Volume xx (200y), Number z, pp. 1–17

State of the art in computational mould design

T. Alderighi[1] L. Malomo[1] T. Auzinger[2] B. Bickel[3] P. Cignoni[1] and N. Pietroni[4]

[1]Istituto di Scienza e Tecnologie dell'Informazione Alessandro Faedo (ISTI-CNR), Visual Computing Lab, Italy
[2]Automold, Austria
[3]Institute of Science and Technology (IST Austria), Austria
[4]University of Technology Sydney, Australia

Abstract

Molding refers to a set of manufacturing techniques in which a mold, usually a cavity or a solid frame, is used to shape a liquid or pliable material into an object of the desired shape. The popularity of molding comes from its effectiveness, scalability, and versatility in terms of employed materials. Its relevance as a fabrication process is demonstrated by the extensive literature covering different aspects related to mold design, from material flow simulation to the automation of mold geometry design. In this state-of-the-art report, we provide an extensive review of the automatic methods for the design of molds, focusing on contributions from a geometric perspective. We classify existing mold design methods based on their computational approach and the nature of their target molding process. We summarize the relationships between computational approaches and molding techniques, highlighting their strengths and limitations. Finally, we discuss potential future research directions.

CCS Concepts
• ***Computing methodologies*** → ***Mesh geometry models; Shape analysis;***

1. Introduction

Molding refers to a set of industrial and "traditional" manufacturing processes in which a cavity, called a mold, is filled with liquefied or expanding material. The hardening of the material will materialize it into the desired solid shape. A solid copy of the object, called a cast, can be safely extracted from the mold once the material is completely hardened. This simple yet effective fabrication method has a millenary tradition in art and industry (see Figure 1 [Dal09]). The origins of molding can be traced back to the dawn of civilization some 5000 years ago, with the earliest known casting dating back to 3200 BC [KSAS17]. In the last centuries, molding has become the standard for industrial mass production. Moreover, it is still a popular technique employed for small and medium-scale productions across various contexts. Artists use molding to manufacture resin miniatures [BOP10], or to design jewelry [Wan11]. In the food industry, molds are employed to manufacture chocolate sculptures [Wor16] or candies.

Figure 1: *Ancient Greek (5th/4th century BC) molds, used to mass produce clay figurines. Picture by Giovanni Dall'Orto, November 9, 2009.*

Despite the recent diffusion of additive manufacturing devices, industrial production still relies heavily on molding and casting techniques. 3D printing allows the accurate fabrication of arbitrarily complex shapes. However, even though these technologies are becoming faster and more precise, they are still limited in scalability because of the required time (cost) per replica. In addition, additive manufacturing devices can operate only with a limited range of materials. Consumer-level 3D printers are mainly limited to FDM (generally targeting plastic materials like PLA, PCA, and ABS) and SLA (targeting photopolymers like UV-curing resins). By contrast, molding can employ a broader range of materials and enables high volume production.

Unfortunately, such power, in terms of scalability and material availability, comes at a price. Unlike additive manufacturing, molding imposes severe limitations on the class of fabricable shapes. These limitations are mainly related to the cast extraction process or the physical properties of the casting material [Cam95]. Hence, molding usually requires extensive planning involving highly specialized engineers whose expertise spans many knowledge domains. Indeed, mold design is very often an iterative process in which designers must manually re-adjust the shape of a manufactured part to comply with the constraints involved in its fabrication [Ree02].

Given its relevance as an industrial manufacturing process, molding has been the object of extensive studies and research covering several engineering areas and involving different aspects of mold design [FFN04, Wan11, Kaz16, Cam15]. Most research focuses on developing algorithms to automatize and optimize the design process, and many heterogeneous elements concur in designing a proper mold. First, a suitable mold must be fabricable, and additive technologies might fit the purpose. However, metallic molds often need to be manufactured using subtractive techniques imposing restrictions on the complexity of the mold itself. Second, a suitable mold must guarantee the extraction of the cast. Also, the reproduction quality is dependent on the material flow within the mold, as a proper flow diminishes the impact of common casting artifacts such as material flashing and seams. Finally, in large-scale production, it is also worth considering other minor aspects of the mold design that have a massive impact on the production cost, such as the casting times or the material used.

Because of the different aspects involved, automatic mold design and optimization are challenging tasks requiring the modeling and analysis of complex problems, such as motion planning, non-linear physical simulation, multiple body contact, and combinatorial optimization. Traditionally, the development of computer-aided solutions for molds design, like Moldex3D [CSC21] or Moldflow [Aut21], has focused more on providing tools for evaluating and then optimizing the engineering aspects, such as material flow simulation [TSJ*18, ILW19]. However, these tools still require the user-in-the-loop to provide limited support for generating the final mold geometry. Recent advancements in shape optimization, geometry processing, and physical simulation in graphics have traced a new path for creating automatic mold design tools.

This survey provides an overview of the automatic tools and algorithms that would allow the non-expert user to generate effective molds for complex, arbitrary shapes created without explicitly accounting for the actual fabrication processes. These techniques bridge the gap between rapid prototyping and production for industrial and artisanal applications. In the future, the automation of complex mold design can impact the industry, where mass customization calls for efficient limited volume production processes. Moreover, assembling an industrial pipeline for molding often requires multiple iterations in an engineer-design-fabricate loop, including prototyping, fabrication, and final testing of the mold to be used. In this scenario, the complete automation of the mold design becomes fundamental to avoid bottlenecks.

We classified different molding techniques (including casting and injection molding) by considering how the mold is assembled and how the cast is extracted. One crucial distinction is whether the mold is sacrificial, like the ones used in break-away molds or reusable, often called permanent molds.

1.1. Sacrificial molds

Sacrificial molds, in the form of lost wax or investment casting, have been extensively used since ancient times to produce intricate metal casts for jewelry and sculptures [Sia05, Hun80]. Nowadays, the two main fabrication processes exploiting sacrificial molds are *investment casting* [SDBP12] (the industrial process for lost-wax casting) and *sand casting* [SSS14]. In the first case, a disposable replica of the desired part, called the (wax) *pattern*, is coated with refractory materials to obtain the sacrificial mold. The pattern is then melted or vaporized, leaving the mold cavity, also called the *investment*, empty. The casting material is finally poured into the mold and released after its solidification by breaking or dissolving the mold itself, a process called *divesting*. Similarly, in sand casting, an investment is assembled from multiple mold parts made of compacted sand. Each sand mold part is obtained by compacting the sand over special reusable molds, called *patterns*, that embed the object details and all the additional geometry required for material flow. Again, after the casting operations, the sand mold is destroyed to release the shape, but is efficiently rebuilt using the reusable patterns.

One of the main advantages of these methods is their ability to handle high melting point metal alloys. Moreover, investment casting does not pose substantial constraints on the class of fabricable shapes. Concurrently, it provides dimensional accuracy and an excellent surface finish. However, in terms of production scaling, both sacrificial molds performance is inferior to reusable ones due to the high per-part costs involved. In the case of investment casting, scaling up to high-volume production would require an efficient way to replicate the wax pattern, such as designing reusable molds for the pattern itself. The integration of 3D printing technologies for the direct fabrication of the (wax) patterns [CCL*05, RMRD21] or the refractory mold itself could make the techniques more scalable for small to medium volume productions.

Recently, Shakeri et al. [SEB*21], proposed a novel take on sacrificial molds for the efficient rapid prototyping of developable surface objects. In their work, they propose a method to design wax-stiffened paper molds computationally, by unfolding the object 3D geometry into a 2D pattern. This pattern can then be easily printed on paper and cut to assemble the single-use mold efficiently.

1.2. Reusable molds

While the design of break-away molds can pose interesting manufacturing problems, the fact that the mold is sacrificial makes it a less attractive problem from a *geometry processing* and *computational fabrication* standpoint. In contrast, designing effective permanent molds poses a set of challenging open research problems that can be effectively modeled and tackled from a *computational geometry* perspective. Moreover, automating the design of reusable molds would also indirectly benefit fabrication processes relying on sacrificial molds by making it easier to replicate the patterns used in the process. For these reasons, the research presented in this survey will focus on automating the design of permanent molds for complex objects for different molding techniques, each offering its own challenges.

A necessary condition for a mold to be reusable (hence allowing multiple casting iterations) is the existence of an extraction process of the cast object from the mold that damages neither of the two. Providing such a guarantee is a challenging task whose complexity is directly related to the geometry of the manufactured object. These problems open up a series of interesting questions from a computational geometry perspective: what does it mean for an object to be extractable? How can we guarantee that the mold can be assembled and disassembled? How can we define an optimality criterion for mold design? How can we automate the generation of such molds blending these metrics and constraints?

2. Automatic mold design, a research survey

In the last two decades, mold design automation has been an active and exciting research topic for the CAD and mechanical engineering community. More recently, digital and advanced fabrication topics are gaining more and more relevance also in the computer graphics research community. In this survey, we focus on re-usable molds, that is, the ones that can be re-utilized to cast a shape multiple times. While this strategy allows amortizing the time used for mold production, manufacturing requires a complex design effort.

We can classify the techniques for re-usable mold design in three high-level classes, based on the target fabrication technique and the physical constraints involved:

Rigid molds design The use of permanent rigid molds is a de-facto standard in the mass production of plastic and resin artifacts. For example, industrial injection molding uses rigid molds. While these methods allow large-scale automatized production, they also impose severe limitations in the class of fabricable shapes. As the mold is rigid, the extraction of the cast object can be seriously limited by the presence of overhangs or handles. Hence, these methods usually split the mold into multiple pieces or, more often, change the target shape to lower its geometric complexity. In Section 4, we will present a comprehensive discussion of the variety of methodologies and design choices that characterize the state-of-the-art for rigid molds design.

Flexible molds design Another class of methods focuses on generating molds composed of flexible materials, like silicone or elastic polymers. The elasticity of the mold material allows for the relaxation of several fabrication constraints imposed by rigid molding. Because of this increase in expressive power, flexible molds expand the class of fabricable shapes. Therefore, they are often used to fabricate objects with rich surface details or complex geometry, especially in the props and decorative industry. Unfortunately, the extraction process is usually much more complicated as it requires imposing complex deformation onto the mold to detach each piece from the cast shape. Consequently, this method is not particularly suitable for automation as it usually requires manual labor. We will provide an overview of techniques based on flexible molds in Section 5.

Alternative molds design This class collects a variety of works that are not strictly related to classical casting. We can extend the definition of mold to other fabrication techniques in which a cover is used to fabricate an object. This class includes, for example, industrial fabrication processes like thermoforming or the use of formworks for architectural and decorative applications. We will provide an overview of techniques for alternative molds design in Section 6.

