Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 92 (2020) 218–223

www.elsevier.com/locate/procedia

16th CIRP Conference on Computer Aided Tolerancing (CIRP CAT 2020)

Tolerancing from STL data: A Legacy Challenge

Torbjørn Langedahl Leirmo[a]*, Oleksandr Semeniuta[a], Kristian Martinsen[a]

[a]Department of Manufacturing and Civil Engineering, Norwegian University of Science and Technology, Teknologivegen 22, 2815 Gjøvik, Norway

* Corresponding author. Tel.: +47 480 88 390. *E-mail address:* torbjorn.leirmo@ntnu.no

Abstract

Part representation in additive manufacturing (AM) is dominated by the stereolithography (STL) file format as a universal mode for communicating and transferring part geometry from one system to another. However, when the CAD model is converted to the triangle mesh constituting the STL file the topology is no longer explicitly defined hence the design intent is lost together with any tolerancing information. Computer aided tolerancing of actual part geometry is hindered by the sparse information about the nominal geometry directly available in STL data, therefore the feature information is often assumed or recreated through reverse engineering methods. This paper investigates how nominal geometry can be deduced from STL data to support quality control by the identification of geometric elements from a triangle mesh. We further discuss how vectorial tolerancing can extend the scope of feature recognition to tolerancing and quality assessment. A method for automatic extraction and tolerancing of features from STL files is described and an application example is provided. The outlined method enables the automation of tolerancing activities and facilitates the integration of STL files into the digital pipeline of modern manufacturing systems.

Keywords: Tolerancing; Additive Manufacturing; Geometry.

Nomenclature

AM	Additive Manufacturing
CAD	Computer Aided Design
CAM	Computer Aided Manufacturing
CAT	Computer Aided Tolerancing
FCS	Feature Coordinate System
FFF	Fused Filament Fabrication
GD&T	Geometric Dimensioning and Tolerancing
STL	Stereolithography (file format)
VT	Vectorial Tolerancing
WCS	Workpiece Coordinate System

1. Introduction

Since its conception in the late 1980s [1], Additive Manufacturing (AM) has evolved from a rapid prototyping process to a family of technologies capable of manufacturing functional parts. In the meantime, the file format originally developed to accommodate the limited computational power at the time has remained unchanged and is still widely used in the AM industry and the AM community at large [2].

As AM is embraced by the industry for the manufacture of end-use and near-net-shape parts, the quality requirements of industry are inevitably imposed on the products. Requirements for geometrical accuracy were originally developed to moderate defects from traditional manufacturing technologies and was later formalized in the standards ISO 1101 [3] and ASME Y14.5 [4] for geometric dimensioning and tolerancing (GD&T), and ISO 286 [5] for linear sizes.

If a component is exported as a stereolithography (STL) file from computer aided design (CAD) software, the design intent is lost together with any tolerancing information since the topology is no longer explicitly defined [6, 7]. The task of recognizing features from a triangle mesh is a simple job for the human brain but turns out to be a complex problem for a computer. The automatic extraction of shape features from mesh data is still an active field of research after several decades [8, 9]. Computer aided tolerancing (CAT) heavily relies on the availability of feature information, but in a situation where the original CAD file is unavailable, this

10.1016/j.procir.2020.05.180

information must then either be assumed or recreated through reverse engineering.

The geometrical accuracy of functional surfaces is of vital importance when manufacturing end-use products – especially when the manufactured component is part of an assembly. As AM is finding its way into modern manufacturing systems, the components are also increasingly used in assemblies where the interfaces need tolerancing. This is, however, a challenge when dealing with STL files because information about the position and orientation of shape features is not readily available. Closed loop tolerance engineering is enabled by the integration of all manufacturing operations in a digital pipeline to which the STL file constitutes a major obstacle [10, 11].

The current work describes how the functional surfaces extracted from STL files may be described by vectors in accordance with vectorial tolerancing (VT) practices. This representation scheme directly enables VT of extracted features, and by extension the automation of quality inspection. Furthermore, the vectorial representation offers a link back to the STL file which enables proactive manipulation of the geometry to accommodate process inaccuracies.

2. Related work

2.1. Geometric inaccuracies in additive manufacturing

The different technologies in the AM family introduce a myriad of variations in actual geometry. The observed geometric deviations may, however, be traced back to four distinct origins as indicated by Dantan, et al. [12]:

- File format and resolution;
- Process planning and parameters;
- Machine specific errors and inaccuracies; and
- Material properties and environmental effects.

