

No. 689

September 2026

**Efficient Geometry Representation Strategies
For the Shape Optimization of
Profile Extrusion Dies**

J. Sasse, M. Esser, M. Mügge, S. Turek

ISSN: 2190-1767

Efficient Geometry Representation Strategies for the Shape Optimization of Profile Extrusion Dies

Jana Sasse, Maximilian Esser, Markus Mügge, Stefan Turek

Institute for Applied Mathematics, LSIII, TU Dortmund University, D-44227 Dortmund Germany

jana.sasse@math.tu-dortmund.de



This work is licensed under a Creative Commons Attribution 4.0 International license (see <https://creativecommons.org/licenses/by/4.0/legalcode.de>)

Abstract: The design of profile extrusion dies remains a challenging task due to the complex rheological behavior of polymer melts and the geometric intricacies of flow channels. Traditional manual optimization approaches, which rely heavily on human experience, are inefficient and often employ unvalidated heuristics. To address these challenges, we present a deterministic and explainable framework for automatic die design based on adjoint-based shape optimization. This approach enables the computation of sensitivities that directly indicate beneficial modifications to the flow channel geometry. A major difficulty in such optimization processes lies in generating boundary-conforming meshes that evolve consistently with changing geometries. To overcome this issue, we employ non-boundary conforming geometry representation methods that eliminate the need for an explicit surface representation along physical boundaries. A dedicated reconstruction technique is developed to recover accurate sensitivity information at the virtual interface between fluid and solid regions. The proposed algorithm is demonstrated on 3D geometries with varying complexity, including realistic extrusion die flow channels. Several objective functionals relevant to industrial applications, such as flow balance at the outflow, are considered. The results highlight significant improvements in performance metrics while maintaining numerical robustness. This work showcases the potential of adjoint-based techniques for automated die design in a domain still largely governed by manual trial-and-error procedures, establishing a foundation for data-efficient, sustainable manufacturing workflows using computational rheology.

Keywords— CFD, Fictitious Boundary Method, Immersed Boundary Method, Shape Optimization, Extrusion

1. Introduction

The design of the flow channel in profile extrusion dies remains challenging. The optimal flow channel geometry creates a uniform outflow velocity profile. However, this is highly dependent on the operating point, i.e., the throughput, processing temperature, and thermo-rheological properties of the plastic melt. As a result, extrusion die design still relies heavily on manual design iterations.

The last years have seen an emergence of computational methods for extrusion die design. However, these methods face various numerical challenges. For complex 3D geometries, the modeling of the computational domain alone is a non-trivial task. In addition, the extruded plastic melt exhibits a complex rheological behavior, since it is shear-thinning, viscoelastic and temperature-dependent. Furthermore, the computational cost of established optimization methods [1] is also significant due to the associated number of solver calls. Adjoint-based sensitivities are a promising alternative at low computational cost [2, 3, 4].

The goal of the research presented in this paper is to identify the most efficient method for adjoint-based optimization of complex profile extrusion dies. This includes the most efficient geometry representation strategy, but also the best strategy for the sensitivity-based optimization algorithm.

2. Numerical Modeling

2.1. Geometry Representation Strategies

The most common method employs a body-fitted mesh that describes the entire fluid domain. While this method has the advantage that it operates on the "exact representation" of the flow channel geometry, it also comes with its drawbacks. For complex flow channel geometries, the computational cost of the mesh generation becomes prohibitively expensive. In addition, large deformations of the flow channel during the optimization process often require re-meshing during runtime.

Non-boundary conforming methods offer an alternative approach to geometry representation. In this paper we will consider two methods, although there are even more established approaches. The fictitious boundary method (FBM) [5] is an implicit non-boundary conforming method that employs an indicator function $\tilde{\alpha}(X)$ to mask points in the fluid domain.

$$\tilde{\alpha}(X) = \begin{cases} 1 & \text{for } X \in \Omega_f \\ 0 & \text{for } X \in \Omega \setminus \Omega_f \end{cases} \quad (2.1)$$

The Navier-Stokes equations are then solved on the filtered domain Ω_f .

