STAG: Smart Tools and Applications in Graphics (2017)
A. Giachetti, P. Pingi and F. Stanco (Editors)

Polycube-based Decomposition for Fabrication

Filippo A. Fanni, Gianmarco Cherchi and Riccardo Scateni

Department of Mathematics and Computer Science, University of Cagliari, Italy

Abstract

In recent years, fabrication technologies developed at a very fast pace. However, some limitations on shape and dimension still apply both to additive and subtractive manufacturing, and one way to bypass them could be the subdivision of the object to build. We present here a simple algorithm, based on the polycube representation of the original shape, able to decompose any model into simpler portions that are better fabricable. The shape is first mapped in a polycube and, then, split to take advantage of the simple polycube subdivision, thus having, quite easily, a partition of the model at hand. The main aim of this work is to study and analyse pros and cons of this simple subdivision scheme for fabrication, in view of using both the additive and subtractive pipelines. The proposed subdivision scheme is computationally light and it produces quite good results, especially when it is applied to models that can be easily decomposed in a small collection of cuboids. The obtained subdivisions are suitable for 3D printing.

CCS Concepts
•***Computing methodologies*** → *Mesh geometry models;* •***Hardware*** → *Design for manufacturability;*

1. Introduction

The introduction of cheap and small 3D printers has boosted the research in the field of digital model representation for fabrication. Novel algorithms and techniques have been developed to make 3D printing a possibility to reliably fabricate resistant, accurate and cheap objects. As a consequence of this explosion, many manufacturers have started selling low-price entry-level 3D printers, leading to a good diffusion between hobbyists. Those kind of 3D printers have, of course, less functionalities and features compared to the high-level ones. One of the most noticeable difference is the dimension of the printing chamber (the maximum volume that can be printed). The only possibility to print big objects is to decompose them into multiple portions, print them separately and later reassemble the object.

Beside 3D printing, that we can call *Additive Manufacturing*, a different approach of fabricating digital shapes, usually called Machining or *Subtractive Manufacturing*, has a large diffusion in the field of mechanical engineering. This latter approach makes use of CNC milling machines and is routinely used since decades, in the industry, to fabricate parts out of metals blocks or, sometimes, other materials like wood or foam. As with 3D printing, the dimension of the object can be a constraining factor, that can be addressed with the same techniques. In machining, other important constraints apply to the shape of the object, and a way to bypass them is to subdivide the object into pieces that satisfy the required features.

An interesting and low cost approach for the subdivision of the 3D model (required in some of the fabrication processes), can be realized via its polycube parametrization [THCM04]. Indeed, polycubes offer a compact and simplified representation of the model, through a set of connected cuboids. Our work aims to subdivide a 3D model by using the computationally lightweight subdivision of its polycube, and to analyse the obtained subdivision explicitly addressing both Additive and Subtractive Manufacturing.

2. State of the art

Our work and analysis are related to different research topics in computer graphics that we cover in the following sections.

2.1. Polycube maps

Polycube maps (or just polycubes) are simple polyhedra composed only by orthogonal faces. This compact representation of a mesh has been proposed by Tarini et al. [THCM04], in order to obtain a seamless texturing. Polycubes have to respect three main constraints: axis-aligned faces, only 90 degree dihedral angle and planar faces. These constraints cause the polycubes to be a simplified representation of the original mesh. As a consequence, polycubes can be useful in some manipulation and analysis. They offer a good compromise among a simplification of the model, the respect of the original shape and its topology.

Polycubes are created to have a one-to-one mapping with the original models, so every vertex and triangle of the model is mapped on the polycube. As a consequence, it is possible to manipulate

DOI: 10.2312/stag.20171220

the polycube's elements, and trivially map the manipulation to the original model. The main advantage is that polycube's elements can be manipulated in their axis-aligned version. Multiple studies have exploited this characteristic in different research fields. The first application is seamless texture-mapping, proposed in [THCM04]. The surface of the polycube is used as the texture-domain, and it can be used in multiple resolution models without any additional computation. Provided the polycubes offer a low distortion mapping, this technique is efficient and has almost no drawbacks.

