Honda Research Institute EU
Honda Research Institute Europe GmbH
https://www.honda-ri.de/

Learning-guided Exploration in Airfoil Optimization

Edgar Reehuis, Markus Olhofer, Bernhard Sendhoff, Thomas Bäck

2013

Preprint:

This is an accepted article published in International Conference on Intelligent Data Engineering and Automated Learning (IDEAL 2013). The final authenticated version is available online at: https://doi.org/[DOI not available]

Learning-guided Exploration in Airfoil Optimization

Edgar Reehuis[1,2], Markus Olhofer[2], Bernhard Sendhoff[2], and Thomas Bäck[1]

[1] Natural Computing Group, LIACS, Leiden University
Niels Bohrweg 1, 2333 CA Leiden, The Netherlands
{ereehuis,baeck}@liacs.nl
http://natcomp.liacs.nl
[2] Honda Research Institute Europe GmbH
Carl-Legien-Straße 30, 63073 Offenbach/Main, Germany
{markus.olhofer,bernhard.sendhoff}@honda-ri.de
http://www.honda-ri.de

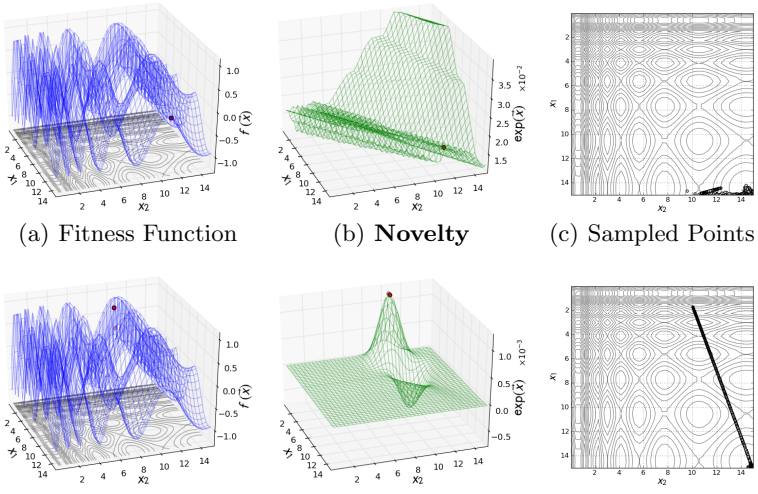
Abstract. A learning-based exploration approach is proposed to escape from the basins of attraction of converged-to optima, by selecting on what is termed the interestingness of a solution. This interestingness is based on the modeling error made by a surrogate model that is trained on all solutions encountered earlier during the search. Compared to multiple standard optimization runs, a learning-guided restart scheme that alternates between a quality optimization phase and an exploration phase directed by interestingness finds solutions that are more diverse and of higher quality.

Keywords: Novelty, interestingness, diversity, restarts, airfoil optimization

1 Introduction

In airfoil optimization the search is sometimes started from a basic initial solution, required when it is not straightforward to generate a random initial solution to start optimization from, in case that simulation of aerodynamic behavior only converges for solutions that meet specific geometric constraints. Different runs of a stochastic search approach may then result in similar optimized solutions.

In order to escape the "default" basin of attraction we propose using an additional search objective specifically aimed at exploration. An overview of possible measures that derive from the history of solutions encountered earlier during the search is given in [4], making a division between similarity-derived, *distance-based* measures, and *learning-based* measures related to fitting a *surrogate model*. A scheme that alternates between a quality optimization phase and a distance-based exploration phase is described in [5], while in this work we present a similar scheme using a learning-based guide, both applied to airfoil optimization. The surrogate model is *not* intended here for reducing the number of used quality function evaluations, but for indicating the modeling error and thereby the *understanding* in a certain region during optimization.



(a) Fitness Function (b) **Novelty** (c) Sampled Points

(d) Fitness Function (e) **Interestingness** (f) Sampled Points

Fig. 1. Novelty vs. Interestingness. Sampling behavior displayed for a (3,6)-CMA-ES [2] selecting on novelty vs. its sampling behavior in selecting on interestingness. Selecting on novelty causes stagnation (see [4]), while in selecting on interestingness newly sampled solutions get high scores that degrade to negative interestingness over time, thereby sustaining exploration.

