



# Modeling Fused Filament Fabrication using Artificial Neural Networks

Paul Oehlmann<sup>1</sup> · Paul Osswald<sup>2</sup> · Juan Camilo Blanco<sup>3</sup> · Martin Friedrich<sup>2</sup> · Dominik Rietzel<sup>2</sup> · Gerd Witt<sup>4</sup>

Received: 8 November 2020 / Accepted: 11 January 2021 / Published online: 7 February 2021  
© The Author(s) 2021

## Abstract

With industries pushing towards digitalized production, adaption to expectations and increasing requirements for modern applications, has brought additive manufacturing (AM) to the forefront of Industry 4.0. In fact, AM is a main accelerator for digital production with its possibilities in structural design, such as topology optimization, production flexibility, customization, product development, to name a few. Fused Filament Fabrication (FFF) is a widespread and practical tool for rapid prototyping that also demonstrates the importance of AM technologies through its accessibility to the general public by creating cost effective desktop solutions. An increasing integration of systems in an intelligent production environment also enables the generation of large-scale data to be used for process monitoring and process control. Deep learning as a form of artificial intelligence (AI) and more specifically, a method of machine learning (ML) is ideal for handling big data. This study uses a trained artificial neural network (ANN) model as a digital shadow to predict the force within the nozzle of an FFF printer using filament speed and nozzle temperatures as input data. After the ANN model was tested using data from a theoretical model it was implemented to predict the behavior using real-time printer data. For this purpose, an FFF printer was equipped with sensors that collect real time printer data during the printing process. The ANN model reflected the kinematics of melting and flow predicted by models currently available for various speeds of printing. The model allows for a deeper understanding of the influencing process parameters which ultimately results in the determination of the optimum combination of process speed and print quality.

**Keywords** Additive manufacturing · Fused Filament Fabrication · 3D printing · Artificial intelligence · Neural networks · Machine learning · Deep learning · Big data · Process control · Process monitoring · Signal processing · Industry 4.0

## 1 Introduction

While the quality and speed of additive manufacturing processes have improved significantly since the first patent in selective laser sintering (SLA) in 1986 [1], AM is still considered a relatively new production method. However, in Fig. 1, a strong growth in the past decade can be observed when comparing total revenues of AM systems and products worldwide between 1993 and 2018. The total revenues grew

by 323% up to 8.3 billion US \$ with a potential growth of 428% to a total value of 35.6 billion US \$ by 2024 [2].

Over the past decade, the degree of production consistency as well as the monitoring and control of these production processes has developed considerably, but still does not compare to that of traditional manufacturing methods. In order to make improvements in areas of industrialization, even pushing towards industry 4.0, a better understanding of the respective processes and what is actually happening during a build process is needed.

Monitoring possibilities and the associated generation of data are only available to a limited extent due to the relatively small number of sensors in use. If available, machine and process data is a large yet disorganized resource which can be tapped for collection, however, the evaluation of such information, especially in large quantities, becomes a core challenge.

The recent, some might say, second coming of artificial intelligence allows for a potential form-fitting cooperation

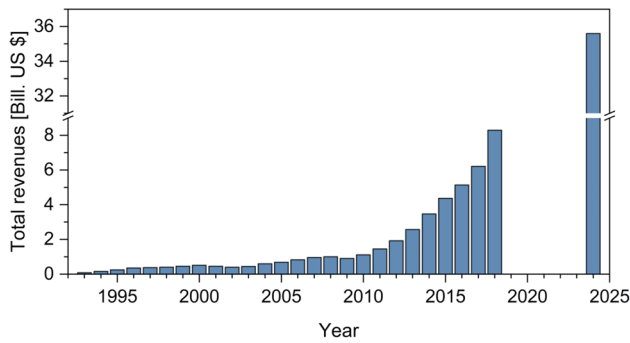
✉ Paul Oehlmann  
paul.oehlmann@tum.de

<sup>1</sup> Technical University of Munich, Department of Mechanical Engineering, Munich, Germany

<sup>2</sup> BMW Group, Munich, Germany

<sup>3</sup> Fused Form Corp, Bogotá, Colombia

<sup>4</sup> Universität Duisburg-Essen, Duisburg, Germany



**Fig. 1** Total revenues of AM services and products between 1993 and 2018 with a prognosis until 2024 [2]

between a field full of excitement yet wrought with problems surrounding quality and stability, and a field that can take large amounts of abstract data and information, characterizing highly complex processes [3].

This study uses an FFF printer as a production system and an artificial neural network to investigate the influence of filament speed and process temperature on the force within the nozzle, which is directly related to print-quality and speed. Large amounts of data recorded in real time using a customized printer with sensors are required as an input for the ANN, achieving a comprehensive proof of concept for process monitoring and control in AM technologies.

### 1.1 Artificial neural networks in additive manufacturing

AM technologies offer a broad range of relatively new processes using different procedures, energy sources and materials for a layer wise production. All have complex manufacturing mechanisms in common, such as surface quality, mechanical properties, process time, cost and many more, suggesting a high number of influential process parameters

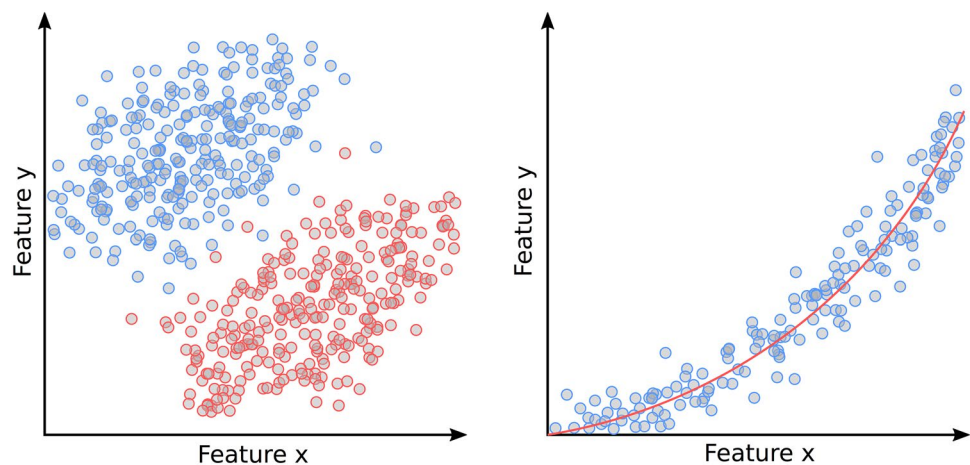
are needed in order to achieve optimum results [4]. This condition in combination with a lack of inline process monitoring and unknown failure mechanisms suggests the possibility to generate large amounts of data, which is an important prerequisite for the use of AI.