3. Problem statement and terminology

This section introduces the main concepts and terminology needed to understand the different techniques. As previously noted, molds are composed of several components that, once assembled, form an inner cavity that is filled with some liquid casting material. The casting material then solidifies as a result of some chemical or physical process, taking the shape of the cavity. Finally, the (permanent) mold can be opened and the cast part ejected so that the mold can be reassembled to repeat the casting process. Finally, the cast parts may require some post-processing operations. These operations vary from simple polishing (to remove excess material along the mold seams and material inlets) to the assembly of different parts (Figure 2). While most of these operative steps can be largely automated, depending on the molding technology used, the most time and expertise-demanding step is the definition of the mold design. Typically, this process is executed by expert designers with the help of CAD and CAM solutions, or manually by artisans. Methods for the automatic design of molds could thus significantly reduce the cost of setting up a molding fabrication process.

The primary requirement of a mold is to ensure the existence of an extraction sequence so we can remove the mold pieces without damaging the cast object or the mold. The task of designing a valid extraction sequence strictly depends on the type of technology used (rigid or flexible molds). Rigid molds require the mold pieces to be rotated or translated. In contrast, flexible molds allow extra degrees of freedom due to the possibility of deforming the mold during the extraction process. Since finding an optimal extraction sequence requires exploring a very complex space of solutions, the methods for automatic mold design consider the extraction process as composed of a sequence of linear extraction paths (translation). This requirement further constrains the problem but, at the same time, makes it easier to solve.

Figure 3 shows the entities involved in a casting operation:

Mold piece a single component of the mold assembly; a mold assembly can include one (for very simple cases) to multiple pieces;
Parting direction the direction used to detach a mold piece from the cast object;
Parting surface the contact surface separating two adjacent mold pieces;
Parting lines closed curves that separate the cast surface among the different mold pieces. These lines are the boundary of the parting surface touching the cast object.

When considering only linear rigid translation for the mold extraction process, the problem of designing a valid mold can be reduced to associating a valid parting direction to every portion of the object surface. This approach is equivalent to finding a partitioning of the object, such that each part is associated with one or more valid extraction directions. On the object surface, the partitioning boundaries define the parting lines of the mold assembly.

A parting direction d is valid for a surface part S_i if S_i is *globally*

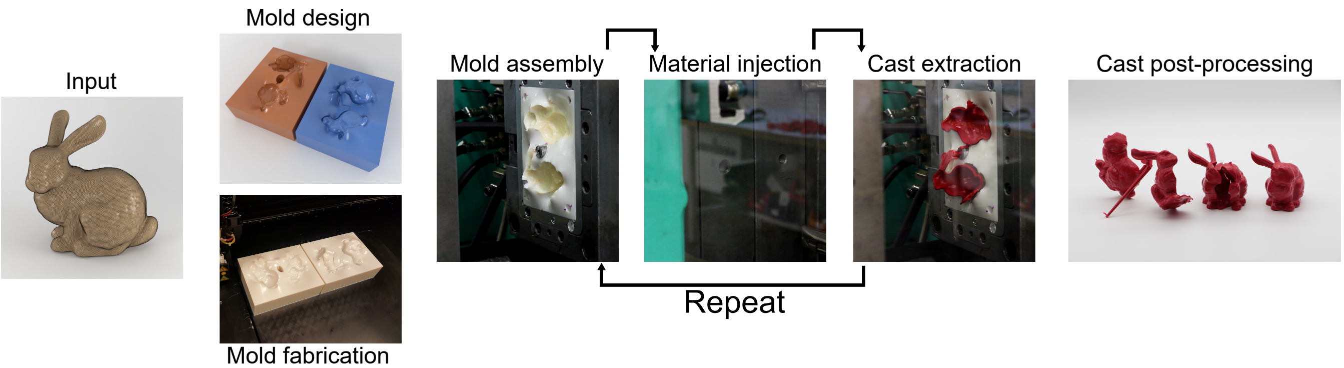


Figure 2: *In general, a molding process pipeline based on reusable molds would take the following steps: first, the desired model geometry is analyzed and an adequate mold is designed, depending on the application requirements. The mold is then fabricated; for industrial applications, the mold is typically milled, while for prototypes or smaller productions, 3D printing is often used. Once the mold is ready, the actual molding cycle can be repeated many times to obtain multiple copies of the desired shape. Finally, the post-processing can involve different operations, from assembly of multiple parts to polishing the cast surface.*

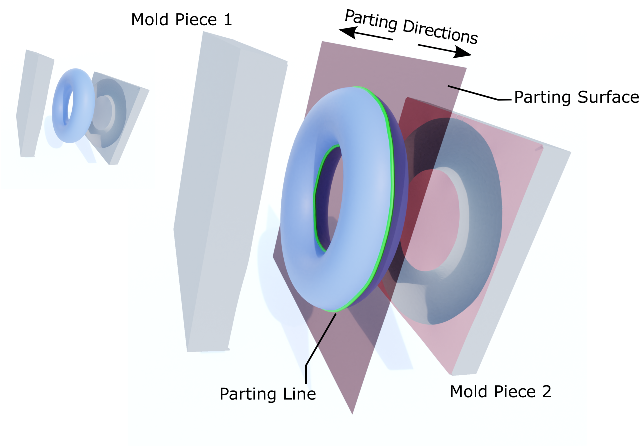


Figure 3: *A simple example illustrating the main concepts and terms that describe a casting mold.*

accessible from the direction d. *Global accessibility* implies that the surface part S_i can translate along the direction d towards the exterior without intersecting any other part of the object. Assuming that no further obstruction exists between mold pieces, this means that we can disassemble a mold piece covering S_i along the direction d. Testing S_i for accessibility along a direction d is equivalent to checking whether, for any point in S_i, casting a ray along d, there is no intersection with any other part of S_i (local accessibility) or the rest of the object (global accessibility). In other words, for S_i to be moldable, it is a necessary condition that S_i can be represented as a height field with respect to direction d. There are two cases in which the accessibility test fails: *overhangs* and *overlaps*.

Overhangs (Figure 4 red) are parts of S_i that violate the local accessibility constraint and thus cannot be locally represented as a height field. Overhang violations can be easily checked by testing the following necessary condition. The outwards normal n_f of any face $f \in S_i$ must satisfy

$$n_f \cdot d \geq 0, \quad \forall f \in S_i.$$

Overlaps (Figure 4 blue) can be formally defined as pairs of faces $f_i, f_j \in S_i$ such that, applying the projection $P_d = I - dd^T$, the intersection is not empty. This implies that parts of S_i overlap along its prescribed extraction direction d, making the extraction impossible. We can verify the overlap condition by checking

$$P_d f_i \cap P_d f_j = \emptyset, \quad \forall f_i, f_j \in S_i.$$

In general, testing for global (ray-) accessibility ensures that both of these problems, generally referred to as *undercuts*, are detected. When casting a ray along the direction d, any front-face intersection defines a part in an overhang, while any back-face intersection detects the presence of an overlap. This is the main reason why

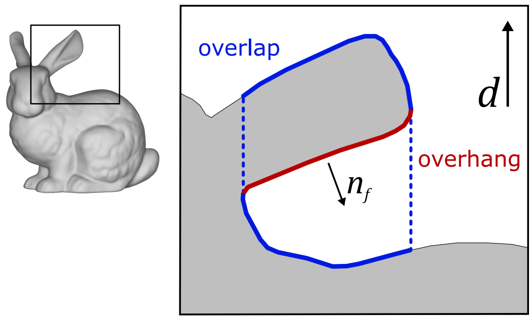


Figure 4: *A schema showing the undercut constraints that need to be addressed when designing a mold. The overhang constraint (in red) is a local constraint requiring that the normals of all faces extracted along direction d must lie in the same hemisphere as d. The overlap constraint (in blue) is a global constraint that requires that the projections along direction d of any pair of faces will have no intersections.*

many state-of-the-art methods [KBM06, LQ14, AMG*18] use GPU rendering or ray-casting to evaluate parting directions, as they exploit the massively parallel nature of the problem.

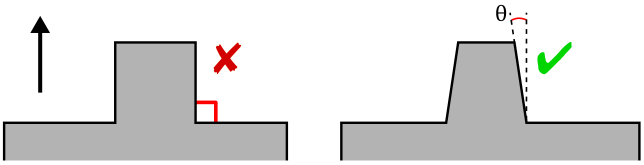


Figure 5: *Vertical walls can create excessive friction during the mold extraction phase. Ensuring a small slope angle* θ *between the extraction direction and the vertical surfaces of the cast part makes the ejection easier.*

In practice, during part and mold design, we can also impose a further constraint to avoid vertical walls aligned with the parting direction. Mold design best practices suggest ensuring a small slope angle, called *draft angle*, to reduce the friction that develops during the sliding action of the mold pieces along vertical parts of the object (Figure 5).

In the case of flexible molds, the elasticity of the mold material allows for more degrees of freedom in the extraction path sequence, as the mold pieces can stretch and deform during extraction. As a consequence, the concepts of global accessibility and undercuts presented above do not pose such an hard constraint as in the case of rigid molds design. This widens the search space for valid flexible mold decompositions. To cope with this less constrained design space, approaches targeting the automatic design of flexible molds rely either on physical simulations of the extraction process [MPBC16] or on geometric heuristics [AMG*18, AMG*19] to enforce the extractability of the mold from the cast object (Section 5 provides a detailed discussion of such approaches).

4. Rigid mold design

Generating rigid molds is a complex design problem characterized by hard geometric constraints that must be satisfied to ensure the feasibility of the casting process. In general, given an input shape, the following properties define the minimum necessary conditions for a valid mold design:

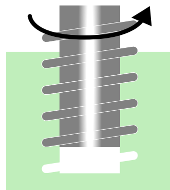
(C1) Surface partitioning: each portion of the cast object surface must correspond to one mold piece;

(C2) Mold (dis)assemblability: each mold component must admit an assembly and disassembly path. This property considers both the geometry of the cast surface and the configuration of the mold pieces participating in the assembly.

These two properties, while very general, already characterize the main challenges and solution methodologies for the state-of-the-art in rigid molds generation.

The first condition (C1) refers to the fact that generating a rigid mold assembly requires the definition of a surface partitioning such that each part is associated with its relative mold piece. This concept of surface decomposition is shared (to the best of our knowledge) by all the previous works from the state-of-the-art and offers a link between rigid molds design and the long-standing literature about surface segmentation and object decomposition.

The second condition (C2) encompasses a whole new set of problems peculiar to molding: each mold component must be detachable from the cast object, and the mold must globally admit an assembly and disassembly sequence. In general, enforcing the existence of a (dis)assembly sequence, as well as a cast detachment path, is not trivial. In particular, the study of rigid parts assemblies and their design is currently an active and open research topic, as demonstrated by the rich analysis and discussion provided in the survey from Wang et al. [WSP21]. Moreover, the verification or planning of a valid mold detachment path is a very complex geometric problem when considered in its general form. In fact, complex molds may require the use of non-linear and articulated extraction paths that are difficult to model and optimize, especially when the search space is limited by other constraints (Figure 6). An example of a design that would require defining non-linear extraction paths is given by objects with inner threaded cavities.



