Paper 021, ENG 103

Slicing Issues in CAD Translation to STL in Rapid Prototyping

Divesh R. Sahatoo
In-Corr-Tech Ltd., San Fernando, Trinidad
drsahatoo@yahoo.com

Boppana V. Chowdary
University of The West Indies, St. Augustine
chowdary@eng.uwi.tt

Fahraz F. Ali
University of The West Indies, St. Augustine
fadeed@gmail.com

Raj Bhatti
School of Engineering, Chatham Maritime, Kent ME4 4TB, UK
r.bhatti@gre.ac.uk

Abstract

The rapid prototyping (RP) process is being used widely with great potential for rapid manufacturing of functional parts. The RP process involves translation of the CAD file to STL format followed by slicing of the model into multiple horizontal layers, each of which is reproduced physically in making the prototype. The thickness of the resulting slices has a profound effect on the surface finish and build time of the prototype. The purpose of this paper is to show the effects of slice thickness on the surface finish, layering error, and build time of a prototype, as well as to show how an efficient STL file can be developed. Three objects were modeled and STL files were generated. One STL file for each object was sliced using different slice thicknesses, and the build times were obtained. Screenshots were used to show the slicing effect on layering error and surface finish and to demonstrate the means to a more efficient STL file. From the results, it is clear that the surface finish and build time are important factors that are affected by slice thickness.

Introduction and Background

In rapid prototyping (RP) processes, the tessellated standard triangular language (STL) model is “sliced” into layers, each of which is reproduced physically by the RP machine in building the prototype. This is an additive process where each slice is placed on top of the preceding one, resulting in the creation of the prototype. The thickness of the slices used to manufacture the prototype would bring about an effect called layering error, stair stepping error [1], or staircase effect [2]. This study demonstrates the effect of slice thickness on the surface finish, layering error, and build time of a prototyped object.

Slicing is carried out after the object's solid model is converted to an STL file by dividing its entire surface into triangles in 3D space. The accuracy of each slice, and the eventual model, is dependent on the position and number of triangular facets used. Commonly, users tend to over-tessellate where one spot size or slice may include several facets, which renders much of the information useless [3]. This paper is aimed at demonstrating the effects of slice thickness on the surface finish, layering error, and build time of a prototype. It is also aimed at showing the relationship between tessellation and slicing in the development of a more efficient and effective STL file.

Literature Review

The issue of slicing plays an important role in the RP process; as such, there have been several pieces of literature focusing on this area. The layering error issue has been investigated by Pandey et al. [2] and Tata et al. [1], while Tyberg and Bohn [4] provided a brief look at it in their study of adaptive slicing in reducing fabrication time. However, there has been no examination of how the slicing issue is directly related to object tessellation besides a mention of the fact that over tessellation may be done, which may lead to wastage of resources [3].

Other literature that deals with the issue of slicing includes Noorani [5], who defines slicing and gives a brief look at the concepts behind it. Tata et al. [1] discussed an adaptive slicing algorithm that can vary the slice thickness depending on the geometry of the object to be prototyped in an attempt to minimize layering error and improve surface finish. Tyberg and Bohn [4] presented an adaptive slicing approach that significantly reduced fabrication time by 17–37 percent compared to conventional methods and improved surface finish by eliminating unnecessary slices. Jamieson and Hacker [6] looked at a way to directly slice CAD models using contours instead of going to STL, hence overcoming some problems that would be encountered. Hope et al. [7] outlined the issues for improving geometric accuracy, while Choi and Kwok [8] dealt with a tolerant slicing algorithm to overcome computer memory constraints and computational problems. A review of various slicing methods for CAD and tessellated models can be seen in the study by Pandey et al. [2].

When an object is sliced, horizontal slice planes are intersected with the sides of the triangular facets at each particular slice height. Those collective intersecting points are then adjacently joined by straight lines to form contours at each height [1, 2, 12]. It is apparent that if there are more intersected triangles, there would be more points available, and thus, the contour would be more accurate. This process is repeated until contours are created for the entire object from top to bottom [12]. The slicing process will bring about a layering error [1] that would affect the surface finish of the prototype. The layering error increases with increased slice thickness [4].

