# Towards a Pittsburgh-style LCS for Learning Manufacturing Machinery Parametrizations

Michael Heider, David Pätzel, Jörg Hähner

University of Augsburg, Germany

ORGANIC COMPUTING

## Motivation

- **How to parametrize industrial machinery** is often learned by human operators through experimental exploration based on prior knowledge.
- **Transferring such acquired knowledge** to other operators can be challenging: Humans have a hard time to
  - exactly attribute parametrization choices to situations and
  - communicate knowledge per se.
- **Supervised learning** (e. g. on datapoints generated during exploration) can support this process.
- To be accepted by operators, the system needs to be **comprehensible**.
- **Learning Classifier Systems** comprehensibly create a global model from multiple interpretable less complex local models.

## SupRB-1 in brief

- **Accuracy-based Pittsburgh-style** Learning Classifier System
- Supervised learning on **continuous** multi-dimensional decision problems

$$q : \mathcal{X} \times \mathcal{A} \to \mathbb{R}, \quad \mathcal{X} \subset \mathbb{R}^n, \mathcal{A} \subset \mathbb{R}^m$$

- Global models
  - evolved by a **genetic algorithm**
  - consist of a set of localized simplistic paraboloid (i. e. linear regression) models each
- Fitness measure: **Bayesian Information Criterion**
  - prediction quality (accuracy) and
  - model complexity (number of parameters to fit)
- Predictions:
  - given $x \in \mathcal{X}$, location of the **mode of** $q_x : \mathcal{A} \to \mathbb{R}$
  - values of $q(x, a)$ (basic **function approximation**)

## Application

- **FDM-based additive manufacturing processes**
  - **Optimal parametrizations** depend on machine and material properties as well as environmental conditions.
  - Parametrizations might have to be **fine tuned frequently** depending on the required part quality. $\Rightarrow$ Traditional optimization (problem evaluated multiple times until optimum found) is usually too costly.
- Quality function of an **abstract additive manufacturing process**: AM-Gauss function
  - Consider abstract situations $(x_1, \dots, x_5)$ and parametrizations $(a_1, \dots, a_6)$ with realistic dimensionalities (derived from expert knowledge).
  - Assume that a single optimum with regard to **part quality** exists for each combination of dimensions with quality decreasing towards the extremes. $\Rightarrow$ Modelled as **two-dimensional Gaussian functions**.
  - **Sum** these Gaussian functions.
  - Allows to actually assess the choices of SupRB-1 (impossible with real data where the true optimum is unknown).

## AM-Gauss function

$$q(y) = \sum_{\substack{j\in 1,\dots,11,\\ k\in 1,\dots,11,\\ k\neq j}} \exp\left(-\left(\begin{pmatrix}y_j\\ y_k\end{pmatrix} - s_{j,k}\right)^T P_{j,k}\left(\begin{pmatrix}y_j\\ y_k\end{pmatrix} - s_{j,k}\right)\right)$$

- $y = (y_1, \dots, y_{11})^T = (x_1, \dots, x_5, a_1, \dots, a_6)^T$
- each $P_{j,k}$ is a positive semi-definite matrix in $\mathbb{R}^{2\times 2}$ with eigenvalues in $[0, 30]$ (ensures sensible scaling)
- $s_{j,k}$ is a vector in $[-1, 1]^2$ specifying the location of the summand's mode

## Experiments

- **Randomly generate an AM-Gauss function** (the required $P_{j,k}$'s and $s$'s generated from random seed 1).
- Sample a training set containing **2000 examples** (1000 for training local models, 1000 for fitness evaluations).
- **20 runs** (consecutive random seeds), 400 generations.
- Evaluate after each generation
  - overall **goodness-of-fit** (separate holdout set of size 1000)
  - **quality of predicted optimal parametrization** compared to quality of actual optimal parametrization (in 1000 random situations)
- Reported results are averages of the runs' populations' **elitists**.
- Baseline: **Two-layer fully connected artificial neural network** evaluated on identical data using 20 runs (same consecutive random seeds).

