

Forming process optimisation for variable geometries by machine learning – Convergence analysis and assessment

ZIMMERLING Clemens^{1,a*} and KÄRGER Luise^{1,b}

¹Karlsruhe Institute of Technology, Institute of Vehicle System Technology, Karlsruhe, Germany

^a clemens.zimmerling@kit.edu, ^b luise.kaerger@kit.edu

Keywords: Forming Optimisation, Surrogate, Machine Learning, Variable Geometries

Abstract. For optimum operation, modern production systems require a careful adjustment of the employed manufacturing processes. Physics-based process simulations can effectively support this process optimisation; however, their considerable computation times are often a significant barrier. One option to reduce the computational load is surrogate-based optimisation (SBO). Although SBO generally helps improve convergence, it can turn out unwieldy when the optimisation task varies, e.g. due to frequent component adaptations for customisation. In order to solve such variable optimisation tasks, this work studies how recent advances in machine learning (ML) can enhance and extend current surrogate capabilities. More specifically, an ML-algorithm interacts with generic samples of component geometries in a forming simulation environment and learns to optimise a forming process for variable geometries. The considered example of this work is blank holder optimisation in textile forming. After training, the algorithm is able to give useful recommendations even for new, non-generic geometries. While the prior work considered initial recommendations only, this work studies the convergence behaviour upon component-specific algorithm refinement (optimisation) at the example of two geometries. The convergence of the new pre-trained ML-approach is compared to classical SBO and a genetic algorithm (GA). The results show that initial recommendations indeed converge to the process optimum and that the speed of convergence outperforms the GA and compares roughly to SBO. It is concluded that – once pretrained – the new ML-approach is more efficient on variable optimisation tasks than classical SBO.

Introduction

Most modern production systems are complex systems and require a careful optimisation during production ramp-up. In current practice, this often involves resource-intensive trial-error campaigns combined with expert-judgment based on experience from prior parts. However, shrinking lot sizes, ever shorter development cycles and increasing product diversity severely challenge such empirical approaches and call efficient process optimisation tasks.

Thus, it is found that recurring optimisation tasks for ever-changing geometries or materials, respectively, are a significant economical barrier [1]. This holds all the more when processing delicate materials, such as textiles used for continuous-fibre reinforced plastics (CoFRP). They are usually processed in elaborate, multi-step processes and most often comprise a forming process of a textile. The wide range of adjustable process parameters and the complex, non-linear material behaviour require place high demands on a suitable process configuration and pose a challenging development task.



To reduce the cost of an experimental process development, numerical simulations have gained attention over the last decades [2]. They allow for detailed analyses of complex processes and help concentrate costly experiments on the most promising variants. Also, their inherently digital nature allows a combination with optimisation algorithms. However, they usually involve significant computational efforts and especially repetitive simulations, e.g. iterative optimisation, quickly renders them impracticable in practice.

One option to reduce the numerical effort in such cases is surrogate-based optimisation (SBO) [3]. Surrogates are numerically efficient, data-driven approximations of expensive simulations based on input-output-observations. Once sufficiently trained, optimisation can be done on the surrogate in short time. Overall, SBO results in significant optimisation speed-ups. However, current SBO-approaches are mostly application-specific and fall short on reusability in new scenarios. Even subtle problem variations, e.g. geometry variations in manufacturing, instantly invalidate the surrogate and require resampling of data and reconstructing the surrogate. Thus, demand for generalised models arose.

At the same time, developments in Machine Learning (ML) have achieved remarkable results in complex tasks and may open up new avenues for advanced surrogates. The overarching concept is to sample process observations for a range of generic geometries and analyse it with ML-techniques [4-6]. Recurring patterns in the data may then guide a process optimisation of a new component. Owing to their reconfigurability and ease of evaluation, physics-based numerical process simulations are used for data sampling. Prior work has shown that such models can issue useful process recommendations for new components [4]. However, although the recommendations are useful, they are not strictly optimal but show some deviations to the true optimum. Thus, this work studies whether or not the initial ML-recommendations converge to the true optimum upon component-specific refinement.

Optimisation Methodology

Optimisation Approaches.

Formally, a forming simulation can be seen as a function $\varphi: P \xrightarrow{G} Q$ which maps variable process parameters $p \in P$ to a part quality descriptor $q \in Q$ for a given component geometry $g \in G$. Process optimisation then amounts to searching the optimal process parameters $p^* = \arg \min q(p)$ which yield the best quality¹. Finding this optimum is a profound task, though, and this work compares three different workflows as shown in Fig. 1.

One approach is to directly couple optimisation algorithms with the simulation φ , e.g. genetic algorithms as Fig. 1a) shows. For a given geometry g , they determine the process parameter optimum p^* by iterative evaluation, variation and combination of parameter combinations. However, φ is generally costly to evaluate and thus, iterating p^* until convergence quickly become prohibitively computation-intensive.

In order to increase efficiency, surrogate-based optimisation (SBO) constructs a numerically efficient substitute function $\mu_{\text{srg}}: P \rightarrow Q$, the “surrogate”. The surrogate seeks approximates φ from a set of pre-sampled observations and allows to do the optimisation in short time on μ_{srg} instead of φ , cf. Fig. 1b) [3]. The obtained candidate solution p_{srg}^* is in turn validated in a simulation run and this new observation is fed back to the database. The procedure then repeats until convergence.

