Materials & Design 214 (2022) 110423

Contents lists available at ScienceDirect

Materials & Design

journal homepage: www.elsevier.com/locate/matdes

Optimisation of manufacturing process parameters for variable component geometries using reinforcement learning

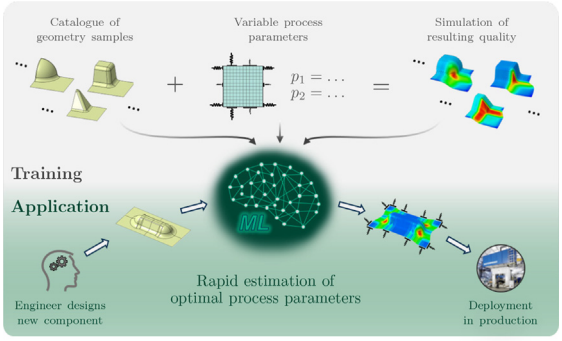
Clemens Zimmerling [a,*], Christian Poppe [a,c], Oliver Stein [b], Luise Kärger [a]

[a] Karlsruhe Institute of Technology (KIT), Institute of Vehicle System Technology, Karlsruhe, Germany
[b] Karlsruhe Institute of Technology (KIT), Institute of Operations Research, Karlsruhe, Germany
[c] Simutence GmbH, Karlsruhe, Germany

HIGHLIGHTS

- Machine Learning assisted optimisation of manufacturing parameters for variable component geometries.
- Reinforcement Learning (RL) algorithm interacts with a process simulation environment.
- Optimisation capability is demonstrated for textile draping in composite manufacturing.
- After training, the ML-algorithm gives recommendations for new geometries.
- Hyperparameter studies reveal best-practice guidelines for algorithm training.

GRAPHICAL ABSTRACT



ARTICLE INFO

Article history:
Received 27 September 2021
Revised 6 January 2022
Accepted 21 January 2022
Available online 29 January 2022

Keywords:
Manufacturing optimisation
Composite forming
Machine learning
Artificial intelligence
Design for manufacture

ABSTRACT

Tailoring manufacturing processes to optimum part quality often requires numerous resource-intensive trial experiments in practice. Physics-based process simulations in combination with general-purpose optimisation algorithms allow for an a priori process optimisation and help concentrate costly trials on the most promising variants. However, considerable computation times are a significant barrier, especially for iterative optimisation. Surrogate-based optimisation often helps reduce the computational effort but surrogate models are typically case-specific and cannot adapt to different manufacturing situations. Consequently, even minor problem variations e.g. geometry adaptions invalidate the surrogate and require resampling of data and retraining of the surrogate. Reinforcement Learning aims at inferring optimal actions in variable situations. In this work, it is used to train a neural network to estimate optimal process parameters ("actions") for variable component geometries ("situations"). The use case is fabric forming in which pressure pads are positioned to optimise the material intake. After training, the network is found to give meaningful parameter estimations even for new geometries not considered during training. Thus, it extracts reusable information from generic process samples and successfully applies it to new, non-generic components. Since data is reused rather than resampled, the approach is deemed a promising option for lean part and process development.

* Corresponding author.
E-mail address: clemens.zimmerling@kit.edu (C. Zimmerling).

https://doi.org/10.1016/j.matdes.2022.110423
0264-1275/

1. Introduction and related work

Industrial manufacturing processes generally require careful parametrisation for optimum operation in terms of part quality, throughput or efficiency. In current practice, identification of such optimum parameters employs resource-intensive, experience-guided trial-error campaigns and often entails significant rework for error correction. This holds all the more when processing complex materials, such as textiles used for continuous-fibre reinforced plastics (CoFRP). Additionally, growing demand for adaptive and customisable production principles (“mass customisation”) fostered development of flexible manufacturing technologies, e.g. additive manufacturing [1], modular production cells [2], adaptive moulds [3] or mould-free technologies [4].

Besides technical readiness, flexible technologies also require efficient approaches for production line ramp-up and process optimisation. For instance, Shamsaei et al. [5] study additive manufacturing and find that recurring optimisation tasks for ever-changing geometries or materials, respectively, are a significant economical barrier. They call for a comprehensive framework to “leverage information from prior similar studies and . . .systematically characterize the relation between process parameters and part features so that the . . .process can be optimized in a more efficient manner”. Recently, similar suggestions have emerged in other domains as well, including but not limited to material forming [6] as addressed in this work.

In textile forming, as in any other process, component geometry, material and process must match in mutual regard [7] and different approaches to ensure and optimise formability have emerged. Among others they comprise qualitative design guidelines [8], analytical formulae for simple cases [9], geometry analysis tools [10] and inverse approaches which directly yield sets of formable shapes [11]. However, most established are simulations, which broadly fall into two categories [12]: kinematic and physics-based simulations. Kinematic methods (cf. [13,14]) are computationally inexpensive but involve severe simplifications. Their counterpart, physics-based simulations – often Finite Element (FE) models – enable a rigorous virtual analysis of numerous process dynamics and material peculiarities [15].

Their inherently digital nature beneficially allows a direct combination with optimisation algorithms, which is often subsumed under “virtual process optimisation”. While such approaches principally enable an automatic identification of process optima [16], the computational effort for iterative optimisation often renders them impracticable in practice.

One option to reduce the numerical effort during optimisation is surrogate-based optimisation (SBO) [17]. 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. Regarding material forming optimisation, metalwork applications dominate literature. See [18] for a review. For composite materials, most SBO-applications concentrate on structural optimisation, e.g. stiffness or buckling characteristics [19,20]. To the author’s knowledge, [21–24] are the first reports on surrogates for virtual process analysis and optimisation in textile forming.

Although literature reports significant speed-ups through SBO, current approaches are mostly application-specific and fall short on reusability in new scenarios. Even subtle problem variations, e.g. geometry variation in manufacturing, instantly invalidate the surrogate and require resampling of data and reconstructing the surrogate. Thus, demand for generalised models has been identified early on, e.g. in [25], and continues to be prevalent.

With the advent of advanced machine learning (ML) techniques, first approaches towards generalisation and automation have been presented. For instance, [26] addresses variable materials in sheet metal forming and [27] presents an ML-approach to select suitable forming technologies for a given component geometry. Variable process parameters are not considered, though.A surrogate-approach for variable process parameters for a complex deep drawn part is discussed in[28]. Unlike classical surrogate techniques, the presented approach allows for a full-field assessment of the strain distribution instead of a single performance metric. The considered geometry remains constant, though. In [29] an ML-approach for an automatic work-path generation for variable geometries in tool-free “metal driving” is presented. Although the results are convincing, the point-wise nature of the process seems to restrict applicability to common tool-based processes such as stamp forming. Thus, outside tailored solutions for niche applications, the development of a surrogate framework for rapid manufacturability assessment and process optimisation remains an open field of research.

1.1. Scope of this work

This work proposes the combination of physics-based process simulations with ML-techniques to extend classical SBO towards variable instead of fixed manufacturing scenarios. The overall aim is to train an ML-model to estimate optimal manufacturing parameters for a new component geometry. To this end, a reinforcement learning (RL) approach from the prior work [30]is investigated. More specifically, an ML-model iteratively interacts with generic sample geometries in a simulation environment. Thereby it gradually learns which geometries require which parameter settings.In addition to [30], this work employs validated and physics-based FE-simulations instead of simplified models, investigates the effect of different hyperparameters, analyses in detail the training behaviour on component scale and evaluates the transferability to new, non-generic components. Fabric forming with variable blank holders (pressure pads) serves as a use case, although the approach is in principle technology-agnostic.

The paper is organised as follows: Section 2 describes the RL approach and the use case including a brief introduction of the FE simulation model. Section 3 at first presents results of an algorithm hyperparameter study and then discusses the algorithm’s performance in two test scenarios. The work is summarised in Section 4 and concludes with an outlook on possible next steps.

2. Methodology

2.1. Optimisation concept and general workflow

Classical surrogate-based optimisation. Formally, an FE process simulation can be seen as a function $\varphi^{\text{fem}} : P \mapsto Q$ which maps process parameters P to a part quality metric Q. The goal during process optimisation is to solve

$$\boldsymbol{p}^* = \arg\max_{\boldsymbol{p}} \varphi^{\text{fem}}, \tag{1}$$

i.e. find a parameter combination $\boldsymbol{p}^* \in P$ which maximises the part quality $q \in Q$. In many cases P is high-dimensional and φ^{fem} nonlinear and non-convex. Under such conditions gradient-free optimisation algorithms, e.g. evolutionary algorithms, can give good results. Yet, they usually require many computation-intensive evaluations of φ^{fem}, which makes a direct application impracticable. Surrogate-based optimisation (SBO) seeks to reduce the computation time by constructing a numerically efficient approximation – a surrogate – and optimise on it instead of φ^{fem}. Usually termed model “training”, the approximation is typically obtained by fitting a preselected model function to a dataset of process observations.

Different classes with countless sub-variations have emerged for the model function ranging from simple polynomial regression to advanced approaches such as support vector machines [17].

Proposed optimisation workflow with reinforcement learning. Although SBO generally increases optimisation efficiency, current surrogates are inevitably task-specific. Consequently, any task-variations, e.g. a geometry change from $g_1 \in G$ to $g_2 \in G$, instantly invalidates the surrogate, which in turn impedes efficient exploration of different part and process designs. As a remedy, this work suggests the construction of a more generalised process-surrogate

$$\Pi : G \mapsto P, \tag{2}$$

which accepts a geometry $g \in G$ as input and directly estimates optimal process parameters $\boldsymbol{p}^*$.

Like a regular surrogate, Π is constructed using process samples. While any surrogate technique could model Π, this work deliberately employs deep neural networks for the following reasons: First, they are universal approximators [31]. That is, given sufficient data, they can reproduce any continuous function regardless of its complexity. Second, due to their remarkable modelling capacity, a rich corpus of efficient training algorithms is at disposal and, third, comprehensive open-source libraries allow for convenient and efficient implementation [32]. Eventually, specialised subtypes of neural networks exist, which take advantage of certain data structures, e.g. images.

Fig. 1 visualises the approach. In the following, circumflexes distinguish estimated from ground truth values, asterisks denote (estimated) optima and vertical bars imply "evaluated for". For instance, $\hat{\boldsymbol{p}}^*|_g$ denotes the estimation of optimal parameters for geometry g. During training, Π is iteratively presented different geometry samples g from a predefined geometry set G and estimates the according optimal process parameters $\hat{\boldsymbol{p}}^*|_g$. A process simulation $\varphi^{\text{fem}}(\hat{\boldsymbol{p}}^*)|_g$ then evaluates the obtained part quality q. The ML-literature usually refers to q as "reward". Overall, Π aims to maximise q to make optimal parameter recommendations per geometry more likely over time. The ultimate goal after training is to give process parameter recommendations even for "new" geometries outside the training data. In ML, the overall approach is known as "Reinforcement Learning" (RL).

Traditionally, RL is an approach to solve an incompletely-known Markov decision process (MDP) and originates from control theory [33]. More recently, it has also drawn attention in neighbouring disciplines such as engineering optimisation, see e.g. [34] for chemical reactions and [35] for material processing, in which RL-algorithms successfully substitute classical optimisation algorithms for a fixed optimisation task. See also [36] for a review on RL in production systems. In this work, the overall aim of RL is to extract manufacturing similarities from generic parts and apply them for efficient process optimisation of a multitude of new, non-generic parts. That is, unlike prior work, variable instead of fixed optimisation tasks are addressed.