Following, Section 2 lays out the learning-based exploration measure and how it is integrated as exploration criterion in the search, after which the results of applying the resulting scheme to an airfoil optimization task are given in Section 3. We conclude with discussion and outlook in Section 4.

2 Novelty and Interestingness

Learning-based novelty is expressed as the error that a surrogate model makes in approximating the mapping of solution vectors to, e.g., quality-function values. In selecting on learning-based novelty, solutions with maximum error are promoted. However, the search can get stuck in areas for which the model's approximation does not improve over time (see Fig. 1). To steer clear from regions that effectively cannot be modeled, we should instead of on maximum error select the solution that induces maximum *learning progress*, that is, the observed improvement in approximating the to-be-learned mapping by including the solution in model training. This is estimated by the *interestingness* of a solution, which is derived from the solution's novelty value as well as the novelty values of earlier sampled solutions (see Fig. 2).

Dispersion in Predictions. For expressing learning-based novelty we use *dispersion in predictions* (DP) [4]. Multiple models are trained on the same data,

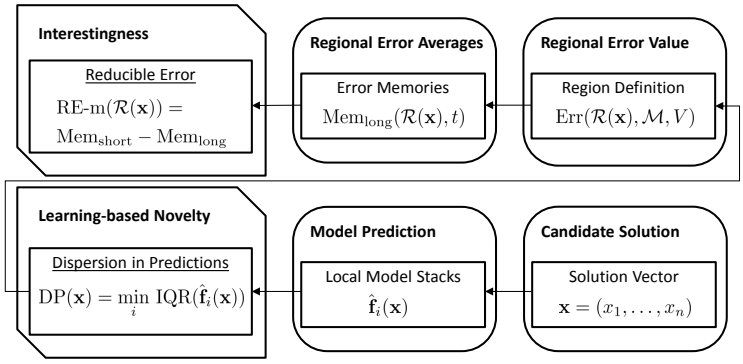


Fig. 2. Calculation of Interestingness Expression RE-m (DP). Starting from the *learning-based novelty* of solutions at their time of sampling, the influence of these earlier observed error values is calculated (*regional error value*) depending on their proximity to the point for which interestingness is to be evaluated. An interestingness value is then obtained based on how the error in the region in which the solution lies has changed over time (*regional error averages*).

and the dispersion between their predictions serves as estimation of the modeling error. DP requires surrogate models with potentially differing outcomes in presence of limited training data. We calculate DP as the *interquartile range* (IQR) of the quality predictions for a solution $\mathbf{x}$ by the $\dim(\hat{\mathbf{f}})$ stacked models,

$$\mathrm{DP}(\mathbf{x}) = \mathrm{IQR}\left(\hat{f}_1(\mathbf{x}), \ldots, \hat{f}_{\dim(\hat{\mathbf{f}})}(\mathbf{x})\right). \tag{1}$$

To be prepared for modeling irregular quality landscapes, we use multiple *local model stacks* that together emulate a global model stack (see Fig. 3). The different local model stacks are trained on disjoint or partly shared data. DP is then calculated as the minimum IQR score resulting from evaluating a solution $\mathbf{x}$ on all local model stacks,

$$\mathrm{DP}(\mathbf{x}) = \min_i \ \mathrm{IQR}\left(\hat{\mathbf{f}}_i(\mathbf{x})\right). \tag{2}$$