Other application fields are trivariate spline fitting, volumetric texturing and hexahedral meshing. Indeed, any polycube can be trivially gridded, obtaining a completely regular hexmesh [LMPS16]. Assuming the polycube mapping is volumetric, it is possible to map this hexmesh to the original shape. The obtained hexmesh is highly regular (the topology is computed on the polycube parametric space), and its quality depends on the polycube mapping, as every corner in the polycube is mapped to the hexmesh as a singularity.

In the past years, multiple algorithms have been proposed to create polycubes, each one addressing specific characteristics. The first algorithm has been proposed in 2008 by Lin et al. [LJFW08]. In 2009 He et al. [HWFQ09] proposed an alternative method, based on the principle of *Divide et Impera*. Gregson et al. in 2011 [GSZ11] analysed the problem of computing hexmeshes using the polycube parametrization. In order to do so, they developed a *Deformation based* approach to create polycubes. In 2013 Livesu et al. [LVS*13] proposed an approach based on the use of a graph-cut classification of the triangles of the mesh, followed by an hill-climbing optimization of the boundaries; this proposal offers a fidelity vs compactness use of parameters that allows for several possible polycubes mapping. Another approach is proposed bu Huang et al. in [HJS*14], and it obtains polycubes with low distortion and low corner count.

As well as polycubes generation algorithms, other relevant works on polycubes have been done. A shape optimizator, very useful for this analysis, have been proposed by Cherchi et al. [CLS16] in 2016. In this work, an algorithm to align polycubes' singularities has been proposed. While the aim of that work is not the creation of polycubes, having a "good-quality" polycube, with a low number of corners and a good singularities alignment, is very helpful. Polycubes with these characteristics proved to be useful in our work.

2.2. Fabrication

Fabrication is an emerging field in computer graphics research. It includes all the processes and techniques that can be used to produce real objects from digital models. The most relevant technology are, but are not limited to, 3D printers and CNC milling machines. These represents two different approaches to fabrication, usually referenced as additive and subtractive fabrication.

Researches in the fabrication field are typically related to additive manufacturing, while a very few works addresses subtractive manufacturing. We will briefly review some works regarding both of them.

2.2.1. Additive manufacturing

Additive manufacturing (or additive fabrication) makes use of machines that build layer by layer the final object. These machines are commonly referenced as 3D printers, and they can use multiple materials. The most common printers use thermoplastic polymers, and deposit the fused filament to build the layer. However, other technologies are available to use other materials, such as liquid resins, metals and various powders.

This kind of manufacturing does not really impose any kind of constraints on the model's shape. However some models requires external support structures, as 3D printers can not directly print steep overhangs or islands. Those structures have to be manually removed after the printing, and in an industrial context they represent a significant waste of material and time. To avoid this waste Hu et al. proposed in [HLZCO14] an algorithm to subdivide the model in *approximate pyramidal shape*, which can be printed without supports. Herholz et al. in [HMA15] propose a similar approach, by exploiting the surface deformation to reduce the number of pieces.

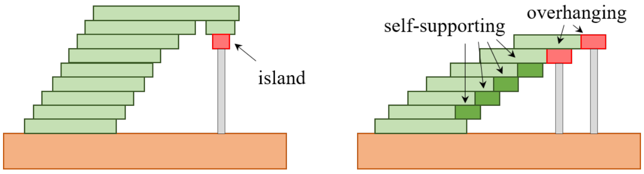


Figure 1: *Islands and overhangs need external support structures.*

The hard constraint imposed by 3D printers regards the dimension of the object, as it is obviously impossible to print anything bigger than the printing chamber. The solution to this problem is to subdivide the model into smaller portions, print them, and reassemble them back. Many works that face this problem have been proposed in the last years, and the most remarkable ones are in [LBRM12], [SFLF15] and [HFW11]. The algorithm proposed by Song et al. in [SFLF15] creates self-interlocking structures, to avoid the use of glue or connectors, and it obtains a strong structure that can be disassembled and reassembled multiple times. The algorithm proposed by Hao et al. in [HFW11] tries to minimize the aesthetic impact of seams. Lastly, the algorithm proposed by Luo et al. in [LBRM12] optimizes six objective function, to give the designer the ability to obtain the desired result. An extensive discussion on pros and cons of additive fabrication, subdivision and related issues can be found in the survey of Livesu et al. [LEM*17].

