Waste-to-Value: Reutilized Material Maximization for Additive and Subtractive Hybrid Remanufacturing

FANCHAO ZHONG, Shandong University, China
ZHENMIN ZHANG, Shandong University, China
LIYUAN WANG, Shandong University, China
XIN YAN, Shandong University, China
JIKAI LIU, Shandong University, China
LIN LU, Shandong University, China
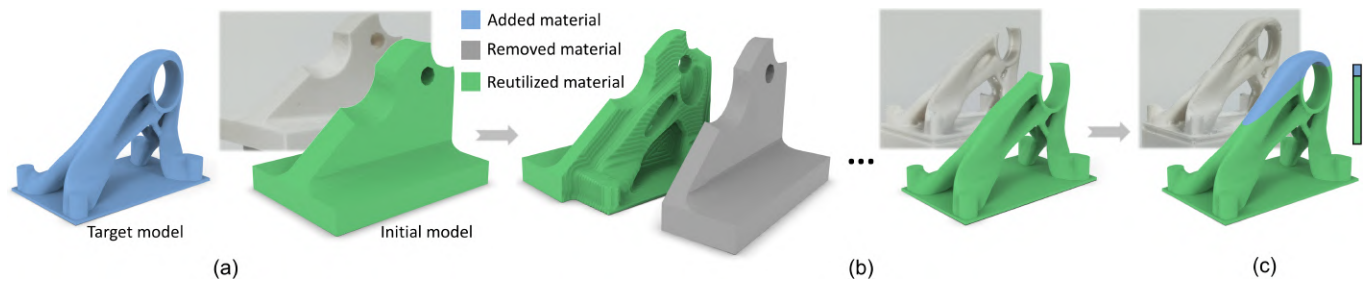HAISEN ZHAO*, Shandong University, China

Fig. 1. Turning damaged or discarded parts into high-value components unlocks hidden potential and gives waste a second life with purpose. Starting with the initial worn *Bearing* model (green) and the target topologically optimized *Bearing* model (blue) (a), our computational framework generates process planning solution aimed at maximizing material savings by reutilizing as much volume as possible while ensuring manufacturability. Excess material is removed via subtractive manufacturing (b), and additional material is subsequently added through additive manufacturing (c) to obtain the target model. Compared to fabricating the target model entirely using additive manufacturing, our remanufacturing process planning solution achieves material savings of up to 90%.

Remanufacturing effectively extends component lifespans by restoring used or end-of-life parts to like-new or even superior conditions, with an emphasis on maximizing reutilized material, especially for high-cost materials. Hybrid manufacturing technology combines the capabilities of additive and subtractive manufacturing, with the ability to add and remove material, enabling it to remanufacture complex shapes and is increasingly being applied in remanufacturing. How to effectively plan the process of additive and subtractive hybrid remanufacturing (ASHRM) to maximize material reutilization has become a key focus of attention. However, current ASHRM process planning methods lack strict consideration of collision-free constraints, hindering practical application. This paper introduces a computational framework to tackle ASHRM process planning for general shapes with strictly considering these constraints. We separate global and local collision-free constraints, employing clipping planes and graph to tackle them respectively, ultimately maximizing the reutilized volume while ensuring these constraints are satisfied. Additionally, we also optimize the setup of the target model that is conducive to maximizing the reutilized volume. Extensive experiments and physical validations on a 5-axis hybrid manufacturing platform demonstrate the effectiveness of our method across various 3D shapes, achieving an average material reutilization of 69% across 12 cases. Code is publicly available at https://github.com/fanchao98/Waste-to-Value.

CCS Concepts: • **Computing methodologies → Computer graphics**; *Shape analysis*.

Additional Key Words and Phrases: Remanufacturing, hybrid manufacturing, process planning, volume optimization

*corresponding author

Authors' addresses: Fanchao Zhong, fanchaoz98@gmail.com, Shandong University, Qingdao, China; Zhenmin Zhang, zhenminzhang.cn@gmail.com, Shandong University, Qingdao, China; Liyuan Wang, 13011651855@163.com, Shandong University, Qingdao, China; Xin Yan, io.yanxin@gmail.com, Shandong University, Jinan, China; Jikai Liu, jikai_liu@sdu.edu.cn, Shandong University, Jinan, China; Lin Lu, lulin.linda@gmail.com, Shandong University, Qingdao, China; Haisen Zhao, haisenzhao@sdu.edu.cn, Shandong University, Qingdao, China.

1 INTRODUCTION

In the face of rising concerns about emissions, resource depletion, and environmental sustainability, finding effective solutions for worn, damaged, and discarded components is increasingly critical. Remanufacturing technology has gained significant attention from both industry and academia [D'Adamo and Rosa 2016; Sitcharangsie et al. 2019]. Remanufacturing restores used or worn products to like-new conditions or modifies their form to meet the performance requirements of new products [Zheng et al. 2020]. Compared to
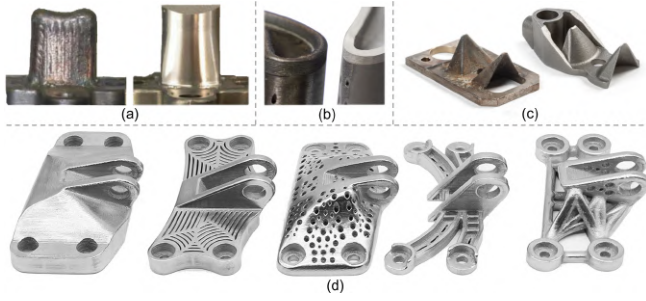
Fig. 2. Typical applications of remanufacturing. (a),(b) The mold cores[1] and impellers[2] are prone to wear during use. Remanufacturing enables localized repair, avoiding the replacement of the entire part and thus saving significant costs. (c) Discarded parts[3] can be remanufactured into new, usable components, reducing inventory. (d) Heavy parts[4] can be simplified into lightweight, topology-optimized parts through remanufacturing.

manufacturing from scratch, remanufacturing reduces material, energy, and time losses, with the primary benefit of extending product life and enhancing functionality by restoring used or damaged components to a near-new condition. Remanufacturing offers significant practical and impactful applications, including 1) part repair, 2) part upgrading and retrofitting, 3) on-site manufacturing, and 4) waste material reuse, providing significant market potential [Fofou et al. 2021]. Minimizing material waste is a primary goal in remanufacturing, which is equivalent to **maximizing reutilized volume** [Paris et al. 2018]. This objective is especially critical in industries such as aerospace and molding, where high-cost materials are prevalent, as illustrated in Figure 2. Notably, remanufactured metal components can achieve mechanical strength close to that of new, monolithic parts [Wilson et al. 2014].

Additive and subtractive hybrid manufacturing (ASHM) combines additive manufacturing (AM) and subtractive manufacturing (SM). It enables both material addition and removal, facilitating the fabrication of geometrically complex models. ASHM is increasingly applied in remanufacturing [Grzesik 2018; Hassanin 2021]. The basic process of **additive and subtractive hybrid remanufacturing (ASHRM)** is illustrated in Figure 3 (a)~(c), where SM removes excess material from initial model, and AM subsequently adds additional material to the remaining material to achieve the target model. We define the removed material as *subtractive volume*, the additive material as *additive volume* and the remaining material as *reutilized volume*[5].

The key task in ASHRM process planning is to determine the three types of volumes while aiming to maximize the reutilized volume. This can initially be calculated through Boolean operations between the initial and target models (Figure 3 (a)), where the intersection resulting volume is the *ideal reutilized volume*, i.e., the maximum achievable reutilized volume.
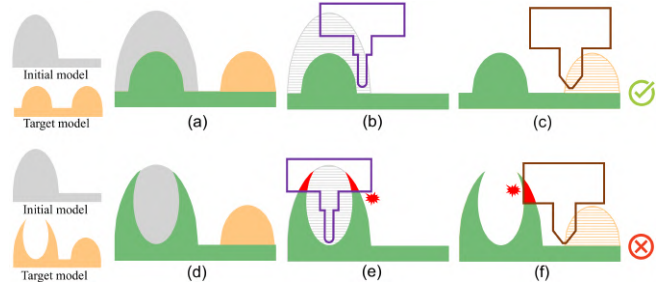
Fig. 3. For a simple pair of initial and target models, using Boolean operation categorize three type of volumes (a): subtractive volume (gray), additive volume (orange), and ideal reutilized volume (green). The ASHRM process can be completed by separately performing SM on the subtractive volume and AM on the additive volume (b, c). However, for more complex model pairs (d), collision-free constraints could be violated during the SM/AM processes (e, f): the cutter and nozzle are inaccessible in the SM/AM processes.
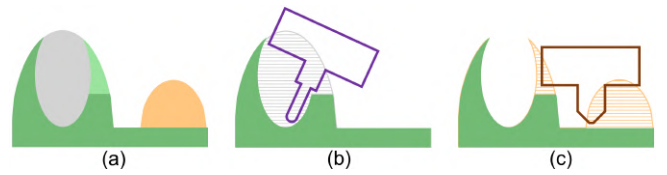

Fig. 4. Transforming the right portion of the ideal reutilized volume into SM-then-AM volume (light green in (a)) maximizes reutilization while ensuring collision-free operations in both SM (b) and AM (c).

However, for complex models, the additive and subtractive volumes often fail to satisfy the required SM/AM collision-free constraints, as shown in Figure 3 (d)~(f). To address this issue, as shown in Figure 4, part of the ideal reutilized volume has to be converted into *SM-then-AM volume* (light green), which represents material first removed by SM and then re-added by AM. The above process can be formulated as a volume optimization problem, **aiming to maximize the "reutilized volume" (shown in dark green in Figure 4 (a)), defined as "ideal reutilized volume" - "SM-then-AM volume", while ensuring that the reutilized volume satisfies all collision-free constraints.**

Ensuring collision-free operations throughout the manufacturing process is essential for successful production. However, current ASHRM process planning often neglects these constraints, increasing the risk of collisions and manufacturing failures, especially for complex models. The complexity of ASHRM process planning under collision-free constraints arises from the following aspects:

1) *Volume Optimization with Dynamic Constraints.* The volume optimization problem can be formulated as a generalized variational problem, aiming to find a volume density function that maximizes the reutilized volume. Collision-free constraints complicate the problem, as the reutilized volume acts as dynamic obstructions—once regions are removed, they are excluded from the constraints, causing the constraint terms to evolve during optimization.

2) *Heterogeneous Collision-Free Constraints.* Collision-free constraints are categorized as local (Figure 5) and global (Figure 3 (e, f)). Local constraints refer to collisions between the nozzle's outlet and the initial printing base which belongs to the reutilized volume surface. In contrast, global constraints account for potential collisions between the entire cutter/nozzle assembly and the dynamically changing existing material. This heterogeneity significantly enlarges the search space, especially when the initial solution violates the constraints.

This paper presents a computational framework for ASHRM process planning on general shapes, aiming to maximize reutilized volume. To address the challenges associated with collision-free constraints, we propose two basic ideas: 1) The generalized variational problem is reformulated as a combinatorial optimization problem by introducing clipping planes. By optimizing their number and poses to determine the reutilized volume, the original volumetric density function is replaced, significantly reducing complexity. 2) A two-stage approach is adopted, where local constraints are resolved first through a finer, localized search, followed by a Monte Carlo optimization to address global constraints.

We deploy our algorithm on 5-axis hybrid manufacturing platform, utilizing 3+2 axis positional manufacturing strategy for both SM and AM. Compared to simultaneous manufacturing[6], positional manufacturing reduces the complexity of process planning and provides better stability during manufacturing.

In conclusion, the main contribution of this paper is the first general computational framework for ASHRM process planning of general shapes, aimed at maximizing reutilized volume while ensuring that the reutilized volume satisfies all collision-free constraints. In our experiment, we validate the effectiveness of our algorithm on various 3D shapes. On average, 12 models save 69% of materials.

2 RELATED WORK

This section first reviews process planning approaches for ASHM and ASHRM, outlines ASHRM applications, and then discusses volume optimization methods for manufacturing in computer graphics.

2.1 Process Planning for ASHM

Additive and Subtractive Hybrid Manufacturing (ASHM) combines material addition and removal processes. Its process planning aims to generate alternating additive and subtractive sequences to fabricate the target model with smooth surfaces while satisfying collision-free constraints using the fewest sequences possible [Dilberoglu et al. 2021]. ASHM process planning has been explored for various model types, including planar surfaces [Zhang et al. 2020], symmetric shapes [Chen and Frank 2019], columnar shapes [Chen et al. 2018], and general free-form 3D objects [Chen et al. 2025; Sun et al. 2024; Zhong et al. 2023b].

In addition to planar slicing strategies, recent studies have explored curved-layer additive manufacturing, often implemented using multi-axis robotic arms [Liu et al. 2024, 2025; Zhang et al. 2022]. Such approaches can relax the local collision-free constraint

by conforming the deposition path to the underlying surface. However, these methods require complex motion planning and specialized hardware, and their integration with subtractive operations in ASHM remains challenging. For these reasons, our work focuses on planar-layer strategies within the hybrid manufacturing framework.

Leveraging ASHM's capability, this paper focuses on the SM-then-AM hybrid manufacturing process that transforms an initial model into a target model while maximizing the reutilized volume.

