ADVANCED THEORY AND SIMULATIONS
www.advtheorysimul.com

# Peridynamic Framework to Model Additive Manufacturing Processes

*Christian Willberg,* Jan-Timo Hesse, Felix Winkelmann, and Robert Hein*

**The study presents a framework for analyzing Additive Manufacturing processes within the Peridynamics (PD) software `PeriLab`. This framewor k employs a mesh-free, point-based numerical approach to approximate the continuum PD equations. Implemented within this framework are thermal, thermo-mechanical, and simple additive models. These models have been validated against analytical solutions, Finite Element (FE) models, and `Peridigm` simulations. To leverage the PD mesh-free implementation, the study introduces a novel boundary detection algorithm. This algorithm is essential because the outer surface area may change during the manufacturing process. It operates without requiring surface or topology information, relying instead on the comparison of neighborhood volume to sphere volume. Additionally, the study introduces a wrapper that generates the mesh necessary for simulating the printing process, based on the G-code machine input path. Finally, the study presents a comprehensive analysis of an L-shaped profile utilizing the developed features, comparing the results with those obtained from an `Abaqus` solution.**

## 1. Introduction

Additive manufacturing allows the tailored production of parts without the need for a mold. It is an established process to manufacture small batches or prototypes.[1] In this paper, the focus is on the Fused Filament Fabrication process. In this process, a polymer is melted and deposited through a nozzle. The nozzle moves above the print bed and builds up the part layer-by-layer.[2] The process parameters, such as extrusion temperature and deposition rate, have a crucial effect on the final material and component properties and their structural mechanical performance.[3] The major problem is that final material properties will evolve during the printing process itself, and representative testing in advance to obtain the strength of the material is challenging. The close material-process-component interactions lead to non-negligible variations in component properties, which need to be considered.

One promising approach is to simulate the manufacturing process of test specimens and then test them virtually. For validation purposes, test coupons can be produced with a variety of process parameters and tested experimentally. Due to their simple geometry and manufacturability, sufficient samples can be produced and evaluated stochastically. Based on this, the deterministic process simulation can be extended with probabilistic parameters and can be correlated to the experimental tests. The statistically validated simulation process can be used to predict the failure behavior of more complex structures. The first step to reaching this goal is to realize the printing process using a modeling approach that also allows failure analysis. Commercial Finite Element Method (FEM) vendors, such as Ansys or `Abaqus`, provide already-specific solutions for additive manufacturing simulation. However, the existing tools have different limitations and time-consuming working steps to set up the simulation. In `Abaqus`, for example, both the tool-path and the Computer Aided Design (CAD) file are required. The last mentioned is not always accessible. Additionally, the CAD needs to be aligned with the G-code, which is often not the case because the G-code and CAD parts operate with different coordinate systems and the alignment needs to be done manually. Furthermore, the pre-processing, i.e., the preparation of the geometry as well as the meshing and the application of boundary conditions, can be very time-consuming.

PD is a very promising approach to solving the presented idea and overcoming some of its limitations. Also, a more time-efficient model setup could be realized through model generation directly and solely from the G-code.

PD was introduced in the early 2000s by Stewart Silling and has been continuously developed since then. The main advantage of the method is that there are no singularities in the region of damage, as in classical continuum mechanics.[4] This is achieved by an integral description of the conservation of momentum. The disadvantage is that an integration domain must

C. Willberg[+]
Magdeburg-Stendal University of Applied Science
Breitscheitstraße 2, 39114 Magdeburg, Germany
E-mail: christian.willberg@h2.de
C. Willberg[+], J.-T. Hesse, F. Winkelmann, R. Hein
German Aerospace Center
Lilienthalplatz 7, 38108 Braunschweig, Germany

The ORCID identification number(s) for the author(s) of this article can be found under https://doi.org/10.1002/adts.202400818

[+]Present address: Breitscheidstrasse 2, 39114 Magdeburg, Germany



**DOI: 10.1002/adts.202400818**

be defined. A lot of research has been done in the last decade, and for more complex large-scale PD problems, the software `Peridigm` was developed.[5–9] To benefit from its advantages, it was used to implement an additive manufacturing process, and the first results were shown in ref. [10]. However, the effort required for implementation in `Peridigm` was very substantial. Also, if other people want to use the code, the installation process is very challenging. As a result, the new framework `PeriLab` was developed.[11,12] It is based on the modern Julia language and simplifies massively the installation and development process. The paper shows model verification and the AM process in `PeriLab` in more detail compared.[10] The numerical computations in `PeriLab` are compared to the classical approach in FEM, analytical solutions and `Peridigm` results.

