KNOWLEDGE-BASED OPTIMIZATION OF ARTIFICIAL NEURAL NETWORK TOPOLOGY FOR PROCESS MODELING OF FUSED DEPOSITION MODELING

Hari P.N. Nagarajan  
Tampere University of Technology  
Tampere, Finland

Hesam Jafarian  
Tampere University of Technology  
Tampere, Finland

Azarakhsh Hamedi  
Tampere University of Technology  
Tampere, Finland

Hossein Mokhtarian  
Tampere University of Technology  
Tampere, Finland

Romaric Prod’hon  
STEIM  
Chevremont, France

Shaima Tilouche  
Ecole Polytechnique Montreuil, Quebec, Canada

Eric Coatanéa  
Tampere University of Technology  
Tampere, Finland

Vladislav Nenchev  
Dynavio Cooperative  
Helsinki, Finland

ABSTRACT  
Additive manufacturing (AM) continues to rise in popularity due to its various advantages over traditional manufacturing processes. AM interests industry, but achieving repeatable production quality remains problematic for many AM technologies. Thus, modeling the influence of process variables on the production quality in AM can be highly beneficial in creating useful knowledge of the process and product. An approach combining dimensional analysis conceptual modeling, mutual information based analysis, experimental sampling, factors selection, and modeling based on Knowledge-Based Artificial Neural Network (KB-ANN) is proposed for Fused Deposition Modeling (FDM) process. KB-ANN reduces the excessive amount of training samples required in traditional neural networks. The developed KB-ANN’s topology for FDM, integrates existing literature and expert knowledge of the process. The KB-ANN is compared to conventional ANN using prescribed performance metrics. This research presents a methodology to concurrently perform experiments, classify influential factors, limit the effect of noise in the modeled system, and model using KB-ANN. This research can contribute to the qualification efforts of AM technologies.

Keywords: Additive Manufacturing, Fused Deposition Modeling, Dimensional Analysis Conceptual Modeling, Artificial Neural Networks, Knowledge Management.

INTRODUCTION  
Major technological and industrial advancements in manufacturing (e.g., additive manufacturing, cloud computing, nano-manufacturing, and advanced materials) have brought about great paradigm shifts in the way products are designed and manufactured. Additive manufacturing (AM) research has enabled the growth of innovative techniques and functional products, framing AM as feasible alternative to subtractive and formative techniques [1]. AM processes are being adopted at an ever-increasing pace for mainstream manufacturing. Particularly, polymer extrusion technology, such as, fused deposition modeling (FDM) are among the most well researched and most widely used AM processes. The FDM process involves successive melting, extrusion, deposition and solidification of thermoplastic polymer melts [2]. The extrusion process is still a source of process innovations, and new technologies are being developed using this approach for metal printing using metal and polymer matrix. The FDM’s thermal process involves storage of thermal energy in the molten material, distribution of this energy into the part.
through a thermal conduction process, and dissipation in the part by convection cooling. The redistribution of the thermal energy ensures the bonding between layers. Several methods exist for thermoplastic delivery in the process, namely, liquefier for self-extruding filament, fluid metering rotary pumps, and high pressure plunger system [3,4]. The liquefier based material delivery method is dominant in most FDM machines. In this research, material delivery using a liquefier, which employs a self-extruding filament, is modeled.

For FDM parts, the cross-sections of the deposited layers is shaped through the direct flow of polymer melt between the previous layer and the printing nozzle. This results in shapes having the form of flattened ellipsoids. Since the 80s’, process models have been developed for understanding the complex behaviors taking place in FDM, such as, thermal transfer, layer creation and bonding process. Existing research on FDM modeling have focused on the cooling behavior of single and multiple filaments, thermal behavior of liquefier, analysis of melt front location, degree of cooling in the nozzle and impact of its design on operational stability, temperature distribution across different configurations, and impact of built file [3,5,6]. This available knowledge provides a set of dispersed sub-models supporting the understanding of localized phenomena, but does not provide an overall system perspective and a global model. Thus, it raises two main questions for qualification of the FDM technology in mainstream manufacturing, such as; i) Are the part requirements achievable with the FDM technology? ii) What are the optimal manufacturing parameters that needs to be selected to achieve required part specification? The existing localized models cannot be effectively used in closed loop control of the FDM machines. Thus, metamodeling approaches must be evaluated to achieve effective modeling and control of the FDM process [7].