The immersed boundary method (IBM) [4, 6] is an explicit non-boundary conforming method that locally blocks the flow by assigning a pseudo-porosity $\alpha(X)$ to each cell of the computational mesh.

$$\alpha(X) = \begin{cases} 0 & \text{for } X \in \Omega_f \\ \alpha_{max} & \text{for } X \in \Omega_s \end{cases} \quad (2.2)$$

The numerical solution is then obtained by solving the extended Navier-Stokes equations on $\Omega = \Omega_f \cup \Omega_s$:

$$\nabla \cdot \mathbf{u} = 0 \quad (2.3)$$

$$\rho(\mathbf{u} \cdot \nabla) \mathbf{u} - \nabla \cdot \boldsymbol{\sigma}(\mathbf{u}, \mathbf{p}) + \alpha \mathbf{u} = 0 \quad (2.4)$$

Both non-boundary conforming methods enable computationally cheap mesh generation that is independent of the geometric complexity of the flow channel. However, this comes at a loss of accuracy, particularly at the interface. While the interface in FBM can be recovered through evaluation of the gradient of the indicator function $\nabla \tilde{\alpha}(X)$, that information is lost in IBM. This, however, is not necessarily an advantage for FBM, since any data projected to the interface are prone to artifacts in areas with insufficient co-planarity between the background mesh and the interface. As a result, IBM can sometimes be more robust regarding interface artifacts or instabilities. However, this comes at the additional cost of interface reconstruction for later processing of the optimization result.

2.2. Adjoint-based Optimization

The goal of adjoint-based optimization is to minimize an objective functional J under the PDE constraints R , which in this case arise from the incompressible steady Navier-Stokes equations:

$$\min_{\mathbf{b}} J(\boldsymbol{\chi}, \mathbf{b}) \quad \text{s. t. } \mathbf{R}(\boldsymbol{\chi}, \mathbf{b}) = \mathbf{0} \quad (2.5)$$

The idea is to introduce a Lagrange functional L with Lagrange multipliers λ , also called the adjoint variables:

$$L := J + \int_{\Omega} \lambda R d\Omega \quad (2.6)$$

This allows for an optimization of a suitable design variable $\mathbf{b}$ through a gradient descent method (Eq. 2.7), where the gradient is provided by the directional derivative of the Lagrange functional with respect to the design variable, also known as the design sensitivity:

$$\mathbf{b}_{n+1} = \mathbf{b}_n - \Delta \frac{\partial L(\mathbf{b}_n)}{\partial \mathbf{b}_n} \quad (2.7)$$

This directional derivative of the Lagrange functional can be calculated after solving the primal state equations and their adjoint problem in only two solver calls [2, 3, 4]. The primal problem can be stated as: Find $(\mathbf{u}, p, \mu)$ of sufficient regularity holding

$$\rho(\mathbf{u} \cdot \nabla) \mathbf{u} - \nabla \cdot \boldsymbol{\sigma}(\mathbf{u}, p, \mu) + \alpha \mathbf{u} = 0, \quad \text{in } \Omega, \quad (2.8)$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega, \quad (2.9)$$

$$\mu - F(\dot{\gamma}) = 0 \quad \text{in } \Omega, \quad (2.10)$$

$$\mathbf{u} = \mathbf{u}_0 \quad \text{on } \partial\Omega_{in}, \quad (2.11)$$

$$\mathbf{u} = 0 \quad \text{on } \partial\Omega \setminus (\partial\Omega_{in} \cup \partial\Omega_{out}), \quad (2.12)$$

$$(2\mu \mathbf{D}(\mathbf{u}) - p \mathbf{I}) \cdot \mathbf{n} = 0 \quad \text{on } \partial\Omega_{out}, \quad (2.13)$$

$$\boldsymbol{\sigma}(\mathbf{u}, p, \mu) := (2\mu \mathbf{D}(\mathbf{u}) - p \mathbf{I}), \quad (2.14)$$

where Eq. 2.10 is an auxiliary problem needed for the shear-rate dependent viscosity in the adjoint model [2].

The corresponding adjoint model reads: Given the primal solution $(\mathbf{u}, p, \mu)$, find $(\mathbf{u}_a, p_a)$ holding

$$\frac{\partial j_\Omega}{\partial \mathbf{u}} + \rho (\mathbf{u} \cdot \nabla) \mathbf{u}_a - \rho (\mathbf{u}_a \cdot \nabla) \mathbf{u} - \sigma(\mathbf{u}_a, p_a, \mu) + \nabla \cdot (\mu_a \mathbf{X}) + \alpha \mathbf{u} = 0 \quad \text{in } \Omega, \quad (2.15)$$

$$\nabla \cdot \mathbf{u}_a - \frac{\partial j_\Omega}{\partial p} = 0 \quad \text{in } \Omega, \quad (2.16)$$

$$\frac{\partial j_\Omega}{\partial \mu} + 2\mathbf{D}(\mathbf{u}) : \mathbf{D}(\mathbf{u}_a) + \mu_a = 0 \quad \text{in } \Omega, \quad (2.17)$$

$$\mathbf{u}_a|_t = 0, \mathbf{u}_a|_n = -\frac{\partial j_\Gamma}{\partial p} \quad \text{on } \partial\Omega \setminus \partial\Omega_{\text{out}}, \quad (2.18)$$