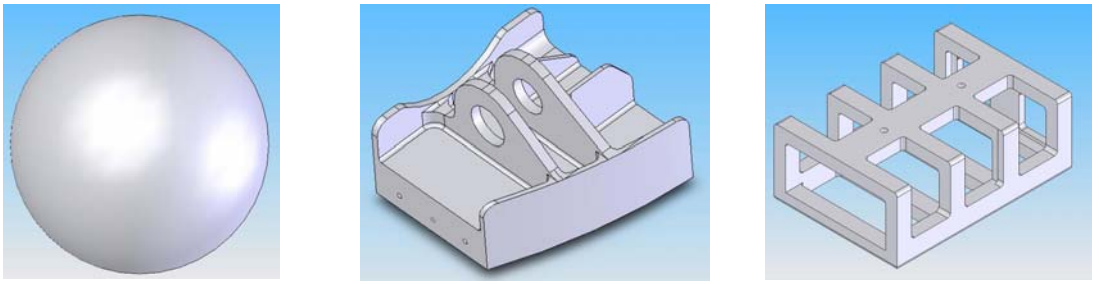
Fadel and Kirschman [3] stated the possibility of over-tessellation with respect to slicing. This occurs when there are triangles present that are not intersected by slice planes and, hence, do not contribute to the slice contours or slices. Therefore, they are not necessary. Determining ways of eliminating such unwanted STL data would lead to the usage of less

hard disk space, less memory, and would require less processing time with respect to tessellation, which leads to a decrease in production time.

Research Approach

For the purpose of investigation, three objects were modeled and used to demonstrate the effect of slice thickness on layering error, surface finish, and build time, as well as the tessellation-slicing relationship. Several screenshots were generated that focused on specific parts of each object to get a better view of the issues. In showing the tessellation-slicing relationship, the triangular facets and slice planes were shown where they lay in relation to each other using different slice thicknesses and tolerances. The screenshot approach has been proven effective in the past [8, 9, 11].

Three objects (refer to figure 1) were modeled using the Solidworks package for demonstration of the research issues. Three STL files of different tolerances were generated for each object. One of these STLs for each object was used to show the effect of slice thickness on the layering error and surface finish. All of the STLs were used to investigate the tessellation-slicing relationship.

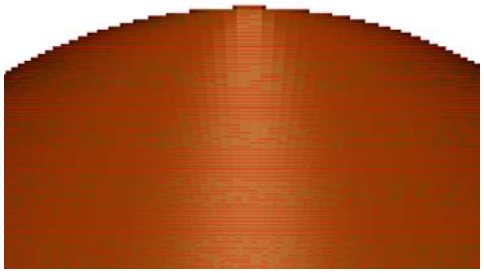


(a) Sphere (b) Alblock (c) Battery Carrier

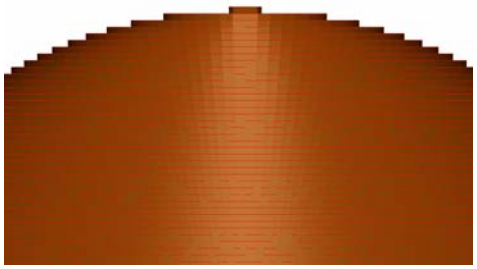
Figure 1. Objects Selected for Investigation

Results

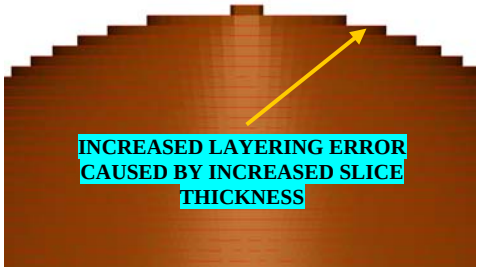
The sliced images (refer to figures 2, 3, and 4) show the effect of slice thickness on the layering error and, hence, the surface finish of the object. For a better understanding of this study, special focus was paid to specific areas of each object where the slicing effects were more evident.