In general, as mentioned in Section 3, methods for rigid mold generation often use a less general definition of the constraint in (C2), restricting the space of extraction paths to simple linear paths. When considering the generation of rigid molds in which only linear extraction processes are admitted, the constraint (C2) can be made stronger and split into two:

(C2.1) Height-fieldness: The portion of surface cast by a mold component must satisfy the height field constraint for a given direction d. This implies that each mold component can be detached from the cast following a linear translation along the direction d.

(C2.2) Linear (dis)assembly: Each mold component must admit a linear assembly sequence along its assigned direction d, without intersecting either the cast or other mold components.

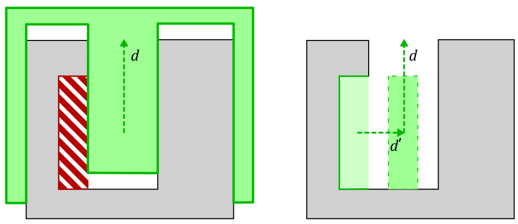


Figure 6: *While the red highlighted area (left) would locally be a height-field with respect to* d' *(C2.1), there is no global mold decomposition that would admit a linear assembly (C2.2). Allowing for a piece-wise linear assembly sequence would make such decomposition valid; at the cost of a greater computational and fabrication machinery complexity.*

These two strengthened constraints, along with constraint (C1), reduce the problem of generating a rigid mold for an input object to finding an association between parts of the object and extraction directions, satisfying at the same time the height field and assemblability constraints. Moreover, reducing the space of extraction paths to linear paths favors the use of the resulting molds in actual industrial applications. Indeed, designing a casting system for molds with only linear assembly sequences requires much simpler machinery and thus reduces lead times and production costs.

In the following, we will provide a comprehensive overview of the leading research contributions that address rigid mold design problems. In particular, we will first discuss some of the early works that introduced the foundational concepts and methodologies that established the base for all subsequent research on mold generation. Then, the discussion will proceed by categorizing the main contributions based on their target rigid mold technologies. Finally, within each of these categories, we will detail the various contributions and methodologies described in the state-of-the-art to provide the reader with a complete view of the available tools and techniques to tackle existing and novel design challenges in the field.

The first important step towards developing a scientific and systematic approach to mold design is to define the criteria that should drive the decision-making behind the design process. Ravi and Srinivasan [RS90] proposed a list of nine decision criteria for optimizing mold design, including the minimization of undercut areas, parting line flatness, parts draft, mold draw, and so on. The decision criteria are formalized to be implemented in algorithms to automate the optimization and assessment of mold design. Early works by Hui et al. [HT92, Hui97] posed the basis for the definition of automatic algorithms for mold design. In their works, the authors formalize the concept of local and global accessibility of a surface with regard to a direction and describe heuristic search strategies to determine a set of extraction directions and parting surfaces for relatively simple objects. Early approaches [CCW93, NFF*97, FFN99, ZZL10] relied on algorithms for the detection of potential undercut features on the target shape to guide the search for the optimal mold parting directions. Unfortunately, such algorithms were mainly tailored to simple shapes and representations such as NURBS and B-Reps and do not perform well with free-form geometries. Later, Khardekar et al. [KBM06] proposed a GPU-based algorithm to display and analyze undercuts by rendering the model from a wide set of potential casting directions. The efficiency of the proposed algorithm allows real-time undercut and draft analysis for simple shapes. However, the running times reported by the authors for the detection of undercut-free directions can get in the order of minutes for shapes of medium complexity.

Another important aspect of mold generation is optimizing the parting line that separates the different mold regions. Li et al. [LML09] proposed an approach for the optimization of parting lines, given an input parting direction, for two-piece molds (Figure 7). The proposed algorithm defines a triangle band within which the parting line must lie, analyzing the surface accessibility for the input parting direction. The triangle band is then used to derive a skeleton that drives the topological optimization of the parting line. Afterward, the parting line is further refined to ensure its smoothness. Finally, the authors propose a surface deformation strategy to remove small undercuts (red area in Figure 7, left) resulting from the parting direction choice. The algorithm iteratively updates the positions of the vertices of undercut triangles to force them to assume feasible normals.

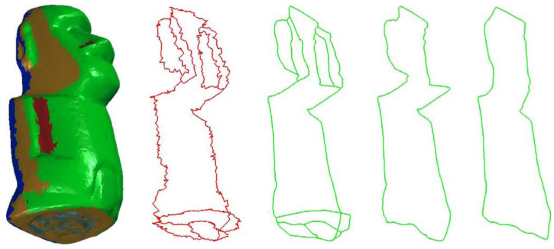


Figure 7: *The parting line optimization method proposed by Li et al. [LML09]. Given a parting direction (and its opposite), the method starts by identifying the set of triangles accessible from both directions (yellow band). The initial parting line topology is defined from the skeleton of the accessible triangles band (red polyline). Finally, the parting line is smoothed and its topology simplified to partition the object in two parts (green polyline).*

4.1. Two-piece molds

The generation of *two-piece rigid molds* is a very challenging and interesting problem, both from the computational geometry and the industrial point of view. Two-piece rigid molds are a standard approach for mass production facilities, in which fabrication machinery such as plastic injection molding machines are designed to support mainly this kind of operation. The problem of defining a two-piece rigid mold for an arbitrarily complex object can be seen as a decomposition problem in which the output parts must satisfy the double-height field constraint. A shape is called a double-height field when it is possible to partition it into two connected parts, such that both are height fields, one with respect to a direction d and the other to its opposite $-d$.

Chakraborty et al. [CR09] proposed one of the first completely automatic approaches for the generation of two-piece rigid molds. The method builds on previous works on parting direction analysis [CCW93, FFN99, NFF*97] and parting line optimization [FNF02]. The proposed algorithm starts by sampling a set of candidate parting directions, driven by the non-convex features of the shape. Then, for each direction, it evaluates the induced parting lines and selects the one that maximizes the parting line flatness and minimizes the presence of undercuts as the best parting direction. If some undercut persists after selecting the optimal parting direction, the method allows the definition of side cores. Side cores are special moving parts within a mold piece that can be independently extracted along a dedicated direction. However, the authors do not detail the generation of side cores and their implication for global mold assemblability. Finally, the method is not suited for complex free-form shapes, for which it is often impossible to define a double-height field partitioning without decomposing the object into multiple parts.

Babaei et al. proposed FabSquare [BRL*17], an interactive tool for designing two-piece molds for injection molding of UV curing

Figure 8: *Nakashima et al. [NAI*18] propose an interactive method for the generation of two-piece molds for thin-shell objects. The decomposition shown on the left is generated by the solver, integrating user defined constraints (left). The method then generates a two-piece mold for each part (middle). Finally, the parts can be assembled together to form the target shape (right).*

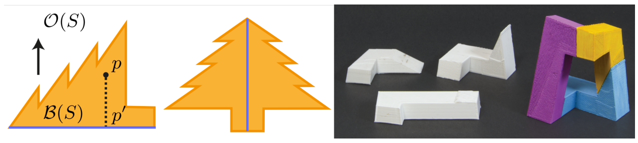


Figure 9: *Pyramidality implies the double height field property by definition. This makes pyramidal decompositions [HLZCO14] valid as height field decompositions for the generation of two piece molds.*

resins. In their work the authors analyze and evaluate the fabrication pipeline, testing different kinds of materials. They propose a CAD-like editor for mold design, which allows the user to automatically orient the target shape with respect to the parting direction, maximizing the accessible surface area. However, when undercuts are still present on the model after this optimization, the editor only alerts the user that the mold is not valid, making the method suitable only for shapes of limited complexity or specifically designed with the accessibility constraints in mind.

In CoreCavity [NAI*18], the authors propose an interactive tool for the decomposition of thin-shell objects into parts that can be independently created using a two-piece mold (Figure 8). The tool consists of an automatic algorithm for the decomposition of the object into a small number of parts that can be cast using two-piece rigid molds and an interface for user interaction that allows the designer to guide the decomposition solver towards better solutions with respect to aesthetics and number of parts. The decomposition produced by CoreCavity is based on a modified version of variational shape approximation (VSA) [CSAD04], which takes into account a moldability criterion. The output of this algorithm is a set of *as moldable as possible* regions that partition the entire object. The next step of the algorithm is the computation of parting lines for each of the object parts. The authors propose an approach based on *graph cut* [BVZ01, DOIB12], inspired by the method of Herholz et al. [HMA15] for approximated height field decomposition. One important thing to note here is that these methods allow violations of the height field constraints in their decompositions to accommodate the geometric complexity of free-form shapes. These violations will be later repaired by modifying the object shape to remove existing undercuts using an ARAP [SA07] morphing scheme.

Stein et al. [SJG19] proposed another interactive tool for the creation of shapes that can be created using a two-piece mold with a single planar cut. Given the very restrictive constraints of rigid molding coupled with the requirement of having a single planar cut, arbitrary shapes are commonly unfeasible under these assumptions. The authors focus then on shape design, providing the user with a tool to deform and adapt a shape to the constraints imposed by fabrication. In their method, the authors propose a 'castability' energy that quantifies how far the shape is from being castable using a two-piece mold, coupled with a gradient-based deformation scheme based on ARAP [SA07] that, similarly to previous works, allows for mesh deformation [HMA15,NAI*18,MLS*18]. Furthermore, the authors present a tool that allows the user to control the output design by specifying the cutting plane parameters and eventually constraining the gradient-based optimization. The authors demonstrate the efficacy of their design tool on relatively simple shapes. However, the method suffers from severe limitations. It can fail on shapes that feature very high violations of the moldability conditions. Since the method relies on a single planar cut for the generated mold, it cannot handle shapes with complex topologies, such as knots.

While practical advantages and immediate application to most industrial contexts are obvious, two-piece molds cannot be applied directly to fabricating complex shapes. In general, two-piece rigid molding is a highly constrained process, and usually the target shape is designed to comply with the constraint of the manufacturing process. An alternative approach consists of decomposing the target shape into several parts and arranging them in two-piece molds. The final object is reassembled once each piece has been cast. Similar to the work in Nakashima et al. [NAI*18], Alderighi et al. [AMB*21] have recently explored this idea to assemble an automatic pipeline to split a target shape into multiple parts that are cast individually using two-piece rigid molds and then arranged to form the target shape. Following this idea, to successfully cast the different components, each part should be represented as a *double height field* for a specified casting direction (and its opposite). The authors propose an algorithm that uses a *graph cut* approach to derive a volumetric decomposition guaranteeing that each part is castable using a two-piece mold, and the whole object can be assembled afterward.