$$\sigma(\mathbf{u}_a, p_a, \mu) \cdot \mathbf{n} = -\frac{\partial j_\Gamma}{\partial \mathbf{u}} - \rho \mathbf{u}_a (\mathbf{u} \cdot \mathbf{n}) + \mu_a \mathbf{X} \cdot \mathbf{n} \quad \text{on } \partial\Omega_{\text{out}}, \quad (2.19)$$

$$\mu = F(\dot{\gamma}), \quad (2.20)$$

$$\mathbf{X} := 2\dot{\gamma}^{-1} F'(\dot{\gamma}) \mathbf{D}(\mathbf{u}), \quad (2.21)$$

$$\mu_a = -2\mathbf{D}(\mathbf{u}) : \mathbf{D}(\mathbf{u}_a) - \frac{\partial j_\Omega}{\partial \mu}. \quad (2.22)$$

Note that the highlighted term $+\alpha \mathbf{u}$ in Eq. 2.8 and Eq. 2.15 is only added when using IBM. The solution of the primal and adjoint system is then used to calculate the design sensitivity for the optimization of the flow channel geometry.

Adjoint-based optimization can be performed using two approaches, shape optimization and topology optimization [3]. In shape optimization, the design variable is the localized surface normal displacement β on the nodes of the design boundary. The corresponding surface sensitivity σ_β is calculated as

$$\delta_b L \cdot \delta \mathbf{b} = \sigma_\beta = \left[-2\mu \mathbf{D}(\mathbf{u}_a) \cdot \mathbf{n} + \mu_a 2\dot{\gamma}^{-1} F'(\dot{\gamma}) \mathbf{D}(\mathbf{u}) \cdot \mathbf{n} \right] \cdot \frac{\partial \mathbf{u}}{\partial \mathbf{n}}. \quad (2.23)$$

The shape update is then performed by moving the points on the design boundary according to the localized surface normal displacements β and performing a subsequent smoothing step on the inner nodes of the mesh. Topology optimization, on the other hand, offers more design freedom since it is not restricted to the initial topology of the design boundary. For our purpose, the design variable is the IBM variable α and the corresponding volume sensitivity σ_{α_i} for each cell i of the computational mesh is calculated using the local cell volume V_i with

$$\delta_b L \cdot \delta \mathbf{b} = \sigma_{\alpha_i} = u_{a,i} \cdot u_i V_i. \quad (2.24)$$

The subsequent topology update ($n \rightarrow n+1$) is performed using a gradient-based update scheme

$$\alpha_{n+1} = \alpha_n (1 + \gamma) + \gamma \min(\max(\alpha_n - \lambda^* \sigma_\alpha, 0), \alpha_{\max}) \quad (2.25)$$

where γ and λ^* refer to an underrelaxation factor and the response step size, respectively. While this approach offers more design freedom, it can also require additional geometric constraints to ensure optimization results that are also viable from an engineering perspective.

2.3. Discussion

The geometry representation strategies presented in Sec. 2.1 and the optimization algorithms presented in Sec. 2.2 allow for different algorithm pairings. However, not all of them are equally suitable for a successful optimization for profile extrusion dies. Both shape optimization using an IBM approach and topology optimization using an FBM approach are not feasible within the described framework since the involved components are inherently incompatible.