Artificial Neural Networks (ANN) as a modeling strategy has been widely used to approximate complex functions. In this context, ANNs can be considered as one type of metamodeling approach [8–11]. ANNs are utilized in numerous domains, and they form the backbone of deep learning algorithms. The main challenges of developing and implementing an ANN is that it demands a large number of training data. Moreover, the architecture of an ANN for a specific purpose is unknown and it costs extra training to explore and progressively generate a suitable architecture via the weights allocated to each of the edges within an ANN. Hence, ANNs have lost their lustre as a metamodeling approach in the past two decades [12]. This is specifically the case when limited amount of training data are available or if the system to be modeled is subject to large variability due to its complexity. Deep learning approaches can be used for those large complex systems but the duration of the training can be extremely long (with up to several years) and extremely costly [12–14]. The amount of training data required is sometimes implying to resources solely available in large corporations. Nevertheless, ANN topology is always depicting a similar pattern and available system knowledge is not considered for designing the ANN topology.

In this research, a methodology to design an optimal ANN topology using available system knowledge is presented. A sampling approach and a selection process to qualify the most influential process variables are proposed. The design of the ANN topology integrates existing process knowledge using the Dimensional Analysis Conceptual Modelling (DACM) framework, which simplifies and visualizes the cause-effect relationships between process variables during early design phase [15]. In addition, the mutual information is computed to detect inter-relationships between process variables in the training data sets for the ANN. This conjoint experimental and modeling approach is compared to a traditional ANN design, and the knowledge-based ANN (KB-ANN) is applied to the modeling of the FDM part. The research is organized as follow: Section 2 provides background on the experimental procedure used in the study and the approaches considered to encode knowledge for optimizing the ANN topology. Section 3 describes the case study and the application of the developed methodology for the case study. Section 4 discusses the key results of the study and Section 5 includes the conclusions of the work, briefly describes limitations and future development efforts.

**BACKGROUND**

**Design of Experiments**

Performing experiments by varying one-factor-at-a-time is cost intensive. Thus, design of experiments (DOE) proposes a set of principles to maximize the efficiency of experiments by minimizing the number of experiments to be conducted. One of those principles is the use of factorial experiments. Factorial DOE explore all the possible combinations of factors and levels [16]. In AM, the number of factors influencing the part quality is potentially huge and it is difficult to explore all the potential combinations of these possible factors. Sampling, which is the use of a subset of the experimental space, is consequently required to explore this space at an acceptable cost [17,18]. Furthermore, one of the fundamental issues with AM technologies is poor repeatability [19,20]. One plausible reason is the existence of noise in the system, which have a significant effect on the factors influencing part quality. Thus, the effect of these noise variables in the system must be investigated during experiment through quality analysis. Fractional factorial designs, such as Plackett-Burman design or Taguchi’s Orthogonal Arrays are proven to be useful for quality analysis [16,21,22]. In this research, Taguchi’s orthogonal design is implemented to study the noise in the system, by extending the experimental design with another orthogonal array simulating the random environment [22].

Additionally, poor repeatability in AM can be associated with the need to control the key variables of the process during
printing, through a closed loop system. In FDM, a difference in the latency can be observed for different variables, which affect the choice of control strategy implemented for the system. In order to reduce variation due to latency issues, signal to noise ratio can be calculated for each control variable with high latency. The high latency control factors can be fixed at a level ensuring a maximization of the signal to noise ratio [23]. The low latency control factors should then be controlled dynamically during the printing process. This time intensive computation can be avoided by implementing a metamodelling strategy using ANNs, to be applied to the control factors.