4.2. Other height field decompositions

Given the importance of height field decomposition as a tool for many computational and fabrication applications, several works that do not explicitly address molding can be considered part of this state-of-the-art.

Among these, the works from Hu et al. [HLZCO14] on approximate pyramidal shape decomposition and from Muntoni et al. [MLS*18] on axis-aligned height field decomposition for subtractive manufacturing could be directly applied to the design of single-piece and two-piece rigid molds, respectively. The first method proposes an algorithm for the decomposition of an object in parts that are pyramidal. A shape is defined as pyramidal if it has a flat base, and the rest of its surface can be represented as a height field with respect to a direction orthogonal to the base (Figure 9). It is straightforward to see how *pyramidality* implies the double-height field property and thus the fabricability of parts using two-piece molds. Furthermore, the existence of a flat base enables the manufacturing using a single-piece rigid mold. However, optimizing for exact pyramidal decomposition is proved to be NP-Hard [FM01]. The solution proposed by the authors outputs approximated pyramidal parts, in which the height field constraint can admit limited violations. Muntoni et al. [MLS*18] later proposed a method for the decomposition of a shape in parts that are also pyramidal but can control the number of constraint violations tolerated in the solution. Their method searches for a decomposition of the volume using non-overlapping axis-aligned pyramidal blocks. The method optimizes for a minimal covering of the surface starting from a dense set of height field blocks. It solves an integer linear problem and finally fixes possible overlaps between blocks.

While these methods can be directly applied to the generation of two-piece rigid molds, the problem they solve is over-constrained with respect to the problem of finding a valid double-height field decomposition. This results in solutions with a high number of parts, making them less amenable for practical use in molding.

Despite solving a different practical problem, the approach of Jacobson [Jac17] tackles a similar class of problem. This paper generalizes the nesting of solid objects in the style of Matryoshka dolls. The method optimizes the placement of nested objects, ensuring the extractability of each internal copy without colliding with the external copy. Similar to other methods, the computational Matryoshka uses a GPU-accelerated evaluation of *height field visibility*.

4.3. Multi-piece molds

Multi-piece molding refers to the use of complex molds for casting where the mold enclosure is composed of multiple (namely more than two) pieces, each one having an associated extraction direction. The use of multiple mold pieces, and their respective parting directions, makes it possible to generate valid molds for complex shapes that would otherwise require decomposition in multiple two-piece moldable parts. However, designing such molds is a complex and time-consuming task that requires highly specialized expertise. While most works in the mold design literature focus on two-piece molding, a few contributions study the automation of multi-piece mold design.

Priyadarshi et al. [PG04] proposed the first complete approach for the automatic design of multi-piece permanent molds. The method they describe builds on a global accessibility test to define facet accessibility with respect to a direction d. Once the accessibility information is computed, the algorithm collects the connected feasible regions for each direction. These will define the candidate mold pieces. The search for (near-)optimal mold decompositions is driven by an algorithm that mixes a greedy incremental approach and a branch-and-bound exhaustive search to look for better results. The greedy approach is used to obtain a baseline solution. At each step, the mold piece with the biggest uncovered area is chosen until a surface covering is reached. The algorithm then explores the solution tree looking for better results, reverting to the greedy solution if the branch-and-bound search does not converge within a user-selected time budget. Finally, the authors describe the automatic generation of the parting surfaces and mold. Similarly, Lin et al. [LQ14] proposed an algorithm for the automatic design of multi-piece permanent molds. The algorithm heuristically selects a small set of candidate directions driven by surface normals and object features (available from the CAD design). The method defines a global ray accessibility test for mold pieces evaluation. For undercuts that are not accessible from any of these directions, they propose a local accessibility estimation based on V-Maps [GWT94]. The authors then present a set of heuristic criteria for extracting a minimal number of mold pieces to cover the surface, optimizing the number of mold pieces and fabricability metrics.

Figure 10: *An example demonstrating the multi-piece molds generation algorithm from Herholz et al. [HMA15]. From left to right: the output of the method when no undercut constraint violations are allowed, the regularized output with surface deformation, the resin cast, and the mold pieces.*

While the methods cited above presented valuable advancements in permanent mold design, they still suffer from some significant limitations, especially when considering their application to free-form shapes that were not explicitly designed for being manufactured using molding techniques. Both methods rely on the assumption that parts are relatively simple or characterized by well-defined features for which it is *easy* to derive valid parting directions. Moreover, both methods were designed and tested only on small faceted models, with support for NURBS free-form surfaces, but not for complex free-form meshes with large triangle counts and complex topological and geometrical features.

In general, the combinatorial nature of the problem makes it difficult to design effective and efficient algorithms based on simple geometric reasoning and heuristics. Therefore modern approaches targeting complex shapes mainly rely on energy minimization or combinatorial optimization approaches that allow for an efficient exploration of the solution space.

An example of this is the work by Herholz et al. [HMA15], which presents a novel approach for the approximation of free-form shapes with height fields applied to the automation of multi-piece mold design. The authors propose a method to automatically select the optimal parting directions and parting lines layout solving an energy minimization problem. The problem is expressed as a multi-labeling problem using the graph cut optimization framework, in which labels represent parting directions and the graph nodes represent the mesh faces. The optimized energy function penalizes solutions over the parting lines length and number of labels being used, while enforcing moldability constraints in a hard way. To keep the resulting mold complexity low, the method allows some violations in the height field constraint during the accessibil-

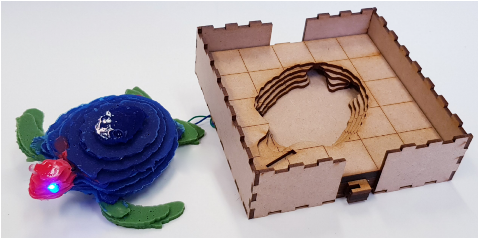


Figure 11: *StackMold [VLAR19] describes an interactive system for the design of multi-stage sacrificial molds based on the concept of shape deposition manufacturing.*

ity evaluation for each direction. Finally, possible height field constraint violations are fixed following an ARAP [SA07] deformation scheme. While the approach can handle free-form shapes which would fail with previous methods, it might produce highly fragmented molds that are not usable in practice. Moreover, due to the limits imposed by the height field constraints, the method could require significant shape deformations. Figure 10 shows an example of the method output on a complex shape. The effects of undercut constraints violations and of shape deformation are shown by the two segmentations on the left. The first image from the left reports the output of the method when no violations are tolerated, while the second image shows the regularized solution whose surface has been deformed to account for the undercuts violations.

4.4. Multi-stage molds

We use the term *multi-stage molds* to refer to a special kind of multi-piece molds in which, additionally, a concept of the stage is introduced. A stage refers to a single casting operation and mold configuration. As the term implies, multi-stage molds feature multiple casting operations performed at different times. The fabrication process is then performed by either changing parts of the mold between one stage and the next, or by extracting the partial casting and positioning it in a completely different mold [GL02].

This concept has been investigated in the preliminary work on rotary-platen multi-shot molds by Li et al. [LG04]. In their paper, the authors provide an algorithm for the automatic design of multi-shot molds for two-materials objects, with limited support for undercuts generation using side cores. While their proposed method is only able to handle two-materials parts and has been demonstrated only on simple mechanical shapes, the formulation proposed by the authors and its algorithmic framework could pose a valuable starting point for the development of a fully automatic pipeline generalizing to free-form and multi-materials objects. This fabrication technique is particularly interesting since, by design, it enables the reproduction of objects utilizing multiple materials, varying in color and mechanical properties. In addition, multi-stage molding enables direct fabrication of complex articulated objects, avoiding costly and error-prone manual assembly of parts by performing the assembly implicitly, using distinct casting operations performed at different stages of the mold [KKG02, PGG*07, BLFG07, PG09].

More recently Valkeneers et al. [VLAR19] proposed StackMold, an interactive system for designing multi-stage molds targeting the fabrication of multi-material objects with embedded electronics. The solution proposed by the authors is inspired by the concepts of multi-stage molding and shape deposition manufacturing (SDM) [MPR*94]. SDM is a layered manufacturing process in which the target shape and the geometry of its (usually sacrificial) mold are partitioned into layers. The process enables building up the object and its mold part by part. This allows the generation of multi-material objects and also the embedding of parts, such as electronics, in their final shape. In StackMold, the authors propose an interactive solution for the design of multi-stage shape deposition molds for which the sacrificial mold layers are fabricated by laser cutting uniform thickness cardboard (Figure 11). Given an input shape, the user can specify different material components and parts embedding and, most importantly, must manually select a casting orientation. The algorithm will subsequently decompose the object and mold shape into uniform layers (orthogonal to the casting direction) and optimize a mold stacking and material deposition sequence to minimize the number of casting steps. This optimization step is performed as a search defined on a tree data structure that holds information about the ordering dependencies between the slices of the shape and the mold layers. Having a fixed casting direction, combined with the fact that the ordering dependencies are induced by the material specifications and shape layering, allows the algorithm to efficiently explore the dependency tree and find a feasible plan.

The automatic design of multi-stage molds for arbitrary shapes remains a challenging open problem. All the contributions presented above exploit specific assumptions to strongly limit the solution space and render the problem more 'tractable' from an optimization point of view. The high complexity behind multi-stage mold design comes from the need to simultaneously tackle the problems of finding an optimal molding decomposition of an object and optimizing the assembly sequence of the object and multi-stage mold parts. These problems are still considered open research topics and are commonly being investigated separately. Wang et al. [WSP21] presented an in-depth analysis of the computational design of assemblies with rigid parts; however combining the assemblability and moldability problems in a single optimization framework still needs further investigation.

The potentialities of multi-stage molding, coupled with the inherent research challenges that it raises, make it a promising topic for future research in the generation of complex molds.