2.2.2. Subtractive manufacturing

Subtractive manufacturing, also known as machining or subtractive fabrication, consists in removing material from a starting block until the desired shape is achieved. The diffusion of CNC milling machines has made this technology very diffused to fabricate regular objects out of wood and metals blocks.

The most diffused and simple milling machines can move their tool on the three axes of the cartesian system and, thus, they have three degrees of freedom. This greatly limits the class of objects they can produce. Indeed, shapes to be milled can be only height-fields, in other words, each line parallel to the z axis can cross the shape only once, as it is shown in Figure 2. In order to produce shapes that are no height-fields, in [HMA15], Herholz et al. proposes an algorithm to subdivide the object. Their subdivision aims to create millable molds which can be used to make the parts of the final model.

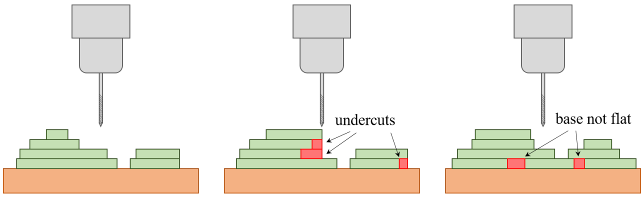


Figure 2: *The model on the left is millable (it is an height-field); the model in the middle is not millable (it is not an height field and, thus, it has undercuts), the model on the right is not millable even if it is an height field because its base is not flat.*

More complex machines have higher degrees of freedom, typically moving the tools over 4 and 5 axes. These machines impose looser constraints over the shape to be produced, but, at the same time, they are more expensive that the 3-axis ones. However, it is possible to add accessories to a 3-axis machine to add a 4th degree of freedom as an axis around which the part can rotate. This is quite useful, as a 4-axis machine can produce any model that, given a rotation axis, exposes every point of the surface in at least one rotation. This constraint is weaker than the one imposed by the 3-axis machines.

3. Problem overview

The goal of our work is to decompose complex models into simpler parts that better suit limitations in current fabrication processes, both additive and subtractive. In order to do so, we want to exploit the polycube structure and reflect this subdivision in the original model. This allows us to have a set of simplified shapes to manage, each one of them with proper features and rotation.

As well stated in [LEM*17], when planning the production of an object in additive manufacturing, it is possible to decide to subdivide the object in multiple pieces. This can be due to multiple reasons but one of the most common situation is when the object is bigger than the printing chamber. An important work on this subject is the one of Luo et al. in [LBRM12]. In subtractive manufacturing the subdivision can be useful if the model violates the constraints imposed by the used technology. Applying a convenient subdivision can possibly lead to pieces that respect the given constraints (height-field condition).

Our work does not aim to solve the problem by proposing a better solution than the ones proposed in literature. Our goal is to study and analyse the subdivision inducted on a shape by its polycube map, in a smart and efficient way. Indeed, while [LBRM12] optimizes a model with six objective functions, giving the designer the ability to obtain multiple different subdivisions, we exploit the unique (and minimal) decomposition of the polycube. Furthermore, we want to study this subdivision addressing both additive and subtractive manufacturing. These technologies impose different constrains, and an explicit differentiation is necessary.

Our proposed production pipeline can be summarized as follows:

1. We start from a 3D input shape
2. We build its polycube representation
3. We subdivide the polycube in cuboids
4. We build a model subdivision induced by the polycube subdivision

The third and the fourth steps of this pipeline are the main focus of this work, as they are the basis for our analysis. A simplified 2D representation of the pipeline is depicted in Figure 3.

We approach the third step of the pipeline creating a perfectly regular hexahedral mesh that correspond to the volume of the polycube. This allows us to perform the decomposition without handling the irregularities of triangles and tetrahedra of the original polycube. This hexmesh is then decomposed with a very simple sweep line algorithm, described in the next section.

The fourth step consists in mapping the cuboids found in the previous step to portions of the original model. Once again, we exploit the hexahedral mesh computed in the previous step, generating a quadrilateral mesh for each polycube's portion.

