Solving MOOP: Non-Pareto MOEA approaches

Debasis Samanta

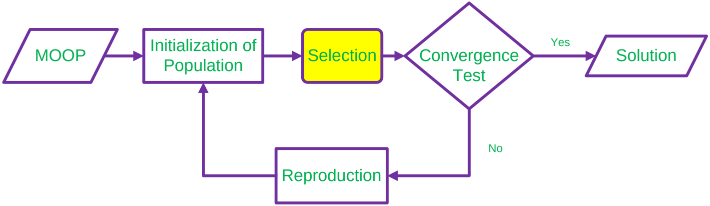
Indian Institute of Technology Kharagpur

dsamanta@iitkgp.ac.in

08.04.2023

Multi-objective evolutionary algorithm

- To distinguish the GA to solve single objective optimization problems to that of MOOPs, a new terminology called Evolutionary Algorithm (EA) has been coined.
- In many research articles, it is popularly abbreviated as MOEA, the short form of **M**ulti-**O**bjective **E**volutionary **A**lgorithm.
- The following is the MOEA framework, where *Reproduction* is same as in GA but different strategies are followed in *Selection*.



Difference between GA and MOEA

1. Difference between GA and MOEA are lying in input (single objective vs. multiple objectives) and output (single solution vs. trade-off solutions, also called Pareto-optimal solutions).

2. Two major problems are handled in MOEA
   - How to accomplish fitness assignment (evaluation) and selection thereafter in order to guide the search toward the Pareto optimal set.
   - How to maintain a diverse population in order to prevent premature convergence and achieve a well distributed Pareto-optimal front.

Classification of MOEA techniques

- MOEA Techniques
  - A priori approach
    - Aggregation (Ordering)
      - Lexicographic ordering
    - Aggregation (Scalarization)
      - Linear fitness evaluation (SOEA)
      - Non-linear fitness evaluation (SOEA)
      - Goal attainment
      - Weighted Min-max method
      - Game theory
  - A posteriori approach
    - Independent sampling
    - Hybrid Selection
    - Criterion selection (VEGA)
    - Pareto Selection
      - Ranking
      - Ranking and Niching
      - Demes
      - Elitist

Classification of MOEA techniques

Note :

- A priory technique requires a knowledge to define the relative importances of objectives prior to search
- A posteriori technique searches for Pareto-optimal solutions from a set of feasible solutions

MOEA technoiques to be discussed

1. A priori approches
   - Lexicographic ordering
   - Simple weighted approach (SOEA)
2. A posteriori approaches
   - Criterion selection (VEGA)
   - Pareto-based approaches
     - Rank-based approach (MOGA)
     - Rank + Niche based approach (NPGA)
     - Non-dominated sorting based approach (NSGA)
     - Elitist non-dominated sorting based approach (NSGA-II)

MOEA technoiques to be discussed

1. **Non-Pareto based approches**
    - Lexicographic ordering
    - Simple weighted approach (SOEA)
    - Criterion selection (VEGA)
2. **Pareto-based approaches**
    - Rank-based approach (MOGA)
    - Rank + Niche based approach (NPGA)
    - Non-dominated sorting based approach (NSGA)
    - Elitist non-dominated sorting based approach (NSGA-II)

Lexicographic Ordering

Lexicographic ordering method

Reference :

"Compaction of Symbolic Layout using Genetic Algorithms" by M.P Fourman in Proceedings of 1st International Conference on Genetic Algorithms, Pages 141-153, 1985.

- It is an a priori technique based on the principle of "aggregation with ordering".

Lexicographic ordering method

Suppose, a MOOP with k objectives and n constraints over a decision space x and is denoted as.

Minimize

$$f = [f_1, f_2, \cdots, f_k]$$

Subject to

$$g_j(x) \leq c_j, \text{ where } j = 1, 2, \cdots, n$$

1. Objectives are ranked in the order of their importance (done by the programmer). Suppose, the objectives are arranged in the following order.

$$f = [f_1 < f_2 < f_3 < \cdots < f_k]$$

Here, $f_i < f_j$ implies f_i is of higher importance than f_j

Lexicographic ordering method

2. The optimum solution $\bar{x}^*$ is then obtained by minimizing each objective function at a time, which is as follows.