The low resolution of STL files may cause distinguishable triangles on the part surface, and numerical imprecisions introduces errors that may cause any downstream process to fail. Process planning includes the placement and orientation of the part in the build space which inevitably impose imprecisions due to raster patterns and layer thickness – a phenomenon commonly referred to as the staircase effect [13]. Moreover, machine imprecisions due to loose components and rounded edges cause deviations from nominal to actual geometry, and finally, the material may introduce variations and could react to environmental factors by shrinking and warping. A comprehensive discussion on the challenges related to tolerancing in AM is presented in [6].

2.2. Achievable tolerances in additive manufacturing

Budinoff and McMains [14] performed a theoretical analysis of achievable tolerances in AM considering the geometric deviations due to the layered approximation. The authors further described a tool for identifying feasible orientation zones given a part with accompanying tolerances.

Minetola, et al. [15] investigated the achievable geometric tolerances of fused filament fabrication (FFF) and mapped them to the international tolerancing grades. Dimitrov, et al. [16] achieved the same objective for binder jetting, and Hanumaiah and Ravi [17] investigated direct metal laser sintering and the stereolithography process for tooling purposes. Geometric accuracy for SLA was also mapped out in [18], but with simplifications with regards to feature orientation. Studies similar to [15] but for dimensional tolerances has been conducted by Lieneke, et al. [19] for FFF and for material jetting by Kitsakis, et al. [20]. A study by Ippolito, et al. [21] compared the accuracy of five AM processes and evaluated them relative to traditional manufacturing technologies.

The optimization of process parameters with respect to achievable tolerances complements the studies mentioned above. Arni and Gupta [13] presented a method for constructing build orientation feasibility regions for flatness tolerances in AM, while the cylindricity error was investigated by Paul and Anand [22] who later combined the two methods and included support structures [23]. Building on this previous work, Das, et al. [24] developed an optimization scheme for minimizing the volume of support structures while satisfying GD&T callouts, and later also considered the accessibility of support structures for post-processing [25]. The input of these optimization methods is described as a CAD file with embedded tolerance callouts.

2.3. Extracting shape features from STL files

Many applications would benefit from the topological information no longer present after converting to STL file format, and thus the task of extracting topological information from STL data has received major research interest. This already troublesome task is made more complicated by export defects such as occasional holes and intersecting triangles corrupting the STL file [8].

Two distinct categories may be identified in the literature: (i) feature recognition where information is extracted for manufacturing purposes [26, 27], and (ii) mesh segmentation which is primarily geared towards computer graphics [28, 29]. The different intended uses result in a pivotal difference in how these methods work. While the former strives to describe the geometry as precisely as possible to enable direct manufacturing, the latter is concerned with the partition of geometries for identification purposes. A mix of the two can be found in reverse engineering applications where a combination of methods may be utilized [30]. For the purpose of tolerancing, we argue that the successful extraction of geometric primitives from STL data is of higher importance than partitioning of freeform surfaces because of their use as functional surfaces.

Moroni, et al. [31] proposed a methodology for estimating the accuracy of cylindrical features in FFF based on STL data. The authors proposed an algorithm effectively slicing the part along all three axes to identify cylindrical features of the part. The method enables the comparison of actual dimensions to nominal data but provided no means to store or communicate the information.

3. Theoretical foundations

3.1. The STL file format

The stereolithography file format originally got its name from the AM process it was intended to serve [2, 32]. Later, the acronym has also been explained as Standard Tessellation (or Triangulation) Language [33]. In addition to the simplicity of the format, the STL files are being used largely due to its availability for import and export in CAD/CAM applications.

The STL file contains an unordered list of triangles (facets) with their unit normal vectors (facet normals) and the coordinates of the three corners (vertices). This requires 12 floating point numbers stored for each facet where the facet normals point towards the exterior.

3.2. Defining coordinate systems

Any coordinate system is defined from the origin O fixed at (0, 0, 0). The Cartesian coordinate system (x, y, z) is defined by three basic unit vectors representing the axes of the coordinate system: $\mathbf{i}=[1,0,0]$, $\mathbf{j}=[1,0,0]$ and $\mathbf{k}=[1,0,0]$. Alternative coordinate systems include the spherical polar coordinates (r, θ, φ), and cylindrical coordinates (r, θ, z). The choice of coordinate system depends on the application as this influences the complexity of computation. Cartesian coordinates are used for the remainder of this paper.

