Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 59-26 (2025) 109–114

Data-driven modeling and optimization of the thermoforming heating phase ⋆

Eva Masero *, **Walter Zoff** *, **Riccardo Scattolini** *

* *Department of Electronics, Information and Bioengineering, Politecnico di Milano, Milan, 20133, Italy (e-mails: {eva.masero, riccardo.scattolini}@polimi.it)*

Abstract: Thermoforming is a crucial technique in plastics manufacturing that relies on the expertise of skilled operators for effective process control. Changing operating conditions require frequent tuning of the thermoforming machine parameters to maintain the quality of the resulting product. However, this tuning often results in a waste of resources that should be minimized. Our work collects input-output data from experiments and investigates how to optimize machine parameters to maintain product quality. Specifically, we perform model identification using neural networks and solve an optimization problem for the heating phase. The results validate our approach, demonstrating its ability to obtain optimized recipe parameters that accurately replicate the desired thermal characteristics.

Keywords: Thermoforming, Model Identification, Machine Learning, Optimization, Industrial Processes.

1. INTRODUCTION

Thermoforming is a widely used manufacturing process in which a plastic sheet is heated to be shaped into a specific geometry using a mold. It consists of three main phases: heating, shaping, and cooling. The most critical phase in the thermoforming process is the heating phase, which has a large impact on the final quality of the product. Thermoforming machines have a series of adjustable parameters, known as *recipe parameters*, such as the heating power, temperature reference, heating and cooling times, which influence the characteristics of the final piece. Since the recipe parameters can be adjusted via a human-machine interface, an expert visually evaluates the quality of the final piece and decides whether to change the recipe parameters for quality improvement. Unsurprisingly, this trial-and-error approach to tuning thermoforming machines involves a waste of resources.

One of the main challenges of the thermoforming process is to understand the fundamental dynamics and synthesize them through mathematical representation due to its complex dynamics. Bergman (2011) provides the foundational principles of heat transfer, including conduction, convection, and radiation, which are crucial to understanding thermal distribution in the thermoforming process. Traditional thermoforming models are based on physics-based simulations using partial differential equations and finite element methods. For example, Ragoubi et al. (2024) propose a thermo-visco-hyper-elastic model to simulate the thermoforming process of complex geometries using ABAQUS, a widely-used software for simulating finite element analysis in engineering problems. Although traditional models provide detailed physical insights, they are computationally expensive and lack adaptability for real-time control.

From the control viewpoint, the Model Predictive Control (MPC) framework can be used to optimize thermoforming processes due to its ability to handle multivariable systems and operational constraints (Rossiter, 2017). MPC optimizes performance by predicting future outputs over a finite time horizon using a dynamic model of the process (Masero et al., 2021). For instance, Chy et al. (2011) propose an MPC approach for managing the temperature distribution of plastic sheets using infrared sensors. Moreover, Gauthier and Boulet (2009) explores a Terminal Iterative Learning Control (TILC) approach combined with Singular Value Decomposition decoupling to control the thermoforming heating phase using a linear model. Adaptive control strategies such as In-Cycle and Cycle-to-Cycle control (Rzepniewski, 2005) are also used to maintain consistent heating profiles in thermoforming by dynamically adjusting power levels based on sensor feedback, see *e.g.*, (Girard et al., 2005). Another approach, proposed by Erchiqui (2018), is a hybrid optimization that combines genetic algorithms and simulated annealing to optimize the infrared heating stage, ensuring a uniform thermal distribution and improving product quality.

Although these control approaches are effective, they have notable limitations *e.g.*, dependence on a precise physical model and high computational complexity when dealing with nonlinear systems. These limitations foster the exploration of data-driven approaches, which can learn complex nonlinear relationships directly from historical data, without relying on explicit physical models. The study of Marchal et al. (2023) proposes the use of integrated sensors in the thermoforming machine to collect data, and then a statistical model based on Design of Experiments

⋆ Financial support by the Next-Generation EU (Italian PNRR-M4 C2, Invest 1.3-D.D.1551.11-10-2022, PE00000004) for grant MICS with CUP: D43C22003120001.