2.2 Process Planning for ASHRM

Existing studies on ASHRM process planning can be divided into two groups oriented toward different model types: CAD models and freeform models. For CAD models, most of studies address the process planning problem based on manufacturing features extracted from CAD models [Das et al. 2024; Zheng and Ahmad 2020a,b], in which a graph is constructed to encode the relationships of manufacturing features [Paris et al. 2017], the initial-part features (IPFS) are defined for the Re-Plan process planning in the proposed iAtrative framework in [Newman et al. 2015]. However, these feature-based approaches cannot be applied to freeform models due to the absence of well-defined manufacturing features in freeform models, and they also lack strict consideration of collision-free constraints.

For freeform models, only a limited number of studies have been conducted so far. [Liu et al. 2020; Xu et al. 2024] integrate ASHRM process planning within topology optimization but does not allow specifying the target model. [Teibrich et al. 2015] achieves ASHRM using a 3D printer enhanced with an additional machining module. Although it considers global collision-free constraint, it does not address local collision and relies solely on a simple greedy algorithm, limiting its applicability to complex geometries. The latest work [Ma et al. 2025] explores adaptive process planning for freeform surfaces but still remains limited to simple height function surface.

To our knowledge, this work is the first to strictly consider collision-free constraints in ASHRM process planning while being applicable to general shapes.

2.3 Applications of ASHRM

In the realm of hybrid manufacturing systems for remanufacturing, 1) part repair is a crucial application, aimed at fixing damaged parts using a subtractive process to generate a clean surface and an additive process to complete the repair work [Kanishka and Acherjee 2023; Liang et al. 2024; Ren et al. 2006; Zhang et al. 2018], especially for critical industrial components, such as Turbine Blades [Jones et al. 2012; Qian et al. 2024; Wilson et al. 2014], and casting die [Zhang et al. 2019; Zhang and Liou 2019]. The ASHRM process planning for part repairing is relatively simple due to the high overlap between the initial and target parts. 2) Upgrading discarded parts using ASHRM enables the transformation of waste parts into valuable, application-specific components [Paris et al. 2017; Teibrich et al. 2015]. It can also be applied to the topology optimization of discarded parts [Liu et al. 2020; Xu et al. 2024]. 3) ASHRM of spare parts enables the rapid production of urgently needed target models from stored simple initial models, such as cubic stock, which holds great potential for applications in extreme environments [Chen and Frank 2019; Kwon and Oh 2023].

---

[6]Positional manufacturing: fixes the rotary axes before manufacturing, with only linear axes active during the process. Simultaneous manufacturing: enables continuous and coordinated motion of all axes during manufacturing.

2.4 Volume Optimization for Manufacturing

As previously mentioned, the ASHRM process planning problem can be reformulated as a specialized volume optimization problem with manufacturing constraints. In the graphics domain, previous studies have implemented various volume optimization techniques tailored to specific manufacturing contexts. For example, large parts are decomposed into smaller pieces to fit within the limited printing space of 3D printers [Song et al. 2016]; multi-attribute inputs are divided into assemblable and surface-conforming single-attribute parts [Araújo et al. 2019]; shell thicknesses are refined to enhance the stability of extrusion-based ceramics printing [Xing et al. 2021]; geometry modifications enable the printability of 3D trees for additive manufacturing [Bo et al. 2017]; shell models are adjusted through global deformation and local thickening to ensure structural stability after fabrication [Xing et al. 2024]. Slight deformations of external surfaces are introduced to ensure height-fieldness [Herholz et al. 2015], facilitating fabrication processes like injection molding that relay on height-field surfaces.

While these methods optimize volumes for specific manufacturing scenarios, ASHRM requires simultaneous consideration of additive and subtractive constraints, making volume optimization uniquely complex.

3 OVERVIEW

Starting with M as the initial model and M^* as the target, our algorithm first determines their relative pose and then aims to generate a feasible process plan for remanufacturing M^* from M, while maximizing the reutilized volume. The resulting subtractive volume and additive volumes should meet the manufacturing constraints detailed in subsection 3.1. This section then presents an overview of our algorithm (subsection 3.2).

3.1 Collision-free Constraints

Recall that we utilize the 3+2 axis positional strategies in both SM and AM. This primarily necessitates avoiding collisions while removing the subtractive volume (V_S) and adding the additive volume (V_A), where the reutilized volume (V_R) and fabrication base serve as fixed "obstructions". Such collision-free requirements can be categorized into two types: global collision-free constraint (additive and subtractive accessibility) and local collision-free constraint (additive plano-convex constraint).

Additive and subtractive accessibility constraint. This constraint prevents potential global collisions between the manufacturing tool (machining cutter or printing nozzle) and the dynamically changing existing material, as illustrated in Figure 3 (e, f). For simplicity, the terms additive and subtractive accessibility constraint are collectively abbreviated as the accessibility constraint in the text below.

Additive plano-convex constraint. According to [Wu et al. 2020], in the 3+2 positional strategy for additive manufacturing, the nozzle must print on *piecewise-plano and convex surfaces* (Figure 5 (c)) to avoid local collisions (Figure 5 (a, b)). We therefore define the additive plano-convex constraint to prevent collisions between nozzle outlet and *initial printing base* ($\mathcal{B}$), i.e., the base surface where printing starts.
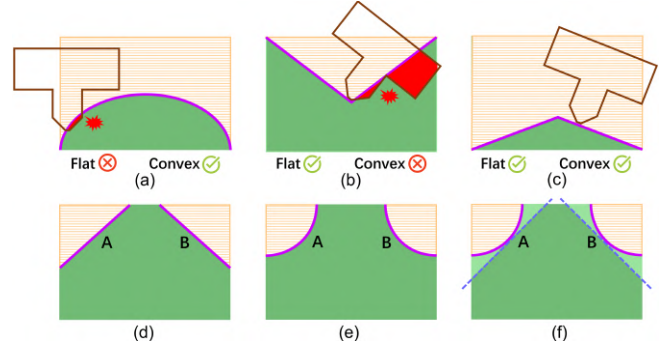

Fig. 5. Illustration of the plano-convex constraint. The printing bases (purple lines) between the reutilized volume and the additive volume. For a single connected component, the base violates the piecewise-flat constraint in (a) and the convex constraint in (b), respectively, while (c) illustrates a valid case. For two connected components, "A" and "B", (d) shows a valid configuration. In contrast, (e) violates the plano-convex constraint, which can be resolved by introducing two clipping planes, as shown in (f).

In ASHRM, $\mathcal{B}$ denotes the contact surfaces between V_R and V_A (purple lines in Figure 5 (d)), on which both "A" and "B" satisfy the plano-convex constraint. In the following, "plano-convex constraint" is used as shorthand for "additive plano-convex constraint".

Infeasible initial printing base under plano-convex constraint. Let $\widetilde{V}_R$, $\widetilde{V}_S$, and $\widetilde{V}_A$ represent the reutilized, subtractive, and additive volumes, respectively, derived from the initial Boolean operation[7]. As depicted in Figure 5 (e), the initial printing base $\widetilde{\mathcal{B}}$ of "A" and "B" does not meet the plano-convex constraint. To resolve this, a set of clipping planes are included to divide the ideal reutilized volume into two sections, resulting that the new initial printing base of V_R and V_A are feasible to the plano-convex constraint; see Figure 5 (f).

3.2 Algorithm Overview

As mentioned in the first basic idea in the introduction, we introduce clipping planes to simplify the volume optimization problem, and reformulated the optimization objective:

$$\begin{array}{ll} \arg\max_{v(\mathbf{x})} \int_{\Omega} v(\mathbf{x})\,d\mathbf{x} & \arg\max_{\mathbb{C}} V_R(\mathbb{C}) \\ \text{s.t. } g_{\text{plano-convex}}(V_A) = 0 & \Rightarrow \quad \text{s.t. } g_{\text{plano-convex}}(V_A) = 0 \\ \quad g_{\text{accessibility}}(V_S, V_A) = 0 & \quad g_{\text{accessibility}}(V_S, V_A) = 0 \end{array}$$

(1)

where $\mathbb{C}$ represents the set of clipping planes, $v(x)$ represents a continuous volumetric density function of the reutilized volume, $V_R(\mathbb{C})$ refers to the reutilized volume clipped by $\mathbb{C}$ and $g(\cdot)$ denotes collision regions. Figure 6 illustrates our algorithm pipeline, with the core two-stage algorithm as follows:

- *Reutilized volume initialization.* The clipping planes $\mathbb{C}$ are initialized by formulating a weighted set cover problem, which narrows the task to a classical combinatorial optimization and allows specialized algorithms to efficiently compute the optimal solution, leading to a larger reutilized volume that

---

[7] $\widetilde{V}_R = M \cap M^*$, $\widetilde{V}_S = M - \widetilde{V}_R$, $\widetilde{V}_A = M^* - \widetilde{V}_R$.
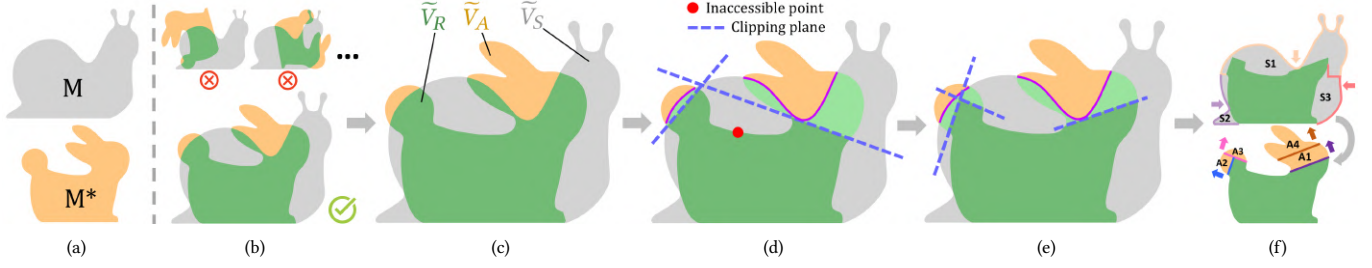
Fig. 6. Overview of our ASHRM process planning framework. Starting with the initial model *Snail* and the target model *Rabbit* (a), we first rotate and translate *Rabbit* to determine a setup that favors achieving the maximum fabricable reutilized volume (b). The three types of volumes are then computed through Boolean operations (c). The framework employs a two-stage strategy. In the first stage (d), a set of clipping planes is initialized to partially clip the ideal reutilized volume into SM-then-AM volume V_T (light green), ensuring that the remaining reutilized volume V_R (dark green) satisfies the plano-convex constraint. In the second stage (e), the clipping planes are refined to maximize the reutilized volume V_R under both accessibility and plano-convex constraints. Finally, the new subtractive volume V_S and additive volume V_A are decomposed into positional fabricable blocks for manufacturing (f).

adheres to the plano-convex constraint (Figure 6 (d), subsection 4.1).

- *Reutilized volume maximization.* The clipping planes are further refined using Monte Carlo Tree Search (MCTS), which can quickly explore the vast search space in combinatorial optimization and uncover distribution patterns, to maximize the reutilized volume (Figure 6 (e), subsection 4.2).

Pre-processing. Before the core algorithm, to enhance the probability of attaining a larger fabricable reutilized volume, the setup of M^* relative to M is optimized through a coarse-to-fine strategy, as illustrated in Figure 6 (b) and elaborated in section 5.

Post-processing. After determining the final reutilized volume through the core algorithm, we introduce a postprocessing step to generate 3+2 axis positional machining and printing blocks, as well as tool paths for subtractive and additive manufacturing, as depicted in Figure 6 (f) and detailed in section 6.

4 REUTILIZED VOLUME MAXIMIZATION

This section presents the proposed two-stage algorithm for maximizing the reutilized volume, illustrated with 2D examples for clarity. For reference, Appendix A summarizes the key terms, and Appendix E provides the algorithm's pseudocode.

4.1 Reutilized Volume Initialization

In this stage, we consider only the plano-convex constraint and aim to maximize the reutilized volume for an optimal initial feasible reutilized volume. The key challenge here is to determine a set of clipping planes that: 1) adhere to the plano-convex constraint and 2) yield the largest reutilized volume V_R. Our approach involves initially creating candidate clipping planes and subsequently choosing the best combination as $\mathbb{C}$, resulting in a reutilized volume $V_R^{\mathbb{C}}$.

