Materials Research Proceedings 54 (2025) 68-77 https://doi.org/10.21741/9781644903599-8

Modelling bead deposition in material extrusion using the smoothed particle hydrodynamics framework PySPH

Lukas HOF[1,a *], Felix FRÖLICH[1,b], Florian WITTEMANN[1,c] and Luise KÄRGER[1,d]

[1]Karlsruhe Institute of Technology (KIT), Institute of Vehicle Systems Technology (FAST) – Lightweight Engineering, 76131 Karlsruhe Germany

[a]lukas.hof@kit.edu, [b]felix.froelich@kit.edu, [c]florian.wittemann@kit.edu, [d]luise.kaerger@kit.edu

Keywords: Material Extrusion, FFF, FDM, Smoothed Particle Hydrodynamics, Heat Balance

Abstract. Parts manufactured using the additive *Material Extrusion* (MEX) process comprise a small-scale mesostructure of beads and voids. In this work, a numerical model is created with the goal of predicting the formation of this mesostructure. Therefore, the *Smoothed Particle Hydrodynamics* (SPH) framework PySPH is extended with a thermal model and used to simulate the extrusion of multiple beads. The SPH method is well suited to model the complex evolution of free surfaces. While the implemented conductive heat transport model agrees well with a thermal *Finite Element* model, the heat losses at the surface due to radiation and convection are significantly overestimated. An extrusion of multiple layers of beads shows that the inability of the used SPH-scheme to capture the fluid's incompressibility and high viscosity prevents it from accurately predicting the formation of the mesostructure.

Introduction

Material Extrusion (MEX) is a family of very common additive manufacturing processes used by industry and hobbyists alike. The material, usually a thermoplastic polymer melt, is extruded through a nozzle and deposited on a build plate in beads that solidify as the material rapidly cools. A part is formed of multiple layers each consisting of such beads. Like other additive processes, it allows for great design freedom with no part-specific tooling required [1]. During manufacturing a typical *mesostructure* is formed as the distinct beads do not fully merge into a monolithic part but leave voids in between them. The final part's performance is highly dependent on the specific mesostructure, the strength of the formed interfaces, and thermal history. These however are strongly influenced by the printing parameters and conditions [2]. Therefore, experimental characterization would require many experiments to cover the wide array of specific configurations possible. Such experiments must be carefully designed to ensure consistency and comparability [3]. An accurate prediction of the formation of the mesostructure and thermal history using numeric methods can thus reduce the necessary effort significantly.

The mesh-free *Smoothed Particle Hydrodynamics Method* (SPH) independently developed by Lucy [4] and Gingold and Monaghan [5] is especially promising to simulate the process as it intrinsically allows to track the evolution of arbitrarily complex free surfaces and emerging contacts without expensive remeshing or elaborate contact models. The method has previously been used alongside the *Finite Volume Method* (FVM) to study the MEX process. However, existing models are not used to predict the formation of a three-dimensional mesostructure while considering the thermal behavior of the extrudate. The models using the FVM can predict the deposition of a single or a few beads [6–8]. Serdeczny et al. [9] developed a model to predict mesostructures, but it is isothermal and previously deposited beads are treated as a boundary condition and cannot be deformed later. Some SPH-models focus on the movement of fibers within the fluid but are restricted to two dimensions [10–12], while others do not include a thermal model [13,14].

The goal of this work is to model the MEX process using the extendable open-source SPH-framework PySPH [15]. As published, the software does not include the ability to solve for temperatures and is therefore extended with a simple thermal model. The implementation of which is compared to a *Finite Element* (FE) model. The extrusion of multiple beads is then simulated considering the temperature evolution of the polymer melt. This work is based on the first author's master thesis [16] which contains the source code to reproduce the results presented in its Appendix.

Thermal Model

Temperature Equation. To simplify the implementation of the thermal model, only conductive heat transfer and heat exchange with the environment through convection are considered. Convection and radiation are modeled as heat sources. Other heat sources such as the change in kinetic energy or phase changes are neglected. Furthermore, the specific heat capacity c and the thermal conductivity κ are assumed to be constant, and the flow is assumed to be incompressible.

According to Cleary and Monaghan [17] the heat conduction equation without sources and sinks (Eq. 1):

$$\frac{\mathrm{d}T}{\mathrm{d}t} = \frac{\kappa}{\rho c}\Delta T \tag{1}$$

can be approximated with the SPH method using the Laplacian of Brookshaw [18] as (Eq. 2):

$$\frac{\mathrm{d}T_a}{\mathrm{d}t} = \sum_b \frac{m_b}{c_a \rho_a \rho_b} \frac{4\kappa_a \kappa_b}{(\kappa_a + \kappa_b)} (T_a - T_b) \frac{\boldsymbol{x}_{ab}}{|\boldsymbol{x}_{ab}|^2} \nabla W_{ab}. \tag{2}$$

T denotes the Temperature and ρ the density. Any variables belonging to the central SPH-particle are denoted with the index $(\cdot)_a$ while variables with the index $(\cdot)_b$ belong to a neighboring particle. m is the particle mass, $\boldsymbol{x}_{ij}$ the spatial vector between two particles with number i and j. W_{ij} is the value of the Kernel function for $\boldsymbol{x}_{ij}$. Heat sources can then be added to the right-hand side of the equation.