ANN algorithms are ideal for taking large amounts of data, which are either directly or indirectly interrelated, and processing it in a way which allows a user to better understand any patterns or effects hidden within this information. Possible ways of processing data with a NN are regression or classification problems [5], shown in Fig. 2. Depending on the inputs and outputs for specific AM applications, an outcome could be a prediction of process parameters or detection of failure within prints, to name a few.

However, understanding the fundamental mechanisms of processes is important when verifying results generated with an ANN, which might be based on experimental input data. This underlines the need for supporting data infrastructure, integrating closed loop iterations between experimental and mathematical/physical modeling [6]. While experimental results may be hard to gain, numerical datasets can also be used for training of an ANN by generating unlimited amounts of data, thus, solving one of the key challenges in data science. This procedure is balancing computational costs and time between conventional mathematical, and data science simulation approaches [7].

Existing approaches using AI in the field of FFF printing include the work by Bayraktar which uses an ANN to predict the mechanical properties using print orientation, nozzle temperature and layer thickness as input parameters [8]. Additionally, in 2016 Wu et al. uses an acoustic sensor to set up a real time state detection system to distinguish the extrusion and material loading state as well as identifying nozzle blockages [9]. While these approaches are novel findings for the area of AI in AM, they do not rely on fundamental process knowledge in polymer processing and additive manufacturing. The model developed in this study combines

**Fig. 2** Exemplary representation of a classification (left) and regression (right) problem



existing physical models of the FFF extrusion process to verify the validity of the ANN.

### 1.2 Kinematics of melting and flow inside the nozzle

During manufacture of a part using the material extrusion process, an amorphous or semi-crystalline thermoplastic filament is melted and extruded through a heated nozzle [10–14]. The extruded filament or bead is deposited and fused onto a previously applied layer and subsequently cooled down and solidified. The central component of this process is the extruder, which consists of the drive system and the nozzle. The nozzle melts the plastic and the movement generated by the drive pushes it through the capillary to form the bead. After three decades of existence, there is still controversy on how the material melts inside the nozzle [11]. It is clear that the limiting factor of the fused filament fabrication (FFF) process is the melting rate which controls the printing speed [11].

Modeling the nozzle that produces the bead, has been of interest to various researchers in past decades [7, 10–17]. There are two general approaches of thought, schematically depicted in Fig. 3, that are plausible, depending on the speed that the filament is pushed through the nozzle:

1. For low filament speeds, the filament melts soon after entering the nozzle. Here, the nozzle is assumed filled with molten polymer and the filament acts as a piston

that pushes the melt through the capillary, much like a syringe pushes a fluid through a needle [10].

2. For high filament speeds, the molten material only exists at the bottom of the nozzle, in form of a thin film, where the filament pushes the melt toward the capillary in a squeezing flow fashion [11].

The first, or slow speed assumption [10, 12], computes the pressures required to push the melt through Sects. 1, 2 and 3 at a certain speed and computes the force required to push the filament based on the sum of those pressures. The first section, denoted by “I” in the schematic, can be approximated by a Hagen-Poiseuille flow, or pressure flow through a tube, with a pressure requirement of  $\Delta p_I$ . It should be pointed out here, that the length of the molten Sect. 1, denoted by  $L_I$ , is actually unknown and depends on the speed the filament is driven through the nozzle. For a shear thinning polymer melt, represented using a power-law viscosity model given by

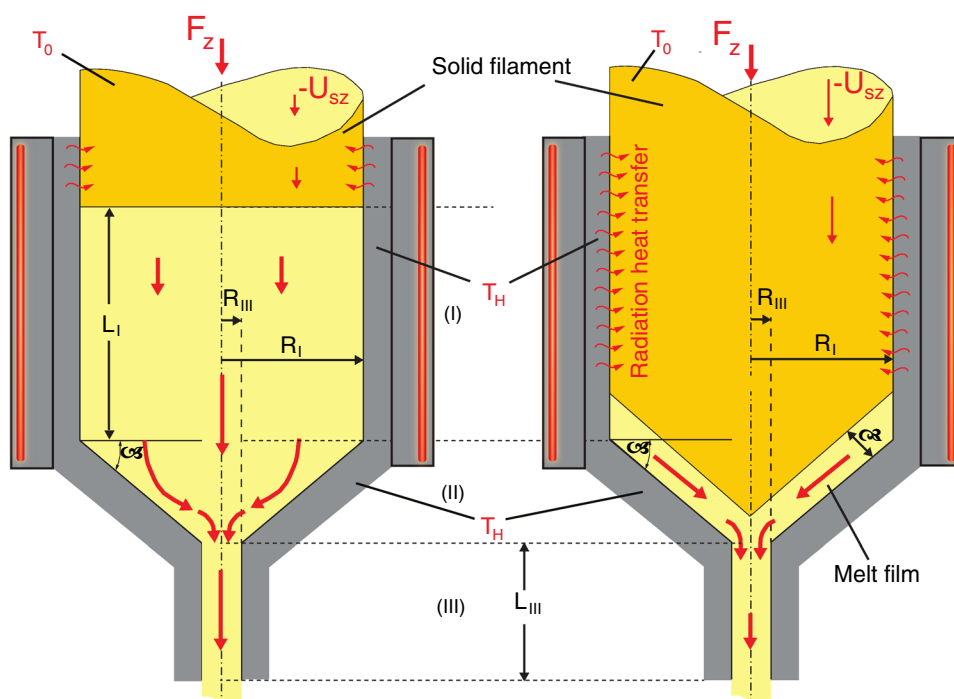
$$\eta = m(\dot{\gamma})^{n-1} \tag{1}$$

the volumetric flow rate is given by

$$Q = \left( \frac{n\pi}{3n+1} \right) \left[ -\frac{R_I^{3n+1}}{2mL_I^{1/n}} \right] \Delta p_I^{1/n} \tag{2}$$

where,  $\dot{\gamma}$  is the magnitude of the rate of deformation tensor,  $n$  is the power-law index,  $m$  is the consistency index,  $L_I$  is the length of the section and  $\Delta p_I$  the pressure required to push the melt through Sect. 1 at the given volumetric flow