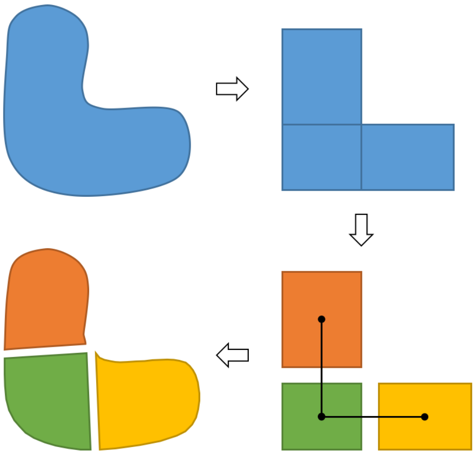
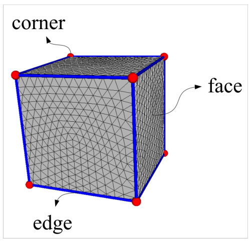


Figure 3: *2D representation of the proposed pipeline.*

4. Polycube subdivision

In the next sections we will refer to the high level representation of a polycube, as shown in the inset, and we will ignore its triangle mesh structure. Basically, a corner is a vertex with at least three adjacent triangles (or faces) having three different normals, an edge is a linking between two corners and a face is a closed chain of corners.



The main step of our pipeline is the Polycube's cuboids decomposition. We follow the idea that every concave edge in the polycube defines a partial decomposition of the model. Each edge is axis-align, hence it lies on two orthogonal planes. The intersections between those planes and the polycube is the aforementioned partial decomposition. Combining the decompositions derived by all the edges we obtain a decomposition in simple cuboids of the polycube.

The first step is to round each corner coordinates to integer values. In this way, by gridding the polycube into an integer lattice, we create a uniform integer grid inside the polycube. The result of this step is an all-regular hexmesh corresponding to the polycube hexmesh. The number of hexahedra composing the hexmesh depends on the dimension of the model, but it can be scaled to easily obtain the desired resolution in the hexmesh.

The next step is to decompose the hexmesh, by taking advantage from the integer values of its vertices. We apply a sweep line algorithm in all the tree axes and we split the lattice at every concave edges.

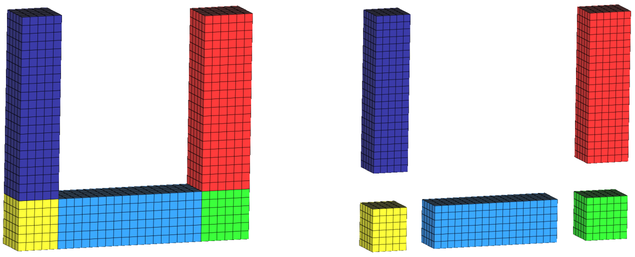


Figure 4: *A grid corresponding to a U-shaped mesh (left) and its natural subdivision (right).*

The major drawback of this method is that the subdivision is computed in the polycube extracted grid, and not in the real polycube. For this reason it cannot be applied to polycubes with geometric inconsistencies, like for example self-intersections. For those situations, a more effective subdivision of the polycube has to be computed.

The final step of the pipeline maps the subdivision onto the model, according to the decomposition of the polycube. We decided to take advantage of the approach used in the previous step, individually quadmeshing all the cuboids in the hexmesh surface.

For each vertex $P(x,y,z)$ of the hexmesh surface we determine in which triangle it lies, using an octree as spatial data structure, and express its position via barycentric coordinates $\omega_0 : \omega_1 : \omega_2$. We then determine the new vertex position P' applying those barycentric coordinates to the triangle in the original model. Given that A, B, C are the vertices of the chosen triangle, we have that $P' = \omega_0 \times A + \omega_1 \times B + \omega_2 \times C$. An example of the final subdivision applied to the model is shown in Figure 5.

This approach is robust and rely on well known techniques in literatures. The complexity in time is linear to the number of vertices in the hexmesh surface. The main advantage of this method is the creation of quite regular boundaries between cuboids. Indeed, as shown in Figure 6, while they are not perfectly planar, the are definitely more regular than surfaces cut out of tetrahedra.

5. Results and Analysis

In this section we report the results of our method on several models. In Figure 7 some examples are shown. For each shape we report the original model, its polycube and the final decomposition. We tested our proposed pipeline with classical 3D models and their polycubes. Furthermore, for some of the model we tested the subdivision with multiple polycubes, generated using the algorithms proposed in [HJS*14] and [LVS*13] and optimized, in terms of corners alignment, using the algorithm proposed in [CLS16]. This allows us to discuss how the polycube's characteristics influence the final results.

