

VASCO: Volume and Surface Co-Decomposition for Hybrid Manufacturing: Supplemental Material

FANCHAO ZHONG, Shandong University

HAISEN ZHAO*, Shandong University

HAOCHEN LI, Shandong University

XIN YAN, Shandong University

JIKAI LIU, Shandong University

BAOQUAN CHEN, Peking University

LIN LU†, Shandong University

Additive and subtractive hybrid manufacturing (ASHM) involves the alternating use of additive and subtractive manufacturing techniques, which provides unique advantages for fabricating complex geometries with otherwise inaccessible surfaces. However, a significant challenge lies in ensuring tool accessibility during both fabrication procedures, as the object shape may change dramatically, and different parts of the shape are interdependent. In this study, we propose a computational framework to optimize the planning of additive and subtractive sequences while ensuring tool accessibility. Our goal is to minimize the switching between additive and subtractive processes to achieve efficient fabrication while maintaining product quality. We approach the problem by formulating it as a Volume-And-Surface-CO-decomposition (VASCO) problem. First, we slice volumes into slabs and build a dynamic-directed graph to encode manufacturing constraints, with each node representing a slab and direction reflecting operation order. We introduce a novel geometry property called hybrid-fabricability for a pair of additive and subtractive procedures. Then, we propose a beam-guided top-down block decomposition algorithm to solve the VASCO problem. We apply our solution to a 5-axis hybrid manufacturing platform and evaluate various 3D shapes. Finally, we assess the performance of our approach through both physical and simulated manufacturing evaluations.

CCS Concepts: • **Computing methodologies** → **Shape modeling**; *Graphics systems and interfaces*.

Additional Key Words and Phrases: Surface decomposition, volume decomposition, sequence planning, hybrid manufacturing

ACM Reference Format:

Fanchao Zhong, Haisen Zhao, Haochen Li, Xin Yan, Jikai Liu, Baoquan Chen, and Lin Lu. 2023. VASCO: Volume and Surface Co-Decomposition for Hybrid Manufacturing: Supplemental Material. *ACM Trans. Graph.* 42, 6 (December 2023), 4 pages. <https://doi.org/10.1145/3618324>

*corresponding author

†corresponding author

Authors' addresses: Fanchao Zhong, fanchao98@gmail.com, Shandong University; Haisen Zhao, haisen.zhao@ist.ac.at, Shandong University; Haochen Li, Shandong University; Xin Yan, Shandong University; Jikai Liu, Shandong University; Baoquan Chen, Peking University; Lin Lu, llu@sdu.edu.cn, Shandong University.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.

0730-0301/2023/12-ART \$15.00

<https://doi.org/10.1145/3618324>

A EVALUATION FUNCTIONS DETAILS

We supplement the details of the sub-evaluation functions here. Note that all of the following values need to be reversed in each iteration within the beam search to ensure that the larger values have higher priority.

Block connectivity. Each block may consist of multiple connected components, which will lead to a large amount of transfer movement of the nozzle in additive manufacturing, thus reducing the manufacturing efficiency. So we would like to generate block that contains as few connected components as possible.

Changing in print directions. We wish each block to use a similar orientation to the previous block. The sub-evaluation function is as follows:

$$f_{orien}(x) = \begin{cases} \|\vec{ori}_x - \vec{ori}_a\|, & a \neq \text{root node} \\ 0, & a = \text{root node} \end{cases} \quad (1)$$

where \vec{ori}_x and \vec{ori}_a respectively are the additive process orientations of x and the ancestor node of x .

Structure soundness. Some regions such as "bridges" and "thin fins" are fragile, the cutting planes should avoid them. We use the geometric analysis method of [Luo et al. 2012] to identify these regions and design a similar function:

$$f_{plane}(x) = \sum_{p \in P} (\min_{v \in V} (\text{Distance}(p, v)) < \lambda) \quad (2)$$

where P is a set of all the cutting planes in x and V indicates all the fragile points in x . λ is the distance threshold, which is set to 6.0 mm in our implementation.

Pyramid-like shapes. As we know, the pyramid structure is extremely stable, while the inverted pyramid is quite the opposite. The main difference between them is the proportion of the area of the contact surface and the projected area of the model towards the contact surface. We wish blocks to form the structure. For making blocks form the pyramidal structure, we propose a sub-evaluation function:

$$f_{stability}(x) = \sum_{p \in P} \frac{\text{Area}(\text{Project}(p))}{\text{Area}(p) * |P|} \quad (3)$$

Project(p) indicates the projected area beyond cutting plane p of a connected component, which is crossed by p .