The objective of this paper is to introduce thermal, thermomechanical and process related models in detail for simulating the process of additive manufacturing, excluding the specific material behavior like solidification of filaments. The code is verified to analytical, finite element solutions as well as the `Peridigm` implementation for a simple plate and for a bracket. A very good agreement was found between the PD solution, the analytical solution, and the FEM solution. Therewith, the PD framework was extended with a solution for AM process simulation. Finally, further research on this topic will be discussed.

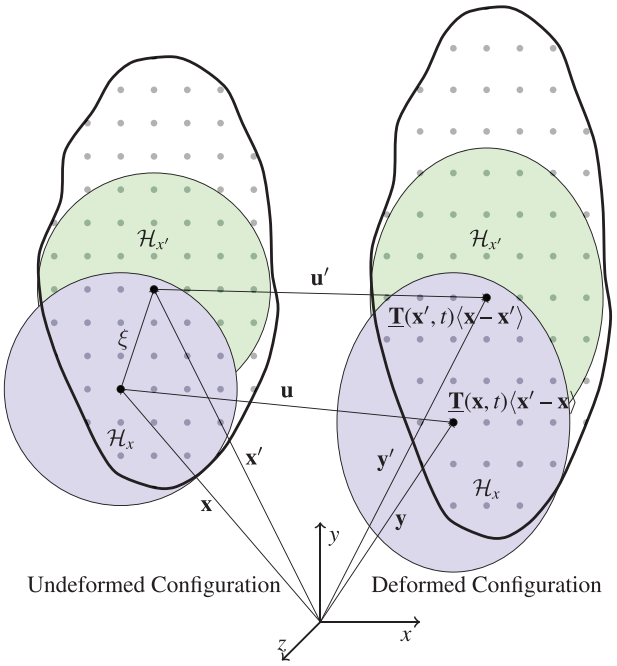



**Figure 1.** Deformation and interaction of material points $\mathbf{x}$ and $\mathbf{x}'$ in the non-ordinary state-based theory and their respective neighborhoods $\mathcal{H}_{\mathbf{x}}$ and $\mathcal{H}_{\mathbf{x}'}$.

## 2. Peridynamic Modeling

### 2.1. Mechanical Model

The paper follows the assumptions and notations from Silling.[4] Within the neighborhood, cf. **Figure 1** $\mathcal{H}$, with the volume $V_{\mathbf{x}}$ and the external force term $\mathbf{b}$, defined by a spherical domain the horizon $\delta$, the force volume density state $\underline{\mathbf{T}}$ for the bond interaction between the positions $\mathbf{x}$ and $\mathbf{x}'$ is defined as the integral balance of momentum, which is equal to the product of mass density $\rho$ and acceleration $\ddot{\mathbf{u}}$:

$$\int_{\mathcal{H}} (\underline{\mathbf{T}}(\mathbf{x}, t)\langle \mathbf{x}' - \mathbf{x} \rangle - \underline{\mathbf{T}}(\mathbf{x}', t)\langle \mathbf{x} - \mathbf{x}' \rangle) dV_{\mathbf{x}} + \mathbf{b} = \rho \ddot{\mathbf{u}} \tag{1}$$

Three variations of the PD model are currently being used, the bond-based, the ordinary state-based and the non-ordinary state-based formulation. In this order, flexibility increases, but so does the complexity of the formulations.