According to the Stefan-Boltzmann law, the heat exchange with the environment due to radiation for an object is (Eq. 3):

$$P_\mathrm{r} = A\varepsilon\sigma(T_\mathrm{A}^4 - T^4) \tag{3}$$

with the emissivity ε, the Stefan-Boltzmann constant σ, the ambient Temperature T_A and the surface Area A. The heat exchange due to convection can be described using Newton's law of cooling (Eq. 4) as:

$$P_\mathrm{c} = h_\mathrm{c} A(T_\mathrm{A} - T), \tag{4}$$

where h_c is the convective heat transfer coefficient. Adding these sources to the right side of Eq. 2 yields the heat equation (Eq. 5) to solve:

$$\frac{\mathrm{d}T_a}{\mathrm{d}t} = \sum_b \frac{m_b}{c_{p,a} \rho_a \rho_b} \frac{4\kappa_a \kappa_b}{(\kappa_a + \kappa_b)} (T_a - T_b) \frac{\boldsymbol{x}_{ab}}{|\boldsymbol{x}_{ab}|^2} \nabla W_{ab} + \frac{A_a \varepsilon \sigma}{m_a c} (T_A^4 - T_a^4) + \frac{A_a h_\mathrm{c}}{m_a c} (T_A - T_a). \tag{5}$$

Surface Detection and Approximation. SPH-particles generally do not have a surface. To allow for a simple approximation of the heat losses at the surface, it is assumed that a particle can at most represent the surface (Eq. 6):

$$A_\mathrm{max} = 6V^{\frac{2}{3}} \tag{6}$$

of a cube with the same volume V as that particle represents. The surface area A represented by a given particle is calculated using the inverse of the renormalization-matrix $\boldsymbol{B}_\mathrm{a}$ (Eq. 7) introduced by Randles and Libersky [19]

$$B_a^{-1} = \sum_b (x_{ba} \otimes \nabla W_{ab})\, V_b \tag{7}$$

as proposed by Doring [20]. The smallest eigenvalue λ is close to one for particles inside the fluid domain and zero for particles without any neighbors within the radius of the kernel function. Hence, it is a simple indicator of whether a particle is part of the free surface and how much surface area it represents. Marrone et al. [21] found that 0.75 is an adequate cut-off value over which the particle should not be considered part of the free surface.

The described surface detection is applied to a square beam. Figure *1* shows a cross section of the beam where the particles are colored according to λ. As expected, λ is close to 1 for particles further than two layers inside the beam and still higher than the cutoff-value of 0.75 for particles covered by only one layer of particles. The particles forming the surface are consistently assigned values smaller than the cutoff with the particles at the edges being assigned the smallest values.

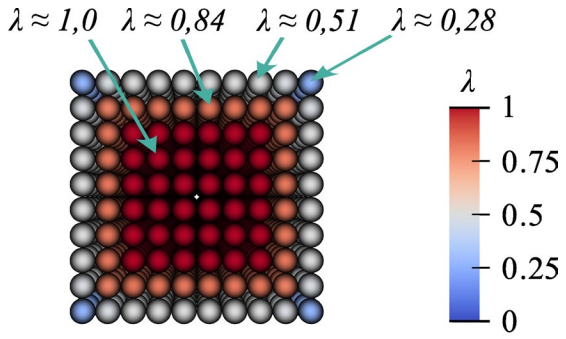

Figure 1 – Application of the surface detection on a square beam.

For simplicity, a linear relation between λ and the surface area of the particle is assumed. The surface area of a particle which is assigned $\lambda < 0.75$ can therefore be calculated as (Eq. 8):

$$A_a = \left(1 - \frac{4}{3}\lambda_a\right) A_{\max}. \tag{8}$$

The approximation is shown in Figure *2* together with value pairs of particles taken from the square beam example above. The exemplary values correspond to the particles marked in Figure *1* with the addition of a particle sitting at the vertex of the beam. A particle situated at the surface but not at the edge of the beam should represent a sixth of $A_{\max}$. A particle at the edge should represent a third of $A_{\max}$ and one at the vertex should represent a half of $A_{\max}$. The approximation used does ensure that particles without neighbors represent the maximum possible surface area while particles inside the fluid represent none. However, in this example, the surface area of the particles sampled from the square beam is severely overestimated.