Numerous case studies across disciplines have reported substantial optimisation speed-ups by SBO, see e.g. [7]. However, current SBO-strategies provide mostly application-specific, one-off models and struggle with unforeseen task variations. This impairs reusability in new scenarios:

¹ In manufacturing the part quality q is often expressed by the extent of defects like cracks or wrinkles which are sought to be minimised, not maximised.

Even a subtle problem variation, e.g. a change of material or geometry, instantly invalidates the surrogate and requires resampling of data and reconstructing the surrogate [4]. Thus, demand for generalised models has been identified early on [8].

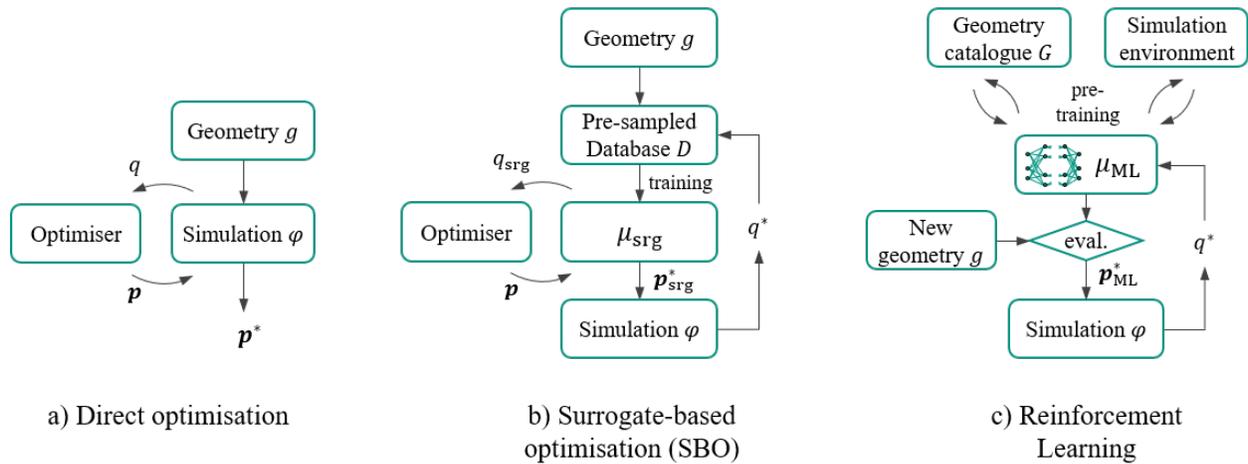


Fig. 1. Workflows of a) direct, b) surrogate-based optimisation (SBO) and c) Reinforcement Learning on multiple geometries.

As a remedy, prior work of the authors [4] and [9] suggests to give up on classical surrogate models μ_{srg} in favour of a more generalised function $\mu: G \rightarrow P$. While classical surrogate models are in most cases used for fix geometries or consider parametric geometries only, μ is meant to learn the underlying part-process-relations from a whole set of non-parametric generic training geometries. After training, it shall be able to give parameter recommendations even for new geometries which are not part of the training geometries. A detailed description of the implementation is given in the prior publications [4] and [9] and thus, only a brief glimpse is provided in this work.

Reinforcement Learning for process optimisation.

The overarching idea is that the ML model function interacts with a set of generic geometries (geometry catalogue G) in a simulation environment. The workflow is visualised in Fig. 2a).

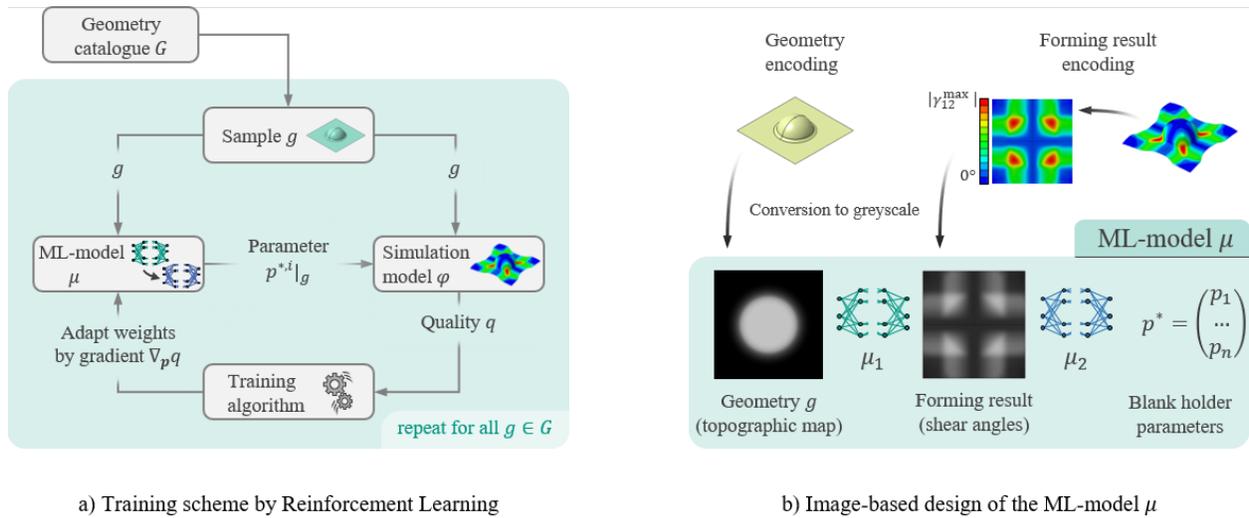


Fig. 2. a) Training scheme via Reinforcement Learning and b) image-based design of the ML-model μ [4,9].