Regardless of the coordinate system, the position of any point in space can be represented by a position vector $\overline{OP}$ which defines the location of the point P with reference to the origin. In manufacturing applications, separate coordinate systems may be defined for each manufacturing feature or functional surface to facilitate local process planning such as machining operations. The feature coordinate system (FCS) is defined with respect to the workpiece coordinate system (WCS) and may be oriented differently as displayed in Fig. 1.

3.3. Vector representation and manipulation

The location of a directional vector in $\mathbb{R}^3$ is typically defined by a translation vector $\mathbf{t}\in\mathbb{R}^3$, while the orientation may be represented as a 3×3 special orthogonal matrix $(\mathbf{R}\in SO(3))$. It is common to combine the rotation matrix $\mathbf{R}$

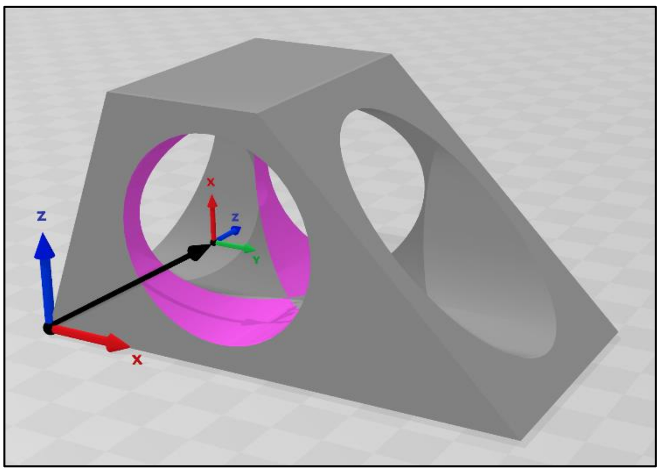

Fig. 1. A part with Workpiece Coordinate System (WCS) and a Feature Coordinate System (FCS) for a cylinder.

and the translation vector $\mathbf{t}$ in a single homogeneous transformation matrix $\mathbf{T}\in SE(3)$:

$$\mathbf{T}=\begin{bmatrix}\mathbf{R} & \mathbf{t}\\ 0_{1\times 3} & 1\end{bmatrix}=\begin{bmatrix}\vdots & \vdots & \vdots & t_x\\ r_x & r_y & r_z & t_y\\ \vdots & \vdots & \vdots & t_z\\ 0 & 0 & 0 & 1\end{bmatrix} \tag{1}$$

The representation of a vector in $\mathbb{R}^3$ may thus be condensed into a representation in the form of $(r_x,r_y,r_z,t_x,t_y,t_z)$. The rotation components are typically denoted A, B and C for the counterclockwise rotation about the x-, y- and z-axis respectively and are formalized for AM in ISO/ASTM 52921:2013(E) [34].

3.4. Vectorial definitions of geometric primitives

Martinsen [35] describes how the location and orientation of geometric primitives can be represented vectorially. This gives rise to six degrees of freedom which can be used to classify the fixed and open dimensions of geometric primitives as displayed in Table 1.

Table 1. Degrees of freedom. F = Fixed, O = Open. Adapted from [35].

Surface type	Translations			Rotations		
	X	Y	Z	A	B	C
Plane	O	O	F	F	F	O
Cylinder	F	F	O	F	F	O
Sphere	F	F	F	O	O	O
Cone	F	F	F	F	F	O
Torus	F	F	F	F	F	O

Similarly, a scheme for vectorial representation may be constructed to define the position, orientation, and size of geometric primitives as shown in Table 2. Relevant sizes comprise the radius of cylinders, spheres, and tori (R), as well as the apex angle of cones (ω).

Table 2. Vectorial surface description with location vector **P**, orientation vector **E**, radius Adapted from [35].

Surface type	Location vector P_0			Orientation vector E			Sizes	
Plane	X_0	Y_0	Z_0	E_x	E_y	E_z		
Cylinder	X_0	Y_0	Z_0	E_x	E_y	E_z	R	
Sphere	X_0	Y_0	Z_0				R	
Cone	X_0	Y_0	Z_0	E_x	E_y	E_z	ω	
Torus	X_0	Y_0	Z_0	E_x	E_y	E_z	R1	R2

The location vector in Table 2 points to the origin of the surface which may be explicitly defined if all translation of the surface type is fixed with regards to every dimension with reference to Table 1. This leaves out planes and cylinders which require additional rules for an unambiguous definition of surface origin. Whenever the exact point of origin is without importance, a random point satisfying the fixed dimension(s) may be selected [35].