**Metamodelling using Artificial Neural Networks**

A metamodelling approach can provide numerous advantages for the FDM process and AM in general. It can enable the development of global predictive models integrating numerous parameters. Furthermore, it can support the implementation of a closed loop control system to improve part quality and process repeatability. However, the cost of experiments for learning algorithms in ANNs is very high. Thus, minimizing the need for large amount of experimental data sets is essential. Through knowledge extraction and management, we can limit the need for experimental data sets by integrating the existing system knowledge available for the observed process into the ANN. Knowledge extraction enables the reuse of existing literature knowledge for manufacturing systems’ design and planning [15,24]. Nevertheless, the existing knowledge in literature is represented in multiple forms and lack interoperability [25]. For this reason, the DACM framework is utilized to translate the different knowledge to be compatible, and visualize the cause-effect relationships in the form of causal graphs.

**Dimensional Analysis Conceptual Modeling**

Coataneea et al. [15] developed a method to extract and encode knowledge associated to system architectures, equations and measuring units. The encoded knowledge is represented in the form of causal graphs. DACM can be an efficient approach to the creation of surrogate models and for adaptively training an ANN. The modeling starts with a designation of the system boundary and definition of model’s objectives. Function representation is used to represent the sequence of functions taking place in the system of interest. Those sequences of functions describe the different behavior of the observed system. DACM transforms the initial function model into a generic functional model formed around a limited set of fundamental functions and uses the causal rules extracted from Bond Graph theory [26–28]. The dimensional analysis is applied to each node of the graph to form behavioral equations. A color pattern is applied to the different variables to highlight their different design nature. The primary result of this modeling is a colored hypergraph with a list of governing equations for the system of interest. The model can then be used for qualitative or quantitative simulations, and search for contradictions. Figure 1 visualizes the different steps in the DACM Framework. The DACM modeling process ends when a computable model of the system of interest is available at the required level of detail.

![Figure 1: Visual representation of DACM Framework approach](image)

Generic functions represented by Bond graph organs are used as an intermediate level between the classical functional models and the final causal graphs [27,28]. To facilitate the systematic assignment of variables to the generic functional representation, regardless of the energy domain, the variables are classified into five generalized categories, namely, Flow, Effort, Momentum, Displacement, and Connecting [29]. The mathematical relationship between generic variables describes how those variables relate to each other. The sequence of functions in the functional modeling provides an initial insight into the global causality. Mapping the functions to the generic functional elements enables the extraction of the causality among the variables characterizing those functions. Table 1 summarizes the causal rules in the DACM approach. Figure 2 represents a causal extraction algorithm used to automate the
DACM modeling process. First, the algorithm checks if a generic functional organ is defined for each functional box. Then, the algorithm explores each functional box from its source to the end of the model to verify that there is no conflict in the coherence of the generic functional representation in terms of causality.

Table 1: Causality for the generic functions and associated Bond Graph elements

<table>
<thead>
<tr>
<th>Bond Graph Element</th>
<th>Schematic View</th>
<th>Bond Graph Element</th>
<th>Schematic View</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source of effort (S)</td>
<td>Fixed effort-out causality</td>
<td>Source of flow (F)</td>
<td>Fixed flow-out causality</td>
</tr>
<tr>
<td>Capacitor (C)</td>
<td>Fixed effort-out causality</td>
<td>Inertia (I)</td>
<td>Fixed flow-out causality</td>
</tr>
<tr>
<td>Resistor (R)</td>
<td>Preferable effort-out causality (Resistive)</td>
<td>Resistor (R)</td>
<td>Preferable flow-out causality (Conductive)</td>
</tr>
<tr>
<td>Transformer (TF)</td>
<td>Maintain incoming causality (two-port element)</td>
<td>Gyrator (GY)</td>
<td>Switch incoming causality (two-port element)</td>
</tr>
<tr>
<td>Effort junction (EF) or (O) (Multipoint element)</td>
<td>( e_1q_1 + e_2q_2 + e_3q_3 + e_4q_4 = 0 )</td>
<td>Flow function (FF) or (I) (Multipoint element)</td>
<td>( f_1x_1 + f_2x_2 + f_3x_3 + f_4x_4 = 0 )</td>
</tr>
</tbody>
</table>

Finally, according to the categories of assigned variables and using the causal rules (Table 1), the cause-effect relationships between variables are established.