10.1016/j.ifacol.2025.12.019

(DoE) is developed to predict geometry quality (minimum thickness) of produced plastic products in real time. This work demonstrates accurate predictions and shows a real-time dashboard for monitoring, reducing human dependency and improving efficiency. As pointed out, a promising future direction is the integration of prediction models with numerical simulations, with the aim of reducing tuning time and assisting operators in configuring recipe parameters.

In this context, this paper proposes a black-box prediction model for the thermoforming heating phase based on neural networks that are trained with experimental data provided by a thermoforming machine builder company. A data-driven approach is then developed to optimize the recipe parameters needed to achieve the desired heat distribution on the sheet. The proposed approach overcomes limitations of physics-based simulators, which often rely on highly complex and inflexible models that require significant computational resources, making them infeasible for real-time applications.

The rest of the paper is organized as follows. Section 2 focuses on the preparation of the dataset for training and the model identification of the thermoforming heating phase via neural networks. Section 3 formalizes the problem of optimizing the recipe parameters and details the algorithm employed. Section 4 shows the optimization results obtained. Finally, Section 5 discusses the main conclusions and some potential future lines.

2. MODEL IDENTIFICATION

This section focuses on the modeling of the thermoforming heating phase using a black-box model. This model is based on neural networks (NN), which are trained with experimental data provided by a thermoforming machine builder company. The dataset used in this work consists of 544 samples, recorded over three working days, each corresponding to a complete thermoforming cycle. Each sample collects the machine input parameters, the so-called recipe parameters, and the thermal image of the sheet at the end of the heating phase, among other machine settings. Prior to using the data to train the NN-based model, data analysis and preprocessing are needed to obtain a suitable model.

2.1 Data Preprocessing

The exploratory analysis revealed a large number of machine input parameters, many of which were constant throughout the dataset. These redundant parameters were removed, reducing the dataset to 21 relevant features. As illustrated in Fig. 1, a cyclic pattern was also observed in the thermal images for three working days.

At the beginning of each day, the sheet exhibited higher temperatures, leading to predominantly purple-colored thermal images. Afterwards, it was followed by a transient phase in which the temperature gradually decreased until it reaches a steady-state region. In this region, the thermal images remained consistently blue until the end of the working day. This recurring pattern is particularly relevant because it reveals a predictable preheating phase followed by a stable thermal regime.

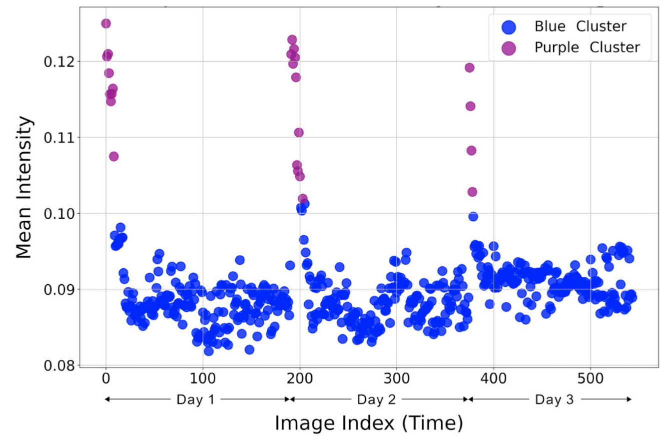

Fig. 1. Trend of mean intensity over sample index (time).