A special formulation of the non-ordinary state-based model was introduced by Silling et al.[4] in 2007. This so-called "correspondence" formulation defines an integral non-local deformation gradient

$$\mathbf{F} = \left[ \int_{\mathcal{H}} \underline{\omega}\langle \boldsymbol{\xi} \rangle \underline{\mathbf{Y}}\langle \boldsymbol{\xi} \rangle \otimes \underline{\mathbf{X}}\langle \boldsymbol{\xi} \rangle dV_{\mathbf{x}} \boldsymbol{\xi} \right] \cdot \mathbf{K}^{-1} \tag{2}$$

to calculate the bond force vector density states, where

$$\mathbf{K} = \int_{\mathcal{H}} \underline{\omega}\langle \boldsymbol{\xi} \rangle \underline{\mathbf{X}}\langle \boldsymbol{\xi} \rangle \otimes \underline{\mathbf{X}}\langle \boldsymbol{\xi} \rangle dV_{\mathbf{x}} \tag{3}$$

is the shape tensor with $V_{\mathbf{x}}$ defined as the neighborhood volume. For each bond $\boldsymbol{\xi} = \mathbf{x}' - \mathbf{x}$, there is an influence function $\underline{\omega}\langle \boldsymbol{\xi} \rangle$, an undeformed vector state $\underline{\mathbf{X}}\langle \boldsymbol{\xi} \rangle = \mathbf{x}' - \mathbf{x}$ and a deformed vector state $\underline{\mathbf{Y}}\langle \boldsymbol{\xi} \rangle = \mathbf{y}' - \mathbf{y}$. The shape tensor is positive definite and symmetric. If in numerical applications damages are considered the shape tensor is only positive semi-definite and depended on the bonds which are broken not invertible. The advantage using the non-local deformation gradient is that classical continuum mechanical models can be used in PD. The PD force density vector state is thus:

$$\underline{\mathbf{T}}\langle \boldsymbol{\xi} \rangle = \underline{\omega}\langle \boldsymbol{\xi} \rangle \mathbf{P} \mathbf{K}^{-1} \boldsymbol{\xi} \tag{4}$$

The Piola–Kirchhoff stress tensor $\mathbf{P}$ with respect to an orthonormal basis and the non-local deformation gradient $\mathbf{F}$ can be determined as

$$\mathbf{P} = \det \mathbf{F} \boldsymbol{\sigma} \mathbf{F}^{-1} \tag{5}$$

In this benchmark we deal with a linear elastic material. In this case the Cauchy stress $\boldsymbol{\sigma}$ can be derived using Hook's law as:

$$\boldsymbol{\sigma} = \mathbf{C} \cdot \cdot \boldsymbol{\varepsilon}_{mechanical} \tag{6}$$

with the fourth order elasticity tensor $\mathbf{C}$ and the Green–Lagrange strain tensor defined as:

$$\boldsymbol{\varepsilon}_{mechanical} = \frac{1}{2}(\mathbf{F}^T \mathbf{F} - \mathbf{I}) \tag{7}$$

For correspondence models, the so called zero-energy modes could occur.[13] These modes are non-physical and lead to unstable or unreasonable solutions. Several stabilization methods

were published to overcome this problem.[14–19] A promising approach was published by Wan et al. in 2019.[20] Instead of a bond-based stabilization method proposed by Silling[21] Wan et al. developed a state-based stabilization method. As positive side effect this method stabilizes the solution for anisotropic material as well. The corrected force density state $\underline{\mathbf{T}}^C$ with suppression of the zero-energy mode is:

$$\underline{\mathbf{T}}^C = \underline{\mathbf{T}} + \underline{\mathbf{T}}^S \tag{8}$$

where $\underline{\mathbf{T}}$ is given in Equation (4). Following Wan et al.[20] the suppression force density state $\underline{\mathbf{T}}^S$ is:

$$\underline{\mathbf{T}}^S\langle\boldsymbol{\xi}\rangle = \underline{\omega}\langle\boldsymbol{\xi}\rangle\mathbf{C}_1\underline{\mathbf{z}} \tag{9}$$

with $\underline{\mathbf{z}}$ as the non-uniform deformation state

$$\underline{\mathbf{z}}\langle\boldsymbol{\xi}\rangle = \underline{\mathbf{Y}}\langle\boldsymbol{\xi}\rangle - \mathbf{F}\boldsymbol{\xi} \tag{10}$$

caused by the zero-energy mode. If the approximated non-local deformation gradient $\mathbf{F}$ exactly maps each undeformed bond to the deformed configuration no zero-energy mode occur. In that case the non-uniform deformation state is zero and the corrected force density state $\underline{\mathbf{T}}^C$ is equal to the force density state $\underline{\mathbf{T}}$. The second order tensor $\mathbf{C}_1$ is given as

$$\mathbf{C}_1 = \mathbf{C} \cdot\cdot \mathbf{K}^{-1} \tag{11}$$

utilizing the elasticity tensor from Equation (6).