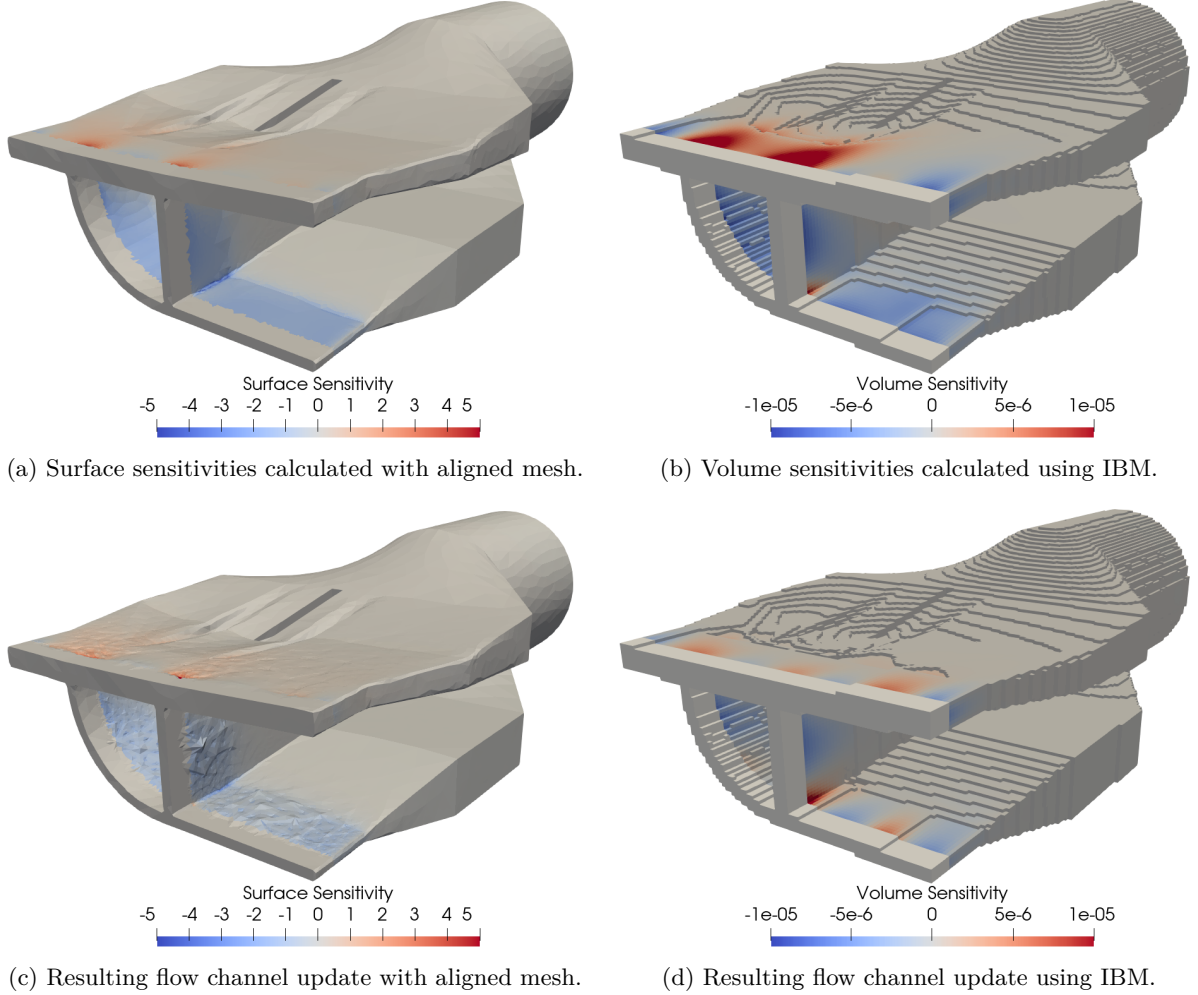


Fig. 2.1. Suitable geometry modeling and optimization strategy pairings for a complex flow channel.

The most intuitive methods are shape optimization using an aligned mesh (see Fig. 2.1a) and topology optimization using IBM (see Fig. 2.1b). While the former leads to a visually superior sensitivity distribution, the optimization result will depend a lot on the computational mesh. The difference becomes apparent in Fig. 2.1c and Fig. 2.1d, where the pattern of the underlying computational mesh dictates the shape of the optimized surface. As a result, any aligned mesh used for this purpose has to adhere to even more restrictions than are usually imposed on meshes for computational fluid dynamics. This makes topology optimization using IBM the less "pretty" but more robust choice for an optimization algorithm for complex geometries.

Topology optimization with an aligned mesh can work. However, this requires an appropriate interface resolution and is usually not feasible for complex geometries. Shape optimization using an FBM geometry representation can be successful. Fig. 2.2a and Fig. 2.2b depict the surface sensitivity for a flatsheet extrusion die using both an aligned mesh and FBM. However, this approach breaks down as soon as the interface is not co-planar with the background mesh. Revisiting the example given in Figs. 2.1a-2.1b, the surface of the flow channel is not co-planar with the background mesh (see Fig. 2.2d). As a result, the projection of the local velocity to the interface leads to wave-patterned artifacts (see Fig. 2.2c). While these can be reduced by smoothing, this approach risks removing real results in the process.

As a result, the immersed boundary method combined with a topology optimization algorithm appears to be the most versatile method for use cases with industrial relevance. However, shape optimization with aligned