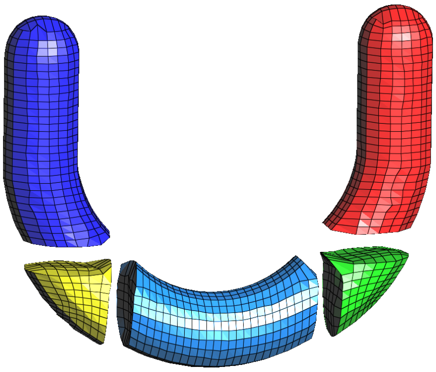


Figure 5: *Explosion of the U-shaped mesh.*

In Table 1 we report, for each model, the number of pieces we decompose it in, and how many of those are internal pieces. This can be relevant as they are not necessary to recompose the object, especially if the structural soundness is not a main goal. We do not report the time elapsed to compute each subdivision, as it depends on the desired resolution of the final model portions and not on the model itself. Furthermore, the resolution of the integer lattice can be changed, allowing us to find the right trade-off between the resolution of the output pieces and the time needed to compute them.

Analysing our results we can state that while the quality of the subdivision is not the best one can we get, it certainly is easy and fast to compute. Our method is not intended to overcome the work of [LBRM12] or other sophisticated decomposition algorithms, but as a study to easily subdivide a mesh for fabrication purposes. Under this aspect, our work is certainly producing good results, as computational time can be well under the minute.

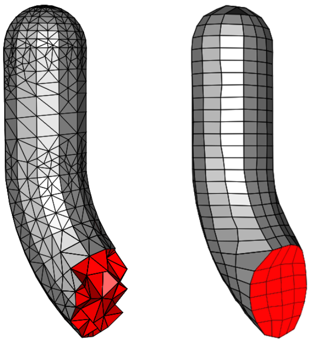


Figure 6: *Comparison between mapping a subdivision cut in the polycube back to tetrahedra (left) and hexahedral mesh (right).*

Model	Produced pieces	Surface pieces	Internal pieces
Airplane	92	85	7
Airplane optimized	27	27	0
Bimba	64	64	0
Bimba Polycut	37	37	0
Bunny	285	231	54
Cubespike	33	32	1
Duck	5	5	0
Eight	18	18	0
Femur	43	43	0
Foot	13	13	0
Good U	5	5	0
Homer	94	89	5
Kitten	163	146	17
Moai	37	37	0
Moai optimized	30	30	0
Sculpt	47	47	0

Table 1: *Results of subdivision performed on multiple models. The optimized models are modified according to [CLS16].*

Looking at the results we can see how different polycubes of the same model yield different subdivisions. Indeed, the more edges (or faces) there have, the more complex the subdivision can get. An interesting aspect is how using optimized polycubes lead to better result, with a good reduction of small and not semantic relevant pieces.

5.1. Subdivision in additive manufacturing

We have already seen in the previous sections that one of the main constraints when using additive technology is the size of the model. With this technology it is possible to print almost any free-form, using temporary supports to handle overhanging features.

The pieces produced by a polycube subdivision allow the complete model to be bigger than the printing chamber. Furthermore, the decomposition allows us to rotate every pieces to minimize the necessity of supports; typically such orientation requires to place the part such as the base layer is the portion that is going to be glued to another. This causes the matching area to be slightly less regular, but composing the model is still easy. On the other hand, if the use of more supports and their placement on the model's surface is not a concern, the border faces matches perfectly.

A condition that is sufficient to avoid supporting structures while printing a shape is the shape being a height-map. This corresponds to the constrain of 3-axis machining as we see in the next section.

5.2. Subdivision in subtractive manufacturing

We have already seen how subtractive technology imposes strict constraints on the model shape. There are differences between 3-axis and 4-axis CNC milling machines manufacturing, since the different constraints causes different results.

Regarding the 3-axis technology, the polycube-based decomposition does not produce a suitable subdivision. Indeed, the border faces are not flat, and this causes the pieces not to be height fields. This is the main weakness identified by our study, and in section 7 we will propose some possible solutions to it.