### 2.2. Thermo-Mechanics

To introduce a thermo-mechanical coupling the mechanical strains in Equation (7) have to be expanded with the thermal strains:

$$\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}_{mechanical} + \boldsymbol{\varepsilon}_{thermal} \tag{12}$$

The thermal strains are defined as

$$\boldsymbol{\varepsilon}_{thermal} = -\boldsymbol{\alpha}\tau \tag{13}$$

with $\tau$ as the temperature increment and $\boldsymbol{\alpha}$ as matrix of the coefficients of thermal expansion. Typically, this matrix is diagonal. The coupled stresses for linear elastic material utilizing Equation (6) is given as

$$\boldsymbol{\sigma} = \mathbf{C} \cdot\cdot \left(\boldsymbol{\varepsilon}_{mechanical} - \boldsymbol{\alpha}\tau\right) \tag{14}$$

### 2.3. Thermal Flux

The mechanical response due to temperature changes is included in the PD model. However, the heat flux must be included as well. The following derivation is based on refs. [22, 23]. Under the assumption that mechanical deformations do not change the temperature, the thermodynamic equilibrium equation can be studied separately to Equation (1).

$$\rho C_v \dot{\tau} = \int_{\mathcal{H}} \left(\underline{h}(\mathbf{x}, t)\langle\boldsymbol{\xi}\rangle - \underline{h}(\mathbf{x}', t)\langle\boldsymbol{\xi}'\rangle\right) dV_{\mathbf{x}} + S_i \tag{15}$$

On the contrary to Equation (1) which is a second order differential equation it is first order. Later on, this simplifies the numerical solving process. The parameters are $\rho$ the mass density, $C_v$ the specific heat capacity, $\dot{\tau}$ the temperature gradient in time, $dV_{\mathbf{x}}$ the volume and $S_i$ the heat sink or heat source. The heat flux of a bond is defined as

$$\underline{h}(\mathbf{x}, t)\langle\boldsymbol{\xi}\rangle = \mathbf{q}^T\mathbf{K}^{-1}(\mathbf{x})\boldsymbol{\xi} \tag{16}$$

with $\mathbf{q}$ as classical heat flux and $\mathbf{K}$ as the shape tensor defined in Equation (3). It follows

$$\nabla \cdot \mathbf{q} = \int_{\mathcal{H}} \left[\mathbf{q}(\mathbf{x}')^T\mathbf{K}^{-1}(\mathbf{x}') + \mathbf{q}(\mathbf{x})^T\mathbf{K}^{-1}(\mathbf{x})\right]\boldsymbol{\xi} dV_{\mathbf{x}} \tag{17}$$

which can be derived utilizing the spatial gradient of the temperature $\nabla\tau$ as

$$\mathbf{q} = -\boldsymbol{\lambda}\nabla\tau \tag{18}$$

$\boldsymbol{\lambda}$ is the $3 \times 3$ matrix of thermal conductivity. Typically, it is a diagonal matrix. Following[23] the spatial temperature gradient $\nabla\tau$ can be derived as

$$\nabla\tau = \mathbf{K}^{-1} \int_{\mathcal{H}} \left[\tau(\mathbf{x}') - \tau(\mathbf{x})\right]\boldsymbol{\xi}\underline{\omega}\langle\boldsymbol{\xi}\rangle dV_{\mathbf{x}} \tag{19}$$

The numerical solving process is then

$$\rho C_v \frac{\tau^{t+dt} - \tau^t}{dt} = \nabla \cdot \mathbf{q} + S_i \tag{20}$$

$$\tau^{t+dt} = dt\frac{\nabla \cdot \mathbf{q} + S_i}{\rho C_v} + \tau^t \tag{21}$$