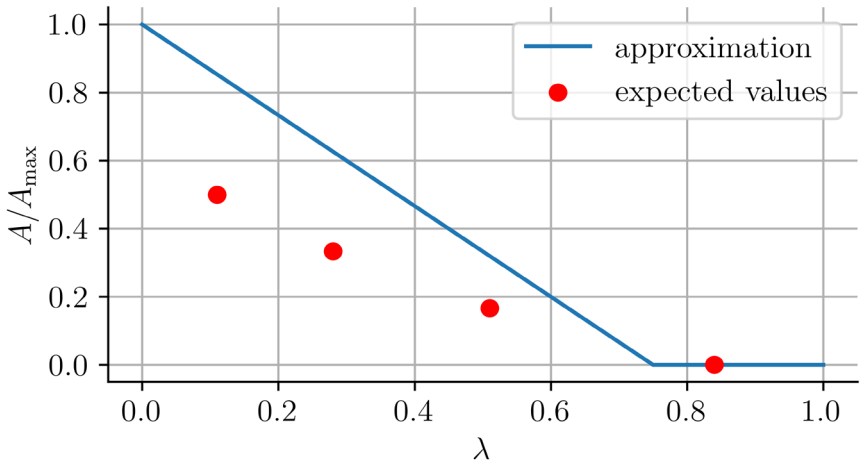

Figure 2 – Assumed relative free surface of particles compared to the used approximation.

Validation of the Thermal Model. To validate the thermal model, a test case is compared to a solution from the commercial finite element solver ABAQUS using DC3D8-elements. The test case consists of a square Aluminum beam with $0.02\ \mathrm{m}$ long sides and a length of $z_{\max} = 0.2\ \mathrm{m}$ to which an initial inhomogeneous temperature distribution along its z-axis (Eq. 9):

$$T(z) = (2500(z - 0.05)^2 + 150)\ °\mathrm{C} \tag{9}$$

is applied. The resulting temperature distribution is compared after 50 seconds.

The two models show good agreement if only conductive heat transfer is considered as shown in Figure *3*. However, the results differ significantly when heat transfer at an ambient temperature of $T_\mathrm{A} = 20\ °\mathrm{C}$ is considered, as shown by the dashed lines. The convective heat transfer coefficient was set to $h_\mathrm{c} = 100\ \frac{\mathrm{W}}{\mathrm{m^2K}}$ and the emissivity to $\varepsilon = 1$. While the final temperature distribution has a similar shape in both models, the newly implemented thermal model significantly overestimates the heat losses to the environment. The overestimation of the free surface contributes to the error significantly. Additionally, the implementation assumes the temperature of any particle at the surface to be the surface temperature. This assumption adds to the error, as the particle temperature represents the temperature within the continuum.

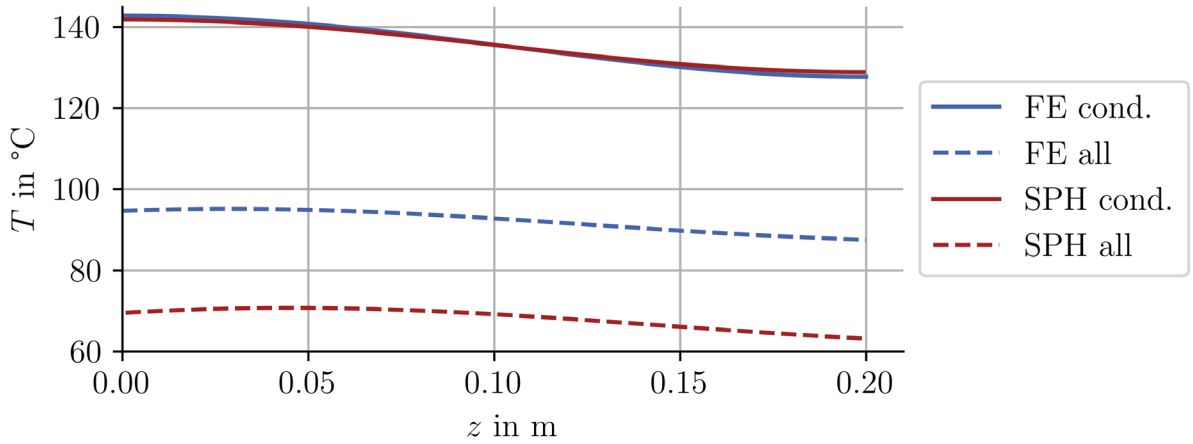

Figure 3 – Final temperature distribution in the FE and SPH models along the centerline of the beam with and without heat transfer through radiation and convection.