3.5. Vectorial tolerancing

The description of geometric primitives by vectors gave rise to the concept of VT in the late 1980s. While traditional tolerancing methodology is based on the premise of 1-dimensional measurements, VT enables the unambiguous representation of nominal part geometry in three dimensions [36]. The application of tolerances on the surface descriptions of Table 2 makes it possible to rigorously quantify the location and orientation of shape features in 3D space. The tolerances can then easily be tabulated as displayed in Table 3 [37]. The table states the nominal location vector **P** and the nominal orientation vector **E** with their respective deviation tolerances $\mathbf{T_P}$ and $\mathbf{T_E}$. Furthermore, the size **S** (if relevant), and form may be specified in the table.

4. Proposed method

The proposed method entails a stepwise transition from a triangle mesh, to a vectorial representation of constituent geometric primitives to which tolerances may be applied. This framework also makes it possible for other processing stages to access higher-order information from the STL file by maintaining the digital thread of CAD/CAM processing. A stepwise description is provided in the following subsections.

4.1. Vectorial representation of geometric primitives

When geometric primitives are extracted from STL data, the jump to vectorial representation is quite short. The current work assumes the preceding feature recognition module to be capable of identifying the feature type and location of surface points in a stable manner. The feature origin may be defined in accordance with the VT paradigm based on these surface points by following predefined rules according to the number of degrees of freedom associated with the surface type (see Table 1). The location vector will then be the vector from the WCS to the feature origin.

The relevant sizes of a feature may also be extracted from STL data by different methods. The apex angle ω of cones are easily deduced from the normal vectors of member facets, and the radius may be calculated as the distance of vertices to a common center.

In consequence, the basic geometric primitives may be automatically extracted from STL data and represented as vectors where the WCS, including its origin, is adopted from the STL file. By directly transferring the WCS, the link back to the STL file is left uncorrupted which facilitates later adjustments for manufacturing optimization at various processing stages including build preparation and quality inspection.

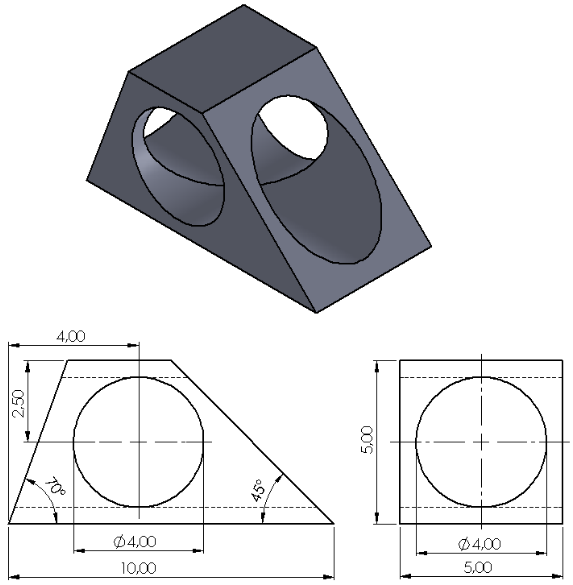

Fig. 2. Drawing of sample part with six planes and two cylinders. Recreated from [20].

4.2. Vectorial part representation

By extending the reasoning from the previous subsection, all primitives of a part may be unambiguously defined and tabulated as displayed in Table 3. This table may be the direct output from automatic feature recognition providing an overview of the constituent primitives of the part, as well as a starting point for tolerance analysis. Depending on the sophistication of the feature recognition module, the resulting table may include an unknown number of features not relevant for tolerancing purposes. Hence, the judgment of an engineer may be required to transform the automatically generated table to a suitable configuration.

5. Application example

To demonstrate the approach described in the previous section, a simple part geometry recreated from [38] is used as a case study (Fig. 2). The STL file contains a list of 860 facets which may be reduced to the 8 constituent features as displayed in Table 4. The table directly enables the specification of the vectorial tolerances associated with the part features.

Table 3. Tolerance table for vectorial tolerances of n shape features. **P** is the nominal location vector and **E** is the nominal orientation unit vector.