### 2.4. Heat Transfer to Environment

Following[24,25] the heat volumetric density at the surface for an assigned heat flux normal to the surface $q_{bc}$ is:

$$S_i = \frac{q_{bc}}{\Delta} \tag{22}$$

where $\Delta$ can be set to $dx$. Thereby, $q_{bc}$ is

$$q_{bc} = \kappa(\tau - \tau_{env}) \tag{23}$$

where $\kappa$ is the heat convection coefficient between solid and environment and $\tau_{env}$ the environmental temperature. For a mesh free model, the question arises how the outer surface and the corresponded surface can be identified. For the outer surface the PD neighborhood $\mathcal{H}$ is utilized. It is assumed that is circle for 2D and a sphere for 3D. Therefore, the following criteria has to be fulfilled for 2D

$$V_{2D} = 2\pi\delta^2 h \geq \int_{\mathcal{H}} dV \tag{24}$$

and 3D

$$V_{3D} = \frac{4}{3}\pi\delta^3 \geq \int_{\mathcal{H}} dV \tag{25}$$

Each point which is next to the surface will have less volume represented be the discrete material points. Defining a limit value

$$f_{limit} \leq V_{specific} = \frac{\int_{\mathcal{H}} dV}{V_{2D\,or\,3D}} \tag{26}$$

allows an easy identification of surface nodes $i$ during the printing process. Combining Equations (21) and (22) allows the calculation of the change in temperature for these nodes $i$ as

$$\tau_i^{t+dt} = dt \frac{\nabla \cdot \mathbf{q}_i + \frac{\kappa(\tau_i^t - \tau_{env})}{dx}}{(\rho C_v)_i} + \tau_i^t \tag{27}$$

The minimum time step for the explicit time integration of the temperature field to obtain a stable solution is given by

$$\Delta t < \min\left( \frac{\left(\rho C_v\right)_i}{\sum_{j=1}^{N} \frac{\max(\mathrm{eig}(\lambda))}{|\xi_{ij}|} V_j} \right) \tag{28}$$

with $N$ the number of neighbors of point $i$.[26]

### 2.5. Additive Model

To realize an additive model all bonds from a point are disconnected from its neighbors by setting all $\underline{\omega}\langle\boldsymbol{\xi}\rangle = 0$. Within the mesh input an activation time is specified $t_{activate}$. If this time is reached during the simulation process the point is activated.

$$\underline{\omega}\langle\boldsymbol{\xi}\rangle \in \mathcal{H}_{\mathrm{x}} = \begin{cases} 0 & \text{for } t < t_{activate} \\ 1 & \text{for } t \geq t_{activate} \end{cases} \tag{29}$$

Depending on the process modeled, additional information can be passed to the point or the bonds connected with it. For this simple printing process, a printing temperature is added utilized the heat source $S_i$ introduced in Equation (15). In the presented model, the bonding is ideal an no phase or chemical changes occur. However, in principal such models are applicable. Also, it must be noted, that due to mechanical or thermo-mechanical loading bonds can be damaged, if a damage model is applied. In that case it won't be activated again because of the criterion defined in Equation (29).

Within this process the $t_{activate}$ can be user defined. However, to reproduce real processes an interface with the G-code is needed. This interface provides the information when the tool arrives at a specific point and defines the activation time. The process will be shown in Section 4.1.

## 3. Verification

In this section the implemented models are verified. For this purpose, the following simple model in **Figure 2** has been defined. Generic material parameters are used and listed in **Table 1**. The parameters are specific heat capacity $C_v$, the density $\rho$, the coefficient of thermal expansion $\alpha$ and the heat conductance $\lambda$.