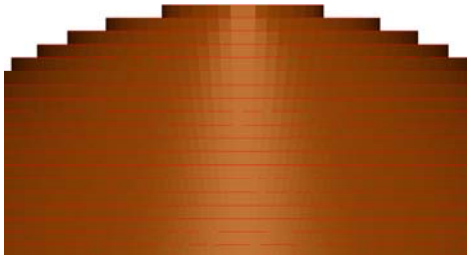
(a) Slice thickness = 0.2mm



(b) Slice thickness = 0.4mm

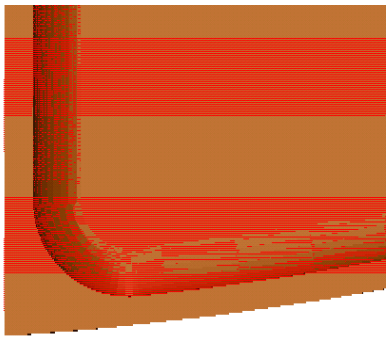


(c) Slice thickness = 0.6mm

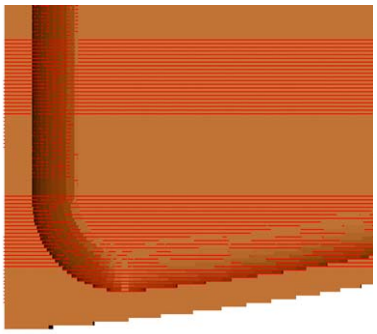


(d) Slice thickness = 0.8mm

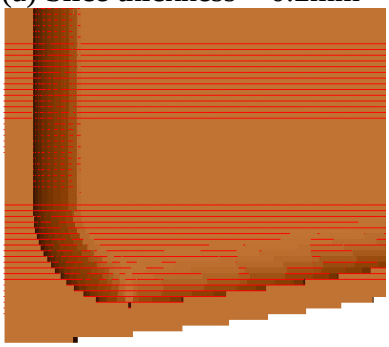
Figure 2. Screenshots of Sliced Sphere STL Files



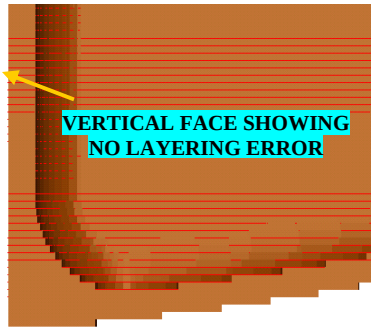
(a) Slice thickness = 0.2mm



(b) Slice thickness = 0.4mm

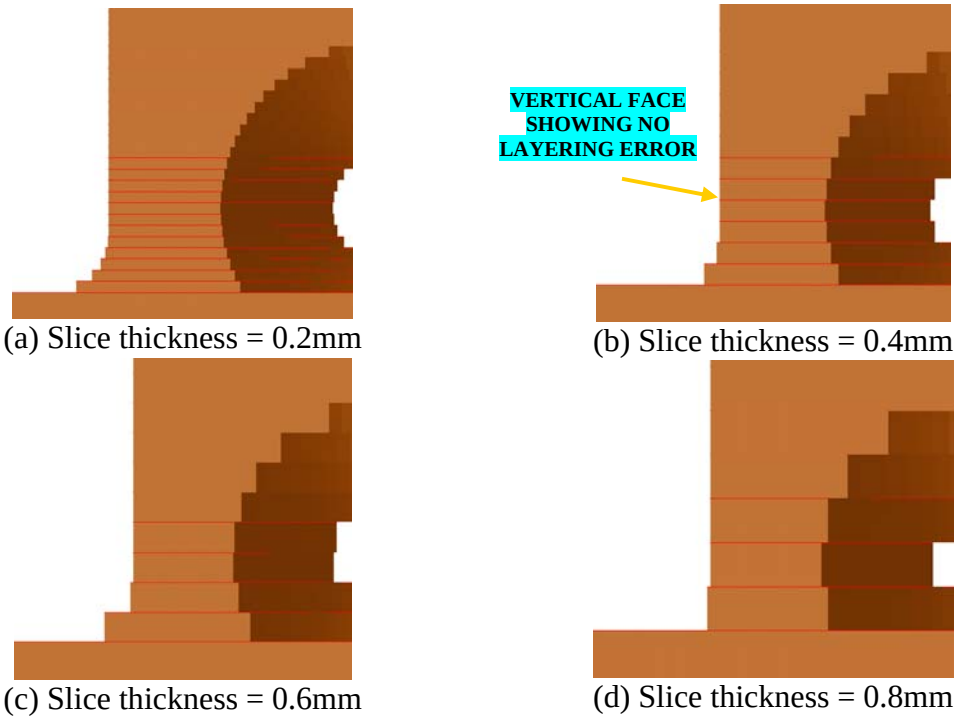


(c) Slice thickness = 0.6mm



(d) Slice thickness = 0.8mm

Figure 3. Screenshots of Sliced Alblock STL Files



(a) Slice thickness = 0.2mm (b) Slice thickness = 0.4mm

(c) Slice thickness = 0.6mm (d) Slice thickness = 0.8mm

Figure 4. Screenshots of Sliced Battery Carrier STL Files

The generated STL files were sliced for each object using different slice thicknesses, and the resulting build times were calculated using Z Corp's Spectrum Z510 3D printer specification of two slices per minute printing speed. The relationships established between build time and slice thickness are presented in figure 5.