Generation of candidate clipping planes. This step builds on the initial printing base $\widetilde{\mathcal{B}}$. For each connected component $b_i \in \widetilde{\mathcal{B}}$, we generate N candidate clipping planes. Each plane $\widetilde{c}_{i,j}$ is defined by a tuple $\langle n_{i,j}, p_{i,j} \rangle$ (Figure 7 (a, b)), where $n_{i,j}$ is a normal vector uniformly sampled from the upper hemisphere, and $p_{i,j}$ is the point
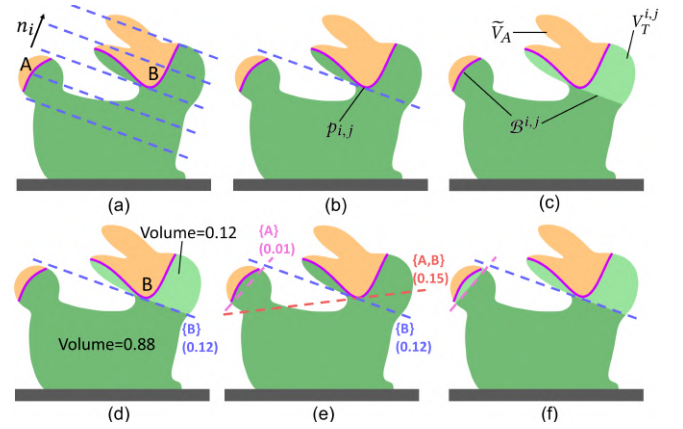

Fig. 7. Demonstration of reutilized volume optimization under the plano-convex constraint. The initial printing base includes two connected components "A" and "B" (a). For each sampled direction n_i for "B", a point $p_{i,j}$ is identified to determine a candidate clipping plane (b). Clipping the ideal reutilized volume with the blue candidate clipping plane yields a new printing base $\mathcal{B}^{i,j}$ (c). The blue candidate clipping plane is associated with a weighted set, where each element in the set represents a connected component of the initial printing bases. The weight is defined as the volume ratio of $V_T^{i,j}$ to $\widetilde{V}_R$ (d). Each candidate clipping plane is associated with a weighted set. Finally, two clipping planes (pink and blue) are selected from the candidates by solving the weighted set cover problem (f).

on b_i with the smallest projection distance to the fabrication base (dark gray region in Figure 7) along $n_{i,j}$.

Each $\widetilde{c}_{i,j}$ produces a reutilized volume $V_R^{i,j}$ and an additive volume $V_A^{i,j}$, with the new initial printing base $\mathcal{B}^{i,j}$ formed between them (see Figure 7 (c)), thereby replacing the unfeasible initial printing base b_i. However, $\mathcal{B}^{i,j}$ may still violate the plano-convex constraint due to other infeasible connected components (purple line in Figure 7 (c)). Therefore, appropriate combinations of these candidates must be selected to satisfy the plano-convex constraint.

Selection of optimal clipping planes. In addition to satisfying the plano-convex constraint, selecting an optimal combination of clipping planes can further maximize the reutilized volume. While clipping planes are generated for each infeasible connected component $b_i \in \widetilde{\mathcal{B}}$, some clipping planes may resolve multiple connected components simultaneously. For instance, the red dashed clipping plane in Figure 7 (e) resolves both "A" and "B", thus satisfying the plano-convex constraint.

To resolve this, we present it as a *weighted set cover problem (WSCP)* [Golab et al. 2015], which originally aims to find the least-weight subset that fully covers the universal set. In our case, we select a subset of clipping planes from the candidate clipping planes to maximize the reutilized volume, while maintaining the plano-convex constraint.

Specifically, we first assign a set $s_{i,j}$ to each $\widetilde{c}_{i,j}$, as shown in Figure 7 (d). If b_k and the fabrication base are on opposite sides of $\widetilde{c}_{i,j}$, we add b_k to the set $s_{i,j}$. Next, we assign a weight $W_{i,j} = 1 - Volume(V_R^{i,j}) / Volume(\widetilde{V}_R)$ to each $\widetilde{c}_{i,j}$. Then, we solve the WSCP by integer linear programming to calculate the optimal solution:

$$
\begin{aligned}
\min \quad & \sum_{i=1}^{N} \sum_{j=1}^{M} W_{i,j} \cdot Z_{i,j} \\
\text{s.t.} \quad \langle 1 \rangle \quad & Z_{i,j} \in \{0,1\}, \quad \forall i = 1,2,\ldots,N, \quad \forall j = 1,2,\ldots,M, \quad (2) \\
\langle 2 \rangle \quad & \sum_{i',j' \in Idx(k)} Z_{i',j'} \geq 1, \quad \forall k = 1,2,\ldots,M,
\end{aligned}
$$

where $Idx(k) = \{i',j' \mid k \in s_{i',j'}, 1 \leq i' \leq N, 1 \leq j' \leq M\}$, which represents the index of a set that contains a subset of element k. $Z_{i,j}$ is a binary integer variable for $\widetilde{c}_{i,j}$. In constraint $\langle 1 \rangle$, "1" indicates selecting $\widetilde{c}_{i,j}$, while "0" indicates not selecting it. For constraint $\langle 2 \rangle$, each initial printing base b_k must be covered by at least one selected clipping plane. We calculate the optimal solution through the GLPK solver [Makhorin 2012], thereby selecting the clipping planes from the candidates.

4.2 Reutilized Volume Maximization

The previous section introduces $V_R^{\mathbb{C}}$, which satisfies the plano-convex constraint. Based on this, we further define:

$$
V_T^{\mathbb{C}} = \widetilde{V}_R - V_R^{\mathbb{C}}, \quad V_A^{\mathbb{C}} = \widetilde{V}_A + V_T^{\mathbb{C}}, \quad V_S^{\mathbb{C}} = \widetilde{V}_S + V_T^{\mathbb{C}}. \quad (3)
$$

Here, $V_T^{\mathbb{C}}$ denotes the SM-then-AM volume, while $V_A^{\mathbb{C}}$ and $V_S^{\mathbb{C}}$ represent the updated additive and subtractive volumes, respectively.

This section incorporates the accessibility constraint and optimizes the clipping planes $\mathbb{C}$ to maximize the reutilized volume $V_R^{\mathbb{C}}$. The main challenges comprise:

- How to determine if the AM process to realize $V_A^{\mathbb{C}}$ and the SM process to carve $V_S^{\mathbb{C}}$ meet the accessibility constraint?
- How to transform $V_R^{\mathbb{C}}$ into a feasible reutilized volume while complying with the accessibility constraint?
- How to maximize $V_R^{\mathbb{C}}$ by optimizing $\mathbb{C}$ while ensuring all collision-free constraints?

Graph-based accessibility analysis. To resolve the first challenge, we utilize a methodology akin to [Zhong et al. 2023b], implementing accessibility analysis with a collision dependency graph G_{cd}

to verify if $V_A^{\mathbb{C}}$ and $V_S^{\mathbb{C}}$ meet the accessibility constraint derived directly from G_{cd}, which simultaneously represents the SM and AM stage accessibility constraints, constructed in preprocessing via a geometric collision detection method.

The collision dependency graph streamlines accessibility constraint verification, completely avoiding geometric collision detection while optimizing $V_R^{\mathbb{C}}$. For given $V_T^{\mathbb{C}}, V_A^{\mathbb{C}}$, and $V_S^{\mathbb{C}}$, first update the initial G_{cd} by removing all nodes within $V_T^{\mathbb{C}}$ (brown nodes in the inset) along with their edges. Then, verify accessibility by checking that no non-accessible nodes remain in the resulting graph. Appendix B provides details for reconstructing the collision dependency graph. The main difference from [Zhong et al. 2023b] involves sampling points from three volume types ($V_T^{\mathbb{C}}, V_A^{\mathbb{C}}$, and $V_S^{\mathbb{C}}$).

Feasible reutilized volume transformation. To transform $V_R^{\mathbb{C}}$ into a feasible reutilized volume under the accessibility constraint, an iterative method is applied. Each clipping plane $c_i \in \mathbb{C}$ is shifted along its normal n_i using an adaptive step, set to 30% of the shortest projection distance along n_i to the fabrication base (dark gray area). After each iteration, the accessibility constraint is evaluated via the collision dependency graph (G_{cd}). All clipping planes in $\mathbb{C}$ are translated iteratively in sync until a feasible reutilized volume is achieved with the final positions of the clipping planes.

Reutilized volume maximization. Given the reutilized volume of $V_R^{\mathbb{C}}$ and the associated clipping planes $\mathbb{C}$ from the previous step, this step seeks to maximize $V_R^{\mathbb{C}}$ by refining $\mathbb{C}$ and maintaining all collision-free constraints.

As discussed in subsection 3.1, each clipping plane is an infinite 3D plane that partitions $\widetilde{V}_R$ into two regions. Such global clipping method limits the potential to maximize reutilized volume by refining these clipping planes. To overcome this, we convert each global clipping plane $\forall c_i \in \mathbb{C}$ into a set of local clipping planes $\mathbb{C}^L = c_1^L, c_2^L, \ldots$, each of which clips only a local region of $\widetilde{V}_R$ and is represented as a quadrilateral on the 3D plane. The inset figure shows the blue plane split into two local planes, in red and yellow.

As shown in the inset, locally refining the clipping planes of $\mathbb{C}^L$ can effectively enlarge the resulting reutilized volume. However, each local clipping plane varies considerably in pose, and since refinement must satisfy all collision-free constraints, optimally configuring them to maximize reutilized volume is non-trivial.

To address this problem, we apply a Monte Carlo Tree Search (MCTS) which excels at handling large search spaces. Specifically, we determine the pose (including normal vector and original point)
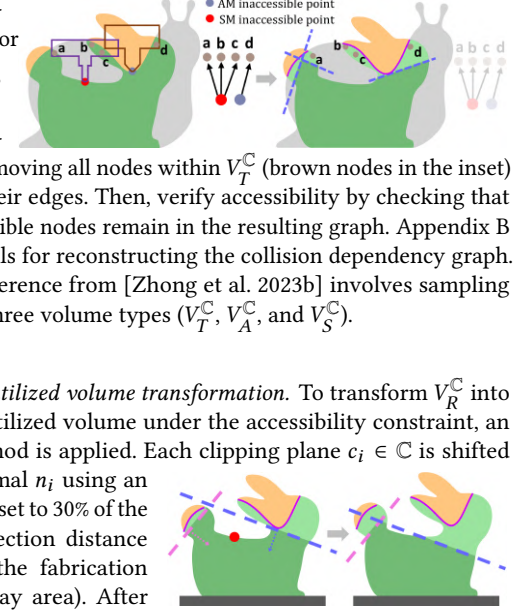
of each c_i^L through four standard iterative stages in the MCTS process: selection, expansion, simulation, and back-propagation. We briefly introduce the structure of the Monte Carlo Tree: Each layer represents a local clipping plane c_i^L, with the number of layers equal to $|\mathbb{C}^L|$. Each node contains a state and two values (left and right). The state represents the normal and original point of c_i^L, while the left and right values correspond to the reward and the number of selections, respectively.

During the MCTS process, we still use the collision dependency graph (G_{cd}) to assess the accessibility constraint of the generated reutilized volume during local refinement of $\mathbb{C}^L$. The reutilized volume obtained through clipping with $\mathbb{C}^L$ is denoted by $V_R^{\mathbb{C}^L}$. The inset figure presents three reutilized volumes $V_R^{\mathbb{C}^L}$ obtained from different MCTS iterations. When the MCTS iterations reach the threshold, the search is terminated, and the $\mathbb{C}^L$ that produce the largest $V_R^{\mathbb{C}^L}$ which satisfies the accessibility and plano-convex constraints are selected as the final solution. Appendix C provides more details for the MCTS process.

5 SETUP DETERMINATION

The above algorithm relies on the setup determination of M and M^*. In this section, we describe the setup algorithm used for pre-processing.

A model's setup denotes its orientation and position, adjustable by rotation and translation. The relative setup between M and M^* directly affects the subsequent maximization of the reutilized volume, where an appropriate setup can yield a larger volume. Unfortunately, simply maximizing the volume of $\widetilde{V}_R$ does not guarantee an optimal setup, as collision-free constraints may lead to a smaller $V_R^{\mathbb{C}^L}$ even when $\widetilde{V}_R$ is large. For example, in Figure 8, the $\widetilde{V}_R$ in (a) is larger than that in (b), but the calculated $V_R^{\mathbb{C}^L}$ is smaller.

In our setup strategy, we decouple rotation and translation and use a coarse-to-fine evaluation method to determine the setup of M^*, while keeping the setup of M fixed, aiming to identify the setup of M^* that is more likely to maximize the final calculated $V_R^{\mathbb{C}^L}$. In the coarse stage, each rotation of M^* is evaluated using the first estimating function, which combines the volume of the initial reutilized region $\widetilde{V}_R$ with a term encouraging M^* to adopt an upright orientation rather than lying flat, as this tends to place the printing bases at a higher position. The five highest-scoring rotations are then selected. In the fine-tuning stage, the translation of M^* is adjusted for these five fixed rotations, with each translation evaluated using the second estimating function. This function again considers $\widetilde{V}_R$ but further incorporates geometric cues, including the relative heights of the printing bases and the height of $\widetilde{V}_R$, to favor setups that leave more feasible space for clipping planes and thus lead to larger $V_R^{\mathbb{C}^L}$. Finally, the setup of M^* with the highest score is selected. The detailed formulations of the two estimating functions and additional technical considerations are provided in Appendix D.
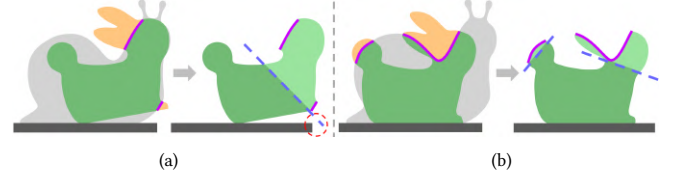

Fig. 8. Demonstration of how different setups affect the computed $V_R^{\mathbb{C}^L}$. Dark gray indicates the fabrication base. (a) The setup of M^* maximizes the volume of $\widetilde{V}_R$. However, to satisfy the plano-convex constraint, a large portion of $\widetilde{V}_R$ is clipped into the SM-then-AM volume, resulting in a smaller $V_R^{\mathbb{C}^L}$. (b) This setup yields a larger $V_R^{\mathbb{C}^L}$ compared to (a).
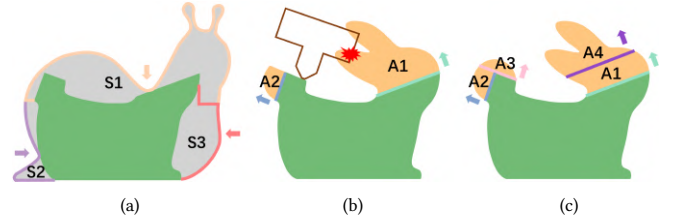

Fig. 9. Illustration of post-processing. The numbers indicate the order. (a): Decompose V_S into three subtractive accessible blocks along different SM directions. (b): Decompose V_A into self-supporting positional blocks using local clipping planes; collision occurs between the nozzle and printed block (A1). (c): Further decompose A1 using the method from [Zhong et al. 2023a] to satisfy collision-free constraints.