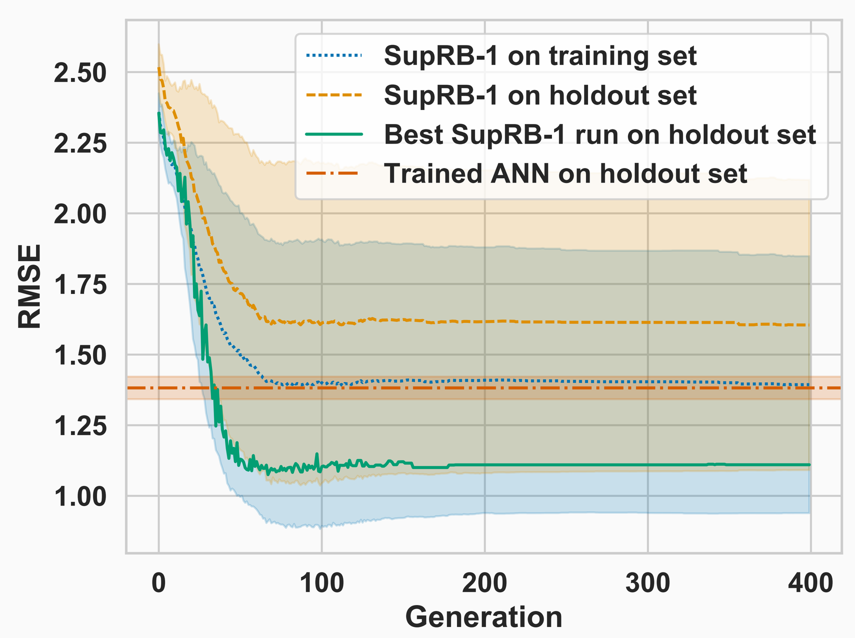


Fig. 1: Goodness-of-fit of **quality predictions** on training and holdout data with ANN baseline.

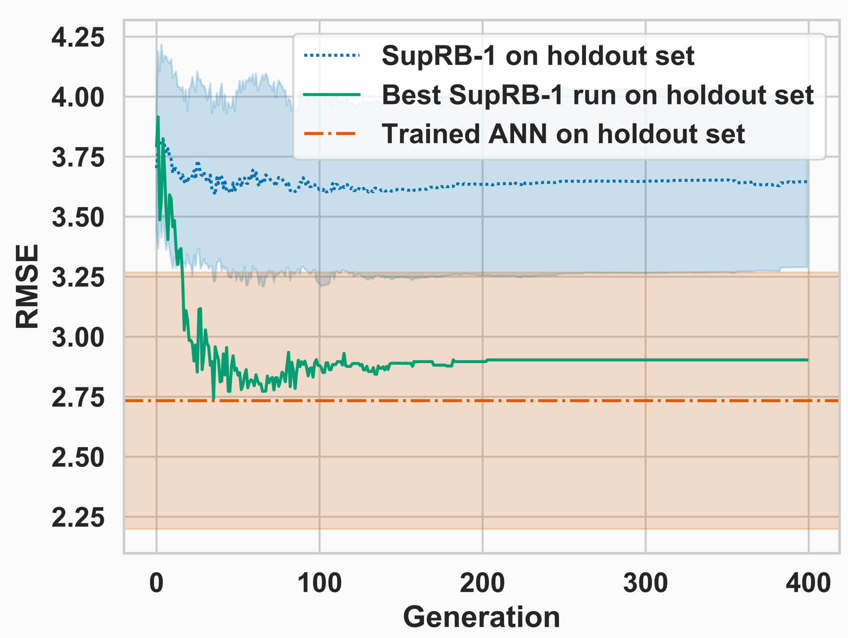


Fig. 2: Goodness-of-fit of **parametrization choices** on holdout data with ANN baseline.