   (a) **Minimize** $f_1(x)$
   Subject to $g_j(x) \leq c_j,\ j = 1, 2, \cdots, n$
   Let its solution be $\bar{x}_1^*$, that is $f_1^* = f_1(\bar{x}_1^*)$

   (b) **Minimize** $f_2(x)$
   Subject to $g_j(x) \leq c_j,\ j = 1, 2, \cdots, n$
   $f_1(x) = f_1^*$
   Let its solution be $\bar{x}_2^*$, that is $f_2^* = f_2(\bar{x}_2^*)$

   ....

   ....

   (c) At the i-th step, we have

   Minimize $f_i(x)$
   Subject to $g_j(x) \leq c_j,\ j = 1, 2, \cdots, n$
   $f_l(x) = f_l^*,\ l = 1, 2, \cdots, i-1$

Lexicographic ordering method

This procedure is repeated until all k objectives have been considered in the order of their importances.

The solution obtained at the end is $\bar{x}_k^*$, that is, $f_k^* = f_k(\bar{x}_k^*)$.
This is taken as the desired solution $\bar{x}^*$ of the given multiobjective optimization problem

Remarks on Lexicographic ordering method

Remarks :

- Deciding priorities (i.e. ranks) of objective functions is an issue. Solution may vary if a different ordering is taken.
- Different strategies can be followed to address the above issues.
    1. Random selection of an objective function at each run
    2. Naive approach to try with $k!$ number of orderings of k objective functions and then selecting the best observed result.

Note :
It produces a single solution rather than a set of Pareto-optimal solutions.

Single Objective Evolutionary Agorithm

SOEA: Single-Objective Evolutionary Algorithm

- This is an a priori technique based on the principle of ”linear aggregation of functions”.
- It is also alternatively termed as (SOEA) ”Single Objective Evoluationary Algorithm”.
- In many literature, this is also termed as Weighted sum approach.
- In fact, it is a naive approach to solve a MOOP.

SOEA approach to solve MOOPs

- This method consists of adding all the objective functions together using different weighting coefficients for each objective.
- This means that our multi-objective optimization problem is transformed into a scalar optimization problem.

  In other words, in order to optimize say n objective functions $f_1, f_2, \cdots, f_n$. It compute fitness using

$$\text{fitness} = \sum_{i=1}^{n} w_i \times f_i(x)$$

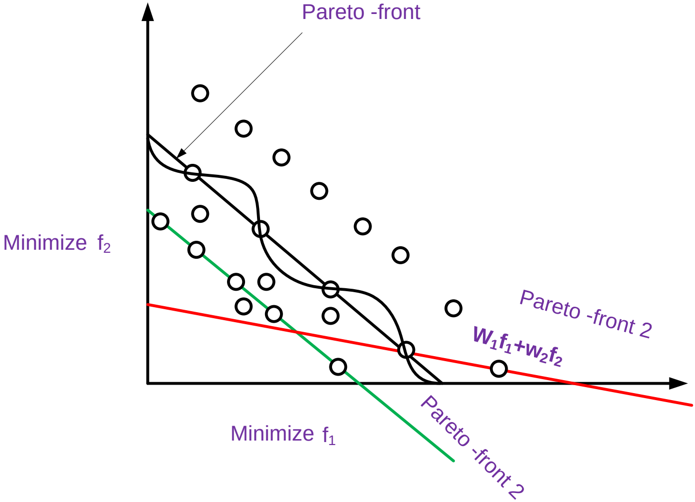
where $w_i \geq 0$ for each $i = 1, 2, ...n$ are the weighting coefficients representing the relative importance of the objectives. It is usually assume that

$$\sum_{i=1}^{n} w_i = 1$$

Comments on SOEA

1. This is the simplest approach and works in the same framework of Simple GA.
2. The results of solving an optimization problem can vary significantly as the weighting coefficient changes.
3. In other words,it produces different solutions with different values of w_i's.
4. Since very little is usually known about how to choose these coefficients, it may result into a local optima.