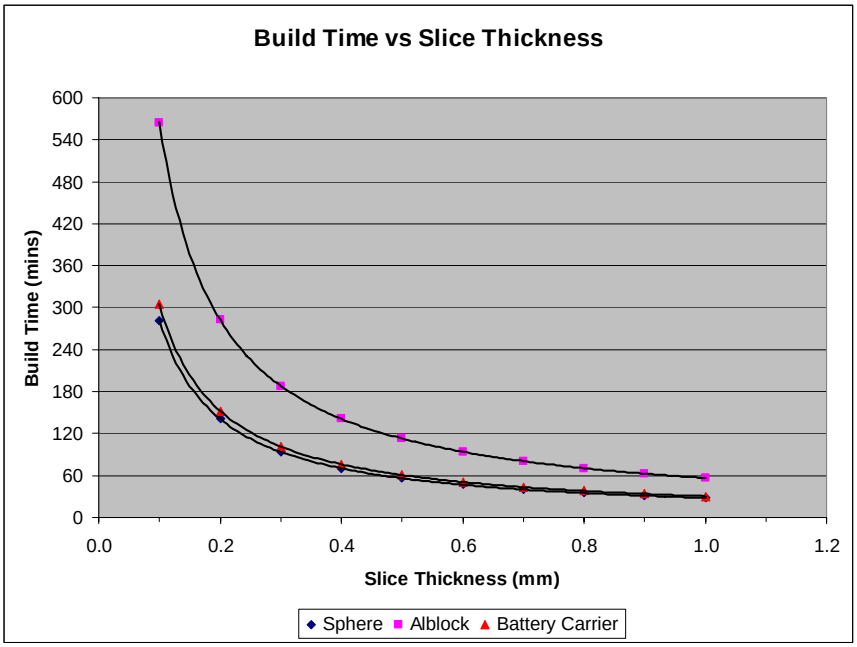
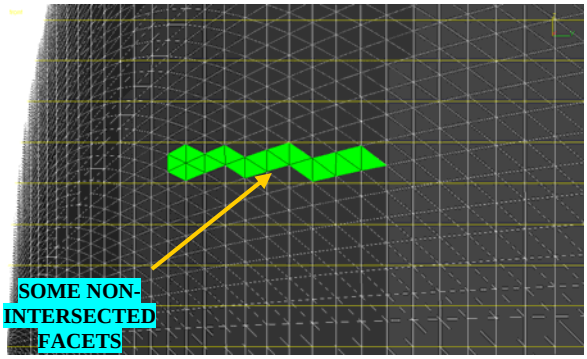
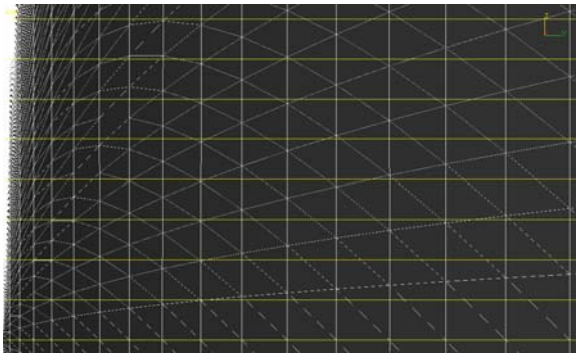


Figure 5. Build Time vs. Slice Thickness

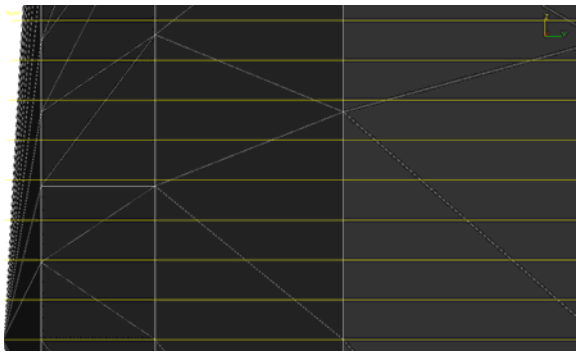
The screenshots shown in figures 6, 7, and 8 were captured to show the slice planes and triangular facets of a section of each object. The slice planes are represented by straight horizontal lines, which cross the sides of the triangular facets creating intersection points, all of which are joined to form slice contours. The various views shown in figures 6 and 7 are portions of the sphere and alblock, respectively; whereas, the views shown in figure 8 are parts of the battery carrier, which depicts a bevel and flat faces. It is noted that the battery carrier has mostly straight faces with some small features such as bevels.