![Figure 2: The causal extraction algorithm](image)

In addition to DACM, several other mathematical knowledge extraction approaches have also been proposed in literature. Segreto et al. [30] uses principal component analysis to extract characteristic features from acquired sensor signals. They apply this knowledge to neural networks for chips classification. Similarly, knowledge discovery approaches, derived from statistics such as Bayesian networks combined with an IT prototype of a design assistant system, demonstrated a high potential for supporting the design of improved product generations [31]. Among the statistical based mathematical knowledge discovery approaches, mutual information has been widely demonstrated for encoding knowledge in machine learning applications. Mutual information is a measure of the amount of information one random variable contains about another [32,33]. The advantage of mutual information is its ability to detect both the linear and non-linear dependencies of the observed variables. Battiti [32] investigates the application of the mutual information criterion to evaluate a set of candidate features and to select an informative subset to be used as input data for a neural network classifier. It has been used as a feature selection method for machine learning techniques. It has been combined with machine learning techniques such as neural networks and has demonstrated good usability in both classification and regression tasks. These combined techniques have been applied in different domains. Bowden et al. [34,35] used mutual information to remove collinearity among input variables. This dimension reduction technique was combined with support vector machines, linear discriminant analysis, and neural networks to predict water salinity in Murray River in Australia. May et al. [36] combined ANN with a nonlinear variable selection technique, based on mutual information, to build a prediction model tested on some synthetic data. While the current use of mutual information remains limited to feature selection, its use can be further investigated to reduce the complexity of machine learning algorithms and help defining their topologies. This research uses the computed matrix of mutual information to measure the interactions or dependences between variables. The mutual information is used to validate and update the causal relationship developed using DACM.

**Mutual Information Computation**

Mutual information between two variables can be computed as follows [37]:

\[
MI(x_1, x_2) = \int_{x_1} \int_{x_2} \log \left( \frac{f(x_1, x_2)}{h(x_1)h(x_2)} \right) h(x_1)h(x_2)dx_1dx_2
\]

(1)

Where, \( f(x_1, x_2) \) represents the joint distribution function, \( h(x_1) \) and \( h(x_2) \) are the marginal probability functions of the two variables. To calculate the mutual information using Eq. 1, the probability distribution of both variables must be known. Since in most real world cases, similar to this study, the dataset is obtained from experiments, the probability distribution of variables are unknown. In this case, the variables must first be classified as either being discrete or continuous depending on the available dataset. If both variables are discrete, a counting approach can be used to calculate the frequency of occurrence.
of each \((x_1, x_2)\) pair in the dataset and the mutual information can be estimated using Eq. 2, [38];

\[
MI(x_1, x_2) = \sum_x \sum_{x_2} \log \left( \frac{p(x_1, x_2)}{p(x_1)p(x_2)} \right) p(x_1)p(x_2)
\]  

(2)

For the case where one or both of the variables are classified as continuous, several direct approaches to estimate the mutual information have been presented in literature [39]. The most commonly used approach is variable discretization, wherein the continuous data is aggregated into several bins and the frequency of occurrence for each state is counted to estimate the mutual information using Eq. 2. However, this discretization method can produce biases in the mutual information value computed. For this study, the bias is avoided by using a validated estimation formula developed by Ross [38]. In [38], first, for each data point \(i\) in the dataset of the two variables \(x_1\) and \(x_2\), the estimator first considers a subset of length \(N_i\) for the continuous variable, which corresponds to the discrete variable value of the data point \(i\). Second, a plot of discrete variable \(x_i\) to \(N_i\) is generated and number of continuous variables that lie within the \(k^{th}\) neighbor’s distance of data point \(i\) is estimated as \(d\). Third, the estimator finds the number of neighbors, \(m\), within the full dataset of the continuous variable, which are in the same distance, \(d\). Thus, a number \(I_i\) is calculated using Eq. 3,

\[
I_i = \psi(N) - \psi(N_i) + \psi(k) - \psi(m)
\]

(3)

Where \(\psi\) is the digamma function [40]. The value of mutual information for the whole dataset would be the average of \(I_i\) over all data points represented by Eq. 4,

\[
I(X, Y) = \frac{1}{N} \sum_i I_i = \psi(N) - \psi(N_i) + \psi(k) - \psi(m)
\]

(4)

The matrix of mutual information and the causal graph developed using DACM are used to design the ANN topology.