6 MANUFACTURABLE BLOCKS DECOMPOSITION

After determining $V_R^{\mathbb{C}^L}$, the new subtractive volume is $V_S = \widetilde{V}_S + V_T$, and the new additive volume is $V_A = \widetilde{V}_A + V_T$. This section outlines the post-processing steps following the core algorithm (section 4). The goal is to decompose V_S and V_A into *positional manufacturable blocks*, where each one can be processed via SM/AM according to one direction while satisfying collision-free constraints.

Subtractive blocks decomposition. The only manufacturing constraint in SM is the accessibility constraint. At this stage, since the entire surface of V_S satisfies the global collision-free constraint, i.e., is cutter-accessible, V_S can be fully decomposed with accessible SM directions. Firstly, we use UG-NX software [SIEMENS 2016] to generate positional rough machining blocks along the four directions of the (+/-) X/Y axes, and then generate positional finish machining cutter paths based on these four directions. These orthogonal directions efficiently cover most of the volumes of V_S. For the remaining inaccessible volumes, we calculate additional accessible directions and further decompose them using UG-NX. Figure 9 (a) shows the subtractive positional block decomposition results for a 2D case.

Additive blocks decomposition. Each local clipping plane $c_i^L \in \mathbb{C}^L$ is applied sequentially to clip $V_R^{\mathbb{C}^L}$, producing additive blocks. To avoid material waste from support structures, each c_i^L is treated as a printing base to further decompose the blocks into self-supporting ones, following [Wu et al. 2020]. However, since we did not consider collisions caused by $V_R^{\mathbb{C}^L}$ itself, collisions may still occur between

blocks at this stage (Figure 9 (b)). To address this issue, the method of [Zhong et al. 2023a] is adopted to further decompose these blocks into collision-free ones. Specifically, we slice each block along its printing direction, perform collision detection with other slice layers of each block, and build a directed graph to record the collision dependency relationships. Then, we perform topological sorting on the nodes to further decompose each block, ensuring they are collision-free, as shown in Figure 9 (c).

Toolpath generation. Finally, we reverse the sequences of these subtractive/additive blocks and input them and their directions to the UG-NX software to generate SM/AM toolpaths.

7 RESULTS AND DISCUSSIONS

This section presents the results of applying our ASHRM process planning to 3D models with varying geometric complexities, along with the corresponding simulative and physical evaluations. We assess the algorithm's efficiency across various parameter settings, conduct ablation studies, compare it with other strategies, and discuss failure cases, manufacturing time optimization, and scalability for complex models.

7.1 Implementation and parameters

The algorithm was implemented in C++, using CGAL [Fabri and Pion 2009], Libigl [Jacobson et al. 2018], and Eigen [Guennebaud et al. 2010] for geometric processing. It was executed on a PC with an Intel Core i7-11700F CPU and 16GB of memory. The source code is publicly accessible at https://github.com/fanchao98/Waste-to-Value.

Additional implementation parameters include a ball-end cutter with a 13 mm shank length and 3 mm diameter, and a nozzle with a 10 mm length and 9 mm diameter, following [Zhong et al. 2023b]. Collision detection used 100 sampled directions for both AM and SM. In subsection 4.1, 100 uniformly sampled normal directions were evaluated, while subsection 4.2 employed an MCTS with 500 iterations. The self-supporting threshold for AM positional blocks was set to 50° in section 6.

7.2 Computational results

Examples and statistics. Figure 1 and Figure 10 present the computational outcomes of our algorithm applied to 12 different 3D models, grouped into four categories of remanufacturing applications: **1) Model Repairing** (Figure 1): Repairing a damaged workpiece, updating it to a new or improved one. **2) Model Scaling** (first two models in Figure 10): Scaling down a large model into a smaller one. **3) Remanufacturing of Spare Parts** (third to sixth models in Figure 10): Remanufacturing spare parts with simple geometries into target workpieces in resource-limited environments. **4) Type Transformation** (seventh to last models in Figure 10): Transforming an existing 3D model into a significantly different shape, such as a duck evolving into a swan or a frog turning into a prince. Figure 11 demonstrates the subtractive and additive positional manufacturing processes, which satisfies all the manufacturing constraints.

Table 1 presents the statistics for all 12 models. In general, the final reutilized volume (RV) largely depends on the accessibility of the cutter/nozzle and the configuration of the initial printing base. When $\widetilde{V}_R$ is accessible by both the cutter and nozzle, and
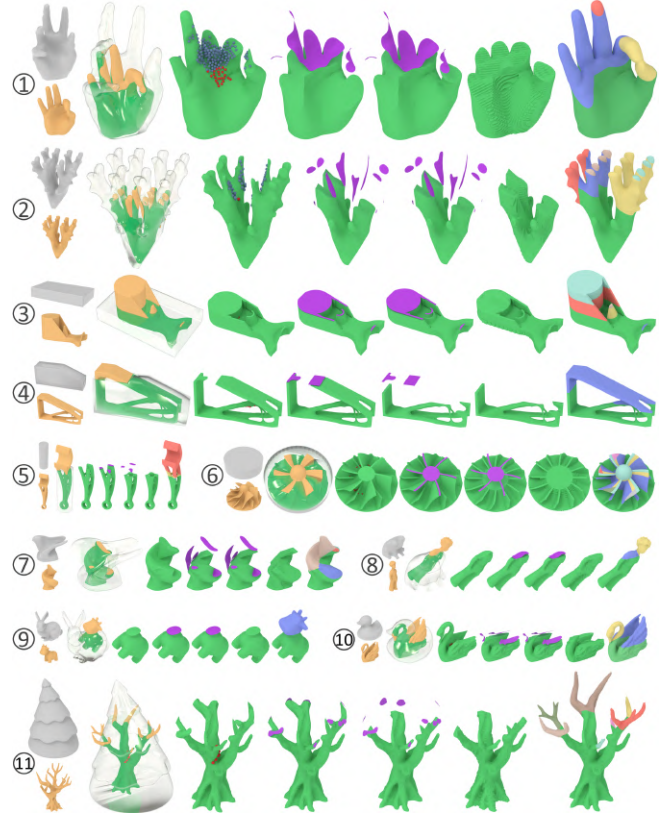

Fig. 10. Computational results gallery of our algorithm. 11 pairs of ASHRM models are arranged in the order of ① *Hand-Hand*, ② *Coral-Coral*, ③ *Cuboid-Bracket*, ④ *Trapezium-Holder*, ⑤ *Cylinder-Opener*, ⑥ *Cylinder-Impeller*, ⑦ *Sculpture-Sculpture*, ⑧ *Frog-Prince*, ⑨ *Bunny-Cow*, ⑩ *Duck-Swan*, ⑪ *Tree-Tree*. The seven columns of each pair respectively demonstrate: 1) the initial model M and the target model M^*; 2) three types of volumes $\widetilde{V}_S$ (gray), $\widetilde{V}_A$ (orange), $\widetilde{V}_R$ (green) with the determined setup of M^*; 3) the SM and AM inaccessible points (red and blue) in $\widetilde{V}_R$; 4) the $V_R^{\mathbb{C}}$ (green) obtained by subsection 4.1, compared to the initial printing bases (purple); 5) the refined $V_R^{\mathbb{C}L}$ obtained by subsection 4.2, compared to the initial printing bases (purple); 6) the simulated positional rough machining results; 7) the additive positional blocks decomposition results.
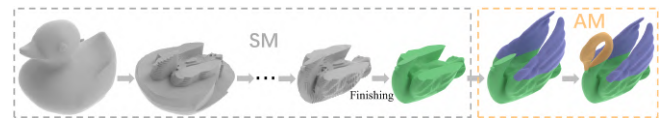
Fig. 11. Illustration of the positional SM and AM processes. In the SM process, excess material is removed from the initial *Duck* model through rough machining, followed by finishing steps to produce $V_R^{\mathbb{C}L}$, with each machining step constrained to a fixed direction. The AM process then adds material along two additive directions on $V_R^{\mathbb{C}L}$ to complete the fabrication of the *Swan* model.

the initial printing base are flat and positioned at the top of $\widetilde{V}_R$, the final reutilized volume closely approximates $\widetilde{V}_R$, as seen in the

Table 1. Statistics of the models shown in the paper. *Size* represents the size of M^* (length × width × height, mm) after its setup is determined. B is the ratio of the minimum Z-value of the additive bases to the height of M after determining the setup of M^*. $\#P_S$ and $\#P_A$ indicates the number of SM and AM inaccessible points on $\widetilde{V}_R$, respectively. R_V is the volume ratio of $\widetilde{V}_R$ to M^*. R_I is the volume ratio of $V_R^{\mathbb{C}}$ to M^* after reutilized volume initialization. $\#C_b$ and $\#C_a$ are the number of global and local clipping planes, respectively. R_F is the final volume ratio of $V_R^{\mathbb{C}^L}$ to M^*. $\#B_S$ and $\#B_A$ are the number of subtractive and additive positional blocks.

Model	Size	B	$\#P_S$	$\#P_A$	R_V	R_I	$\#C_b$	$\#C_a$	R_F	$\#B_S$	$\#B_A$
						Sec. 4.1		Sec. 4.2		Sec. 6	
Bearing	51×40×34	0.51	0	5	0.98	0.89	2	2	**0.90**	7	2
Hand	56×33×77	0.31	23	167	0.91	0.77	1	2	**0.80**	5	3
Coral	44×39×53	0.41	2	57	0.92	0.63	2	6	**0.62**	8	5
Bracket	104×108×57	1.00	5	0	0.55	0.53	1	3	**0.51**	5	4
Holder	46×109×58	1.00	8	0	0.77	0.75	1	2	**0.37**	5	1
Opener	48×46×191	0.94	7	0	0.54	0.55	3	3	**0.52**	6	3
Impeller	74×74×35	1.00	48	0	0.92	0.91	1	1	**0.74**	9	7
Sculpture	29×28×46	0.26	0	3	0.98	0.66	4	5	**0.69**	4	5
Prince	10×10×33	0.89	0	0	0.78	0.75	2	2	**0.75**	5	2
Cow	49×34×46	0.57	0	0	0.74	0.73	1	1	**0.73**	4	1
Swan	49×41×53	0.43	21	3	0.87	0.73	1	2	**0.75**	6	2
Tree	42×41×55	0.49	7	0	0.93	0.93	7	12	**0.85**	7	8

Table 2. The program running time (second) of the algorithm steps described in each section. T is the total running time (minute).

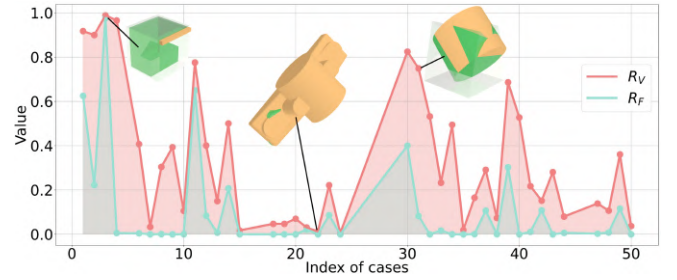| Model	Sec. 4.1	Sec. 4.2	Sec. 5	Sec. 6	T
Bearing	13	203	166	2	6.4
Hand	21	565	215	5	13.4
Coral	68	169	114	4	5.9
Bracket	8	457	111	4	9.7
Holder	10	238	92	5	5.8
Opener	27	439	145	81	11.5
Impeller	8	547	197	37	13.2
Sculpture	37	258	195	6	8.3
Prince	9	9	205	2	3.8
Cow	2	6	63	1	1.2
Swan	21	420	178	8	10.5
Tree	51	106	157	8	5.4

Fig. 12. Demonstration of large-scale random testing. In all cases, the initial and target models were randomly paired. The red and blue curves represent R_V and R_F, respectively, as defined in Table 1.
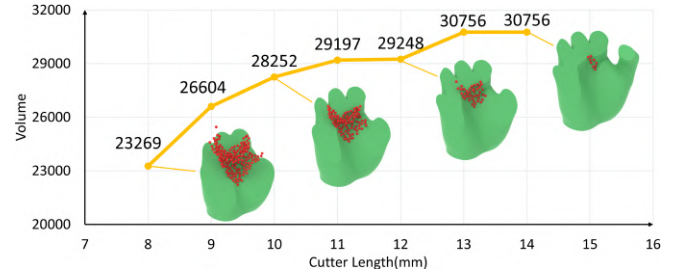

Fig. 13. Illustration of the cutter length (horizontal-axis) effect for the volume of $V_R^{\mathbb{C}^L}$ (vertical-axis). The red points indicate the SM inaccessible points in $\widetilde{V}_R$, and the green models indicate the final determined $V_R^{\mathbb{C}^L}$.

