A neural network-based build time estimator for layer manufactured objects

Article *in* The International Journal of Advanced Manufacturing Technology · November 2011

DOI: 10.1007/s00170-011-3284-8

2 authors:

Luca Di Angelo
University of Studi dell'Aquila, Italy

Paolo Di Stefano
Università degli Studi dell'Aquila

Int J Adv Manuf Technol (2011) 57:215–224
DOI 10.1007/s00170-011-3284-8

ORIGINAL ARTICLE

A neural network-based build time estimator for layer manufactured objects

Luca Di Angelo · Paolo Di Stefano

Received: 1 September 2010 / Accepted: 18 March 2011 / Published online: 5 April 2011

Abstract A correct prediction of build time is essential to calculate the accurate cost of a layer manufactured object. The methods presented in literature are of two types: detailed–analysis- and parametric-based approaches. The former require that a lot of data, related to the kinematic and dynamic performance of the machine, should be known. Parametric models, on the other hand, are of general use and relatively simple to implement; however, the parametric methods presented in literature only provide a few of the components of the total build time. Therefore, their performances are not properly suited in any case. In order to overcome these limitations, this paper proposes a parametric approach which uses a more complete set of build-time driving factors. Furthermore, considering the complexity of the parametric build time function, an artificial neural network is used so as to improve the method flexibility. The analysis of the test cases shows that the proposed approach provides a quite accurate estimation of build time even in critical cases and when supports are required.

Keywords Rapid prototyping · Build–time estimation · Artificial neural network

L. Di Angelo (✉) · P. Di Stefano
Department of Mechanical Engineering, Energy and Management, University of L'Aquila,
L'Aquila, Italy
e-mail: luca.diangelo@univaq.it

P. Di Stefano
e-mail: paolo.distefano@univaq.it

1 Introduction

Rapid prototyping (RP) is a technology for quickly fabricating physical models, functional prototypes and small batches of parts, by stacking two-dimensional layered features, directly from computer-aided design (CAD) data. RP plays an important role in both minimising the development time required for a new product and reducing costs [1, 2].

One of the current tendencies is for companies which offer rapid prototyping services to be equipped with numerous prototyping technologies so as to satisfy the wide range of customer needs. The various layer manufacturing technologies differ not only in the material used but also in the operating principle and the modalities by which the material is layered on each single slice of the prototype. The cost of an object made with rapid prototyping depends on a number of factors, some of which are easy to determine, while some others are more closely linked to the prototyping technology used. An important factor which affects cost is the time employed in object manufacturing. It affects the fixed costs of the manufactured object.

Some of the most significant factors which affect build time can be evaluated relatively easily; for example, the volume of the material or slice number. Other factors, however, are not so easy to predict. Such is the case of the object complexity which also requires, in the specific case of build time estimation, a way to be defined. The common practice for estimating build time is to assume it to be proportional to the total volume of the prototype. Actually, this approach provides only one component of the total build time. A detailed analysis of the forming tool movements which need to be made in order to manufacture a given object is necessary so as to correctly predict build time; nevertheless, this analysis cannot be performed by general purpose software. It is difficult to standardise a method for build time estimation which is independent from the specific type of machine.

In order to solve this problem, a parametric approach, which is based on an artificial neural network, is proposed to estimate build time. Most of the significant factors which affect build time are identified and then analysed. They constitute the input factors of the neural network which is purposely trained beforehand to estimate build time. When using a typical parametric approach, some problems are found both in the identification of the parametric function and in the estimation of the coefficients of the formula. In a neural network-based approach this task is performed at the training stage of the network, when the weight factors are determined. Choosing the set of training samples is a critical step, which greatly affects the performance of the neural network. In order to define the criteria to select the training samples and the sample size, a controlled trial-and-error approach is suggested. Furthermore, the method proposed in this paper can be adapted for the build time estimation of the main layer manufacturing technologies. It has been tried out through a series of test cases related to 3DP and FDM technologies and the results which have been obtained are discussed hereinafter. The chosen RP technologies are the most different ones in terms of parameters required to estimate build time.

2 Related works

Build time t_f depends on the characteristics of the layer manufacturing process, as well as on the prototype's geometry, its dimensions and the orientation in the RP machine [3]. It is difficult to accurately calculate build time without considering all the process parameters which affect machine movement, including acceleration and deceleration of the manufacturing tool (nozzle or laser beam) [4]. Build time can be determined as the sum of the forming time for each layer and the delay time between the subsequent layers' manufacture. Layer forming time depends on the tool movements which are required to form the *layer contours* and the *tool path loops* which are necessary to form the internal part of the layer. In most cases, the time required to execute a single tool path depends on the tool path length and on the repositioning number of the forming tool. The delay time between subsequent layers' deposition, called *recoating time*, takes into account the time which is necessary for the cooling or the solidification of the deposited material, the non-productive time for the prototype's vertical translation and other auxiliary unproductive activities such as nozzle cleaning.

Generally speaking, the build time of layer manufactured objects can result from the sum of seven different components:

$$t_f = T_{\mathrm{c-mat}} + T_{\mathrm{h-mat}} + T_{\mathrm{c-sup}} + T_{\mathrm{h-sup}} + T_{\mathrm{rep-mat}} + T_{\mathrm{rep-sup}} + T_{\mathrm{delay}} \quad (1)$$

where:

- $T_{\text{c-mat}}$ is the total scanning time of the material contour
- $T_{\text{h-mat}}$ is the total time for hatching material
- $T_{\text{c-sup}}$ is the total scanning time of the supports' contour
- $T_{\text{h-sup}}$ is the total time for hatching supports
- $T_{\text{rep-mat}}$ is the total time for the repositioning of the material deposition tool
- $T_{\mathrm{rep-sup}}$ is the total time for the repositioning of the supports' deposition tool
- T_{delay} is the total delay time between subsequent layers' deposition.