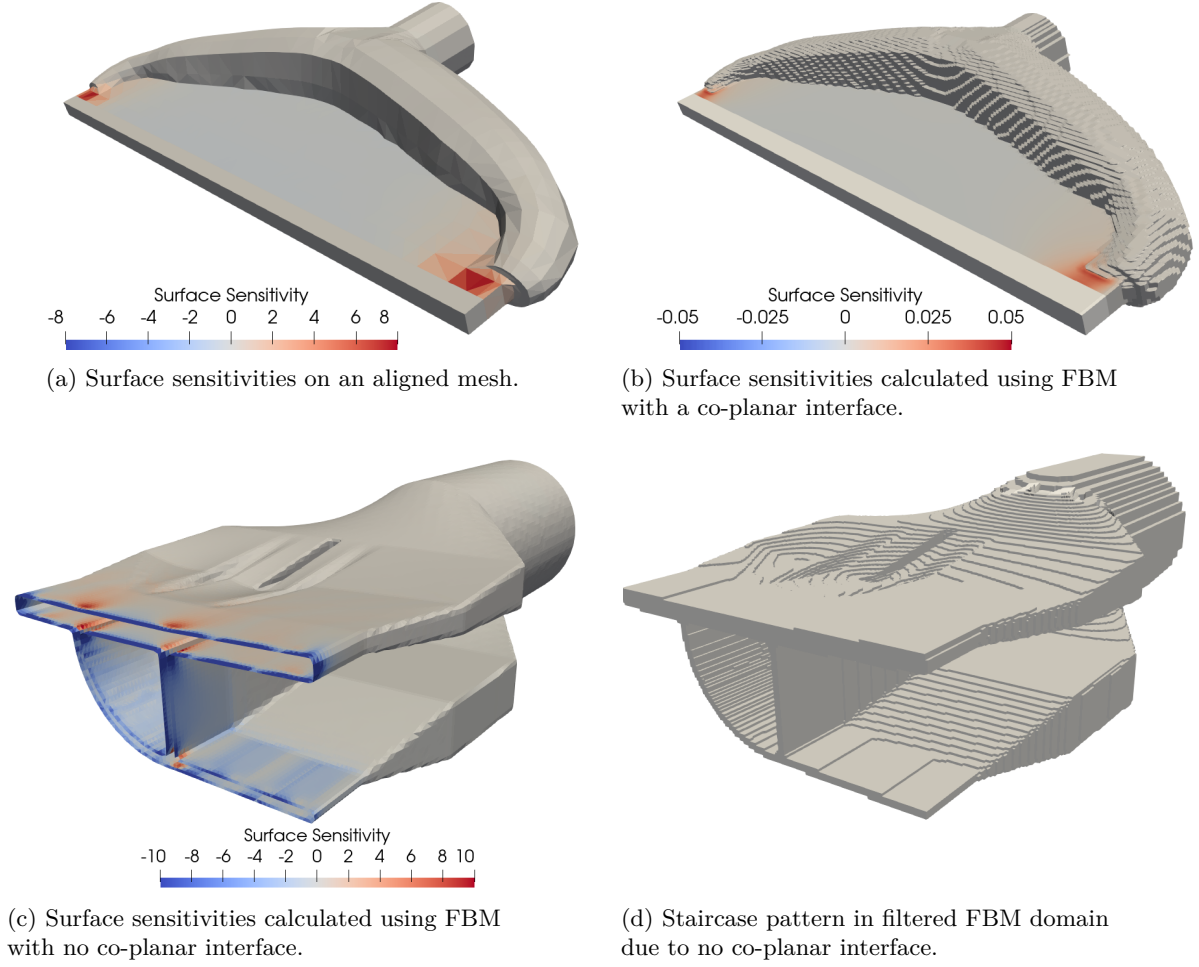


Fig. 2.2. Example where FBM leads to artifacts in the calculated surface sensitivity.

meshes will still produce the most reliable result, provided an appropriate aligned mesh is available.

3. Application to Profile Extrusion

3.1. Setup and Numerical Model

This flow channel geometry is based on a real profile extrusion die for an extruder with a 45 mm diameter. It shapes the melt into the complex outflow profile depicted in Fig. 3.1a. In addition to other typical features in traditional extrusion die design, it includes a 15 mm parallel zone near the outlet where the geometry cannot be modified in order to control the die swell by allowing for sufficient melt relaxation. The initial and boundary conditions for this use case are also taken from a real process operating point with a throughput of 10 kg/h and a processing temperature of 220 °C. The extruded material is a polyethylene of high density (PE-HD) and is modeled with a Carreau-WLF approach. The first optimization objective for this use case is the maximization of the flow balance at the outlet (initial configuration shown in Fig. 3.1b), which is equivalent to the minimization of the deviation of the local velocity from the average velocity at the outlet $\bar{\mathbf{u}}$.

$$\min J = \int_{\Gamma_{out}} \frac{1}{2} (\mathbf{u} - \bar{\mathbf{u}})^2 \quad (3.1)$$

The imbalance in the outflow velocity distribution is exacerbated by the high viscosity and shear-thinning behavior of the extruded material. In practice, this means that the flow channel of the die is not filled appropriately (see Fig. 3.2).