To capture this transient trend and improve model learning, the feature `NumDone` was introduced to represent the cycle number within the working day, which will allow NNs to recognize and adapt to systematic variations in thermal distribution that occur throughout the day. Additionally, K-means clustering (Chong et al., 2021) was applied to group images according to the mean intensity of their pixels, classifying as ‘Blue’ those with low mean color intensity and ‘Purple’ the rest. After analyzing and classifying the available thermal images into the two clusters, a significant imbalance was observed. The dataset was heavily skewed towards blue images, leading to issues such as biased predictions, reduced generalization, and poor performance on underrepresented classes. For this reason, an undersampling strategy with a 10:1 ratio (blue vs. purple) was employed to balance the dataset by reducing the number of samples in the majority (blue) cluster.

Further analysis of the input correlation matrix revealed that some parameters were highly correlated, with cases of multicollinearity. To address this, Principal Component Analysis (PCA) was employed. PCA is a dimensionality reduction technique that transforms correlated variables into a set of uncorrelated principal components, capturing the most significant variations in the data (Abdi and Williams, 2010). This data reduction not only resolves multicollinearity, but also reduces the complexity of the dataset. By preserving 95 % of the total variance, five principal components were selected at the end of the analysis. After these preprocessing steps, the dataset was ready for training the model.

2.2 Black-box model

The black-box model is based on a MultiLayer Perceptron (MLP), which is a fully connected NN formed by three types of layers: input, output, and hidden layers, particularly well-suited for learning nonlinear mappings from inputs to outputs. Recall that data input is the recipe parameters and output is the thermal image generated at the end of the heating phase.

As illustrated in Fig. 2, the model consists of an input layer that takes as inputs the 5 principal components extracted via PCA, followed by two hidden layers with 128 and 64 neurons, respectively. Finally, the output layer produces a vector corresponding to the total number of pixels in the

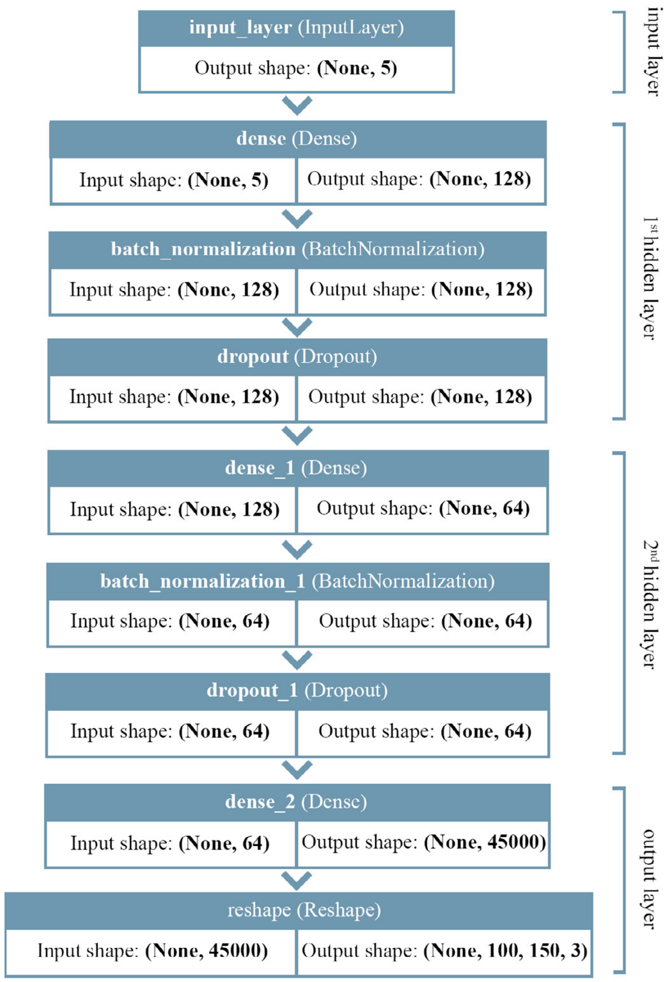

Fig. 2. Scheme of the MLP network architecture.

processed image. To preserve spatial patterns, the RGB image is multiplied by 255 to undo the normalization and restore the original intensity range, and then reshaped into the original dimensions by applying a factor of 10, enabling a direct mapping from input features to the corresponding original thermal distributions.