Local optimum solution in SOEA



Comments on SOEA

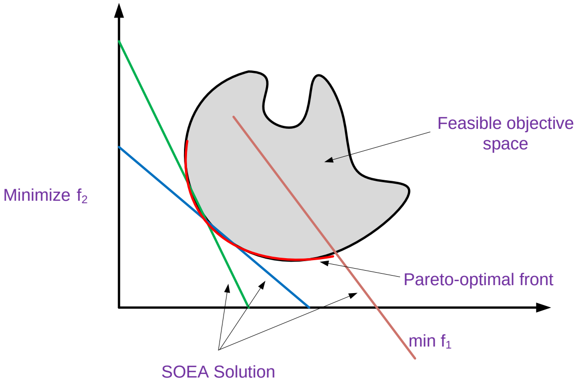
3. As a way out of this, it is necessary to solve the same problem for many different values of w_i's.
4. The wighting coefficients do not proportionally reflects the relative importance of the objectives, but are only factors, which, when varied, locate points in the Pareto set.
5. This method depends on not only w_i's values but also on the units in which functions are expressed.
6. In that case, we have to scale the objective values. that is

$$fitness = \sum_{i=1}^{n} w_i \times f_i(x) \times c_i$$

where c_i's are constant multipliers that scales the objectives properly.

Naive Approach : Weighted sum approach

7. The technique can not be used to find Pareto-optimal solutions which lie on the non-convex portion of the Pareto optimal front. In that case, it gives only one solution, which might be on the Pareto front.



Vector Evaluated Genetic Agorithm

Vector Evaluated Genetic Algorithm (VEGA)

- Proposed by David Schaffer (1985) in

  ”Multiple objective optimization with vector evaluated genetic algorithm - Genetic algorithm and their application”: Proceeding of the first international conference on Genetic algorithm, 93-100, 1985.

- It is normally considered as the first implementation of a MOEA
- VEGA is an a posteriori MOEA technique based on the principle of Criterion selection strategy.

Vector Evaluated Genetic Algorithm (VEGA)

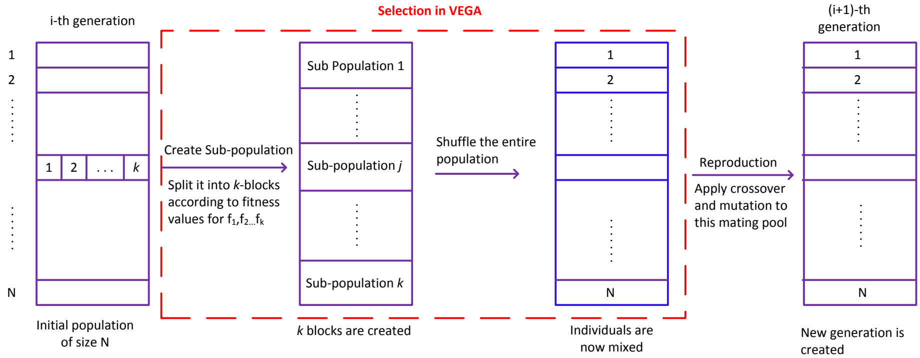
About VEGA :

- It is an extension of Simple Genetic Algorithm (SGA).
- It is an example of a criterion (or objective) selection technique where a fraction of each succeeding population is selected based on separate objective performance. The specific objective for each fraction are randomly selected at each generation.
- VEGA differs SGA in the way in which the selection operation is performed.

Basic steps in VEGA

1. Suppose, given a MOOP is to optimize k objective functions $f_1, f_2, \cdots, f_k$
2. A number of sub-population is selected according to each objective function in turn.
3. Thus, k-subpopulations each of size $\frac{M}{k}$ are selected, where M is the size of the mating pool ($M \leq N$), and N is the size of the input population.
4. These sub-population are shuffled together to obtain a new ordering of individuals.
5. Apply standard GA operations related to reproduction.
6. This produced next generation and Steps 2-5 continue until the termination condition is reached.