Bunny-Cow case ($R_F = 0.73, R_V = 0.74$). Conversely, when $\widetilde{V}_R$ has a complex geometry with initial printing base located on its sides, additional volumes must undergo SM followed by AM to satisfy collision-free constraints. For example, in the *Duck-Swan* case, the swan's head requires SM-then-AM to meet accessibility constraint. On the other hand, our general framework is capable of handling a wide range of models, from simple ones like the *Frog-Prince* to geometrically complex ones such as the *Tree-Tree*, which features numerous branches. The use of local clipping planes allows for finer clipping, thereby increasing material reutilization.

Computing time. Our algorithm requires an average of 8 minutes to process the 12 test models. Table 2 details the time consumption for each step of the algorithm. The MCTS process (Sec. 4.2) is identified as the primary bottleneck. While redundant collision detection is reduced by updating and referencing the collision dependency graph G_{cd}, significant time is still needed to update connected components, perform localized mesh clipping, and adjust G_{cd} during MCTS iterations.

Large-scale random testing. To further evaluate the robustness of our algorithm, we randomly paired manifold mesh models from Thingi10K [Zhou and Jacobson 2016] and a classical graphics model repository [Jacobson 2020] as initial and target models. Figure 12 presents 50 successful cases.

When the initial and target models are similar in geometry and size, it is easier to retain a high reutilized volume (left example). However, randomly paired models often exhibit substantial differences in geometry and size (middle example), making it difficult to identify a setup that yields a large R_V. Moreover, collision-free constraints can further reduce the reutilized volume, especially when

geometric differences are pronounced (right example), leading to a significant drop from R_V to R_F.

Cutter size effect. The tool size directly influences the accessibility constraint, which subsequently determines the size of reutilized volume. As shown in Figure 13, we tested our algorithm's ability to generate the expected $V_R^{\mathbb{C}^L}$ by varying the cutter size. The results indicate that a longer cutter allows for a larger $V_R^{\mathbb{C}^L}$ due to easier accessibility. This experiment demonstrates that our algorithm effectively adapts to different cutter sizes to produce the desired $V_R^{\mathbb{C}^L}$.
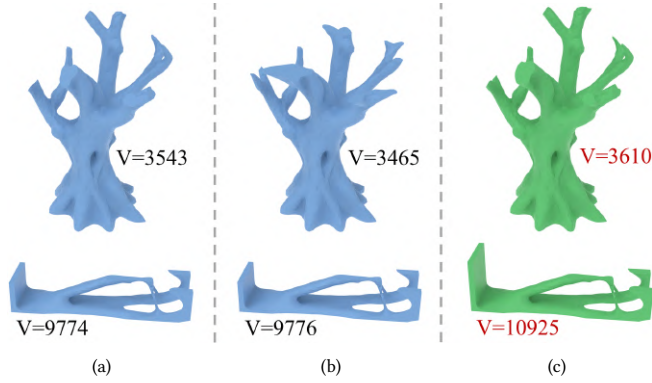
Fig. 14. Comparison with other reutilized volume maximization strategies. "V" indicates the volume. Blue indicates the results of other strategies, while green represents ours. Top: Tree-Tree case; Bottom: Cuboid-Bracket case. (a) Top-down strategy: after determining the clipping planes by solving Equation 2, all planes are sequentially offset until the reutilized volume satisfies both plano-convex and accessibility constraints. (b) Bottom-up strategy: starting from an empty volume, a single clipping plane is iteratively applied to optimize the reutilized volume while ensuring both collision-free constraints are met at each iteration. (c) Our two-stage strategy achieves the largest reutilized volume that satisfies all constraints in both cases.
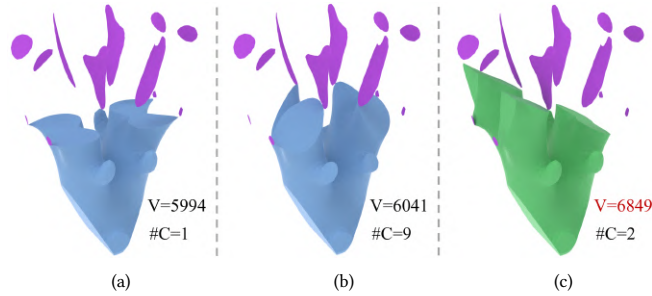

Fig. 15. Comparison of WSCP solving methods. "#C" is the number of clipping planes. The figure illustrates three approaches for solving the weighted set cover problem, each producing a corresponding reutilized volume $V_R^{\mathbb{C}}$. (a) Select a single clipping plane that covers all printing bases with the minimum weight. (b) Use a greedy iterative approach to select the lowest-weight clipping plane for each component's printing base, repeating until full coverage. (c) Apply our integer linear programming strategy to compute the optimal solution.

7.3 Comparisons and Ablation

Comparison of reutilized volume maximization strategies. To demonstrate the efficiency of our two-stage approach, we conduct a comparison experiment with other strategies, as shown in Figure 14. Given the same runtime, the top-down strategy (a) employs global clipping planes, leading to an inefficient search. Conversely, the bottom-up strategy (b) uses a single clipping plane to optimize reutilized volume, but its vast search space lowers efficiency.

Comparison of WSCP solving methods. We compare three methods for solving the weighted set cover problem in subsection 4.1, as shown in Figure 15. When using only one clipping plane (a), a large
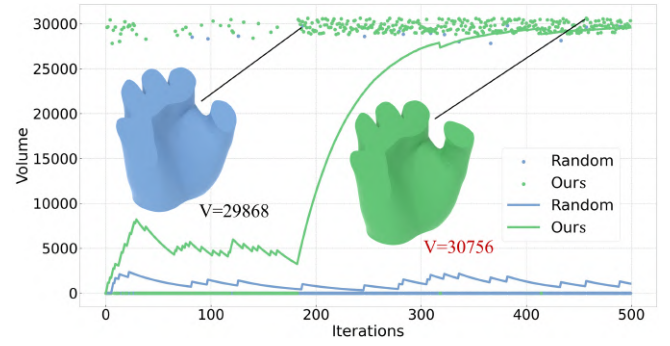
Fig. 16. Comparison of MCTS strategies. Blue and green indicate the results of random strategy and our MCTS approach, respectively. Each point corresponds to a specific set of posed local clipping planes and their resulting $V_R^{\mathbb{C}^L}$. If $V_R^{\mathbb{C}^L}$ violates collision-free constraints, its volume is set to 0 in the figure. The curves are obtained using an exponential smoothing function $V_t = 0.02 * X_t + 0.98 * V_{t-1}$, where V_t denotes the smoothed value at iteration t and X_t is the recorded value at iteration t. These curves provide a rough indication of the search efficiency, noting that the 'horizontal line' with volume = 0 is in fact composed of a stack of points.

portion of $\widetilde{V}_R$ is translated to SM-then-AM volume to meet plano-convex constraint. In the greedy method (b), although it uses plenty of clipping planes, it fails to find the optimal solution, leaving a much smaller reutilized volume compared to our method. In contrast, our approach preserves the largest reutilized volume.

Comparison of MCTS strategies. In subsection 4.2, we perform MCTS to determine the poses of the local clipping planes. The efficiency of the search is enhanced by the UCB function (Equation 5), which guides the selection process within MCTS. We conducted a comparison experiment to verify the effectiveness of our UCB function. As shown by the blue line and points in Figure 16, the random search strategy is inefficient, with most searched poses of clipping planes failing to produce $V_R^{\mathbb{C}^L}$ satisfying the collision-free constraints. In contrast, our strategy (green in Figure 16) identifies valid poses more effectively by prioritizing those that generate a larger $V_R^{\mathbb{C}^L}$ while satisfying collision-free constraints.

Ablation experiment of setup strategy. Our algorithm determines the setup of M^* by a rough estimate (section 5). Figure 17 shows an ablation experiment of the estimate strategy, where we separately disregarded the consideration of two terms in both Equation 7 and Equation 8. In the volume-only strategy (a), although $\widetilde{V}_R$ achieves the maximum volume, a large portion of $\widetilde{V}_R$ translate to SM-then-AM volume to satisfy the plano-convex constraint. In contrast, the plano-convex-only strategy (c) results in a smaller $\widetilde{V}_R$ and a correspondingly smaller $V_R^{\mathbb{C}}$. Our method balances these two approaches, yielding a $\widetilde{V}_R$ with considerable volume while maximizing $V_R^{\mathbb{C}}$.

Comparison of setup strategies. To further demonstrate the advantage of our setup strategy, we compare it with the denser-sampling joint optimization method on the first six model pairs from the gallery, as shown in Figure 18. While the joint optimization slightly outperforms ours on the *Coral-Coral case*, it fails on the *Hand-Hand*
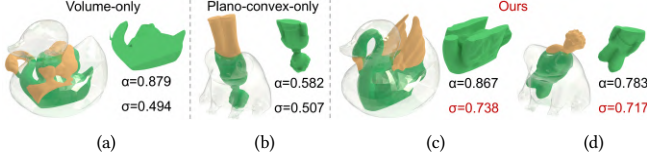
Fig. 17. Ablation experiment of setup strategy. α and σ respectively indicate the volume ratio of $\widetilde{V}_R$ and $V_R^{\mathbb{C}}$ to M^*. (a) Volume-only strategy: set $w_1 = 1$, $w_2 = 0$ in Equation 7 and Equation 8. (b) Plano-convex-only strategy: set $w_1 = 0$, $w_2 = 1$. (c) and (d) show our trade-off strategy with $w_1 = 0.4$ and $w_2 = 0.6$, resulting in a larger $V_R^{\mathbb{C}}$ compared to the previous strategies.
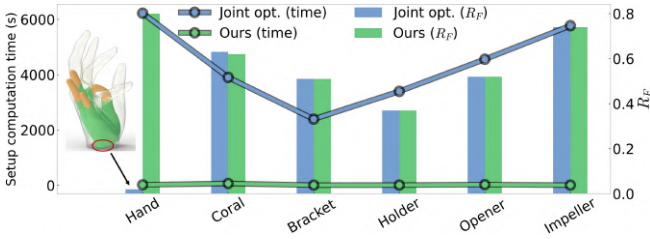

Fig. 18. Comparison of setup methods. We adopt a coarse-to-fine strategy (green) to decouple rotation and translation. The joint optimization method (blue) simultaneously optimizes rotation and translation, using denser sampling (double density) with Equation 8 as the evaluation function.

case—without considering the 'stand up' heuristic in Equation 7, it selects a tilted setup that creates a narrow subtractive inaccessible region (red circle), resulting in zero reutilized volume. Moreover, our setup strategy is far more efficient in computation time.

7.4 Simulative and Physical experiments

ASHM machine setting. Limited by the availability of existing 5-axis ASHM machines—particularly the highly expensive metal-based machines—we conducted physical experimental verification using a 5-axis ASHM machine modified from a typical desktop 5-axis CNC machine (YORNEW Benchtop 5-axis CNC MX220), as instructed in [Zhong et al. 2023b]. Due to the modified machine's inability to achieve high-temperature sintering for metal AM, we were unable to use metal materials for verification. However, this limitation does not affect the validation of the SM/AM toolpaths generated by our algorithm, as they still satisfy the manufacturing constraints when tested with PLA materials. The inset figure shows the modified ASHM machine, which features a machining space of 220mm × 120mm × 200mm.

Further details about the machine can be found in [Zhong et al. 2023b]. Additionally, we incorporated a retraction function into the additive module. Details of the modifications are provided in Appendix F.

Fig. 19. Simulations in Unigraphics-NX. In the SM stage, NX generates collision-free rough machining toolpaths based on the given direction (a), the simulated result shows noticeable surface deviations from the target model (b), and further finish machining is applied to smooth the surface and achieve high precision (c). In the AM stage, collision-free AM toolpaths are generated (d), resulting in the complete simulated target model (e).

Simulative validation. As mentioned in section 6, we generate SM (rough and finish machining) and AM toolpaths by the software UG-NX. For the input AM blocks, we perform the contour-parallel path, set the path width to 0.5 mm and the layer thickness to 0.2 mm, and take 30% filling rate. For the SM blocks, we first take the deep cavity milling strategy to rough machining with 0.2 mm residual height. Then we perform area milling to finish the workpiece.

Figure 19 presents the simulation results of two examples (*Bearing* and *Sculpture*) in NX. With automatic collision detection enabled, the toolpaths consistently satisfy the collision-free constraints, validating the effectiveness of our algorithm. All examples in Figure 10 passed NX's SM and AM simulation tests, with the sixth column displaying the rough machining results from the NX simulation.

Manufacturing validation. We used the toolpaths generated during the simulation stage for manufacturing. Figure 20 shows two of our manufacturing results. Compared with pure AM, our method significantly reduces material usage (by 90% and 69%). The accompanying video demonstrates the entire manufacturing process for the two models.

However, as the first work to perform physical verification for general models, our manufacturing results still exhibit several noticeable defects (see the zoomed-in figure in Figure 20): 1) Apparent seam lines between blocks (red frame): Printing discontinuities result in visible seam lines between additive manufacturing positional blocks. Modifications to the machine have also caused misalignment of the rotary axis, exacerbating this defect. 2) Poor surface quality of AM blocks (blue frame): Surface quality issues arise from the precision limitations of additive manufacturing and the staircase effect caused by layer-by-layer printing. Additionally, unstable extrusion from the additive modules of our equipment further worsens this defect. We added an SM process after AM to finish the surface, where the machining directions are generated using the method described in section 6, further improving the manufacturing quality of the remanufactured part (the fourth column of Figure 20).