The hidden layers utilize the ReLU activation function:

$$f(x) = \max(0, x),$$

which enhances computational efficiency, mitigates the vanishing gradient problem, and promotes sparsity. Moreover, after each hidden layer, batch normalization (Ioffe and Szegedy, 2015) and dropout (Srivastava et al., 2014) techniques were applied, respectively, to adjust activations and mitigate data overfitting. The output layer applies a sigmoid activation function:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

which constrains predicted pixel intensities within the range [0, 1] and thus aligns with the normalized thermal image format. Finally, hyperparameter tuning such as the number of neurons, the learning rate, the batch size, and the number of epochs, was conducted using Bayesian optimization, achieving a well-balanced trade-off between training and validation loss.

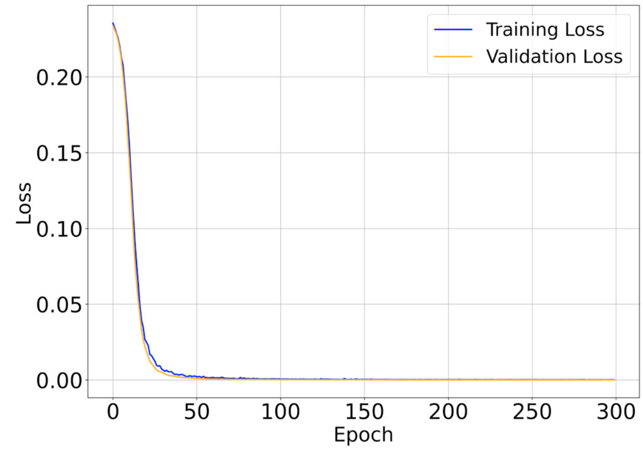

Fig. 3. Training and validation loss curves over 300 epochs.

2.3 Training of the model

To prepare the training and validation datasets, the PCA-transformed features (input) were combined with the corresponding thermal images (output). The samples were shuffled to prevent the model from learning unintended temporal sequences or trends unrelated to the heating phase (Bengio, 2012). The dataset was then split into 70 % for training the model and 30 % for model evaluation during the hyperparameter tuning, while preserving the 10:1 ratio between blue and purple images to maintain class balance. Due to the limited amount of data, we split the available data into training and validation sets only.

The model was trained using the Adam optimizer (Kinga et al., 2015) with a learning rate of 0.001, a batch size of 16, for 300 epochs, and a dropout rate of 0.3. The evolutions of the training and validation loss over 300 epochs are shown in Fig. 3. The loss decreases rapidly during the initial training phase, indicating efficient learning of the thermal patterns. As training progresses, the loss stabilizes, suggesting that the model has converged. The close alignment between the training and validation loss curves indicates a good generalization, with no significant overfitting observed.

The evaluation process involved comparing the thermal images predicted by the trained model with the corresponding ground truth images from the validation set. Since the validation set consists of samples not used for training, this approach ensures an unbiased assessment of the generalizability of the model to accurately reconstruct thermal distributions from new input data. The effectiveness of the trained model for predicting thermal distributions was evaluated using the Mean Squared Error (MSE) (Wang and Bovik, 2009). This MSE value quantifies the average squared difference between predicted pixel intensities $\hat{y}_i$ and actual values y_i, for a total of N pixels:

$$\mathrm{MSE} = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

Fig. 4 reports the distribution of MSE values in the validation set, providing a quantitative assessment of the model accuracy. In particular, the mean MSE is 0.00015, which indicates a good predictive performance of the