On the other hand, the 4-axis technology imposes less strict constraints, and a further extension of this method could produce suitable subdivisions, once identified the correct rotation axes.

The main drawback is that, in both cases, the number of parts of a complex model is too high, and assembling them is a painful and prone to errors task.

5.3. 3D printing examples

We were able to make some of the computed decompositions. We present real results only on additive manufacturing since, as explained before, for the subtractive one we are far from a suitable solution.

We fabricated the *Duck* and the *U-shaped model*, the first one is a simple but real model of a common free-form toy, while the second one is an abstract regular surface. These models have pretty simple polycube mappings, and they are both partitioned into five pieces (see Figure 8 for the two decompositions). The real fabrication allowed us to state that the non-planarity of the border surface does not have a major impact when reassembling the parts. On the other hand, we were forced to use support structures while printing them. In Figure 9 we show the reconstruction of the *Duck model*.

6. Limitations

Our analysis showed that the polycube-based subdivision produces too many pieces when it is used to decompose complex models. This causes the subdivision to include really small pieces, that does not correspond to any relevant portion of the model. Optimized polycubes lead to a more compact subdivision, but still present this problem. In Figure 10 we can see the differences between the *Moai* model when it is subdivided by using the original and the optimized polycubes. Passing from 37 to 30 pieces does not really help and

