Active learning of Pareto fronts with disconnected feasible decision and objective spaces

Paolo Campigotto[1], Andrea Passerini[1], Roberto Battiti[1]

Department of Information Engineering and Computer Science
University of Trento
Via Sommarive 14, I-38123 Povo, TN, Italy
{campigotto,passerini,battiti}@disi.unitn.it

1 Introduction

A multi-objective optimization problem (MOP) is formulated as the joint minimization of m conflicting objective functions $f_1(\mathbf{x}), \ldots, f_m(\mathbf{x})$ w.r.t a vector $\mathbf{x}$ of n decision variables. Typically, $\mathbf{x} \in \Omega$, where $\Omega \subset \mathbb{R}^n$ is the *feasible region*, defined by a set of constraints on the decision variables. Objective vectors are images of decision vectors and can be written as $\mathbf{z} = \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_m(\mathbf{x}))$, with $\mathbf{z} \in \Phi$, where Φ is the image of Ω, i.e., $\mathbf{f} : \Omega \to \Phi \subset \mathbb{R}^m$. An objective vector $\mathbf{z}$ is said to *dominate* $\mathbf{z}'$, denoted as $\mathbf{z} \succ \mathbf{z}'$,if $z_k \leq z'_k$ for all k and there exists at least one h such that $z_h < z'_h$. A point $\mathbf{x}$ is Pareto-optimal if there is no other $\mathbf{x}' \in \Omega$ such that $\mathbf{f}(\mathbf{x}')$ dominates $\mathbf{f}(\mathbf{x})$. The set of Pareto-optimal solutions is called *Pareto set* (PS). The corresponding set of Pareto-optimal objective vectors is called *Pareto front* (PF).

The Active Learning of Pareto fronts (ALP) algorithm [1] learns an *analytical* model of the Pareto front from a training set of approximated Pareto-optimal vectors. The training Pareto-optimal vectors are obtained by solving different scalarized instances of the original MOP. In order to minimize the computational effort (measured as number of evaluations of the MOP objective functions), informative training objective vectors are selected by applying active learning principles. The experimental results reported in [1] show that ALP outperforms the state-of-the-art MMEA and NSGA-II algorithms over widely-used continuous-optimization benchmarks, including a set of four well-known MOPS with disconnected Pareto front. However, the benchmarks considered in the experimental comparison have *connected* feasible decision and objective spaces.

This paper highlights a possible generalization of ALP to tackle continuous MOPs where *the feasible decision and objective spaces are disconnected*. To validate the ALP extension, the formulation of a well-known continuous MOP is modified to obtain disconnected feasible decision and objective spaces. We are not aware of established benchmark problems in the literature with this feature. Our contribution can also be considered a first attempt to fulfill this lack, in the spirit of simulating real-world optimization tasks.

2 The ALP algorithm

Under mild smoothness conditions on the objective functions, the PF exhibits a regular structure. In particular, the PF of a problem with m objectives is a $(m-1)$-dimensional piecewise-continuous manifold [2]. Furthermore, the dominance relation enables a functional formulation of the PF. As matter of fact, the PF can be characterized by expressing an arbitrary objective as a function g of the remaining objectives.

The current version for the ALP algorithm focuses on bi-objective problems, where the Pareto front is a piecewise-continuous curve $z_2 = g(z_1)$. An equivalent formulation consists of expressing z_1 as a function of z_2. Without losing generality, we adopt the former formulation. Possible generalizations of ALP with an arbitrary number of objectives are under investigation. Under the above regularity assumptions, ALP casts the identification of the Pareto front into a supervised regression task. The target of the regression task is an approximation $\tilde{g}$ of the unknown function g, with the input and the output of the regression problem being the independent objective z_1 and the dependent objective z_2, respectively. The approximation $\tilde{g}$ is learnt from a set of approximated Pareto-optimal vectors, each one providing a training example (z_1, z_2). The training set is generated iteratively by applying the uncertainty sampling principle (active learning). At each iteration, a new informative training example is generated first by selecting the input $\hat{z}_1$ where the prediction of the current PF model $\tilde{g}$ is most uncertain and then by computing the supervised information for $\hat{z}_1$. The supervised information consists of the output $\hat{z}_2 = g(\hat{z}_1)$, obtained by solving the single-objective optimization task formulated by the mathematical program (1), where $\Omega \subseteq \mathbb{R}^n$ identifies the decision region of the MOP. Let $\hat{\mathbf{x}}$ be the *exact* solution of program (1). Then, $\hat{z}_2 = f_2(\hat{\mathbf{x}})$. The objective vector $\hat{\mathbf{z}} = \{\hat{z}_1, \hat{z}_2\}$ is Pareto-optimal, i.e., $\hat{z}_2 = g(\hat{z}_1)$. The slack variable ϵ in program (1) relaxes the equality constraints $f_1(\mathbf{x}) = \hat{z}_1$. When an *approximated* solution of the above problem is obtained, a *noisy* training example is generated.