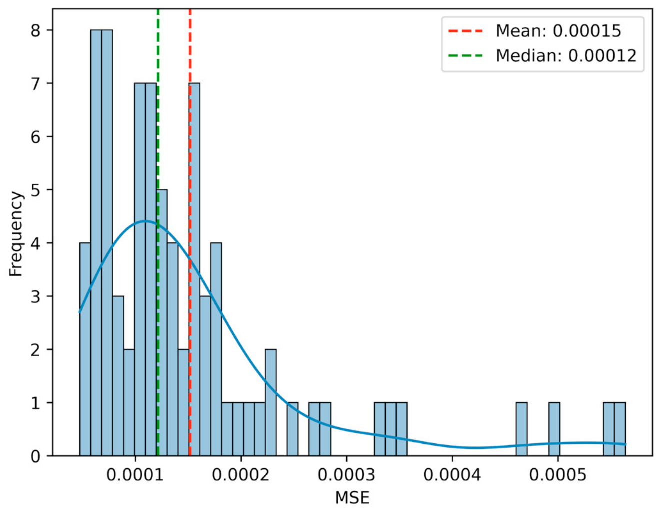

Fig. 4. Distribution of MSE on the validation dataset.

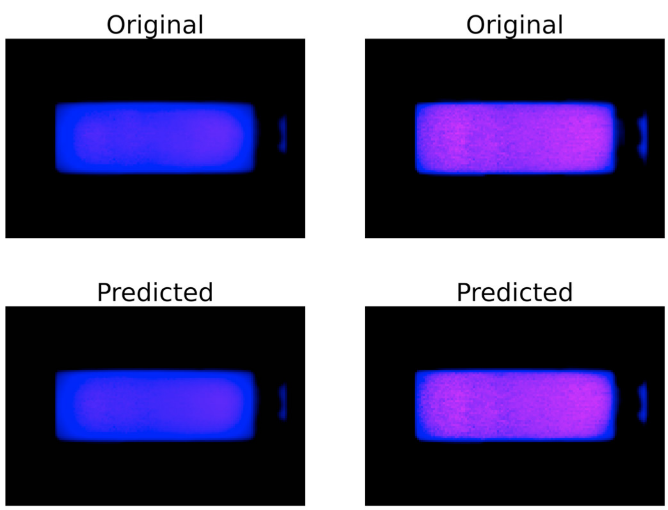

Fig. 5. Comparison between predicted and original thermal images: Blue images (left-hand side) and purple images (right-hand side).

model (the lower, the better), as MSE represents the deviation between predicted and actual pixel intensities.

Finally, a direct visual comparison between the predicted and actual thermal images is also presented. As shown in Fig. 5, the reconstructed thermal images closely resemble the ground truth, demonstrating the model's ability to capture the spatial temperature distribution with high fidelity. The results confirm the practical applicability of the model for reproducing the heating phase of the thermoforming process.

3. RECIPE OPTIMIZATION

The objective is to determine the optimal input parameters that produce a thermal distribution on the sheet that matches a predefined target image, which in this case is characterized by a uniform medium-temperature distribution across the plastic sheet. The predictive capabilities of the previously trained neural network serve as a black-box model for estimating the resulting thermal distribution on the heated sheet, given a set of input parameters.

In order to ensure that the predicted thermal distribution closely matches the target, the optimization problem is formulated to minimize the quadratic cost function:

$$J = \left(T_{\text{pred}}(x) - T_{\text{target}}\right)^2 + \left(\delta_{\text{pred}}(x) - \delta_{\text{target}}\right)^2 \qquad (1)$$

where $x \in \mathbb{R}^{n_x}$ with $n_x = 22$ represents the input vector that contains the recipe parameters to optimize. Here, $T_{\text{pred}}(x)$ denotes the predicted mean temperature of the thermal image generated by the trained NN model given the input configuration x, while T_{target} represents the target mean temperature associated with the desired thermal distribution. The cost function also accounts for the uniform heat distribution by considering the temperature standard deviation: $\delta_{\text{pred}}(x)$ is the standard deviation of the predicted thermal distribution and δ_{target} represents the desired uniformity. It is worth mentioning that the black regions from the predicted and target RGB images (see Fig. 5) were excluded before computing the mean and standard deviation.