5. Flexible molds design

Flexible molds are a fabrication technology commonly used in non-industrial contexts, such as cinema prop-making, culinary art, custom jewelry, and sculpting. The use of flexible molds is particularly effective for reproducing shapes with complex geometric features, such as high-frequency surface details that are difficult to capture using height field segmentation (the main requirement for rigid mold design). However, the advantages of using a flexible mold come with increased complexity associated with the handling of the mold during the extraction and assembly processes. This makes flexible molds less suited for mass production manufacturing processes, in which adapting automated manufacturing machinery to handle soft, deformable molds would be unpractical. For these rea-

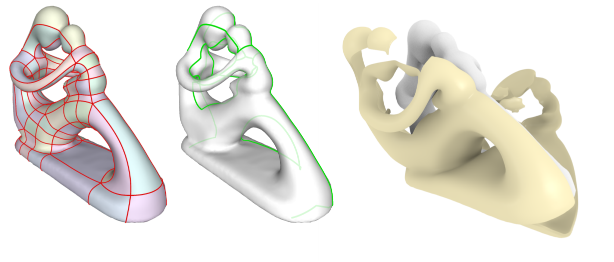


Figure 12: *FlexMolds [MPBC16] presents an automatic method for the generation of flexible molds defined as a thin layer of elastic material. The method automatically optimizes the cut layout required to open the mold and release the cast, based on a physically-based simulation of the extraction process.*

Figure 13: *Alderighi et al. [AMG*19] propose an algorithm for the automatic generation of flexible, composite molds, in which the soft silicone part is supported by a hard plastic shell. The approach is based on a volumetric analysis of the mold surrounding the object to automatically generate the set of cuts needed to safely extract the cast object. On the left, the cut layout is composed of a set of internal cuts (red surfaces) and a parting surface (blue surface) separating the mold in the two pieces. On the right, the generated mold with one of the pieces opened to highlight the cut layout.*

sons, flexible molds are commonly used for artisanal and small volume manufacturing. Arguably, these details made flexible molding a less developed research field for the engineering research community, leading to scarcity of scientific literature on the topic. It is only recently that flexible molds design has gained some interest from the computational manufacturing research community.

The most important advantage of using flexible molds is the fact that the elastic behavior of the mold allows the relaxation of the height field constraint (Section 4 C2.1). This makes the technique robust for input shapes with very complex geometry, but also makes the solution space larger with respect to rigid molds. Consequently, there is a need for novel computational approaches that can efficiently and effectively 'navigate' this large design space.

The first contribution that advanced an original solution to the automatic computational design of flexible molds is *FlexMolds* [MPBC16]. In this work, the authors proposed an algorithm for the automatic design of single-piece, thin, flexible molds whose cut design is driven by a physically-based simulation of the extraction process. The novelty of the method comes both from the manufacturing process and the mold design algorithm. From the manufacturing standpoint, the key idea is to use a single-piece mold made of a thin layer of flexible material that has the shape of the desired cast. This mold can be opened using a set of cuts in the material whose geometry is automatically optimized by the proposed algorithm. The cut layout optimization algorithm works in a greedy bottom-up manner. It starts from a dense cut layout defined as a quadrilateral patching of the object surface (Figure 12, left). The assumption is that such a dense cut layout defines an extractable mold. Each patch represents a separate mold piece that can be extracted individually, leveraging the thin-shell nature of the mold. The rationale is then to close one cut at a time from this layout iteratively. The choice of the cut to close is guided by a physically-based simulation of the extraction process that applies a set of outward forces to the elements of the thin-shell geometry and verifies that for the given cut layout, there exists an extraction path in a finite time. Such simulation is also used to approximate the strain imposed on the mold geometry by the detaching process. The final optimized cut layout (Figure 12, middle) defines the set of cuts in the mold (Figure 12, right).

Although this method can handle complex shapes, it also has some drawbacks, mainly related to its manufacturing process. First, the cuts in the thin-shell need to be manually sealed using silicone or similar means, making the molds prone to cast material leakage. Moreover, due to the thin-shell nature of the molds, large deformations can occur when casting large objects, for which the effect of the pressure of the casting material would not be negligible. Finally, the physical simulation that drives the optimization algorithm relies heavily on the thin-shell nature of the mold to avoid having to handle complex contact behavior and friction between mold pieces. These limitations would make it hard to adapt such a solution to the general case of molds made of a thick layer of material.

More recently, to overcome such limitations, Alderighi et al. [AMG*18] proposed a novel algorithm for the automatic generation of multi-piece silicone molds. Silicone molds mitigate the leakage and deformation problems from FlexMolds thanks to the use of thick mold parts that are sealed by the adhesion of large parting surfaces. The algorithm proposed by the authors enables the fabrication of multi-piece molds by casting silicone into 3D printed containers called Metamolds. The number of mold pieces, the geometry of the metamolds, and thus the parting surfaces needed to open the mold, are automatically defined. Metamolds work by solving a surface segmentation problem, similar to the one proposed for rigid mold design [HMA15, NAI*18]. The algorithm is based on surface visibility and a novel moldability score that reflects how difficult it would be to detach a flexible mold piece along a direction from the object's surface. The output surface decomposition associates a mold piece and an extraction direction to every surface element. An interesting novelty of the method is the introduction of the concept of cut membranes to handle topological features. Cut membranes are thin membranes in the 3D printed metamold geometry that will then become cuts in the silicone mold, allowing the extraction of parts that feature topological handles.

The concept of adding cut membranes in the metamold geometry, to define the cuts necessary to open the silicone mold and release the cast object, was further developed by Alderighi et al. [AMG*19] who proposed an algorithm for the automatic design of two-piece flexible molds (Figure 13). In this work, the authors propose a method for the volumetric analysis of the mold surrounding the cast object. The volumetric analysis is based on the concept of *escape paths*: a geometric approximation of the shortest path that a volume element would follow when pulled away from the cast object during mold extraction. The algorithm computes escape paths as shortest paths computed on the graph of the edges of a volumetric tessellation of the mold volume, connecting interior vertices of the tessellation to its external boundary. Given the set of escape paths, the proposed volumetric analysis then automatically determines the geometry of the cut membrane layout needed to open the mold (Figure 13, left). For each pair of adjacent vertices of the volumetric tessellation, a cut membrane should separate these vertices if their escape paths reach the external boundary passing on different sides of a portion of the cast object.

The simple geometric formulation provided by Alderighi et al. [AMG*19] allows to robustly generate a valid mold internal cut layout. This method is effective for objects with complex topological and geometrical features on which the previous state of the art would fail.

Figure 14: *Zhang et al. [ZFS*19] propose a method for the automatic generation of fabric formworks for free-form shapes. The method optimizes the fabric patterns that, once sewn together, will form the flexible formwork for the target shape.*

6. Alternative Molds

This section presents a brief outline of recent works on the computational design of concrete formworks and thermoforming processes. While not directly related to the previously discussed techniques, these are two common and industrially relevant fabrication techniques that leverage the use of molds.

6.1. Formworks

Formworks are a particular kind of mold used in architectural fabrication for casting concrete to create medium and large-sized elements such as slabs, pillars, and thin concrete shells. Given the impact of formworks on the design and resources cost in construction (representing up to 80% of the total costs of complex concrete elements), considerable research has been devoted to the application of digital fabrication processes and computational design in this context [JD21]. Some research contributions investigate the use of 3D printing of reduced impact materials, such as PVA and PLA, to generate (sub)millimeter thickness formworks for complex free-form elements [JBR*17, BLFS*20]. In Mesh-Mould [HL14], the authors propose a lattice-like, robotically generated structure that acts both as a formwork on which a viscous concrete mix is sprayed or poured and, after curing, as a reinforcement structure for the concrete panel. The industry further demonstrates the relevance of these research efforts, in which digital fabrication is integrated in complex formworks productions. Adapa [ada] industrializes the flexible mold approach [Sch15], providing 'easily' reconfigurable formworks for the fabrication of large doubly-curved concrete panels. FreeFAB [fre] uses instead a digital fabrication pipeline, exploiting large-scale 3D printing and CNC milling, to fabricate large formworks with complex designs out of reusable wax. These industrial applications of recent developments in the digital fabrication of formworks drastically reduce resource waste and lead times in the fabrication of large concrete panels. This report focuses on the computational design aspects involved in this process. In the case of fabric formworks, the mold is composed of a set of fabric panels that are sewn together and then filled with casting material. The weight of the material under gravity and the sewn patterns in the panels will take the formwork in the desired shape. In their work, Zhang et al. [ZFS*19] propose the first computational method to automate the design of fabric formworks for casting 3D free-form shapes (Figure 14). In particular, they propose a formulation to find the optimal fabric panel design and casting directions as an inverse design problem. The inverse problem is defined as an energy function minimization problem, in which the energy is computed as a combination of a set of metrics penalizing the final shape deviation and favoring formwork stability under suspension, and smoothness of the fabric seams. The problem variables are the fabric panels' shape, the casting direction, and the deformed target shape. The minimization is subject to a set of constraints that model the physical process of casting in the resulting fabric formwork. In particular, the constraint requires the system (the fabric panels filled with material under a given casting direction) to be in equilibrium. Equilibrium is computed by simulating the effects of fluid pressure on the fabric formwork. The method also accounts for the automatic placement of external supports, such as rods and strings, to help the fabric take the target shape. While the proposed process takes a first step towards automating fabric formworks design, it also has several limitations. Finding optimal solutions to the proposed inverse problem is computationally expensive. Thus the authors targeted only low-resolution meshes. The resolution not only affects the visual quality of the shape but, as reported by the authors, also hides the occurrence of artifacts due to the fabric wrinkling under compression (visible on the concrete reproduction in Figure 14, right). Moreover, the model ignores the rigidity induced by seams in the fabric leading to eventual discrepancies between the physical object and the simulation.

6.2. Thermoforming

Thermoforming is a fabrication method in which a sheet of material is heated to transition to a plastic state and then deformed to a target shape using a mold. This method has extensive use in industry for the production of low-cost everyday objects [Kle09]. While significantly different, this method shares notable similarities with geometric constraints in molding. As for rigid molding, this process requires the modeled shape to be a height field. As the produced object is composed of a thin layer of material, this production method tolerates small deformations during the extraction process.

The correct transfer of color properties on the produced shape is one of the major problems of this technique. Indeed, as the material deforms from a flat sheet, the printed image is distorted by the manufacturing process. To alleviate this effect, methods proposed by Zhang [ZTZ17], and by Schueller [SPG*16] use physically-based simulations to anticipate the introduced distortion in the flat image and correct it so that each pixel will fall in the correct position once deformed to the final 3D shape.

7. Industrial perspective

Molding has developed into the dominant industrial manufacturing method for a wide variety of materials and products such as metal die casting or plastics injection molding, to name prominent examples. At the same time, an ever-increasing specialization has taken place that adapted this process to new materials, new product shapes, or increased production speeds and volumes. This makes it impractical to give an exhaustive overview of all methods in current use; we will focus on the most industrially relevant examples. For a comprehensive listing, refer to the handbook by Bralla [Bra99].

A fundamental trade-off in molding exists between production efficiency (e.g., the rate at which parts can be fabricated with a given mold) and other criteria, such as the geometric or material complexity of the part. On one end of the spectrum, where geometric freedom and large part sizes are deciding factors, we find sacrificial molds, which are destroyed after a single use, (see Section 1.1) or formworks, which require elaborate assembly and disassembly after each use (see Section 6).

On the other end of the spectrum, where high production rates are paramount, we find permanent reusable molds used for the creation of metal parts with *die casting* and plastic parts in *injection molding*. While these techniques operate in repetitive cycles of mold closing, mold filling, mold opening, and part ejection, simple shapes with a uniform cross-sectional profile can be produced continuously using *extrusion molding*, where the material is pressed through suitably shaped holes.