**Artificial Neural Network based Process Modeling**

ANNs with numerous architectures and training algorithms are utilized in process modeling and forecasting output variables. ANNs with the assistance of data standardization, data preprocessing, and model performance optimization, have become a key enabler in modeling different processes. The main advantages of ANNs in modeling when compared to other process modeling methods are, (i) its ability to handle noisy and ambiguous data, (ii) lower cost of implementation than empirical models, (iii) their suitability for accurate representation of dynamic problems, and (iv) the ability to provide novel solutions for complex systems inputs-outputs [41]. However, it is only possible to perform black box modeling using conventional ANNs. Hence, there is limited information about the hidden layers and relations between the layers. This can result in overfitting models due to the empirical nature of ANNs [42]. In addition, ANNs require large number of experimental tests for training which demands high computational power to reach a fitting model. Thus, research must be focused on designing ANN architectures that are transparent and require less computational power to improve cost effectiveness.

**Knowledge-based Artificial Neural Networks**

A Knowledge based artificial neural network (KB-ANN) is a hybrid-learning network that uses both theoretical knowledge and empirical data to construct an appraisal model. Knowledge extraction and encoding in a KB-ANN can enable superior interpolation and extrapolation that estimates unmeasurable parameters [43]. The main aim of the KB-ANN is to apply, transfer, and translate the knowledge into a hybrid neural network. This allows for consolidation of already existing knowledge into the modeling process to develop a globalized model. The architecture of a KB-ANN can be constructed using a rules-to-network translation algorithm [44]. The rules-to-network translation establishes the rules between modeled parameters and input variables using existing knowledge of the system. However, the direct translation of existing models and theoretical knowledge of the observed system is taxing. This is due to the lack of interoperability of different forms of knowledge in literature. In this study, the DACM network is used to eliminate this issue and translate the causal relationship into the rules. The established rules are then converted into a neural network topology following these two constraints;

a) Rules must be propositional to improve prediction capability. Including predicate variables can affect the accuracy of the developed network.

b) Rules must be hierarchical and must have information about the intermediate variables that connect the input to the output.

A conventional ANN and a KB-ANN are developed for the case study to model a FDM process.

**CASE STUDY- FUSED DEPOSITION MODELING**

In this section, the authors demonstrate that an exploration of the manufacturing space can be effectively associated with artificial neural network modeling to improve AM parts’ qualities, while keeping the required data sets to a low level. The printed FDM part used in this case study is shown in Figure 3.

The part specification studied in this case study has a wall thickness \(e = 0.5+/-0.05\) (mm) constant for the entire profile. A concurrent experimental and modeling process is presented for the process.
The concurrent modeling and experimental method is decomposed into seven steps summarized below (Figure 4).

**Step 1:** Four initial printing tests are completed using pre-selected parameters proposed by the slicing software (Repetier) selected for the case study.

**Step 2:** The initial prints are used to detect key potential defects and zone of defects on the printed geometry.

**Step 3:** The printing process parameters are analyzed and potential causes for the observed defects are extracted. A list of possible parameters are selected by the team of experts.

**Step 4:** A rapid evaluation of the effects of those parameters on the detected defects is performed with few supplementary experiments implying a simultaneous variation of the parameters, and using two orthogonal arrays. One orthogonal array for varying the controlled parameters and one for the noise factors.