$$\begin{aligned} \min_{\mathbf{x},\epsilon} \quad & f_2(\mathbf{x}) \\ \text{subject to} \quad & \\ & f_1(\mathbf{x}) = \hat{z}_1 + \epsilon \\ & |\epsilon| \leq 10^{-2} \\ & \mathbf{x} \in \Omega \end{aligned}$$

Program 1: Generation of the supervised information for a selected input $\hat{z}_1$.

$$\begin{aligned} \min_{\mathbf{x}} \quad & \hat{z}_1 - f_1(\mathbf{x}) \\ \text{subject to} \quad & \\ & z_1^{cl} \leq f_1(\mathbf{x}) \leq \hat{z}_1 \\ & \mathbf{x} \in \Omega \end{aligned}$$

Program 2: Identification of the infeasible-interval lower bound.

The ALP algorithm consists of an initialization phase followed by a refinement phase. The former stage provides an initial approximation $\tilde{g}$ from s approximated Pareto optimal vectors obtained by solving s instances of the problem (1) for s inputs $\hat{z}_1$ selected uniformly at random in the regression domain. We used $s = 1$ in the experiments reported below. The latter stage refines $\tilde{g}$ by iteratively increasing the training set with the example selected by the active learning principle, and re-training the model for the following refinement. ALP stops when the predictive uncertainty of the model $\tilde{g}$ is negligible over the whole regression domain (i.e., the information gain obtained from *any* additional training example is negligible), or a limit on the number of training iterations or MOP function evaluations has been reached. A detailed description of ALP can be found in [1].

3 Extension to tackle disconnected feasible decision and objective spaces

The work in [1] also extends the basic framework of ALP, sketched in the previous section, to solve continuous MOPs with a disconnected Pareto front, but with *connected* feasible decision and objective spaces. In this case, ALP algorithm learns an approximation $\tilde{h}$ of the whole lower boundary h of the objective space. By definition, h entails the Pareto front of the MOP. An approximation $\tilde{g}$ of g can thus be obtained from $\tilde{h}$ by applying the dominance relation.

However, real-world continuous-optimization tasks may exhibit *disconnected* feasible decision and objective spaces. When the feasible objective space is disconnected, the problem (1) may be infeasible. In this case, the input $\hat{z}_1$ belongs to an interval where h is not defined (infeasible interval). Our extension of ALP identifies the lower bound and the upper bound of the infeasible interval and updates the regression domain by removing the infeasible interval. From now on, the ALP extension will be referred to by the acronym ALPI, where the final letter “I” denotes the ability in handling infeasible intervals.

The lower bound is obtained by solving the non-linear program (2). Let us denote by $z_1^{cl} = f_1(\mathbf{x}^{cl}) < \hat{z}_1$ the training input closest to the infeasible query input $\hat{z}_1$. Program (2) consists of searching for the feasible decision vector $\mathbf{x}$, with $f_1(\mathbf{x}) \in [\hat{z}_1^{cl}, \hat{z}_1]$, minimizing the distance between $\hat{z}_1$ and $f_1(\mathbf{x})$. A robust approach to solve the non-linear program (2) is provided by the multi-start continuous local search method introduced in [3] and implemented in the Matlab global optimization toolbox. The continuous local search algorithm is initialized with the feasible starting point $\mathbf{x}^{cl}$. When there are no training inputs $z_1 \leq \hat{z}_1$, ALPI first identifies one feasible point $\mathbf{x}$ with $f_1(\mathbf{x}) < \hat{z}_1$ by maximizing the distance $\hat{z}_1 - f_1(\mathbf{x})$ and then solves the non-linear program (2). An analogous procedure is adopted to identify