The flow channel geometry was modeled using an IBM approach, embedding the geometry in a cartesian background mesh. α_{max} was chosen at 10^5 , balancing blockage of the flow with numerical stability. The optimization algorithm calculates the volume sensitivity according to Eq. 2.24. The topology update is effectively reduced to a shape update by restricting the geometry update in Eq. 2.25 to cells at the interface, i. e., cells that have vertices on the fluid and the solid side of the interface. No underrelaxation was used in this case ($\gamma = 1$) and the response

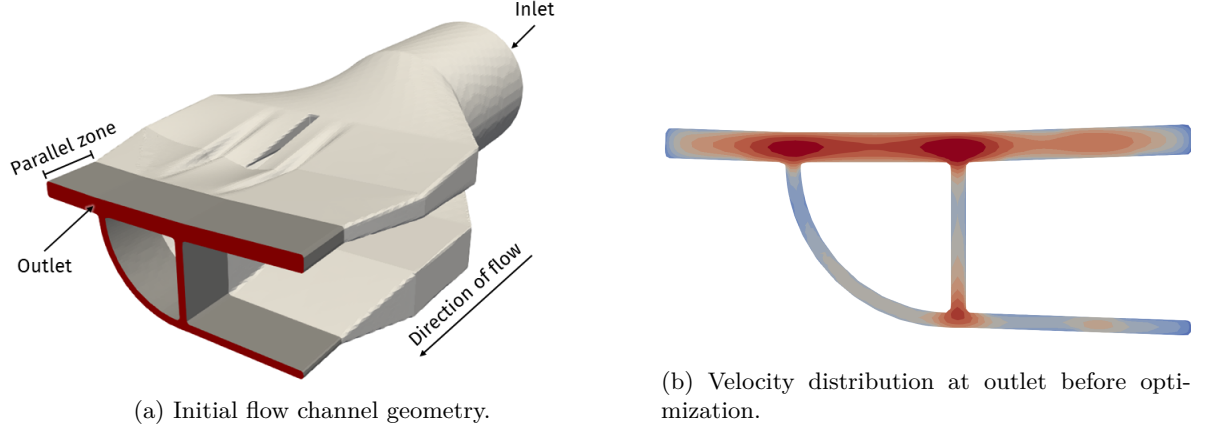


Fig. 3.1. Setup of the demonstrator profile extrusion die.

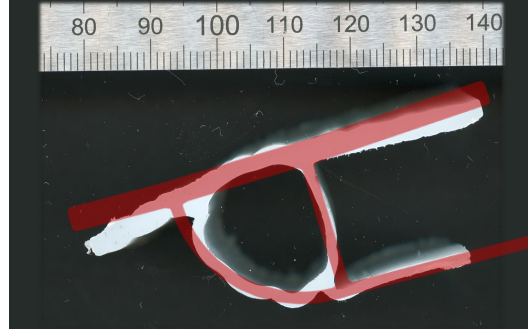


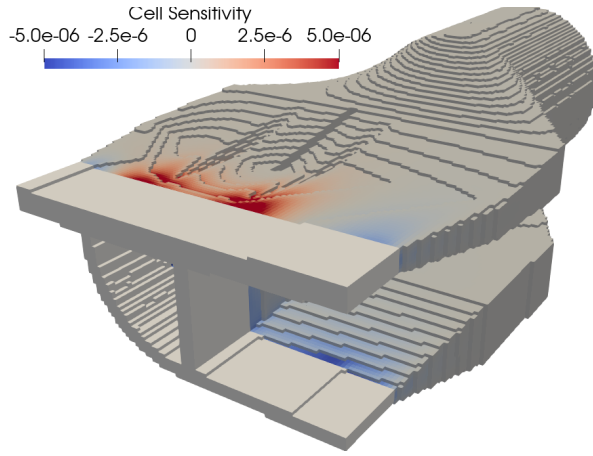
Fig. 3.2. Difference between the target profile (red) and the actual profile (white) in extrusion trials before flow channel optimization. Image provided by Kunststofftechnik Paderborn (KTP).

step size was manually fitted to $\lambda^* = 10^{10}$. A total of 20 optimization loops were performed, meaning that the results shown later will represent the optimization result after 19 optimization steps. The resulting flow channel geometry is reconstructed using a SurfaceNets algorithm [7], ensuring a 2-manifold surface using the approach by Schaefer et al. [8], and exported as a surface mesh file.