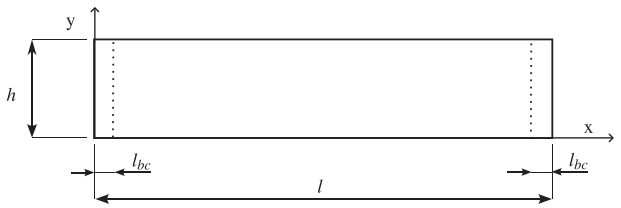



**Figure 2.** Verification model.

### 3.1. Thermal Strain

To verify the thermo-mechanical coupling a constant temperature of $\tau_{env} = 2 \times 10^{-6}$ K has been applied to the whole specimen, which can be displaced freely. Utilizing the simple 1D thermal strain equation

$$\varepsilon_{thermal} = \alpha\tau \tag{30}$$

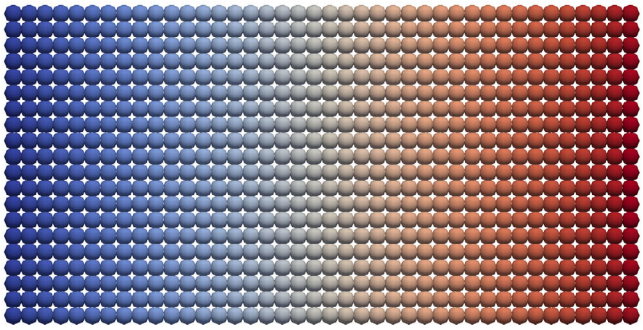
the $u_x$ and $u_y$ displacements in **Figure 3** can be determined, and compared to the numerical PD results in **Table 2**. The numerical results for a horizon of $3dx$ with a discretization length of $5 \times 10^{-4}$ m are shown exemplary in Figure 3a for $u_x$ displacement and Figure 3b for $u_y$ displacement for the `Peridigm` solution, respectively. For the `PeriLab` framework the result looks identical and is therefore not shown. Compared to the analytical solution the difference is lower than 3 % for all simulations in `Peridigm` and `PeriLab`. The values are shown in Table 2. For finer discretization's the error will shrink. It must be noted, that a large numerical damping is needed, because by applying a constant temperature the model starts to vibrate, dominating the deformations compared to the small deformations caused by the temperature change. Therefore, the error varies by changing the damping parameter.

### 3.2. Heat Flow

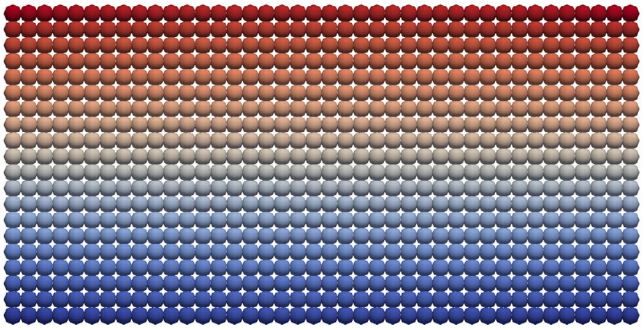
To verify the heat flow model a constant temperature at left-hand side of $\tau = 10$ K and a zero temperature of $\tau = 0$ K at the right-hand side have been applied. The parameters are given in Table 1. It is expected, that after some time the heat flow will be in a steady state resulting into a linear temperature profile between the boundary conditions. **Figure 4** shows the `PeriLab` and `Peridigm` temperature distribution in comparison to the analytical solution. After $1 \times 10^{-4}$ s the temperature distribution over the length of the model for the numerical analyses is nearly linear. The bond-based variant shown in the Figure brings a better result for `Peridigm` and `PeriLab`. For the correspondence formulation more points are needed at the boundaries to avoid

**Table 1.** Parameter for isotropic thermal model used for verification.

| Parameter | Value | Unit |
|---|---|---|
| $C_v$ | 0.33 | $\mathrm{Jkg^{-1}\,K^{-1}}$ |
| $\rho$ | $2 \times 10^3$ | $\mathrm{kgm^{-3}}$ |
| $\alpha$ | $2.5 \times 10^{-2}$ | $\mathrm{K^{-1}}$ |
| $\lambda$ | $2 \times 10^3$ | $\mathrm{Wm^{-1}\,K^{-1}}$ |



(a) $u_x$ displacement with $u_{scale}$ = [0 m, 10 × $10^{-10}$ m] calculated in Peridigm.



(b) $u_y$ displacement with $u_{scale}$ = [0 m, 5 × $10^{-10}$ m] calculated in Peridigm.

