Received: 7 February 2024 | Revised: 22 April 2024 | Accepted: 27 April 2024

DOI: 10.1002/app.55693

RESEARCH ARTICLE

Applied Polymer Science WILEY

Machine learning investigation of polylactic acid bead foam extrusion

Karim Ali Shah[1] | **Christian Brütting**[1] | **Rodrigo Q. Albuquerque**[1] | **Holger Ruckdäschel**[1,2]

[1]Department of Polymer Engineering, University of Bayreuth, Bayreuth, Germany

[2]Neue Materialien Bayreuth GmbH, Bayreuth, Germany

Correspondence
Holger Ruckdäschel, Department of Polymer Engineering, University of Bayreuth, Universitätsstrasse 30, Bayreuth 95447, Germany.
Email: holger.ruckdaeschel@uni-bayreuth.de; ruckdaeschel@uni-bayreuth.de

Funding information
German Research Foundation (DFG), Grant/Award Number: RU 2586/5-1; Bayerischen Staatsministerium für Wissenschaft und Kunst, Grant/Award Number: F.2-M7426.10.2.1/4/16

Abstract

This study employs machine learning algorithms to analyze the bead foam extrusion process and to assess the impact of processing parameters, specifically focusing on their effects on bead foam density and melt pressure in under water granulation (UWG) for polylactic acid (PLA). These interrelated parameters, influenced by processing parameters such as temperature, screw speed, and blowing agent, possess challenges for traditional empirical methods to capture. The key factors that significantly impact the prediction of melt pressure in UWG are blowing agent, injector pressure, temperature in B-extruder and die size. Likewise, essential parameters for predicting bead foam density comprise blowing agent, injector pressure, temperature in B-extruder, die plate temperature, melt temperature in B-extruder, and melt pressure in B-extruder. Machine learning (ML) models were employed to forecast bead foam density and melt pressure in UWG using various processing parameters in PLA bead foam extrusion. The random forest model achieved a high coefficient of determination R^2 score of 0.96 for predicting melt pressure in UWG. Additionally, the decision tree model demonstrated effective predictions for bead density, with the R^2 score: 0.81. These ML models can be applied to diverse materials, leading to more sustainable, efficient processes for bead foam extrusion.

KEYWORDS

bead foams, foam extrusion, machine learning, PLA, twin-screw extruder

1 | INTRODUCTION

Polymer foams are widely used in today's materials industry due to many application in the modern material world, due to their enhanced mechanical properties, their ability to absorb energy, lightweight nature, and outstanding insulating and cushioning properties.[1] Bead foaming has evolved into a well-established manufacturing pathway for the production of foams. This innovative technology facilitates not only the substantial expansion of foams through extrusion foaming but also accommodates the intricate geometrical complexities inherent in the resulting parts. New polymeric materials are vital in the search for sustainable solutions, as they hold the key to addressing a lot of challenges across diverse industries. The desirable features of polymeric

materials grow complex in the quest for improved performance and environmental friendliness. These materials should exhibit a harmonious blend of mechanical strength, flexibility, and durability, catering to the specific needs of various applications. Moreover, incorporating recyclability and biodegradability is essential to minimize the environmental footprint of these materials. The development of new polymeric materials with the desired properties requires interdisciplinary collaboration, drawing insights from chemistry, materials science, engineering, and environmental science.[2,3]

Polymeric materials, with properties tailored to meet specific needs, are crucial for sustainable solutions across industries. Among these materials, polylactide acid (PLA) stands out as a promising condidate.[2,3] PLA is a thermoplastic polymer that is biodegradable and biobased which is derived from renewable resources like corn starch or sugarcane.[4] PLA's desirable properties contribute significantly to its role in sustainable material solutions. Its biodegradability ensures a reduced environmental impact, making it an attractive alternative to traditional petroleum-based plastics. PLA also exhibits good mechanical properties, allowing it to be used in various applications that require both strength and flexibility.[5]

Machine learning (ML) is very promising branch of artificial intelligence (AI). ML encompasses diverse definitions that have evolved over recent decades. Simply, it refers to a category of AI that enhances software or system processes through continuous learning from experience, without the need for explicit programming. According to Alpaydin,[6] ML involves programming computers to optimize performance criteria based on sample data or past experiences. Technically, ML techniques are classified based on the types of learning processes they undertake. ML was initially categorized into two types: supervised and unsupervised learning. However, upon closer examination, it can be further subdivided into four distinct types: supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning.[7]

Furthermore, each of these types involves various algorithms tailored to address specific problems based on the task's nature, problem type, and statistical data.

ML is gaining significant importance in modern research within the field of material science.[8–12] In recent decades, there has been a significant increase in the practical applications of ML, which has been followed by several significant advancements in the underlying algorithms and technique. Its application extends far beyond traditional approaches, providing innovative solutions and insights that were previously challenging to attain. Recent study on the utilization of ML in the material design, identification, modeling, analysis, and characterization of polymers was published.[13] In polymer science, the integration of ML models has proven to be a formidable asset, particularly in the analysis and optimization of various polymer processes,[14,15] including the intricate domain of polymer bead foaming.[16,17] Researchers have increasingly turned to ML algorithms to gain deeper insights into the complexities of these processes, and the exploration of such methodologies is particularly pronounced in the context of PLA. In the study by Albuquerque et al.,[18] the investigation of low-density PLA batch foams in the autoclave through machine learning was thoroughly explored, and achieved the mean absolute error of 30 kg m^{-3} and a coefficient of determination (R^2) of 0.94. Another ML model using random forest (RF) and principal component analysis (PCA) for the prediction of the mechanical properties of extruded PLA using an instrumental slit die, has been studied by Mulrennan et al.[19]

However, the ML offers high potential but no literature on PLA bead foam is available to our knowledge. The absence of ML studies in the literature for PLA bead foam extrusion underscores the aim of the present research initiative. The main goal of this study is to fill a crucial gap by employing ML techniques to uncover novel insights into the processes involved in PLA bead foam extrusion and to investigate and predict the influence of various processing parameters on the properties of PLA bead foams during the extrusion process.