#	Location						Orientation						Size		Form	
	Nominal			Limit deviation (±) [mm]			Nominal			Limit deviation (±) ×0.001			Nominal	Limit dev. (±)	Nominal	Limit deviation
	P_x	P_y	P_z	T_{Px}	T_{Py}	T_{Pz}	E_x	E_y	E_z	T_{Ex}	T_{Ey}	T_{Ez}	S[mm]	T[mm]	[type]	[mm]
1	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
2	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
∣	∣	∣	∣	∣	∣	∣	∣	∣	∣	∣	∣	∣	∣	∣	∣	∣
n	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…

To conduct a proper case study, the part should be assigned a function. Since the true purpose of the part is unknown, two assumptions are made about the design intent:

- The component is a standardized part of an assembly where it is intended to connect two or more shafts; and
- One possible application of the component requires support on the angled planes.

Based on the assumptions above, the main emphasis of tolerancing is on the cylindrical holes (features 1 and 2 in Table 4) because they are regarded as functional surfaces. Of secondary importance, the accuracy of the angled planes should be within certain limits (features 3 and 4 in Table 4). Certainly, additional tolerances can easily be added to the table as additional columns or supplementary rows beneath the relevant feature. The table enables the tolerancing of all identified features, but this is not deemed appropriate for this case study.

Table 4 can later be used to evaluate the feasibility of an AM process, quality assessment, or as a tool for process planning if combined with process-specific knowledge.

6. Discussion

6.1. Digital continuity

The era of industry 4.0 calls for digital integration of all manufacturing processes to establish a two-way connection between upstream and downstream operations [11]. Closed loop tolerance engineering provides a framework for this integration in modern manufacturing systems [10] which greatly benefits from CAT [39]. However, this integration is impeded by intermediate file formats such as STL [6].

The vectorial representation of geometric primitives offers a two-way link between the STL file and the subsequent processes. This link may be utilized to improve the accuracy of the realized geometry by manipulation of the STL file to mitigate inaccuracies for the specific part in the next iteration of manufacturing. Over time, the aggregated data on vectorial deviations enables the utilization of intelligent computation methods such as machine learning to make predictive changes to the STL file towards first-time-right manufacturing.

A major benefit of VT is how it facilitates the automatic integration of tolerance considerations in CAD/CAM applications. Increased automation and digital integration improve the traceability of tolerances in the manufacturing system which in turn facilitates intelligent process planning.

6.2. Quantification of inaccuracies

The STL file format introduces certain inaccuracies brought about by round off errors as well as the discretization of smooth curves resulting in the characteristic tessellated surface. Due to these errors, some uncertainty regarding the true size and position of features is inevitable when extracting information from STL data.

One solution to this problem is to discretize the coordinate space and move vertices to their closest valid values. This approach is however invalid as it assumes that the coordinate system of the STL file is the same as the one utilized in the design phase, while it may have been subjected to several file manipulations including translation, rotation, and scaling.

Another solution to file inaccuracies is to allow the user to do corrections after the features are extracted. This approach requires a cost analysis to determine what is most costly: the time spent by an engineer to correct the data or the problems caused by these errors. Most likely, the errors will be negligible and not significantly affect the final product. Consideration of the entire tolerance chain should reveal the necessity of addressing this issue for each case.

6.3. Freeform surfaces

While primitive geometries may be easily defined by standard sizes, freeform surfaces require a flexible scheme to be accurately described. The proposed method is geared towards primitive geometric shapes and is not directly applicable to freeform surfaces such as the organic structures that characterize topology optimized designs. The current work is believed to provide a basis for future work which could include freeform surfaces, as well as integration with other methods for assigning quality measures to design features. Future research could entail complimenting the current work with representation and specification of organic structures and internal geometries with other relevant quality requirements such as mechanical properties or graded material specifications.

Table 4. Tolerance table for sample part. Nominal location vector **P**, nominal orientation unit vector **E** and nominal sizes extracted from STL file.