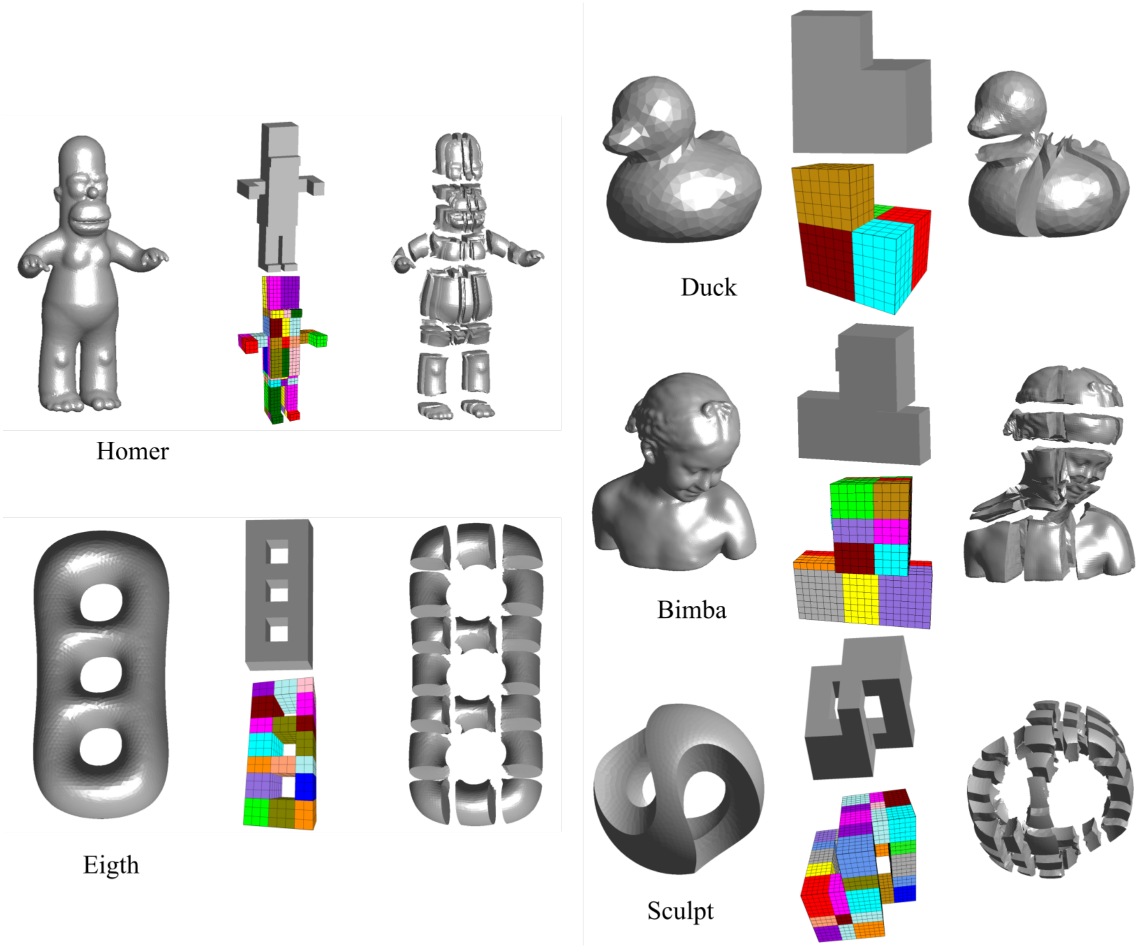


Figure 7: *For each shape, on the left there is the input model, in the central column its polycube(top), and its decomposed polycube (bottom), and, on the right, its subdivision.*

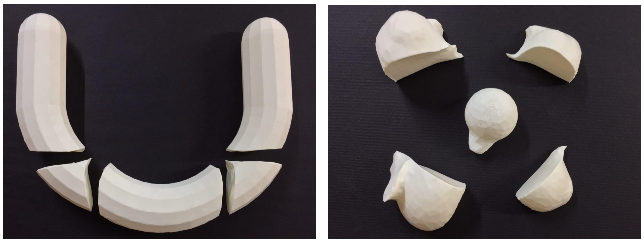


Figure 8: *Subdivision the U-shaped model (left) and the Duck model (right).*

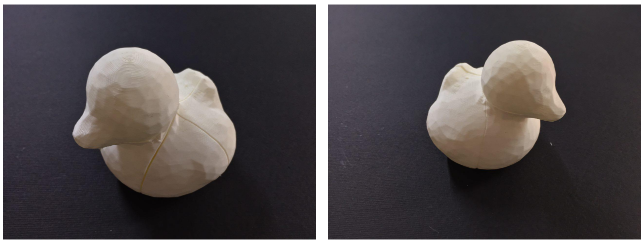


Figure 9: *Pictures of the reassembled Duck.*

both decompositions would be hard to recompose. Moreover, the unoptimized decomposition has small parts and the optimization step does not eliminate all of them.

Another important flaw highlighted by our analysis is the non-planarity of the surface between the produced pieces. This characteristic is well visible in Figure 11. If we could solve this issue, the majority of the pieces would be height-maps, therefore they would be usable in a 3-axis fabrication process.

7. Future work

Our proposed subdivision can be improved in several ways. The main issue we plan to address is to make the subdivision suitable for 3-axis CNC milling machines. In particular, as seen before, the

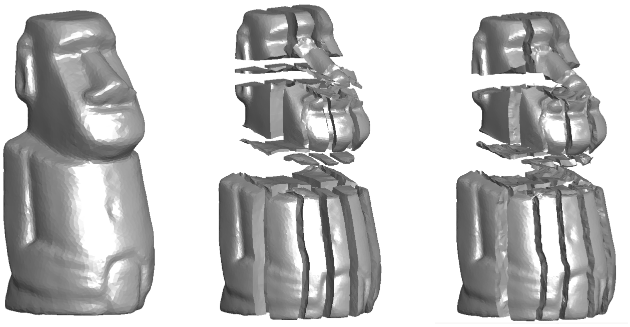


Figure 10: *Moai model (left), and its subdivision using the original and the optimized polycubes (respectively centre and right).*

touching surfaces between pieces are not planar, causing the pieces to be not millable even if they are height-fields.

A possible solution to this problem could be to fit a plane between such surfaces, and move each vertex onto the plane. This would lead to completely planar surface, that could be used as the base surface in machining. However, this solution would not be enough for all the models. In some of the simplest models, pieces are not completely convex, and the subdivision would still be not usable. In more complex models, however, the subdivision is much finer, and it turns to be unuseful in 3-axis machining. Another possible solution could be to split each portion in two sub-portions, by using a plane. Choosing an appropriate plane, every piece of the subdivision (or at least the majority of them) can be split into two height-fields.

Another interesting topic to further investigate is a post-processing step to reduce the number of the produced portions. As it is shown in Table 1, we can have more than a hundred parts, and recomposing them would be an extremely tough task. A possible strategy to face this problem could be to merge adjacent pieces in clusters, keeping in account the printing chamber size (see Figure 12). This would not be a trivial step, as we would have to apply the right constraints to maintain the subdivision suitable for fabrication. Possible constraints may regard the size, to avoid to generate clusters bigger than the printing chamber, and shape, to avoid the creation of concave clusters.