Comparison with various manufacturing strategies. As shown in Figure 21, the pure AM strategy (a) wastes 9 times more material compared to our ASHRM approach. On the contrast, the pure
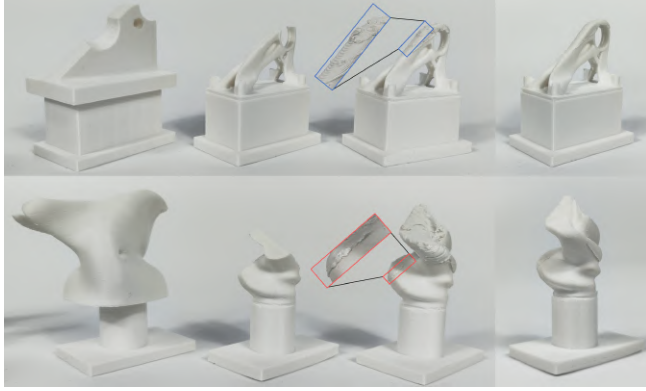
Fig. 20. The fabrication results of repairing the broken *Bearing* into a new topology-optimized one (top row), and transforming the existing *Sculpture* to another radically different shape (bottom row). Each row contains four columns representing: 1) the initial model; 2) the reutilized volume $V_R^{\mathbb{C}^L}$ after SM; 3) the target model after AM; 4) further perform finishing on the surface of the additive volume through SM, yielding a model with high surface quality. Imperfect equipment and the staircase of AM result in defects at the block interfaces and model surface (red and blue frames).
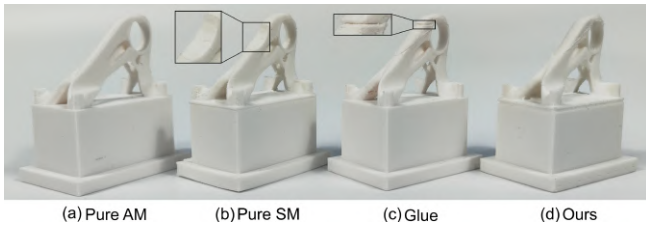

(a) Pure AM   (b) Pure SM   (c) Glue   (d) Ours

Fig. 21. Comparison of various manufacturing strategies: (a) Pure AM: fabricating the target model M^* solely through additive manufacturing. (b) Pure SM: removing excess material from the initial model M exclusively using subtractive manufacturing. (c) Glue: individually printing each connected component of the additive volume $V_R^{\mathbb{C}^L}$, then assembling them by gluing to $V_R^{\mathbb{C}^L}$. (d) Our ASHRM strategy.

SM strategy (b) avoids additional material consumption but leads to incomplete regions in the model. Another potential approach, separately printing AM components and gluing them together (c), introduces manual processes, resulting in 1) challenges in ensuring accurate positioning and 2) increased time and labor demands.

7.5 Discussion and limitations

Shapes with highly overlapping surfaces. Our methodology employs Boolean operations from CGAL [Fabri and Pion 2009] to decompose meshes M and M^* into three volume types. However, a key limitation arises when M and M^* have highly overlapping surfaces, such as when they differ by only a single hole. In such cases, Boolean operations can become error-prone due to difficulties in accurately defining boundaries, leading to ambiguous results, as shown in the inset figure. Even
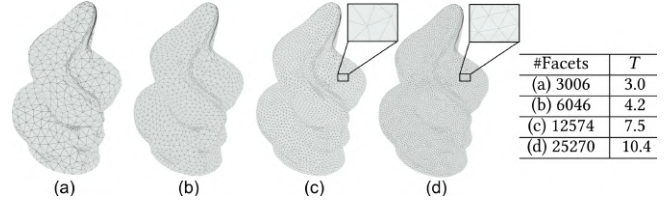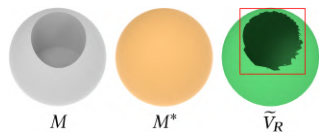
Fig. 22. Illustration of the impact of triangle mesh quantity in the target model (*Sculpture*) on running time. From (a) to (d), we count the total runtime time T (minutes) of our algorithm with varying numbers of faces (#Facets).

	#Facets	T
(a)	3006	3.0
(b)	6046	4.2
(c)	12574	7.5
(d)	25270	10.4

recently proposed robust Boolean operation methods cannot solve it [Trettner et al. 2022]. To address this problem, one could manually segment the volumes to avoid Boolean operations on highly similar meshes or develop a specialized algorithm to handle these complex cases.

Optimizing manufacturing time. Our algorithm prioritizes material savings and does not currently optimize positional block decomposition or tool paths with regard to manufacturing time. A straightforward improvement would involve adjusting the post-processing algorithm to minimize the number of blocks. Alternatively, integrating this objective into the algorithm itself could be considered, though it would require a strategy to balance minimizing blocks with maximizing reutilized volume.

Algorithmic scalability. Geometrically complex models often require more triangle faces to capture intricate details. To evaluate how this affects algorithm runtime, we conducted an experiment measuring runtime against the number of triangle faces. As shown in Figure 22, the runtime of our algorithm scales approximately linearly with the number of faces. Consequently, models with extremely high numbers of faces may require several hours to process. The extended runtime is mainly due to intensive geometric operations, such as Boolean operations. Implementing faster geometric processing methods could significantly enhance the performance of our algorithm. Notably, thin-shell models [Xing et al. 2024] face extra loading challenges in additive and subtractive hybrid manufacturing, which remain underexplored. Our algorithm applies only to watertight solid models.

8 CONCLUSION AND FUTURE WORK

This paper presents the first computational framework for ASHRM process planning on general shapes, explicitly enforcing collision-free constraints and aiming to maximize reutilized volume. We adopt a two-stage approach to handle collision-free constraints, introducing clipping planes to address local constraint and a collision dependency graph to manage global constraints. The effectiveness of our algorithm is demonstrated across a variety of 3D shapes, and its efficiency in tackling the ASHRM process planning problem is validated through physical experiments, simulations, and computational evaluations.

This work offers meaningful progress in ASHRM process planning, yet several critical aspects remain unexplored and merit further study: 1) Incorporating multiple SM-AM sequences to further enlarge the reutilized volume, and add additional SM steps after each

AM step to finish the model surface, ensuring high surface quality. 2) Replacing the current sampling-based orientation strategy with continuous optimization by transforming discrete Boolean operations into continuous ones, such as those achieved using a fuzzy logic strategy. 3) Accommodating slightly concave initial printing base. Although flat initial printing bases are theoretically required to avoid collisions, minor collisions caused by slightly concave interfaces may be acceptable in practice, significantly easing the restrictions imposed by the plano-convex constraint. However, extensive experiments are needed to determine the tolerable degree of concavity and to design a new algorithm that satisfies this constraint. 4) Replacing the current positional manufacturing with simultaneous manufacturing could bypass the plano-convex constraint and overcome the inability of [Wu et al. 2020]'s method to strictly ensure self-supporting, though it further increases the complexity of process planning. 5) Extension to ASHRM machines with different DOFs, requiring minor algorithm adjustments such as restricting clipping planes to the horizontal direction for 3-DOF systems. 6) Enhancing the MCTS strategy using deep reinforcement learning. The primary challenge lies in obtaining high-quality training data and achieving robust generalization performance.

ACKNOWLEDGMENTS

REFERENCES

Chrystiano Araújo, Daniela Cabiddu, Marco Attene, Marco Livesu, Nicholas Vining, and Alla Sheffer. 2019. Surface2Volume: surface segmentation conforming assemblable volumetric partition. *ACM Transactions on Graphics* 38, 4 (aug 2019), 1–16. https://doi.org/10.1145/3306346.3323004

Zhitao Bo, Lin Lu, Andrei Sharf, Yang Xia, Oliver Deussen, and Baoquan Chen. 2017. Printable 3D trees. In *Computer Graphics Forum*, Vol. 36. Wiley Online Library, 29–40.

Robert Bridson. 2007. Fast Poisson disk sampling in arbitrary dimensions. *SIGGRAPH sketches* 10, 1 (2007), 1.

Li Chen, Tak Yu Lau, and Kai Tang. 2020. Manufacturability analysis and process planning for additive and subtractive hybrid manufacturing of Quasi-rotational parts with columnar features. *Computer-Aided Design* 118 (jan 2020), 102759. https://doi.org/10.1016/j.cad.2019.102759

Li Chen, Ke Xu, and Kai Tang. 2018. Optimized sequence planning for multi-axis hybrid machining of complex geometries. *Computers & Graphics* 70 (feb 2018), 176–187. https://doi.org/10.1016/j.cag.2017.07.018

Niechen Chen and Matthew Frank. 2019. Process planning for hybrid additive and subtractive manufacturing to integrate machining and directed energy deposition. *Procedia Manufacturing* 34 (2019), 205–213. https://doi.org/10.1016/j.promfg.2019.06.140

Yuanzhi Chen, Dong He, Jiancheng Hao, Xiaoke Deng, Wenze Zhang, Zichuan Fu, Yichen Wang, Li Chen, Pengcheng Hu, and Kai Tang. 2025. Five-axis hybrid manufacturing with DED and milling for complex multi-branched metallic parts. *International Journal of Computer Integrated Manufacturing* (2025), 1–30.

Ankan Das, Tanmoy Medhi, Sajan Kapil, and Pankaj Biswas. 2024. Different build strategies and computer-aided process planning for fabricating a functional component through hybrid-friction stir additive manufacturing. *International Journal of Computer Integrated Manufacturing* 37, 3 (2024), 350–371.

Ugur M. Dilberoglu, Bahar Gharehpapagh, Ulas Yaman, and Melik Dolen. 2021. Current trends and research opportunities in hybrid additive manufacturing. *The International Journal of Advanced Manufacturing Technology* 113, 3-4 (jan 2021), 623–648. https://doi.org/10.1007/s00170-021-06688-1

Idiano D'Adamo and Paolo Rosa. 2016. Remanufacturing in industry: advices from the field. *The International Journal of Advanced Manufacturing Technology* 86 (2016), 2575–2584.

Andreas Fabri and Sylvain Pion. 2009. CGAL: The computational geometry algorithms library. In *Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems*. 538–539.

Raoul Fonkoua Fofou, Zhigang Jiang, and Yan Wang. 2021. A review on the lifecycle strategies enhancing remanufacturing. *Applied Sciences* 11, 13 (2021), 5937.

Lukasz Golab, Flip Korn, Feng Li, Barna Saha, and Divesh Srivastava. 2015. Size-constrained weighted set cover. In *2015 IEEE 31st international conference on data engineering*. IEEE, 879–890.

Wit Grzesik. 2018. Hybrid additive and subtractive manufacturing processes and systems: A review. *Journal of Machine Engineering* 18, 4 (2018), 5–24.

Gaël Guennebaud, Benoît Jacob, et al. 2010. Eigen v3. http://eigen.tuxfamily.org. (2010).

H Hassanin. 2021. Powder-based laser hybrid additive manufacturing of metals: A review. *The International Journal of Advanced Manufacturing Technology* (2021).

Philipp Herholz, Wojciech Matusik, and Marc Alexa. 2015. Approximating Free-form Geometry with Height Fields for Manufacturing. *Computer Graphics Forum* 34, 2 (may 2015), 239–251. https://doi.org/10.1111/cgf.12556

Alec Jacobson. 2020. Common 3D Test Models. https://github.com/alecjacobson/common-3d-test-models. (2020). Accessed: 2025-05-20.

Alec Jacobson, Daniele Panozzo, et al. 2018. libigl: A simple C++ geometry processing library. (2018). https://libigl.github.io/.

Jason B Jones, Phil McNutt, Riccardo Tosi, Clinton Perry, and David Ian Wimpenny. 2012. Remanufacture of turbine blades by laser cladding, machining and in-process scanning in a single machine. (2012).

Kumar Kanishka and Bappa Acherjee. 2023. A systematic review of additive manufacturing-based remanufacturing techniques for component repair and restoration. *Journal of Manufacturing Processes* 89 (2023), 220–283.

Soonjo Kwon and Yosep Oh. 2023. Optimal process planning for hybrid additive–subtractive manufacturing using recursive volume decomposition with decision criteria. *Journal of Manufacturing Systems* 71 (2023), 360–376.

Jiawei Liang, Dongbo Wu, Shibo Liu, Hui Wang, and Jie Yu. 2024. Research on hybrid remanufacturing process chain of laser cladding, CNC machining and ultrasonic rolling for aero-engine blades. *Journal of Manufacturing Processes* 110 (2024), 331–349.

Jikai Liu, Yufan Zheng, Yongsheng Ma, Ahmed Qureshi, and Rafiq Ahmad. 2020. A topology optimization method for hybrid subtractive–additive remanufacturing. *International Journal of Precision Engineering and Manufacturing-Green Technology* 7 (2020), 939–953.

Tao Liu, Tianyu Zhang, Yongxue Chen, Yuming Huang, and Charlie CL Wang. 2024. Neural slicer for multi-axis 3D printing. *ACM Transactions on Graphics (TOG)* 43, 4 (2024), 1–15.