In each training iteration i , a geometry g is drawn from the geometry catalogue. Then the ML-model μ analyses g and issues an estimation of optimal parameters $p^*|_g$. This parameter recommendation is then evaluated in a simulation run φ to determine the resulting part quality q . If $p^*|_g$ indeed improves the part quality, then the ML-model is encouraged to give similar recommendations for similar geometries in the future. Formally, this means that the gradient $\nabla_p q$ is determined and used to adapt μ 's parameter recommendations in direction of increasing quality q . The procedure iterates until the forming quality across all geometries in the catalogue ceases to improve.

In the context of material forming, image-based approaches for μ – as illustrated in Fig. 2b) – have been proposed [4-6]. Compared to conventional geometry parameters, e.g. length, width, fillet radii or angles, image-based geometry descriptions have proven robust and versatile when applicability to variable geometries is key. In this work and the prior works of the authors, μ consists of two nested functions, μ_1 and μ_2 , both of which are image-processing neural networks, so-called convolutional neural networks. The functions serve different purposes as Fig. 2b) visualises: μ_1 interprets the geometry and issues an estimation where and to what extent material strains are likely to occur. More specifically, the shear strain is evaluated as in-plane shear is the dominant deformation mechanism in engineering fabrics. Then, μ_2 interprets this strain-estimation and devises a parameter recommendation p^* .

Simulation Model and Optimisation Task

This work picks up on the models from the prior works [4,9] and is based on experimentally validated simulation approaches. It considers forming of cuboid geometries with the aid of pressure pads, cf. Fig. 3a). The cuboids can be varied in their dimensions w_1 and w_2 (width and length) as sub-image b) shows. For all geometries, the pads can be positioned freely around the perimeter of the geometries in order to control the draw-in of the fabric during forming.

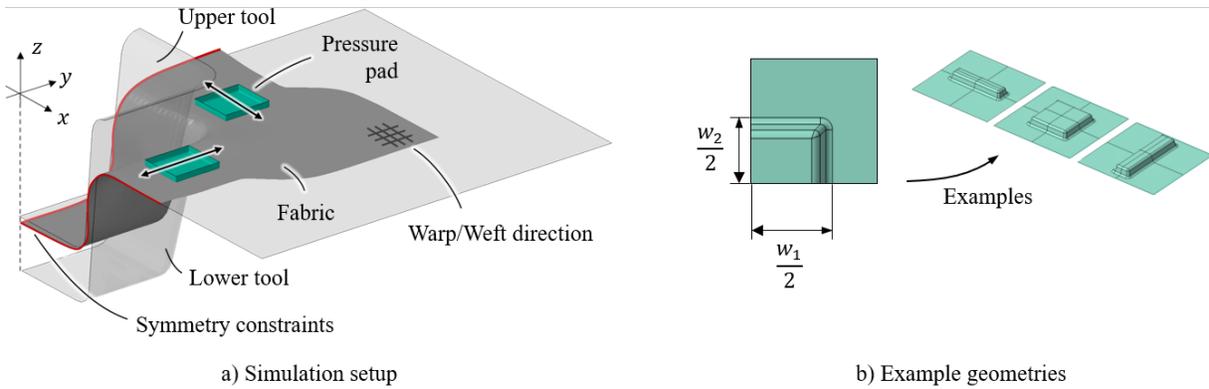


Fig. 3. a) Simulation setup for fabric forming with pressure pads; b) sample geometries in the geometry catalogue. Images are adapted from the prior work [4].

The cuboids are deliberately severe and show a strong tendency towards wrinkling. The overall quality goal during forming is to obtain a wrinkle-free forming result. As a direct measure of wrinkling, the fabric curvature κ is evaluated according to [11]. More specifically, the 99.5 % percentile κ_{qnt} of a Weibull fit to the fabric curvature distribution quantifies the quality, cf. Fig. 4a). The contour plot in Fig. 4b) shows κ_{qnt} for a single geometry as a function of the position of the pressure pads $p_{1,2}$ along with a top-view-visualisation of the pressure pads. The plot has been established by a fine grid-sampling of all possible pad positions for a single geometry. Note that these grid-samples are *by no means* involved in algorithm training but serve only for visualisation of κ_{qnt} (objective function) from an ‘omniscient’ perspective. The yellow marker illustrates how a specific pad position relates to the contour plot and vice-versa. Note that the plot is normalised so as to facilitate comparability with the following plots.

