2011

Reference Set Metrics for Multi-objective Algorithms

Chilukuri K. Mohan
Syracuse University, ckmohan@syr.edu

Kishan Mehrotra
Syracuse University, mehrotra@syr.edu

Recommended Citation
Mohan C, Mehrotra K. Reference Set Metrics for Multi-objective Algorithms. In Swarm, Evolutionary, and Memetic Computing: Panigrahi B, Suganthan P, Das S, Satapathy S, editors.: Springer Berlin / Heidelberg; 2011. p. 723-30.

Reference Set Metrics for Multi-objective Algorithms

Chilukuri K. Mohan and Kishan G. Mehrotra
Department of Electrical Engineering & Computer Science

Syracuse University, Syracuse, NY, USA
ckmohan/mehrotra@syr.edu

Abstract. Several metrics and indicators have been suggested in the past to evaluate multi-objective evolutionary and non-evolutionary algorithms. However, these metrics are known to have many problems that make their application sometimes unsound, and sometimes infeasible. This paper proposes a new approach, in which metrics are parameterized with respect to a reference set, on which depend the properties of any metric.

Keywords: Multi-objective Algorithms, Reference Set, Metrics

1 Introduction

Evaluating and comparing single-objective evolutionary algorithms is a relatively straightforward task: *evaluate whether or not a particular solution quality was achieved, how often (over various trials) such quality was achieved, and how much computational effort was required.* By contrast, the evaluation of multi-objective evolutionary and non-evolutionary algorithms, collectively abbreviated as MOAs in this paper, is rendered difficult by the lack of simple and satisfactory performance metrics[1]. The main reason is that the output of an MOA run is a collection of vectors forming a non-dominated set.

Comparative results are generally shown in graphical form indicating which algorithm performs better [7, 8]. Some oft-used metrics are discussed in [6, 10]; a comprehensive list of metrics is available in Table 1 on page 41 of a recent survey article by Zhou et. al [11]. For example, several of the classical metrics are defined with respect to the Pareto set consisting of all non-dominated solutions to the problem. Unfortunately, the Pareto set is unknown or impossible to enumerate in many practical problems. Other metrics rely on computing the hyper-volume dominated by solutions produced by an algorithm; an algorithm that dominates greater volume is considered to be better than another that dominates less volume.

[1] The word "indicator" better describes these, but MOA literature appears to prefer the slightly inaccurate usage –"metric"; used in this paper as well

The unsoundness or unsatisfactoriness of several oft-used metrics has been discussed by Knowles and Corne [6]. Some difficulties are: potential non-transitivity[2], consequences of Arrows theorem [1], and Condorcets voting paradox [12], resolutions to which have been discussed by researchers in Economics and Game Theory.

In this paper we propose that all comparative evaluations of MOAs should be with respect to a Reference Set (RefSet). Inferences and conclusions based on such evaluations cease to be valid when the context of the RefSet is removed, yet in some instances, this is the only possible approach. Various metrics can be proposed, similar to those existing in the literature, but parameterized by a specific RefSet. The properties satisfied by a metric would then depend on the choice of the RefSet. Valid statistical arguments can be constructed to evaluate algorithms in case one algorithm is shown to be better than other with many unbiased choices of RefSets. Depending on the choice of RefSets, it may be possible to combine the results with appropriate weights to obtain an *overall* performance measure.

Section 2 presents the main idea of this paper, discussing why and how RefSets may be constructed to facilitate algorithm comparison. Sections 3-5 focus on RefSet-based metrics related to solution set cardinality, domination area, and solution set diversity. Predictably, the last section presents concluding remarks.

2 Reference Sets

A Reference Set (RefSet) is a collection of candidate solutions with respect to which we can compare two algorithms. We may distinguish between RefSets that:

- Focus exclusively on non-domination, abbreviated NRefSets;
- Focus exclusively on diversity, abbreviated DRefSets; and
- Consider both non-domination and diversity, abbreviated DNRefSets.

For example, a possible NRefSet that can be used by the metric is the Pareto set consisting of all non-dominated solutions to the problem. Unfortunately, as mentioned earlier, the Pareto set is unknown or impossible to enumerate in many practical problems. Such metrics can instead be replaced by others that are parameterized with an appropriately chosen RefSet. The following are examples of RefSets that can be used to compare algorithms with respect to a specific problem instance:

- U: the union of all solutions to the problem instance obtained by all means known to humanity for some benchmark problems, the Pareto set may be available for use as U.
- NU: the subset of U consisting only of mutually non-dominating solutions, i.e., obtained by deleting all elements of U that are dominated by other elements in U.