Tao Liu, Tianyu Zhang, Yongxue Chen, Weiming Wang, Yu Jiang, Yuming Huang, and Charlie CL Wang. 2025. Neural co-optimization of structural topology, manufacturable layers, and path orientations for fiber-reinforced composites. *ACM Transactions on Graphics (TOG)* 44, 4 (2025), 1–17.

Wei Ma, Tianliang Hu, Chengrui Zhang, and Qizhi Chen. 2025. Adaptive remanufacturing for freeform surface parts based on linear laser scanner and robotic laser cladding. *Robotics and Computer-Integrated Manufacturing* 91 (2025), 102855.

Andrew Makhorin. 2012. libigl: A simple C++ geometry processing library. (2012). https://www.gnu.org/software/glpk/.

Stephen T Newman, Zicheng Zhu, Vimal Dhokia, and Alborz Shokrani. 2015. Process planning for additive and subtractive manufacturing technologies. *CIRP annals* 64, 1 (2015), 467–470.

Henri Paris, Guillaume Mandil, et al. 2017. Process planning for combined additive and subtractive manufacturing technologies in a remanufacturing context. *Journal of Manufacturing Systems* 44 (2017), 243–254.

Henri Paris, Guillaume Mandil, et al. 2018. The development of a strategy for direct part reuse using additive and subtractive manufacturing technologies. *Additive Manufacturing* 22 (2018), 687–699.

Ning Qian, Zhengcai Zhao, Yao Li, Yucan Fu, Wenfeng Ding, and Jiuhua Xu. 2024. Adaptive machining of damaged blades repaired by additive manufacturing: a novel double-constrained method for model reconstruction. *Machining Science and Technology* 28, 4 (2024), 547–569.

Lan Ren, Ajay Panackal Padathu, Jianzhong Ruan, Todd E Sparks, and Frank W Liou. 2006. Three dimensional die repair using a hybrid manufacturing system. (2006).

SIEMENS. 2016. SIEMENS NX 2206. https://www.plm.automation.siemens.com/. (Jan. 2016).

Sakraan Sitcharangsie, Winifred Ijomah, and TC Wong. 2019. Decision makings in key remanufacturing activities to optimise remanufacturing outcomes: A review. *Journal of Cleaner Production* 232 (2019), 1465–1481.

Peng Song, Bailin Deng, Ziqi Wang, Zhichao Dong, Wei Li, Chi-Wing Fu, and Ligang Liu. 2016. CofiFab: coarse-to-fine fabrication of large 3D objects. *ACM Transactions on Graphics (TOG)* 35, 4 (2016), 1–11.

Haoyi Sun, Xiang Cheng, Yuanyong Liu, Ruichun Dong, Huanbao Liu, and Guangming Zheng. 2024. Study on hybrid 3D printing and milling process for customized polyether-ether-ketone components. *Proceedings of the Institution of Mechanical

Engineers, Part H: Journal of Engineering in Medicine 238, 10 (2024), 964–972.

Alexander Teibrich, Stefanie Mueller, François Guimbretière, Robert Kovacs, Stefan Neubert, and Patrick Baudisch. 2015. Patching physical objects. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. 83–91.

Philip Trettner, Julius Nehring-Wirxel, and Leif Kobbelt. 2022. EMBER: exact mesh booleans via efficient & robust local arrangements. *ACM Transactions on Graphics (TOG)* 41, 4 (2022), 1–15.

J Michael Wilson, Cecil Piya, Yung C Shin, Fu Zhao, and Karthik Ramani. 2014. Remanufacturing of turbine blades by laser direct deposition with its energy and environmental impact analysis. *Journal of Cleaner Production* 80 (2014), 170–178.

Chenming Wu, Chengkai Dai, Guoxin Fang, Yong-Jin Liu, and Charlie C. L. Wang. 2020. General Support-Effective Decomposition for Multi-Directional 3-D Printing. *IEEE Transactions on Automation Science and Engineering* 17, 2 (apr 2020), 599–610. https://doi.org/10.1109/tase.2019.2938219

Yu Xing, Xiaoxuan Wang, Lin Lu, Andrei Sharf, Daniel Cohen-Or, and Changhe Tu. 2024. Shell stand: Stable thin shell models for 3D fabrication. *Computational Visual Media* 10, 4 (2024), 643–657.

Yu Xing, Yu Zhou, Xin Yan, Haisen Zhao, Wenqiang Liu, Jingbo Jiang, and Lin Lu. 2021. Shell thickening for extrusion-based ceramics printing. *Computers & Graphics* 97 (2021), 160–169.

Shuzhi Xu, Jikai Liu, Kentaro Yaji, and Lin Lu. 2024. Topology optimization for hybrid additive-subtractive manufacturing incorporating dynamic process planning. *Computer Methods in Applied Mechanics and Engineering* 431 (2024), 117270.

Tianyu Zhang, Guoxin Fang, Yuming Huang, Neelotpal Dutta, Sylvain Lefebvre, Zekai Murat Kilic, and Charlie CL Wang. 2022. S3-slicer: A general slicing framework for multi-axis 3D printing. *ACM Transactions on Graphics (TOG)* 41, 6 (2022), 1–15.

Weijun Zhang, Masakazu Soshi, and Kazuo Yamazaki. 2020. Development of an additive and subtractive hybrid manufacturing process planning strategy of planar surface for productivity and geometric accuracy. *The International Journal of Advanced Manufacturing Technology* 109, 5-6 (jul 2020), 1479–1491. https://doi.org/10.1007/s00170-020-05733-9

Xinchang Zhang, Wenyuan Cui, Wei Li, and Frank Liou. 2019. A hybrid process integrating reverse engineering, pre-repair processing, additive manufacturing, and material testing for component remanufacturing. *Materials* 12, 12 (2019), 1961.

Xinchang Zhang, Wei Li, Wenyuan Cui, and Frank Liou. 2018. Modeling of worn surface geometry for engine blade repair using Laser-aided Direct Metal Deposition process. *Manufacturing letters* 15 (2018), 1–4.

Xinchang Zhang and Frank W Liou. 2019. Part Remanufacturing using Hybrid Manufacturing Processes. (2019).

Yufan Zheng and Rafiq Ahmad. 2020a. Automated feature extraction for hybrid additive-subtractive remanufacturing. *Procedia CIRP* 93 (2020), 56–61.

Yufan Zheng and Rafiq Ahmad. 2020b. Feature extraction and process planning of integrated hybrid additive-subtractive system for remanufacturing. *Mathematical Biosciences and Engineering* 17, 6 (2020), 7274–7301.

Yufan Zheng, Jikai Liu, and Rafiq Ahmad. 2020. A cost-driven process planning method for hybrid additive–subtractive remanufacturing. *Journal of manufacturing systems* 55 (2020), 248–263.

Fanchao Zhong, Yonglai Xu, Haisen Zhao, and Lin Lu. 2023a. As-continuous-as-possible extrusion-based fabrication of surface models. *ACM Transactions on Graphics* 42, 3 (2023), 1–16.

Fanchao Zhong, Haisen Zhao, Haochen Li, Xin Yan, Jikai Liu, Baoquan Chen, and Lin Lu. 2023b. VASCO: Volume and Surface Co-Decomposition for Hybrid Manufacturing. *ACM Transactions on Graphics (TOG)* 42, 6 (2023), 1–17.

Qingnan Zhou and Alec Jacobson. 2016. Thingi10K: A Dataset of 10,000 3D-Printing Models. *arXiv preprint arXiv:1605.04797* (2016).

A TERMINOLOGY LIST

Table 3. This table lists and clarifies all symbols used in our algorithm.

Symbol	Significance	Usage
M	initial model	Section 3
M^*	target model	Section 3
$\widetilde{V}_R$	ideal reutilized volume	Section 3
$\widetilde{V}_S$	initial subtractive volume	Section 3
$\widetilde{V}_A$	initial additive volume	Section 3
V_T	SM-then-AM volume	Section 3
V_R	new reutilized volume, $= \widetilde{V}_R - V_T$	Section 3
V_S	new subtractive volume, $= \widetilde{V}_S + V_T$	Section 3
V_A	new additive volume, $= \widetilde{V}_A + V_T$	Section 3
$\mathcal{B}$	initial printing base	Section 3
$\mathbb{C}$	a set of clipping planes	Section 3
$V_R^{\mathbb{C}}$	the reutilized volume resulting from clipping with $\mathbb{C}$	Section 4.1
$\widetilde{\mathcal{B}}$	initial initial printing base between $\widetilde{V}_R$ and $\widetilde{V}_A$	Section 4.1
b_i	a connected component of $\widetilde{\mathcal{B}}$	Section 4.1
M	the number of connected components of $\widetilde{\mathcal{B}}$	Section 4.1
$\widetilde{c}_{ij}$	a candidate clipping plane	Section 4.1
N	the number of candidate clipping planes	Section 4.1
$n_{i,j}$	the normal of $\widetilde{c}_{ij}$	Section 4.1
$p_{i,j}$	original point of $\widetilde{c}_{ij}$	Section 4.1
V_T^{ij}	SM-then-AM volume, clipped from $\widetilde{V}_R$ using $\widetilde{c}_{ij}$	Section 4.1
V_R^{ij}	new reutilized volume, $= \widetilde{V}_R - V_T^{ij}$	Section 4.1
V_A^{ij}	new additive volume, $= \widetilde{V}_A + V_T^{ij}$	Section 4.1
$\widetilde{\mathcal{B}}^{ij}$	new initial printing base between V_R^{ij} and V_A^{ij}	Section 4.1
$s_{i,j}$	a sub-set of $\widetilde{\mathcal{B}}$ for $\widetilde{c}_{i,j}$	Section 4.1
$W_{i,j}$	a weight for $\widetilde{c}_{i,j}$	Section 4.1
$Z_{i,j}$	a binary integer variable for $\widetilde{c}_{i,j}$	Section 4.1
$V_T^{\mathbb{C}}$	new SM-then-AM volume, $= \widetilde{V}_R - V_R^{\mathbb{C}}$	Section 4.2
$V_A^{\mathbb{C}}$	new additive volume, $= \widetilde{V}_A + V_T^{\mathbb{C}}$	Section 4.2
$V_S^{\mathbb{C}}$	new subtractive volume, $= \widetilde{V}_S + V_T^{\mathbb{C}}$	Section 4.2
G_{cd}	a collision dependency graph	Section 4.2
$\mathbb{C}^L$	a set of local clipping planes	Section 4.2
c_i^L	a local clipping plane	Section 4.2
$V_R^{\mathbb{C}^L}$	the reutilized volume resulting from clipping with $\mathbb{C}^L$	Section 4.2

B COLLISION DEPENDENCY GRAPH CONSTRUCTION

Sampling. As illustrated in Figure 23 (a), we sample points within three types of volumes ($\widetilde{V}_S$, $\widetilde{V}_A$, $\widetilde{V}_R$) for subsequent collision detection. Since the core algorithm focuses on obstructions caused by $\widetilde{V}_R$, uniform sampling is performed within $\widetilde{V}_R$ by voxelizing it and using the voxel centers as sampling points $\mathbb{S}_R$. This ensures that any obstruction caused by the surface of the reutilized volume during optimization is accounted for. In contrast, for $\widetilde{V}_S$ and $\widetilde{V}_A$, we perform uniform sampling solely on their surfaces using Poisson disk sampling [Bridson 2007], as their outer surfaces fully enclose the internal volume, ensuring that any internal point is accessible if the entire surface is collision-free [Chen et al. 2020]. These surface sampled points are denoted as $\mathbb{S}_S$ and $\mathbb{S}_A$. We sample all points at an interval of 2mm.
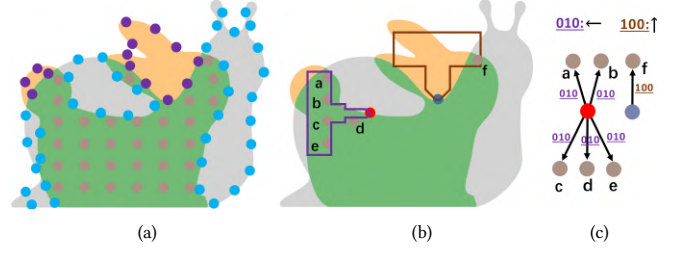

Fig. 23. Illustration of the collision dependency graph construction. (a): Sample points in the surface of $\widetilde{V}_S$ (blue) and $\widetilde{V}_A$ (purple). For $\widetilde{V}_R$, sample points within its volume (brown). (b): Perform collision detection to calculate the SM and AM inaccessible points from the sampled points of $\widetilde{V}_S$ and $\widetilde{V}_A$. The red and dark blue points respectively represent the SM and AM inaccessible points obscured by sampling points in $\widetilde{V}_R$. Here we just demonstrate one SM/AM direction. (c): Construct the collision dependency graph. If an inaccessible point is obscured by a sampled point of $\widetilde{V}_R$ in a certain direction, add an edge to the graph, and use the bit-vector to represent the direction. In the 2D case, we totally sample three directions: $\uparrow, \leftarrow, \rightarrow$.