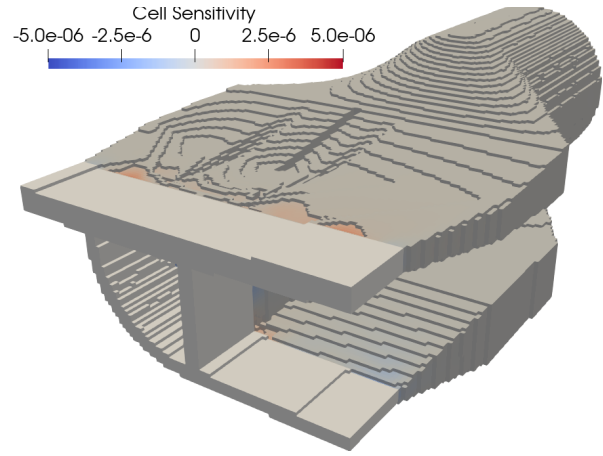
The numerical model was implemented in the open-source finite element software FEAT3, which is part of the FEATFLOW software family [9]. The spatial discretization is performed using Q^2/P_{disc}^1 elements, which are inf-sup stable and particularly suited for non-Newtonian flows without the need for additional stabilization [10]. The numerical model is solved as follows [11]: The nonlinear primal system is solved directly into its steady state using a monolithic Newton solver. The adjoint system does not require this step since it is already linear. The linearized systems are then solved using a monolithic geometric multigrid solver. Using a background mesh with 3,750 hexahedral elements on the coarsest level with three refinement levels, this results in a total of $\sim 48,000,000$ degrees of freedom per optimization loop. The solver uses a Vanka-type cell-wise additive Schwarz smoother with a GMRES solver at the coarsest level. Simulations were run using two compute servers, each containing two AMD EPYC 9354 CPUs with a total of 64 cores and 1.5 TB of DDR5 @ 4800 MHz RAM, resulting in a run time of 20 minutes in the first optimization loop. Later optimization steps are initialized using the solutions from the previous step to reduce computational cost.

3.2. Results

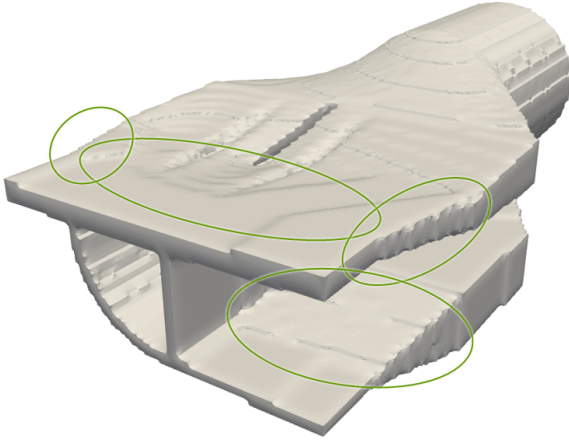
Fig. 3.3a depicts the resulting volume sensitivities in the first optimization loop. Positive sensitivities (red areas) highlight areas where the algorithm wants to restrict the flow channel, while negative sensitivities (blue areas) highlight parts of the flow channels that should be expanded to improve the flow balance. The former primarily affects the flow channel near the upper T-junction, where the outflow velocities are too high, while the latter mainly affects the lower parts of the flow channel, where the small channel widths previously led to an incomplete filling of the die's flow channel (see Fig. 3.2). Figs. 3.3c-3.3d show how the flow channel was adapted according to Eq. 2.25, reducing the cost functional from 439.435 to 207.800 in the process. This results in a reduction of the



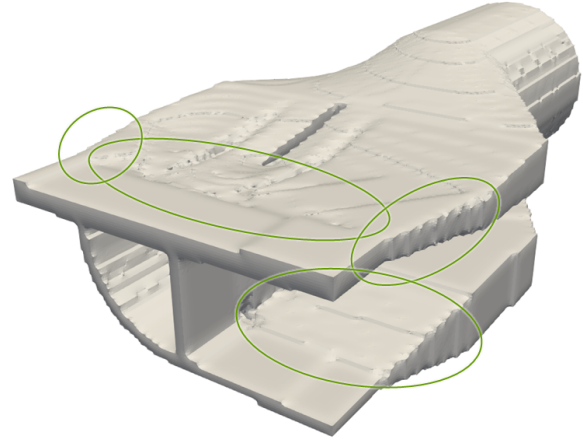
(a) Volume sensitivity before optimization.



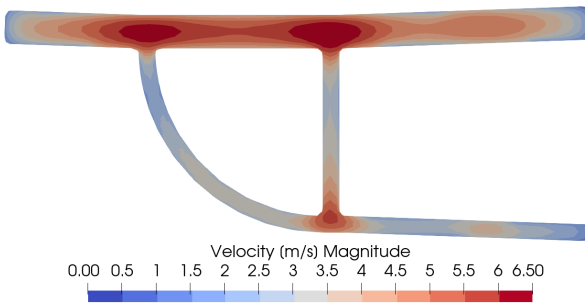
(b) Volume sensitivity after 19 optimization steps.



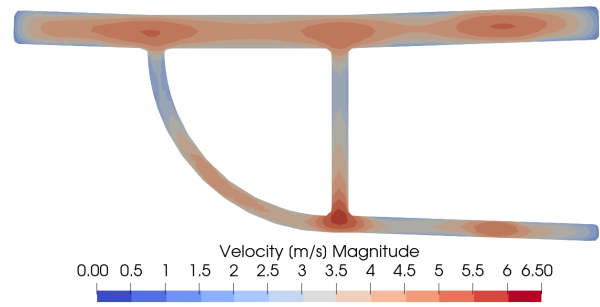
(c) Flow channel geometry before optimization.