2 | MATERIALS AND METHODS

2.1 | Materials

In this study, PLA, specifically Ingeo 2003D (NatureWorks Ltd., Minnetonka, MN), was utilized. This is a high-molecular-weight extrusion-grade material with a D-content approximately equal to 4.3% and a melting point of 150°C.[4]

2.2 | Bead foam extrusion

Extrusion is a common process in the manufacturing of bead foams. This involves carefully controlled extrusion foaming of polymers, like PLA in this study. In foam extrusion, the material is melted in the first A-extruder, which is a twin-screw extruder (L/D = 42 with a screw diameter of 25 mm), and CO_2 is injected as a blowing agent into the molten material. In this section the temperature was keep constant through all trials at 200°C The twin-screw then homogenizes the melt and CO_2 to insure uniform foaming. Subsequently, the molten

material proceeds to the B-extruder, a single-screw extruder (L/D = 30 with a screw diameter of 45 mm), where the melt is cooled down. This decrease in temperature leads to an increase in pressure in the B-extruder. The material then moves to the underwater granulation (UWG) through the die, where a high-pressure drop occurs, leading to expansion or foaming. In the UWG, the foam strand is cut into beads by a knife under a defined water pressure. Temperature, screw speed, CO_2 content, and other processing parameters affect the bead foam density and melt pressure in UWG in our case. A schematic diagram of the bead foam extrusion is shown in Figure 1. The processing parameters were obtained by the machine setup and the density was measured based on the Archimedes principle, consistent with other publications.[20]

2.3 | ML models

In this study, we extensively utilized a diverse set of supervised machine learning algorithms designed specifically for regression tasks. Regression models explain the association between a set of predictor variables (x) and one or more responses ($\hat{y}$). Typically, these models or equations take on a linear form.

$$E(\hat{y}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p \quad (1)$$

where β are coefficients to be determined from data.

The selection of these models is grounded in their effectiveness in learning from labeled datasets to provide accurate predictions for variables such as the melt pressure in UWG and bead foam density. Notable algorithms include decision tree regressor (DT), random forest regressor (RF), lasso regressor (LASSO), gradient boosting regressor (GB), and support vector regressor (SVR) with linear regression (LR) as reference model. All models were implemented using Python libraries (e.g., NumPy, scikit-learn)[a,b,c] within the Jupyter notebook environment. The code was written in Python 3 using Jupyter Notebook.

We screened a number of machine learning regression models before selecting the best one to use as the most appropriate model. The performance of the regression models does not only depend on the quality and quantity of data but are highly influenced by the ML algorithm we are using for the data.[21] In this study, we

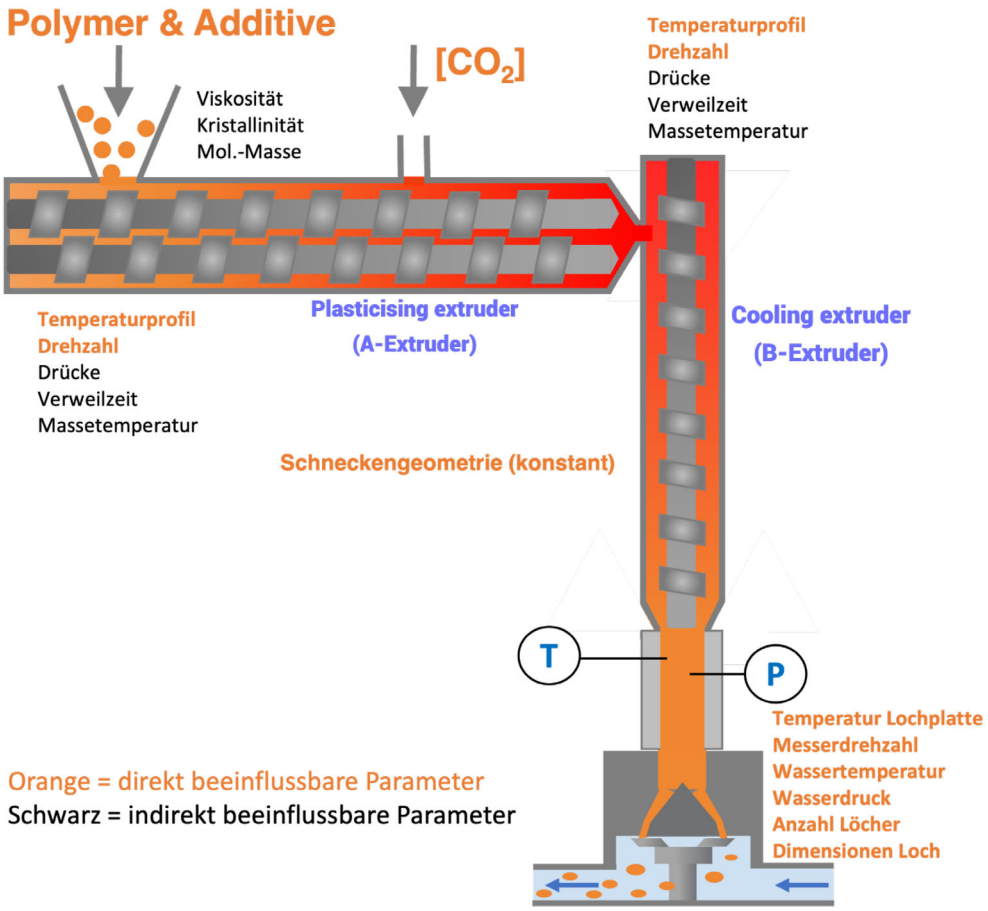

FIGURE 1 Schematic description of bead foam extrusion mechanism, highlighting the key processes involved (Table 1). [Color figure can be viewed at wileyonlinelibrary.com]

TABLE 1 Overview of varied process parameters.