Reducible Error. Using the global model stack $\mathcal{M}(T_t)$ we obtain from observed DP-values a *regional error value* for the region $\mathcal{R}(\mathbf{x})$ in which solution $\mathbf{x}$ lies [4],

$$\mathrm{Err}(\mathcal{R}(\mathbf{x}), \mathcal{M}(T_t), V_t) = \max_{\mathbf{v} \in V_t} \left(\underbrace{\mathrm{DP}(\mathbf{v})}_{\text{on } \mathcal{M}(T_t)} \cdot \underbrace{(2\pi)^{-\frac{1}{2}|\mathbf{v}|} \exp\left(-\tfrac{1}{2}(\mathbf{x}-\mathbf{v})'\mathbf{I}(\mathbf{x}-\mathbf{v})\right)}_{\mathrm{pdf}_{\mathcal{N}(\mathbf{v},\mathbf{I})}(\mathbf{x})} \right), \tag{3}$$

that is, we determine test point $\mathbf{v}$ with the highest resulting DP-value after decreasing it relative to the distance between $\mathbf{x}$ and $\mathbf{v}$, based on the probability

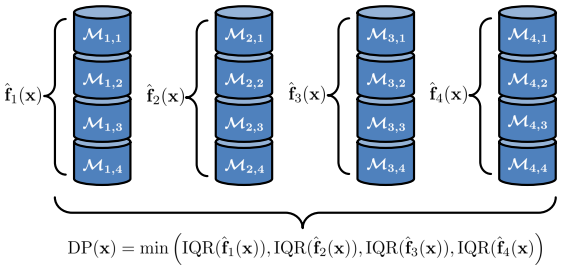


Fig. 3. Local Model Stacks. A global model stack is obtained using an ensemble of local model stacks. Within local stack i all surrogate models $\mathcal{M}_{i,j}$ are trained on the same training data, while different local stacks use data from different areas in the search space. The lowest DP score is selected after evaluating a solution $\mathbf{x}$ on all stacks.

density function of a normal distribution around $\mathbf{v}$. The global model stack is trained on the set T_t of all solutions evaluated before time step t, and test set V_t contains all candidate solutions $\mathbf{x}$ at time step t. If no training data is available at time t, $\mathrm{Err}(\mathcal{R}(\mathbf{x}), \mathcal{M}(T_t = \emptyset), V_t)$ is set to 0.

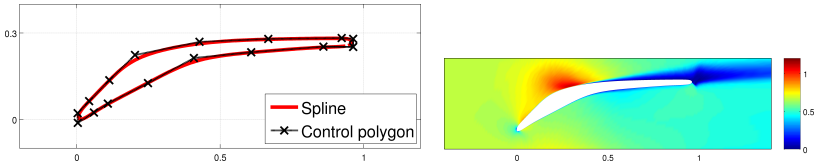
From the regional error values we obtain *regional error averages* at different points in time, in the form of a *short-term* and *long-term memory*. We express interestingness by assuming that the learning progress will be maximized in a region with high *current* error, as the model is still able to greatly improve there, but account for regions that cannot be modeled by subtracting the *irreducible* error, that is, the error in the region at an earlier time step. This results in the *reducible error* (RE) [4], proposed by Luciw et al. [3],

$$\text{RE-m}(\mathcal{R}(\mathbf{x})) \;=\; \underbrace{\mathrm{Mem}_{\mathrm{short}}(\mathcal{R}(\mathbf{x}), t)}_{\underline{\text{current}}\text{ error in }\mathcal{R}(\mathbf{x})} \;-\; \underbrace{\mathrm{Mem}_{\mathrm{long}}(\mathcal{R}(\mathbf{x}), t)}_{\underline{\text{irreducible}}\text{ error in }\mathcal{R}(\mathbf{x})}\,, \tag{4}$$

using *exponential smoothing* with weighting for the memories as proposed in [3],

$$\begin{aligned}\mathrm{Mem}_{\mathrm{short}}(\mathcal{R}, t) &= (1 - 0.2) \cdot \mathrm{Mem}_{\mathrm{short}}(\mathcal{R}, t-1) + 0.2 \cdot \mathrm{Err}(\mathcal{R}, \mathcal{M}(T_t), V_t),\\ \mathrm{Mem}_{\mathrm{long}}(\mathcal{R}, t) &= (1 - 0.1) \cdot \mathrm{Mem}_{\mathrm{long}}(\mathcal{R}, t-1) \;+ 0.1 \cdot \mathrm{Err}(\mathcal{R}, \mathcal{M}(T_t), V_t).\end{aligned} \tag{5}$$