**Figure 3.** Thermal strain model described in Table 1 with constant temperature $\tau = 2 \times 10^{-6}$ K. Geometry and discretization is given as $l = 1 \times 10^{-2}$ m, $h = 5 \times 10^{-3}$ m, $dx = 5 \times 10^{-4}$ m, $\delta = 1.501 \times 10^{-3}$ m, cf. Figure 2.

over shooting of the temperature. Eventually, a linear distribution will occur.

**Figure 5** shows the change in temperature over time at position $x = 9.2 \times 10^{-3}$ m. It can be seen, that the temperature starts by zero and converges to the steady state temperature of its position, cf. Figure 4. For verification the analytical 1D heat transfer equation described by Fourier's Law is solved.[27]

$$\frac{\partial \tau}{\partial t} = -k \frac{\partial^2 \tau}{\partial x^2} \tag{31}$$

With $k = \frac{\lambda}{\rho C_v} = 3\ \mathrm{m^2\,s^{-1}}$ the and section A, Equation (31) can be solved to create the illustrated curve. Both the analytical reference curve and the PD solutions are in good agreement.

**Table 2.** Comparison of the thermal deformation.

| Displacement | Analytical | Peridigm | PeriLab |
|---|---|---|---|
| $u_x$ | $10 \times 10^{-10}$ m | $9.76 \times 10^{-10}$ m | $9.81 \times 10^{-10}$ m |
| $u_y$ | $5 \times 10^{-10}$ m | $4.9 \times 10^{-10}$ m | $4.98 \times 10^{-10}$ m |

### 3.3. Cooling

For the verification of the heat flow to environment the following model was set up in PD, cf. **Figure 6a**. In parallel the model was also set-up in the commercial FE Abaqus from Dassault systems. The material properties are chosen such, that rapid cooling is possible ($\rho = 1 \times 10^3\ \mathrm{kgm^{-3}}$, $C_V = 100\ \mathrm{Jkg^{-1}\,K^{-1}}$, $\lambda = 100\ \mathrm{Wm^{-1}\,K^{-1}}$ and $\alpha = 1 \times 10^3\ \mathrm{Wm^{-2}\,K^{-1}}$). They are real physical properties. An initial temperature of $\tau = 373.15$ K was predefined to all nodes of the two-dimensional square with side lengths $l = 0.01$ m. An ambient temperature of $\tau = 273.15$ K is defined and the cooling due to free convection is calculated. In Abaqus a film coefficient of $\alpha = 1 \times 10^3\ \mathrm{Wm^{-2}\,K^{-1}}$ was applied to all outer edges as boundary condition. For the PD model the boundary was not pre-defined but identified using Equation (26) with $f_{limit} = 0.75$. The result of the specific volumes are shown in Figure 6b. The blue colored points are used as edge nodes. This method is not perfectly exact, because at the edges also one node is used as edge node which lies in the second row or column.

**Figure 7** shows the temperature distribution of Abaqus and PeriLab. The color was chosen in an equal distribution. The temperature distribution is similar. For better comparability, **Figure 8** shows the temperature distribution after $t = 100$ s at

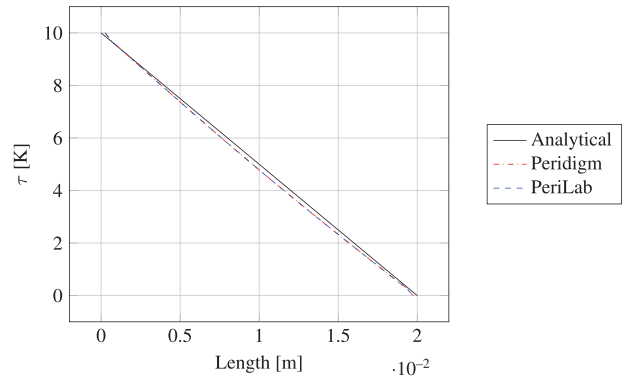



**Figure 4.** Temperature distribution for the steady state case. The width of the boundary condition on the left ($\tau = 10$ K) and right hand ($\tau = 0$ K) side is $l_{bc} = 1 \times 10^{-3}$ m. The geometry and discretization are given as $l = 2 \times 10^{-2}$ m, $l = 2 \times 10^{-2}$ m, $h = 1 \times 10^{-3}$ m, $\delta = 3.005 \times 10^{-3}$ m, cf. Figure 2. The model needs $1 \times 10^{-4}$ s to reach the nearly stable solution.