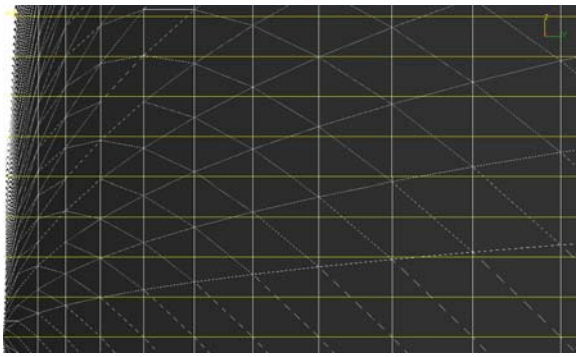
(a) Tolerance = 0.005 mm
Slice Thickness = 0.8 mm



(b) Tolerance = 0.005 mm
Slice Thickness = 0.2 mm

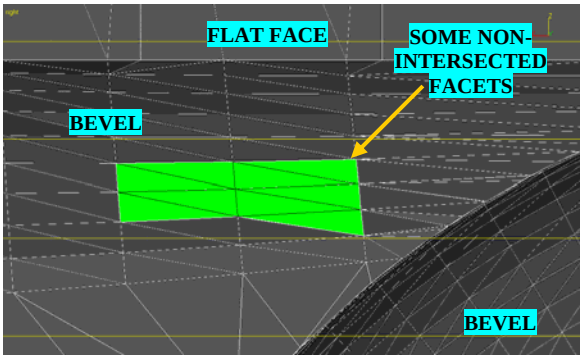


(c) Tolerance = 0.10 mm
Slice Thickness = 0.2 mm

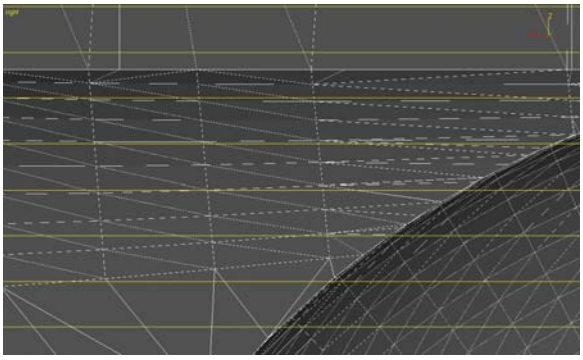


(d) Tolerance = 0.01 mm
Slice Thickness = 0.2 mm

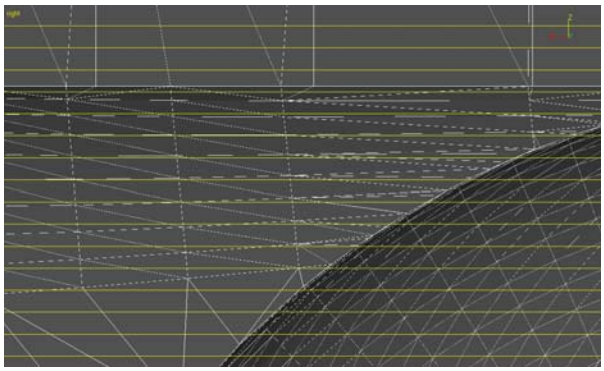
Figure 6. Screenshots of Tessellated Sphere STL Files with Slices



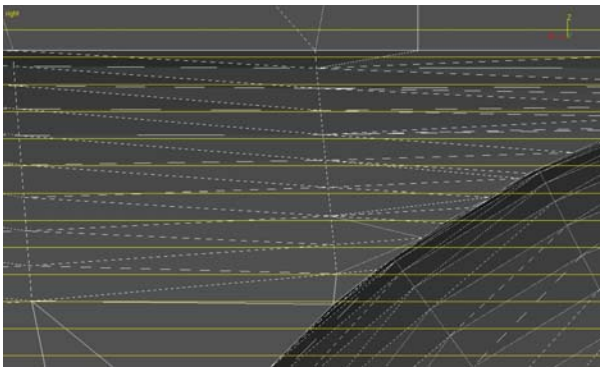
(a) Tolerance = 0.01 mm
Slice Thickness = 0.8 mm