Overview of the VEGA



VEGA selection strategy

VEGA consists of the following three major steps:

1. Creating k sub-populations each of size $\frac{M}{k}$
2. Shuffle the sub-populations
3. Reproduction of offspring for next generation (same as in SGA)

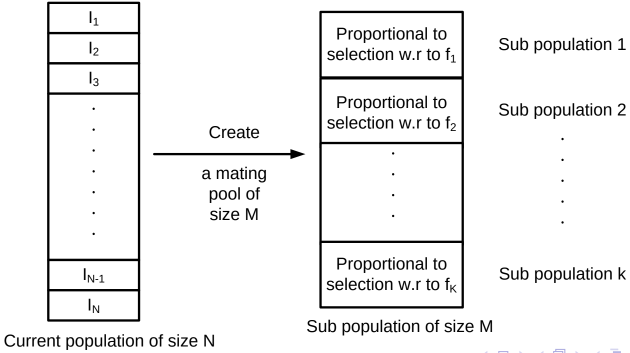
We explain the above steps with the following consideration:

- Suppose, given a MOOP, where we are to optimize k number of objective functions $f = f_1, f_2, \cdots, f_k$.
- Given the population size as N with individual $I_1, I_2, \cdots, I_N$
- We are to create a mating pool of size M, where $(M \leq N)$.

VEGA: Creation of sub-populations

1. **Create a mating pool of size M ($M \leq N$)**

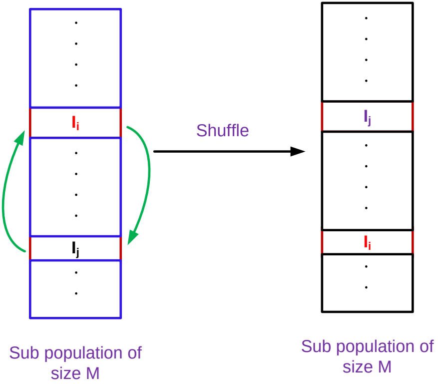
   Generate i-th subpopulation of size $\frac{M}{k}$ where $i = 1, 2, \cdots, k$.
   To do this follow the proportional selection strategy (such as Roulette-wheel selection) according to the i-th objective function only at a time.



VEGA: Shuffle the sub-populations

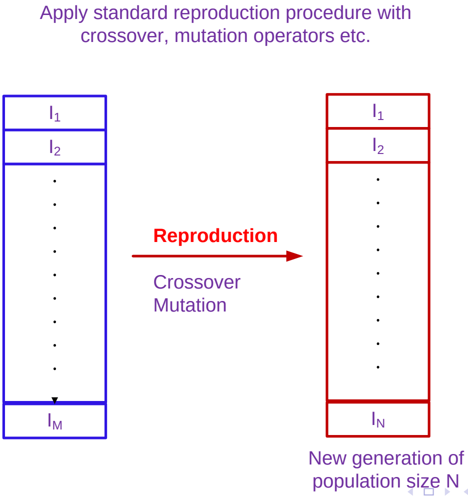
2. **Shuffle the sub-populations**

   Using some shuffling operation (e.g. generate two random numbers i and j between 1 and M both inclusive and then swap I_i and I_j which are in the i and j sub-populations.



VEGA: Reproduction

3. **Reproduction:** Perform reproduction to produce new generation of population size N.



Comments on VEGA

Advantages:

1. VEGA can be implemented in the same framework as SGA (only with a modification of selection operation).
2. VEGA can be viewed as optimizing $f_1, f_2, \cdots, f_k$ simultaneously. That is, $f(x) = \hat{e}_1 f_1(x) + \hat{e}_2 f_2(x) + \cdots + \hat{e}_k f_k(x)$, where e_i is the i-th vector.

   Thus, VEGA is a generalization from scalar genetic algorithm to vector evaluated genetic algorithm (and hence its name!).
3. VEGA leads to a solution close to local optima with regard to each individual objective.

Coments on VEGA

Disadvantages:

1. The solutions generated by VEGA are locally non-dominated but not necessarily globally dominated. This is because their non-dominance are limited to the current population only.
2. ”Speciation” problem in VEGA : It involves the evolution of ”Species” within the population (which excel on different objectives).
3. This is so because VEGA selects individuals who excel in one objective, without looking at the others.
4. This leads to ”middling” performance (i.e. an individual with acceptable performance, perhaps above average, but not outstanding for any of the objective function.

An Questions?