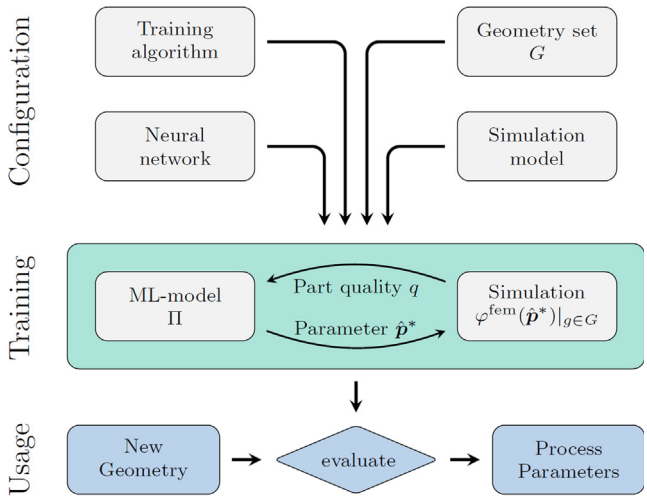


Fig. 1. Training and application scheme of the envisaged ML-model Π. During training Π learns to relate individual geometries $g \in G$ to their optimal process parameters. After training, it can give meaningful process recommendations even for "new" geometries.

2.2. Use case "fabric forming"

2.2.1. Process overview

Continuous-fibre reinforced plastics (CoFRP) are increasingly applied in load-bearing structures, yet their complex and non-linear material behaviour requires diligent process optimisation for defect-free manufacture. Thus, CoFRP-manufacture is deemed a relevant use case, although the envisaged RL-assisted optimisation approach is in principle not restricted to a certain process technology.

CoFRP-processing generally comprises multiple steps and often involves a stamp-forming step of a textile. This work focusses on stamp forming of a dry woven fabric ("preforming"), an important process step during Resin Transfer Moulding (RTM) [37]. As woven fabrics generally show low shear resistance, (trellis-) shear is the predominant in-plane-deformation mode, usually quantified by the shear angle γ_{12} (cf. Fig. 2).

Due to progressive yarn compaction and according in-plane compressive stress accumulation, woven fabrics show a material-dependent shear limit, the "locking angle" $\gamma_{12}^{\text{lock}}$ [39]. Shearing beyond $\gamma_{12}^{\text{lock}}$ significantly increases the likelihood of forming defects such as wrinkling or textile folding [40]. Also, yarn compaction reduces resin permeability during subsequent infiltration and may lead to so-called "dry spots". Adverse appearance aside, such defects may significantly impair the component's load-bearing capacity [38,41].

One option to improve the forming result and mitigate defects are restraining forces which control the local material draw-in dur-

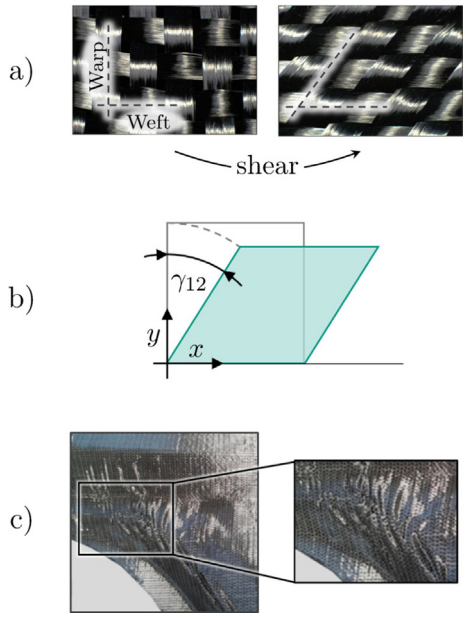


Fig. 2. Visualisation of textile deformation: a) plain-weave fabric before and after shearing (close-up), b) shear angle definition, c) example of textile wrinkling [38].

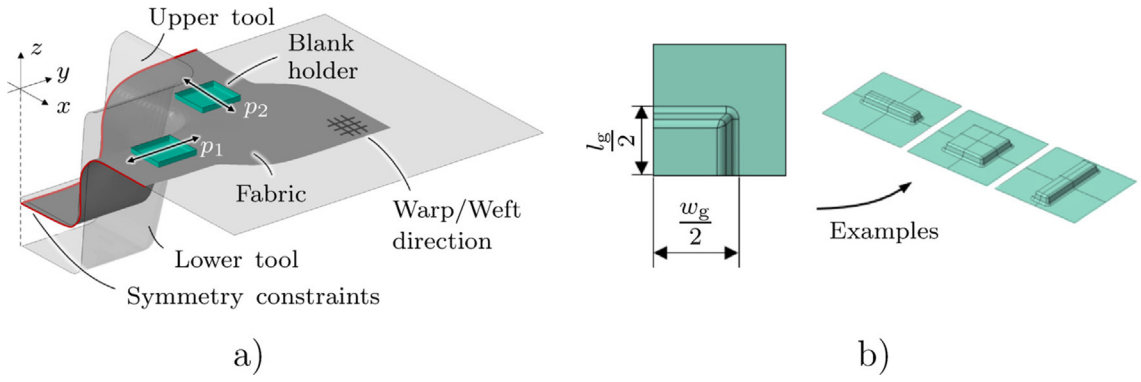


Fig. 3. a) Schematic illustration of an exemplary process simulation setup for blank holder assisted draping with applied symmetry conditions. b) Parameter definition for the box-geometries (width w_g and length l_g) along with geometry examples for visualisation.

ing tool stroke. Active or passive systems can introduce these forces [42] such as textile grippers [43], clamping systems [44], drawbeads [45] or global blank holders [46,47]. If adjusted appropriately to geometry and material, these forces evoke local tensile stresses in the textile which alleviate defects to a certain extent. Despite comprehensive knowledge on underlying physical mechanisms only little and mostly qualitative guidance is available on how to identify optimal parameters for a given geometry.

2.2.2. Simulation task

Setup. The studied use case is optimisation of two pressure pad positions $\boldsymbol{p} = (p_1, p_2)$ during draping of a woven fabric single layer into corners of variable cuboid geometries g ("boxes", cf. Fig. 3 a). The quadratic pressure pads (25×25 mm) can be positioned along the component perimeter with 25 mm distance to the stamp opening line. A Coulomb friction law with a constant, isotropic coefficient of friction of 0.25 and pad downforces of 7 N impose tangential stresses which control the deformation of the fabric during draping. The deliberately double symmetric scenario enables symmetry constraints for reduced computational effort.

The fabric is quadratic and measures $w_f \times l_f = 600 \times 600$ mm in width and length with a constant thickness of 0.3 mm. Warp and weft direction align with the x- and y-axes, respectively. To facilitate engineering interpretation of the results and the RL-algorithm behaviour, this work focuses on cuboid geometries whose corners are deliberately severe in order to evoke strong defect tendencies (cf. Fig. 3 b)). While the geometries' height is fixed (150 mm),their width w_g and length l_g are bound relative to the fabric by $\frac{1}{6} \leqslant \frac{w_g}{w_f}, \frac{l_g}{l_f} \leqslant \frac{1}{2}$. Within these bounds Latin Hypercube Sampling ensures a space-filling sampling of the geometry parameters.

Simulation model. Regarding forming simulation, a macroscopic FE-based modelling approach is utilised, which allows for the constitutive description of relevant deformation mechanisms of the studied fabric. The simulation model, comprising parameterised membrane, bending and contact models for the balanced plain weave carbon fabric T700SC-12K-50C by ZOLTEK, is embedded within the commercially available FE-solver ABAQUS/EXPLICIT using VUMAT and VUGENS user-subroutines.

Superimposed membrane and shell elements account for the textile-typical decoupling of membrane and bending behaviour. Membrane behaviour [48] is implemented using a hyper-viscoelastic material formulation, which correctly accounts for fibre reorientation under large deformation. Shear locking is addressed by a non-linear shear modulus. Bending behaviour is implemented using a hypo-viscoelastic formulation coupled with a Voigt-Kelvin approach in a non-orthogonal, curvilinear fibre-parallel frame [49]. Both, membrane and bending behaviour have been validated in experiments, cf. [48,49]. ABAQUS' built-in general contact algorithm models the tool-ply interface. The tools are modelled as rigid surfaces and close in 2 s at constant speed.

Model generation. An in-house developed, PYTHON-based framework [50,51] is used for fully automatic model generation, solution and result evaluation. It further enables concurrent handling of multiple jobs to keep the overall computation time within reasonable bounds. Please note that the applied symmetry conditions may confine process manipulability. As discussed in e.g. [52] allowing for asymmetric material intake leads to a wider range of forming results than a strictly symmetric intake. Exploiting symmetry for numerical efficiency thus inadvertently limits the process optimisation potential. However, this concession is deemed acceptable since evaluation of the general applicability is prioritised over optimisation impact. Once the methodological principle is validated, follow-up work may address a wider range of process conditions or geometries.

2.2.3. Optimisation task for the use case of stamp forming

Part quality definition. Any process optimisation requires at least one scalar part quality measure q for maximisation or minimisation, i.e. the objective function. In fabric forming optimisation, the shear angle γ_{12} is a frequently used proxy for defect formation (cf. Section 2.2). More specifically, it is common practice to evaluate the maximum shear angle $\gamma_{12}^{\max}$ for optimisation [16].

However, $\gamma_{12}^{\max}$ loses expressiveness near the locking angle ($\gamma_{12}^{\max} \approx \gamma_{12}^{\text{lock}}$) since forming defects such as wrinkling may continue to grow although $\gamma_{12}^{\max}$ remains practically constant. Hence, this work applies the modified mean Gaussian curvature κ as proposed by Haanappel [53] as a direct quantification of local fabric wrinkling. In accord with the forming literature [51,49], curvature evaluation takes place just before tool closure (80% tool stroke) when wrinkles are well pronounced.

As κ is a field quantity it requires mapping to a scalar quantity for optimisation. In theory, the maximum curvature $\kappa_{\max}$ could be evaluated, yet in practice this proves erratic and susceptible to physically implausible outliers. In such cases literature recommends characterising the defects by a global statistical distribution instead of a local criterion [54]. Fig. 4 shows a histogram of all fabric elements' κ along with an exemplary forming result for visualisation. While the vast majority of the fabric experiences mild curvature, only few elements show severe curvature.

Under such conditions, literature suggests evaluating a sufficiently high quantile κ^{qnt} of a fitted Weibull distribution as a proxy for maximum curvature [54]. Fig. 4 shows an exemplary Weibull fit along with the 99%-quantile κ^{qnt} as used in this work to reproducibly quantify part quality q, i.e. $\kappa^{\text{qnt}} = q = \varphi^{\text{fem}}$.

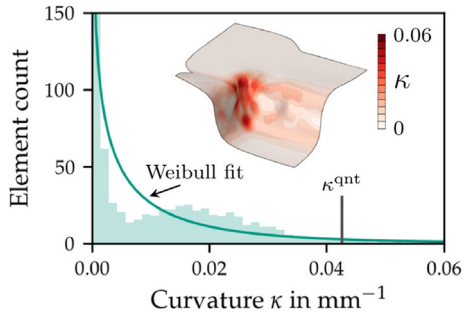


Fig. 4. Exemplary fabric forming result and histogram of the local fabric curvature κ. The histogram also shows a fitted Weibull distribution, whose 99%-quantile quantifies part quality during optimisation.