Apart from these core molding methodologies, a wide variety of specialized variants exist. In the context of multi-stage molding (see Section 4.4), we name *n-k molding* (e.g., 2k, 4k, ...), where successive cavities are used to add *n* different material components to the same part, and *blow molding*, where a molded part is inflated through gas injection to achieve thin-walled products, such as containers and bottles. Sheet materials can be shaped by forcing them to conform to a mold, which is the case of *thermoforming* (see Section 6.2) and *vacuum forming* in the case of plastics and *deep drawing* and *hydroforming* for metals.

In its current form, industrial mold design relies heavily on personal experience and expertise to navigate the challenges and complexities of these methods. While cavity design is a central element in any of the aforementioned production methods, a variety of other design aspects need to be taken into account—and might become dominant—especially for large series production: (*i*) design for manufacturability of the product; (*ii*) material selection based on material characterization and rheology simulation; (*iii*) fill channel, heating and cooling system design based on flow and thermal simulations; (*iv*) ejection design and integration with molding and automation machinery; and (*v*) machine parameter setup based on the previous steps. Further mold design aspects, that might be taken into account are durability under long-term use, ease of maintenance and repair, ease of monitoring, and purely practical constraints, such as the capabilities of existing molding machines or the available space during transit or installation.

7.1. Showcasing injection molding

While injection molding constitutes one of the most important manufacturing methods for plastic products—with its corresponding market size on the order of 300 billion euros [FBI20]—it can also be used for the production of metal parts by injecting metal powder with binder material and subsequent debinding and sintering. The value chain for injection molding can be divided into three major stages—product design, mold making, and production—which are generally performed by different actors and companies. Product design is usually the domain of the eventual provider of the final product, who is aware of its design constraints and performance requirements [Mal10]. *Mold makers* are responsible for designing and fabricating suitable molds and molding tools, which are then used by *contract manufacturers* in conjunction with injection molding machines for series production. Important secondary players in this process are the providers of CAD, CAE, and CAM software tools for the creation and validation of product and mold designs, CNC machine manufacturers for the mold making, injection molding machine manufacturers for series production, and suppliers of raw materials (e.g., metal powders, polymers, etc.). It should be noted that the three stages are seldom executed in a sequential fashion and, depending on part complexity, multiple iterations might be necessary [KB18]: initial product designs that are inadequately tailored to mold-based manufacturing might require costly side actions or multi-stage production.

Looking at the mold making phase in detail—itself a market with an annual global production value of 30 billion euros [BSK*18]—the usual process is: (*i*) fill simulation, which determines the parameters for the injection process, gate locations for the material injection into the cavities, as well as cooling and heating requirements, (*ii*) mold design, which defines the shape and requirements for all parts of the molding tool, (*iii*) mold fabrication design, which converts the mold designs into machine instructions for the machinery used in mold making (e.g., CNC mills, electrical discharge machines, metal 3D printers) and post-processing (e.g., polishing, etching), (*iv*) mold fabrication, (*v*) tool assembly, (*vi*) inspection and production tests [Kaz16].

Computational mold design, as discussed in this report, mainly addresses challenges in process step (*ii*), namely the creation of cavity geometry. In industry, such mold design tasks are part of the larger tool design, which is generally performed by experts using CAD tools, ranging from general purpose CAD solutions (e.g., Dassault Systèmes Solidwork) to specialized mold-specific tools (e.g., 3D Systems Cimatron: Mold, Hexagon VISI Mold) to full-fledged product lifecycle management systems (e.g. Siemens NX, Dassault Systèmes CATIA) and their mold design extensions (e.g., NX Mold Wizard, CATIA Mold and Tooling). Depending on the tool complexity, such design tasks can range from several person hours—for simple two-piece molding tools, whose mold making costs start at 2k euros—to several person weeks—for high-end tools that can cost several 100k euros.

Table 1: *Overview of the methods described in this state-of-the-art report.*
**: The method allows for the addition of side cores to handle simple undercuts.*

		Paper	Shape Deformation ✦ Yes ✧ No	✦ Automatic ✧ Supervised	Object Parts (#)	Mold Pieces (#)	Mold Handling	Shape Complexity
Rigid	Two-Piece	[CR09]	✧	✦	1	2*	easy	★
		[BRL*17]	✧	✦	1	2	easy	★
		[NAI*18]	✦	✧	≥ 1	2	easy	★★
		[SJG19]	✦	✧	1	2	easy	★
		[AMB*21]	✧	✦	≥ 1	2	easy	★★★
	Multi-Piece	[PG04]	✧	✦	1	≥ 2	medium	★
		[LQ14]	✧	✦	1	≥ 2	medium	★
		[HMA15]	✦	✦	1	≥ 2	medium	★★
	Multi-Stage	[KKG02]	✧	✦	≥ 1	≥ 2	complex	★★
		[PGG*07]	✧	✦	≥ 1	≥ 2	complex	★★
		[PG09]	✧	✦	≥ 1	≥ 2	complex	★★★
		[VLAR19]	✧	✦	≥ 1	≥ 2	complex	★★
Flexible		[MPBC16]	✧	✦	1	1	complex	★★★
		[AMG*18]	✧	✦	1	≥ 2	medium	★★★
		[ZFS*19]	✦	✦	1	≥ 2	complex	★★
		[AMG*19]	✧	✦	1	2	medium	★★★★

Such manual effort would benefit from automated mold design tools, and recent research efforts provided preliminary results for a variety of challenges: parting direction identification [YM18], parting line generation [HHZL21], and side-action design [HPH*21]. Comprehensive automated tool design is still an open problem offering a wide variety of research avenues, such as parting surface design, interlock design to prevent lateral movement, channel and gate design, selection of standard parts, cooling system design, and many more. Treating these design aspects jointly to achieve optimal cavity layout would constitute the subsequent major step.

Apart from these research questions focused on mold design itself, a tight integration with the preceding flow simulation methods and succeeding CAM design [SMGMFL18] presents research opportunities that allow designing according to optimality criteria that take production process parameters and mold fabrication effort into account.

Taking more advanced molding processes into account, such as the aforementioned blow molding or multi-component injection molding, add novel simulation and geometric challenges.

8. Conclusions

Molding as a manufacturing technique has been extremely popular because of its effectiveness, scalability, and versatility in employed materials. Despite advancements described in this survey, designing high-quality molds is still highly challenging due to the many criteria to take into account. These include the geometry of the cast object, the choice of parting lines and directions, the layout of gates, runners, and sprues, the employed casting material, and other manufacturing parameters that influence the performance of a mold.

In this report, we focused on automatic methods for mold design, with contributions from a geometric perspective. Table 1 provides an overview of the described methods, classified according to the molding technique they address (two-piece, multi-piece rigid molds, multi-stage molds, and flexible molds). Each row shows a single research paper, and the columns summarize the main characteristics of the proposed mold design technique. The last two columns of the table report two qualitative measures describing the complexity of the generated molds, their handling, and the complexity of the manufactured shapes. *Shape complexity*, in particular, measures the complexity of the topological and geometric details of the results shown in the respective research papers. We expressed it as a relative measure to highlight a ranking between the cited approaches.

The last two columns show a correlation between the complexity of the designed molds and the complexity of the manufacturable shapes using the respective mold design approach. However, the methods proposed by Nakashima et al. [NAI*18], and Alderighi et al. [AMB*21] appear as outliers in this trend. In fact, these two methods exploit the decomposition of the input shapes into multiple parts for which it is possible to define valid two-piece rigid molds. This approach allows the fabrication of complex shapes that would otherwise be impossible to reproduce using two-piece or multi-piece rigid molds. However, this design choice comes with the disadvantage of requiring an assembly process of the object as a post-processing step. This assembly step can be detrimental to both manufacturing efficiency and final assembly robustness.

The table also reports works from the *multi-stage* category as methods in which the object is composed of multiple parts. However, in these cases, the parts assembly phase is part of the molding strategy in which different stages often correspond to different materials or different parts of the assembly. In this way, the actual assembly process is implicitly handled by the molding process itself, which, as a consequence, can become particularly complex.

Another interesting detail that is evident from Table 1 is the focus, for nearly every approach reported, on the automation of the mold design process. While mold design automation is undoubtedly an exciting research challenge, we also believe that integrating user interaction in the computational mold design process could have an even more significant impact. The complexity of the design process and the variety of quantitative and qualitative factors impacting the quality of a mold make interactive design tools particularly interesting. Among the works listed in the table, only two [NAI*18, SJG19] allow direct user control over the aesthetics and design choices of the mold generation process.

As an outlook, we believe the following directions will be both highly relevant in practice and also pose interesting research challenges:

Automatize Most state-of-the-art methods for automatic mold design target hobbyists and limited scale production. However, we believe that there are exciting opportunities for some of the presented technologies to evolve into tools for large-scale industrial scenarios.

User-Interaction Current automatic mold design algorithms usually require long processing times, and they are still far from supporting interactive optimization. However, computational design contexts demand tools to explore the space of feasible molds in a semi-interactive way to understand the implications of different design decisions.

Bridging different representations While current tools for injection molding design are almost exclusively CAD-based, the new generation of computational design tools should integrate processing with alternative representations, such as those coming from scanning-based reverse engineering or topology optimization-based generative design methods. Integrating these formats with existing and novel design methodologies poses further research challenges.

Integrate A plethora of aspects might influence the performance of a mold, such as cavity and parting line design, interlock design to prevent lateral movement, channel and gate design, selection of standard parts, cooling system design, and many more. A modular, collaborative platform that allows integrating various simulation tools and makes them 'differentiable' or otherwise suited for more automatic design would be appealing.

Design for manufacturability As a long-term vision, we believe that modern product design tools should integrate manufacturability into the loop. This would allow the product designer to make informed judgments regarding product features, tooling feasibility, and cost. Alternatively, tool design considerations can be used in automated product design [WXM*20].

A public dataset Mold design is a significantly expensive phase of product design. Hence, most industries do not distribute the geometry and design of the molds used. However, this would represent an invaluable dataset that can be used to assess algorithms or to train algorithms.