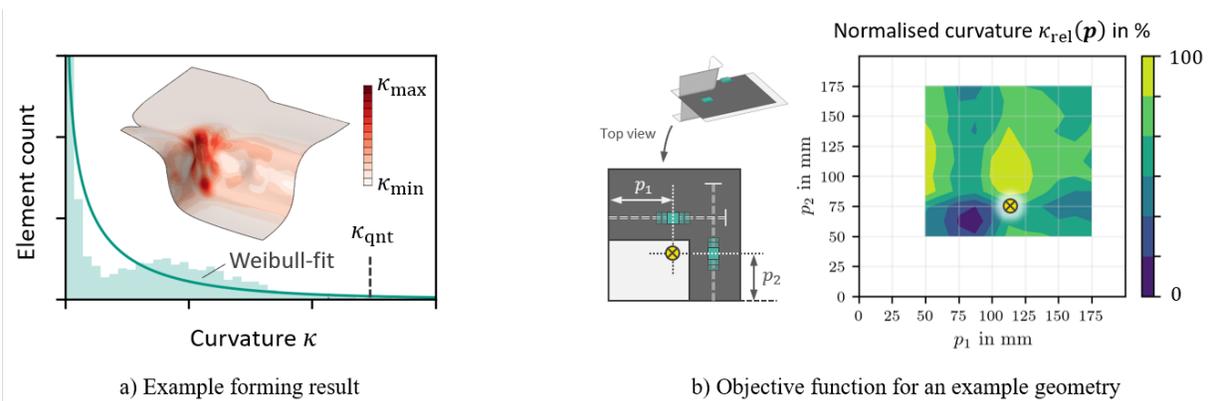


Fig. 4. a) Exemplary forming result with wrinkling, histogram of the fabric curvature plus Weibull fit; b) Contour plot of the obtained curvature (quality) depending on the pad positions $p_{1,2}$. Images adapted from prior work [4].

Clearly, different pad positions significantly alter the obtained curvature and a distinct optimum, i.e. minimal curvature, can be observed. All geometries show such an optimum; however, its location varies and μ shall estimate its position for each geometry in the catalogue.

Results and Discussion

In the prior work [9], μ is trained on a set of box-geometries shown in Fig. 3b). The results show that it gives useful estimations not just for new box-geometries but also – to a certain degree – for ‘non-box’ geometries. See [9] for further details on the results. Two of these test geometries shall be further analysed in this work. They are shown in Fig. 5 on the left: A rotational symmetric shell with conical tips (g_1) and a double-dome geometry (g_2).

The contour plots in Fig. 5 show the forming quality (curvature) for all pad positions obtained by a grid-sampling approach. Note that they are not a subset of the box-geometries but, as they are doubly-symmetric and near-convex, they will still show a similar forming behaviour: Both contour plots feature a minimum similar to the box-geometry in Fig. 4b). Some qualitative differences can be observed, though: For g_1 , the optimum is not a sharp point but a plateau-like region in the bottom-left of the plot. In contrast, the optimum for g_2 is comparably distinct and surrounded by two, connected maxima in a funnel-like shape while the majority of the plot is of approximately constant, mediocre quality.

Additionally, the plots feature three types of markers, namely two large and 30 small markers. The blue large marker visualises the best quality obtained during grid-sampling which is assumed to be the ‘true optimum’ p^* . The yellow large marker denotes μ ’s initial process recommendation $p_{ML}^{*,0}$ immediately after training on box-geometries. If both the blue and the yellow marker coincided, that would imply that μ had made a perfect estimation of the process optimum. However, such a perfect estimation is highly unlikely since μ makes inference for a new geometry on the basis of generic box-geometries. Thus, the two markers are close to each other but still lie some distance apart. More specifically, an overestimation of $\approx 18\%$ (g_1) $\approx 13\%$ (g_2) relative to the parameter range $25\text{ mm} \leq p_{1,2} \leq 200\text{ mm}$ is observed.

However, the question arises, whether the yellow marker approaches the blue marker (true optimum) upon component-specific continuation of training. That is, having been pretrained on generic boxes, μ interacts now only with geometry g_1 or g_2 , respectively. Thereby μ can iteratively refine on these specific geometries. The evolution of μ ’s recommendation $p_{ML}^{*,i}$, is visualised by the smaller markers whose hue denotes their order of appearance, i.e. refinement iteration $i_{\text{rfn}} = 1$ (black) to $i_{\text{rfn}} = 30$ (white).