Nonetheless, in the related literature, there has not yet been introduced a convenient and general function to estimate build time components. Specific formulas have been defined, each of which are adapted for a specific machine that uses a specific technology. The methods presented in literature can be classified into two main categories: detailed analysis- and parametric-based approaches.

The first group includes methods that calculate the total build time by computing the geometry through a detailed analysis of the activities associated with the prototyping process [4–9]. These methods can only be used if combined with the slicing software and the tool path generator, which are dedicated to a specific layer manufacturing technology.

If we are to develop a general purpose method for build time estimation, the parametric approach has undoubtedly a greater potentiality. Cheng [10] and Frank [11], with the intent of looking for the minimal cost build direction, propose a simple parametric approach which assumes build time to be proportional to the slices' number (b_z/L). Xu et al. [12] proposed a more sophisticated parametric approach which only considers two components: layers' deposition time and total delay time between subsequent layers' deposition.

$$t_f = T_{\mathrm{h-mat}} + T_{\mathrm{delay}} = (V_{\mathrm{mat}}/L_{\mathrm{mat}}) \cdot t_s + (b_{z-\mathrm{mat}}/L_{\mathrm{mat}}) \cdot t_w \quad (2)$$

where:

- $b_{\text{z-mat}}$ prototype's height
- L_{mat} layer thickness
- t_s solidifying rate or material deposition rate in the build time
- t_w delay time between subsequent layers' deposition
- V_{mat} volume of material to be formed.

Formula (2) is easy to implement but it does not take into account the complexity of the geometric model associated with the presence of holes and with the complexity of layers' contours; neither does it consider the unproductive time for the movements and the repositioning of the manufacturing tool.

Ruffo et al. in [13] introduced the bounding box volume of the part which is to be manufactured by the selective laser sintering as the build-time driving factor that identifies its shape complexity. Even though the results reported in the paper show errors less than the 13% in the build time estimation, the method is nevertheless developed for a specific machine and cannot be directly extended to other technologies, especially those that require supports. In order to overcome the typical limitations of a parametric approach, Munguìa et al. in [14] proposed a neural network-based model for build time estimation for selective laser sintering using the same input parameters proposed by Ruffo et al. in [13] (the $z-$height, the volume and the bounding box of the part to be manufactured). The results reported prove that using the neural network reduces errors in build time estimation if compared with the traditional parametric approach.

In order to take into account shape complexity, Campbell et al. in [15] introduced the area to be hatched as a build-time driving factor for the stereo lithography machine. The authors proposed the following parametric model to estimate build time:

$$t_f = T_{\mathrm{h-mat}} + T_{\mathrm{h-sup}} + T_{\mathrm{delay}} \quad (3)$$

The laser hatching time for an interior region and for support is considered to be directly proportional to the area of the region and inversely proportional to the hatch spacing and the scanning velocity. The time delay takes into account only the recoating time and it is assumed as a constant value. As pointed out by the authors, the method does not consider all the terms of build time, such as the contour scanning time, or all the unproductive times which add to the total delay time between subsequent layers'.

In order to quickly calculate the total hatching area of all layers, Nezhad et al. in [16] proposed the *estimator algorithm*. The validation, despite being verified only in the hatching area estimation of two simple models, seems to be promising.

An important build time component is associated with the layer contour depositions. This time component is already considered in the method proposed by Lan and Ding in [17]. They introduce the following formula to evaluate the build time of an object layered by a stereo lithography machine:

$$t_f = T_{\mathrm{c-mat}} + T_{\mathrm{h-mat}} + T_{\mathrm{h-sup}} + T_{\mathrm{delay}} \quad (4)$$

The first three terms depend on the mean velocities of contour scanning, of hatching and of supports' scanning, respectively. These velocities cannot be easily determined for technologies other than the laser stereo lithography. The first term is evaluated roughly as a linear function of the total length contour and, for this reason, it is always overestimated. Also in this case, the last term of the Eq. 4 only takes into account the total recoating time.

The analysis of the parametric models presented in literature evidenced that none of the methods consider all the important components of build time. In a particularly competitive market, even the smallest error in build time estimation can significantly affect build cost estimation by producing a non-profitable or an excessive price estimation.

3 Build time driving factors

The build time driving factors here defined must be identified from the STL standard definition of the geometric

Table 1 Sub-processes involved in typical commercial rapid prototyping technologies

Technologies	Sub-processes			
	Material		Support	
	Border scanning	Area scanning	Border scanning	Area scanning
Fused deposition modelling (FDM)	Yes	Yes	Yes	Yes
Stereo lithography (SLA)	Yes	Yes	Yes	Yes
Selective laser sintering (SLS)	Yes	Yes	No	No
Laminated object manufactured (LOM)	Yes	No	No	Yes
Multi jet modelling (MJM)	No	Yes	No	Yes
Three dimensional printing (3DP)	No	Yes	No	Yes
Electron beam sintering (EBM)	Yes	Yes	No	No
Selective laser melting (SLM)	Yes	Yes	No	No

