and L. Jiao, "Adaptive replacement strategies for MOEA/D," *IEEE Transactions on Cybernetics*, vol. 46, no. 2, pp. 474–486, Feb 2016.
- S.-Z. Zhao, P. Suganthan, and Q. Zhang, "Decomposition-based multiobjective evolutionary algorithm with an ensemble of neighborhood sizes," *Evolutionary Computation, IEEE Transactions on*, vol. 16, no. 3, pp. 442–446, June 2012.
- 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 Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577 – 601, 2014.

References VII

- H. Jain and K. Deb, "An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, Part II: Handling constraints and extending to an adaptive approach," *Evolutionary Computation, IEEE Transactions on*, vol. 18, no. 4, pp. 602–622, Aug 2014.
- H. Ishibuchi, Y. Setoguchi, H. Masuda, and Y. Nojima, "Performance of decomposition-based many-objective algorithms strongly depends on pareto front shapes," *IEEE Transactions on Evolutionary Computation*, vol. PP, no. 99, pp. 1–1, 2016.
- R. Gonalves, C. Almeida, and A. Pozo, "Upper confidence bound (UCB) algorithms for adaptive operator selection in moea/d," in *Evolutionary Multi-Criterion Optimization*, ser. Lecture Notes in Computer Science, A. Gaspar-Cunha, C. Henggeler Antunes, and C. C. Coello, Eds. Springer International Publishing, 2015, vol. 9018, pp. 411–425.
- M. Asafuddoula, T. Ray, and R. Sarker, "A decomposition-based evolutionary algorithm for many objective optimization," *Evolutionary Computation, IEEE Transactions on*, vol. 19, no. 3, pp. 445–460, June 2015.
- R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, "A reference vector guided evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. PP, no. 99, pp. 1–1, 2016.