Parameter	Units	Min-value	Max-value
CO_2-content	%	0	6
Water temperature	°C	35	55
Knife cutting speed	U/min	2000	4500
Water pressure	bar	0	10
Die hole (Diameter)	mm	2 × 1,4	1 × 2,8
Die plate temperature	°C	160	190

used data from 71 bead foam extrusion trails. The data are used to train four different machine learning models to predict two target properties: melt pressure in UWG and bead foam density. An 80:20 split of the total dataset was made into training and test sets, respectively. We next assessed the models' performance metrics in predicting unknown data using the 20% set aside for the test set. This evaluation allows us to evaluate while contrasting the effectiveness of each model in making predictions on new, unseen data. We also used a 5-fold cross validation technique to ascertain the optimal hyperparameters for all models. A brief description of each regression model is provided below.

2.3.1 | Decision tree regressor

For regression problems, the DT Regressor is a versatile algorithm for predicting the target.[22] A decision tree's basic concept is to divide a difficult decision into several easier ones, which may result in a solution that is easier to understand. The subset to be assigned is referred to as the target variable in the decision tree technique, whilst the data characteristics act as predictor variables. The non-parametric algorithm effectively manages large, intricate datasets without requiring a sophisticated parametric framework. A decision tree is a model that resembles a tree in which every internal node denotes a test on a particular feature, every branch denotes the test's result, and every leaf offers a prediction. The regression task's mean square error is minimized by finding the best split (best variable and threshold value) at each node. Let $\mathbf{X}$ be the input features matrix and $\mathbf{y}$ be the target vector. The DT model recursively partitions the dataset into subsets $X_1, X_2, \ldots, X_m$ based on the optimal split criteria. At each node n, the algorithm minimizes the mean squared error (MSE) to determine the split:

$$F(n) = \frac{1}{|n|}\sum_{i \in n}(y_i - \bar{y}_n)^2 \quad (2)$$

where $|n|$ is the number of samples in node n, y_i is the target value of sample i, and $\bar{y}_n$ is the mean target value in node n.

2.3.2 | Random forest regressor

An ensemble of uncorrelated tree predictors makes up a random forest, where each tree is dependent on a random vector's values.[23] This vector has an identical distribution among all the trees in the prediction and is sampled separately for each tree. Random features are chosen for every tree during the induction phase, adding a bit of variation to each unique model in the ensemble. The ensemble's predictions are combined to determine the RF's prediction.[24] In regression tasks, the predictions are averaged. This ensemble strategy reduces over-fitting and captures a wider range of patterns in the data, improving the model's resilience and capacity for generalization. The average of the forecasts made by each decision tree may be used to represent the Random Forest prediction.

$$F(\mathbf{x}) = \frac{1}{B}\sum_{b=1}^{B} f_b(\mathbf{x}) \quad (3)$$

Where $F(\mathbf{x})$ is the overall prediction, B is the total number of trees in the forest, and $f_b(\mathbf{x})$ represents the prediction of the b-th tree for input features $\mathbf{x}$.

In a way similar to the decision tree regressor, the splitting rule at each internal node is established by randomly selecting a feature and threshold. By maximizing each decision tree's parameters to reduce the mean square error, the random forest regressor is trained. By minimizing overfitting and identifying a broader variety of patterns in the data, the average of all decision trees yields the final prediction, which enhances prediction performance, robustness, and generalization.

2.3.3 | Linear regression

Linear regression is a statistical technique used a lot in modeling the relationship between a target or response variable and one or more predictors or features. The linear regression model is one of the easy model to build, as by default the model has no regularization term. The linear regression equation is given by:

$$Y = \beta_0 + \beta_1 \cdot X_1 + \beta_2 \cdot X_2 + \ldots + \beta_n \cdot X_n + \epsilon \quad (4)$$

Here, $X_1, X_2, ..., X_n$ are the predictors, and $\beta_0, \beta_1, ..., \beta_n$ are the corresponding coefficients.

The primary objective of linear regression is to find the value of the coefficients that minimize the sum of the square differences between the observed values of the dependent variable and the value predicted.

2.3.4 | LASSO regressor

Lasso regression with L1 regularization is a linear regression approach in which a penalty term is added based on the absolute values of the regression coefficients.[18] By setting certain coefficients to absolutely zero, this regularization technique encourages sparse models and works well for feature selection. The LASSO minimize the objective function:

$$\hat{\beta}_0, \hat{\beta}_{\text{lasso}} = \arg\min_{\beta_0, \beta} \left\{ \sum_{i=1}^{N} \left(Y_i - \beta_0 + \beta^T X_i\right)^2 + \lambda \sum_{j=1}^{P} |\beta_j| \right\} \tag{5}$$

Where, β_0 is the intercept term, β is the vector of coefficients, X_i is the feature vector for the i-th sample, Y_i is the experimental value for the i-th sample, and λ is the regularization parameter, controlling the strength of the regularization. Ultimately, LASSO improves the interpretability and effectiveness of the modeling process by providing accurate predictions and serving as a useful tool for automated feature selection.

2.3.5 | Gradient boosting regressor

Gradient Boosting Regression is a powerful ensemble learning method that builds a predictive model by combining the strengths of several weak learners, frequently represented as decision trees. The fundamental idea is to fit novel models iteratively to the residuals of predictions made by previous models. The performance of the model as a whole is gradually improved through this repeated refining process. The model is the sum of all the weak learners,

$$F(x) = F_0(x) + \epsilon \sum_{m=1}^{M} \gamma_m h_m(x) \tag{6}$$

Where, $F(x)$ is the current approximation of the target function. $h_m(x)$ is a weak learner to be added to improve the approximation and ϵ is the learning rate and $m = 1$ to M (number of boosting rounds).

This iterative, optimization-based method makes sure that every new weak learner is designed with the intention of correcting errors found in the ensemble's overall predictions. GBR systematically improves the model's ability to detect subtle patterns and subtleties in the data by giving priority to the reduction of residuals. The resultant ensemble not only yields robust and precise predictions but also demonstrates exceptional adaptability to intricate relationships inherent in real-world datasets. The resulting ensemble is able to adapt very well to complex interactions found in real-world datasets, while still producing strong and accurate predictions.