(b) Tolerance = 0.01 mm
Slice Thickness = 0.4 mm

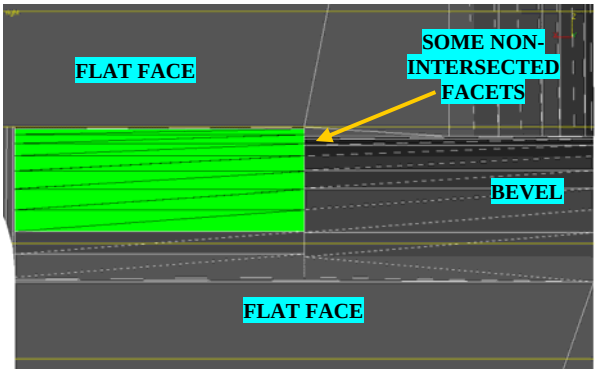


(c) Tolerance = 0.01 mm
Slice Thickness = 0.2 mm

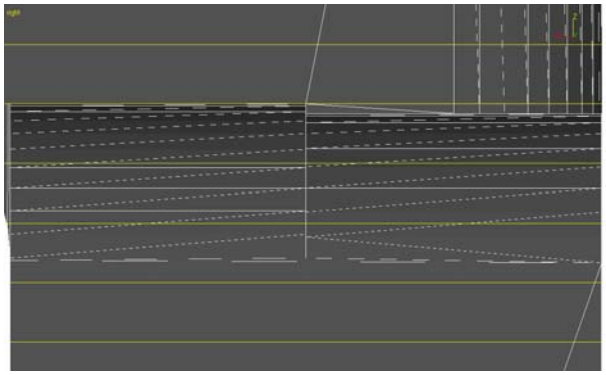


(d) Tolerance = 0.05 mm
Slice Thickness = 0.2 mm

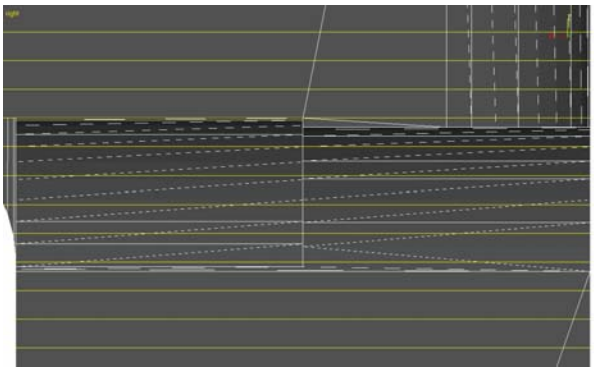
Figure 7. Screenshots of Tessellated Alblock STL Files with Slices



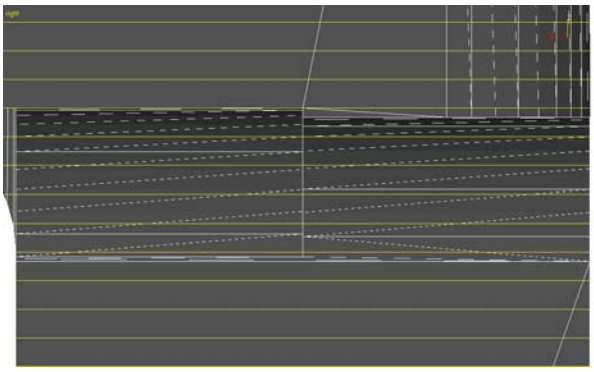
(a) Tolerance = 0.004 mm
Slice Thickness = 0.8 mm



(b) Tolerance = 0.004 mm
Slice Thickness = 0.4 mm



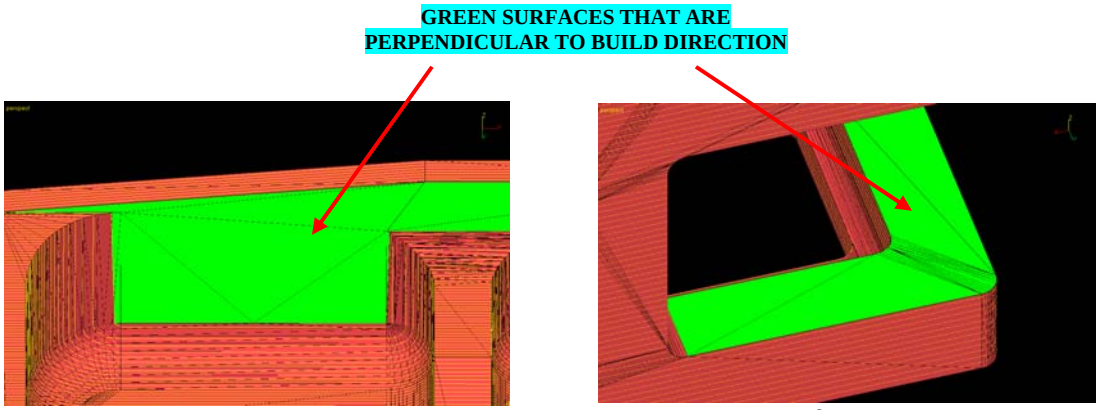
(c) Tolerance = 0.004 mm
Slice Thickness = 0.2 mm



(d) Tolerance = 0.005 mm
Slice Thickness = 0.2 mm

Figure 8. Screenshots of Tessellated Battery Carrier STL Files with Slices

Figure 9 shows selected views of the alblock and the battery carrier with green colored surfaces. These surfaces represent the areas that are perpendicular to build direction (i.e., parallel to the slice planes).