#	Location						Orientation						Size		Form	
	Nominal			Limit deviation (±) [mm]			Nominal			Limit deviation (±) ×0.001			Nominal	Limit dev. (±)	Nominal	Limit deviation
	P_x	P_y	P_z	T_{Px}	T_{Py}	T_{Pz}	E_x	E_y	E_z	T_{Ex}	T_{Ey}	T_{Ez}	S [mm]	T [mm]	[type]	[mm]
1	4.0	0	2.5	0.1	0.1	0.1	0	1	0	5	5	5	4	0.05	Cylinder	0.05
2	0.18	2.5	2.5	0.1	0.1	0.1	1	0	0	5	5	5	4	0.05	Cylinder	0.10
3	0	0	0	0.2	0.2	0.2	-sin(70)	0	cos(70)	10	10	10	-	-	Plane	0.15
4	5.0	0	5	0.2	0.2	0.2	sin(45)	0	cos(45)	10	10	10	-	-	Plane	0.15
5	1.82	0	5.0	-	-	-	0	0	1	-	-	-	-	-	Plane	-
6	0	5.0	0	-	-	-	0	0	-1	-	-	-	-	-	Plane	-
7	0	5.0	0	-	-	-	0	1	0	-	-	-	-	-	Plane	-
8	0	0	0	-	-	-	0	-1	0	-	-	-	-	-	Plane	-

7. Summary

The industrialization of AM entails tolerancing of AM products. The continued use of legacy file formats such as the STL file brings about novel challenges especially in maintaining the digital thread throughout the product life cycle. When a product geometry is converted to STL file format, any tolerancing information is lost along with the design intent. To perform a tolerance analysis for the product, this information must then either be assumed or recreated through reverse engineering.

This paper described how functional surfaces of STL files may be converted to vectorial representations which directly enables VT of shape features. A case study demonstrated a practical application of the method with an accompanying tolerancing table. It is argued that the proposed method fits well into the digital pipeline of contemporary manufacturing systems, and constitutes a meaningful approach to the tolerancing of AM products.

Acknowledgments

This research is funded by the Norwegian Ministry of Research and Education and is associated with SFI Manufacturing funded by the Norwegian Research Council.