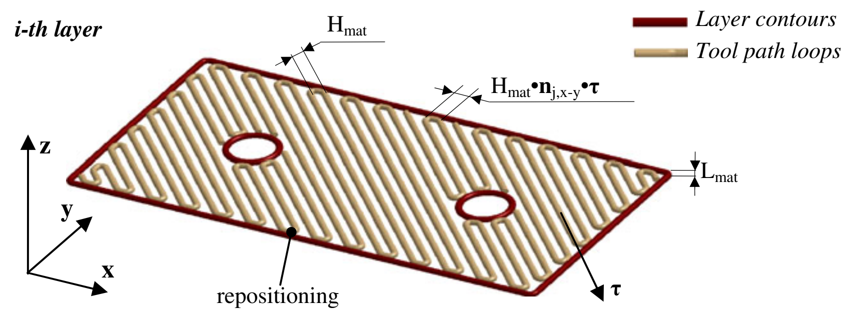


Fig. 1 Hatching distance and deposition tool repositioning number

model. In order to develop a robust build time estimator, these parameters should be, on the one hand, independent from each other so as to avoid cross-correlation and, on the other hand, independent from the specific technology which has been used to produce the prototype.

Table 1 reports the sub-processes involved in typical commercial rapid prototyping technologies ([18–21]). In addition, as previously evidenced, for every technology, non-productive activities such as vertical translation and nozzle cleaning must be considered.

As highlighted in Table 1, some technologies require the deposition of the layers' contours. The related time is proportional to the sum of the contour length of each layer. For a geometric model described by triangular facets, the total length of the layers' contour to be deposited is [22]:

$$p_{\mathrm{mat}} = \frac{\sum_{j=1}^{n_T} \sqrt{\left(1 - \left(\mathbf{z} \bullet \mathbf{n}_j\right)^2\right)} \cdot S_j}{L_{\mathrm{mat}}} \tag{5}$$

where:

$\mathbf{n}_j$ unit normal vector at triangular facet jth
n_T number of triangular facets.
S_j area of jth triangular facet
$\mathbf{z}$ model building direction.

As opposed to the expression proposed by Nezhad et al. in [16], Eq. 5 calculates the total layers' perimeter much faster, but it is not suitable when the slicing is carried out with non uniform thickness.

The time for deposition tool repositioning ($T_{\text{rep-mat}}$) is a function of the number of repositioning movements ($n_{r\text{-mat}}$) involved in hatching the internal part of the layers. This number depends on the prototype's orientation around the model building direction (**z**) with respect to the hatching vector (**τ**), which defines the direction of the tool path line segments (Fig. 1). To estimate $n_{r\text{-mat}}$, the following formula is introduced [22]:

$$n_{r-\mathrm{mat}} = \frac{1}{H_{\mathrm{mat}} \cdot L_{\mathrm{mat}}} \times \sum_{j=1}^{n_T} \sqrt{\left(1 - \left(\mathbf{z} \bullet \mathbf{n}_j\right)^2\right)} \cdot S_j \cdot \left(\left|\boldsymbol{\tau}_c \bullet \mathbf{n}_{j,xy}\right|\right) \tag{6}$$

where:

H_{mat} hatching distance between two subsequent segments of tool path (Fig. 1)
$\mathbf{n}_{j,xy}$ unit normal vector at j-th triangular facet projected onto the stratification plane
$\boldsymbol{\tau}_\mathbf{c}$ complementary vector on the stratification plane of the hatching vector.

Many commercial RP technologies, such as FDM, EBM and SLM, layer material by alternating hatching vectors

Table 2 The influence in build time of each of the eight driving factors being considered when using typically commercial rapid prototyping technologies

Technologies	Build time driving factors							
	Material				Support			
	$\frac{V_{\mathrm{mat}}}{L_{\mathrm{mat}}}$	$\frac{b_{z-\mathrm{mat}}}{L_{\mathrm{mat}}}$	p_{mat}	$n_{r\text{-mat}}$	$\frac{V_{\mathrm{sup}}}{L_{\mathrm{sup}}}$	$\frac{b_{z-\mathrm{sup}}}{L_{\mathrm{sup}}}$	$p_{\text{-sup}}$	$n_{r\text{-sup}}$
FDM	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
SLA	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
SLS	Yes	Yes	Yes	Yes	No	No	No	No
LOM	Yes	Yes	Yes	No	Yes	Yes	No	Yes
MJM	Yes	Yes	No	Yes	Yes	Yes	No	Yes
3DP	Yes	Yes	No	Yes	Yes	Yes	No	Yes
EBM	Yes	Yes	Yes	Yes	No	No	No	No
SLM	Yes	Yes	Yes	Yes	No	No	No	No

FDM fused deposition modelling, *SLA* stereo lithography, *SLS* selective laser sintering, *LOM* laminated object manufactured, *MJM* multi-jet modelling, *3DP* three-dimensional printing, *EBM* electron beam sintering, *SLM* selective laser melting

Table 3 Dependencies between build time driving factors and build time components for FDM technology