Note that if complex plastic shapes need to be generated, a more sophisticated cost function would be required. For example, one that accounts not only for the errors in the mean temperature and standard deviations but also for the sum of the temperature errors across the entire thermal map, evaluating discrepancies at each individual point.

To ensure realistic and practical parameter adjustments, some constraints are included in the optimization problem. Only the parameters of $x = [p,\ q]$ that directly influence the heating phase are allowed to vary freely, as they have the most significant impact on the thermal distribution. Specifically, seven free parameters p_i, are restricted within a range of $\pm 20\,\%$ of their historical mean m_i:

$$0.8\ m_i \leq p_i \leq 1.2\ m_i, \quad i = 1, 2, \ldots, 7.$$

The remaining input parameters, q_j, are restricted to their historical values Φ_j to maintain consistency with feasible configurations:

$$q_j \in \Phi_j \ \ , \quad j = 1, 2, \ldots, 15.$$

This approach ensures that the optimization process explores only physically meaningful solutions.

3.1 Genetic algorithm

A genetic algorithm (GA) is employed to optimize the heating phase due to its ability to explore complex, high-dimensional search spaces (Holland, 1992). Unlike gradient-based methods, GAs do not require a differentiable or continuous cost function, making them ideal for non-linear and discrete optimization problems. Algorithm 1 outlines the workflow of the genetic algorithm, summarizing the key steps of the optimization process.

The algorithm begins by randomly initializing a population of candidate solutions. In this work, the population size was set to `POP_SIZE` = 70 individuals to allow sufficient exploration of the search space while maintaining computational efficiency. To enhance the quality of the initial population, the algorithm incorporates a strategy that includes a high-quality individual extracted from historical data. This fact ensures that at least one promising solution

Algorithm 1 Genetic Algorithm for Optimization

Require: Population size POP_SIZE, mutation probability MUTPB, crossover probability CXPB, max. generations N_GEN.

Ensure: Optimized set of input parameters.

1: Initialize a population of POP_SIZE individuals:
- Include one high-quality individual from dataset.
- Initialize free parameters within $\pm 20\,\%$ of historical means.
- Randomly sample constrained parameters from dataset.

2: **for** generation $g = 1$ to N_GEN **do**

3: Evaluate the fitness of each individual by minimizing the cost function J (1), which incorporates the trained NN model to compute the predicted temperature and its associated standard deviation.

4: Select parents using tournament selection.

5: Apply Blend Crossover with probability CXPB.

6: Apply mutation:
- Apply Gaussian mutation to the free parameters.
- Randomly sample constrained parameters from historical data with probability MUTPB.

7: Replace the least fit individuals in the population with offspring.

8: **end for**

9: **return** Best solution found after N_GEN generations.

is present from the beginning. The remaining individuals are randomly generated within the allowed parameter ranges, promoting diversity and broad exploration of the search space. This approach provides a balance between leveraging historical knowledge and exploring new regions of the search space.

Once the population is initialized, the fitness of each individual is evaluated using the cost function J, which measures the deviation from the target mean temperature and standard deviation. By minimizing J, the algorithm guides the population toward solutions that achieve the desired thermal distribution. The parent selection is performed using a tournament selection method. In each tournament, a group of individuals is randomly chosen from the population, and the one with the lowest cost function value is selected as the parent. This process is repeated until a pool of parents is selected to produce the next generation. Tournament selection ensures that individuals with lower cost function values are more likely to be chosen as parents while maintaining diversity within the population.

Afterwards, crossover is applied to the selected parents to recombine their genes and produce offspring with inherited characteristics from both parents. In particular, a Blend Crossover technique is used, which combines the genes of two parents with a random proportion. This approach allows the offspring to explore new regions of the solution space by inheriting a mixture of characteristics from both parents. The probability of crossover is set to $\texttt{CXPB} = 0.5$, ensuring a balanced exploration of the search space while preserving the best solutions.