References

[1] Hull CW. Apparatus for production of three-dimensional objects by stereolithography. US patent no. US4575330 A. 1986.
[2] Gibson I, Rosen DW, Stucker B. Additive manufacturing technologies. 2nd ed. New York: Springer; 2015.
[3] ISO 1101:2017(E). Geometrical product specifications (GPS) – Geometrical tolerancing – Tolerances of form, orientation, location and run-out. 2017.
[4] ASME Y14.5M-2018. Dimensioning and Tolerancing. 2018.
[5] ISO 286-1:2010(E). Geometrical product specifications (GPS) – ISO code system for tolerances on linear sizes – Part 1: Basis of tolerances, deviations and fits. 2010.
[6] Ameta G, Lipman R, Moylan S, Witherell P. Investigating the Role of Geometric Dimensioning and Tolerancing in Additive Manufacturing. J Mech Des N Y. 2015;137(11):111401–10.
[7] Moroni G, Petrò S, Polini W. Geometrical product specification and verification in additive manufacturing. CIRP Ann Manuf Technol. 2017;66(1):157–60.
[8] Szilvśi-Nagy M, Mátyási G. Analysis of STL files. Math Comput Model. 2003;38(7):945–60.
[9] Shi Y, Zhang Y, Baek S, De Backer W, Harik R. Manufacturability analysis for additive manufacturing using a novel feature recognition technique. Comput Aided Des Appl. 2018;15(6):941–52.
[10] Krogstie L, Martinsen K. Closed Loop Tolerance Engineering – A Relational Model Connecting Activities of Product Development. Procedia CIRP. 2012;3:519–24.
[11] Schleich B, Wärmefjord K, Söderberg R, Wartzack S. Geometrical Variations Management 4.0: towards next Generation Geometry Assurance. Procedia CIRP. 2018;75:3–10.
[12] Dantan J-Y, Huang Z, Goka E, Homri L, et al. Geometrical variations management for additive manufactured product. CIRP Ann Manuf Technol. 2017;66(1):161–4.
[13] Arni R, Gupta SK. Manufacturability analysis of flatness tolerances in solid freeform fabrication. J Mech Des N Y. 2001;123(1):148–56.
[14] Budinoff H, McMains S. Prediction and visualization of achievable orientation tolerances for additive manufacturing. Procedia CIRP. 2018;75:81–6.
[15] Minetola P, Iuliano L, Marchiandi G. Benchmarking of FDM Machines through Part Quality Using IT Grades. Procedia CIRP. 2016;41:1027–32.
[16] Dimitrov D, van Wijck W, Schreve K, de Beer N. Investigating the achievable accuracy of three dimensional printing. Rapid Prototyp J. 2006;12(1):42–52.
[17] Hanumaiah N, Ravi B. Rapid tooling form accuracy estimation using region elimination adaptive search based sampling technique. Rapid Prototyp J. 2007;13(3):182–90.
[18] Lynn-Charney C, Rosen DW. Usage of accuracy models in stereolithography process planning. Rapid Prototyp J. 2000;6(2):77–87.
[19] Lieneke T, Denzer V, Adam GAO, Zimmer D. Dimensional Tolerances for Additive Manufacturing: Experimental Investigation for Fused Deposition Modeling. Procedia CIRP. 2016;43:286–91.
[20] Kitsakis K, Kechagias J, Vaxevanidis N, Giagkopoulos D. Tolerance Analysis of 3d-MJM parts according to IT grade. IOP Conf Ser Mater Sci Eng. 2016;161:012024.
[21] Ippolito R, Iuliano L, Gatto A. Benchmarking of Rapid Prototyping Techniques in Terms of Dimensional Accuracy and Surface Finish. CIRP Ann Manuf Technol. 1995;44(1):157–60.
[22] Paul R, Anand S. Optimal part orientation in Rapid Manufacturing process for achieving geometric tolerances. J Manuf Syst. 2011;30(4):214–22.
[23] Paul R, Anand S. Optimization of layered manufacturing process for reducing form errors with minimal support structures. J Manuf Syst. 2015;36:231–43.
[24] Das P, Chandran R, Samant R, Anand S. Optimum Part Build Orientation in Additive Manufacturing for Minimizing Part Errors and Support Structures. Procedia Manuf. 2015;1:343–54.
[25] Das P, Mhapsekar K, Chowdhury S, Samant R, Anand S. Selection of build orientation for optimal support structures and minimum part errors in additive manufacturing.Comput Aided Des Appl. 2017;14(sup1):1–13.
[26] Zhang J, Li Y. Region segmentation and shape characterisation for tessellated CAD models. Int J Comput Integr Manuf. 2016;29(8):907–15.
[27] Sunil VB, Pande SS. Automatic recognition of features from freeform surface CAD models. Comput Aided Des. 2008;40(4):502–17.
[28] Shamir A. A survey on Mesh Segmentation Techniques. Comput Graph Forum. 2008;27(6):1539–56.
[29] Rodrigues RSV, Morgado JFM, Gomes AJP. A contour-based segmentation algorithm for triangle meshes in 3D space. Comput Graph. 2015;49:24–35.
[30] Mejia D, Ruiz-Salguero O, Sánchez JR, Posada J, et al. Hybrid geometry/topology based mesh segmentation for reverse engineering. Comput Graph. 2018;73:47–58.
[31] Moroni G, Syam WP, Petrò S. Towards Early Estimation of Part Accuracy in Additive Manufacturing. Procedia CIRP. 2014;21:300–5.
[32] Hull CW, Spence ST, Albert DJ, Smally DR, et al. Cad/cam stereolithographic data conversion. US patent no. US182830. 1989.
[33] ISO/ASTM 52900:2015(E). Standard Terminology for Additive Manufacturing – General Principles – Terminology. 2015.
[34] ISO/ASTM 52921:2013(E). Standard terminology for additive manufacturing – Coordinate systems and test methodologies. 2013.
[35] Martinsen K. Vectorial tolerancing for all types of surfaces. In: Advances in Design Automation; Albuquerque, USA: ASME; 1993.
[36] Britten W, Weber C. Transforming ISO 1101 Tolerances into Vectorial Tolerance Representations — A CAD-Based Approach. In: van Houten, F, Kals, H, editors. Global Consistency of Tolerances; Dordrecht: Springer Netherlands; 1999.
[37] Bialas S, Humienny Z, Kiszka K. Relations Between ISO 1101 Geometrical Tolerances and Vectorial Tolerances—Conversion Problems. In: ElMaraghy, HA, editor. Geometric Design Tolerancing: Theories, Standards and Applications: Springer; 1998. p. 88–99.
[38] Cheng W, Fuh JYH, Nee AYC, Wong YS, et al. Multi-objective optimization of part-building orientation in stereolithography. Rapid Prototyp J. 1995;1(4):12–23.
[39] Schleich B, Wartzack S. How can Computer Aided Tolerancing Support Closed Loop Tolerance Engineering? Procedia CIRP. 2014;21:312–7