	T_{delay}	$T_{h\text{-mat}}$	$T_{c\text{-mat}}$	$T_{rep\text{-mat}}$	$T_{h\text{-sup}}$	$T_{c\text{-sup}}$	$T_{\text{rep-sup}}$
$b_{z\text{-mat}}/L_{\text{mat}}$	*H*	–	–	–	–	–	–
$V_{\text{mat}}/L_{\text{mat}}$	*L/M*	*H*	–	–	–	–	–
p_{mat}	*L*	*L/M*	*H*	*M*	–	–	–
$n_{r\text{-mat}}$	*L*	*L/M*	*M*	*H*	–	–	–
$V_{\text{sup}}/L_{\text{sup}}$	*L/M*	–	–	–	*H*	–	–
$b_{z\text{-sup}}/L_{\text{sup}}$	*H*	–	–	–	–	–	–
p_{sup}	*L*	–	–	–	*L/M*	*H*	*M*
$n_{r\text{-sup}}$	*L*	–	–	–	*L/M*	–	*H*

which are orthogonal in consecutive stratification layers. In these cases, the tool repositioning number can be evaluated as the average value of $n_{\text{r-mat}}$ calculated for two orthogonal hatching directions:

$$n_{r-mat} = \frac{1}{2 \cdot H_{\text{mat}} \cdot L_{\text{mat}}} \sum_{j=1}^{n_T} \sqrt{\left(1 - (\mathbf{z} \bullet \mathbf{n}_j)^2\right)} \cdot S_j \cdot \left(\left|\boldsymbol{\tau}_0 \bullet \mathbf{n}_{j,xy}\right| + \left|\boldsymbol{\tau}_{90} \bullet \mathbf{n}_{j,xy}\right|\right) \tag{7}$$

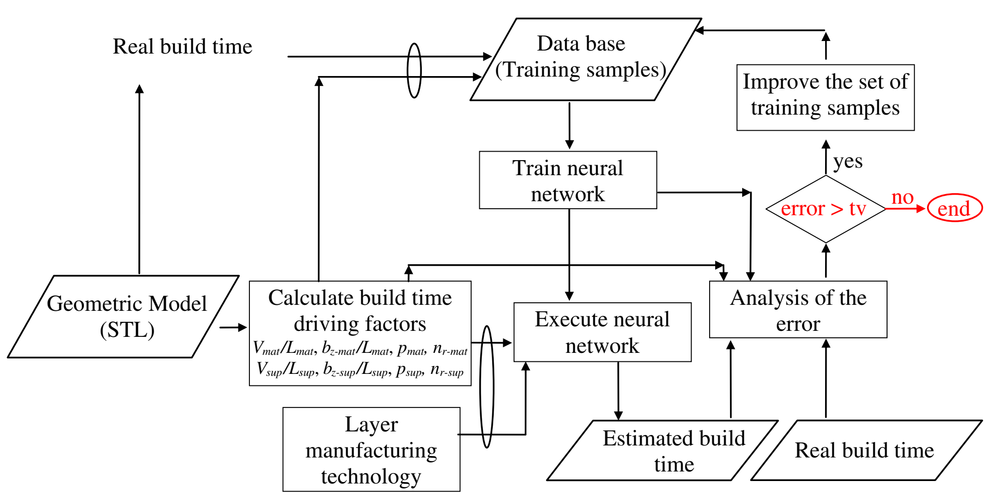
Formulas (5), (6) and (7) can only be calculated once the geometric model and the technological parameters (L_{mat}, τ and H_{mat}) are known.

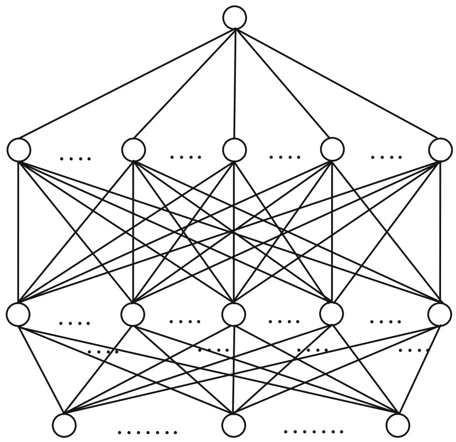
The approach used to calculate the build-time driving factors for material stratification can also be applied to supports' build-time estimation, for those technologies that require them.

Table 2 lists the eight build-time driving factors here calculated. The same table reports whether the driving factors affect build time for the different commercial technologies.

These eight parameters ($V_{\text{mat}}/L_{\text{mat}}$, $b_{z\text{-mat}}/L_{\text{mat}}$, p_{mat}, $n_{r\text{-mat}}$, $V_{\text{sup}}/L_{\text{sup}}$, $b_{z\text{-sup}}/L_{\text{sup}}$, p_{sup}, $n_{r\text{-sup}}$) only consider the dimensional and geometric features of the object to be manufactured and the dimensional technological parameters. The kinematic factors of the layer manufacturing process are not taken into consideration; namely, the hatching and moving velocity, the non-productive machine movements, acceleration and deceleration of the manufacturing tool. All these parameters are specific for the assigned process and are not furnished by the machine builder. In any case, these parameters are difficult to implement in a general purpose model of build time estimation. In other words, it is not easy to define exhaustive functions which represent the dependency existing between build time components and driving factors. A qualitative evaluation of these dependencies is shown in Table 3 for the FDM technology, where the letters *H*, *M* and *L* stand for high, medium and low dependency, respectively. The identified parameters affect all the build time components. Furthermore, Table 3 highlights the fact that each time component has a high

Fig. 2 The process for build time estimation





$$\text{Output layer (O)} \rightarrow \begin{cases} t_f \equiv y^0 = x^O \\ x^O = \sum_{k=1}^{n_{SH}} w_k^{SH} \cdot y_k^{SH} \end{cases}$$

$$\text{Second hidden layer (SH)} \rightarrow \begin{cases} y_k^{SH} = \dfrac{2}{1+e^{-2\cdot x_k^{SH}}} \\ x_k^{SH} = \sum_{j=1}^{n_{SH}} w_j^{FH} \cdot y_j^{FH} \end{cases}$$

$$\text{First hidden layer (FH)} \rightarrow \begin{cases} y_j^{FH} = \dfrac{2}{1+e^{-2\cdot x_j^{FH}}} \\ x_j^{FH} = \sum_{i=1}^{n_I} w_i^{I} \cdot y_i^{I} \end{cases}$$

$$\text{Input layer (I)} \rightarrow \mathbf{y}^{I}$$

Fig. 3 Configuration of the artificial neural network

dependency on a driving factor, with the exception of T_{delay}.