2.2.4. Temporary substitute function for efficient hyperparameter tuning

The performance of neural networks depends on an appropriate adjustment of so called hyperparameters, e.g. number of neurons per layer. They are typically determined empirically by parametric studies which requires numerous algorithm trials and simulations. Since each evaluation of φ^{fem} (simulation run) takes about 20 min to complete, hyperparameter tuning could become prohibitively expensive when done directly on φ^{fem}. However, in this work the deliberate simplicity of the geometries allows to engineer a substitute function $\varphi^{\text{sub}}|_g$, which temporarily replaces $\varphi^{\text{fem}}|_g$ during hyperparameter tuning (Section 3).

To outline the underlying reasoning, the objective function $\varphi^{\text{fem}}|_g$ of an example geometry g is visualised through a full-factorial (FF) sampling of all possible pad positions: Fig. 5 a) illustrates the variation of the pad positions p_1 and p_2 for an example geometry g and Fig. 5 b) shows the according contour plot of $\varphi^{\text{fem}}|_g(p_1, p_2)$ for 13×13 different pad positions. Please note that the FF-samples serve for visualisation only. They are in no way involved in algorithm training. Clearly, φ^{fem} shows a distinct minimum and diagonally above two maxima. The yellow marker in Fig. 5 a) and b) relates pad position and φ^{fem}. Imagined movements of the marker, e.g. during optimisation, simultaneously reposition the pads and alter φ^{fem}.

The overall shape of φ^{fem} with minima and maxima can be qualitatively reproduced by an analytical function

$$\begin{aligned}\varphi^{\text{sub}}(\boldsymbol{p})|_g &= \varphi^{\text{sub}}(p_1, p_2)|_g \\ &= \varphi_1^{\text{sub}}(p_1)|_g + \varphi_2^{\text{sub}}(p_2)|_g\end{aligned} \tag{3}$$

wherein

$$\varphi_{1,2}^{\text{sub}}(p_{1,2})|_g = \frac{1}{2}\left[\exp\left(a_{1,2}|_g\right) - \exp\left(b_{1,2}|_g\right)\right] \tag{4}$$

In essence, Eq. 3 is a sum of shifted (Gaussian) bell curves. It is designed to qualitatively mimic $\varphi^{\text{fem}}|_g$ and its location of maxima

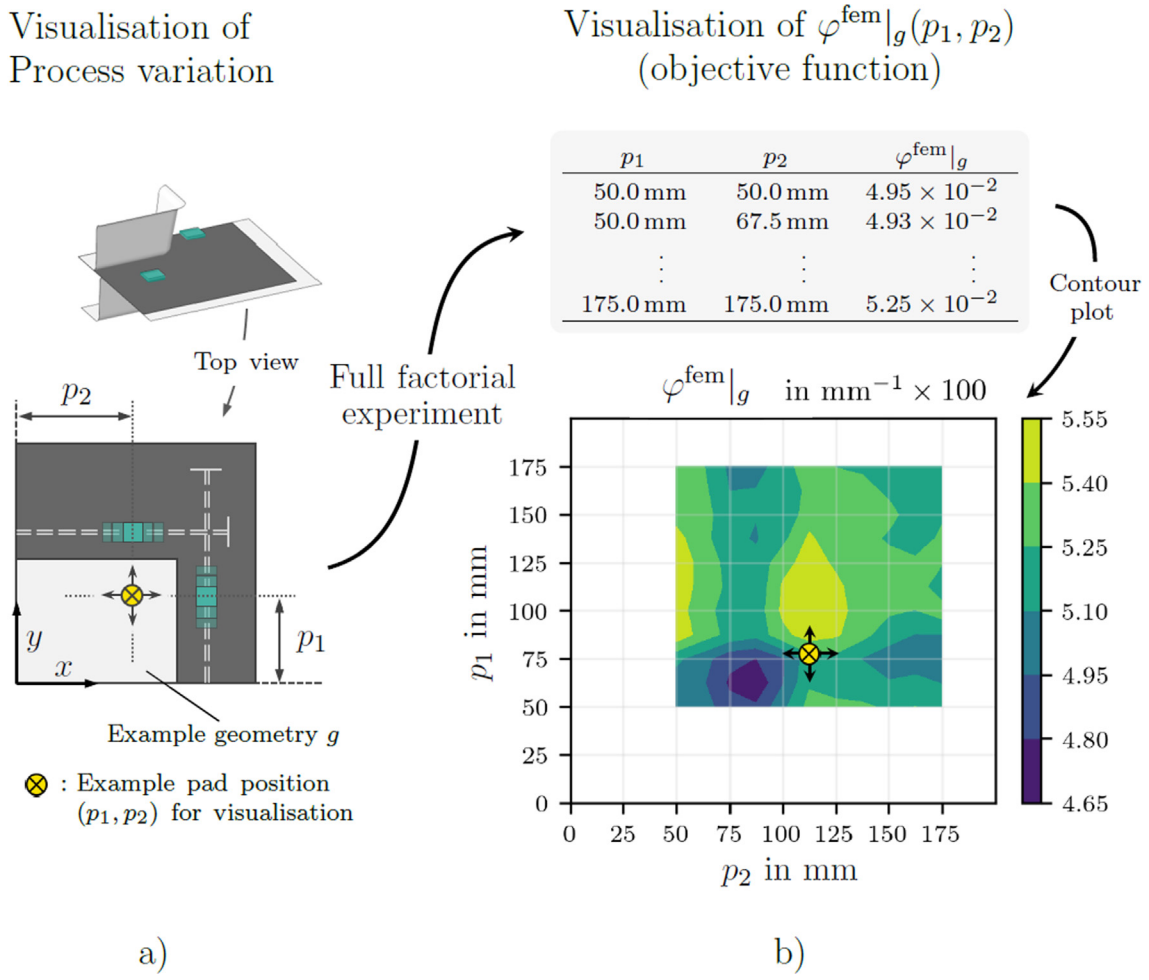


Fig. 5. a) Example visualisation of the process variation through repositioning of pressure pads. b) Contour plot of the objective function $\varphi^{\text{fem}}|_g$ of an example geometry g obtained through full-factorial sampling of all possible pad positions.

and minima for a geometry g of width w_g and length l_g. It is additionally normalised to the interval $[-1, 1]$ for value-wise distinction from φ^{fem}. The coefficients, the underlying rationale and a comparison between φ^{sub} and φ^{fem} are given in the Appendix. Once suitable hyperparameters are found, φ^{sub} is discarded and training takes place on physics-based simulations φ^{fem}.

2.3. Two-step ML-model for Π

An automatic part evaluation, as aspired in this work, requires a geometry representation format. Several approaches have been developed in the past, which broadly split into local and global descriptions. Local approaches often describe the geometry by individual points with fundamental properties such as principal curvatures. This point-wise description suits well point-wise processes such as metal-driving [29]. However, interactions of neighbouring geometric features are observed in stamp forming, which point-wise approaches cannot describe. Thus, geometry descriptions on component level have emerged.

In some cases, geometry parameters, e.g. corner radii or aspect ratios, can efficiently describe component features [6,55]. While this may be sufficient in some cases, such approaches are always restricted to a predefined parameterisation scheme and thus hardly transferable to arbitrary geometries. Other geometry representations localise distinct features, e.g. cutouts or beads, and quantify their relative position [56]. Although complex geometries can be encoded, such approaches still rely on hand-crafting features and prove susceptible to modelling errors. As a remedy, ML-inspired approaches have been proposed,e.g. [57,58,27,59], which suggest non-parametric geometry encoding in grid-structured data such as pixels or voxels.

Such grid-based approaches can efficiently retain spatial relations, which often govern manufacturing scenarios. In case of textile forming, a close correlation between component curvature and material strain can be observed. A collision-free tool closure requires an undercut-free component geometry and allows for projection into the tool-plane. Grey-scale values can quantify the local elevation above the tool-plane similar to a topographic map (Fig. 6). Likewise, grey-scale values represent the material strain field of the simulation with each pixel representing an element's shear angle γ_{12}. See[59]for details on the encoding scheme.

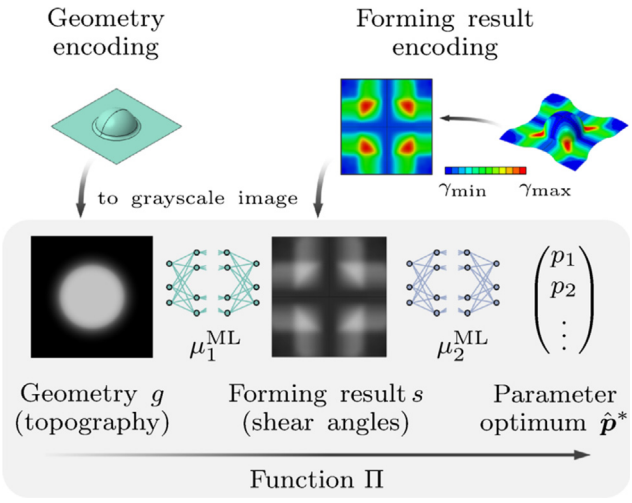


Fig. 6. Image-based implementation of function Π with two sub-functions μ_1^{ML} and μ_2^{ML} inspired by[59,30]. After conversion to a topographic map, μ_1^{ML} interprets the geometry g and yields an estimation of the shear angle distribution s. Subsequently, μ_2^{ML} estimates optimal process parameters $\hat{\boldsymbol{p}}^*$ on the basis of s.

Convolutional neural networks (CNNs), a sub-type of neural networks specialised in image processing, are deliberately designed to identify patterns in spatially structured data. See [60] for a detailed discussion of CNNs. Given a sufficient number of layers and templates ("deep" network), complex relationships can be encoded. In this work, CNNs learn the relation between (box-) geometries, forming results and optimum process parameters. More specifically, the envisaged function Π consists of two composed sub-functions μ_1^{ML} and μ_2^{ML}, each implemented as a CNN (Fig. 6):

$$\hat{\boldsymbol{p}}^* = \Pi(g) = \mu_2^{\text{ML}}\left(\mu_1^{\text{ML}}(g)\right) \tag{5}$$

At first, μ_1^{ML} evaluates a geometry grey-scale-image g and returns an estimation of the spatial shear angle distribution s. Then, μ_2^{ML} evaluates s and yields a process parameter recommendation $\hat{\boldsymbol{p}}^*$. This work concentrates on μ_2^{ML} and uses an existing, pre-trained implementation of μ_1^{ML} from the prior work [59].

2.4. Training of μ_2^{ML} by reinforcement learning

Formal concept of reinforcement learning. The function μ_2^{ML} shall learn to estimate optimal process parameters given an estimated forming result $s = \mu_1^{\text{ML}}(g)$. From an RL-perspective, this corresponds to finding a function – often termed "policy" – which yields the optimal action $\boldsymbol{p}^*$ (process parameters) in a given state s (forming result). The optimal action is the one that yields the maximal reward q (part quality)

$$\mu_2^{\text{ML}} = \arg\max_{\boldsymbol{p}} q(s, \boldsymbol{p}). \tag{6}$$

This work assumes constant process parameters $\boldsymbol{p}$, i.e. $\boldsymbol{p}$ is set once and kept constant during the tool stroke. Consequently, only a single decision must be taken and state transitions do not occur.

For completeness, please note that other process technologies may require parameter variations over time, though. For instance, consider in-situ adaptions of the tool closing speed to avoid defects from excessive or uneven cavity pressures. In such cases, the process passes through a whole sequence of states (pressure distributions) and according actions (increase/decrease speed), which can be seen as an MDP. It can also be solved using RL but goes beyond the scope of this work.