Learning-guided Restart Scheme. To apply interestingness as exploration criterion, we take inspiration by Cuccu et al. [1] and use a restart scheme that alternates between a fitness optimization phase and an exploration phase, originally proposed for distance-based exploration in [5]. The alternating scheme is aimed at escaping basins of attraction of earlier converged-to optima. For the considered airfoil optimization, the search space consists largely of areas with *infeasible* solutions (i.e., violating some geometric constraint, or for which the performance cannot be simulated). We perform optimization on fitness until some convergence criterion is met, and then optimize on interestingness until all solutions in the current population have become infeasible. Following, the fitness optimization phase is restarted, using a variable penalty for infeasible



(a) Spline Representation (b) Mach Number Distribution

Fig. 4. Initial Geometry. In (a) the 16 *control points* and their *non-uniform rational B-spline* (NURBS) decoded geometry is shown. In (b) the *Mach number distribution* (the relative air speed divided by the local speed of sound) of the simulated *flow field* around the geometry is displayed, showing a deviating outflow for the initial geometry.

solutions depending on the degree of infeasibility to be able to find a way back to feasible space. The alternating scheme is run on top of the *CMA-ES* [2]: Between restarts its covariance matrix is reinitialized and its stepsize reset to the, problem-specific, initial value.

3 Experiments

Two-dimensional optimization is performed on the *outlet guide vanes* (OGVs) for the *low-pressure compressor* in a small turbofan engine, with 16 control points for adjusting a geometry, thus 32 decision variables (see Fig. 4). OGVs are *stator blades*, non-moving blades intended for straightening the air flow, arranged around a central axis in a cylindrical fashion. Simulating aerodynamic behavior is done using an in-house deterministic *Navier-Stokes flow solver*.

The learning-guided restart scheme is run on top of a (12,24)-CMA-ES for 5000 generations (120k evaluations by the simulator). It is compared against multiple standard optimization runs using the same (12,24)-CMA-ES for 1000 generations each (24k evaluations). A problem-specific initial stepsize σ_{init} of 1×10^{-3} is used.

Within the learning-guided restart scheme we stop the fitness optimization phase when the stepsize is smaller than $0.5 \times \sigma_{\text{init}}$. In the exploration phase the stopping condition is that either all candidates have become strictly infeasible (a geometry is disconnected, i.e., contains loops, or the simulator does not converge for it), or the search is in a maximally erratic area for still allowing to find a way back to a high-performing solution (the average fitness value of the best individual over the last 5 generations exceeds 1×10^3).

A local model stack consists of 10 *feed-forward neural networks* (FFNN) and each generation a new stack is trained on the last 3 generations of evaluated solutions. An ensemble of disjoined local stacks is maintained by keeping the local model stack of each third generation. The FFNN have linear outputs and a hidden layer of 25 sigmoidally-activated nodes, a bias node, and are trained using improved Rprop [4] for a maximum of 10k epochs from initial connections weights in $[-0.1, 0.1]$. They are fully connected, including direct connections from input to output nodes. All data is used for training, no validation set is

used. The CMA-ES and learning-guided restart scheme have been implemented using the Shark Machine Learning Library v2.3.3.

The target of the search is twofold: Finding a single solution of *maximal* quality and producing a set of *diverse* solutions with *acceptable* quality. To omit a priori specifying a level of acceptable quality, we take the quality value of each solution in a method's result set as possible minimal level. This gives rise to multiple alternative *level set* approximations within the result set for which we can determine the diversity. Each such "solution level set" contains the solution itself and all other found solutions that have quality equal to or better than the solution's quality score, and this way we associate a diversity score with each solution. From the resulting quality/diversity pairs a two-dimensional *Pareto front* is obtained that illustrates the performance of the method.

Fitness Function. The quality of a solution is based on seven properties that are aggregated into a to-be-minimized fitness value: The resulting deviation angle of the air flow, the coefficient of the occurring pressure loss, the extent of violation of four minimal thickness constraints along the blade, and a penalty depending on the extent to which the solver has not converged.