4 Build Time Estimation System

In the approach herein proposed the need for approximating build time, which is a very complex and non-linear function of the previously defined build-time driving factors, is satisfied by using a specifically designed artificial neural network.

Generally speaking, artificial neural networks can be used to perform function approximations and they are capable of generating a complete representation when only just a part of the whole information is given. Like biological neural networks, the artificial one consists of a collection of units (neurons) communicating with each other through axon connections. The artificial neural network is an adaptive system which changes its structure based on external or internal information that flows through the network. The function defined by a neural network lies on the synaptic weights that are adaptively trained by a learning mechanism. That makes it possible to store knowledge in such a flexible way that it is updated when new data are available. The neural network has an intrinsic parallel structure and its execution is very rapid. The capability of neural networks to describe knowledge strictly depends on their structure, on the number of nodes and on the synaptic connections. A multi-layer feed-forward neural network with an arbitrarily large number of units in the hidden layers can approximate any real continuous function [23].

The proposed approach for build time estimation, a diagram of which is shown in Fig. 2, takes advantage of the characteristics of artificial neural networks. The Build Time Estimation System (BTES) can be trained with the known build-time data of a set of sample cases estimated for a given build direction and layer manufacturing technology. This approach does not require that the coefficient of a complex parametric formula should be estimated. The capability of a neural network to reproduce non-linear functions gives the possibility of taking account of the complex interactions which exist between the input factors and which are not easily predictable. In this specific case, it means that the previously defined kinematic factors are implicitly considered without needing to be known.

This approach has an intrinsic adaptive capability; if for a prototype the error in build time estimation is greater than a threshold value, it is added to the set of training samples for the artificial neural network. Therefore, the build time estimator evolves as further historical data are acquired.

4.1 The architecture of the artificial neural network

BTES is based on a typical back-propagation artificial neural network (ANN) with two hidden layers (Fig. 3). It consists of eight nodes in the first layer plus a number of nodes, for logical input (1, 0), each of which is dedicated to a single manufacturing technology. Particular attention has been paid to the choice of the number of nodes in the

Table 4 Principal technical specifications of the FDM layer manufacturing machine used

Company	Technimold s.r.l.
Process	FDM
Machine	Dimension SST
Maximum working volume	203×203×205 mm
τ material and support (*)	45°
L material and support	0.254 mm
H material and support (*)	0.5 mm
(*) view figure 1	

Table 5 Correlation coefficients of the 16 training samples

	$V_{\text{mat}}/L_{\text{mat}}$	$b_{z\text{-mat}}/L_{\text{mat}}$	$n_{r\text{-mat}}$	p_{mat}	$V_{\text{sup}}/L_{\text{sup}}$	$b_{z\text{-sup}}/L_{\text{sup}}$	$n_{r\text{-sup}}$
$b_{z\text{-mat}}/L_{\text{mat}}$	0.152						
$n_{r\text{-mat}}$	0.144	0.339					
p_{mat}	0.128	0.444	0.478				
$V_{\text{sup}}/L_{\text{sup}}$	0.134	0.248	0.227	0.226			
$b_{z\text{-sup}}/L_{\text{sup}}$	0.094	0.584	0.069	0.157	0.532		
$n_{r\text{-sup}}$	0.045	0.384	0.158	0.085	0.569	0.576	
p_{sup}	0.122	0.069	0.084	0.162	0.194	0.225	0.166

hidden layers. In fact, there is no simple method to select an appropriate number of neurons for the hidden layers; it depends on the complexity of the problem and should be set empirically [24]. With too few nodes, the network may not converge in training, whilst with too many hidden-layer nodes the network starts to lose generalisation ability. Considering the aim being pursued through this work, the best results have been obtained with 75 nodes for each hidden layer of the neural network. The output node makes an estimation of the build time. The activation functions used are sigmoid–tangent for the hidden layers and linear for the output layer.