Model of the Material Extrusion Process

The domain of the numerical model contains only a build plate and a particle inlet representing the machine's nozzle. The build plate consists of three layers of *ghost* particles for which the

conservation equations do not need to be solved. During the simulation, particles flow into the domain from the inlet.

Through a slight modification, the inlet can be moved according to a user-defined function. The material parameters of the polylactic acid (PLA) melt are taken from the material database of the software AUTODESK MOLDFLOW 2021 [22]. The emissivity is set to $\varepsilon = 0.98$ and the convective heat transfer coefficient to $h_c = 100\ \frac{W}{mK}$. The velocity of the nozzle and the particles flowing into the domain are both set to $v = 60\ \frac{mm}{s}$.

PYSPH offers various *schemes* that set the equations to be solved, the kernel function as well as the type of integration. Running the model for the deposition of a single bead using various available explicit schemes shows the *Entropically Damped Artificial Compressibility* (EDAC) scheme is best suited for the case. A promising implicit scheme could not be used due to instability. The remaining explicit schemes compared unfavorably to the EDAC scheme when extruding in a single bead. The comparison of the available schemes is discussed in more detail in the original thesis [16]. The EDAC scheme is based on the work of Clausen [23] and was later adapted and implemented in PySPH by Ramachandran and Puri [24]. In contrast to many other schemes, the pressure p is not determined by a state equation but by the *EDAC-equation* (Eq. 10):

$$\dot{p} = -\rho c_s^2 \nabla \cdot \boldsymbol{v} + \nu_{EDAC} \nabla^2 p, \tag{10}$$

where $\boldsymbol{v}$ is the fluid velocity, c_s is an artificial speed of sound usually set to a multiple of the maximum fluid velocity and ν_{EDAC} is an artificial viscosity diffusing the pressure [24].

Results and Discussion

Due to stability constraints, the artificial speed of sound that should ensure the assumed incompressibility of the fluid is set to $c_s = 6\ \frac{m}{s}$. Similarly, the model is unstable for the high viscosities typical for the melt. The viscosity is therefore set to $\eta = 0.1$ Pa s. Furthermore, the nozzle is set to a height equal to its diameter, as lower heights will again lead to instability. Figure 4 shows a cross-section of the deposition of a single bead. The particles are colored according to the density with the fluid's rest density set to white. It is clearly visible that the assumed compressibility is not fulfilled, as the density varies significantly within the bead. The bead also varies in thickness while the width is almost constant except for the start point on the left where it is more severely deformed. The particles are also further spread out in this area. The adjacent region immediately to the right by contrast shows visibly smaller particle distances and higher density.

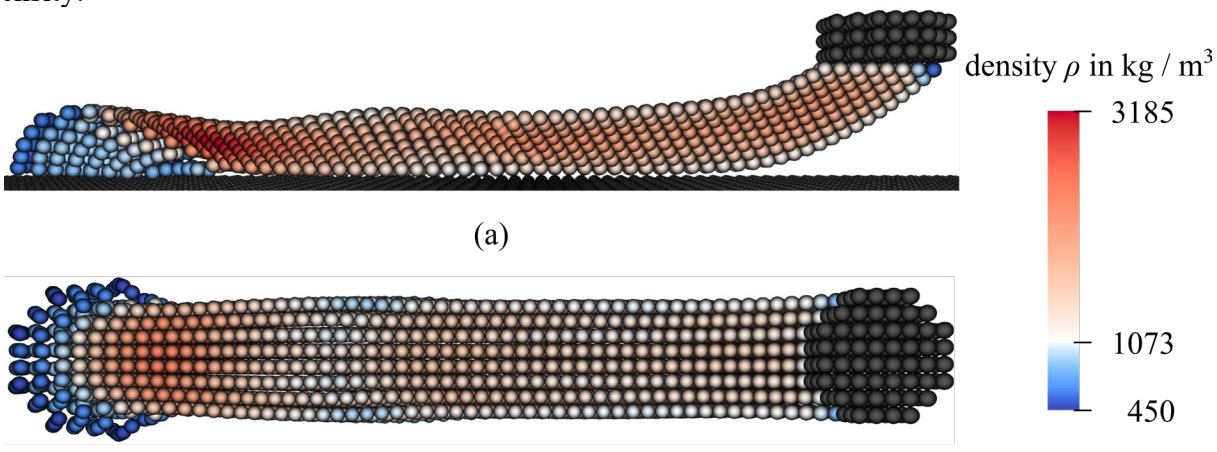

Figure 4 – Deposition of a single bead as a cross section from the side in (a) and in a top-down view in (b). The particles are colored according to density.