Diversity Calculation. We use *Solow-Polasky* [5] to express diversity in a set of solutions S by applying *domain-specific distance measure* $\mathrm{d}_{\mathrm{airfoil}}$ [5],

$$\mathrm{D}_{\mathrm{SP}}(S) = \sum_{i,j} b_{ij}, \tag{6}$$

with $\boldsymbol{B} = \boldsymbol{A}^{-1}$, $a_{ij} = \exp(-\mathrm{d}_{\mathrm{airfoil}}(\mathbf{x}_i, \mathbf{x}_j))$, and $1 \leq i, j \leq |S|$, that is, calculate the sum of the elements b_{ij} of $\boldsymbol{B}$, the inverse of $\boldsymbol{A}$. The elements a_{ij} of $\boldsymbol{A}$ depend on the difference between all pairs of solutions $\mathbf{x}_i, \mathbf{x}_j$ from set S. $\mathrm{D}_{\mathrm{SP}}(S)$ results in a real number in the interval $[1, |S|]$. For calculating $\mathrm{d}_{\mathrm{airfoil}}$ a geometry $P(\mathbf{x})$ is sampled in 200 points equidistantly along the spline. Each sample is divided in a *suction* side (side featuring higher air speed, upper side in Fig. 4b) set and *pressure* side (side featuring lower air speed, lower side in Fig. 4b) set, and the *summed-up Hausdorff distance* $\mathrm{d}_{\Sigma\mathrm{H}}$ [5] between the respective sets is added up,

$$\mathrm{d}_{\mathrm{airfoil}}(\mathbf{x}_1, \mathbf{x}_2) = \mathrm{d}_{\Sigma\mathrm{H}}(\mathrm{s}(P(\mathbf{x}_1)), \mathrm{s}(P(\mathbf{x}_2))) + \mathrm{d}_{\Sigma\mathrm{H}}(\mathrm{p}(P(\mathbf{x}_1)), \mathrm{p}(P(\mathbf{x}_2))), \tag{7}$$

with

$$\mathrm{d}_{\Sigma\mathrm{H}}(A, B) = \sum_{\mathbf{a}\in A} \min_{\mathbf{b}\in B} \|\mathbf{a} - \mathbf{b}\| + \sum_{\mathbf{b}\in B} \min_{\mathbf{a}\in A} \|\mathbf{a} - \mathbf{b}\|. \tag{8}$$

Results. In Fig. 5 the outcome of a single run of the learning-guided restart scheme is plotted against the outcome of 7 standard fitness optimization runs combined (using 24 processor cores in parallel per run, time to complete per run approximately 3 weeks and approximately 30 hours, respectively), and in Fig. 6 the geometry and Mach number distribution of three found solutions per

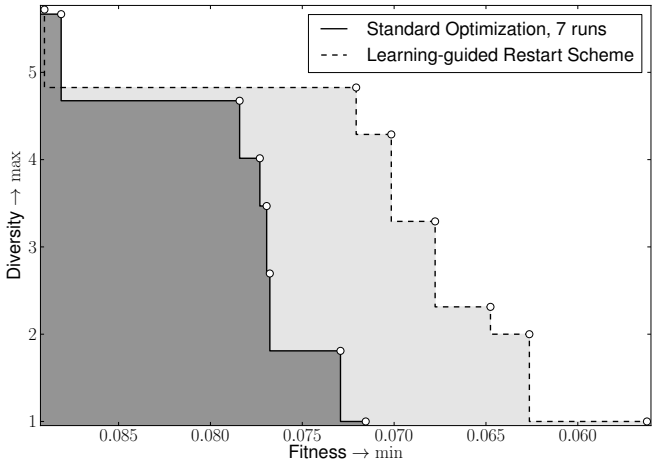


Fig. 5. Fitness vs. Diversity. Per method the diversity and quality threshold of each possible level set in its result set is plotted, governed by the fitness values of the solutions contained in the result set. The learning-guided scheme results in a larger dominated surface in fitness/diversity space, while using only 120k vs. 168k evaluations.