Mutation is introduced to maintain diversity within the population and to prevent premature convergence. In this study, a Gaussian mutation is used to perturb the values of the free parameters within their allowed range. A small

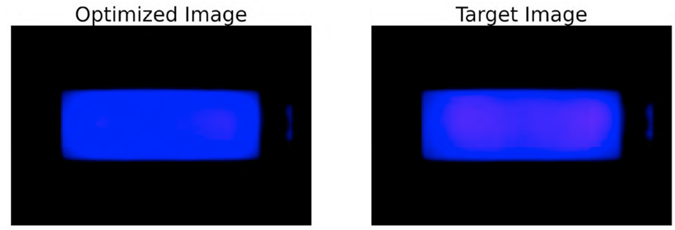

Fig. 6. Graphic comparison between the optimized image and the target image.

random value drawn from a normal distribution is added to each free parameter, ensuring that the mutated solutions remain realistic. For the constrained parameters, mutation involves randomly selecting a new historical value from the dataset to ensure consistency with historically valid configurations. The probability of mutation was set to $\texttt{MUTPB} = 0.2$, allowing for controlled exploration of the solution space while maintaining population diversity. This approach preserves the authenticity of the input configurations and ensures that the generated solutions remain consistent with the actual system's behavior.

The fitness of the new generation is then evaluated using the same cost function J, and the population is updated by replacing the current generation with the newly produced offspring. This iterative process of selection, crossover, mutation, and fitness evaluation continues until the termination criterion is met. In this study, the algorithm terminates after a maximum of $\texttt{N\_GEN} = 100$ generations, ensuring a sufficient number of iterations for convergence while maintaining computational efficiency.

4. OPTIMIZATION RESULTS

In Fig. 6, we illustrate a graphical comparison between the image predicted using the optimized input parameters obtained from Algorithm 1 and the target thermal image. The GA-based optimization yields a cost $J = 2.77 \cdot 10^{-4}$. Although minor discrepancies remain, the overall thermal distribution is preserved, validating the optimization approach. An increase in optimization time or a refined hyperparameter tuning could further enhance accuracy. However, a key aspect of this optimization is its ability to provide rapid solutions, making it suitable for industrial applications where computational efficiency is crucial. Rather than aiming for a perfect replication of the target image, the optimization process provides a well-initialized, near-optimal recipe, offering a starting point for designing the input parameters.

To complement Fig. 6, a comparison of the original input parameters with respect to the optimized ones is shown in Fig. 7. The results indicate that the genetic algorithm effectively adjusted key parameters while maintaining values close to historical configurations. This suggests that the optimization process successfully navigated the search space, leading to a solution that improves the thermal distribution without excessively deviating from the feasible process conditions.

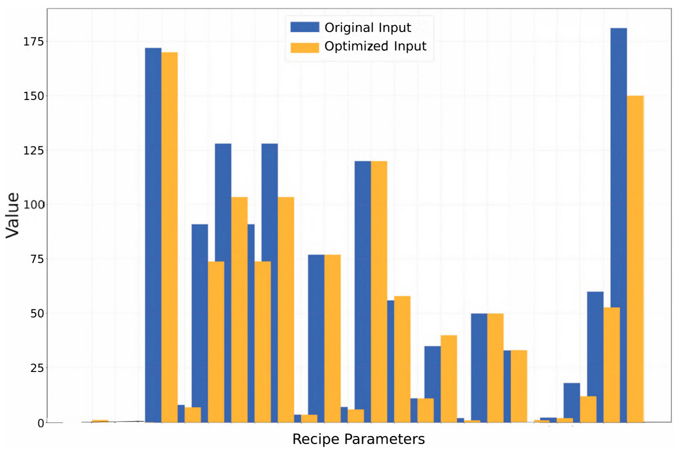

Fig. 7. Comparison between the original and optimized values of the recipe input parameters.