In theory, classical optimisation techniques could solve Eq. 6. As this is usually time-consuming, Actor-Critic-techniques eliminate the optimisation by parametrising μ_2^{ML}, e.g. through a neural network with adjustable weights $\boldsymbol{\theta}$, and gradually tuning $\boldsymbol{\theta}$ in direction of increasing part quality q during training.

Network training. The training scheme is visualised in Fig. 7 and outlined in the following. In one training iteration i, at first new observations for each geometry $g \in G$ are generated. To this end, μ_1^{ML} draws a sample g from the geometry set G and estimates the forming result s (shear angles). Then, μ_2^{ML} interprets s and infers a parameter recommendation $\hat{\boldsymbol{p}}|_g$. An according forming simulation $\varphi^{\text{fem}}|_g(\hat{\boldsymbol{p}}|_g)$ determines the resulting quality q. Eventually, the observation tuple $\{s, \hat{\boldsymbol{p}}|_g, q\}$ is appended to a process memory M. The procedure repeats for the remaining geometries in G.

The subsequent network training involves two networks: the desired network μ_2^{ML} (the "actor" in RL terms) and an auxiliary network $\mu_{\text{aux}}^{\text{ML}} : S, A \mapsto Q$ ("critic"). At first, the critic-parameters $\boldsymbol{\theta}_{\text{aux}}^{\text{ML}}$ are adjusted to match the observations in the memory M. See [30] for details of its training. The updated critic then provides information to compute the gradient $\nabla_{\theta} q$ for the actor-update, i.e. the direction of increasing quality q:

$$\boldsymbol{\theta}_{2,i+1}^{\text{ML}} = \boldsymbol{\theta}_{2,i}^{\text{ML}} + \eta \ \nabla_{\theta} q. \tag{7}$$

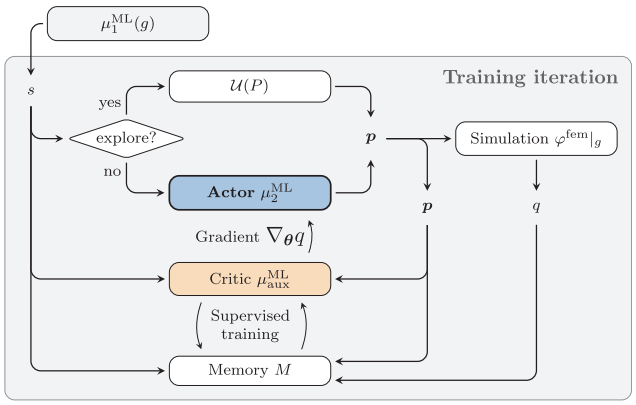


Fig. 7. Training scheme of μ_2^{ML}: Process parameters $\boldsymbol{p}$ are either inferred by the actor μ_2^{ML} ("exploitation") or drawn randomly ("exploration"). A simulation $\varphi^{\mathrm{fem}}|_g(\boldsymbol{p})$ evaluates the according quality q and completes a new observation for the memory M. Based on M, the critic is updated and provides gradient-information for the eventual actor update.

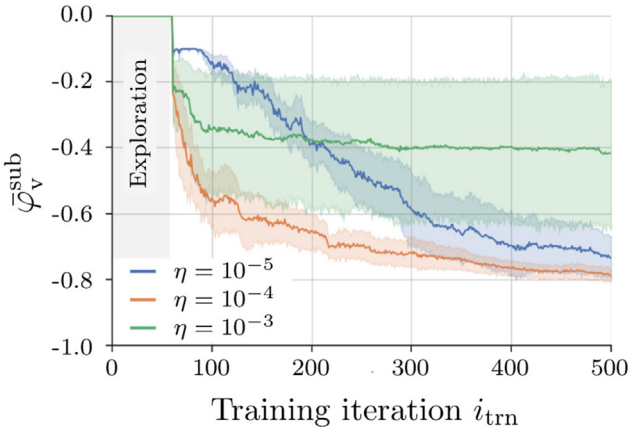


Fig. 8. Evolution of $\bar{\varphi}_{\mathrm{v}}^{\mathrm{sub}}$ for different learning rates η. Best performance is observed for $\eta = 10^{-4}$ with which $\bar{\varphi}_{\mathrm{v}}^{\mathrm{sub}}$ decreases most rapidly, yields the lowest value of $\bar{\varphi}_{\mathrm{v}}^{\mathrm{sub}}$ and shows least scatter.

Therein, η represents the step size ("learning rate"). As it controls both training stability and speed, η is an important hyperparameter to tune (cf. Section 3.1). A detailed derivation of $\nabla_{\theta} q$ is given in the Appendix.

The network parameters $\boldsymbol{\theta}_2^{\mathrm{ML}}$ are initialised randomly. Thus, gradient-information will be of low significance at training begin and a direct application of Eq. 7 most likely makes μ_2^{ML} approach the local minimum in closest proximity ("exploitation") without searching the parameter space for better, ideally global optima ("exploration"). To reconcile this exploration-vs-exploitation dilemma, this work employs a variant of "ε-greedy" exploration: Actor updates (Eq. 7) are delayed until the critic has reached sufficient accuracy. Prior work hints $i_{\mathrm{trn}} = 60$ iterations sufficient [30]. Until then, parameters are drawn randomly from a uniform distribution $\mathcal{U}(P)$ only. Once the actor training starts ($i_{\mathrm{trn}} > 60$), the chance $\varepsilon \in [0, 1]$ of probing random parameters gradually decreases until μ_2^{ML} acts virtually deterministic ($\varepsilon \approx 0$). See [30] for implementation details.

3. Results and discussion

The results split up into two parts: Hyperparameter studies for best practices during training (Section 3.1) and assessment of optimisation performance in the draping use case (Sections 3.2 and 3.3).

3.1. Algorithm hyperparameter study using φ^{sub}

ML-algorithms require tuning of hyperparameters for optimal learning performance, typically through extensive parametric studies. As this becomes time-consuming with actual simulations φ^{fem}, the substitute function φ^{sub} (Section 2.2.4) is used instead. Results hint three hyperparameters as most decisive: The learning rate η, data augmentation and the number of training geometries n_{T}. As individual learning runs scatter (random network initialisation), ten independent runs are performed for each hyperparameter configuration and their average $\bar{\varphi}^{\mathrm{sub}}$ is evaluated.

In order to test μ_2^{ML}'s optimisation performance on new geometries, this work considers five separate validation box-geometries $g_{\mathrm{v1\ldots5}}^{\mathrm{sub}}$. The validation geometries stem from a separate Latin Hypercube Sampling of the box-geometries' width w_{g} and length l_{g} and resemble the geometry samples in Fig. 3b). Owing to the separate sampling, they are not part of the training geometries and always new to μ_2^{ML}. In the following plots, solid lines show the average performance $\bar{\varphi}_{\mathrm{v}}^{\mathrm{sub}}$ of μ_2^{ML} across the validation geometries $g_{\mathrm{v1\ldots5}}^{\mathrm{sub}}$ during training. The shaded areas visualise the according 95 %-confidence interval.

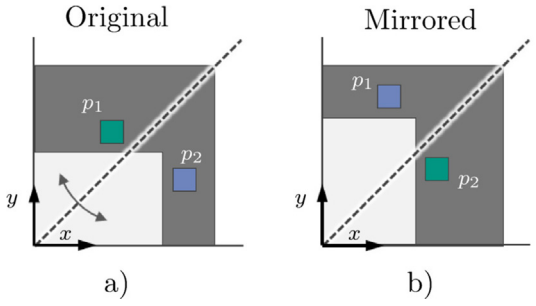


Fig. 9. Visualisation of data augmentation through mirroring (top view). a) Before and b) after mirroring about the x-y-bisector (dashed line). Both yield the same part quality q.

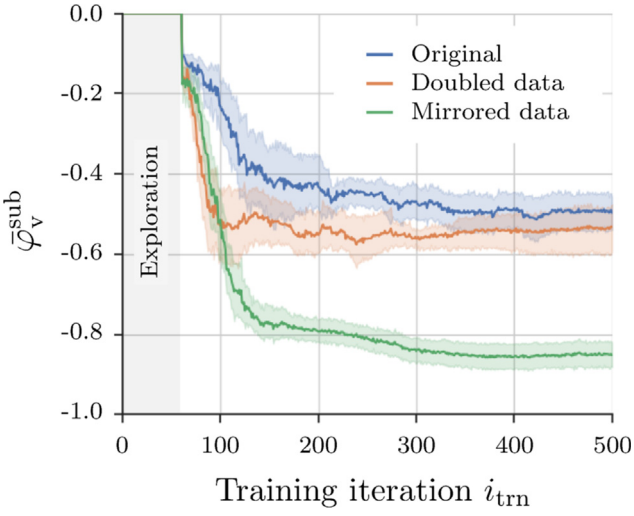


Fig. 10. Evolution of $\bar{\varphi}_{\mathrm{v}}^{\mathrm{sub}}$ for different data treatments (no treatment, data doubling and mirroring). Mirroring significantly improves both learning speed and final result of $\bar{\varphi}_{\mathrm{v}}^{\mathrm{sub}}$ at negligible cost.

Learning rate. According to [60], the learning rate η (step-size) is one of the most relevant hyperparameters with a suggested value range $10^{-6} < \eta < 1$. In general, smaller values stabilise learning but in turn significantly prolong training. Typically, complex problems require smaller learning rates and thus η allows for a rough qualitative assessment of problem complexity.

Fig. 8 shows the evolution of $\bar{\varphi}_{\mathrm{v}}^{\mathrm{sub}}$ for $\eta = 10^{-3}, 10^{-4}, 10^{-5}$. Lower and higher values did not show any convergence at all and are thus omitted for readability. After the initial exploration phase ($i \leqslant i_{\mathrm{trn}}$), the graphs decline rapidly before they asymptotically approach a constant value. The highest learning rate $\eta = 10^{-3}$ performs worst (highest final value, most scatter). Best performance is observed for $\eta = 10^{-4}$. Not only yields it the best (i.e. lowest) final value ($\bar{\varphi}_{\mathrm{v}}^{\mathrm{sub}} \approx -0.8$), it also converges the fastest and shows least scatter. Comparison to the suggested range $10^{-6} \leqslant \eta \leq 1$ implies a medium-complex learning problem.

Data augmentation. The second aspect concerns the use of data augmentation strategies. In general, ML algorithms benefit from larger datasets, yet in many situations, data is scant and acquisition of further samples is expensive. In ML a common strategy to improve performance is to conjure additional data by careful variation of existing data. For instance, in image recognition transformations such as mirroring, image rotation or distortion can be applied [61].

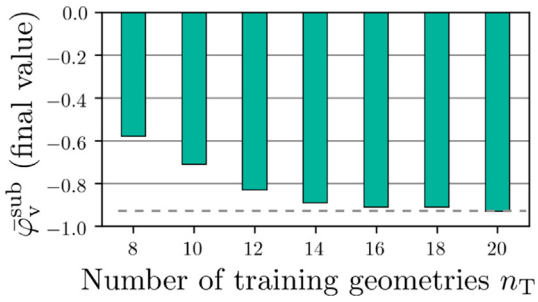


Fig. 11. Bar plot of the best (lowest) value of $\bar{\varphi}_{\mathrm{v}}^{\mathrm{sub}}$ for different numbers of training geometries n_{T} in G. Increasing n_{T} initially improves $\bar{\varphi}_{\mathrm{v}}^{\mathrm{sub}}$ until it stays practically constant from $n_{\mathrm{T}} \approx 14$ on.