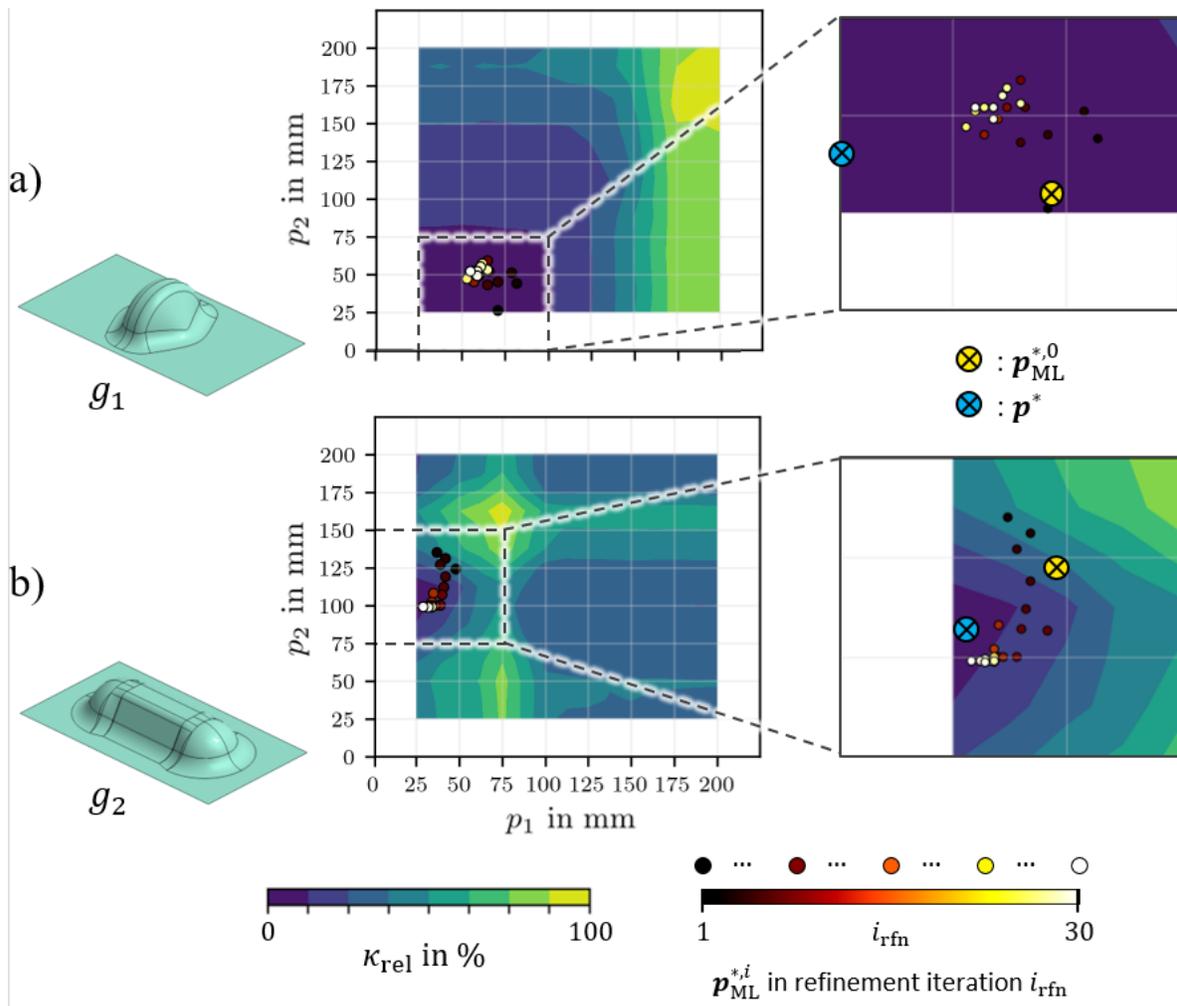


Fig. 5. Contour plots with μ 's process recommendations during component-specific continuation of training for a) geometry g_1 (rotational symmetric shell) and b) g_2 (double-dome).

However, the question arises, whether the yellow marker approaches the blue marker (true optimum) upon component-specific continuation of training. That is, after the training on generic box-geometries, μ interacts now only with geometry g_1 or g_2 , respectively. Thereby μ is meant to iteratively refine its recommendations on these specific geometries. The evolution of μ 's recommendation $p_{ML}^{*,i}$, is visualised by the smaller markers, while their hue denotes their order of appearance, i.e. refinement iteration $i_{rfn} = 1$ (black) to $i_{rfn} = 30$ (white).

The plots reveal a disparate refinement behaviour: For geometry g_1 in subplot a), the refined markers appear somewhat incoherently scattered around the initial recommendation. At most, a light tendency to the top left can be observed. In contrast, for g_2 (double-dome) in subplot b), the markers do accumulate and show a coherent evolution: At first, the markers move to the top left. But, since this deteriorates the results, they reverse and gradually move downwards before concentrating near the minimum.

These observations can be explained when examining the objective functions in the contour plots: For g_1 in subplot a), the plateau-characteristic makes it difficult to identify a gradient for an improved pad position. Accordingly, the recommendations (markers) appear erratic. In contrast, a distinct optimum is observable for g_2 and – after a few iterations for ‘orientation’ – the markers move in direction of improved part quality. For some reason however, they only approach but do

not reach the observed optimum (blue marker). This may be due to numerical noise when evaluating the forming simulation.

Ultimately, Fig. 6 illustrates the optimisation progress for both geometries by means of the objective function. The diagrams support the above line of thought: Sub-image a) shows that the forming quality κ_{rel} stays practically constant for g_1 through all refinement iterations, while it improves (declines) for g_2 after correction of an initial peak. Overall, the results indicate that – like a regular surrogate – μ can indeed refine its initial recommendations with new evidence.

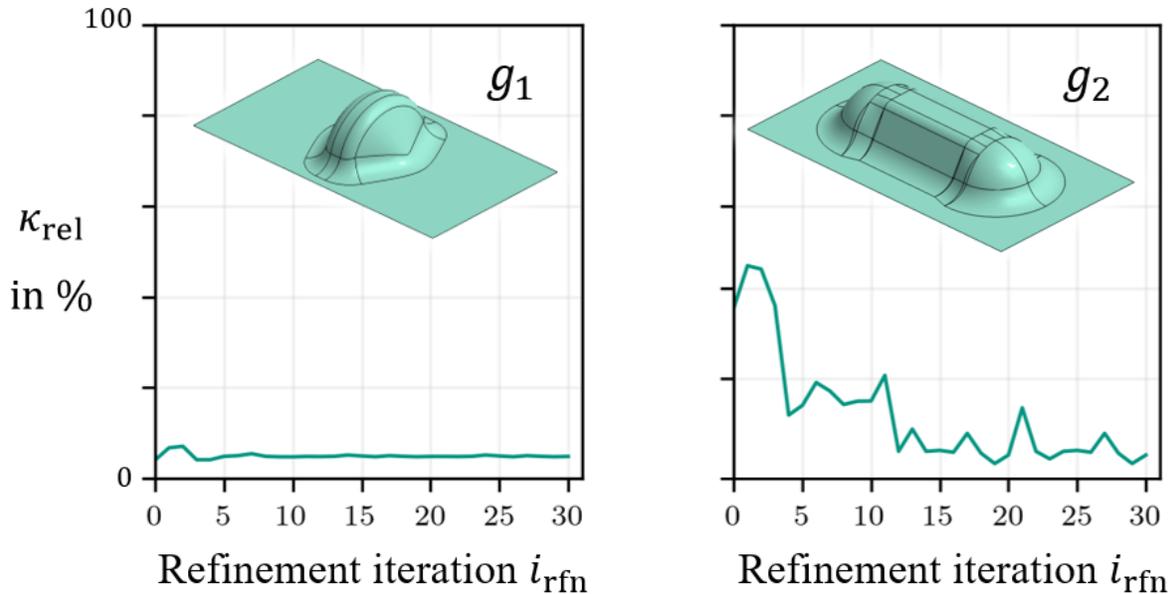


Fig. 6. Evolution of the optimisation objective κ_{rel} during component-specific refinement of μ on g_1 and g_2 .