B TOOL SETTING

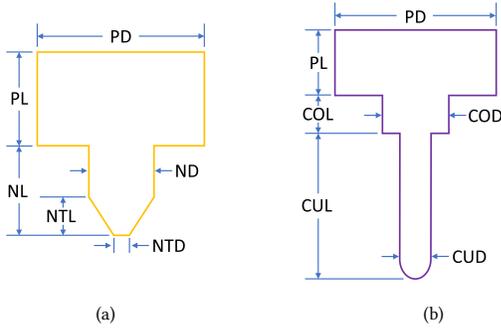


Fig. 1. Illustration of the shape of the printing tool and cutting tool we used in our setting. We use a taper nozzle (a) to conduct additive manufacturing, while a ball-end CNC cutter (b) is being used for subtractive manufacturing.

We illustrate the shape and dimensions of the printing head used in the AM steps. The printing head consists of a platform on the top and a taper nozzle on the bottom, as depicted in Figure 1 (a). The platform with length PL and diameter PD is mainly served as obstacles for collision detection. In our setting, we set the platform as an infinite plane, that is, PD is positive infinity. Thus PL can be any positive real number, which doesn't affect the result. The taper nozzle's length (NL) is 15mm and its diameter (ND) is 9mm, while nozzle tip length (NTL) is 5mm and its diameter (NTD) is 2mm.

The cutting tool used in our SM setup is depicted in Figure 1 (b). The cutting tool consists of three parts, a platform on the top, a collet in the middle and a ball-end CNC cutter on the bottom. Similar to the printing head configuration, the platform and collet primarily serve as collision detection obstacles. We consider the platform as an infinite plane, where PD is set to positive infinity. Consequently, PL can take any positive real value without affecting the results. The collet, with a length (COL) of 30mm and a diameter (COD) of 50mm, enables access to the void of the model and facilitates processing of deeper surfaces. While the ball-end CNC cutter is employed for carving the external 3D surface, it also serves as a collision object. The cutter has a total length (CUL) of 26.5mm and a diameter (CUD) of 3mm.

C TERMINOLOGY LIST

Table 1. This table lists all parameters used in our VASCO algorithm. Each row indicates the parameter (**Param**), the meaning of the parameter (**Significance**), and the related section where the parameter appears for the first time (**Usage**).

Param	Significance	Usage
A_i	additive manufacturing step	Section 1
S_i	subtractive manufacturing step	Section 1
\mathbb{B}	ordered sequence blocks	Section 3.1
B_i	the i th block	Section 3.1
N	number of blocks	Section 3.1
M	input 3D object	Section 3.1
M_i	sub-part of M	Section 3.1
$\bar{U}M$	inner volume of M	Section 3.1
∂M	outer surface of M	Section 3.1
D_i^{AM}	printer head directions in AM to fabricate $\bar{U}M_i$	Section 3.1
D_i^{SM}	CNC cutter directions in SM to fabricate ∂M_i	Section 3.1
M_i	the current realizing object	Section 3.1
α_{max}	the maximal self-supporting angle	Section 3.2
n_p	the unit normal vector of point p	Section 3.2
i	iteration of the top-down decomposition	Section 4
$d_{i,j}^{AM}$	a randomly sampling printer head direction	Section 4
G_i^{ASM}	hybrid block graph	Section 4
G_i^{SM}	subtractive block graph	Section 4
G_i^{AM}	additive block graph	Section 4
\mathcal{D}^{SM}	the set of sample potential SM cutter direction	Section 4.1
\mathcal{P}	the set of surface sample points	Section 4.1
p_i	the i th surface sample point	Section 4.1
W_{Beam}	the number of candidate solutions in a beam search	Section 5
N_{Dir}^{AM}	number of potential sample potential AM directions	Section 5.1
\mathcal{D}^{AM}	the set of sample potential AM directions	Section 5.1
w_k	weight parameter in an equation	Section 5.2
S_B	the set of all slabs in B	Section 5.2
s_B	one slab of S_B	Section 5.2
$Length(s_B)$	the length of the bottom contour of s_B	Section 5.2
$Priority(s_B)$	the total priority value of all surface points of s_B	Section 5.2

Table 2. This table lists all the terms used in this paper. Each row indicates the meaning of the terminology(term) and the related section where the term used for the first time(Usage).