**Step 5:** The most significant parameters to achieve the expected thickness $e = 0.5 +/− 0.05$ mm are selected for developing a predictive model. The other parameters are fixed to a level that maximizes the Signal/Noise ratio to target value using Eq. 5, [45]:

$$\eta_i = 10\log \left[ \frac{Y_i^2}{\sigma_i^2} \right]$$  \hspace{1cm} (5)

Where, $\eta_i$ represents the signal to noise ratio, $Y_i$ represents the target value, and $\sigma$ represents the variance.

**Step 6:** The prediction model of the thickness ($e$) is built for the remaining control factors, nozzle travel speed (TS) and layer height (hi). A colored causal graph is first developed using the DACM approach to encode the literature knowledge and the Mutual information is applied to the data set. The variables are classified into four main classes (i.e. colors):

- **Exogenous** variables (shown in black) are outside the system border and part of the environment of the system. They cannot be modified by the designer but are imposed to the system. **Independent** design variables (shown in green) are not influenced by any other variables in the system. Their value can be modified by designers. The nozzle travel speed (TS) and layer height (hi) are considered independent. **Dependent** design variables (shown in blue) are influenced by other variables such as exogenous and independent variables. It is more difficult to modify and control the dependent variables. Finally, **performance** variables (shown in red) are the objective variables. They usually belong to the category of dependent variables as well. They are selected by the designers to evaluate the performance of a system. Model thickness ($e$) is selected as the performance variable. The developed causal graph with mutual information is then translated into an ANN topology. A conventional ANN and a KB-ANN topology is evaluated in this study. The ANN’s topology is designed for maximum compactness to maintain all the connections in the causal graph. The KB-ANN topology is developed using the following algorithm:

1) Organizing the rules based on causal relationships and mutual information with no disjunction amid the relations.
2) Construct hierarchical relations for the rules.
3) Appointing levels for each node in the ANN.
4) Adding hidden nodes and unknown relations as a network module.
5) Importing input nodes into KB-ANN architecture.
6) Assigning synapses between nodes in the KB-ANN topology.
7) Composing a hybrid KB-ANN topology.

**Step 7:** The conventional ANN and the KB-ANN are compared for performances and prediction capability.

**RESULTS AND DISCUSSION**

In steps 1, and 2, initial experimental prints were performed and the defects in the printed part were detected. The observed defects are shown in Figure 5.
In step 3, the most influencing factors are taken into consideration, the layer height (hi) in mm, the extruder temperature (Tset) in °C, the nozzle travel speed (TS) in mm/s and the fan speed (Fan) in rpm.

In step 4, the four initial input parameters hi, Tset, TS, and Fan influencing the wall thickness (e) are analyzed at three possible levels. A ‘3³’ standard orthogonal array implying nine experiments was conducted for the controlled parameters. Three noise factors were included in the DOE at two levels and their influence was conjoined. The three noise factors considered are (i) variation of the printing trajectory (lines and radiuses), (ii) variation of the room temperature, and (iii) liquefier clogging. A ‘2³’ orthogonal array was selected for varying the noise factors. The conjoint orthogonal DOE plan is shown in Figure 6.

In step 5, the variations of the extruder temperature (Tset) and the Fan are removed from the model because of the latency of their effects on the wall thickness (e). Those parameters are fixed to a value minimizing the effect of the noise on the model by maximizing the signal/noise ratio to target as explained in the previous section. The values for Tset is fixed at 210° C and Fan as ON at 50%. In step 6, the prediction model for thickness, e is developed. The causal graph developed using DACM framework for FDM is presented in Appendix A. The independent variables TS and hi are represented in green. The intermediate or dependent variables of the system are represented in blue. The performance or target variable is represented in red. The grey boxes in the causal graph represent areas where the available knowledge discovered is not sufficient to draw direct causal relationships between variables. The different nodes of the causal graph from DACM are connected using black arrowed lines, where the arrows represent the direction of the causality. The mutual information between the independent variables and performance variables was also computed.

