Opposition based Solution Generation in Multiobjective Evolutionary Algorithms

August 5, 2015 Author1[1]
Author2[2]

1 Introduction

Talk about some introductory stuffs and do some literature reviews to set the scenario, also discuss basic issues like "what is MOP" etc.

2 An Alternative Interpretation of Opposition

As we have seen in the previous section, mostly the idea of *opposition* is employed as the incorporation of new solutions with a certain kind of *opposite traits* into the existing population. Such *traits* could be interpreted according to different perspectives, an *opposite* solution could be – i) the one with an opposite representation (w.r.t the current best solution), ii) the solutions with the opposite values from the other end of the variable bounds (i.e. real valued optimization). However, such re-route from the continuing search trajectory could be a misleading one – in a sense that the opposite solutions could only be useful if the search space follows a desired pattern. For example, if the problem is to solve the N-queens problem, then the opposite representation of the current best solution could result into another valid solution. One can easily verify this fact by computing the reverse assignment of queens from a existing optimal solution and eventually find another global optima. This fact also indicates another interesting property of the search space – that it also needs to be multi-modal. As the reverse representation of the current best solution needs to be the optimum for another peak. Therefore, most of the standard opposition based algorithms inject the opposite solutions during the optimization start-up; or maintain a constant (generally low) ratio of opposite solutions into the current population. Therefore, the standard opposite solution injection scheme could only be effective given the aforementioned assumptions on the underlying search space. For this reason, such injection scheme is quite unwieldy to incorporate into a large scale numerical optimization problems, like in mutli-objective optimization problems (MOPs).

Moreover, we have also found some examples of opposition based algorithm for Q/TD-learning like scenario. Similar argument can be made, as the reinforcement-learning like search algorithm is inherently greedy (as it follows the Bellman's optimality principle), and the opposite actions during the learning phase introduces a noise; so that the algorithm can branch out to alternative choices. In that sense, we can say that the injection of opposite solutions can be considered as a different form of variation operation in population based stochastic search.

However, for the case of MOPs, a generic opposition scheme may not be effective enough just because of the sheer complexity of the search space of interest, moreover, we have almost no room to make any assumption about the multi-modality and/or biasness in the solutions (i.e. as it could be done in N-queen Problem). Therefore, in this paper, we have revised the notion of opposition in terms of the search algorithm's behaviour. For example, most Evolutionary Multiobjective Optimization (EMO) algorithms aim to maximize two principal aspects – i) the convergence and ii) the diversity, as the quality of a MOP solution depends on this two factors. This change now

[1]Responsible for the GP module
[2]Responsible for the Simulator module

let us re-consider the opposite point generation/injection in a different perspective –

- Convergence: A solution *far* from the true Pareto-front is *opposite* of any solution that is *closer* to the true Pareto-front.
- Diversity: An *isolated* solution on the true Pareto-front is *opposite* of a *crowded* solution.

By taking the above two notion into account, we will deterministically generate opposition solutions during the search. Obviously, the deterministic point generation scheme will only consider the *opposite trait* that is *good*. In the next section we will see, how the existing EMO algorithm has limitations maintaining this two *opposite traits* during the variation (i.e. solution generation) operations.

3 Limitations with Canonical MOP Algorithms: NSGA-II

Show that standard algorithms like NSGA-II converges as a Pareto-line/surface for many problems

4 The Basic Algorithm: Opposite Point Generation Scheme

Describe the point generation scheme.

5 Finding the Extreme Points

Describe how the extreme points are important and how to find them efficiently.

6 Experiments with the Multi-objective Problem Sets

Intro to this section.

6.1 ZDT Problem Set

Discuss ZDT problem set results.

6.2 DTLZ Problem Set

Discuss DTLZ problem set results.

6.3 Constrained Problems

Discuss constrained problem (OSY) results.

6.4 Rotated Problems

Discuss rotated problem (NSGA-II) results.

7 Comparative Analysis

Discuss the comparison.

8 NSGA-II Equipped with Extreme Points

Discuss if NSGA-II is given two extreme points, how it behaves (less robust)

9 NSGA-II Compensated for the Extra Function Evaluations

Discuss what if we run the NSGA-II with the compensated function evaluations.

10 Algorithm without the Deterministic Opposite Point Generation

Now the reader may ponder that the what if we just mutate solutions near the extreme points? Show that such scheme actually does not work.

References