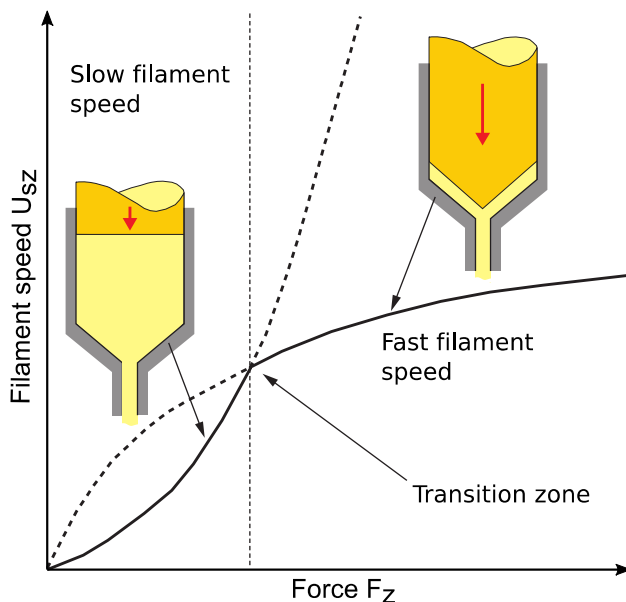
**Fig. 3** Melting modes inside the nozzle for low filament movement (left) and high filament speed (right)



rate. The second section is where the melt transitions from a larger radius  $R_I$  to a smaller radius  $R_{III}$ , a contraction that leads to a pressure loss  $\Delta p_{II}$ . Finally, the third section is also a Hagen-Poiseuille flow that for a certain volumetric throughput has a pressure requirement of  $\Delta p_{III}$ . Although the model starts with the assumption that the nozzle is filled with molten polymer, the pressure required to push the melt through this section is negligible compared to the pressure required to push the melt through the capillary. For example, a typical filament measures 1.75 mm or 2.85 mm in diameter, while the capillary typically measures 0.4 mm. From the Hagen-Poiseuille equation, we know that the pressure requirement is proportional to  $\frac{1}{R^4}$ . Therefore, assuming the length of the sections are of the same order of magnitude, the pressure required to push the melt through the 0.4 mm capillary is 366 times higher than pushing melt through a 1.75 mm nozzle ( $\frac{R_I^4}{R_{III}^4} = 366$ ). Similarly, the pressure requirement to push the resin through the conical contraction is proportional to  $(\frac{1}{R_{III}^4} - \frac{1}{R_I^4})$ , making it also insignificant when comparing it to the flow requirement through the 0.4 mm capillary. For this reason, the Bellini model [10] works quite well, as long as there is a significant amount of melt inside the nozzle body. Hence, the melting rate, or filament speed  $U_{sz}$ , is a function of the force  $F_z$  applied to the filament, as schematically depicted in Fig. 4, and given by

$$U_{sz} \propto F_z^{1/n} \quad (3)$$

Note that for a power-law index  $n = 0.5$ ,  $U_{sz} \propto F_z^2$ .



**Fig. 4** Schematic of filament speed as a function of force applied to the filament

The second, or fast speed assumption [11], is when the speed is fast enough that not enough time is given to melt the plastic in the main body of the nozzle and the solid front at the tip of the filament reaches the bottom of the nozzle. Here, the melting is similar to melting a stick of butter against the hot surface of a pan, except the melt is not pressed outward but toward the center and out the capillary. Within the polymer processing community this type of melting is known as *melting with pressure flow removal* [11]. This model is based on a mass, momentum and energy balance of the melt within the melt film. With the resulting model, the melting rate is proportional to the fourth root of the force applied to the filament [10], as schematically depicted in Fig. 4 and given by

$$U_{sz} \propto F_z^{1/4} \quad (4)$$

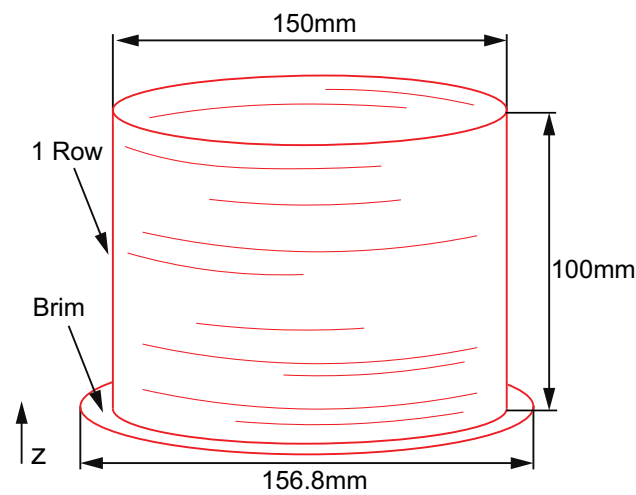
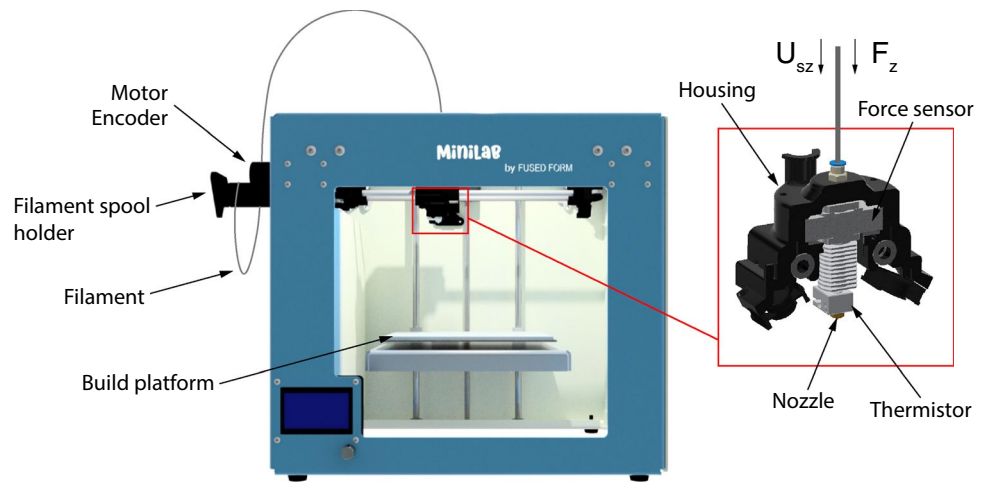
The melt film between the heated nozzle is much smaller than the radius of the filament, typically under 100  $\mu\text{m}$  in thickness [11].

Most publications [12–14, 16, 17] use the first model, proposed by Bellini et al. [10]. However, as the filament speed increases and the space between solid front and the tip of the nozzle  $\delta$  becomes small, the melting and flow modes transition to *melting with pressure flow removal*, basically representing an upper bound or limiting factor of any FFF printer [11].

## 2 Materials and methods

The custom-made FFF printer *Minilab* by Fused Form Corp., shown in Fig. 5, is used in this study to generate data produced in the printing process. The *Minilab* was equipped with a force sensor, an encoder and a thermistor in order to monitor filament force, filament speed, nozzle- and surrounding temperature while 3D printing. This data was collected using an Arduino board connected to Matlab for data visualization, processing and logging. A special extruder in combination with a compact diaphragm capsule force sensor was developed with the purpose of having a force feedback of the extrusion process while keeping the original performance of the printer. This allows for the recording of the actual filament extrusion forces which occur during printing. This was achieved by placing the force sensor just above the hot end in a Bowden extruder architecture. The filament extrusion length and speed were detected with a two quadrature AB-channel, incremental rotary *GTS-AB Series* encoder placed before the Bowden extruder motor as shown in Fig. 5. By separating the motor from the encoder, the system is also able to detect filament slipping when reaching the maximum speeds and forces.