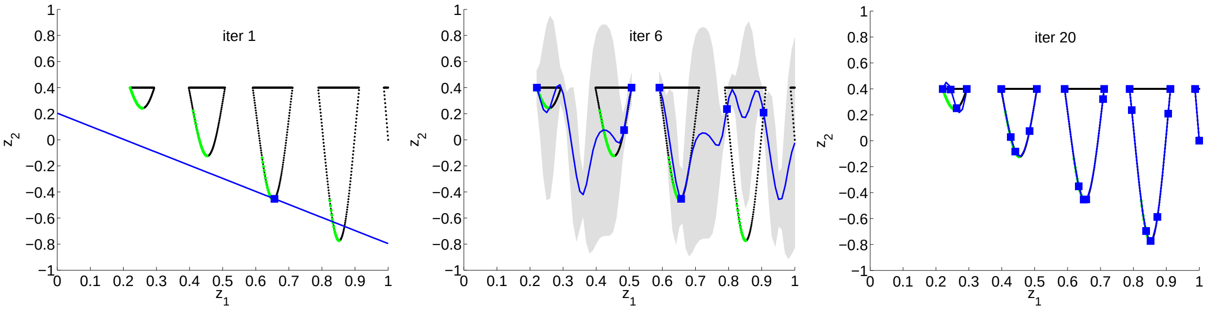

Figure 1: Selected iterations from a sample run of ALPI, showing the identification of the domain of h and the generation of informative training examples. The solid black lines define the boundary of the disconnected components of the feasible objective space, while the disconnected PF is highlighted with the green color. The solid blue lines define the piecewise-continuous approximation $\tilde{h}$ learnt by ALP from the training set represented by the square-marked points. The gray-shaded area denotes the predictive uncertainty of the learnt model.

the upper bound of the infeasible interval.

4 Experimental results

The formulation of the widely-used ZDT3 MOP from the benchmark suite [4] has been modified to obtain disconnected feasible decision and objective spaces. In Fig. 1, the solid black lines define the disconnected components of the feasible objective space, while the disconnected PF is highlighted with the green color. The solid blue lines define the piece-wise continuous approximation $\tilde{h}$ learnt by ALP form the training set represented by the square-marked points. The gray-shaded area denotes the predictive uncertainty of the PF model (a larger area corresponds to more uncertain predictions). The different figures refer to refinement iterations one (i.e., the initialization phase, with only one initial training example), six and twenty of a selected ALPI run. Comparable results are observed for different runs. When increasing the number of iterations, the disconnected domain of $\tilde{h}$ is correctly identified. At the sixth refinement iteration, the infeasible interval located between the values 0.5 is and 0.6 is recovered. At iteration twenty all the infeasible intervals are accurately identified, while the predictive uncertainty of the learn model is sensibly reduced. Therefore, ALPI can learn an accurate model $\tilde{h}$ of the lower boundary h, with the exception of the leftmost feasible component of the disconnected feasible objective space, where noisy training examples affect the quality of the approximation. Future work will be devoted to increase the robustness to noisy training examples and to reduce the computational effort (in terms of the number of evaluations of the original MOP objectives).

Acknowledgments. This research was partially supported by grant PRIN 2009LNP494 (Statistical Relational Learning: Algorithms and Applications) from Italian Ministry of University and Research.

References

[1] Paolo Campigotto, Andrea Passerini, and Roberto Battiti. Active learning of Pareto fronts. Technical Report number DISI-13-001, University of Trento, Italy, 2012.

[2] C. Hillermeier. *Nonlinear Multiobjective Optimization: A Generalized Homotopy Approach*. International Series of Numerical Mathematics. Birkhaüser Verlag, 2001.

[3] Z. Ugray, L. S. Lasdon, J. C. Plummer, F. Glover, J. P. Kelly, and R. Marti'. Scatter Search and Local NLP Solvers: A Multistart Framework for Global Optimization. *INFORMS Journal on Computing*, 19(3):328–340, 2007.

[4] E. Zitzler, K. Deb, and L. Thiele. Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. *IEEE Transactions on Evolutionary Computation*, 8(2):173–195, 2000.