(d) Flow channel geometry after 19 optimization steps.



(e) Velocity distribution at outlet before optimization.



(f) Velocity distribution at outlet after 19 optimization steps.

Fig. 3.3. Changes to flow channel geometry and resulting changes to outflow velocity distribution.

standard deviation of the velocity at the outlet from 2507.75 to 1128.65, which in turn means an improvement in the flow balance by 55% (see Figs. 3.3e-3.3f). Although the optimization algorithm has not fully converged at this point, the most significant changes to the shape are finished; the remaining areas with potential for optimization focus on the parts of the flow channel that could still be expanded a bit more (see Fig. 3.3b).

4. Conclusion and Outlook

The results of this study demonstrate that adjoint-based optimization can be successful using various pathways, but not all of them are equally suited for engineering applications. In particular, not all methods suited for CFD analysis are automatically suited for shape or topology optimization. For complex profile extrusion dies, topology optimization using an immersed boundary method is the most versatile approach. Applying the optimization algorithm to a complex profile extrusion die led to an improvement of the flow balancing by 55% using fewer solver calls than other established CFD-based optimization methods.

Future work will focus on the investigation of further improvements to geometry representation, e.g., through adaptive mesh refinement or local deformation of the mesh [12]. In addition, the developed optimization algorithm will be extended to fully non-isothermal flows and routines improving the robustness of the optimization algorithm will be added. The final step in this research will be the manufacturing of an optimized profile extrusion die based on these optimization results for subsequent validation in lab trials.

Acknowledgments

This project is supported by the Federal Ministry for Economic Affairs and Energy (BMWE) on the basis of a decision by the German Bundestag.

References

- [1] A. Gaspar-Cunha, J. A. Covas, J. Sikora. Optimization of Polymer Processing: A Review (Part I—Extrusion), *Materials*, Vol. 15, No. 1, p. 384, 2022.
- [2] G. Bletsos, N. Kühn, T. Rung. Adjoint shape sensitivities of blood flows considering non-Newtonian properties. *International Journal for Numerical Methods in Fluids*, Vol. 95, No. 11, pp. 1791–1819, 2023.
- [3] C. Othmer. A continuous adjoint formulation for the computation of topological and surface sensitivities of ducted flows. *International Journal for Numerical Methods in Fluids*, Vol. 58, No. 8, pp. 861–877, 2008.
- [4] J. Sasse, Adjoint Topology Optimization of Polymer Melt Flow Channels Producing by Additive Manufacturing, Dissertation, RWTH Aachen University, 2025.
- [5] S. Turek, D. Wan, L. S. Rivkind. The Fictitious Boundary Method for the Implicit Treatment of Dirichlet Boundary Conditions with Applications to Incompressible Flow Simulations. *Challenges in Scientific Computing - CISC 2002, Lecture Notes in Computational Science and Engineering*. pp. 37–68, 2003.
- [6] K. Khadra, P. Angot, S. Parneix, J.-P. Caltagirone. Fictitious domain approach for numerical modelling of Navier-Stokes equations. *International Journal for Numerical Methods in Fluids*, Vol. 34, No. 8, pp. 651–684, 2000.
- [7] S. F. F. Gibson. Constrained elastic surface nets: Generating smooth surfaces from binary segmented data, *Medical Image Computing and Computer-Assisted Intervention — MICCAI’98*, Springer Berlin Heidelberg, pp. 888–898, 1998.
- [8] S. Schaefer, T. Ju, J. Warren. Manifold Dual Contouring, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 13, No. 3, pp. 610–619, 2007.
- [9] Chair for Applied Mathematics and Numerics (LS3), TU Dortmund University. FEATFLOW Software Homepage. Available at: <https://wwwold.mathematik.tu-dortmund.de/~featflow/en/index.html>. Accessed: 08 September 2026.
- [10] D. Boffi, F. Brezzi, M. Fortin, *Mixed Finite Element Methods and Applications*, Springer Berlin Heidelberg (2013).
- [11] M. Esser, S. Turek. On the Automation and Scaling of a Fictitious-Boundary Method for Non-Newtonian Extrusion Die Flows. *Ergebnisberichte des Instituts für Angewandte Mathematik* No. 687, TU Dortmund (2026).
- [12] M. Grajewski, M. Köster, S. Turek. Mathematical and Numerical Analysis of a Robust and Efficient Grid Deformation Method in the Finite Element Context. *SIAM Journal on Scientific Computing*, Vol. 31, No. 2, pp. 1539–1557, 2009.