Figure 5 shows the deposition of three layers each consisting of three parallel short beads. While the bead remains reasonably well-defined when extruding only a single short bead, the low viscosity causes the beads to lose their shape under gravity after only a short amount of time as shown in Figure *5* (a) where only the first layer is deposited completely. After the extrusion of another two layers in Figure *5* (b), the structure is severely deformed under its own weight. As within a single bead, the fluid density varies greatly throughout the domain. The second and third layer lose their shape quicker than the first as they are deposited onto an already deformed surface that sits lower compared to the nozzle. As the fluid flows to the sides, large areas of low density form.

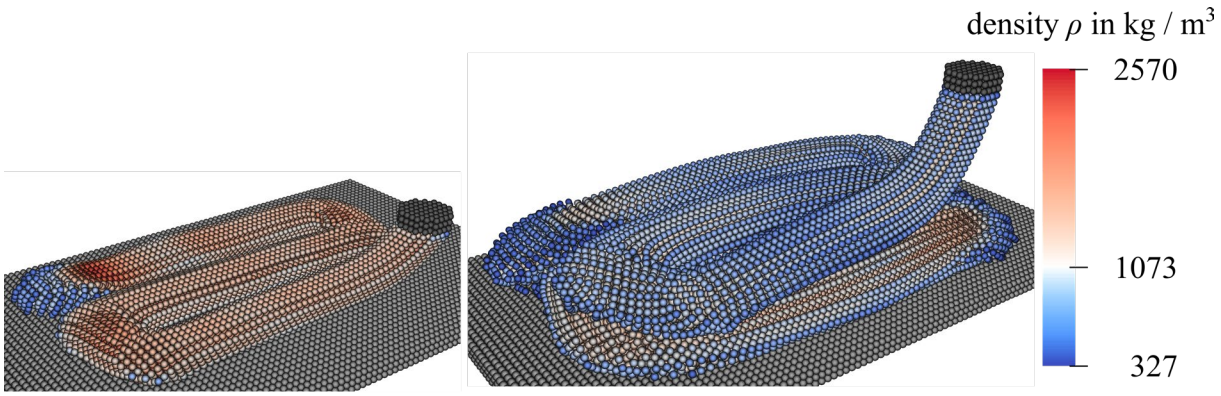

Figure 5 – Deposition of nine beads after depositing the first (a) and the third layer (b). The particles are colored according to the density ρ.

The cross section of nine deposited beads normal to the direction of extrusion in Figure *6* confirms that the particles are too tightly packed in some regions and too loosely in others. Overall, the beads can barely be distinguished as the low viscosity allows the fluid to rapidly flow to the sides. The lower two layers appear to be squished by the fluid above leading to regions of very high density forming. Furthermore, the particles partly retain the order with which they are extruded and form distinct slats in which their vertical distances are far too small, but their horizontal distances are too large. These slats seem to remain intact as the space between them greatly increases. This is a numerical artifact of the method previously described in literature [25].

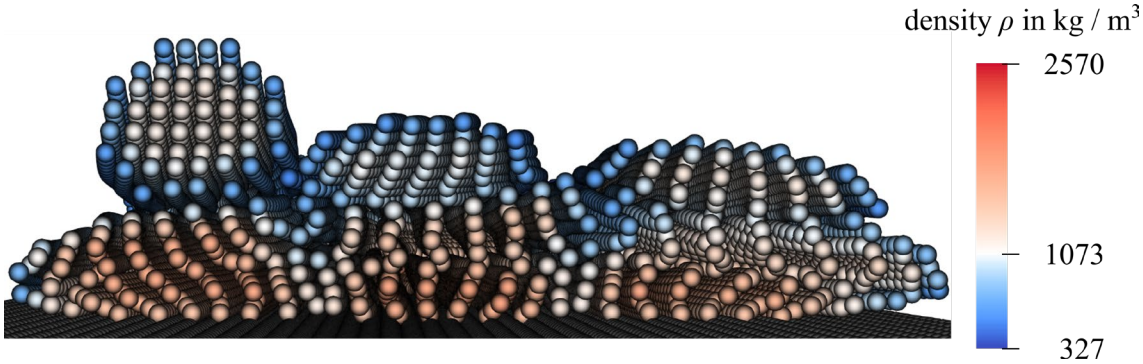

Figure 6 – Cross section normal to the extrusion direction through nine deposited beads. The particles are colored according to the density ρ.

The described arrangement of the particles leads to issues with the surface detection algorithm. Figure *7* again shows the cross section normal to the extrusion direction with the particles colored according to their λ value used to determine the amount of free surface they represent. While the implemented method can identify surface particles correctly in regions of even particle spacing, there are regions of clearly misidentified particles where the aforementioned slats have large gaps

between them. These gaps widen significantly as the top as the low-viscosity-material flows to the side due to gravity. In areas where this artifact is less pronounced, e. g. the bead currently being deposited on the upper left, the detection of the surface particles seems to work correctly. The overall accuracy of the surface detection cannot be easily quantified for this more complex case, but the results from the test case in the previous section do strongly suggest that the surface area will per particle is again significantly overestimated.