Summary

Having seen that the RL-based optimisation method indeed shows optimisation behaviour, the three optimisation approaches from Fig. 1 – direct optimisation with a genetic algorithm (GA), SBO and RL – can be compared to each other. Each method is applied to the geometries g_1 and g_2 for inspection of the optimisation behaviour. The GA stems from the publicly available optimisation toolbox ‘Dakota’ [12] and the SBO reference method is described in detail in [13].

Note that the optimisation approaches employ different iterative schemes: The GA utilises a generation-based principle with each generation comprising $n_{idv} = 20$ individuals. It is set to terminate after $n_g = 15$ generations, i.e. after a total of $n_g \cdot n_{idv} = 15 \cdot 20 = 300$ simulations. This limit prevents excessive computation times and was determined empirically in a prior analysis on a fast – but much simpler – substitute model, cf. [4]. The limit is set such that at least one individual becomes (near-)optimal; yet it does not necessarily imply full convergence of the whole population around the optimum. For prevention of a potential quality-loss, the best found solution from the previous generation is (unaltered) carried over to the next generation (so-called ‘elitist’ selection). Both SBO and RL employ $n_i = 30$ refinement iterations (simulations). However, SBO requires a component-specific sampling – in this work Latin Hypercube Sampling – which accounts for additional $n_s = 20$ a-priori simulations before optimisation start. Also note, that RL needs no specific sampling due to its pretraining on box-geometries. Thus, it can directly start refining its recommendations for optimisation.

Geometry g_1 .

For a direct comparison, Fig. 7 plots the progress of each optimiser on the objective κ_{rel} in one diagram along with a detail view for closer inspection (grey shade). Besides the sheer sequence of κ_{rel} (thin dashed line), the plots also show their lower envelope (bold solid line). Essentially, it gives for each iteration the ‘so-far-best’ solution and visualises, how fast each optimiser reduces the objective function. Overall, all graphs show a successful optimisation behaviour as they all approach the minimum ($\kappa_{rel} = 0\%$).

However, although all approaches improve the solution over the iterations, the gain during additional optimisation iterations is limited as the first solutions are already near-optimal ($\kappa_{rel} \approx 1..5\%$). Thus, they offer little room for improvement with further iterations. This phenomenon can be explained by inspection of the contour plot, cf. Fig.5a): Since the optimum is not a sharp point but a comparably large plateau, the optimisers find it right at the beginning and thus have no room for improvement during iterations.

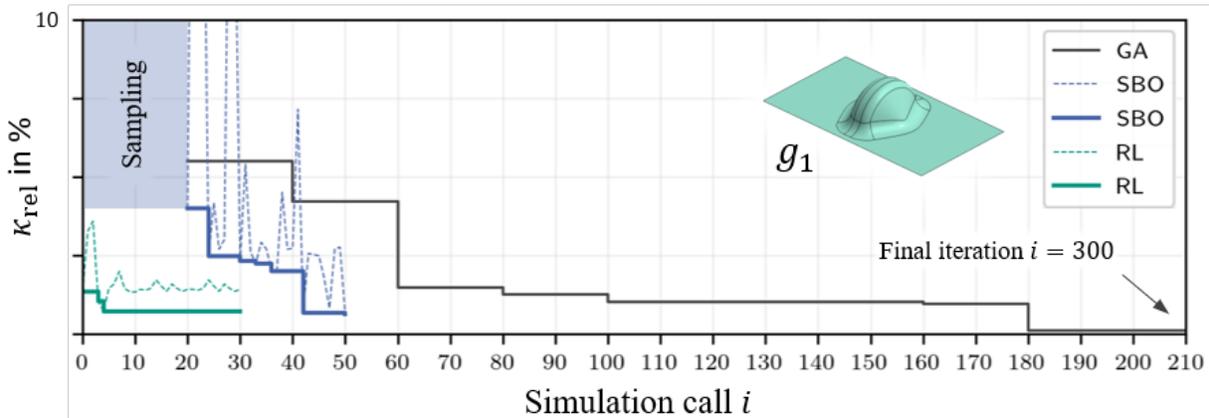


Fig. 7. Juxtaposition of κ_{rel} during optimisation on g_1 for the GA, SBO and the RL-approach. The plots of GA and SBO are offset to account for the simulations until a first iteration occurs. Note that κ_{rel} is near-optimal directly from the beginning ($\kappa_{rel} \approx 1..5\%$). In accord, the plot is magnified with a y-axis focus on the lower 10%.