In this work, any augmentation strategy must retain the pixels' spatial information. Thus, eligible operations reduce to rotation and mirroring. The applied symmetry conditions, cf. Fig. 3 a), preclude rotation operations, leaving mirroring about the x-y-bisector as the only available augmentation option. The studied textile ZOLTEK T700SC-12K-50C is a balanced, plain weave fabric. This implies identical membrane and bending behaviour in warp- and weft-direction[62,63] and yields an additional symmetry plane at 45 ° (dashed line). Thus, original and mirrored configuration give identical, yet mirrored, forming results φ^{fem}. Fig. 9 visualises the concept.

Fig. 10 visualises the impact of data augmentation on the learning performance. It shows three different dataset configurations: original data (no treatment), mirrored (augmented) data and doubled data. The latter refers to original data which is merely copied and appended to the dataset. Thereby no additional information is introduced, yet the dataset is of the same size as the augmented dataset. This prevents possible graph distortions from unequal dataset sizes, as can be seen when comparing original to doubled data. Although they use the same information content, the doubled dataset seems to outperform original dataset (faster decline). Yet, this is a premature conclusion as the dataset size directly determines the number of network adjustments per iteration (i.e. training speed). For a valid "mirrored-vs-original"-comparison equal dataset sizes must be ensured – here done by "data doubling".

Both graphs, doubled and mirrored data, initially decline at the same rate, yet final values differ: The doubled data approaches $\bar{\varphi}_{\mathrm{v}}^{\mathrm{sub}} \approx -0.5$ while the mirrored data achieves $\bar{\varphi}_{\mathrm{v}}^{\mathrm{sub}} \approx -0.85$. Hence, exploitation of known symmetries (data augmentation) indeed introduces new, usable information and significantly improves the learning result at low cost. Note that in this work, only one symmetry plane can be exploited. Yet the impact of data augmentation expectedly grows when more symmetries are present.

Training geometries. The third factor on training performance is the number of training geometries n_{T} in the geometry set G. Using only few training geometries most likely does not provide sufficient information for generalisation. Conversely, when already using many geometries, adding even more geometries probably introduces only little additional information. Thus, a threshold number of training geometries must exist, which provides just sufficient information for generalisation.

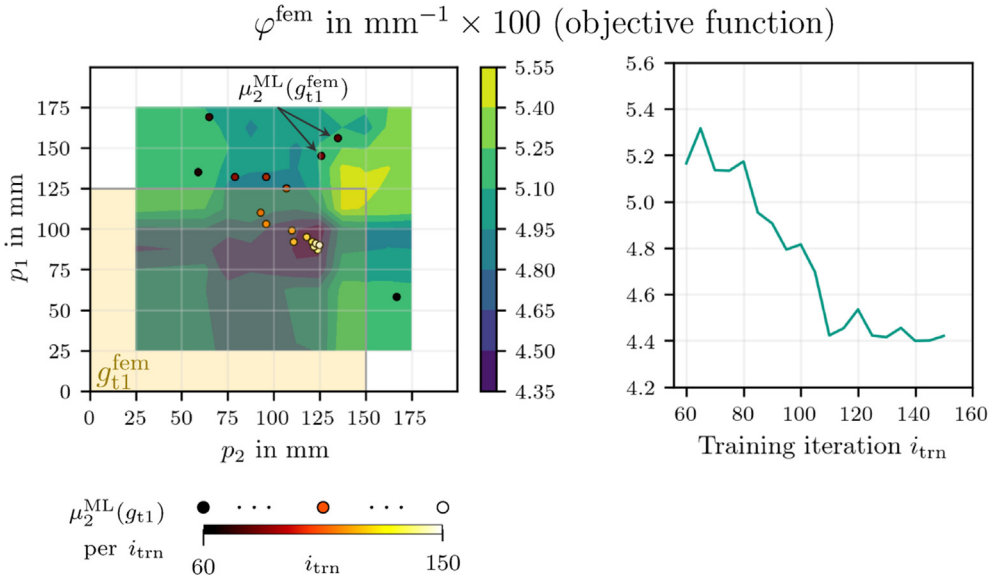


Fig. 12. Visualisation of the objective function φ^{fem} (contour plot) and μ_2^{ML}'s parameter recommendations (markers) during training for training geometry $g_{\mathrm{t1}}^{\mathrm{fem}}$. The line plot allows for convergence assessment of φ^{fem}.

To validate this hypothesis, μ_2^{ML} is trained with $n_{\mathrm{T}} = 8, 10, \ldots, 20$ geometries in G: The obtained graphs resemble the previous plots, i.e. initial descent followed by an asymptote. However, significant differences appear regarding the final values of $\bar{\varphi}_{\mathrm{v}}^{\mathrm{sub}}$ which the bar plot in Fig. 11 summarises.

Overall, adding geometries to G always improves $\bar{\varphi}_{\mathrm{v}}^{\mathrm{sub}}$, yet the first additional geometries contribute most. This supports the above intuition of a "marginal utility" of geometries. More specifically, from $n_{\mathrm{T}} = 14$ on, the $\bar{\varphi}_{\mathrm{v}}^{\mathrm{sub}}$ remains approximately constant. This hints a threshold of geometry samples, beyond which additional geometries contribute just little additional information. In general terms, the results suggest that a finite number of geometry samples n_{T} in G holds sufficient information to analyse any new sample from G. Although n_{T} will probably grow with complexity of G, the results hint that it will remain finite. Note that this finiteness is a necessary precondition when aiming for a fully geometry-independent surrogate in the long term.

3.2. Algorithm deployment on FE-simulations

The hyperparameter studies in the previous section reveal a promising RL-configuration for training on φ^{fem} (FE-simulation). In accord with Section 3.1, training involves $n_{\mathrm{T}} = 14$ geometries per iteration i with $i^{\max} = 150$ iterations in total. Data augmentation is also applied. Intermediate evaluations of μ_2^{ML} on the training and validation geometries allows monitoring the training progress. In the following, testing results are first discussed for one exemplary training geometry $g_{\mathrm{t1}}^{\mathrm{fem}}$ followed by three validation geometries $g_{\mathrm{v1\ldots3}}^{\mathrm{fem}}$.

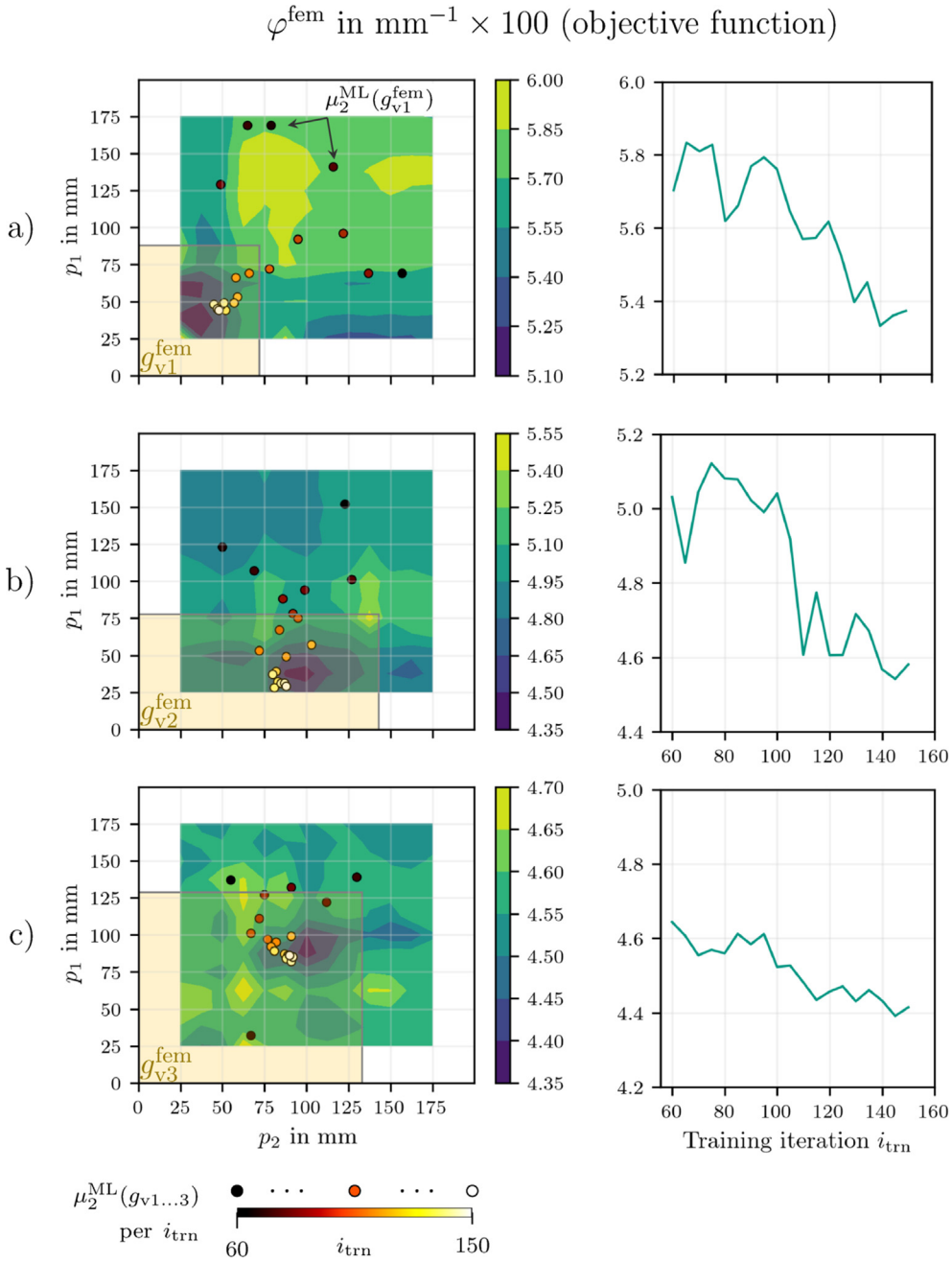


Fig. 13. Visualisation of the objective function φ^{fem} (contour plot) and μ_2^{ML}'s parameter recommendations (markers) during training for (unknown) validation geometries $g_{\mathrm{v1\ldots3}}^{\mathrm{fem}}$. The line plot allows for convergence assessment of φ^{fem}.

Training geometries. Fig. 12 shows a contour plot of φ^{fem} (objective function) obtained through full-factorial sampling along with a plan view of the geometry dimensions analogous to Fig. 5. As before, these full-factorial samples serve for visualisation only. Markers visualise μ_2^{ML}'s parameter recommendations over the course of training. Their hue denotes the order of appearance during training.Analogous to $\bar{\varphi}_{\text{v}}^{\text{sub}}$ in Section 3.1, the line plot on the right visualises φ^{fem} to assess convergence.

The plots show a successful optimisation: The markers initially appear almost random with significant scatter, then begin to concentrate and eventually accumulate around the optimum. As a result, the convergence plot initially ($60 < i_{\text{trn}} < 80$) shows some wavering around $\varphi^{\text{fem}} \approx 5.2 \times 10^{-2}\text{mm}^{-1}$ before it decreases by $\approx 13\%$ within 30 iterations ($80 < i_{\text{trn}} < 110$) to an approximately constant value of $\varphi^{\text{fem}} \approx 4.5 \times 10^{-2}\text{mm}^{-1}$. Analogous results are obtained for the other training geometries, which leads to the conclusion that μ_2^{ML} is able to estimate optimal parameters for a range of "known" geometries.