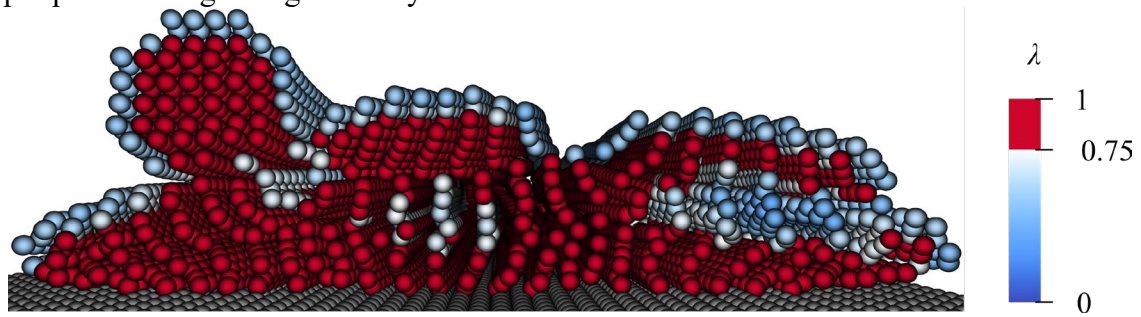

Figure 7 – Cross section normal to the extrusion direction through nine deposited beads. The particles are colored according to the indicator of surface area λ with the red particles not being part of the free surface.

The temperature field mostly agrees with expectations, as shown in Figure *8*. The first layer rapidly cools from the extrusion temperature set to a typical $T_{\mathrm{Ex}} = 220\ °\mathrm{C}$ due to the conductive heat loss to the steel build plate with a fixed temperature of $T_{\mathrm{BP}} = 55\ °\mathrm{C}$. The second and third layer cool slower as the conduction within the fluid is low. The effect of the heat losses at the surface are therefore more pronounced in the upper layers.

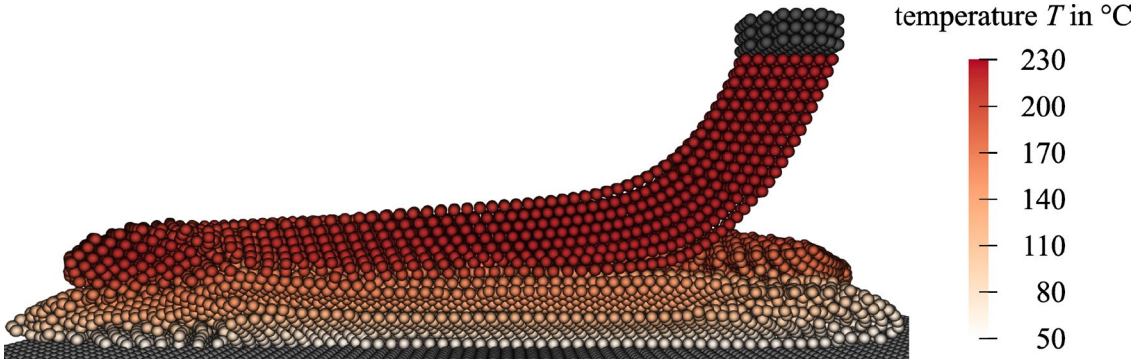

Figure 8 – Cross section through the extrusion of nine beads with the particles colored according to their temperature T.

While the misidentified surface particles undoubtedly lead to some additional cooling within the structure, the effect is somewhat mitigated due to the particles being labeled mostly correct when the material is exiting the nozzle and is therefore the hottest. The regions of misidentified surface particles are therefore not clearly visible in Figure *9* showing the same cross section as Figure *7* but with the particles colored according to their temperature. This is despite a relatively high convective heat transfer coefficient $h_{\mathrm{c}} = 100\ \frac{\mathrm{W}}{\mathrm{m^2K}}$ and emissivity $\varepsilon = 0.98$.

