Localized Probability of Improvement for Kriging based Multi-Objective Optimization

Yinjiang Li, Song Xiao, Mihai Rotaru, and Jan K. Sykulski, *Fellow, IEEE*
Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK
yl10e09@ecs.soton.ac.uk, jks@soton.ac.uk

***Abstract*—The paper introduces a new approach to kriging based multi-objective optimization by utilizing a local probability of improvement as the infill sampling criterion and the nearest neighbor check to ensure diversification and uniform distribution of pareto fronts. The proposed method is computationally fast and linearly scalable to higher dimensions.**

Keywords—kriging; multi-objective optimization; pareto front; surrogate-based optimization.

I. Introduction

Research on multiple objective optimization (MO) has been attracting significant attention of the engineering community since 1980s; with the aid of fast computers solutions to many complex optimization problems have been made possible. The Vector Evaluated Genetic Algorithm (VEGA) [1] is one of the earliest examples of Multi-Objective Evolutionary Algorithms (MOEAs). The more recent developments include NSGA-II [2] and its modified versions as well as Particle Swarm based methods [3]. For a comprehensive review of problem definitions and non-EA based solution methods, readers are referred to [4].

There is an increasing number of indicator-based MOEAs that have been proposed in recent years; the indicator is used as a fitness measure for a set of pareto points, and – by optimizing the indicator function – the MO problem essentially becomes a single objective optimization problem as the solver only needs to locate the optimal value of the indicator value and update the generation based on it. One of the best-known indicators is the hypervolume [5]; it has been successfully applied to both EAs and surrogate-based algorithms. Despite its unique feature of being strictly monotonic to pareto improvement [6], its high computational cost for higher dimensions is also widely known.

The general opinion favors EAs as advantageous in solving MO problems by often being population based, thus multiple solutions can be obtained in a single run. However, solutions to practical problems are usually expensive in terms of computational time and effort. In the context of electromagnetic devices the finite element method is a common design tool; it often takes hours or even days to obtain a single solution, therefore surrogate model based algorithms are often preferred.

In this study we propose a novel indicator focused Localized Probability of Improvement (LPoI) approach for MO problems; its implementation requires the predicted mean and mean square errors to be available, hence it is not applicable to other EAs, but for Gaussian based surrogate models (including those relying on kriging) it has the advantage of being linearly scalable to problems with higher number of objectives.

II. Localized Probability of Improvement

Compared to other surrogate modeling methods, kriging has the advantage of providing both the predicted mean and the associated mean square error (MSE) at an unknown location. The probability of improvement PoI at any location is given by

$$PoI(\boldsymbol{x}) = \Phi\left(\frac{y_t(\boldsymbol{x}) - \hat{y}(\boldsymbol{x})}{\hat{s}(\boldsymbol{x})}\right) \tag{1}$$

where y_t is the target of improvement, $\hat{y}$ is the kriging predicted mean at location $\boldsymbol{x}$, $\hat{s}$ is a square root of the mean square error at location $\boldsymbol{x}$ and $\Phi(\cdot)$ is the cumulative distribution function.

The algorithm presented in this paper uses a localized approach to define the probability of improvement at an unknown location; for simplicity we define the indicator as the LPoI which at a given point is calculated as follows

$${PoI_{ref}}^n(\boldsymbol{x}) = \Phi\left(\frac{{y_t}^n(\boldsymbol{x}) - \hat{y}^n(\boldsymbol{x})}{\hat{s}^n(\boldsymbol{x})}\right) \tag{2}$$

$${PoI_{ext}}^n(\boldsymbol{x}) = \Phi\left(\frac{{y_{ext}}^n - \hat{y}^n(\boldsymbol{x})}{\hat{s}^n(\boldsymbol{x})}\right) \tag{3}$$

$$LPoI_{ref}(\boldsymbol{x}) = \min\{PoI_{ref}(\boldsymbol{x})\} \tag{4}$$

$$LPoI(\boldsymbol{x}) = \max\{LPoI_{ref}, PoI_{ext}\} \tag{5}$$

where ${y_t}^n$, $\hat{y}^n$, $\hat{s}^n$, ${y_{ext}}^n$, ${PoI_{ref}}^n$ and ${PoI_{ext}}^n$ are the corresponding measures of the n^{th}objective function, ${PoI_{ref}}^n$ is the probability of improvement calculated on the basis of reference points, ${PoI_{ext}}^n$ is the probability of improvement calculated based on minimum objective function, y_t is the target improvement using reference points, y_{ext} is the target improvement over each individual objective, and finally PoI_{ref} and PoI_{ext} are collections of ${PoI_{ref}}^n$ and ${PoI_{ext}}^n$, respectively.

The ${PoI_{ext}}^n$ term, as described by (3), is due to the fact that the minimum of each individual objective function is always present in the pareto front, hence PoI at each location $\boldsymbol{x}$ over the optimal target of that function is always considered. This term also contributes to the diversification of the pareto front, while the associated improvement target y_{ext} is given by

$${y_{ext}}^n = {y_{min}}^n \cdot (1 - p) \tag{6}$$

where ${y_{min}}^n$ is the known minimum value of the n^{th} objective function and p is the percentage of improvement, the analytical form of which is shown later by (9).