Validation geometries. For assessment of μ_2^{ML}'s performance on "unknown" geometries Fig. 13 gives analogous plots for three validation geometries $g_{\text{v1...3}}^{\text{fem}}$.

Overall, the observed behaviour resembles Fig. 12, yet two remarkable differences prevail: First, unlike the training geometry g_{t1}, the markers approach the optimum but do not exactly pinpoint it. More specifically, in Fig. 13 a) p_1^* and p_2^* are over-, and in c) and b) underestimated. Second, the initial wavering around the constant value of φ^{fem} takes a little longer ($60 < i_{\text{trn}} < 100$). This suggests that μ_2^{ML} requires more training iterations to also improve on the validation geometries $g_{\text{v1...3}}^{\text{fem}}$ compared to the training geometries $g_{\text{ti}}^{\text{fem}}$.

The differences are well explicable bearing in mind the difference between $g_{\text{t1}}^{\text{fem}}$ and $g_{\text{v1...3}}^{\text{fem}}$: Since $g_{\text{t1}}^{\text{fem}}$ is part of the training geometries, μ_2^{ML} directly explores the corresponding objective function $\varphi^{\text{fem}}|_{g_{\text{t1}}^{\text{fem}}}$ during training. Thus, evaluating μ_2^{ML} on $g_{\text{t1}}^{\text{fem}}$ solely amounts to "recalling" information from a previous (i.e. "known") process situation. In contrast, the validation geometries $g_{\text{v1...3}}^{\text{fem}}$ are not part of the training geometries and their objective functions $\varphi^{\text{fem}}|_{g_{\text{v1...3}}^{\text{fem}}}$ are "unknown" to μ_2^{ML} during testing. Consequently, μ_2^{ML} must make statistical inference on the basis of previous training situations, which introduces certain deviations. Yet these deviations are deemed acceptable as μ_2^{ML}'s recommendations are always near-optimal.

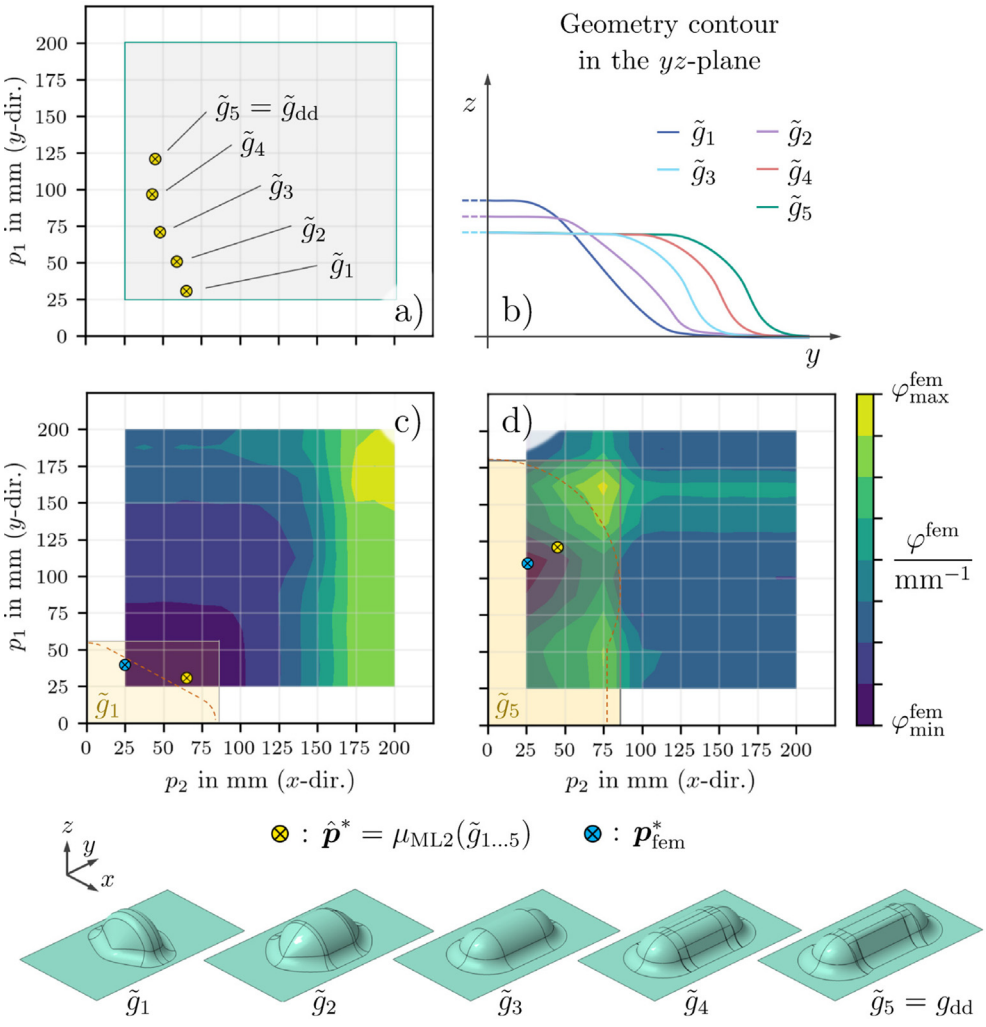


Fig. 14. a) Visualisation of μ_2^{ML}'s parameter recommendations $\hat{\boldsymbol{p}}^*$ for the five test geometries $\tilde{g}_{1...5}$ shown at the bottom, b) their contour line for comparison of height and length. Sub-image c) and d) show φ^{fem} for the 'limit'-geometries $\tilde{g}_{1,5}$. A blue marker shows the true optimum $\boldsymbol{p}^*_{\text{fem}}$ and a yellow marker μ_2^{ML}'s parameter recommendation $\hat{\boldsymbol{p}}^*$ after training.

Overall, μ_2^{ML} is capable of extracting overarching process patterns from the supplied training geometries in G and utilises them for unknown (validation-) geometries.

3.3. Transferability outside the training geometry class

Having tested μ_2^{ML}'s performance on unknown geometries inside the training geometry-class of boxes ($g_{\mathrm{v1...3}}^{\mathrm{fem}} \in G$), this section discusses its prediction performance on five geometries outside this class, i.e. $\tilde{g}_{1...5} \notin G$. The gallery at the bottom of Fig. 14gives a visual impression. Starting with a rather compact geometry $\tilde{g}_1$, they stepwise elongate in y-direction by 20 mm per step. For the first two steps, also their height and width reduce by 10 mm and their tips morph from conical to spherical. Sub-image b) shows their contour line (silhouette) in the yz-plane for visualisation. The geometries $\tilde{g}_{1...3}$ are generated by rotating this contour about the y-axis; the geometries $\tilde{g}_{4,5}$ additionally feature a non-rotational, trapezoidal part in the centre. The last geometry $\tilde{g}_5$ is identical to the double-dome g_{dd}, a common benchmark geometry in fabric forming. It features several forming-relevant characteristics, e.g. non-, single- and double-curved areas. Note that $\tilde{g}_{1...5}$ are all near-convex and thus show qualitatively a forming behaviour similar to the training box-geometries.

Fig. 14 a) plots μ_{ML2}'s parameter recommendations $\hat{\boldsymbol{p}}^*$ with a yellow marker for each geometry. Similar to the boxes, $\hat{\boldsymbol{p}}^*$ moves – with some wavering – in positive y- and negative x-direction as the geometries stretch. To validate the optimality of the recommendations, the subplots c) and d) give 'omniscient' contour plots of φ^{fem} (objective function) for the 'limit'-cases $\tilde{g}_1$ and $\tilde{g}_5$ analogous to Fig. 5. The ochre-shaded area marks their bounding box for comparison with previous plots; the dashed line outlines its plan-view contour. A yellow marker locates μ_2^{ML}'s parameter recommendation $\hat{\boldsymbol{p}}^*$, a blue marker the actual optimum $\boldsymbol{p}_{\mathrm{fem}}^*$ observed during a full-factorial experiment.

Although slightly overestimated, μ_2^{ML}'s parameter recommendations $\hat{\boldsymbol{p}}^*$ are in vicinity of the actual optima $\boldsymbol{p}_{\mathrm{fem}}^*$ and react in a plausible way to geometry variations. Thus, μ_2^{ML}'s recommendations are deemed useful suggestions even for geometries outside the training geometry-class of boxes. Note that μ_2^{ML} has only been trained on boxes (Section 3.2). This implies that the employed RL-based approach has successfully extracted tacit process-knowledge from process-samples and is able to apply it to new situations. This applicability across geometries significantly enhances current surrogate capabilities, which are typically geometry-specific "one-off"-models. Follow-up research may further investigate, if a component-specific continuation of training refines μ_2^{ML}'s initial recommendations so that they converge to the actual optimum similar to a classical surrogate during optimisation.

4. Conclusion

This work presents a Reinforcement Learning (RL) based approach for estimation of optimal manufacturing parameters for variable component geometries. In this case study, pressure pads must be positioned to optimise the material draw-in during fabric forming. Unlike classical surrogate-based optimisation (SBO) the presented approach trains a function Π which takes the component geometry as input and directly estimates optimal process parameters (output). The range of considered geometries is deliberately simplified to cuboid boxes to facilitate engineering interpretation. Algorithm training takes place on an FE-simulation environment.

Two aspects became evident during hyperparameter tuning. First, data augmentation can be used to improve training performance. Second, a finite number of training geometries introduces sufficient information to train Π – a necessary precondition for algorithm generalisability. After deployment on FE-simulations, the results hint that Π gives useful parameter estimations for geometries inside and – to a certain degree – outside the training geometry class. This implies that the proposed approach is generally capable of extracting process "knowledge" from supplied process samples and applying it in new situations.

Further development is still envisaged. One important issue addresses the set of training geometries G. Presumably, Π's applicability shrinks with growing dissimilarity to training situations. On top of a profound criterion for validity limits, improvements on the generalisation capabilities are desirable. To this end, transfer learning has proven a data-efficient option[64]. Another option is to study the effect of different classes of training geometries G and identify, which geometry characteristics introduce most information for maximal generalisability [65]. A second aspect concerns component-specific algorithm refinement. Although generally reasonable, the parameter estimations are not strictly optimal (cf. Section 3.2). Thus, a follow-up study may evaluate whether – and if so how fast – Π converges to the actual optimum upon component-specific continuation of training.

Overall, numerical process simulation combined with ML-techniques appears a viable option to facilitate virtual engineering at early stages of product development.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

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 has also been funded by "MathSEE", the KIT Center for interdisciplinary mathematical research at the Karlsruhe Institute of Technology and is also part of the Young Investigator Group (YIG) "Green Mobility", generously funded by "Vector Stiftung".

Appendix A

A.1. Derivation of the substitute function φ^{sub}

The top of Fig. 15 shows $\varphi^{\mathrm{fem}}(\boldsymbol{p})$ for two geometries g_1 and g_2. The ochre-shaded areas demarcate the plan view dimensions of $g_{1,2}$. By comparing the two contour plots a relation between geometry dimensions (grey marker) and process optimum (blue marker) becomes apparent. More specifically, an approximately constant offset $\boldsymbol{\Delta}_{\boldsymbol{p}^*}$ (black arrow) between both is observed. Analogous analyses with additional geometries confirm the correlation.

This observation also aligns with the experimental and numerical findings in [44,16]. They report optimal positions of blank holders just next to doubly-curved areas, such as corners. Restraining forces at these positions counteract the material flow into the shear zone and thereby reduce shear deformation.