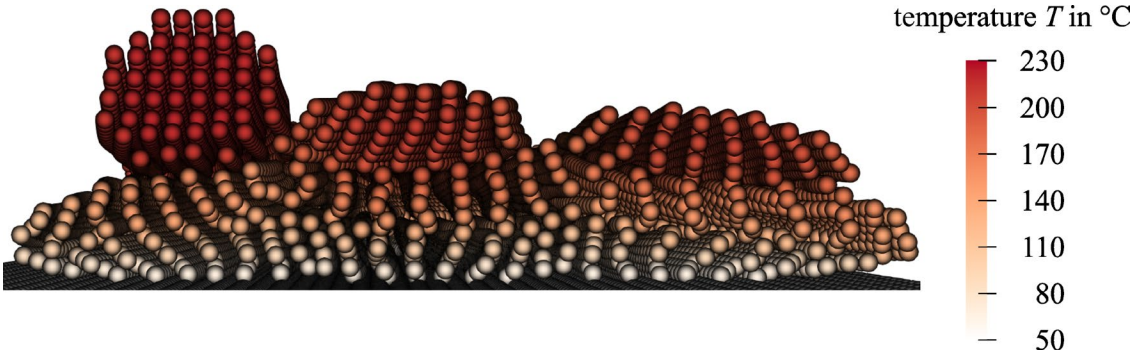

Figure 9 – Cross section normal to the extrusion direction through nine deposited beads. The particles are colored according to their temperature T.

Conclusion

A model of the deposition of single as well as multiple beads onto the build plate is presented and examined. The PySPH framework is extended by a thermal model allowing to model not only conductive heat exchange but also heat losses due to radiation and convection at the surface. While the conductive flow of heat agrees very well with a FE model, the heat losses at the surface are significantly overestimated, mostly due to an inaccuracy of the surface detection algorithm. An improvement of this algorithm would therefore directly improve the prediction of the thermal history of the bead.

Since the chosen *EDAC*-scheme cannot achieve the assumed incompressibility and high viscosity, the overall goal of accurately predicting the formation of a typical mesostructure is not achieved. Instead, the deposited structure is deformed severely. Furthermore, the formation of undesired slats within the extrudate is observed. These issues must be overcome to make useful deductions from the model. Other SPH formulations might be able to more accurately capture the behavior of the polymer melt. None of the other schemes built into the software seem to be able to capture the process more adequately. To address this in future work, another solver should be chosen, or a new scheme should be added to PySPH.

Furthermore, it is desirable to increase the accuracy of the inflow boundary condition by incorporating the real movement of the nozzle that currently moves with a constant speed. The Python-package pyGCodeDecode [26] allows to simulate the firmware's interpretation and realization of the G-code and will therefore be used in future work.

References

[1] I. Gibson, D. Rosen, B. Stucker, Additive Manufacturing Technologies, Springer New York, New York, NY, 2015. https://doi.org/10.1007/978-1-4939-2113-3

[2] O. Luzanin, D. Movrin, V. Stathopoulos, P. Pandis, T. Radusin, V. Guduric, Impact of processing parameters on tensile strength, in-process crystallinity and mesostructure in FDM-fabricated PLA specimens, Rapid Prototyp. J. 25 (2019) 1398–1410. https://doi.org/10.1108/RPJ-12-2018-0316

[3] F. Frölich, L. Bechtloff, B.M. Scheuring, A.L. Heuer, F. Wittemann, L. Kärger, W.V. Liebig, Evaluation of mechanical properties characterization of additively manufactured components, Prog. Addit. Manuf. (2024). https://doi.org/10.1007/s40964-024-00700-2

[4] L.B. Lucy, A numerical approach to the testing of the fission hypothesis, Astron. J. 82 (1977) 1013. https://doi.org/10.1086/112164

[5] R.A. Gingold, J.J. Monaghan, Smoothed particle hydrodynamics: theory and application to non-spherical stars, Mon. Not. R. Astron. Soc. 181 (1977) 375–389. https://doi.org/10.1093/MNRAS/181.3.375

[6] D. Hesse, P. Hohenberg, M. Stommel, Z. Kunststofftechnik, Entwicklung eines rechnergestützten Ansatzes zur Abbildung des Strangablegens in der extrusionsbasierten additiven Fertigung, Z. Kunststofftechnik J. Plast. Technol. 17 (2021) 3

[7] H. Xia, J. Lu, G. Tryggvason, Simulations of fused filament fabrication using a front tracking method, Int. J. Heat Mass Transf. 138 (2019) 1310–1319. https://doi.org/10.1016/j.ijheatmasstransfer.2019.04.132

[8] B. Šeta, M. Sandberg, M. Brander, Md.T. Mollah, D.K. Pokkalla, V. Kumar, J. Spangenberg, Numerical modeling of fiber orientation in multi-layer, isothermal material-extrusion big area additive manufacturing, Addit. Manuf. 92 (2024) 104396. https://doi.org/10.1016/j.addma.2024.104396

[9] M.P. Serdeczny, R. Comminal, D.B. Pedersen, J. Spangenberg, Numerical simulations of the mesostructure formation in material extrusion additive manufacturing, Addit. Manuf. 28 (2019) 419–429. https://doi.org/10.1016/j.addma.2019.05.024