**Fig. 5** FFF printer “Minilab” by Fused Form Corp. equipped with sensors



**Fig. 6** Cylindrical geometry used to test the neural network

The sensor provides an accuracy of  $\pm 20$  g or 0.2 N. The temperature of the nozzle was detected in the hot end cube with a NTC100K sensor placed aside of the thermistor used for controlling the nozzle temperature during printing.

A cylindrical helix shaped part (Fig. 6) made out of a single row of filament with a wall thickness of 0.4 mm and a diameter and height of 150 mm  $\times$  100 mm is used to monitor the variation of speed and force in combination with different nozzle temperatures throughout the printing process. A brim consisting of one layer with an outer diameter of 156.8 mm is first printed around the cylinder to prevent warpage as well as a detachment from the build platform. The geometry is selected due to a constant speed of the x and y axis while printing the continuous helix shape along the z axis. The filament speed is therefore set to a certain value, increasing every ten rotations to capture the whole range in one print. The adjustment of the speed is done by a command in the gcode at the end of a full rotation. The

**Table 1** FFF printing parameters

Print Nr	Material	Nozzle temperature (°C)	Printed temperature (°C)	Filament speed (mm/s)
1	PLA	200	60	1.0–7.0
2	PLA	215	60	1.0–7.0
3	PLA	230	60	1.0–7.0

transition between speeds is done seamless without stopping the filament feed. Three temperatures from 200 to 230 °C are tested in total.

A total of approximately 12.000 data points are recorded for each print. A 0.4 mm nozzle is used in combination with a 1.75 mm filament. The prints are performed with the commercially available *Natural PLA PRO*, by Matterhackers, to reduce effects caused by colors and additives existing in the material. The print bed is preheated for every test to 60 °C. The nozzle temperature is set as well before starting a print. The parameters used for printing are summarized in Table 1.

### 3 Artificial neural networks

The idea of deep learning is actually one that is based on how humans process information and learn. Our memory, our actions, our thoughts are controlled by our central nervous system which is composed of neurons. The concept on which ANN are based upon is how our brains process and store information through connections between various neurons and their relative strengths. With this in mind, we can basically describe a neural network as a group of connected neurons, where the number of layers (depth) and number of neurons per layer (width) define the networks architecture. Each input is multiplied with a weight, connecting the input to the neuron itself. These weights are determined during



training of the neural network using application specific data. The data is then fed to an activation function, modifying the input with non-linear functions and passing the output to the next neuron, able to model highly complex relationships. [18]

A simple artificial neural network (ANN) consists of at least three layers. The first layer is called the input layer, followed by a hidden layer, and finally, by the output layer neurons, as schematically depicted in Fig. 7.

Information, such as filament velocity,  $U_{sz}$ , and nozzle temperature,  $T_H$ , is fed into the input layer, and using the connections between the neurons, passes through the hidden layers and delivers a value to the output layer which is then compared to the actual output value. The output contains information such as filament force,  $F_z$ , that is physically dependent on the input parameters. Every connection in the ANN has a weight,  $w_{ij}$ , associated with it, and each neuron has a bias or threshold value,  $b_j$ , and an activation function,  $\phi$ , designated to it. This allows nodes in the input layer, or neurons in the hidden layer to be connected to a neuron on its right-hand side, schematically depicted in Fig. 8.

In equation form, the diagram depicted in Fig. 8 is written as

$$y_i = \Phi(w_{ij}x_i + b_i) \quad (5)$$

During the training of an ANN, the input values are passed through each of the hidden layers in the network with a starting set of weights and biases. The complex procedure involves numerous steps, parameters and

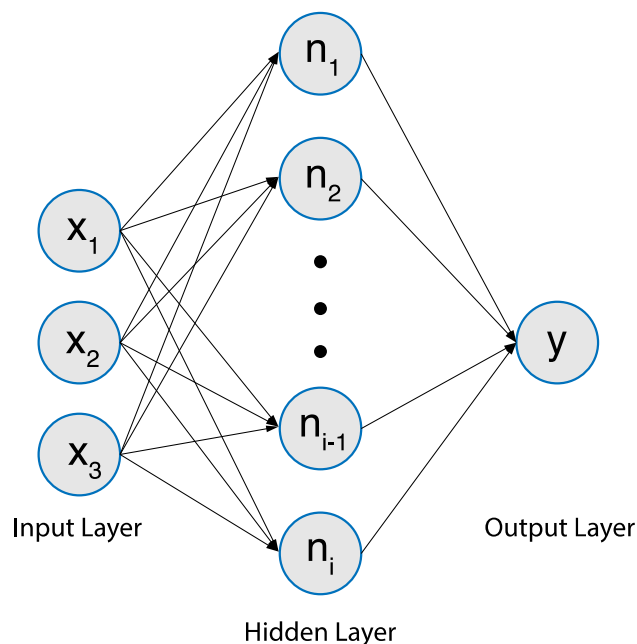


Fig. 7 Architecture of a neural network (according to [19])

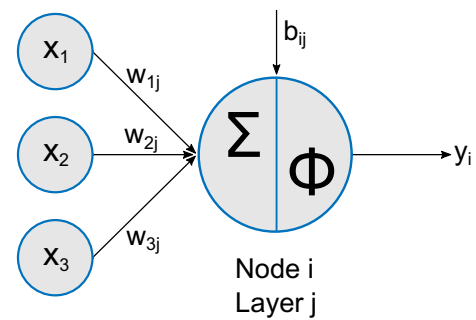


Fig. 8 Neuron with input parameters  $x_i$ , weights  $w_{ij}$ , bias  $b_{ij}$ , activation function  $\Phi$  and output  $y_i$  (according to [19])

hyperparameters to be chosen based on the application in order to generate the best possible results. The dataset and an initialized set of parameters is fed to the model, calculating the model's predictions based on the given input. The computed output is compared to the desired output that was fed to the ANN and the weights and biases are adjusted using a gradient descent back-propagation error minimization scheme [19].

The ANN executes many iterations, continuously adjusting the weights and biases. The error is constantly calculated during training using a loss function. In order to minimize the error, the derivative of the loss-function is calculated, and weights and biases are adjusted before the next iteration. This basic principle refers to the gradient descent algorithm for solving an optimization problem. The starting point for the algorithm is set by the initialized weights for the training of the ANN. First, the gradient of the loss function is calculated, and the local steepest direction of descent is chosen for the next iteration, using a specific step size for adjusting the weights. Dynamically adjusting the step size and choosing the best algorithm is of particular importance in optimizing functions to ensure a fast convergence of the problem and also depends strongly on the application.