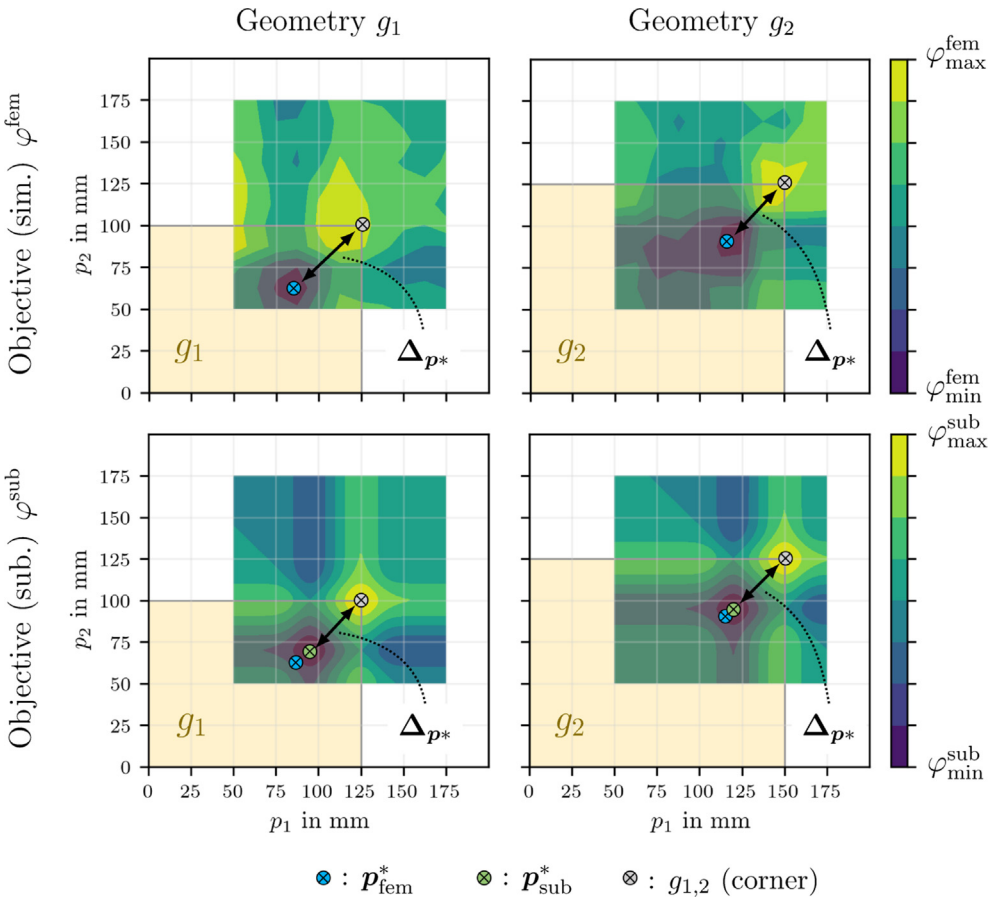


Fig. 15. Juxtaposed objective functions for two different geometries g_1 and g_2 obtained from FE process simulations φ^{fem} (top) and the devised substitute function φ^{sub} (bottom) for temporary replacement of φ^{fem} during hyperparameter studies.

This offset is used to construct an analytical function for the forming result $\varphi^{\text{sub}}(\boldsymbol{p})|_g$ as a temporary substitute for expensive simulations $\varphi^{\text{fem}}|_g$, cf. Eq. 3. The exponents in Eq. 8 are given as

$$\begin{aligned} a_{1,2}|_g &= a(p_{1,2})|_g = -\left(\tfrac{p_{1,2}-\tilde{p}_{1,2}|_g}{5}\right)^2 \\ b_{1,2}|_g &= b(p_{1,2})|_g = -\left(\tfrac{p_{1,2}-p^*_{1,2}|_g}{5}\right)^2. \end{aligned} \tag{8}$$

Therein $\tilde{p}_1|_g = w_g$ and $\tilde{p}_2|_g = l_g$ denote the worst and $p^*_1|_g = w_g - \Delta_{p_1*}$ and $p^*_2|_g = l_g - \Delta_{p_2*}$ the optimal positions (green marker) per geometry g with $\Delta_{p_{1,2}*} = 30$ mm. For comparison of with φ^{fem} contour plots of φ^{sub} are given at the bottom of Fig. 15.

A.2. Calculation of the gradient $\nabla_\theta q$

The gradient $\nabla_\theta q$ in Eq. 7 is obtained as follows: Following the chain rule from calculus it can be cast as

$$\nabla_\theta q = \nabla_{\boldsymbol{p}} q \cdot \nabla_\theta \boldsymbol{p}. \tag{9}$$

Therein, the first factor $\nabla_{\boldsymbol{p}} q$ is the gradient of the critic $\mu^{\text{ML}}_{\text{aux}}$ w.r.t. process parameters $\boldsymbol{p} = (p_1, p_2, \dots)^{\text{T}}$, i.e.

$$\nabla_{\boldsymbol{p}} q = \left(\frac{\partial q}{\partial p_1}, \frac{\partial q}{\partial p_2}, \dots\right)^{\text{T}} = \left(\frac{\partial \mu^{\text{ML}}_{\text{aux}}}{\partial p_1}, \frac{\partial \mu^{\text{ML}}_{\text{aux}}}{\partial p_2}, \dots\right)^{\text{T}} \tag{10}$$

and the second factor $\nabla_\theta \boldsymbol{p}$ is the Jacobian of the actor μ^{ML}_2 whose jl-th entry is the partial derivative of the j-th process parameter p_j w.r.t. the l-th actor parameter θ^{ML}_{2l}:

$$\frac{\partial p_j}{\partial \theta^{\text{ML}}_{2l}} = \frac{\partial \mu^{\text{ML}}_{2j}}{\partial \theta^{\text{ML}}_{2l}} \tag{11}$$

As discussed in [66], Eq. 9 is a policy gradient, i.e. its repeated application yields a policy which optimally solves an MDP.

Data availability

The raw and processed data required to reproduce these findings cannot be shared at this time as the data also forms part of an ongoing study.

References

[1] P. Reeves, C. Tuck, R. Hague, Additive manufacturing for mass customization, in: F.S. Fogliatto, G.J. Da Silveira (Eds.), Mass customization, Springer series in advanced manufacturing, Springer, London, 2011, pp. 275–289, https://doi.org/10.1007/978-1-84996-489-0_13.
[2] P. Moll, M. Schäfer, S. Coutandin, J. Fleischer, Reconfigurable modular production plant for thermoplastic hybrid composites, Prod. Eng. Res. Devel. 13 (3–4) (2019) 469–477, https://doi.org/10.1007/s11740-019-00898-z.
[3] M. Abosaf, K. Essa, A. Alghawail, A. Tolipov, S. Su, D. Pham, Optimisation of multi-point forming process parameters, Int. J. Adv. Manuf. Technol. 92 (5–8) (2017) 1849–1859, https://doi.org/10.1007/s00170-017-0155-y.
[4] D. Opritescu, W. Volk, Variation of components by automated driving, Int.J. Mater. Form. 9 (1) (2016) 9–19, https://doi.org/10.1007/s12289-014-1195-6.
[5] N. Shamsaei, A. Yadollahi, L. Bian, S.M. Thompson, An overview of direct laser deposition for additive manufacturing; part ii: Mechanical behavior, process parameter optimization and control, Addit. Manuf. 8 (2015) 12–35, https://doi.org/10.1016/j.addma.2015.07.002.