[10] D. Yang, K. Wu, L. Wan, Y. Sheng, A Particle Element Approach for Modelling the 3D Printing Process of Fibre Reinforced Polymer Composites, J. Manuf. Mater. Process. 1 (2017) 10. https://doi.org/10.3390/jmmp1010010

[11] E. Bertevas, J. Férec, B.C. Khoo, G. Ausias, N. Phan-Thien, Smoothed particle hydrodynamics (SPH) modeling of fiber orientation in a 3D printing process, Phys. Fluids 30 (2018) 103103. https://doi.org/10.1063/1.5047088

[12] Z. Ouyang, E. Bertevas, D. Wang, B.C. Khoo, J. Férec, G. Ausias, N. Phan-Thien, A smoothed particle hydrodynamics study of a non-isothermal and thermally anisotropic fused deposition modeling process for a fiber-filled composite, Phys. Fluids 32 (2020). https://doi.org/10.1063/5.0004527

[13] M. Makino, D. Fukuzawa, T. Murashima, M. Kawakami, H. Furukawa, Analysis of deposition modeling by particle method simulation, Microsyst. Technol. 23 (2016) 1177–1181. https://doi.org/10.1007/s00542-016-3047-4

[14] P. Pibulchinda, E. Barocio, A.J. Favaloro, R.B. Pipes, Influence of printing conditions on the extrudate shape and fiber orientation in extrusion deposition additive manufacturing, Compos. Part B Eng. 261 (2023) 110793. https://doi.org/10.1016/j.compositesb.2023.110793

[15] P. Ramachandran, A. Bhosale, K. Puri, P. Negi, A. Muta, A. Dinesh, D. Menon, R. Govind, S. Sanka, A.S. Sebastian, A. Sen, R. Kaushik, A. Kumar, V. Kurapati, M. Patil, D. Tavker, P. Pandey, C. Kaushik, A. Dutt, A. Agarwal, PySPH: A Python-based Framework for Smoothed Particle Hydrodynamics, ACM Trans. Math. Softw. 47 (2021) 1–38. https://doi.org/10.1145/3460773

[16] L. Hof, Entwicklung eines Simulationsmodells zur Strangablage in der Materialextrusion mit der Smoothed Particle Hydrodynamics Methode, Masterarbeit, Karlsruher Institut für Technologie (KIT), 2022

[17] P.W. Cleary, J.J. Monaghan, Conduction Modelling Using Smoothed Particle Hydrodynamics, J. Comput. Phys. 148 (1999) 227–264. https://doi.org/10.1006/jcph.1998.6118

[18] L. Brookshaw, A Method of Calculating Radiative Heat Diffusion in Particle Simulations, Publ. Astron. Soc. Aust. 6 (1985) 207–210. https://doi.org/10.1017/S1323358000018117

[19] P.W. Randles, L.D. Libersky, Smoothed Particle Hydrodynamics: Some recent improvements and applications, Comput. Methods Appl. Mech. Eng. 139 (1996) 375–408. https://doi.org/10.1016/S0045-7825(96)01090-0

[20] M. Doring, Développement d'une méthode SPH pour les applications à surface libre en hydrodynamique, These de doctorat, Nantes, 2005. https://www.theses.fr/en/2005NANT2116 (accessed September 23, 2022).

[21] S. Marrone, A. Colagrossi, D. Le Touzé, G. Graziani, Fast free-surface detection and level-set function definition in SPH solvers, J. Comput. Phys. 229 (2010) 3652–3663. https://doi.org/10.1016/j.jcp.2010.01.019

[22] Autodesk Inc., Moldflow Insight Help | About material databases, Autodesk Moldflow Insight 2021 (2021). https://help.autodesk.com/view/MFIA/2021/ENU/?guid=GUID-9C852155-1ECC-4A68-A45D-F3FC5E79E057 (accessed September 21, 2022).

[23] J.R. Clausen, Entropically damped form of artificial compressibility for explicit simulation of incompressible flow, Phys. Rev. E 87 (2013) 013309. https://doi.org/10.1103/PhysRevE.87.013309

[24] P. Ramachandran, K. Puri, Entropically damped artificial compressibility for SPH, Comput. Fluids 179 (2019) 579–594. https://doi.org/10.1016/j.compfluid.2018.11.023

[25] G. Oger, S. Marrone, D. Le Touzé, M. de Leffe, SPH accuracy improvement through the combination of a quasi-Lagrangian shifting transport velocity and consistent ALE formalisms, J. Comput. Phys. 313 (2016) 76–98. https://doi.org/10.1016/j.jcp.2016.02.039

[26] J. Knirsch, F. Frölich, L. Hof, F. Wittemann, L. Kärger, pyGCodeDecode: A Python package for time-accurate GCode simulation in material extrusion processes, J. Open Source Softw. 9 (2024) 6465. https://doi.org/10.21105/joss.06465