As regards its implementation, a programme has been written by the MATLAB Neural Network Toolbox [25]. The performance of the neural network, when trained with historical data, has been measured by calculating the time percentage error (TPE), which is expressed as follows:

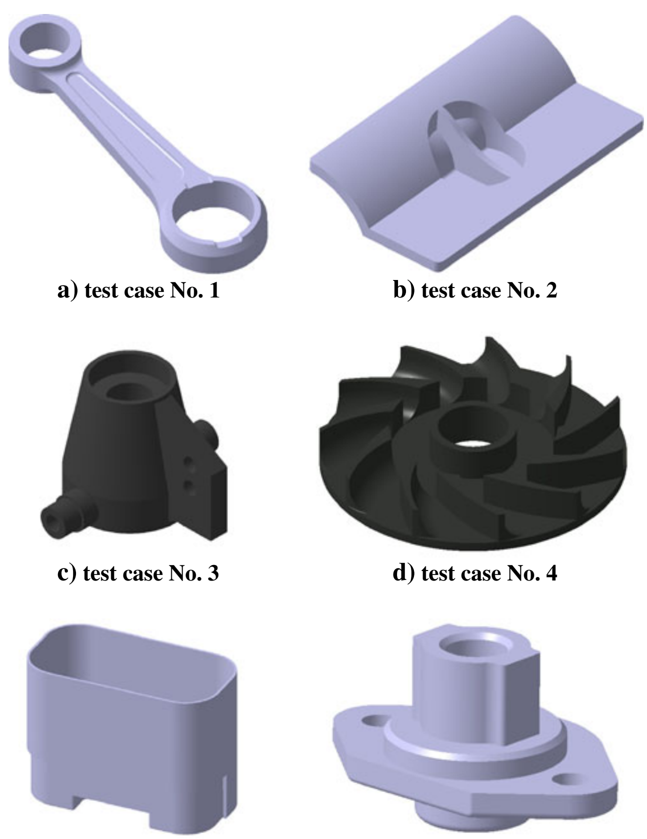
$$\mathrm{TPE}(i) = \left|\frac{\mathrm{ET}(i) - \mathrm{TT}(i)}{\mathrm{TT}(i)}\right| \cdot 100\% \tag{8}$$

where, $\mathrm{ET}(i)$ is the estimated build time of sample i and $\mathrm{TT}(i)$ is the targeted build time.

5 Training and testing of the BTES

The neural network in BTES must be trained with real known data related to the build times of a given set of objects that are manufactured with assigned technologies. The training set of samples should be representative of the correlation between each factor and the corresponding build time components. In other words, during the training phase, sufficient knowledge should be transferred into the neural network for the build time estimation to be able to be applied also to those cases for which the ANN has not been directly trained. There is not an only way to determine how many, and which, data items are appropriate for training a neural network. In literature some approaches are presented, which define the minimal number of training samples for an ANN [26].

In this work, as training samples, we have chosen objects whose build time driving factors are in the range of objects susceptible of prototyping, for assigned technologies, and are as much orthogonal as possible between them. The orthogonality of the build time driving factors ensures the independence of the effect of each factor on build time. The orthogonality of the factors of the training samples makes it possible to minimise the number of test cases which are necessary to train the ANN. It is not possible to choose build time factors which are completely independent. It is precisely because of their interactions that they are so complex and difficult to predict. Thus it is not easy to select a set of



a) test case No. 1 **b) test case No. 2**

c) test case No. 3 **d) test case No. 4**

e) test case No. 5 **f) test case No. 6**

Fig. 4 Test cases used to validate the proposed build time estimator

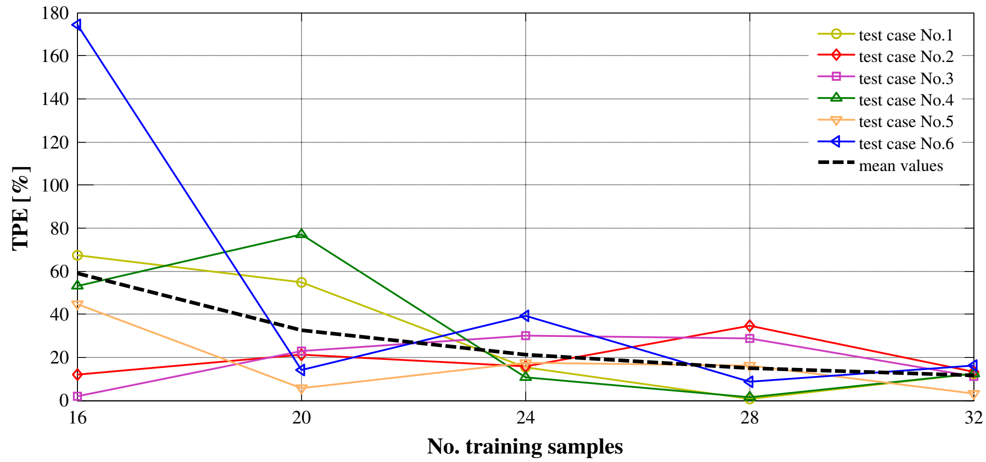


Fig. 5 TPE of the test cases at different levels of the ANN training for the FDM machine

training objects which allows us to selectively control each build time driving factor. In this paper, in order to guide the selection process of orthogonal factors, the orthogonality of the training samples is verified by means of the following correlation function:

$$B(\mathbf{X}_h, \mathbf{X}_k) = \frac{\sum_i \left(x_{h,i} - \overline{x}_h\right)\left(x_{k,i} - \overline{x}_k\right)}{\sqrt{\left[\sum_i \left(x_{h,i} - \overline{x}_h\right)\right]^2 \cdot \left[\left[\sum_i \left(x_{k,i} - \overline{x}_k\right)\right]^2\right]}} \qquad h \neq k \qquad 0 < B(\mathbf{X}_h, \mathbf{X}_k) \leq 1 \tag{9}$$

where:

$x_{h,i} \in \mathbf{X}_h$ ($\mathbf{X}_h$ is the array containing the values for the h-th build time driving factor of the training samples)

$\overline{x}_h$ is the mean value of $\mathbf{X}_h$.