This process can be done in batches by splitting the data into  $k$  folds. The model is then trained on folds  $2-k$  and tested on fold 1. This process is repeated  $k$  times, using the remaining fold (2, 3, ...,  $k$ ) as a test fold. By averaging the loss function, the out-of-sample model performance can be estimated. A commonly used loss function can be the root mean squared error (RMSE). Choosing a function again depends on the problem and highly affects the model's performance. The previously explained steps on training are repeated using different values for the models hyperparameters. These include in particular a dropout rate, used to prevent a model from overfitting as well as the depth and width. Depending on the application, others are available and should be considered based on the application [20].

#### 4 Development of an artificial neural network FFF model

The process of developing an artificial neural network model requires several important steps in order to achieve suitable results, and strongly depends on the existing problem. Predicting the force within the nozzle of an FFF printer as a continuous value using filament speed and nozzle temperatures as input data is defined as a supervised regression problem. The generated data provides the basis for parameters and methods to be defined during the process and must therefore be closely examined and pre-processed. The model's architecture is defined by the number of input parameters, hidden layers (depth), neurons within the hidden layers (width) and output parameters.

Training and validation of the model describes the main process step and is key to achieve accurate results for the existing problem when testing it. When training a model with the amount of predefined labeled data, the resulting output  $y_{i,m}$  is constantly compared to the actual value  $y_{i,d}$  of the dataset, forming an error. As described previously, the weights and biases are optimized by minimizing the error using a loss function within a gradient descent algorithm, until the weights converge. The number of iterations the dataset is passed through the model is controlled by the variable epochs. When validating the model, the remaining data which has not been used for training is fed to the model. In this process, the actual value is compared to the target value, without any further adjustment of the neuron properties to ultimately calculate the model's accuracy.

Choosing the optimum configuration of width and depth involves starting with a rough guess as there is no generic way on how to determine the best values a priori. This is also a result of an interaction of width and depth with other hyperparameters of the model. A possible starting point can be choosing the number of neurons and hidden layers closely to the number of input parameters and build up complexity based on the model's output [21]. The next step involves hyperparameter tuning which determines the networks structure as well as how the network is trained. The goal is always set so prevent the model from overfitting while increasing its complexity and ability to generalize well. This means that a model trained on certain data is able to make accurate predictions on new data from the same class as the training set [22]. A common technique, the dropout rate is a regularization method to avoid overfitting by dropping units randomly with a certain rate during training of the neural network. This prevents units within the network from co-adapting too much [23].

With each input variable of the total number of 20.000 datapoints being on different scales (filament speed between 1 and 7 mm/s and temperatures from 200 to

230 °C), a feature standardization shown in Eq. (6) is used to improve the performance of the model [24]. By subtracting the mean value  $\bar{x}$  from the original value  $x_i$  and dividing the sum by the standard deviation  $s$ , the product  $z_i$  has a mean of 0 with a standard deviation of 1.

$$z_i = \frac{x_i - \bar{x}}{s} \quad (6)$$

A network architecture with three hidden layers, consisting of five neurons each was found as an effective and economical ANN. The decision process is based on recording the training and validation error of both, the training and validation dataset over the number of epochs. The results were compared to other configurations of width and depth and showed, that increasing the number of neurons and/ or layers resulted in overfitting while decreasing led to underfitting and higher errors. Here, 80% of the datapoints were used to train the ANN and 20% were used to validate and test the model. The common loss- and evaluation functions for regression problems, mean square error (MSE) and mean absolute error (MAE) are used to record the error vs. epochs for training and validation loss. By plotting the development of both values, the most suitable number of epochs can be visualized, avoiding excessive computing time or overfitting of the model, and again evaluating the chosen network architecture.

Figure 9 shows the results by plotting the mean absolute error ( $F_z$ ) of the training and validation dataset over the epochs for the previous selected architecture. An early stopping function is also used to prevent overfitting of the model by stopping the training process, if the validation loss is not decreasing after a preset number of epochs [5]. Using this model, the mean absolute error was minimized to 0.72 N.

Figure 10 shows the predictions made on the test dataset against the actual, true values (left), and the distribution

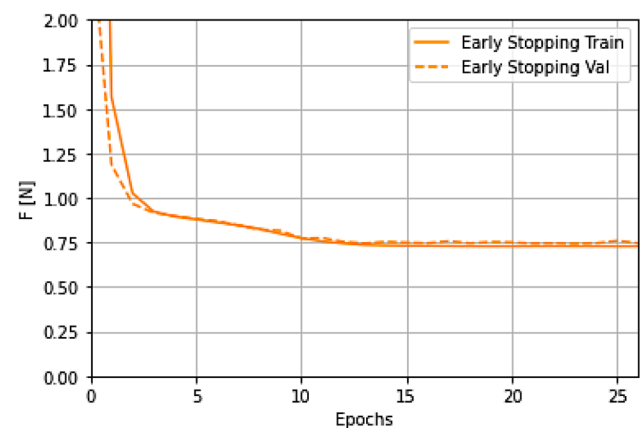


Fig. 9 Mean absolute error of training (Train) and validation (Val) dataset using early stopping function

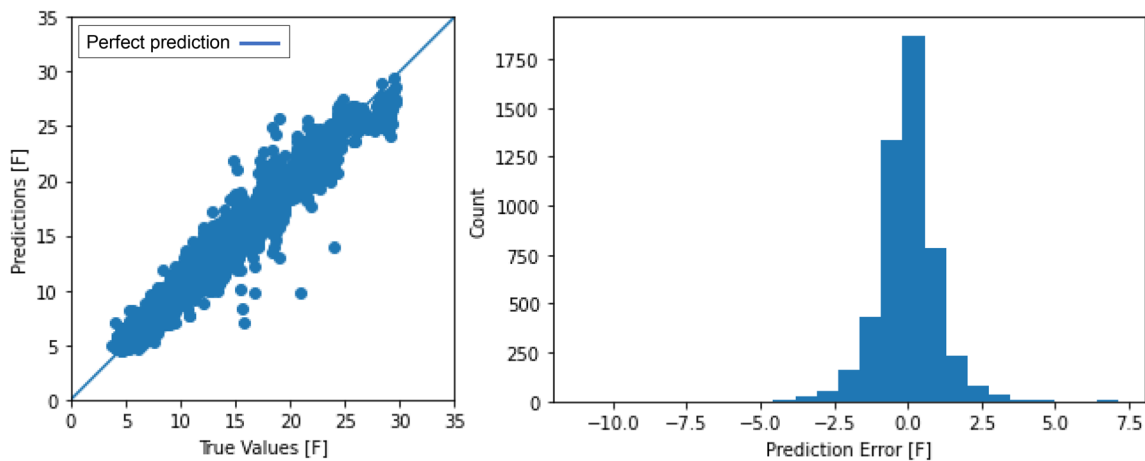


Fig. 10 Test data predictions against true values (left) and distribution of the error (right)