Term	Usage
slab : the portion between two consecutive planar layers.	Section 1
sequence planning : the task to determine of the alternating additive and subtractive step sequence.	Section 1
tool accessibility : the print head or CNC cutter does not allow any collisions with the realized shape and the ASHM machine.	Section 1
manufacturing dependency constraints : manufacturing operation can not be applied before the precedence steps on which it relies.	Section 1
hybrid-fabricability : the property of a hybrid-fabricable block.	Section 1
hybrid-fabricable block : a pair of 3D volumes and surfaces that can be accessible by the associated manufacturing directions of ASHM.	Section 1
block : abbreviates the hybrid-fabricable block as block.	Section 1
dynamic accessibility(shape) : the shape to be realized changes dramatically after each "AM-then-SM" stage, with the resulting dynamic variations of tool accessibility.	Section 1
slab-based dynamic-directed graph : a graph where each node is a slicing slab and edges between nodes are well-defined to formulate the tool accessibility and the manufacturing dependency constraints.	Section 1
inaccessible regions : surface regions that the subtractive cutter cannot reach, supposing the entire model has been additively manufactured	Section 1
manufacturing constraints : additive accessibility, subtractive accessibility, manufacturing dependency constraint, and self-support constraint.	Section 3
realized part(shape, object) : $\cup_{j=i+1}^N M_j$ as the realized shape for M_i , M_i can be only manufactured once all the subsequent $\cup_{j=i+1}^N M_j$ have been manufactured by ASHM.	Section 3
realizing part(shape, object) : the current part $\overline{M}_i = M \setminus \cup_{j=1}^{i-1} M_j$.	Section 3.1
Additive accessibility constraint : collision-free between the print head and the realized part.	Section 3.2
Subtractive accessibility constraint : collision-free between the machining cutter and the realized part.	Section 3.2
hybrid block graph : a dynamic-directed graph whose nodes are slicing slabs and edges encode ASHM manufacturing constraints.	Section 4
subtractive block graph : encodes the subtractive accessibility constraint to carve $\partial\overline{M}_i$ by SM from a group of potential SM directions.	Section 4.1
additive block graph : encodes the additive accessibility constraint, additive dependency constraint, and self-support constraint to print $\overline{U}\overline{M}_i$ by AM.	Section 4.2
block desired properties : the properties helps to shorten the fabrication sequence.	Section 5.2

D ALGORITHM PSEUDO-CODE

This section provides the pseudocode for the proposed algorithms. Algorithm 1 corresponds to the content in Section 4 of the main text. Algorithm 2 describes an algorithm mentioned in Section 3.1. Algorithm 3 corresponds to the content in Section 5 of the main text of our paper.

Algorithm 1 Co-decomposition Method

```

1: Function CoDecompositionMethod ( $M, i, \overline{M}_i, d_i^{AM}$ )
2: Input: Input 3D object  $M$ ; Block decomposition iteration  $i$ ;
   Realizing object  $\overline{M}_i$ ; A AM direction  $d_i^{AM}$ ;
3: Output:  $B_i$ ; // the  $i$ -th block of  $M$ 
   // Data structure setting
4: Hybrid block graph  $G_i^{ASM}$ ; Subtractive block graph  $G_i^{SM}$ ;
5: Additive block graph  $G_i^{AM}$ ;
   // Generate the subtractive block graph  $G_i^{SM}$ 
6: Uniformly sample  $N_{Dir}^{SM} \leftarrow 200$  SM directions  $\mathcal{D}^{SM}$ ;
7:  $n \leftarrow N_{Dir}^{SM}$ ;  $\mathcal{D}^{SM} \leftarrow \{d_1^{SM}, d_2^{SM}, \dots, d_n^{SM}\}$ ;
8: if  $i == 1$  then
9:   Uniformly sampling points  $\mathcal{P} = \{p_1^{SM}, p_2^{SM}, \dots, p_n^{SM}$  on  $\partial M$ 
   by the Lloyd's Voronoi relaxation algorithm;
10:  Build  $G_1^{SM}$  following the method in section 4.1;
11: else
12:  Build  $G_i^{SM}$  by removing nodes and edges of  $G_1^{SM}$ ;
13: end if
   // Generate the additive block graph  $G_i^{AM}$ 
14: Build  $G_i^{AM}$  following the method in section 4.2;
   // Generate the hybrid block graph  $G_i^{ASM}$ 
15: Build  $G_i^{ASM}$  by integrating  $G_i^{SM}$  and  $G_i^{AM}$ ;
   // Generate the hybrid-fabricable block  $B_i$ 
16: Generate  $B_i$  by a node merging procedure (section 4.4);
17: return  $B_i$ ;