[2] It is possible that algorithm A is considered better than algorithm B, and B better than C, as well as C better than A are fundamental to any 3-party election

- U_D : a subset of U obtained by deleting elements of U that are near others according to a minimal distance threshold condition D specified to hold between any two elements in the set. The threshold conditions may be
  - In data space, or
  - In objective function space.

  The latter is desirable from the perspective of sampling different parts of the Pareto surface, whereas the former is of interest in applications such as routing where robustness is to be achieved by finding multiple substantially distinct solutions. The distance threshold condition D may be parameterized based on the threshold value used to evaluate proximity. Also, different choices of U_D can be obtained for the same $U$and the same condition D, since the choice of the elements being deleted may be arbitrary.
- NU_D : obtained by applying the distance threshold restriction D to NU.
- Uk, NUk, Uk_D, NUk_D, : similar to above; consisting of the union of solutions to the problem instance obtained by k algorithms being compared, where $k \geq 2$.
- $U-, NU-, U_D, NU_D$: similar to above, beginning with the union of solutions to the problem instance obtained by a collection of algorithms **excluding** the one being evaluated.

The choice of the reference set depends on the properties considered to be of importance for a specific algorithm comparison. Reference sets formulated without utilizing the solutions obtained using a specific algorithm are ideal, providing objective criteria that do not depend on the vagaries of the algorithms being evaluated; all the sets listed above satisfy this condition, except Uk, NUk, Uk_D, NUk_D, As a reasonable and sound experimental methodology, for instance, when two specific algorithms are being evaluated against each other, all **other** available algorithms may be used to generate the reference set.

Example: When

(a) the requirements of considering both non-domination and solution set diversity hold,
(b) only two algorithms are being compared,
(c) the Pareto set is unknown, and
(d) no other algorithms have so far been applied to that problem instance,

the right choice of the reference metric is expected to be $NU2_D$, where D eliminates some candidate solutions whose distance to others in the set (in objective space) is less than a prespecified problem-specific threshold. For instance, for the automobile buyer's decision-making problem with twin objectives of cost and comfort, this approach may delete one of two elements whose cost differs by less than $200 and estimated comfort level differs by less than 0.1. Alternatively, D may be specified in data space instead of objective space, e.g., $NU2_D$ may be prohibited from containing two cars made by the same manufacturer. Note, as in these examples, that the distance-related criterion D need not depend on

Euclidean distance in multi-dimensional space. Also, the choice of D does not uniquely determine $NU2_D$, in general.

The high-level approach we propose introduces the RefSet parameter into existing metrics. The next three sections explore the approach, developing RefSet-based metrics that are similar to several kinds of existing metrics.

3 Solution Set Cardinality

Counting the number of non-dominated solutions produced by an algorithm is a procedure that has come under some criticism, e.g., [6]. An algorithm could be rated highly because it produces a large number of candidate solutions, even when compared to another algorithm that produces a single Pareto-optimal solution that dominates all the others.

However, set cardinality can make more sense with respect to a RefSet, obtaining ordinal or comparative measures such as the following:

- An algorithm can be evaluated using the fraction of elements in the RefSet that are contributed by another algorithm; an algorithm can be considered to be better than another, with respect to the RefSet, if it contributes more elements to the RefSet.
- A milder, more practical criterion is to count the fraction of elements in the RefSet that are very near (according to some reasonable distance threshold in the objective space) to candidate solutions produced by an algorithm.

These measures do not directly rely on the cardinality of the solution set produced by the algorithms, but instead depend on the relationship between the RefSet and the solutions produced by the algorithms. The following properties hold for the above criteria:

- Algorithms are not rewarded for producing multiple near-identical candidate solutions.
- Algorithms are not directly penalized for producing only a small number of very good solutions.

Example: A RefSet consisting of $NU2$, augmented by problem-specific knowledge, contains three solutions from algorithm A, two solutions from algorithm B, and two others that are almost identical in objective function values to other solutions from B. Then, the first comparative measure argues that A is better than B since $3 \geq 2$, whereas the second measure supports the opposite conclusion, since B has generated solutions that are very near to four solutions in the RefSet.