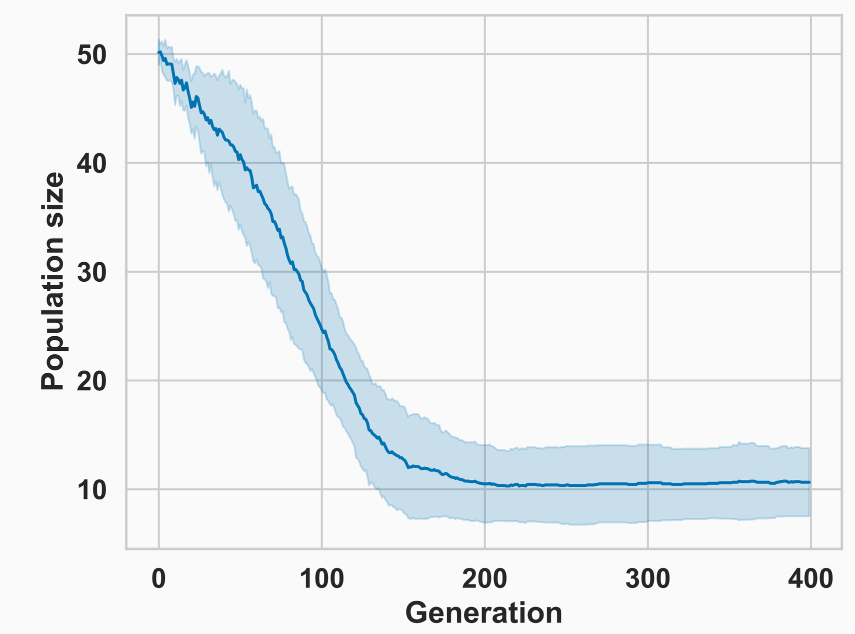


Fig. 3: Elitist's number of classifiers.

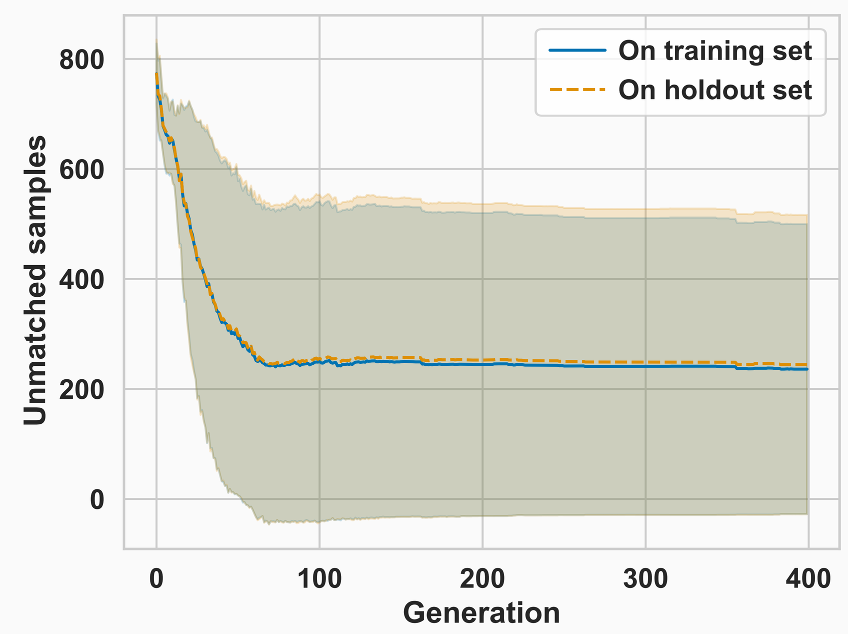


Fig. 4: Samples matched by default classifier only.

## Results

1. SupRB-1's **quality predictions**' RMSE on holdout data (Figure 1)
   - improves rapidly over the first 70 generations and then seems to converge at around 1.67 which falls short of the ANN baseline.
   - In contrast to the average, the best run not only converges slightly faster, but also achieves much better results at an error of about 1.1 surpassing the baseline. The most likely source of this is **premature convergence** to local optima.
2. Regarding the RMSE of the **parametrization choices** on holdout data (Figure 2),
   - the average results only improve slightly over time and **miss the baseline by far**.
   - However, the best run comes close to the baseline and is able to beat several ANNs performance-wise in the process.
3. For SupRB-1, **good results in prediction quality** often seem to correlate with **good results on choosing parametrizations**, whereas many ANNs that performed well on the former had vastly worse results on the latter.
4. Elitists of each run converge to 10 local models with a low standard deviation, the overall best performing individuals having 11 to 12 local models.
5. As most runs evolved solutions of similar size but with substantial differences in errors we **suspect that the models were ill placed at local optima**, supporting the suspected premature convergence issue.
6. We also shortly investigated the performance on 29 other AM-Gauss functions (seeds 2 through 30), albeit with only one run each, achieving comparable results.

## Improvements

We see several opportunities to increase the overall performance:

- Use **non-parabolic local models** that can better fit the problem function
- Prefer **good fit around higher predictions** of to better predict the location of optima
- Nurture a **more healthy and diverse population** of solutions