Despite the limited gain, the GA finds the overall-best solution ($\kappa_{rel} = 0.2\%$), while both RL and SBO ($\kappa_{rel} = 0.82\%$) remain slightly inferior ($\kappa_{rel} = 0.78\%$). Note however, that RL and SBO require far fewer iterations to reach their final value, i.e. they converge faster. Also note that all algorithms involve elements of randomness. Thus, re-running them will probably lead to slightly different graphs and optimisation results.

Geometry g_2 .

In a similar manner, Fig. 8 shows graphs for each optimisation approach on g_2 (double-dome). The overall behaviour of the algorithms stays the same as before: A successful optimisation can be observed. However, this time a notable effect of additional iterations is observed. Due to the majority of mediocre process responses (plateaus), all methods start at $\kappa_{rel} \approx 40..50\%$ in their first iteration, but then the progresses differ.

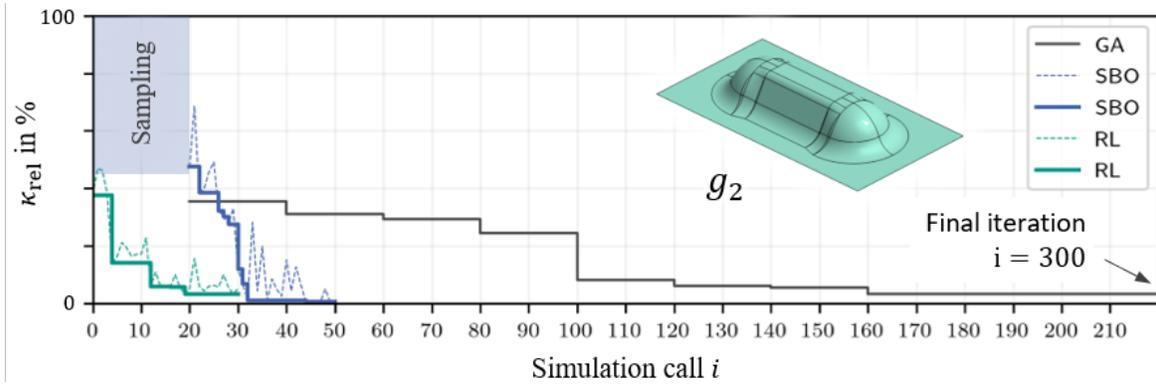


Fig. 8. Juxtaposition of κ_{rel} during optimisation on g_2 (double-dome) for the GA, SBO and the RL-approach. The plots of GA and SBO are offset to account for the number of simulations until a first iteration.

Overall, the plot resembles the previous: The graphs decline in a monotonous manner to the minimum ($\kappa_{rel} = 0\%$) and RL and SBO outperform the GA. However, this time all graphs show a substantial improvement of the objective function κ_{rel} during optimisation. All optimisers find a solution within a 5 %-range around the optimum, while SBO finds the overall best-solution with $\kappa_{rel} \approx 2\%$. The GA reaches the 5 %-range after 160 simulations and RL after 18 iterations.

Speed of convergence. Ultimately, the speed of convergence shall be briefly assessed. To this end, the convergence metric

$$C = \int_{i=0}^{i_{max}} \kappa_{rel}(i) di = \sum_{i=0}^{i_{max}} \kappa_{rel}(i) \Delta i = \sum_{i=0}^{i_{max}} \kappa_{rel}(i) \quad (1)$$

is used. Essentially, it is the area under the (solid) graphs until each optimiser has reached its final value in iteration i_{max} . Due to the integer-abscissa, $di = \Delta i = 1$ holds which simplifies the integral to a sum. Overall, a smaller value of C implies faster convergence. If an approach finds a good solution early on, the following summands and thus C becomes smaller. This reflects the desired behaviour during optimisation of expensive functions, i.e. rapid minimisation with only few function evaluations. Table 1 below summarises the convergence metrics C and numbers of simulations i_{max} each optimisation approach required until its final value.

Table 1: Convergence comparison of GA, SBO and RL according to the convergence metric C and the required number of iterations i_{max} .

Geometry	GA		SBO		RL	
	C	i_{max}	C	i_{max}	C	i_{max}
g_1	4.39	180	0.79	42	0.31	4
g_2	37.13	160	13.50	32	4.02	19
sum	-	340	-	74	-	23

For both geometries, g_1 and g_2 , SBO and the new RL-based method outperforms the GA by a large margin: While the GA requires 160 or 180 simulations, respectively, SBO and RL require less than 50 simulations to reach an approximately equal forming result. Equally large differences can be observed for the C . When comparing SBO and RL directly, RL appears more efficient than SBO. This is mainly because the RL-based approach does not require a component-specific sampling due to its pretraining on generic (box-)geometries. It must be noted however, that the computational effort for this pretraining may be considerable.

The advantage of the pretrained RL-model becomes even more evident when optimising multiple geometries: Suppose, both geometries g_1 and g_2 need to be manufactured and their processes must be optimised. Optimising both geometries with SBO requires in total 74 simulations of which $2 \cdot n_s = 40$ simulations are required just for sampling. In contrast, the RL-model can do without component-specific sampling due to its pretraining on boxes and takes only 23 simulations in total. Although the exact numbers will certainly vary in different applications, the non-necessity of the a-priori sampling substantially cuts the computational effort.

Summary

This work studies the convergence behaviour of a previously proposed RL-based approach for process optimisation of variable geometries. The approach centres around an ML-model which interacts iteratively with a catalogue of generic geometries in a simulation environment. It thereby learns which geometry requires which optimal process configuration. The approach is studied at an example from prior work, namely optimisation of pressure pad positions during textile forming. Prior work has shown that – after the (pre-)training on generic geometries – the RL-approach indeed gives useful, near-optimal process recommendations [4,9]. This work picks up on these results and investigates, whether the recommendations converge to the true optimum upon component-specific refinement and if so, how fast.