(a) View of Alblock

(b) View of Battery Carrier

Figure 9: Selected Object Views of Flat Faces Perpendicular to Build Direction

Discussion

Figures 2, 3, and 4 show an increase in layering error and a decrease in the quality of surface finish with increasing slice thickness. Thus, it is desirable to have a slice thickness that would provide the best surface finish possible with minimum build time and resource wastage. It is noticed that the layering error has no effect on flat horizontal and vertical faces of an object, as shown in figures 3 and 4. Hence, the surface finish is not an issue here.

Figure 5 shows a decrease in prototype build time with increasing slice thickness for all three objects. The relationship is surprisingly not a straight line but a curve. It is desirable to choose the largest possible slice thickness available such that the prototype would be completed in minimal time. However, this needs to be balanced with the issue of surface finish and resource wastage. It is noted that the build height of the object has a direct impact on the build time, as shown in figure 5. The sphere and battery carrier have almost the same build height, so their graphs are almost identical. It is observed that the build time is halved when the slice thickness is increased from 0.1 mm to 0.2 mm for all the selected objects of this study. This build time is almost halved again when the slice thickness is increased from 0.2 mm to 0.3 mm. A slice thickness of between 0.3–0.4 mm seems like the most efficient choice, since this would give minimum layering error with a better build time; however, this would mainly depend on the relationship with tolerance values.

In creating the slices, slice contours must first be established. In establishing these contours, triangular facets are intersected by slice planes. To find the most efficient setting, the following factors must be taken into consideration:

- Tolerance value
- Slice thickness
- Number of non-intersected facets

As shown in figures 6, 7, and 8, there are cases where facets are not intersected by any slice planes. This brings about resource wastage, as never used facet data is stored but needs to be processed. The sphere shows many facets that are not intersected by slice planes, as seen in figure 6(a). It is notable, however, that figure 6(b) shows very little or no wastage of facets, with most of them being intersected. Figure 6(d) shows no signs of non-intersecting facets. Figure 6(c) shows multiple intersections per facet. The surface finish in this case, however, is negatively affected by tessellation and may not produce the best prototype. Thus, figure 6(d) seems to be the best choice of all the models since it provides the best model at a tolerance of 0.01 mm and a slice thickness of 0.2 mm.

For the alblock, there are several non-intersecting facets present in figure 7(a). Figure 7(b) shows fewer non-intersecting facets, while figures 7(c) and 7(d) show none. Since the surface finish of figure 7(d) is worse than figure 7(c) because of a higher tolerance value, figure 7(c) would represent the best settings for this object, with a tolerance of 0.01 mm and a slice thickness of 0.2 mm. It would also be the best choice since the features shown are small bevels, which require a small tolerance and slice thickness that would provide more points in forming slice contours and, hence, would provide a more accurately prototyped object. It was observed in the study that non-intersecting facets are present on curved surfaces but are

seldom seen on flat surfaces. This can be seen in figures 7 and 8 for the alblock and battery carrier, respectively, where the uppermost part of each image consists of a flat face.

For the battery carrier, there are several non-intersecting facets that can be seen in figure 8(a). Fewer non-intersecting facets are seen in figure 8(b), while figures 8(c) and 8(d) show none. Since a bevel is present here, more points are better for prototyping this small feature; hence, figure 8(c) would be the best model. Thus, in generating the most efficient STL files for RP purposes, the tolerance and slice thickness values would have to be fine-tuned until an appropriate number of intersections are present, producing the least amount of non-intersected facets.

When slicing is carried out, each slice is oriented horizontally and built in a vertical direction. As mentioned previously, each horizontal slice plane intersects triangular facets to form contours that make up each slice. Noorani [5] and Fadel and Kirschman [3] mention that the STL file structure comprises each vertex of each triangle, along with a normal vector pointing outward. As seen in figure 9, there are faces present that are perpendicular to the build direction (i.e., parallel to the slice planes). The triangular facets on these faces are not intersected by slice planes and hence need not be considered in the entire process. In this regard, further research efforts are necessary for developing a means of specifying the build direction or orientation of the object that would avoid the tessellation of flat faces perpendicular to the build direction.