```

Algorithm 2 Iteration Top-down Block Decomposition

```

1: Function TopDownBlockDecomposition( $M$ )
2: Input:  $M$ ; // input 3D object
3: Output:  $\mathbb{B} = \{B_1, B_2, \dots, B_N\}$ ; // a sequence of blocks
   // Data structure setting
4: Top-down decomposition iteration  $i \leftarrow 1$ ;
5: Realizing object  $\overline{M}_i$  of  $i$ -th iteration;  $\overline{M}_i \leftarrow M$ ;
6: A queue of blocks  $\mathbb{B} \leftarrow \emptyset$ ;
   // Top-down block decomposition
7: while  $\overline{M}_i \neq \emptyset$  do
8:   Randomly determine an AM direction  $d_i^{AM}$ ;
9:    $B_i \leftarrow$  CoDecompositionMethod ( $M, i, \overline{M}_i, d_i^{AM}$ );
10:   $\mathbb{B} \leftarrow \mathbb{B} \cup \{B_i\}$ ;
   // Update data structure
11:   $i \leftarrow i + 1$ ;
12:   $\overline{M}_i = M \setminus \cup_{j=1}^{i-1} M_j$ ;
13: end while
14: return  $\mathbb{B}$ ;

```

Algorithm 3 Beam-guided Optimization

```

1: Function BeamGuidedOptimization ( $M$ )
2: Input:  $M$ ; // input 3D object
3: Output:  $\mathbb{B} = \{B_1, B_2, \dots, B_N\}$ ; // a sequence of blocks
   // Data structure setting
4: A queue of blocks  $\mathbb{B} \leftarrow \emptyset$ ; Beam search iteration  $k \leftarrow 1$ ;
5: Beam width  $W_{Beam} \leftarrow 8$ ; Solution sets  $\mathbb{C} \leftarrow \emptyset, \mathbb{T} \leftarrow \emptyset$ ;
   // AM direction sampling
6: Uniformly sample  $N_{Dir}^{AM} \leftarrow 100$  AM directions,  $\mathcal{D}^{SM}$ ;
7:  $n \leftarrow N_{Dir}^{AM}$ ;  $\mathcal{D}^{AM} \leftarrow \{d_1^{AM}, d_2^{AM}, \dots, d_n^{AM}\}$ ;
   // Beam search initialization
8: for each AM direction  $d^{AM} \in \mathcal{D}^{SM}$  do
9:    $\mathbb{C} \leftarrow \mathbb{C} \cup \{\text{CoDecompositionMethod}(M, k, M, d^{AM})\}$ ;
10: end for
   // Beam search optimization
11: while True do
   // Top candidates selection
12:    $\mathbb{C} \leftarrow$  optimal  $W_{Beam}$  solution of  $\mathbb{C}$  with  $f_{priority}(B)$ ;
   // Terminal condition verification
13:   for each block  $B_j$  in  $\mathbb{C}$  do
14:      $M_j \leftarrow$  the decomposed sub-part of  $B_j$  and its previous
     blocks during the beam search iteration;
15:     if  $M_j == M$  then
16:        $\mathbb{T} \leftarrow \mathbb{T} \cup \{B_j\}$ ;
17:     end if
18:   end for
19:   if  $\mathbb{T} \neq \emptyset$  then
20:      $B_j \leftarrow$  the block in  $\mathbb{T}$  with the highest  $f_{priority}$ ;
21:      $\mathbb{B} \leftarrow$  previous blocks of  $B_j$  during beam search;
22:     return  $\mathbb{B} \leftarrow \mathbb{B} \cup \{B_j\}$ ;
23:   end if
   // Candidate solution generation
24:   for each block  $B_j$  in  $\mathbb{C}$  do
25:      $M_j \leftarrow$  the decomposed sub-part of  $B_j$  and its previous
     blocks during the beam search iteration;
26:      $\overline{M}_j = M \setminus M_j$ ;
27:     for each AM direction  $d^{AM} \in \mathcal{D}^{SM}$  do
28:        $\mathbb{T} \leftarrow \mathbb{T} \cup \{\text{CoDecompositionMethod}(M, k, \overline{M}_j, d^{AM})\}$ ;
29:     end for
30:   end for
31:    $\mathbb{C} \leftarrow \mathbb{T}; \mathbb{T} \leftarrow \emptyset; k \leftarrow k + 1$ ;
32: end while

```

REFERENCES

Linjie Luo, Ilya Baran, Szymon Rusinkiewicz, and Wojciech Matusik. 2012. Chopper: partitioning models into 3D-printable parts. *ACM Transactions on Graphics* 31, 6 (nov 2012), 1–9. <https://doi.org/10.1145/2366145.2366148>