Properties: We consider here some of the properties satisfied by a specific RefSet-based metric, suggesting the usefulness of this methodology. Let the ordinal metric $F(X; R)$ be defined as the fraction of solutions in RefSet R obtained

using algorithm X, and the associated comparison metric $B_F(X, Y; R)$ according to which algorithm X is better than algorithm Y if $F(X; R) \geq F(Y; R)$. This metric satisfies the following properties, in some cases only for appropriate choices of R:

- Transitivity: ($B_F(X, Y; R)$ and $B_F(Y, Z; R)$) implies $B_F(X, Z; R)$.
- Antisymmetry: $B_F(X, Y; R)$. implies $\neg B_F(Y, X; R)$.
- Compatibility with outperformance relations[3] [4]: Let all points in B be equal to or dominated by points in A, and let A contain at least one point that is not in B. Then $BF(A, B; NU2)$, but not necessarily $BF(A, B; NU)$ nor $BF(A, B; NU-)$ since none of the solutions in NU (or $NU-$) may come from A. In other words, an algorithm may produce solutions that dominate the solutions of another algorithm. But it is possible that both algorithms are equally bad as far as the RefSet is concerned, especially when the RefSet is constructed independently of the algorithms being compared.

Such analysis can be carried out for various metrics, identifying RefSets for which various properties such as [weak] compatibility with weak/strong/complete outperformance hold.

4 Domination Volume

Problems with the metric that uses domination volume involve the choice of the reference point (the origin in the graph if all objectives are to be maximized), as well as the incommensurability of different objective function dimensions[5]. Changing the units of measure from feet to meters or pounds to kilograms can result in substantially different results with respect to volume-comparison measures. The first decision, concerning the choice of the origin or reference point, involves asking the following question: which parts of the objective function space are truly of interest to the practitioner? For example, the automobile purchase decision-maker may rule out all vehicles costing more than $60,000, as well as those that do not meet minimal comfort standards. The second obstacle to volume-comparison measures, viz., non-commensurability of objective function dimensions, can be addressed to some extent by using a non-uniform problem-dependent scaling of the axes (objective functions) prior to the volume computation.

For instance, the decision-maker can be queried to determine the interval of values between which he/she is relatively indifferent, at various possible values for each objective function.

[3] Knowles and Corne [6] define the concepts of 'outperformance' and 'compatibility'.

[4] For ease of reading, we abuse notation, identifying algorithms with the nondominated solution sets they produce; Greek letters can be invoked if needed.

[5] Apples and oranges can be multiplied, but products of differences in the numbers of apples and oranges cannot be compared!

Example: An automobile purchaser may consider that a difference of $500 is significant when the purchase price is about $10,000, but not significant when the purchase price is about $60,000. Using this strategy, the two-dimensional objective function space could be divided into a collection of rectangular cells using non-uniformly spaced grid lines, and the volume comparison metric is transformed into one of comparing which cells are dominated by solutions generated by one algorithms but not the other. This approach provides a coarse measure for algorithm comparison that is more robust than the volume comparison metric, although it remains subject to the criticism that a cell in one part of the objective function space cannot be considered equivalent in size or importance to a cell in a completely different part of the space.

For example, it is possible that twenty cells are dominated by one algorithm but not by the other, whereas forty cells are dominated by the second algorithm but not the first. Superficially, the second algorithm is better using the cell-counting metric, but the justifiability of this comparison depends on the appropriateness of the problem-specific choice of the grid lines (or planes or hyperplanes) used to separate the objective function space into cells.

By moving from objective function dimension-based volume measures to more abstract cell-counting measures, the above discussion provides a guideline for the choice of volume-like RefSet-based metrics. These metrics are constructed using elements in the RefSet to determine cells, using an approach such as Voronoi tessellation in which each element of the RefSet corresponds to one cell or region in objective function space.

The applicability of this approach relies to some extent on the degree to which elements of the RefSet satisfactorily describe the Pareto set. When the true Pareto set is unavailable, of course, such reasoning is based on the assumption that the RefSet is the best available approximation to the Pareto set. Further, there is an implicit assumption that two cells are roughly equivalent in importance, suggesting that the RefSet should be modified or thinned out (e.g., NU_D defined in Section 2) to eliminate elements that are near-identical in objective function space.

5 Diversity

Classic Pareto set coverage and related metrics address how well the solutions generated by an algorithm encompass the Pareto-optimal solutions appropriate to the problem instance being considered. Where the Pareto set is known or easily determinable, this is exactly the special case where the RefSet is the same as the Pareto set.

In practical situations, we would instead have to use the RefSets described in Section 2, instead of the true Pareto set. The coverage issue is partly addressed by the metrics mentioned in Section 3 and 4, but the general question still remains: how do we evaluate an algorithm that generates some solutions that belong to the RefSet, and others that are near but are not identical to other solutions in the RefSet?