Conclusion

This research highlighted the issues in the translation of CAD models to STL files for RP purposes. To meet the research objectives, three objects were modeled using the Solidworks package. STL files of different tolerances and slice thicknesses were generated to study the effect of slice thickness on surface finish and build time. It has been shown that the layering error increases, and the surface finish quality and build time decreases, with increasing slice thickness. The use of screenshots in this research provides a clear view of how tessellation and slicing are related and how fine-tuning the tolerance and slice thickness values can lead to a more efficient STL representation. It is clear in this study that non-tessellation of flat faces perpendicular to build direction would also contribute to a more efficient STL file representation. The factors discussed in this paper will be useful to the industry for optimal utilization of computer resources and improvement in production time.

References

[1] Tata, K., Fadel, G., Bagchi, A., and Aziz, N. (1998), “Efficient slicing for layered manufacturing,” *Rapid Prototyping Journal*, Vol. 4, No. 4, pp. 151–167.

[2] Pandey, P.M., Venkata, N., Dhande, R., and Dhande, S.G. (2003), “Slicing procedures in layered manufacturing: a review,” *Rapid Prototyping Journal*, Vol. 9, No. 5, pp. 274–288.

[3] Fadel, G.M. and Kirschman, C. (1996), "Accuracy issues in CAD to RP translations," *Rapid Prototyping Journal*, Vol. 2, No. 2, pp. 4–17.
[4] Tyberg, J. and Bohn, J.H. (1998), "Local adaptive slicing," *Rapid Prototyping Journal*, Vol. 4, No. 3, pp. 118–127.
[5] Noorani, R. (2006), *Rapid Prototyping: Principles and Applications*, Wiley: U.S.
[6] Jamieson, R. and Hacker, H. (1995), "Direct slicing of CAD models for rapid prototyping," *Rapid Prototyping Journal*, Vol. 1, No. 2, pp. 4–12.
[7] Hope, R.L., Roth, R.N., and Jacobs, P.A. (1997), "Adaptive slicing with sloping layer surfaces," *Rapid Prototyping Journal*, Vol. 3, No. 3, pp. 89–98.
[8] Choi, S.H. and Kwok, K.T. (2002), "A tolerant slicing algorithm for layered manufacturing," *Rapid Prototyping Journal*, Vol. 8, No. 3, pp. 161–179.
[9] Gibson, I., Cheung, L.K., Chow, S.P., Cheung, W.L., Beh, S.L., Savalani, M., and Lee, S.H. (2006), "The use of rapid prototyping to assist medical applications," *Rapid Prototyping Journal*, Vol. 12 No. 1, pp. 53–58.
[10] Holzer, F. and Fadel G. (2002), "Design of a 3-degrees of freedom platform for the stereolithography apparatus," *Rapid Prototyping Journal*, Vol. 8, No. 2, pp. 100–115.
[11] Swann, S. (1996), "Integration of MRI and stereolithography to build medical models: A case study," *Rapid Prototyping Journal*, Vol. 2, No. 4, pp. 41–46.
[12] Kirschman, C. and Jara-Almonte, C.C. (1992), "A parallel slicing algorithm for solid freeform fabrication processes," Proceedings of the 1992 Solid Freeform Fabrication Symposium, August 3–5, Austin, TX.

Biographies

DIVESH R. SAHATOO is currently an Inspection Engineer. Having completed his Bachelor's degree in Mechanical and Manufacturing Engineering at the University of the West Indies St. Augustine, he is currently involved in engineering, inspection and research work.

BOPPANA V. CHOWDARY is a faculty member in the Department of Mechanical and Manufacturing Engineering at the University of the West Indies, St. Augustine, Trinidad. His research interests are in the areas of flexible manufacturing, product design and development, and CAD/CAM.

FAHRAZ ALI is currently a post-graduate student in the Department of Mechanical and Manufacturing Engineering at the University of the West Indies, St. Augustine, Trinidad. He is working on an MPhil project within the areas of product design and development and CAD/CAM.

RAJ BHATTI has a PhD in Automated Assembly from King's College London. He is currently a Principal Lecturer in the School of Engineering at the University of Greenwich. He is also the Associate Head of Post-Graduate Studies.

Acknowledgements

This research was supported in part by the School of Graduate Studies and Research of the University of the West Indies, St. Augustine Campus, Trinidad, under grant CRP.3GT.