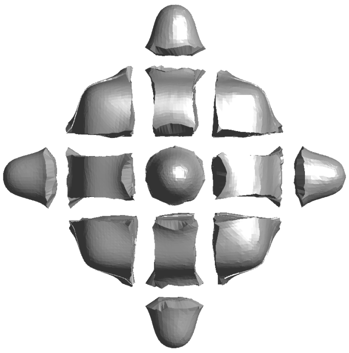


Figure 11: *Front view of an exploded subdivision of the Cubespike.*

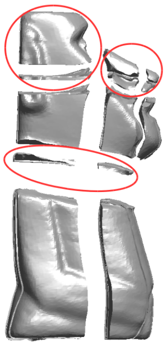


Figure 12: *An example of possible clustering in the Moai model.*

References

[CLS16] Cherchi G., Livesu M., Scateni R.: Polycube simplification for coarse layouts of surfaces and volumes. *Computer Graphics Forum 35*, 5 (2016), 11–20. 2, 4, 5

[GSZ11] Gregson J., Sheffer A., Zhang E.: All-hex mesh generation via volumetric polycube deformation. *Computer Graphics Forum 30*, 5 (2011), 1407–1416. 2

[HFW11] Hao J., Fang L., Williams R. E.: An efficient curvature-based partitioning of largescale stl models. *Rapid Prototyping Journal 17*, 2 (2011), 116–127. 2

[HJS*14] Huang J., Jiang T., Shi Z., Tong Y., Bao H., Desbrun M.: ℓ1basedd construction of polycube maps from complex shapes. *ACM Trans. Graph. 33*, 3 (June 2014), 25:1–25:11. 2, 4

[HLZCO14] Hu R., Li H., Zhang H., Cohen-Or D.: Approximate pyramidal shape decomposition. *ACM Trans. Graph. 33*, 6 (Nov. 2014), 213:1–213:12. 2

[HMA15] Herholz P., Matusik W., Alexa M.: Approximating free-form geometry with height fields for manufacturing. *Computer Graphics Forum 34*, 2 (2015), 239–251. 2

[HWFQ09] He Y., Wang H., Fu C.-W., Qin H.: A divide-and-conquer approach for automatic polycube map construction. *Computers & Graphics 33*, 3 (2009), 369 – 380. IEEE International Conference on Shape Modelling and Applications 2009. 2

[LBRM12] Luo L., Baran I., Rusinkiewicz S., Matusik W.: Chopper: Partitioning models into 3d-printable parts. *ACM Trans. Graph. 31*, 6 (Nov. 2012), 129:1–129:9. 2, 3, 4

[LEM*17] Livesu M., Ellero S., Martínez J., Lefebvre S., Attene M.: From 3d models to 3d prints: an overview of the processing pipeline. *Computer Graphics Forum 36*, 2 (2017), 537–564. 2, 3

[LJFW08] Lin J., Jin X., Fan Z., Wang C. C. L.: Automatic polycube-maps. In *Advances in Geometric Modeling and Processing: 5th International Conference, GMP 2008, Hangzhou, China, April 23-25, 2008. Proceedings*, Chen F., Jüttler B., (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 3–16. 2

[LMPS16] Livesu M., Muntoni A., Puppo E., Scateni R.: Skeleton-driven adaptive hexahedral meshing of tubular shapes. *Computer Graphics Forum 35*, 7 (2016), 237–246. 2

[LVS*13] Livesu M., Vining N., Sheffer A., Gregson J., Scateni R.: Polycut: Monotone graph-cuts for polycube base-complex construction. *ACM Trans. Graph. 32*, 6 (Nov. 2013), 171:1–171:12. 2, 4

[SFLF15] Song P., Fu Z., Liu L., Fu C.-W.: Printing 3d objects with interlocking parts. *Computer Aided Geometric Design 35* (2015), 137 – 148. Geometric Modeling and Processing 2015. 2

[THCM04] Tarini M., Hormann K., Cignoni P., Montani C.: Polycube-maps. *ACM Trans. Graph. 23*, 3 (Aug. 2004), 853–860. 1, 2