References

[ada] Adapa adaptive moulds. https://adapamoulds.com/. Accessed: 2022-01-20. 11

[AMB*21] ALDERIGHI T., MALOMO L., BICKEL B., CIGNONI P., PIETRONI N.: Volume decomposition for two-piece rigid casting. *ACM Trans. Graph. 40*, 6 (Dec. 2021). URL: https://doi.org/10.1145/3478513.3480555, doi:10.1145/3478513.3480555. 7, 12, 14

[AMG*18] ALDERIGHI T., MALOMO L., GIORGI D., PIETRONI N., BICKEL B., CIGNONI P.: Metamolds: Computational design of silicone molds. *ACM Trans. Graph. 37*, 4 (July 2018), 136:1–136:13. URL: http://doi.acm.org/10.1145/3197517.3201381, doi:10.1145/3197517.3201381. 4, 5, 10, 12

[AMG*19] ALDERIGHI T., MALOMO L., GIORGI D., BICKEL B., CIGNONI P., PIETRONI N.: Volume-aware design of composite molds. *ACM Trans. Graph. 38*, 4 (July 2019), 110:1–110:12. URL: http://doi.acm.org/10.1145/3306346.3322981, doi:10.1145/3306346.3322981. 5, 10, 11, 12

[Aut21] AUTODESK: Moldflow, 2021. URL: https://www.autodesk.com/products/moldflow/overview. 2

[BLFG07] BANERJEE A. G., LI X., FOWLER G., GUPTA S. K.: Incorporating manufacturability considerations during design of injection molded multi-material objects. *Research in Engineering Design 17*, 4 (2007), 207–231. URL: https://doi.org/10.1007/s00163-007-0027-9, doi:10.1007/s00163-007-0027-9. 9

[BLFS*20] BURGER J., LLORET-FRITSCHI E., SCOTTO F., DEMOULIN T., GEBHARD L., MATA-FALCÓN J., GRAMAZIO F., KOHLER M., FLATT R. J.: Eggshell: Ultra-thin three-dimensional printed formwork for concrete structures. *3D Printing and Additive Manufacturing 7*, 2 (2020), 48–59. URL: https://doi.org/10.1089/3dp.2019.0197, arXiv:https://doi.org/10.1089/3dp.2019.0197, doi:10.1089/3dp.2019.0197. 11

[BOP10] BRUCKNER T., OAT Z., PROCOPIO R.: *Pop sculpture*. Watson-Guptill Publications, 2010. 1

[Bra99] BRALLA J. G. (Ed.): *Design for manufacturability handbook*, 2 ed. McGraw-Hill, 1999. URL: https://www.accessengineeringlibrary.com/content/book/9780070071391. 12

[BRL*17] BABAEI V., RAMOS J., LU Y., WEBSTER G., MATUSIK W.: Fabsquare: Fabricating photopolymer objects by mold 3d printing and uv curing. *IEEE Comput. Graph. Appl. 38*, 3 (May 2017), 34–42. URL: https://doi.org/10.1109/MCG.2017.37, doi:10.1109/MCG.2017.37. 6, 12

[BSK*18] BOOS W., SALMEN M., KELZENBERG C., STARK M., SCHULTES T., GRABERG T.: *World of Tooling 2018*, 1 ed. WBA Aachener Werkzeugbau Akademie GmbH, Laboratory for Machine Tools and Production Engineering(WZL) of RWTH Aachen University, Dec. 2018. 13

[BVZ01] BOYKOV Y., VEKSLER O., ZABIH R.: Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence 23*, 11 (2001), 1222–1239. doi:10.1109/34.969114. 7

[Cam95] CAMPBELL J.: Review of fluidity concepts in casting. *Cast Metals 7* (01 1995), 227–237. doi:10.1080/09534962.1995.11819183. 1

[Cam15] CAMPBELL J.: *Complete Casting Handbook: Metal Casting Processes, Metallurgy, Techniques and Design: Second Edition*. Butterworth-Heinemann, 01 2015. doi:10.1016/C2014-0-01548-1. 2

[CCL*05] CHEAH C. M., CHUA C., LEE C., FENG C., TOTONG K.: Rapid prototyping and tooling techniques: a review of applications for rapid investment casting. *The International Journal of Advanced Manufacturing Technology 25*, 3 (2005), 308–320. URL: https://doi.org/10.1007/s00170-003-1840-6, doi:10.1007/s00170-003-1840-6. 2

[CCW93] CHEN L.-L., CHOU S.-Y., WOO T. C.: Parting directions for mould and die design. *Computer-Aided Design 25*, 12 (1993), 762–768. 6

[CR09] CHAKRABORTY P., REDDY N. V.: Automatic determination of parting directions, parting lines and surfaces for two-piece permanent molds. *Journal of Materials Processing Technology 209*, 5 (2009), 2464–2476. URL: http://www.sciencedirect.com/science/article/pii/S0924013608004883, doi:10.1016/j.jmatprotec.2008.05.051. 6, 12

[CSAD04] COHEN-STEINER D., ALLIEZ P., DESBRUN M.: Variational shape approximation. *ACM Trans. Graph. 23*, 3 (Aug. 2004), 905–914. 7

[CSC21] CORETECH SYSTEM CO. L.: Moldex3d, 2021. URL: https://www.autodesk.com/products/moldflow/overview. 2

[Dal09] DALL'ORTO G.: Athens - stoá of attalus museum - moulds, 2009. URL: https://commons.wikimedia.org/wiki/File:3312_-_Athens_-_StoÃă_of_Attalus_Museum_-_Moulds_-_Photo_by_Giovanni_Dall%27Orto,_Nov_9_2009.jpg. 1

[DOIB12] DELONG A., OSOKIN A., ISACK H. N., BOYKOV Y.: Fast approximate energy minimization with label costs. *International journal of computer vision 96*, 1 (2012), 1–27. doi:10.1007/s11263-011-0437-z. 7

[FBI20] Injection Molded Plastics Market, 2020. URL: https://www.fortunebusinessinsights.com/injection-molded-plastics-market-101970. 13

[FFN99] FU M., FUH J., NEE A.: Generation of optimal parting direction based on undercut features in injection molded parts. *IIE Transactions 31*, 10 (1999), 947–955. 6

[FFN04] FUH J. Y., FU M. W., NEE A. Y. C.: *Computer-aided injection mold design and manufacture*. CRC Press, 2004. 2

[FM01] FEKETE S. P., MITCHELL J. S. B.: Terrain decomposition and layered manufacturing. *International Journal of Computational Geometry & Applications 11*, 06 (2001), 647–668. URL: https://doi.org/10.1142/S0218195901000687, doi:10.1142/S0218195901000687. 7

[FNF02] FU M., NEE A. Y., FUH J. Y.: The application of surface visibility and moldability to parting line generation. *Computer-Aided Design 34*, 6 (2002), 469–480. 6

[fre] Freefab. https://www.freefab.com/. Accessed: 2022-01-20. 11

[GL02] GOODSHIP V., LOVE J.: *Multi-Material Injection Moulding*. Rapra Review, 2002. 9

[GWT94] GAN J. G., WOO T. C., TANG K.: Spherical Maps: Their Construction, Properties, and Approximation. *Journal of Mechanical Design 116*, 2 (06 1994), 357–363. URL: https://doi.org/10.1115/1.2919386, doi:10.1115/1.2919386. 8

[HHZL21] HOU B., HUANG Z., ZHOU H., LI D.: A hybrid hint-based and fuzzy comprehensive evaluation method for optimal parting curve generation in injection mold design. *The International Journal of Advanced Manufacturing Technology 112*, 7 (jan 2021), 2133–2148. doi:10.1007/s00170-020-06461-w. 13

[HL14] HACK N., LAUER W. V.: Mesh-mould: Robotically fabricated spatial meshes as reinforced concrete formwork. *Architectural Design 84*, 3 (2014), 44–53. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/ad.1753, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/ad.1753, doi:https://doi.org/10.1002/ad.1753. 11

[HLZCO14] HU R., LI H., ZHANG H., COHEN-OR D.: Approximate pyramidal shape decomposition. *ACM Trans. Graph. 33*, 6 (Nov. 2014), 213:1–213:12. 7

[HMA15] HERHOLZ P., MATUSIK W., ALEXA M.: Approximating free-form geometry with height fields for manufacturing. *Comput. Graph. Forum 34*, 2 (May 2015), 239–251. URL: http://dx.doi.org/10.1111/cgf.12556, doi:10.1111/cgf.12556. 7, 8, 10, 12

[HPH*21] HOU B., PENG J., HE S., HUANG Z., ZHOU H., LI D.: A similarity-based approach for the variant design of core-pulling mechanism in injection mold design. *The International Journal of Advanced Manufacturing Technology 115*, 1 (may 2021), 329–344. doi:10.1007/s00170-021-06962-2. 13

[HT92] HUI K., TAN S.: Mould design with sweep operations — a heuristic search approach. *Computer-Aided Design 24*, 2 (1992), 81–91. URL: https://www.sciencedirect.com/science/article/pii/001044859290002R, doi:10.1016/0010-4485(92)90002-R. 6

[Hui97] HUI K.: Geometric aspects of the mouldability of parts. *Computer-aided design 29*, 3 (1997), 197–208. 6

[Hun80] HUNT L.: The long history of lost wax casting. *Gold bulletin 13*, 2 (1980), 63–79. URL: https://doi.org/10.1007/BF03215456, doi:10.1007/BF03215456. 2

[ILW19] ISLAM A., LI X., WIRSKA M.: Injection moulding simulation and validation of thin wall components for precision applications. In *Advances in Manufacturing II* (2019), Gapiński B., Szostak M., Ivanov V., (Eds.), vol. 4, Springer, pp. 96–107. Sixth International Scientific-Technical Conference Manufacturing ; Conference date: 19-05-2019 Through 22-05-2019. doi:10.1007/978-3-030-16943-5_9. 2

[Jac17] JACOBSON A.: Generalized matryoshka: Computational design of nesting objects. *Comput. Graph. Forum 36*, 5 (Aug. 2017), 27–35. URL: https://doi.org/10.1111/cgf.13242, doi:10.1111/cgf.13242. 8

[JBR*17] JIPA A., BERNHARD M., RUFFRAY N., WANGLER T., FLATT R., DILLENBURGER B.: skelethon formwork 3d printed plastic formwork for load-bearing concrete structures. *Blucher Design Proceedings 3*, 12 (2017), 345–352. URL: https://www.proceedings.blucher.com.br/article-details/27649, doi:http://dx.doi.org/10.5151/sigradi2017-054. 11

[JD21] JIPA A., DILLENBURGER B.: 3d printed formwork for concrete: State-of-the-art, opportunities, challenges, and applications. *3D*

Printing and Additive Manufacturing 0, 0 (2021), null. URL: https://doi.org/10.1089/3dp.2021.0024, doi:10.1089/3dp.2021.0024. 11

[Kaz16] KAZMER D. O.: *Injection mold design engineering*, 2 ed. Carl Hanser Verlag, 2016. URL: https://www.sciencedirect.com/book/9781569905708/injection-mold-design-engineering. 2, 13

[KB18] KERKSTRA R., BRAMMER S.: *Injection Molding Advanced Troubleshooting Guide*. Carl Hanser Verlag, may 2018. doi:10.3139/9781569906460. 13