These connections are represented in the causal graph as blue arrowed lines. From the developed causal graph structure, two types of ANNs are developed, namely, conventional ANN and KB-ANN. The two ANNs are developed to accept thickness (e) as input and predict the layer height (hi) and the nozzle travel speed (TS) for optimization. The KB-ANN is designed with 13 hidden layers and a maximum of 17 neurons per layer. The conventional ANN is a fully connected 13x17 ANN. The Figures 7a and 7b presents the two topologies of the ANNs. It is visible from Figure 7b that the amount of weights to be trained is much smaller for the KB-ANN when compared to conventional ANN.

The dataset used for the training and validation is composed of 5x9 design of experiments following the DOE presented in Figure 6, with five repetitions. A 70/30 split of the data is performed for training and validation of the ANNs. Table 2, compares the two ANNs developed in this study.
It is seen that the KB-ANN has a higher average trained cost when compared to a conventional ANN. This is due to the high amount of pre-processing required to convert the causal graph structure into the KB-ANN topology.

| Table 2: Comparison of KB-ANN and Conventional ANN |
|-----------------|-----------------|------------------|
|                  | KB-ANN | Conventional ANN |
| Average trained cost | 0.34 | 0.23 |
| Average training iterations | 8.03 | 651.92 |
| Average training evaluations | 17797.42 | 2835178.54 |
| Average validation cost | 0.99 | 0.36 |

The KB-ANN converges faster than the conventional ANN does, due to optimized node allocation using the causal relationships. Thus, it enables faster convergence to the local optimum with much lower training iterations and evaluations. However, the average prediction error in the KB-ANN model is slightly higher than the conventional ANN. The higher error is due to the pruned topology wherein, some nodes or synapses are missing due to unavailability of accurate knowledge. In addition, any errors or lacking knowledge from literature can also result in inaccurate allocation of synapses at certain nodes consequently leading to a higher validation cost. Thus, knowledge extraction (using DACM and mutual information) and encoding must be complete and accurate for effective allocation of nodes and synapses. The conventional ANN compensates for this issue of missing nodes by introducing extra nodes and synapses. These extra nodes assume the role of any undiscovered influencing factors of the process. Thus, this increases the amount of training data required for convergence.

The KB-ANN was used to predict the optimal value of $h_i$ and TS (KB-ANN optimization for $e = 0.5$ mm), and limiting the printing pattern (noise) to line for optimization. The network predicted $h_i = 0.29$ mm and nozzle travel speed $TS = 10.27$ mm/s for a line print pattern. These predicted parameters was used to print a line as shown in Figure 5. It can be seen that the parameter tuning significantly improved the quality of final part by eliminating the defects identified during the initial test prints with default parameter values from Repetier software. These parameters were predicted for each geometric pattern (i.e. line and radius) to maximize the specific signal/noise ratio.

**CONCLUSIONS**

The study developed a combined approach for experimentation and modeling using ANNs. The objective were to limit the amount of required experiments and to develop a meta-model capable of dynamically predicting the fast changing control factors of FDM. Benefit was gained from causal graph representation and mutual information, to be able to design KB-ANN with only the necessary weights to be computed. The results demonstrated the superiority of the KB-ANN over conventional ANN to converge with less training iterations. The validation experiments demonstrated visible improvements to the printing quality in FDM. The KB-ANN was associated with a high training cost due to the large amount of time required for knowledge extraction, conversion to rules, and encoding into the ANN. The case study was limited to the optimization of one target variable, in comparison to the large number of target variables that essentially need to be optimized for a complex FDM system. The current work should be seen as an initial proof of concept of techniques and approaches that we can use to combine knowledge in modular manner, to solve optimization problems in a modular manner and to reduce dimensionality of complex.
problems using knowledge extraction, representation, and integration techniques such as dimensional analysis theory.

LIMITATIONS

The current study developed ANNs with large size and with pre-trained weights. A possibility of vanishing gradient is probable due to pre-trained weights. The problem is found in training artificial neural networks with gradient-based learning methods and backpropagation. According to Hochreiter et al. [46], in such methods, each of the neural network's weights receives an update proportional to the gradient of the error function with respect to the current weight in each iteration of training. The problem is that in some cases, the gradient will be vanishingly small, preventing the weight from changing its value. In the worst case, this may completely stop the neural network from training.