2.3.6 | Support vector regressor

For regression problems, SVR is a potent machine learning technique that works differently from conventional regression techniques. The decision boundaries around certain data points called support vectors, which can be defined by the features in the dataset are optimized by SVR. SVR aims to find the optimal hypertube that best fits the data points while minimizing prediction errors. Epsilon (ϵ), a parameter introduced by SVR, is the margin that represents the model's error tolerance and is often referred to as the ϵ-insensitive hypertube.

$$y = f(x) = \langle w, x \rangle + b \tag{7}$$

where:

- y is the predicted output,
- x represents the input features,
- w is the weight vector,
- $\langle w, x \rangle$ denotes the dot product between w and x,
- b is the bias term.

The training of SVR involves finding the optimal values for w and b while considering the tube defined by ϵ. SVR seeks to minimize the following objective function:

$$\min_{w,b} \frac{1}{2} \| w \|^2 + C \sum_{i=1}^{n} \left(\max(0, \| f(x_i) - y_i \| - \epsilon)\right)^2 \tag{8}$$

In this case, the regularization value C strikes a compromise between obtaining a narrow margin and a low training error. SVR is an adaptable technique that works well for a variety of regression problems. Kernel-based

TABLE 2 Statistical measures of different ML models for predicting the melt pressure in UWG for the test set.

Model	R^2 score	MAE	RMSE
Decision Tree (DT)	0.861	1.533	2.090
Random Forest (RF)	0.964	0.851	1.071
Gradient Boosting (GBR)	0.919	1.140	1.546
Lasso Regressor (LASSO)	0.933	1.096	1.396

TABLE 3 Statistical measures of different ML models for predicting the bead density for the test set.

Model	R^2-score	MAE	RMSE
Decision Tree (DT)	0.835	0.185	0.225
Random Forest (RF)	0.682	0.529	0.579
Gradient Boosting (GBR)	0.632	0.241	0.337
Support Vector Regressor (SVR)	0.543	0.576	0.716

SVR is excellent at describing non-linear correlations in data.

2.3.7 | Performance metrics

To analyze and interpret the results of ML, it is always important to present the value of the error metrics to show the effectiveness of the respective model. The statistical error is calculated by using the following mathematical formulation or equations 9, 10, and 11, that is, the coefficient of determination (R^2 score), the mean absolute error (MAE), and the root mean squared error (RMSE).

MAE is define as:

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{9}$$

where n represent the samples, and y_i and $\hat{y}_i$ denote the true predicted target properties, respectively, for sample i. RMSE is expressed as:

$$RMSE = \left(\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2\right)^{1/2} \tag{10}$$

The R^2 score quantifying the ratio of explained variance to total variance in a regression model, is defined by:

$$R^2(y,\hat{y}) = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{11}$$

In this equation, $\bar{y}$ represent the average target property.

3 | RESULTS AND DISCUSSION

This work employed four distinct regression ML models to develop a prediction model for the bead foam density and the melt pressure in UWG in the bead foam extrusion, which were extensively discussed earlier in the section 2.3. Linear regression model served as the reference model. The linear regression model cannot captured the relationship between the target and feature, yield a notably negative R^2 score. For getting deep insight, we explore more advanced model. Tables 2 and 3 presents statistical measurements for all machine learning models that predict melt pressure and bead foam density. Three key metrics were used to thoroughly assess the models: the R^2 score, the MAE, and the RMSE. These metrics provide information on the precision and accuracy of each model's predictions.

1. R^2 score is the percentage of variance in the target variable (melt pressure) that can be accounted for by each of the models. A value that is near to 1.0 indicates a better fit.
2. MAE is the average size of error between the expected and actual values. Better forecast accuracy is shown by lower MAE values.
3. RMSE quantifies the overall magnitude of errors, giving more weight to larger errors. Lower RMSE values indicate better precision in predictions.

Also, the graphical representation of all the performance metrics are shown in Figure 4. The Figure illustrate the comparison of all the error metrics for both target properties, that is, melt pressure in UWG and bead foam density.

3.1 | Correlation analysis of processing parameters

A correlation analysis serves as a statistical tool to quantitatively assess the direction and strength of a linear relationship between variables. In numerical terms, this association is typically quantified by a decimal value referred to as the correlation coefficient, called Pearson correlation coefficient. In our study on PLA bead foam extrusion, we have examined various processing

parameters, including extrusion temperature, pressure, screw speed, and die design. The correlation matrix, visualized in Figure 2, offers a comprehensive overview of the pairwise correlations between these parameters. The visual representation in the heatmap provides a quick and intuitive understanding of how each processing parameter relates to each others. The range of Pearson correlation is $-1 < \text{corr} < 1$. This range indicates that a correlation coefficient close to 1 signifies a robust positive linear relationship, while a coefficient near -1 signifies a strong negative correlation. Values close to 0 suggest a weak or negligible linear relationship. This statistical approach aids in identifying potential interdependencies between processing parameters, which assist the