[KBM06] KHARDEKAR R., BURTON G., MCMAINS S.: Finding feasible mold parting directions using graphics hardware. *Computer-Aided Design 38*, 4 (2006), 327–341. 4, 6

[KKG02] KUMAR M., K. GUPTA S.: Automated design of multistage molds for manufacturing multi-material objects. *Journal of Mechanical Design - J MECH DESIGN 124* (09 2002). doi:10.1115/1.1485741. 9, 12

[Kle09] KLEIN P.: Fundamentals of plastics thermoforming. 11

[KSAS17] KHAN M. A. A., SHEIKH A. K., AL-SHAER B. S.: Evolution of metal casting technologies—a historical perspective. In *Evolution of Metal Casting Technologies*. Springer, 2017, pp. 1–43. 1

[LG04] LI X., GUPTA S. K.: Geometric algorithms for automated design of rotary-platen multi-shot molds. *Computer-Aided Design 36*, 12 (2004), 1171–1187. URL: https://www.sciencedirect.com/science/article/pii/S0010448503002331, doi:https://doi.org/10.1016/j.cad.2003.11.003. 9

[LML09] LI W., MARTIN R. R., LANGBEIN F. C.: Molds for meshes: Computing smooth parting lines and undercut removal. *IEEE Transactions on Automation Science and Engineering 6*, 3 (2009), 423–432. doi:10.1109/TASE.2009.2021324. 6

[LQ14] LIN A. C., QUANG N. H.: Automatic generation of mold-piece regions and parting curves for complex cad models in multi-piece mold design. *Computer-Aided Design 57* (2014), 15–28. URL: http://www.sciencedirect.com/science/article/pii/S0010448514001420, doi:10.1016/j.cad.2014.06.014. 4, 8, 12

[Mal10] MALLOY R. A.: *Plastic Part Design for Injection Molding*, 2 ed. Carl Hanser Verlag, oct 2010. doi:10.3139/9783446433748. 13

[MLS*18] MUNTONI A., LIVESU M., SCATENI R., SHEFFER A., PANOZZO D.: Axis-aligned height-field block decomposition of 3d shapes. *ACM Transactions on Graphics 37*, 5 (2018). doi:10.1145/3204458. 7

[MPBC16] MALOMO L., PIETRONI N., BICKEL B., CIGNONI P.: Flexmolds: Automatic design of flexible shells for molding. *ACM Trans. Graph. 35*, 6 (Nov. 2016), 223:1–223:12. URL: http://doi.acm.org/10.1145/2980179.2982397, doi:10.1145/2980179.2982397. 5, 10, 12

[MPR*94] MERZ R., PRINZ F., RAMASWAMI K., TERK M., WEISS L.: Shape deposition manufacturing. In *1994 International Solid Freeform Fabrication Symposium* (1994). 9

[NAI*18] NAKASHIMA K., AUZINGER T., IARUSSI E., ZHANG R., IGARASHI T., BICKEL B.: Corecavity: Interactive shell decomposition for fabrication with two-piece rigid molds. *ACM Trans. Graph. 37*, 4 (July 2018), 135:1–135:13. URL: http://doi.acm.org/10.1145/3197517.3201341, doi:10.1145/3197517.3201341. 7, 10, 12, 14

[NFF*97] NEE A., FU M., FUH J., LEE K., ZHANG Y.: Determination of optimal parting directions in plastic injection mold design. *CIRP Annals 46*, 1 (1997), 429–432. 6

[PG04] PRIYADARSHI A. K., GUPTA S. K.: Geometric algorithms for automated design of multi-piece permanent molds. *Computer-Aided Design 36*, 3 (2004), 241–260. URL: https://www.sciencedirect.com/science/article/pii/S0010448503001076, doi:10.1016/S0010-4485(03)00107-6. 8, 12

[PG09] PRIYADARSHI A. K., GUPTA S. K.: Algorithms for generating multi-stage molding plans for articulated assemblies. *Robotics and Computer-Integrated Manufacturing 25*, 1 (2009), 91–106. URL: https://www.sciencedirect.com/science/article/pii/S0736584507001044, doi:10.1016/j.rcim.2007.10.002. 9, 12

[PGG*07] PRIYADARSHI A. K., GUPTA S. K., GOUKER R., KREBS F., SHROEDER M., WARTH S.: Manufacturing multi-material articulated plastic products using in-mold assembly. *The International Journal of Advanced Manufacturing Technology 32*, 3 (Mar 2007), 350–365. URL: https://doi.org/10.1007/s00170-005-0343-z, doi:10.1007/s00170-005-0343-z. 9, 12

[Ree02] REES H.: *Mold Engineering*. Hanser Publications, 2002. 2

[RMRD21] RENÉ W.-J., MAURICE T., ROMAN K., DIRK P.: Water-soluble sacrificial 3d printed molds for fast prototyping in ceramic injection molding. *Additive Manufacturing 48* (2021), 102408. URL: https://www.sciencedirect.com/science/article/pii/S2214860421005613, doi:https://doi.org/10.1016/j.addma.2021.102408. 2

[RS90] RAVI B., SRINIVASAN M.: Decision criteria for computer-aided parting surface design. *Computer-Aided Design 22*, 1 (1990), 11–18. 6

[SA07] SORKINE O., ALEXA M.: As-rigid-as-possible surface modeling. In *Proceedings of the Fifth Eurographics Symposium on Geometry Processing* (Aire-la-Ville, Switzerland, Switzerland, 2007), SGP '07, Eurographics Association, pp. 109–116. URL: http://dl.acm.org/citation.cfm?id=1281991.1282006. 7, 8

[Sch15] SCHIPPER H.: *Double-curved precast concrete elements: Research into technical viability of the flexible mould method*. PhD thesis, Delft University of Technology, 2015. doi:10.4233/uuid:cc231be1-662c-4b1f-a1ca-8be22c0c4177. 11

[SDBP12] SAROJRANI P., D. BENNY K., P.K. J.: Developments in investment casting process—a review. *Journal of Materials Processing Technology 212*, 11 (2012), 2332–2348. URL: https://www.sciencedirect.com/science/article/pii/S0924013612001823, doi:https://doi.org/10.1016/j.jmatprotec.2012.06.003. 2

[SEB*21] SHAKERI H., ELBAGGARI H., BUCCI P., XIAO R., MACLEAN K. E.: Foldmold: Automating papercraft for fast diy casting of scalable curved shapes. In *Proceedings of Graphics Interface 2021* (2021), GI 2021, Canadian Information Processing Society, pp. 77 – 86. doi:10.20380/GI2021.10. 2

[Sia05] SIAS F. R.: *Lost-Wax Casting: old, new, and inexpensive methods*. Woodsmere press, 2005. 2

[SJG19] STEIN O., JACOBSON A., GRINSPUN E.: Interactive design of castable shapes using two-piece rigid molds. *Computers & Graphics 80* (2019), 51–62. URL: https://www.sciencedirect.com/science/article/pii/S0097849319300238, doi:10.1016/j.cag.2019.03.001. 7, 12, 14

[SMGMFL18] SARMIENTO-MERIDA L. A., GUEVARA-MORALES A., FIGUEROA-LÓPEZ U.: Determining the optimum parting direction in plastic injection molds based on minimizing rough machining time during mold manufacturing. *Advances in Polymer Technology 37*, 1 (Jan. 2018), 194–201. doi:10.1002/adv.21656. 13

[SPG*16] SCHÜLLER C., PANOZZO D., GRUNDHÖFER A., ZIMMER H., SORKINE E., SORKINE-HORNUNG O.: Computational thermoforming. *ACM Trans. Graph. 35*, 4 (July 2016), 43:1–43:9. URL: http://doi.acm.org/10.1145/2897824.2925914, doi:10.1145/2897824.2925914. 11

[SSS14] SAHOO M., "SAM" SAHU S.: *Principles of metal casting*. McGraw-Hill Education, 2014. URL: https://www.accessengineeringlibrary.com/content/book/9780071789752. 2

[TSJ*18] TAO P., SHAO H., JI Z., NAN H., XU Q.: Numerical simulation for the investment casting process of a large-size titanium alloy thin-wall casing. *Progress in Natural Science: Materials International 28*, 4 (2018), 520–528. URL: https://www.sciencedirect.com/science/article/pii/S1002007118301977, doi:https://doi.org/10.1016/j.pnsc.2018.06.005. 2

[VLAR19] VALKENEERS T., LEEN D., ASHBROOK D., RAMAKERS R.: Stackmold: Rapid prototyping of functional multi-material objects with selective levels of surface details. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (New York, NY, USA, 2019), UIST '19, Association for Computing Machinery, p. 687–699. URL: https://doi.org/10.1145/3332165.3347915, doi:10.1145/3332165.3347915. 9, 12

[Wan11] WANNARUMON S.: Reviews of computer-aided technologies for jewelry design and casting. *Naresuan University Engineering Journal 6*, 1 (2011), 45–56. 1, 2

[Wor16] WORLD L. C.: How is the lindt goldbunny actually being made, 2016. URL: https://www.youtube.com/watch?v=9uuVuQKPdeU. 1

[WSP21] WANG Z., SONG P., PAULY M.: State of the art on computational design of assemblies with rigid parts. *Computer Graphics Forum 40*, 2 (2021), 633–657. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.142660, doi:10.1111/cgf.142660. 5, 9

[WXM*20] WANG C., XU B., MENG Q., RONG J., ZHAO Y.: Topology optimization of cast parts considering parting surface position. *Advances in Engineering Software 149*, 102886 (nov 2020), 1–15. doi:10.1016/j.advengsoft.2020.102886. 14

[YM18] YUSOF M. M., MANSOR M. S. A.: Alternative method to determine parting direction automatically for generating core and cavity of two-plate mold using B-rep of visibility map. *The International Journal of Advanced Manufacturing Technology 96*, 9 (feb 2018), 3109–3126. doi:10.1007/s00170-018-1695-5. 13

[ZFS*19] ZHANG X., FANG G., SKOURAS M., GIESELER G., WANG C. C. L., WHITING E.: Computational design of fabric formwork. *ACM Trans. Graph. 38*, 4 (July 2019), 109:1–109:13. URL: http://doi.acm.org/10.1145/3306346.3322988, doi:10.1145/3306346.3322988. 11, 12

[ZTZ17] ZHANG Y., TONG Y., ZHOU K.: Coloring 3d printed surfaces by thermoforming. *IEEE Transactions on Visualization and Computer Graphics 23*, 8 (2017), 1924–1935. 11

[ZZL10] ZHANG C., ZHOU X., LI C.: Feature extraction from freeform molded parts for moldability analysis. *The International Journal of Advanced Manufacturing Technology 48*, 1 (Apr 2010), 273–282. URL: https://doi.org/10.1007/s00170-009-2273-7, doi:10.1007/s00170-009-2273-7. 6