A high value of B shows that build time driving factors ($\mathbf{X}_h$, $\mathbf{X}_k$) are correlated. This may entail a poor training of the neural network which could not then be able to discriminate the contribution of each build time factor.

Build-time driving factors are evaluated by computing the selected geometric models, defined in STL standard, with the original software we have developed. A first experiment is carried out by using the FDM technology (fused deposition modelling) as reference technology, whose technical specifications are quoted in Table 4; eight build time driving factors seem to be particularly suited to take into account the sub-processes involved in prototype manufacturing with this technology. Firstly, 16 different samples, which identify a quite orthogonal set, have been chosen. The relative correlation coefficients are quoted in Table 5. Although the training samples are very different from each other, the correlation coefficients in Table 5 evidence some dependencies. These dependencies can vary with the set of test cases analysed. Consequently, the training of the neural network is not carried out with an ideal set of samples.

The ANN has been trained by using the back-propagation training rules. The weights and biases of the network are adjusted to minimise the sum of the squared network errors. The initial training of the neural network has been carried out by making use of 16 samples and it has required 22,000 training cycles. The training process has been interrupted for a training mean-square error less than 10^{-8}. The maximum TPE value of the neural network for the 16 objects used in the training process is 3.8×10^{-5}%. Despite the autocorrelation existing between the build time driving factors (Table 5), ANN reproduces quite perfectly the build time of the prototypes used as training examples.

Table 6 Results obtained for six test cases (training sample size 32) for the FDM machine

Test case	Build time [min] Real	Build time [min] Estimated	Error [%]
No.1	70	61.3	−12.43
No.2	229	198.6	13.28
No.3	63	70.1	11.27
No.4	69	77.6	12.46
No.5	558	575.5	3.14
No.6	86	99.86	16.12

Table 7 Principal technical specifications of the 3DP layer manufacturing machine used

Company	Object
Process	3DP
Machine	EDEN 350V
Maximum working volume	340×340×200 mm
τ material and support	0°
L material and support	0.016 mm
H material and support	60 mm (this is the width of the slice layered with one stroke by the jetting head along the *y*-direction)

In order to check the ANN generalisation ability, the trained network has been tested out as regards the estimation it makes of the build times of six test samples (Fig. 4) that have never been used before to train ANN. The values for the build time driving factors of these test samples are included in the range of values used during the training process. These cases present some critical aspects that contribute to the difficulty in predicting build time, such as small thicknesses and holes. The build time estimation of the six test cases has been performed by training the ANN with different sets of training samples, which have been obtained by adding to the initial set of examples (16 objects) four other groups of new objects, each group consisting of four geometric models. For each stage of the training process (16, 20, 24, 28 and 32 training examples) the build times of the six objects shown in Fig. 4 have been estimated and then compared with the real build times; the results are shown in Fig. 5.

The ANN mean error decreases as the training level increases. This means that important parts of knowledge, concerning the determination of the build time function, have been progressively transferred into the network.

Table 6 quotes the build time errors which have been yielded by ANN for each test case, when 32 training examples have been used. The results obtained show a good performance of the proposed method, especially when compared with the results presented in literature [5, 6, 11, 17]. These results appear to be better, considering that the build time estimators analysed in literature are dedicated to a specific rapid prototyping technology and are designed to perform a detailed analysis of the forming tool movement.

A second experiment is carried out by using the 3DP technology (Table 7) as reference, for which the driving factors being considered are not properly suited to the specific deposition strategy. It is owing to the previous considerations that a greater number of training samples have been required (Fig. 6) to adequately train the ANN. Furthermore, build time estimation errors appear to be more dispersed: the mean value error is equal to 12% and the maximum value is 20.3% (Table 8).

6 Conclusion

This paper presents a new approach to estimate the build time of layer manufactured objects. The driving factors, which typically affect build time in the main layer manufacturing technologies, are identified. In order to automatically evaluate these factors, starting from the STL standard file of the object to be manufactured, the methods to analyse the significant geometric features are proposed. Therefore, and by means of a specifically designed artificial neural network, we are able to obtain an approximation of

Fig. 6 **TPE of** the test cases at different levels of the ANN training for the 3DP machine

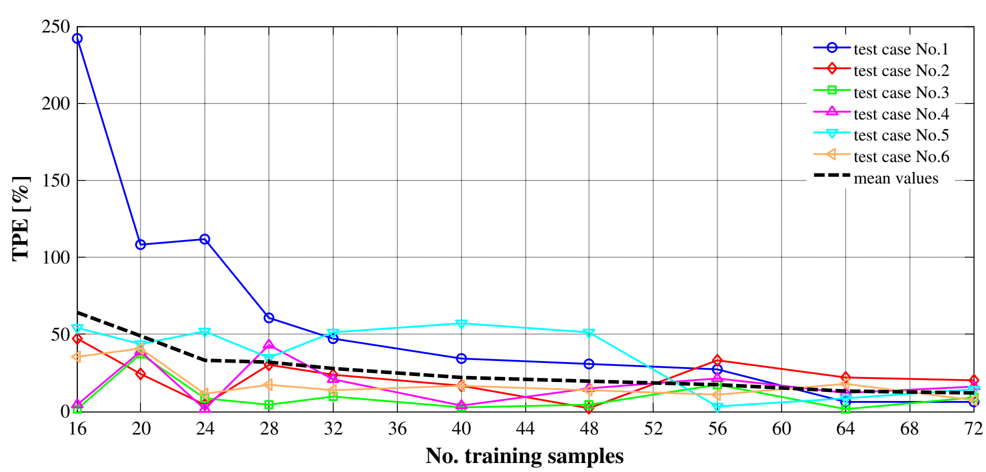


Table 8 Results obtained for six test cases (training sample size 72) for the 3DP machine

Test case	Build time [min] Real	Estimated	Error [%]
No.1	48	45.1	6.07
No.2	349	279.1	20.3
No.3	81	75.5	9.24
No.4	83	69.8	15.91
No.5	776	666.3	14.14
No.6	154	142.3	7.59

the build time, which is a very complex and non-linear function of the previously defined build time driving factors.