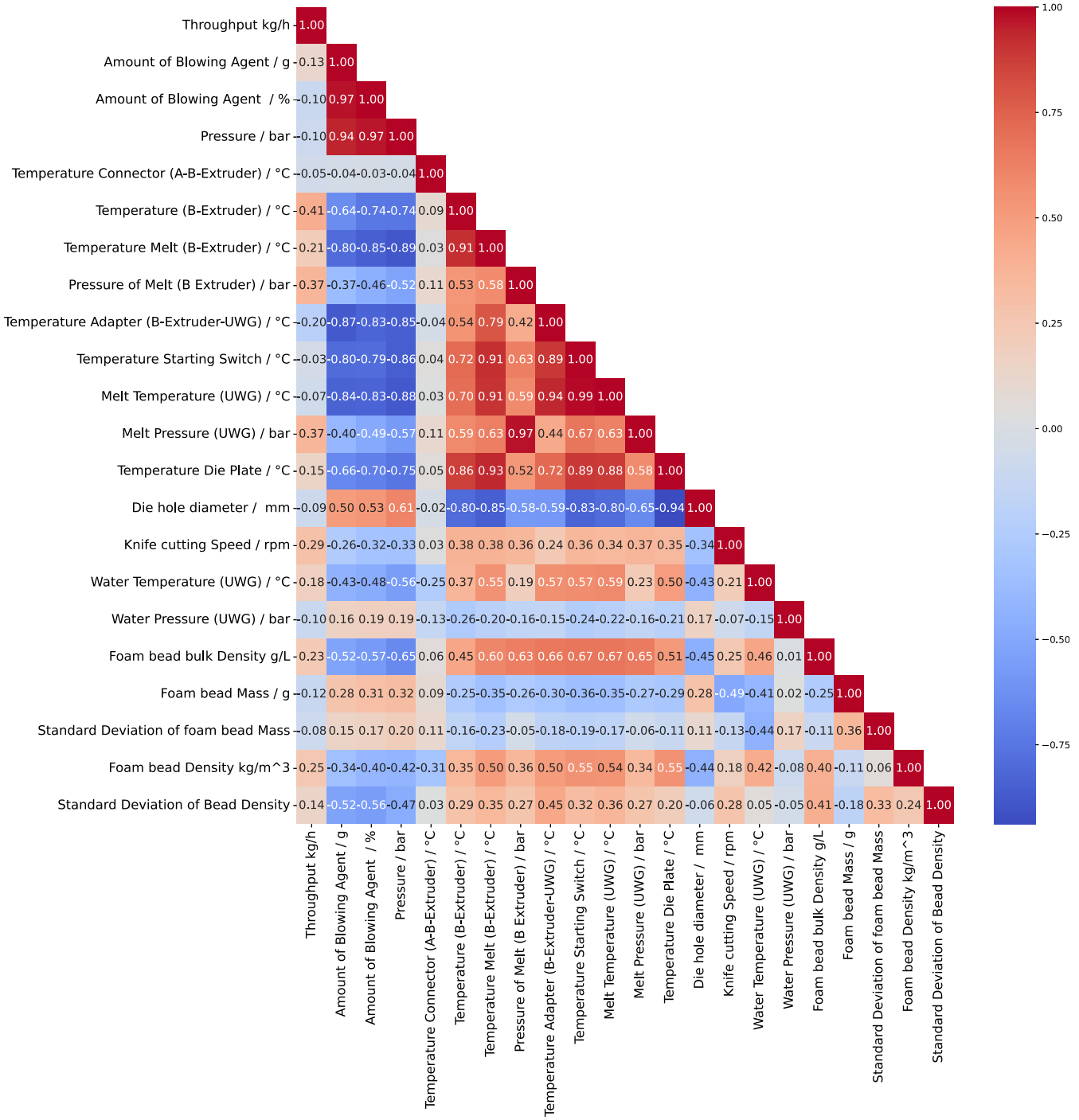

FIGURE 2 Heatmap matrix illustrating correlations between processing parameters, encompassing both features and target properties in the dataset. Red cells denote positive correlations, while blue cells represent negative correlations. [Color figure can be viewed at wileyonlinelibrary.com]

researchers and scientists in making informed decisions regarding process optimization and control. It's important to note that correlation does not imply causality, and further analyses may be needed to uncover the underlying mechanisms driving these relationships.

Note that, at this point, no model has been constructed. Figure 2 merely highlights wide patterns across the entire dataset.

Upon a detailed analysis of the correlation heatmap presented in Figure 2, it becomes apparent that, throughout the bead foam extrusion process, all the processing parameters exert some level of influence on each other. This intricate interconnectivity implies a sophisticated interplay among the parameters, contributing to the complex dynamics observed in the studied system. Significantly, specific parameters exhibit a more pronounced impact, evident in the emergence of block correlations within the heatmap. These correlations manifest as clusters of variables demonstrating a robust interdependence of higher degree. Notably, a distinct dense red color block or cluster is observed at the top of the heatmap, indicating a strong positive correlation (0.94) between the Amount of Blowing Agent and the Pressure. This correlation suggests that an increase in the blowing agent's quantity is strongly correlated with an increase in injector pressure.

In our analysis, we observe instances of negative correlation within the heatmap, manifesting as distinct blue clusters across the visualization. Notably, a clear negative correlation exists between specific parameters. For instance, the parameters temperature adapter (B-Extruder-UWG), temperature starting switch and melt temperature (UWG) exhibit a negative correlation with amount of blowing agent and the injector pressure, that is, (Correlation $= -0.80, -0.83, -0.85$), respectively. These coefficients signify a robust negative correlation between these parameters, implying a highly inverse relationship.

According to the insights provided by Figure 2, we obtain a comprehensive overview and visualization of the correlations among the target properties. Focusing on the first target property, which is the melt pressure in UWG, we observe strong positive correlations with pressure of melt in B-extruder and temperature starting switch, while negative correlations are evident with die size and pressure, respectively. Analyzing the second target property, that is, bead foam density, we observe numerous correlations, with the most significant ones being positively correlated with CO_2 content or amount of blowing agent, temperature of melt in B-extruder, and water pressure in UWG, respectively. Due to the intricate interrelation of various processing parameters on both target properties, we have established a threshold for correlation values. Specifically, we consider correlations with an absolute value $|\text{corr}| < 0.20$ as statistically insignificant for our subsequent machine learning analysis. This threshold helps filter out weak correlations, allowing us to focus on more meaningful relationships in the dataset.

3.2 | Prediction models for melt pressure in UWG

For melt pressure in UWG, major features include CO_2 content, injector pressure (bar), temperature in B-extruder, temperature starting switch, temperature die plate, and die hole size. This relationship is visually shown in the correlation heat-map Figure 2. After a comprehensive evaluation of various machine learning models, encompassing both linear and non-linear approaches, the objective was to identify the optimal model capable of accurately capturing the complex relationship between the identified features and the target variable (melt pressure) in bead foam extrusion. MAE and RMSE, quantify the deviation between predicted and experimental melt pressure.