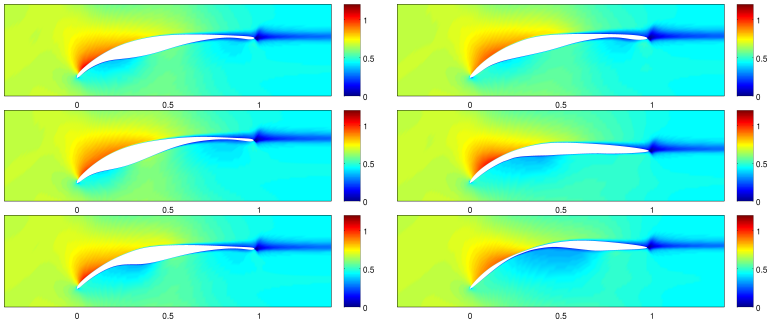
method is displayed. While using less evaluations, the learning-guided restart scheme finds solutions of better quality than the standard optimization runs, and as can be seen in Fig. 6, its solutions are more diverse. However, given that its best solution adheres to the thickness constraints, the learning-guided scheme seems to require adjustment of these as the geometry appears too thin to be feasible in practice.

4 Conclusion and Outlook

An alternating restart scheme was proposed that after convergence of a standard quality optimization phase starts an exploration phase in which is selected on a learning-based interestingness measure [4]. This measure directs the search into areas for which a surrogate model trained on all sampled solutions features high error, but also causes each such area to be left again after a few time steps and thereby sustains exploration.

In earlier work [5] we described a similar alternating scheme that applies a distance-based exploration measure which depends on the similarity to earlier sampled solutions with respect to a domain-specific distance measure. Both schemes were applied to an airfoil optimization task in which the exploration phase continues until the search ends up in infeasible space. A search space containing large pockets of solutions that violate geometric constraints or for which aerodynamic performance cannot be simulated is typical for this problem class.

Compared to multiple standard optimization runs, the learning-guided restart scheme finds solutions with strictly higher quality, and that are more diverse with



(a) Standard Optimization (b) Learning-guided Restart Scheme

Fig. 6. Geometries Found. The geometries found by standard optimization resemble each other more than those found by the learning-guided scheme. However, all Mach number distributions show high resemblance, seemingly as the left part of the suction side of all blades is similar. Fitness value of solutions in (a) 0.0729, 0.0881, 0.0715, and in (b) 0.0890, 0.0678, 0.0562.

respect to the domain-specific distance measure mentioned above. For future work a statistical comparison is to be performed with standard optimization, as well as the distance-based scheme. Also, a scheme that generates a random point and then starts moving toward it from a feasible start position is to be considered as a baseline approach to be outperformed, given the assumption that the learning-guided scheme is able to find the boundary of feasible space quicker.

Acknowledgments. The authors thank Giles Endicott for proof reading.

References

1. Cuccu, G., Gomez, F., Glasmachers, T.: Novelty-Based Restarts for Evolution Strategies. In: 2011 IEEE Congress on Evolutionary Computation, CEC 2011. pp. 158–163. IEEE Computer Society (2011)
2. Hansen, N.: The CMA Evolution Strategy: A Comparing Review. In: Towards a New Evolutionary Computation, Studies in Fuzziness and Soft Computing, vol. 192, pp. 75–102. Springer (2006)
3. Luciw, M., Graziano, V., Ring, M., Schmidhuber, J.: Artificial Curiosity with Planning for Autonomous Perceptual and Cognitive Development. In: 2011 IEEE International Conference on Development and Learning, ICDL 2011. IEEE Press (2011)
4. Reehuis, E., Olhofer, M., Emmerich, M., Sendhoff, B., Bäck, T.: Novelty and Interestingness Measures for Design-space Exploration. In: 15th Annual Conference on Genetic and Evolutionary Computation, GECCO'13. ACM Press (2013)
5. Reehuis, E., Olhofer, M., Sendhoff, B., Bäck, T.: Novelty-guided Restarts for Diverse Solutions in Optimization of Airfoils. In: EVOLVE 2013, A Bridge between Probability, Set-Oriented Numerics, and Evolutionary Computation (2013)