5. CONCLUSIONS

This work proposes a data-driven framework for optimizing the heating phase in thermoforming. Using a predictive model based on neural networks, the methodology allows for efficient determination of machine input settings that achieve the desired thermal distribution. Unlike traditional physics-based simulators, which require complex and computationally expensive models, the proposed approach relies solely on empirical real data provided by a thermoforming machine builder company. Although this fact inherently introduces certain limitations, mainly related to the quantity, quality, and variability of the collected data, the data-driven nature of the approach ensures a high degree of generalization and adaptability to similar thermoforming machines and operational conditions. Finally, the simulated results demonstrate that the trained NN model accurately predicts thermal distributions, providing a foundation for real-time optimization. The efficiency of the proposed approach, combined with its low computational requirements, enables direct applicability in industry without the need for high-performance computing resources.

A promising direction for future development is the integration of the thermoforming machine with a dedicated software platform capable of embedding the optimization tools for the heating phase directly into the production environment. Such a system would autonomously adapt to specific operating conditions, taking into account variations in process parameters and material properties. By providing real-time support to the operator, it could enhance process efficiency, improve thermal uniformity, and ultimately contribute to higher manufacturing quality and reproducibility.

REFERENCES

Abdi, H. and Williams, L.J. (2010). Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4), 433–459.

Bengio, Y. (2012). Practical recommendations for gradient-based training of deep architectures. *arXiv preprint arXiv:1206.5533*.

Bergman, T.L. (2011). *Fundamentals of heat and mass transfer*. John Wiley & Sons.

Chong, B. et al. (2021). K-means clustering algorithm: a brief review. *Academic Journal of Computing & Information Science*, 4(5), 37–40.

Chy, M.M.I., Boulet, B., and Haidar, A. (2011). A model predictive controller of plastic sheet temperature for a thermoforming process. In *Proceedings of the 2011 American control conference*, 4410–4415. IEEE.

Erchiqui, F. (2018). Application of genetic and simulated annealing algorithms for optimization of infrared heating stage in thermoforming process. *Applied Thermal Engineering*, 128, 1263–1272.

Gauthier, G. and Boulet, B. (2009). Terminal iterative learning control design with singular value decomposition decoupling for thermoforming ovens. In *2009 American Control Conference*, 1640–1645. IEEE.

Girard, P., di Raddo, R., Thomson, V., and Boulet, B. (2005). Advanced in-cycle and cycle-to cycle on-line adaptive control for thermoforming of large thermoplastic sheets. Technical report, SAE Technical Paper.

Holland, J.H. (1992). *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. MIT Press.

Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *32nd International Conference on Machine Learning (ICML)*.

Kinga, D., Adam, J.B., et al. (2015). A method for stochastic optimization. In *International conference on learning representations (ICLR)*, volume 5. San Diego, California.

Marchal, N., Ducloud, G., Agazzi, A., and Le Goff, R. (2023). Data-based model applied to thermoforming process control. *The International Journal of Advanced Manufacturing Technology*, 129(11), 5347–5358.

Masero, E., Maestre, J.M., Ferramosca, A., Francisco, M., and Camacho, E.F. (2021). Robust coalitional model predictive control with predicted topology transitions. *IEEE Transactions on Control of Network Systems*, 8(4), 1869–1880.

Ragoubi, A., Ducloud, G., Agazzi, A., Dewailly, P., and Le Goff, R. (2024). Modeling the thermoforming process of a complex geometry based on a thermo-visco-hyperelastic model. *Journal of Manufacturing and Materials Processing*, 8(1), 33.

Rossiter, J.A. (2017). *Model-based predictive control: a practical approach*. CRC press.

Rzepniewski, A.K. (2005). *Cycle-to-cycle control of multiple input-multiple output manufacturing processes*. Ph.D. thesis, Massachusetts Institute of Technology.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*.

Wang, Z. and Bovik, A.C. (2009). Mean squared error: Love it or leave it? a new look at signal fidelity measures. *IEEE Signal Processing Magazine*.