[6] H.R. Attar, N. Li, A. Foster, A new design guideline development strategy for aluminium alloy corners formed through cold and hot stamping processes, Mater. Des. 207 (2021) 109856, https://doi.org/10.1016/j.matdes.2021.109856.
[7] I. Dostaler, Avoiding rework in product design: evidence from the aerospace industry, Int. J. Qual. Reliab. Manage. 27 (1) (2010) 5–26, https://doi.org/10.1108/02656711011009281.
[8] V. Butenko, A. Albers, Improving the knowledge transfer from research to industry by developing demand-oriented design guidelines for fibre-reinforced plastics, Procedia CIRP 70 (2018) 41–46, https://doi.org/10.1016/j.procir.2018.03.275.
[9] R. Robertson, T.-J. Chu, R. Gerard, J.-H. Kim, M. Park, H.-G. Kim, R. Peterson, Three-dimensional fiber reinforcement shapes obtainable from flat, bidirectional fabrics without wrinkling or cutting. part 1. A single four-sided pyramid, Composites Part A 31(7) (2000) 703–715. https://doi.org/10.1016/S1359-835X(00)00013-0.
[10] P. Böhler, F. Härtel, P. Middendorf, Identification of forming limits for unidirectional carbon textiles in reality and mesoscopic simulation, Key Eng. Mater. 554–557 (2013) 423–432, https://doi.org/10.4028/www.scientific.net/KEM.554-557.423.
[11] S.G. Hancock, K.D. Potter, Inverse drape modelling—an investigation of the set of shapes that can be formed from continuous aligned woven fibre reinforcements, Composites Part A 36 (7) (2005) 947–953, https://doi.org/10.1016/j.compositesa.2004.12.001.
[12] P. Bussetta, N. Correia, Numerical forming of continuous fibre reinforced composite material: A review, Composites Part A 113 (2018) 12–31, https://doi.org/10.1016/j.compositesa.2018.07.010.
[13] A.K. Pickett, G. Creech, P. de Luca, Simplified and advanced simulation methods for prediction of fabric draping, Revue Européenne des Éléments Finis 14 (6–7) (2005) 677–691, https://doi.org/10.3166/reef.14.677-691.
[14] B. Fengler, L. Kärger, F. Henning, A. Hrymak, Multi-objective patch optimization with integrated kinematic draping simulation for continuous–discontinuous fiber-reinforced composite structures, J. Compos. Sci. 2 (2) (2018) 22, https://doi.org/10.3390/jcs2020022.
[15] P. Boisse (Ed.), Advances in composites manufacturing and process design, Woodhead publishing series in composites science and engineering, vol. 56, Woodhead Publishing, Cambridge, UK, 2015.
[16] S. Chen, L.T. Harper, A. Endruweit, N.A. Warrior, Formability optimisation of fabric preforms by controlling material draw-in through in-plane constraints, Composites Part A 76 (2015) 10–19.
[17] A.I.J. Forrester, A. Sóbester, A.J. Keane, Engineering design via surrogate modelling: A practical guide, Wiley, Chichester, 2008.
[18] H. Wang, F. Ye, L. Chen, E. Li, Sheet metal forming optimization by using surrogate modeling techniques, Chin. J. Mech. Eng. 30 (1) (2017) 22–36, https://doi.org/10.3901/CJME.2016.1020.123.
[19] M. Arian Nik, K. Fayazbakhsh, D. Pasini, L. Lessard, A comparative study of metamodeling methods for the design optimization of variable stiffness composites, Compos. Struct. 107 (2014) 494–501, https://doi.org/10.1016/j.compstruct.2013.08.023.
[20] A. Kaveh, A. Dadras Eslamlou, S.M. Javadi, N. Geran Malek, Machine learning regression approaches for predicting the ultimate buckling load of variable-stiffness composite cylinders, Acta Mech. https://doi.org/10.1007/s00707-020-02878-2.
[21] J. Pfrommer, C. Zimmerling, J. Liu, L. Kärger, F. Henning, J. Beyerer, Optimisation of manufacturing process parameters using deep neural networks as surrogate models, Procedia CIRP 72 (2018) 426–431, https://doi.org/10.1016/j.procir.2018.03.046.
[22] C. Zimmerling, P. Schindler, J. Seuffert, L. Kärger, Deep neural networks as surrogate models for time-efficient manufacturing process optimisation, ESAFORM2021 Liège/Belgium (accepted for publication), 2021.
[23] A. Hürkamp, S. Gellrich, A. Dér, C. Herrmann, K. Dröder, S. Thiede, Machine learning and simulation-based surrogate modeling for improved process chain operation, Int. J. Adv. Manuf. Technol. https://doi.org/10.1007/s00170-021-07084-5.
[24] A. Hürkamp, R. Lorenz, T. Ossowski, B.-A. Behrens, K. Dröder, Simulation-based digital twin for the manufacturing of thermoplastic composites, Procedia CIRP 100 (2021) 1–6, https://doi.org/10.1016/j.procir.2021.05.001.
[25] H. El Kadi, Modeling the mechanical behavior of fiber-reinforced polymeric composite materials using artificial neural networks—a review, Compos. Struct. 73 (1) (2006) 1–23, https://doi.org/10.1016/j.compstruct.2005.01.020.
[26] G. Ambrogio, C. Ciancio, L. Filice, F. Gagliardi, Innovative metamodelling-based process design for manufacturing: an application to incremental sheet forming, Int.J. Mater. Form. 10 (3) (2017) 279–286, https://doi.org/10.1007/s12289-015-1276-1.
[27] E. Hamouche, E.G. Loukaides, Classification and selection of sheet forming processes with machine learning, Int. J. Comput. Integr. Manuf. 31 (9) (2018) 921–932, https://doi.org/10.1080/0951192X.2018.1429668.
[28] C. Schwarz, P. Ackert, R. Mauermann, Principal component analysis and singular value decomposition used for a numerical sensitivity analysis of a complex drawn part, Int. J. Adv. Manuf. Technol. 94 (5–8) (2018) 2255–2265, https://doi.org/10.1007/s00170-017-0980-z.
[29] C. Hartmann, D. Opritescu, W. Volk, An artificial neural network approach for tool path generation in incremental sheet metal free-forming, J. Intell. Manuf. 30 (2) (2019) 757–770, https://doi.org/10.1007/s10845-016-1279-x.
[30] C. Zimmerling, C. Poppe, L. Kärger, Estimating optimum process parameters in textile draping of variable part geometries - a reinforcement learning approach, Proc. Manuf. 47 (2020) 847–854, https://doi.org/10.1016/j.promfg.2020.04.263.
[31] K. Hornik, Approximation capabilities of multilayer feedforward networks, Neural Netw. 4 (2) (1991) 251–257, https://doi.org/10.1016/0893-6080(91)90009-T.
[32] F. Chollet et al., Keras, https://keras.io (2015).
[33] R.S. Sutton, A. Barto, Reinforcement learning: An introduction, second edition, Adaptive computation and machine learning, The MIT Press, Cambridge, MA and London, 2018.
[34] Z. Zhou, X. Li, R.N. Zare, Optimizing chemical reactions with deep reinforcement learning, ACS Central Sci. 3 (12) (2017) 1337–1344, https://doi.org/10.1021/acscentsci.7b00492.
[35] J. Dornheim, L. Morand, S. Zeitvogel, T. Iraki, N. Link, D. Helm, Deep reinforcement learning methods for structure-guided processing path optimization, J. Intell. Manuf. 1 (12) (2021) 4022, https://doi.org/10.1007/s10845-021-01805-z. URL http://arxiv.org/pdf/2009.09706v4.
[36] M. Panzer, B. Bender, Deep reinforcement learning in production systems: a systematic literature review, Int. J. Prod. Res. (2021) 1–26, https://doi.org/10.1080/00207543.2021.1973138.
[37] F. Henning, L. Kärger, D. Dörr, F.J. Schirmaier, J. Seuffert, A. Bernath, Fast processing and continuous simulation of automotive structural composite components, Compos. Sci. Technol. 171 (2019) 261–279, https://doi.org/10.1016/j.compscitech.2018.12.007.
[38] L. Kärger, S. Galkin, C. Zimmerling, D. Dörr, J. Linden, A. Oeckerath, K. Wolf, Forming optimisation embedded in a cae chain to assess and enhance the structural performance of composite components, Compos. Struct. 192 (2018) 143–152, https://doi.org/10.1016/j.compstruct.2018.02.041.
[39] P. Boisse, N. Hamila, E. Vidal-Sallé, F. Dumont, Simulation of wrinkling during textile composite reinforcement forming. influence of tensile, in-plane shear and bending stiffnesses, Compos. Sci. Technol. 71 (5) (2011) 683–692, https://doi.org/10.1016/j.compscitech.2011.01.011.
[40] P. Boisse, J. Colmars, N. Hamila, N. Naouar, Q. Steer, Bending and wrinkling of composite fiber preforms and prepregs. A review and new developments in the draping simulations, Compos. Part B: Eng. 141 (2018) 234–249. https://doi.org/10.1016/j.compositesb.2017.12.061.
[41] E. Kunze, S. Galkin, R. Böhm, M. Gude, L. Kärger, The impact of draping effects on the stiffness and failure behavior of unidirectional non-crimp fabric fiber reinforced composites, Materials (Basel, Switzerland) 13(13). https://doi.org/10.3390/ma13132959.
[42] F. Nosrat Nezami, T. Gereke, C. Cherif, Active forming manipulation of composite reinforcements for the suppression of forming defects, Composites Part A 99 (2017) 94–101, https://doi.org/10.1016/j.compositesa.2017.04.011.
[43] F. Förster, F. Ballier, S. Coutandin, A. Defranceski, J. Fleischer, Manufacturing of textile preforms with an intelligent draping and gripping system, Procedia CIRP 66 (2017) 39–44, https://doi.org/10.1016/j.procir.2017.03.370.
[44] R. Jagpal, R. Butler, E.G. Loukaides, Towards flexible and defect-free forming of composites through distributed clamping, Procedia CIRP 85 (2019) 341–346, https://doi.org/10.1016/j.procir.2019.09.008.
[45] S. Chen, O. McGregor, L.T. Harper, A. Endruweit, N.A. Warrior, Optimisation of local in-plane constraining forces in double diaphragm forming, Compos. Struct. 201 (2018) 570–581, https://doi.org/10.1016/j.compstruct.2018.06.062.
[46] L. Kärger, S. Galkin, E. Kunze, M. Gude, B. Schäfer, Prediction of forming effects in ud-ncf by macroscopic forming simulation – capabilities and limitations, ESAFORM (2021), https://doi.org/10.25518/esaform21.355.
[47] E. Kunze, B. Schwarz, T. Weber, M. Müller, R. Böhm, M. Gude, Forming analysis of internal plies of multi-layer unidirectional textile preforms using projectional radiography, Procedia Manuf. 47 (2020) 17–23, https://doi.org/10.1016/j.promfg.2020.04.110.
[48] C. Poppe, D. Dörr, F. Henning, L. Kärger, Experimental and numerical investigation of the shear behaviour of infiltrated woven fabrics, Composites Part A 114 (2018) 327–337, https://doi.org/10.1016/j.compositesa.2018.08.018.
[49] C. Poppe, T. Rosenkranz, D. Dörr, L. Kärger, Comparative experimental and numerical analysis of bending behaviour of dry and low viscous infiltrated woven fabrics, Composites Part A 124 (2019) 105466, https://doi.org/10.1016/j.compositesa.2019.05.034.
[50] F.J. Schirmaier, D. Dörr, F. Henning, L. Kärger, A macroscopic approach to simulate the forming behaviour of stitched unidirectional non-crimp fabrics (ud-ncf), Composites Part A 102 (2017) 322–335, https://doi.org/10.1016/j.compositesa.2017.08.009.
[51] D. Dörr, F.J. Schirmaier, F. Henning, L. Kärger, A viscoelastic approach for modeling bending behavior in finite element forming simulation of continuously fiber reinforced composites, Composites Part A 94 (2017) 113–123, https://doi.org/10.1016/j.compositesa.2016.11.027.
[52] F. Nosrat Nezami, T. Gereke, C. Cherif, Analyses of interaction mechanisms during forming of multilayer carbon woven fabrics for composite applications, Composites Part A 84 (2016) 406–416, https://doi.org/10.1016/j.compositesa.2016.02.023.
[53] S. Haanappel, Forming of UD fibre reinforced thermoplastics: A critical evaluation of intra-ply shear, PhD thesis, University of Tewente (2013). https://doi.org/10.3990/1.9789036535014.
[54] S. Chen, A. Endruweit, L.T. Harper, N.A. Warrior, Inter-ply stitching optimisation of highly drapeable multi-ply preforms, Composites Part A 71 (2015) 144–156, https://doi.org/10.1016/j.compositesa.2015.01.016.

[55] C. Zimmerling, D. Dörr, F. Henning, L. Kärger, A machine learning assisted approach for textile formability assessment and design improvement of composite components, Composites Part A 124 (2019) 105459, https://doi.org/10.1016/j.compositesa.2019.05.027.
[56] R.K. Gupta, B. Gurumoorthy, Classification, representation, and automatic extraction of deformation features in sheet metal parts, Comput. Aided Des. 45 (11) (2013) 1469–1484, https://doi.org/10.1016/j.cad.2013.06.010.
[57] X. Guo, W. Li, F. Iorio, Convolutional neural networks for steady flow approximation, in: Krishnapuram, Shah et al. (Hg.) 2016 – Proceedings of the 22nd ACM, pp. 481–490. https://doi.org/10.1145/2939672.2939738.
[58] V. Hegde, R. Zadeh, Fusionnet: 3d object classification using multiple data representations. http://arxiv.org/pdf/1607.05695v4.
[59] 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, in: AIP Conference Proceedings, AIP Publishing, 2019, p. 020007. https://doi.org/10.1063/1.5112512.
[60] Y. Bengio, Practical recommendations for gradient-based training of deep architectures. http://arxiv.org/pdf/1206.5533v2.
[61] C. Shorten, T.M. Khoshgoftaar, A survey on image data augmentation for deep learning, J. Big Data 6(1). https://doi.org/10.1186/s40537-019-0197-0.
[62] P. Boisse, N. Hamila, E. Guzman-Maldonado, A. Madeo, G. Hivet, F. dell'Isola, The bias-extension test for the analysis of in-plane shear properties of textile composite reinforcements and prepregs: a review, Int.J. Mater. Form. 10 (4) (2017) 473–492, https://doi.org/10.1007/s12289-016-1294-7.
[63] P. Badel, S. Gauthier, E. Vidal-Sallé, P. Boisse, Rate constitutive equations for computational analyses of textile composite reinforcement mechanical behaviour during forming, Compos. Part A: Appl. Sci. Manuf. 40 (8) (2009) 997–1007, https://doi.org/10.1016/j.compositesa.2008.04.015.
[64] Y. Lockner, C. Hopmann, Induced network-based transfer learning in injection molding for process modelling and optimization with artificial neural networks, Int. J. Adv. Manuf. Technol. 112 (11–12) (2021) 3501–3513, https://doi.org/10.1007/s00170-020-06511-3.
[65] C. Zimmerling, B. Fengler, L. Kärger, Formability assessment of variable geometries using machine learning - analysis of the influence of the database, Key Engineering Materials: Proceedings of the 25th ESAFORM conference, (2022) (accepted for publiction).
[66] David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, Martin Riedmiller, Deterministic policy gradient algorithms, vol. 32, PMLR, Bejing, China, 2014, pp. 387–395. http://proceedings.mlr.press/v32/silver14.html.