The coefficient of determination R^2 score, MAE, and RMSE values are presented in the Table 2. These results provide a comprehensive overview of the predictive capabilities of each model. The higher R^2 score indicate better overall performance, while lower MAE and RMSE values signify greater precision in predicting the pressure of melt within the UWG. Figure 4a demonstrate the comparison graphically based on the models used to predict the melt pressure.

In this case, the RF model emerges as the most effective, possess the R^2 score at 0.96, accompanied by the lowest MAE of 0.851 and RMSE of 1.071. These results underscore the model's performance and efficiency in capturing complex relationships within the data. DT, LASSO, and GBR models also demonstrate acceptable results, showing robust performance and highlighting their potential for accurate predictions. Collectively, these outcomes provide valuable insights into the strengths and limitations of each model, facilitating the selection of an optimal predictive tool for estimating melt pressure in the UWG. The relationship between the predicted values from the regression model and the actual values is visually presented in Figure 3. Data points closely aligned with the perfect prediction line indicate a substantial compatibility between the predicted and observed values. The closer the point is to the perfect prediction line, the better is the model performance. Some of the predictions are showing significantly large, which is the result of the limited number of samples used to train the ensemble model.

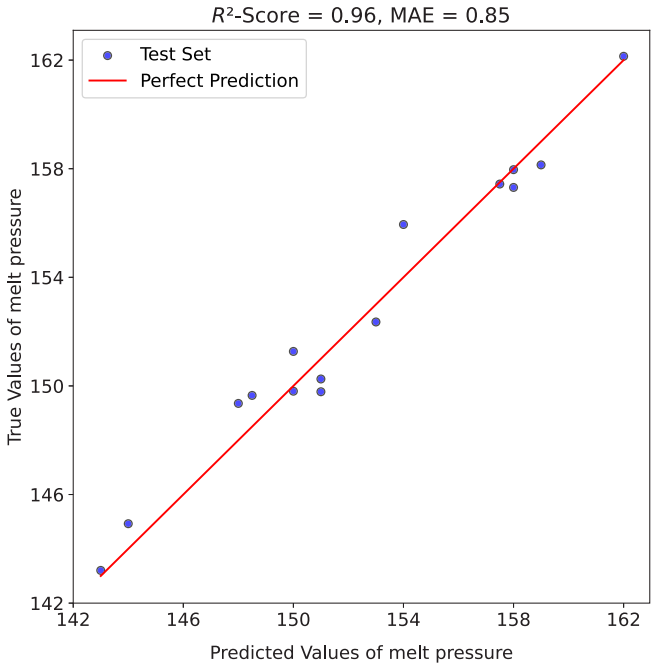

FIGURE 3 Comparison between the true value of melt pressure and predicted value of the melt pressure in UWG based on random forest regressor. The model is based on the MAE, RMSE and R^2 score after cross validation. [Color figure can be viewed at wileyonlinelibrary.com]

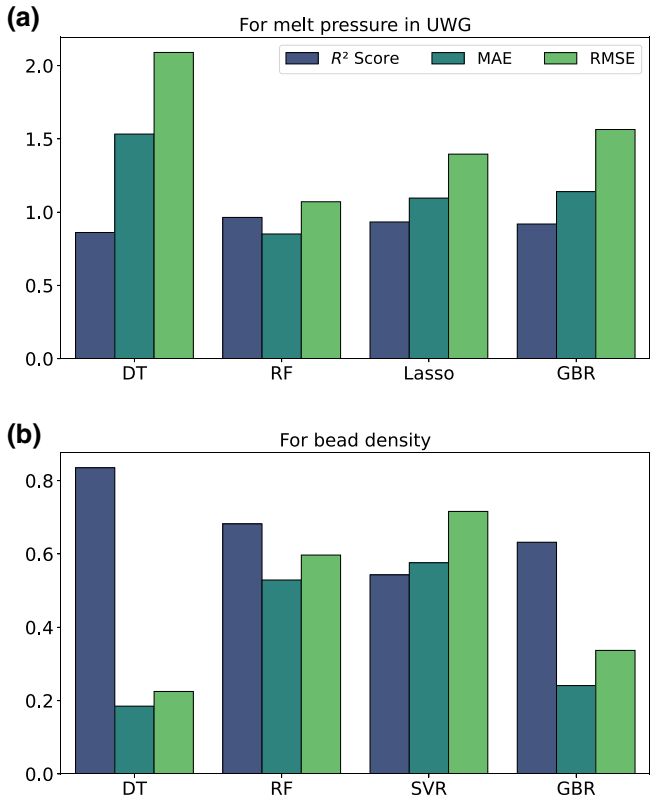

FIGURE 4 Graphical representation of performance metrics for melt pressure in UWG in subplot (a) and bead density across different models in subplot (b). [Color figure can be viewed at wileyonlinelibrary.com]

3.3 | Prediction models for bead density

Predicting bead density proves to be a challenging task, due to interconnected and correlated factors that influence the final outcome. To address this challenge, several machine learning models were trained and tested specifically for this task. However, only the most promising models demonstrated the capability suitable for obtaining improved results, which can be used for further study. A significant hurdle in dealing with bead density lies in the dataset's inherent deviation. Given that the primary objective in bead foam extrusion is to achieve low-density bead foams, the dataset exhibits considerable bias. Key features influencing bead density prediction include CO_2 content, Injector pressure (bar), Temperature in B-extruder, temperature starting switch, die plate temperature, melt temperature in B-extruder, and melt pressure in B-extruder. Analyzing the heatmap in Figure 2 revealed that density is influenced by numerous other features beyond the primary parameters, because of the process's complexity. The data complexity is visually represented in Figure 6a, illustrating a log-normal distribution of density data, signifying positive skewness.[25] This positively skewed distribution adds complexity to the data, making it challenging to train a machine learning model for accurate prediction. To address this challenge, a proper data normalization technique was essential. Utilizing a standardized log-normalized approach, the data was processed, resulting in the distribution shown in Figure 6b, which now conforms to a normal distribution. This normalization step is crucial for facilitating the training of a machine learning model to predict the target variable, that is, bead density. The log-normal distribution is a helpful tool that is used often in material and chemical research because of diverse physical mechanism and modeling. Normalizing the data ensures that comparisons between different datasets are meaningful and reliable, as it brings uniformity to the data by removing biases introduced by factors such as differences in sample size or measurement units. This ensures fair comparison and prevents skewed interpretation of experimental results.