In addition, a problem-independent way to choose a good network topology is not available in literature, although research in this direction have been active for years [47]. The size of ANNs is currently defined only by some heuristic rules. In [48] the authors provide a set of simple rules to define the rough size of a neural network.

- The number of hidden neurons should be in the range between the size of the input layer and the size of the output layer.
- The number of hidden neurons should be 2/3 of the input layer size, plus the size of the output layer.
- The number of hidden neurons should be less than twice the input layer size.

Following these rules will generate rather small ANNs. For example, if a problem is implying 111 input values and 3 output values, the total amount of hidden neurons proposed is 10. Those hidden neurons can be organized in different layers. If the amount of hidden neurons is too high, the risk is to face overfitting. If it is too low, the ANN might not obtain a proper fitting. If those rules are applied to the topology of the neural network proposed in this current article, it is clear that the fully connected ANN is containing far too many neurons, thus facing a risk of over-fitting.

FUTURE WORK

The authors are expanding the case study analysis to optimize a larger set of target variables using the current methodology. The DACM method in this study generated causal graphs that were used as elementary topology of ANNs. Such ANNs can grow quickly in size and be faced with the problems presented above in the limitations. Thus, a multi-level hierarchy approach to ANN topology creation as well as the use of regulators for the flow of values similar to long short-term memory neural networks is being investigated. In addition, the idea that the message lost when deep networks are used is more important than the loss in shallow neural network is considered, i.e., splitting of a big neural network into smaller chunks can solve the problem. ResNets developed by Microsoft research center yielded lower training error (and test error) than their shallower counterparts simply by reintroducing outputs from shallower layers in the network to compensate for the vanishing data [49]. For example, causal graphs such as the one presented in the appendix A can be seen as a combination of modules. Those modules being smaller ANNs that can be trained separately. The intermediate blue nodes, present in an explicit manner, probable sensor locations in the AM process in order to collect the required data needed to train the local ANNs. The authors are currently researching the above-mentioned improvements as well as qualification methods for improving the accuracy and completeness of extracted and encoded knowledge.

NOMENCLATURE

- \( h_i \) : Layer height (mm)
- \( TS \) : Nozzle travel speed (mm/s)
- \( dx/tx \) : Nozzle velocity in x direction for dx (mm/s)
- \( dy/ty \) : Nozzle velocity in y direction for dy (mm/s)
- \( FFR \) : Filament flow rate (mm/s)
- \( T_0 \) : Output Temperature (°C)
- \( T_w \) : Wall Temperature (°C)
- \( AV \) : Change in nozzle travel velocity (mm/s)
- \( T_i \) : Initial filament temperature (°C)
- \( T_{ref} \) : Reference temperature (°C)
- \( dy/dt \) : Velocity of molten polymer (mm/s)
- \( D_i \) : Diameter at \( i \)th section of liquefier nozzle (mm)
- \( k \) : Coefficient of conduction
- \( \theta \) : Dimensionless Temperature
- \( C_p \) : Heat capacity (J/kg/K)
- \( \rho \) : Density of Filament (kg/m³)
- \( \beta \) : Conical angle of liquefier geometry
- \( L_i \) : Length at \( i \)th section of liquefier geometry (mm)
- \( \mu \) : Viscosity of polymer filament (kg/m)
- \( \Delta P_i \) : Pressure drop (kg.m/s²)
- \( VFR \) : Volumetric flow rate of filament (mm³/s)
- \( \mu_i \) : Kinematic viscosity (m²/s)
- \( e \) : Intended set thickness (mm)
ACKNOWLEDGMENTS

The authors would like to acknowledge Dr. Henri Paris from G-SCOP laboratory, University of Grenoble Alpes, Di Wu, Vadim Tsarkov, and Siddhanth Modi from Simon Fraser University, Samuel Bassetto from Ecole Polytechnique, and Saoussen Dimassi from Tampere University of Technology for their support and invaluable input in the initial stages of this research.

REFERENCES