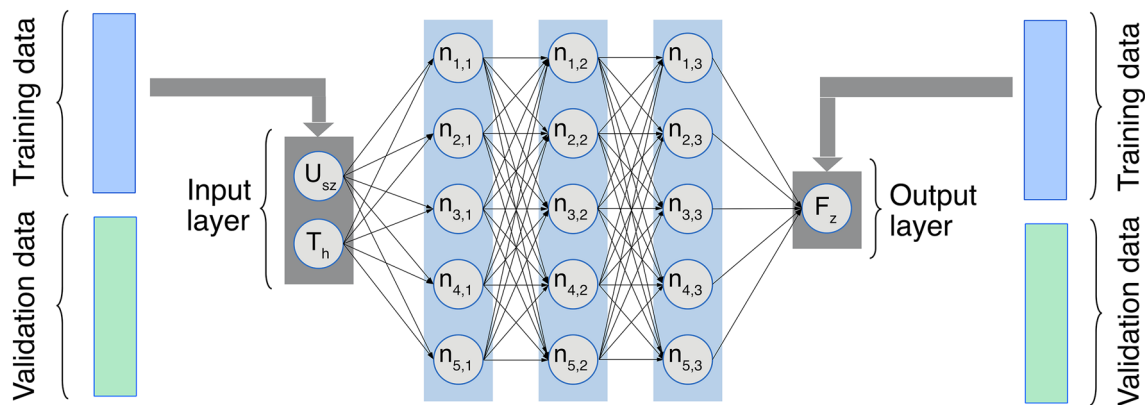


Fig. 11 Sequential deep neural network used for the regression problem

of the prediction error (right). A diagonal line with the slope of 1 is added to the actual vs. predicted plot, showing the region where the prediction is equal to the values computed by the algorithm. It can be seen that the model shows good results especially below force values of 25 N. Predictions above 25 N relate to observations made during the print, showing a high rate of filament slipping at higher print speeds especially at lower temperatures.

The developed ANN is shown in Fig. 11. Each neuron within the hidden layers is using the ReLU (Rectifier Linear Unit) activation function. The lines between input nodes, output nodes and each neuron symbolize a possible connection for the dataset to be processed. It should be considered that certain arrangements are particularly suitable for specific problems and configurations of data. The network presented here shows good results in this particular case and has therefore been selected for this application. The parameters used to set up the model are summarized in Table 2.

Table 2 Neural network summary

Layer	Layer type	Specification	Nodes	Dropout rate
1	Input	$U_{sz}, T_h$	2	–
2	Hidden	ReLU, dense	5	0.2
3	Hidden	ReLU, dense	5	0.2
4	Hidden	ReLU, dense	5	0.2
5	Output	$F_z$	1	–

Finally, the trained deep learning model is saved with the defined weights and bias and can be applied for datasets generated with different temperatures trained previously, ultimately replacing the force sensor. Future work includes printing and training of more datasets, expanding the model to a higher range of process temperatures.



## 5 Experimental results

Raw data generated during the print, using speed, force and temperature sensors, is first exported and plotted to verify the assumptions of kinematics of melting and flow inside the nozzle. Figure 12 shows the raw data of prints using PLA filament performed with three temperature profiles from 200 °C (right) to 230 °C (left). The filament speed was successively increased until a significant decrease in the part quality was detected. This could be either delamination between layers, changes in wall thickness due to an unsteady feed of melted material or a catastrophic print failure. This can be experienced when the feeding unit is reaching a certain threshold in throughput and slippage occurred between the feeding wheel and the filament.

It can be seen that especially in areas of lower speeds the data follows the model proposed by Bellini et al. [10]. At higher filament speeds however, the melting and flow modes transition to the model *melting with pressure flow removal* [11] as previously assumed in Fig. 4. A shift of the curves with higher temperatures to the left is also clearly visible, showing a decrease of force at constant filament speeds.

A scattering of the datapoints is also observed in all prints with filament speeds exceeding the maximum range. This behavior can be seen especially when printing PLA material with nozzle temperatures below 215 °C (middle and right). Observations show that the scattering of data points is due to filament slipping in the feeding unit, especially when reaching a certain speed. Smaller deflections in the interaction of the filament, the feeding unit and the speed signal also result in scattering of data points around the mean. Improving the signals quality in this study will be within the range of post-processing tools for data analysis and not based on hardware

improvements. While the datasets with 200 °C and 215 °C show this behavior above 5 mm/s, prints performed with 230 °C have a higher resistance against filament slipping. This can be attributed to the decreased forces while using equivalent filament speeds.

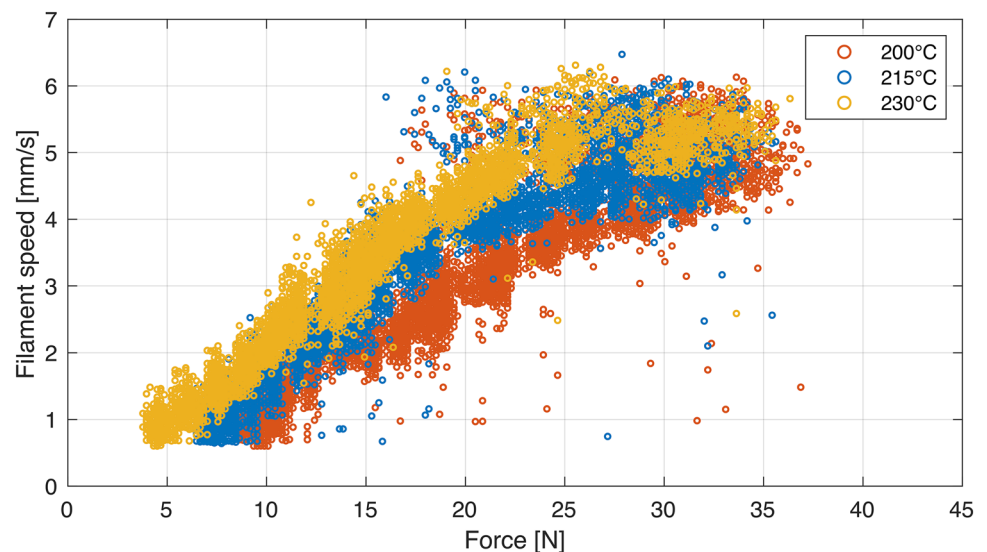
The raw datasets were processed before training the model to generate accurate results. Therefore, data recorded incorrectly with speed values below 0 mm/s and above 5 mm/s, as well as forces below 0 N and above 35 N were removed from the dataset. Finally, the dataset containing the pre-processed data is shuffled to avoid any dependency on the order of the final results.