Collision detection and building graph. We then perform cutter/nozzle collision detection based on the sampling points to identify which regions of $\widetilde{V}_S$ and $\widetilde{V}_A$ are inaccessible due to obstructions from $\widetilde{V}_R$. Specifically, for each sampling point $s_i \in \mathbb{S}_S/\mathbb{S}_A$, we check whether cutter/nozzle collides with any $s_j \in \mathbb{S}_R$ along a given cutter/nozzle direction $d_k \in D$, where D are uniformly sampled within the upper hemisphere. We classify $s_i \in \mathbb{S}_S/\mathbb{S}_A$ as "inaccessible point", (marked in red/blue in Figure 23 (b) if:

$$\forall d_k \in D, \exists s_j \in \mathbb{S}_R, s_j \cap cutter/nozzle = \varnothing. \tag{4}$$

Subsequently, we apply the method proposed in [Zhong et al. 2023b] to construct a collision dependency graph (G_{cd}), which record the collision information between $s_j \in \mathbb{S}_R$ and inaccessible points. To represent the cutter/nozzle collision directions, we use a bit-vector, as shown in Figure 23 (c), where each bit represents a specific cutter/nozzle direction.

It is worth noting that our method and [Zhong et al. 2023b] differ completely in both formulation and overall methodology — [Zhong et al. 2023b]'s approach relies on a single shape and minimizes additive–subtractive switches. Methodologically, our two-stage framework is entirely distinct from theirs, except that the second stage adopts their graph-based method to evaluate global collision-free constraints.

C IMPLEMENTATION DETAILS OF MCTS

We first introduce the structure of the Monte Carlo Tree: Each layer represents a local clipping plane c_i^L, with the number of layers equal to $|\mathbb{C}^L|$. Each node contains a state and two values (left and right). The state represents the normal and original point of c_i^L, while the left and right values correspond to the reward and the number of selections, respectively. Figure 24 shows the (n+1)-th iteration of the MCTS process. The four MCTS stages are as follows:

(1) Selection (Figure 24 (a)): From the root node in the tree, choose the "most potential" leaf node that is not fully expanded, using the
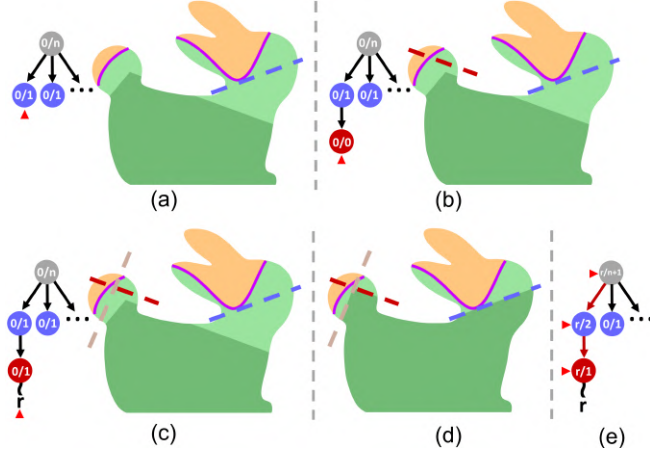
Fig. 24. Demonstration of MCTS for clipping planes poses modification. (a)~(e) show one complete iteration of MCTS. (a) Selection stage: select the blue node by the UCB function, which represents a pose of the first local clipping plane. (b) Expansion stage: expand the blue node, get the new red node which represents a pose of second local clipping plane. (c), (d) Simulation stage: randomly determine the pose of remaining local clipping plane (brown), and calculate the $V_R^{\mathbb{C}^L}$. (e) Back-propagation stage: back-propagation the reward "r" calculated by Equation 6 to the current expanded node and its ancestor nodes.

Upper Confidence Bound (UCB) function:

$$\mathbb{UCB}(i) = \frac{Q(i)}{N(i)} + \sqrt{2} \cdot \frac{log(T)}{N(i)}, \tag{5}$$

where $Q(i)$ and $N(i)$ are the left and right values of node i, respectively, and T is the total number of iterations. The reward $Q(i)$ is the key term driving selection, designed to maximize the volume of $V_R^{\mathbb{C}^L}$. The reward is defined as:

$$Q(i) = \begin{cases} 0, \text{if } V_R^{\mathbb{C}^L} \text{ violates collision-free constraints,} \\ \frac{V_R^{\mathbb{C}^L} - V_R^{\mathbb{C}}}{\widetilde{V}_R - V_R^{\mathbb{C}}}, \text{if } V_R^{\mathbb{C}^L} < V_{max}, \\ \frac{V_R^{\mathbb{C}^L} - V_R^{\mathbb{C}}}{\widetilde{V}_R - V_R^{\mathbb{C}}} \cdot \left(\frac{V_R^{\mathbb{C}^L}}{V_{max}}\right)^3, \text{otherwise.} \end{cases} \tag{6}$$

Here, V_{max} is the maximum volume of $V_R^{\mathbb{C}^L}$ across all simulations. This formulation ensures that poses of $\mathbb{C}^L$ resulting in larger $V_R^{\mathbb{C}^L}$ are favored.

(2) Expansion (Figure 24 (b)): Expand a child node for the selected node, i.e., generate a new normal and original point for c_{i+1}^L. We first randomly determine a normal and then offset c_{i+1}^L along the normal by a random distance to determine the original point.

(3) Simulation (Figure 24 (c,d)): Randomly determine the normals and original points of $c_{i+2}^L, .., c_{|\mathbb{C}^L|}^L$ to quickly generate a candidate $V_R^{\mathbb{C}^L}$ through simulation. Then, we check whether $V_R^{\mathbb{C}^L}$ satisfies both collision-free constraints.

(4) Back-propagation (Figure 24 (e)): Update the left and right values of the current expanded node and its ancestors: add the reward to the left value and increment the right value by one.

D SETUP DETERMINATION

The coarse-to-fine evaluation is based on our two insights: 1) Although a larger $\widetilde{V}_R$ does not guarantee a larger $V_R^{\mathbb{C}}$, it is often helpful in achieving a larger $V_R^{\mathbb{C}}$ as it provides a better initial volume. 2) When the lowest point of the initial printing base (along the vertical direction) is positioned low, the feasible locations for the clipping plane are severely constrained to avoid collisions with the fabrication base, leading to a smaller $V_R^{\mathbb{C}}$. Conversely, initial printing base with a higher lowest point are beneficial for increasing the volume of $V_R^{\mathbb{C}}$, as shown in the comparison in Figure 8 (a,b). We state the details of setup determination below.

In the coarse stage, we uniformly sample 100 directions on the Gaussian sphere, take the centroid of M^* as the center of rotation, and rotate it by each direction separately. In addition, align the bottom of the rotated M^* and M along the Z-axis (in the 3D scene), and make their bottom centroids coincide. Then we score each direction by the function as follows:

$$Score_1 = w_1 \cdot Volume(\widetilde{V}_R) + w_2 \cdot Area(\mathcal{T}), \tag{7}$$

where $Area(\mathcal{T})$ is the sum of the areas of the triangle meshes in the rotated M^* that satisfy $n_f \cdot d_v \geq -sin(50°)$. Here, n_f is the normal of a triangle, and d_v is the vertical direction. For the second term, since both rotation and translation significantly affect the position of initial printing base, we do not use their positions directly as criteria. As shown in the inset figure, when using the height of the printing base

as the second term, an appropriate rotation may be misjudged due to the influence of translation (the first and third cases). Instead, we adopt a rough estimate as the second term, which encourages M^* to "stand up" instead of "lie down", which is conducive to the initial printing base being located at a high position. In all of our experiments, we set the weight factors $w_1 = 0.4, w_2 = 0.6$.

In the fine-tuning stage, we pick the five highest-scoring M^* with determined rotation from the coarse stage and translate them by the distance $d_t = d_{X/Y} * step/10$, where $step = \{-2, -1, .., 2\}$ along the X and Y axes separately, and $d_{X/Y}$ indicates the size of M^* along the X/Y axes. Then we score the position of M^* by another function:

$$Score_2 = w_1 \cdot Volume(\widetilde{V}_R) + w_2 \cdot (\frac{Min_{\mathcal{B}}}{Max_V} + (1 - \frac{Area(\mathcal{B})}{Area(M^*)})^2), \tag{8}$$

where w_1 and w_2 are the same as those in Equation 7. $Min_{\mathcal{B}}$ and Max_V respectively denote the minimum Z-value of initial printing base and the maximal Z-value of $\widetilde{V}_R$. Finally, we determine the rotation and translation of M^* with the highest score.

E THE PSEUDO-CODE

This section provides the pseudocode for our core algorithm (Sec. 4).

Algorithm 1 Reutilized volume maximization

1: **Input:** M and M^* with determined setup;
2: **Output:** The largest reutilized volume that satisfies both collision-free constraints;
3: $\widetilde{V}_S \leftarrow M - M^*, \widetilde{V}_A \leftarrow M^* - M, \widetilde{V}_R \leftarrow M \cap M^*$
    // The first stage: only consider plano-convex constraint
4: $\mathcal{B} \leftarrow$ all connected components from initial printing base;
5: **for** each b_i of $\mathcal{B}$ **do**
6:     **for** j = 0; j < 100; j++ **do**
7:         Sample normal direction n_j on Gaussian sphere;
8:         Generate candidate clipping plane $\widetilde{c}_{i,j}$ with normal n_j, which intersects b_i at the point closet to the fabrication base $\mathcal{B}$;
9:         **for** each $b_k \in \mathcal{B}$ ($b_k \neq b_i$) **do**
10:             **if** b_k and $\mathcal{B}$ are on opposite sides of $\widetilde{c}_{i,j}$ **then**
11:                 $s_{i,j} \leftarrow \{b_k, s_{i,j}\}$;
12:         Clip $\widetilde{V}_R$ by $\widetilde{c}_{i,j}$, get the remaining volume $V_R^{i,j}$;
13:         $W_{i,j} \leftarrow 1 - Volume(V_R^{i,j})/Volume(\widetilde{V}_R)$;
14: $\mathbb{C} \leftarrow$ solutions by solving the integer linear program Eq. (2);
15: $V_R^{\mathbb{C}} \leftarrow$ clip $\widetilde{V}_R$ by $\mathbb{C}$;

    // The second stage: consider both the plano-convex constraint and the accessibility constraint
16: Graph $G_{cd} \leftarrow \varnothing$;
17: Uniformly sample points $\mathbb{S}_S$ and $\mathbb{S}_A$ on the surface of $\widetilde{V}_S$ and $\widetilde{V}_A$ using Poisson disk sampling;
18: Uniformly sample points $\mathbb{S}_R$ within $\widetilde{V}_R$ by voxelization.
19: Uniformly sample $\mathcal{S}^S \leftarrow 100$ directions; $\mathcal{S}^A \leftarrow 100$ directions;
20: **for** each direction of $\mathcal{S}^S$ and $\mathcal{S}^A$ **do**
21:     **for** each point of $\mathbb{S}_S$ and $\mathbb{S}_A$ **do**
22:         Collision detection between the point and all $\mathbb{S}_R$;
23:         Update G_{cd} by adding nodes and edges;
24:         Add bit-vector for each edge;
25: Iteratively Offset all $c_i \in \mathbb{C}$ with the step = 30% of the distance between c_i and $\mathcal{B}$ until the accessibility constraint is satisfied;
26: SM-then-AM volume $V_T = \widetilde{V}_R - V_R^{\mathbb{C}}$; $\mathbb{C}^L \leftarrow \varnothing$;
27: **for** each connected component of V_T **do**
28:     Generate local cutting planes c_i^L;
29:     $\mathbb{C}^L \leftarrow \{\mathbb{C}^L, c_i^L\}$;
30: **for** itr = 0; itr < 500; itr++ **do**
31:     Choose the leaf node N_i^n from the root node by Eq. (4);
32:     Add a child node N_{i+1}^m for N_i^n, set left and right values = 0;
33:     Randomly determine the normal of c_{i+1}^L;
34:     Offset c_{i+1}^L along the normal direction with random distance;
35:     Randomly determine the normal and origin of $\{c_{i+2}^L, c_{i+3}^L, ..\}$;
36:     $V_R^{\mathbb{C}^L} \leftarrow$ clip $V_R^{\mathbb{C}}$ by C^L;
37:     Update and check G_{cd} according to $V_R^{\mathbb{C}^L}$;
38:     **if** $V_R^{\mathbb{C}^L}$ satisfy both collision-free constraints **then**
39:         reward $\mathcal{R} \leftarrow$ Eq. (5);
40:     **else**
41:         $\mathcal{R} = 0$;
42:     Left values of N_{i+1}^m and its ancestor nodes += $\mathcal{R}$;
43:     Right values of N_{i+1}^m and its ancestor nodes += 1;
44: Take the largest $V_R^{\mathbb{C}^L}$ as the final reutilized volume.

F ASHM EQUIPMENT MODIFICATION

The ASHM equipment modified in [Zhong et al. 2023b] has an obvious defect: it lacks a retraction function, leading to excessive stringing during the AM process. To address this issue and improve surface quality in AM, we enhanced the equipment by incorporating a retraction module. Specifically, we modified the I/O module circuit hardware of the 5-axis CNC machine to enable communication with external attached modules. An Arduino UNO R3 microcontroller board was used to process signals between the CNC machine and the added additive extruder. The extruder is instructed by the controller of the 5-axis subtractive machine through G-code commands. We defined custom G-code commands for precise control:

- M70/M71: Open and close the extruder module.
- M72/M73: Enable and disable extrusion movement.
- M74/M75: Enable and disable retraction movement.