For predicting the bead density, out of all the ML model used, the Decision Tree model demonstrated high performance, with R^2 score of 0.83, signifying that it explains approximately the 83% of the variance in predicting the bead density. Moreover, it exhibited lower MAE and RMSE values compared to the other models. These values highlight the capability of DT model to effectively capture and clarify a substantial portion of the variance within the dataset. The visual representation of the correlation between the predicted values derived

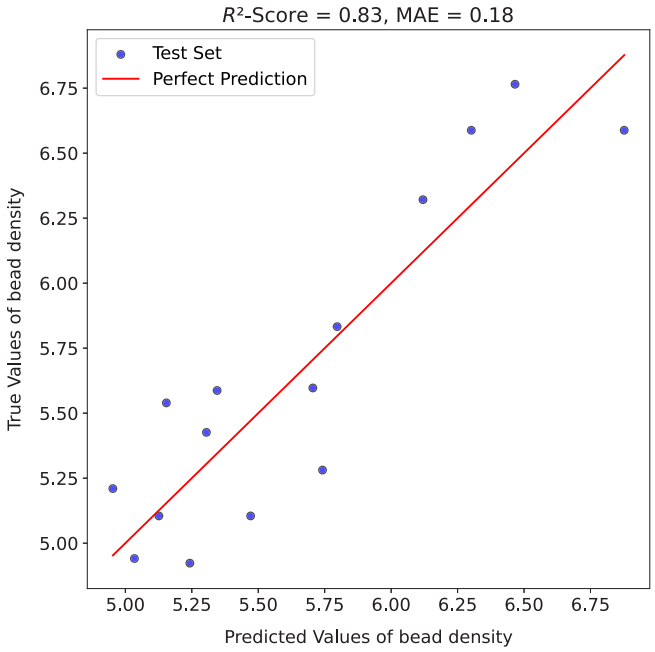

FIGURE 5 Comparison between the true value of density and predicted value of the density for decision tree regressor. The model is based on MAE, RMSE and R^2 score after cross validation. [Color figure can be viewed at wileyonlinelibrary.com]

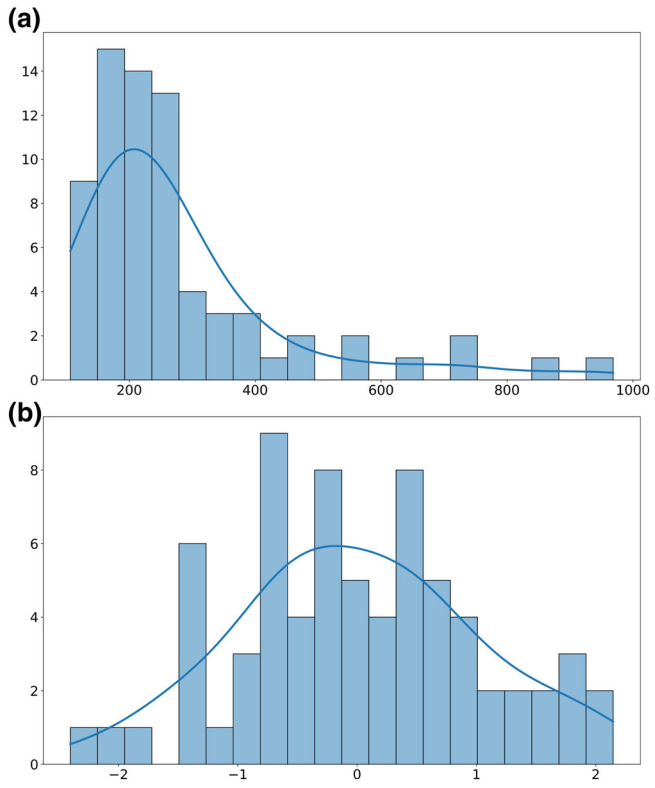

FIGURE 6 Comparison of distribution of data before transformation plot (a) and after transformation in plot (b) when predicting the bead density. A clear skewed can be seen in the subplot (a). [Color figure can be viewed at wileyonlinelibrary.com]

from the regression model and the corresponding actual values can be observed in Figure 5.

On the other hand, the other models also shows performance with R^2 score of 0.68, 0.63 and 0.54, respectively.

4 | CONCLUSION

In the present study, we have successfully showcased a machine learning approach to predict key properties of PLA bead foams, with a specific focus on melt pressure in the UWG and bead foam density. This innovative approach enhances our ability to understand and predict processing parameters and targets in bead foam extrusion more accurately. The study includes a detailed description of the best-performing ML model. Notably, the RF model exhibits excellent performance in predicting melt pressure in UWG, while the DT model performs well in predicting bead density. These models, in particular, have demonstrated effectiveness in capturing and illustrating the relationships among all processing parameters in PLA bead foam extrusion.

The study also clarifies the correlation of different processing parameters, such as temperature, pressure, screw speed, die plate temperature, and bead properties (i.e., density), leading to a comprehensive understanding of their impact on each other. It underscores the significance of optimizing these parameters to achieve desired foam properties and enhance the efficiency of the extrusion process. The correlation analysis has revealed key relationships between specific processing parameters and the final target. For example, variations in temperature were observed to significantly influence foam expansion and bead density. Additionally, an increase in CO_2 content was found to have high impact on pressure in the B-extruder. The most influential parameters for predicting melt pressure in UWG were identified as CO_2 content, injector pressure, temperature in B-extruder, temperature starting switch, temperature die plate, and die hole size. Similarly, crucial parameters for predicting density included CO_2 content, injector pressure, temperature in B-extruder, temperature starting switch, temperature die plate, melt temperature in B-extruder, and melt pressure in B-extruder.