One aspect of this problem involves checking whether the solutions obtained by an algorithm are spread out widely in the objective function space, with respect to elements in the RefSet. If two algorithms generate equal numbers of elements in the RefSet, but one generates elements that are clustered together in objective function whereas the other does not, we may argue that the second is better than the first. One metric, to which this discussion leads, involves computing the average distance from each RefSet element to the nearest candidate solution generated by an algorithm, averaging over all RefSet elements but not averaging over all the solutions produced by the algorithm.

A related but different issue is the internal diversity of the solution set obtained by an algorithm (without reference to a RefSet). How widely spaced apart are elements of a solution set? However, spacing measures that focus exclusively on this issue are not very desirable; in particular, an algorithm that generates a vast number of well-spaced candidate solutions is not necessarily a good one, since many of these solutions may be of relatively poor quality with respect to the domination criterion, e.g., they may all be dominated by a single solution generated by a different algorithm. Such diversity metrics make sense only when considered along with a specific RefSet, as in the metric mentioned in the preceding paragraph.

6 Discussion

This paper has proposed that multi-objective optimization algorithms be evaluated using metrics that depend on Reference Sets. The goals of the comparison and problem details would be relevant in choosing the appropriate reference set. Some reasonable choices of reference sets were listed in Section 2. We have also discussed how we may adapt the metrics previously proposed in the literature to evaluate the number, diversity, and coverage properties of algorithms. However, we have only provided a framework for the formulation of such metrics in this paper. A careful study of the properties of a specific metric, in the lines of the evaluative study of [6], is needed before applying it to evaluate a specific algorithm. The last paragraph at the end of Section 3 shows that the properties of the metric depend significantly on the choice of the RefSet.

Acknowledgments

The authors thank Ramesh Rajagopalan and Pramod Varshney for prior discussions that helped formulate the ideas in this paper.

References

1. Arrow, K. J., "Social Choice and Individual Values", Second Edition, John Wiley & Sons, Inc. New York, 1963.

2. C. A. Coello Coello, " A Comprehensive Survey of Evolutionary-Based Multiobjective Optimization Techniqes", *Knowledge and Information Systems: An International Journal*, 1(3), 269–3087, August 1999.
3. K. Deb, "Multi-Objective Optimization using Evolutionary Algorithms", John Wiley & Sons, Chichester, UK, 2001.
4. C. M. Fonseca and P.J. Fleming, "An overview of evolutionary algorithms in multiobjective optimization", *Evolutionary Computation,* 3(1), Spring 1995.
5. J.D. Knowles, "Local-Search and Hybrid Evolutionary Algorithms for Pareto Optimization", *The University of Reading,* Reading, UK, January 2002.
6. J.D. Knowles and David Corne, "On Metrics for Comparing Nondominated Sets",*Congress on Evolutionary Computation (CEC2002)*, IEEE Service Center, Piscataway, New Jersey, 1, 711–716, May 2002.
7. R. Rajagopalan, C.K. Mohan, K.G. Mehrotra, and P.K. Varshney, "EMOCA: An Evolutionary Multi-Objective Crowding Algorithm", *Journal of Intelligent Systems,)*, 17 (1-3), 107–123, 2008.
8. David A. van Veldhuizen and Gary B. Lamont, "Multiobjective Evolutionary Algorithms: Analyzing the State-of-the-art", *Evolutionary Computation,* 8(2): 125–147, 2000.
9. A. Zhou, B-Y Qu, H. Li, S-Z Zhao, P. N. Suganthan, Q. Zhang, "Multiobjective evolutionary algorithms: A survey of the state-of-the-art", Swarm and Evolutionary Computation, Vol. 1, No. 1, pp. 32–49, March 2011.
10. Echart Zitzler, Lothar Thiele, Marco Laumanns, Carlos M. Fonseca, and Viviane Grunert da Fonseca, "Performance Assessment of Multiopjective Optimizers: An Analysis and Review", *IEEE Transactions on Evolutionaryy Computation*, 7(2), 117–130, 2003.
11. A. Zhou, B-Y. Qu, H. Li, S-Z. Zhao, P. N. Suganthan, Q. Zhang, "Multiobjective evolutionary algorithms: A survey of the state-of-the-art", *Swarm and Evolutionary Computation*, 1(1), 32–49, Mar. 2011.
12. A wikipedia document discussing Condorcets Voting Paradox. http://en.wikipedia.org/wiki/Voting_paradox.