In a next step, filament speeds and nozzle temperature of 230 °C were fed to the trained neural network as input data (Fig. 13). The trend between the *slow printing model* [10] and the *fast melting with pressure flow removal model* [11] can be observed well, showing a clearly visible transition point around the filament speed of 4.1 mm/s and a force of 17 N.

The results can also be seen for predictions made with the 200 °C dataset, used as a lower bound for nozzle temperatures. Here, a transition can be seen around 4.10 mm/s of filament speed and 25 N of force. For the purpose of providing an optimal overview, only data generated with 200 °C and 230 °C nozzle temperature is used in Fig. 13. The contrast between resulting forces at equivalent filament speeds increases with higher speeds and a maximum variation of about 9 N at 4 mm/s.

Finally, the ANN model was tested to predict filament force sensor response during one whole 3D printing job that lasted 3000 s. The simple cylindrical geometry presented in Fig. 6 was chosen for the printing job. The same PLA used in the other tests was employed, and the nozzle was set to

**Fig. 12** FFF print data recorded with filament force and -length sensors

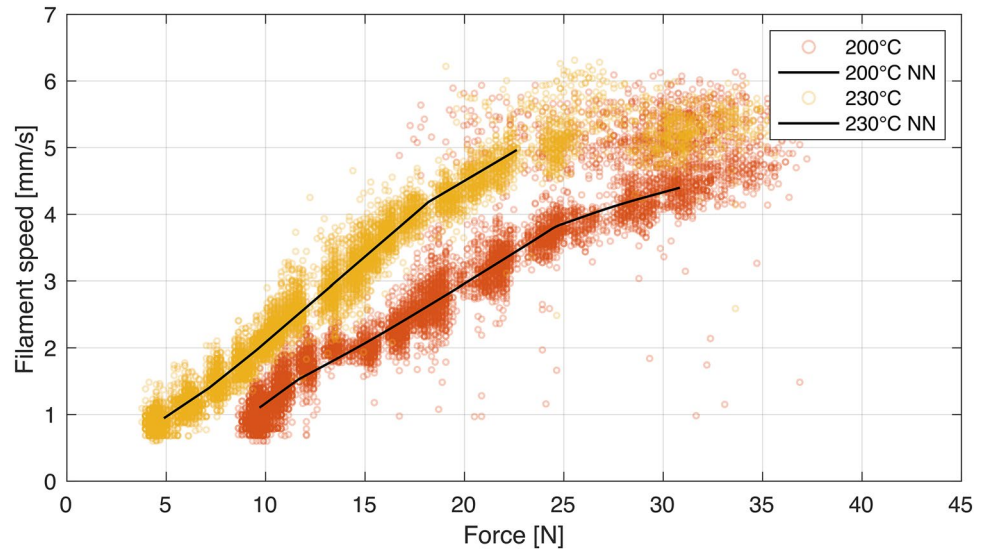


230 °C. Figure 14 presents the recorded filament velocity and force sensor output data during the 3000 s printing job.

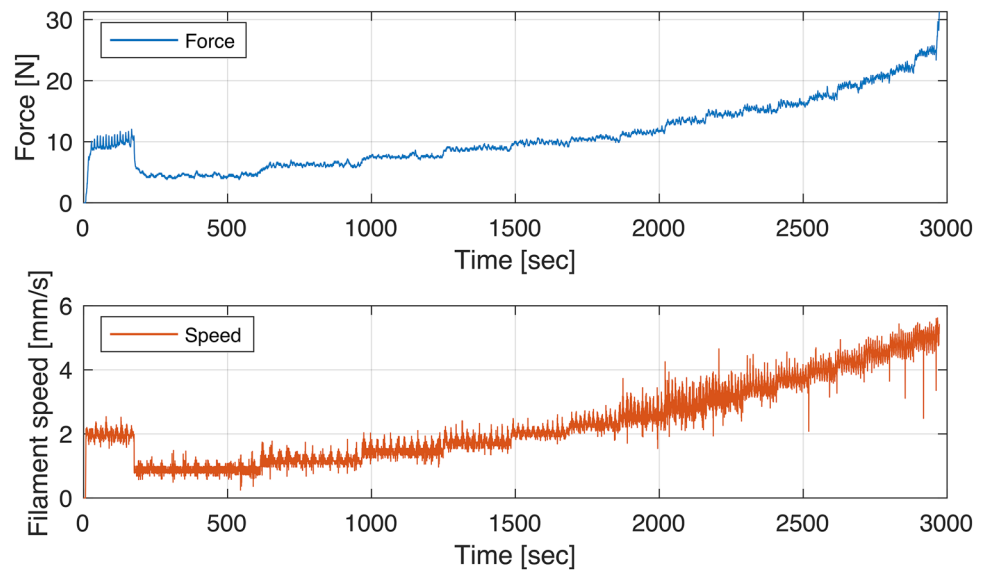
To test the ANN model, the unprocessed filament velocity data was given as input data, and using the set temperature

of 230 °C, the NN was used to predict the axial filament force. Figure 15 presents the comparison of the measured and NN-predicted force required to drive the filament at the given speed. It is interesting to note that the NN prediction

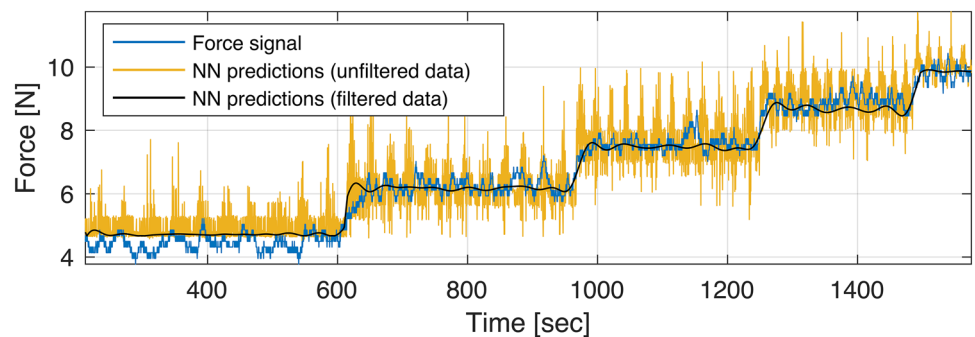
**Fig. 13** Neural network model predictions for 230 °C (left) and 200 °C (right)



**Fig. 14** Filament force and velocity sensor data during printing of the cylinder



**Fig. 15** Comparison of the measured force signal with NN-predicted force using filtered and unfiltered speed signal data



reflects the variability reflected by the unprocessed velocity sensor data.