The results are twofold: First, the results on two demonstrator geometries hint that the proposed RL-model indeed converges to the true process optimum. Second, a comparison to two conventional optimisation approaches – a genetic algorithm (GA) and a ‘classical’ surrogate-based approach (SBO) – shows that RL and SBO outperform the GA and show a similar convergence during refinement iterations. However, SBO needs a component-specific a-priori sampling while the RL-based method is pre-trained on generic geometries and can directly start refining its recommendations. Thus – once trained – it speeds up the optimisation process similar to a classical surrogate but beneficially cuts the component-specific sampling effort. However, this comes at the cost of substantial numerical efforts for algorithm pretraining.

The results of this work and the prior work show that it is indeed possible to extract usable part-process-relations from generic samples and use them during process optimisation for new components. While this work outlines their principal potential, the developed techniques yet need to be advanced to more complex and application-centred use-cases from industrial practice. Regarding textile-forming, this means the integration of more complex process scenarios, e.g. more complex geometries, additional process parameters and variable material properties. In the wider sense, the methods can also be tested on other processes inside and outside material forming.

Acknowledgment

This work is part of the IGF research project *OptiFeed* (21949 N) of the research association *Forschungskuratorium Textil e.V.*, Reinhardtstraße 14-16, 10117 Berlin, funded via the *AiF* within the program for supporting the Industrial Collective Research (IGF) from funds of the Federal Ministry of Economic Affairs and Climate Protection (BMWK). It is also part of the Young Investigator Group (YIG) *Green Mobility*, generously funded by *Vector Stiftung*.

References

- [1] N. Shamsaei, A. Yadollahi, L. Bian, S.M. Thompson, An overview of direct laser deposition for additive manufacturing - part 2, *Additive Manuf.* 8 (2015) 12-35. <https://doi.org/10.1016/j.addma.2015.07.002>
- [2] L. Kärger, A. Bernath, F. Fritz, S. Galkin, D. Magagnato, A. Oeckerath, A. Schön, F. Henning, Development and validation of a cae chain for unidirectional fibre reinforced composite components, *Compos. Struct.* 132 (2015) 350-358. <https://doi.org/10.1016/j.compstruct.2015.05.047>

- [3] S. Koziel, L. Leifsson, *Surrogate-Based Modeling and Optimization: Applications in Engineering*. Springer Science+Business Media, New York, 2013. <https://doi.org/10.1007/978-1-4614-7551-4>
- [4] C. Zimmerling, C. Poppe, O. Stein, L. Kärger, Optimisation of manufacturing process parameters for variable component geometries using reinforcement learning, *Mat. Des.* 214 (2022) 110423. <https://doi.org/10.1016/j.matdes.2022.110423>
- [5] H.R. Attar, H. Zhou, A. Foster, N. Li, Rapid feasibility assessment of components to be formed through hot stamping: A deep learning approach, *J. Manuf. Process.* 68 (2021) 1650-1671. <https://doi.org/10.1016/j.jmapro.2021.06.011>
- [6] H. Zhou, Q. Xu, Z. Nie, N. Li, A study on using image-based machine learning methods to develop surrogate models of stamp forming simulations, *J. Manuf. Sci. Eng.* 144 (2022) 021012. <https://doi.org/10.1115/1.4051604>
- [7] R.T. Haftka, D. Villanueva, A. Chaudhuri, Parallel surrogate-assisted global optimization with expensive functions – a survey, *Struct. Multidisc. Optimization* 54 (2016) 3-13. <https://doi.org/10.1007/s00158-016-1432-3>
- [8] H. El Kadi, Modeling the mechanical behavior of fiber-reinforced polymeric composite materials using artificial neural networks - a review, *Compos. Struct.* 73 (2006) 1-23. <https://doi.org/10.1016/j.compstruct.2005.01.020>
- [9] C. Zimmerling, C. Poppe, L. Kärger, Estimating optimum process parameters in textile draping of variable part geometries - A reinforcement learning approach, *Procedia manuf.* 47 (2020) 847-854. <https://doi.org/10.1016/j.promfg.2020.04.263>
- [10] C. Zimmerling, D. Trippe, B. Fengler, L. Kärger, An approach for rapid prediction of textile draping results for variable composite component geometries using deep neural networks, *AIP Conference Proceedings* 2113, ESAFORM 2019 in Vitoria-Gasteiz/Spain (2019), Art. 020007. <https://doi.org/10.1063/1.5112512>
- [11] S. Haanappel, *Forming of UD fibre reinforced thermoplastics: A critical evaluation of intra-ply shear*. PhD-Thesis, Universiteit Twente, 2013. <https://doi.org/10.3990/1.9789036535014>
- [12] Sandia National Laboratories. DAKOTA user's manual - version 6.15, www.dakota.sandia.gov/sites/default/files/docs/6.15/Users-6.15.0.pdf (2021), last accessed: 31.08.2022.
- [13] C. Zimmerling, P. Schindler, J. Seuffert, L. Kärger: Deep neural networks as surrogate models for time-efficient manufacturing process optimisation. *PoPuPS of ULiège Library, ESAFORM 2021 in Liège/Belgium* (2021). <https://doi.org/10.25518/esaform21.3882>