This successful demonstration of machine learning applications not only advances our understanding of PLA bead foam but also paves the way for further optimization of bead foam extrusion in other polymers.

AUTHOR CONTRIBUTIONS

Karim Ali Shah: Conceptualization (lead); data curation (lead); formal analysis (lead); investigation (lead);

methodology (lead); software (lead); validation (lead); visualization (lead); writing – original draft (lead). **Christian Brütting:** Investigation (equal); supervision (equal); writing – review and editing (equal). **Rodrigo Q. Albuquerque:** Conceptualization (equal); methodology (equal); software (lead); supervision (equal); writing – review and editing (equal). **Holger Ruckdäschel:** Funding acquisition (lead); project administration (lead); writing – review and editing (lead).

ACKNOWLEDGMENTS

The authors would are thankful to the German Research Foundation (DFG) for funding this project with grant number RU 2586/5-1. We extend our sincere gratitude for the financial support provided. Support from the "Bayerischen Staatsministerium für Wissenschaft und Kunst" under grant number F.2-M7426.10.2.1/4/16 in Germany is acknowledged. Open Access funding enabled and organized by Projekt DEAL.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Holger Ruckdäschel https://orcid.org/0000-0001-5985-2628

ENDNOTES

[a] https://scikit-learn.org.

[b] https://numpy.org.

[c] https://jupyter.org.

REFERENCES

[1] V. Altstädt, G. Krausch, *Polymer* **2015**, *56*, 56.
[2] C. Brütting, J. Dreier, C. Bonten, V. Altstädt, H. Ruckdäschel, A. C. S. Sustain, *Chem. Eng.* **2023**, *11*, 6676.
[3] H. Long, H. Xu, J. Shaoyu, T. Jiang, W. Zhuang, M. Li, J. Jin, L. Ji, H. Ying, C. Zhu, *Polymer* **2023**, *15*, 895.
[4] T. Standau, H. Long, S. Murillo Castellón, C. Brütting, C. Bonten, V. Altstädt, *Materials* **2020**, *13*, 1371.
[5] N.-A. A. B. Taib, M. R. Rahman, D. Huda, K. K. Kuok, S. Hamdan, M. K. B. Bakri, M. R. M. B. Julaihi, A. Khan, *Polym. Bull.* **2023**, *80*, 1179.
[6] E. Alpaydin, MIT Press. **2020**.
[7] L. Erhard, R. Heiberger, *in Research Handbook on Digital Sociology*, Edward Elgar Publishing, Ireland **2023**, p. 130.
[8] J. Wei, X. Chu, X.-Y. Sun, K. Xu, H.-X. Deng, J. Chen, Z. Wei, M. Lei, *InfoMat* **2019**, *1*, 338.
[9] C. Gao, X. Min, M. Fang, T. Tao, X. Zheng, Y. Liu, X. Wu, Z. Huang, *Adv. Funct. Mater.* **2022**, *32*, 2108044.
[10] M. J. Hooshmand, C. Sakib-Uz-Zaman, M. A. H. Khondoker, *Materials* **2023**, *16*, 7173.
[11] A. Frankel, C. M. Hamel, D. Bolintineanu, K. Long, S. Kramer, *Comput. Methods Appl. Mech. Eng.* **2022**, *391*, 114492.
[12] T. Zhou, Z. Song, K. Sundmacher, *Engineering* **2019**, *5*, 1017.
[13] W. Sha, Y. Li, S. Tang, J. Tian, Y. Zhao, Y. Guo, W. Zhang, X. Zhang, S. Lu, Y.-C. Cao, S. Cheng, *InfoMat* **2021**, *3*, 353.
[14] R. Neelam, S. A. Kulkarni, H. Bharath, S. Powar, M. Doddamani, *Results Eng.* **2022**, *16*, 100801.
[15] T. E. Gartner III, A. Jayaraman, *Macromolecules* **2019**, *52*, 755.
[16] J. Jiang, L. Wang, F. Tian, Y. Li, W. Zhai, *Macromol* **2023**, *3*, 782.
[17] J. Kuhnigk, T. Standau, D. Dörr, C. Brütting, V. Altstädt, H. Ruckdäschel, *J. Cell. Plast.* **2022**, *58*, 707.
[18] R. Q. Albuquerque, C. Brütting, T. Standau, H. Ruckdäschel, *E-Polymers* **2022**, *22*, 318.
[19] K. Mulrennan, J. Donovan, L. Creedon, I. Rogers, J. G. Lyons, M. McAfee, *Polym. Test.* **2018**, *69*, 462.
[20] P. Xu, X. Ji, M. Li, W. Lu, *npj Comput. Mater.* **2023**, *9*, 42.
[21] C. Kaun, N. Jhanjhi, W. W. Goh, S. Sukumaran, MATEC Web Conf., 2021, 335, 04002, EDP Sciences.
[22] J. Ali, R. Khan, N. Ahmad, I. Maqsood, *Int. J. Comput. Sci. Issues (IJCSI)* **2012**, *9*, 272.
[23] J. Dou, A. P. Yunus, D. T. Bui, A. Merghadi, M. Sahana, Z. Zhu, C.-W. Chen, K. Khosravi, Y. Yang, B. T. Pham, *Sci. Total Environ.* **2019**, *662*, 332.
[24] H. Mouri, *Phys. Rev. E* **2013**, *88*, 042124.
[25] D. B. Siano, *J. Chem. Educ.* **1972**, *49*, 755.

SUPPORTING INFORMATION

Additional supporting information can be found online in the Supporting Information section at the end of this article.

How to cite this article: K. A. Shah, C. Brütting, R. Q. Albuquerque, H. Ruckdäschel, *J. Appl. Polym. Sci.* **2024**, *141*(30), e55693. https://doi.org/10.1002/app.55693