Particularly, the new method has been applied to calculate the build time of some test cases that are characterised by some critical aspects which add to the difficulty in predicting build time. The results obtained show a good performance, especially when they are compared with those of the methods dedicated to specific technologies presented in literature. The errors made in the build time estimation of the six test cases analysed are due to the fact that the approach being used does not perform a detailed analysis of the activities required to manufacture an object. For this reason, some build time components are left out of consideration, such as the time which is necessary for the forming tool to repose after each tool path. Other build time components cannot be strictly evaluated but, rather, must be assumed as medium values; for instance, the velocity of the deposition tool is implicitly assumed to be constant.

References

1. Rajenthirakumar D, Jagadeesh KA (2009) Analysis of interaction between geometry and efficiency of impeller pump using rapid prototyping. Int J Adv Manuf Technol 44:890–899
2. Vinodh S, Devadasan SR, Maheshkumar S, Aravindakshan M, Arumugam M, Balakrishnan K (2010) Agile product development through prototyping technologies: an examination pump-manufacturing company. Int J Adv Manuf Technol 46(5–8):663–679
3. Kechagias J, Maropoulos S, Karagiannis S (2004) Process build time estimator algorithm for lamined object manufacturing. Rapid Prototyping J 10(5):297–304
4. Alexander P, Allen S, Dutta D (1998) Part orientation and build cost determination in layered manufacturing. Comput-Aided Des 30(5):343–356
5. Byun H-S, Lee KH (2006) Determination of the optimal build direction for different rapid prototyping processes using multi-criterion decision making. Robot Comput-Integr Manuf 22:69–80
6. Luo RC, Tzou JH, Lan CC (2003) The development of web-based E-business system for rapid prototyping manufacturing. Industrial Electronics Society, IECON’03 2:1290–1295
7. Giannatsis J, Dedoussis V, Laios L (2001) A study of the build-time estimation problem for stereolithography systems. Robot Comput-Integr Manuf 17:295–304
8. Zhao J, Liu W, Wu J (2006) Determination of optimal build orientation based on satisfactory degree theory for RPT, J CAD/CAM, 6 (1)
9. Byun HS, Lee KH (2006) Determination of optimal build direction in rapid prototyping with variable slicing. Int J Adv Manuf Technol 28:307–313
10. Cheng W, Fuh J, Nee A, Wong YS, Loh HT, Miyazawa T (1995) Multi-objective optimization of part building orientation in stereolithography. Rapid Prototyping J 1(4):339–345
11. Frank D, Fadel G (1995) Export system-based selection of the preferred direction of build for rapid prototyping processes. J Intell Manuf 6(5):339–345
12. Xu F, Oh TL, Wong YS (1999) Consideration of optimal orientation for different rapid prototyping. Rapid Prototyping J 5 (2):54–60
13. Ruffo M, Tuck C, Hague R (2006) Empirical laser sintering time estimator for Duraform PA. Int J Prod Res 44(23):5131–5146
14. Munguía J, Ciurana J, Riba C (2009) Neural-network-based model for build-time estimation in selective laser sintering. Proc Inst Mech Eng, B: J Eng Manuf 223(8):995–1003
15. Campbell I, Combrink J, de Beer D, Barnard L (2008) Stereolithography build time estimation based on volumetric calculation. Rapid Prototyping J 14(5):271–279
16. Nezhad AS, Vatani M, Barazandeh F, Rahimi A (2010) Build time estimator for determining optimal part orientation. Proc Inst Mech Eng B J Eng Manuf 224(12):1905–1913
17. Lan HB, Ding YC (2007) Price quotation methodology for stereolithography parts based on STL model. Comput Ind Eng 52(2):241–256
18. Pharm DT, Gault RS (1998) A comparison of rapid prototyping technologies. Int J Mach Tools Manuf 38:1257–1287
19. Yan X, Gu P (1996) A review of rapid prototyping technologies and systems. Comput-Aided Des 28(4):307–318
20. Cooper KG (2001) Rapid prototyping technology: selection and application. Dekker, New York
21. Chua CK, Leong KF (1997) Rapid prototyping: principles of applications and manufacturing. Wiley, Singapore
22. Di Angelo L, Di Stefano P (2010) Parametric cost analysis for WEB based e-commerce of layer manufactured object. Int J Prod Res 48(7):2127–2140
23. Nielsen RH (1989) Theory of the back-propagation neural networks. Proceedings IEEE, International Conference on Neural Networks 593–605
24. Sukthomya W, Tannock J (2005) The training of neural networks to model manufacturing processes. J Intell Manuf 16:39–51
25. The MathWorks Inc (2000) Neural network toolbox for use with MATLAB
26. Chang CC, Chang TYP, Xu YG, To WM (2002) Selection of training samples for model updating using neural networks. J Sound Vib 249(5):867–883