Due to the nature of the feeding unit and the derivative of the length signal to the speed signal, reflections in the data can be observed. In order to reduce the variability of the predicted force signal, simple signal processing techniques have been applied to process the original input speed data. Therefore, the signal has been analyzed in the frequency domain using Fourier transformation. The information was then used to design and apply a low pass filter to the signal. The filtered speed data was again used as input data for the NN model and predictions were plotted against the unprocessed data shown in Fig. 15. The graph shows a major improvement in results using the filtered signal. However, it is clear that with any given filament velocity and nozzle temperature, the ANN model can accurately predict the required axial filament force.

## 6 Conclusion

The artificial neural network model developed for monitoring and predicting the force within the nozzle of an FFF printer, using filament speed and nozzle temperature as input data, accurately reflects the physical behavior within the nozzle. The model uses a neural network, where sensor data collected in real time can be pre-processed prior to the training and validation of the NN model, achieving an accuracy of 0.72 N. Comparing the results with existing mathematical models verifies previous theories, and is able to combine those into a single powerful model that is able to characterize the relationship between process -speed, -temperatures and -forces. The model can ultimately be used to choose process parameters for printing based on the gained process knowledge.

The neural network was able to capture the two modes of melting and flow proposed by various researchers, helping to shed some light on the underlying physics that control the heat transfer and flow within FFF nozzles. Acting as a digital shadow, the ANN meets the requirements of process monitoring and control during production, being cost effective and fast with low computational effort. The combination of AM and AI enables a new family of high-performance technologies, establishing AM 4.0, and further solving complex relationships with excellent results.

Furthermore, the process monitoring and control procedures in this work can be extended with predicting structural strength of printed parts, within the context of a failure surface development for FFF, first presented by Mazzei Capote et al. [25]. While generating a failure surface, complex testing methods using combined multiaxial loadings have to be conducted. Predicting the shear strength for various combinations of tensile- and compressive stresses, testing can be

simplified without the need of special equipment for applying combined loading stress states. This allows ultimately to generate a powerful model, that captures the entire process of selecting the optimum parameters, the actual printing as well as the component quality with mechanical properties.

While applications of artificial intelligence are already gaining ground in the field of additive manufacturing, there is no use so far in FFF desktop applications contributing extended process knowledge to the state of the art. This study has shown great potential for generating a deeper understanding of these complex processes and is considered to enable a process monitoring- and control solution for a wide range of AM technology, capturing applications from desktop- to high-end solutions used in production industries. The model is also capable of replacing the force sensor unit when properly trained for a range of temperatures and materials for a printer, making the technology accessible to the general public.

The implementation of this model to existing printers will allow the user to model the process which will lead to a better understanding of the process windows for various materials and printing parameters such as nozzle temperature and filament speed. Additionally, this model will allow the user to verify the quality of the print by continuously monitoring the printing conditions.

**Funding** Open Access funding enabled and organized by Projekt DEAL.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

1. Charles WH (1986) Apparatus for production of three-dimensional objects by stereolithography. Priority date August 8, 1984 (1986). US Patent 4,575,330
2. Wohlers T, Caffrey T, Campbell RI, Diegel O, Kowen J (2019) Wohlers Report 2019: 3D printing and additive manufacturing state of the industry. Annual Worldwide Progress Report, Wohlers Associates
3. Lee K-F (2018) AI superpowers: China, Silicon Valley, and the new world order. Houghton Mifflin Harcourt: 9–14
4. Obst P et al (2020) Investigation of the influence of exposure time on the dual-curing reaction of RPU 70 during the DLS

- process and the resulting mechanical part properties. *Addit Manuf* 32:101002
5. Goodfellow I, Yoshua B, Aaron C (2016) *Deep learning*. MIT press, Cambridge
  6. Popova E, Rodgers TM, Gong X et al (2017) Process-structure linkages using a data science approach: application to simulated additive manufacturing data. *Integr Mater Manuf Innov* 6:54–68. <https://doi.org/10.1007/s40192-017-0088-1>
  7. Qi X et al (2019) Applying neural-network-based machine learning to additive manufacturing: current applications, challenges, and future perspectives. *Engineering*. 5(4):721–729
  8. Bayraktar Ö et al (2017) Experimental study on the 3D-printed plastic parts and predicting the mechanical properties using artificial neural networks. *Polym Adv Technol* 28.8:1044–1051
  9. Wu H, Zhonghua Yu, Wang Y (2017) Real-time FDM machine condition monitoring and diagnosis based on acoustic emission and hidden semi-Markov model. *Int J Adv Manuf Technol* 90(5–8):2027–2036
  10. Bellini A, Güçeri S, Bertoldi M (2004) Liquifier dynamics in fused deposition. *J Manuf Sci Eng* 126:237
  11. Osswald TA, Puentes J, Kattinger J (2018) Fused Filament Fabrication melting model. *Addit Manuf*. <https://doi.org/10.1016/j.addma.2018.04.030>
  12. Ramanath HS, Chua CK, Leong KF, Shah KD (2008) Melt flow behavior of poly-ε-caprolactone in fused deposition modelling. *J Mater Sci* 19:2541
  13. Turner BN, Strong R, Gold SA (2014) A Review of melt extrusion additive manufacturing processes: I. Process design and modeling. *Rapid Prototyp J* 20/3:192
  14. Heller BP, Smith DE, Jack DA (2016) Effects of extrudate swell and nozzle geometry on fiber orientation in Fused Filament Fabrication nozzle flow. *Addit Manuf* 12:252
  15. Stammers E, Beek WJ (1969) The melting of a polymer on a hot surface. *Polym Eng Sci* 9(1):49–55
  16. Yardimci M, Güçeri S, Danforth SC (1997) Thermal analysis of fused deposition, 1997 international solid freeform fabrication symposium
  17. Go J, Schiffres SN, Stevens AG, Hart AJ (2017) Rate limits of additive manufacturing by fused filament fabrication and guidelines for high-throughput system design. *Addit Manuf* 16:1–11
  18. Murphy, Kevin P (2012) *Machine learning: a probabilistic perspective*. MIT press, Cambridge
  19. Frochte J (2019) *Maschinelles Lernen: Grundlagen und Algorithmen in Python*. Carl Hanser Verlag GmbH Co KG, München
  20. Refaeilzadeh P, Tang L, Liu H (2009) Cross-validation. *Encyclopedia Database Syst* 5:532–538
  21. Heaton J (2008) *Introduction to neural networks with Java*. Heaton Research, Inc.:157–159
  22. Mohri M, Afshin R, Ameet T (2018) *Foundations of machine learning*. MIT press, Cambridge
  23. Srivastava N et al (2014) Dropout: a simple way to prevent neural networks from overfitting. *J Mach Learn Res* 15:1929–1958
  24. Bishop CM (1995) *Neural networks for pattern recognition*. Oxford University Press, Oxford
  25. Mazzei C et al (2019) Failure surface development for ABS fused filament fabrication parts. *Addit Manuf*. <https://doi.org/10.1016/j.addma.2019.05.005>

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.