The term ${PoI_{ref}}^n$ in (2) is calculated at different locations of $\boldsymbol{x}$ based on the local improvement targets; a target is calculated for a reference point, and the reference point is taken based on the location of $\boldsymbol{x}$. In order to obtain the reference points, the algorithm finds the pareto front for existing design sites, using non-dominated sorting, and then for each closest set of pareto points (the number of points equals to the number of objectives), calculates the corresponding reference point. The coordinate of the reference point for each dimension is equal to the maximum value of these pareto points in the same dimension. Hence the reference point in the n^{th} dimension is given by

$$ {y_{ref}}^n = \max\{Y^n\} \tag{7} $$

where Y^n is the collection of the n^{th} objective values of all points in that Pareto set. The improvement target ${y_t}^n$ associated with each reference point is then given by

$$ {y_t}^n = {y_{ref}}^n \cdot (1 - p) \tag{8} $$

The percentage of improvement p controls the convergence rate of the algorithm and is given by

$$ p = p_{initial} \cdot \max\{LPoI_{prev}\} \tag{9} $$

where $p_{initial}$ is the initial improvement target to be defined (it is set to 0.1 in our tests) and $LPoI_{prev}$ is the set of $LPoI$ from all points at the previous iteration. To obtain the next infill sampling point, the algorithm finds the location $\boldsymbol{x}$ associated with the maximum LPoI measure in the criterion space.

The solver based on the LPoI criterion will tend to minimize the localized PoI and converge towards the pareto front. When the design space is well explored, or p is very small, the solver will converge to existing pareto fronts; at this stage it is common for a multiple number of unknown sites to have the LPoI equal or close to 1 (very likely to improve over the target point). In order to obtain a uniformly distributed pareto front, the algorithm selects candidates which have the largest Euclidean distance to existing pareto points as the next infill sampling points. Because of this reason given above, the maximum value of LPoI can be capped at 0.95 or 0.99 for faster exploitation of the existing pareto front.

III. Test Examples and Results

In this section we illustrate the proposed method using a bi-objective example.

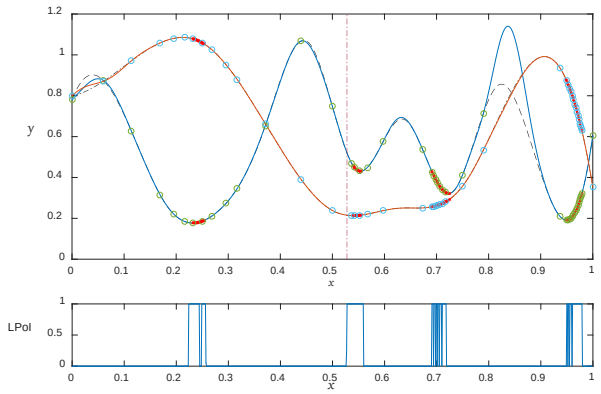
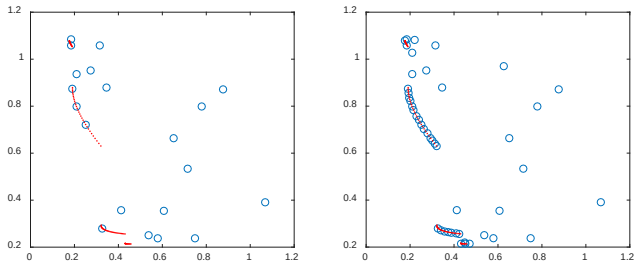


Fig. 1. The kriging model and the LPoI criterion in the search space.



(a) (b)

Fig. 2. (a) Existing design sites in the criterion space at 20th iteration; (b) Existing design sites in the criterion space at 45th iteration.

The top graph in Fig. 1 shows the kriging model (solid line) after the 45th iteration, with the red crosses plotted at the true pareto front, while the bottom plot shows the proposed indicator value for the unknown sites. As can be seen, the algorithm has correctly converged to all four pareto point clusters in the search space and thus further sampling will lead to more exploitation on the pareto front. The sampled design sites in the criterion space are plotted in Fig. 2, where the red dots indicate the location of the true pareto front.

IV. Solving the New TEAM Problem

A new TEAM problem is about to be proposed at the forthcoming Compumag conference in Korea, June 2017. This is devoted specifically to multi-objective optimization and the intention is to test our algorithm on this new benchmark problem; full details will be provided in the extended version.

V. Conclusion

A novel approach to kriging-based multi objective optimization is proposed and details are discussed in this paper. A bi-objective problem is illustrated, while the method will be tested against the new TEAM benchmark problem. The proposed method addresses efficiently both the diversification and uniformity of the pareto solution, is computationally efficient and is linearly scalable to higher number of objectives.

References

[1] J. David Schaffer, "Multiple objective optimization with vector evaluated genetic algorithms," Proceedings of the 1st International Conference on Genetic Algorithms, pp. 93-100, 1985.

[2] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan, "A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II," in Schoenauer M. et al. (eds), Parallel Problem Solving from Nature PPSN VI, 2000.

[3] K.E. Parsopoulos and M.N. Vrahatis, "Particle swarm optimization method in multiobjective problems," SAC '02 Proceedings of the 2002 ACM symposium on Applied computing, pp. 603-607, 2002.

[4] R.T. Marler and J.S. Arora, "Survey of multi-objective optimization methods for engineering," Structural and Multidisciplinary Optimization, vol. 26, no. 6, pp. 369, 2004.

[5] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach," IEEE Transactions on Evolutionary Computation, vol. 3, no. 4, pp. 257-271, 1999.

[6] J. Knowles and D. Corne, "On metrics for comparing nondominated sets," Evolutionary Computation, Proceedings of CEC'02, Honolulu, HI, pp. 711-716, 2002.