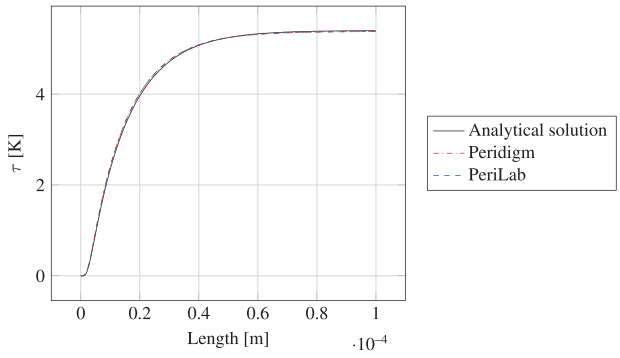
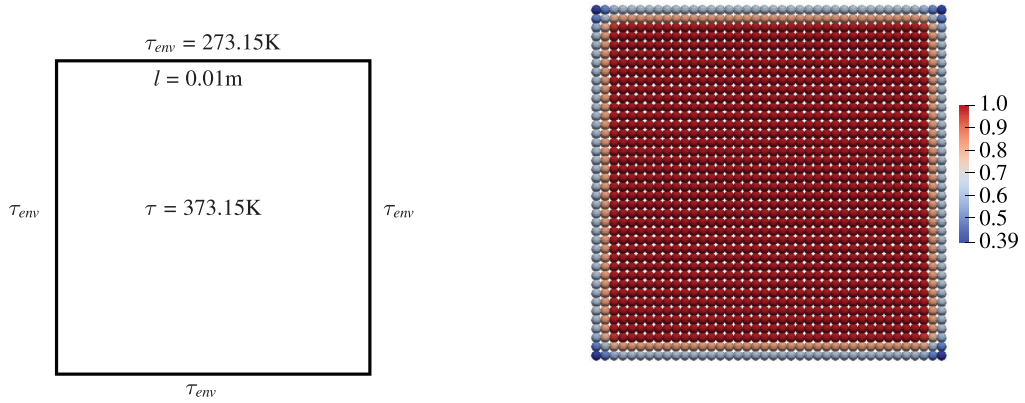



**Figure 5.** Temperature distribution over length at position $x = 9.2 \times 10^{-3}$ m. The width of the boundary condition on the left ($\tau = 10$ K) and right hand ($\tau = 0$ K) side is $l_{bc} = 1 \times 10^{-3}$ m. The geometry and discretization are given as $l = 2 \times 10^{-2}$ m, $h = 1 \times 10^{-3}$ m, $\delta = 3.005 \times 10^{-3}$ m, cf. Figure 2.




(a) Cooling model with parameters $\rho = 1 \times 10^3$ kg m$^{-3}$, $C_V = 1.8 \times 10^3$ J kg$^{-1}$ K$^{-1}$, $\lambda = 0.12$ W m$^{-1}$ K$^{-1}$ and $\alpha = 1.5 \times 10^3$ W m$^{-2}$ K$^{-1}$

(b) Specific volume distribution using Equation 26 with $f_{limit} = 0.75$ to identify the boundary points.

**Figure 6.** Geometry and discrete model for the verification of the cooling process.

$y = 5 \times 10^{-3}$ m) in $x_1$ direction. It can be seen that both models are in very good agreement. Comparing the PD models with the Abaqus solution, it can be seen that the temperatures of the outer edges are slightly higher. This is due to boundary effects and the lower convergence of the material point based PD compared to FEM. With a finer discretization this effect will be reduced, which can be seen in the Figure as well.

### 3.4. Conclusion

The verification shows that all thermal and thermo-mechanical models were implemented in a correct way. The errors compared to the reference solutions are low and can be reduced by using a finer discretization, if the spatial distribution is not sufficient.

Additionally, it is worth mentioning that the value of the horizon $\delta$ is quite crucial for the correct temperature distribution. However, the effects are out of the scope for this paper.

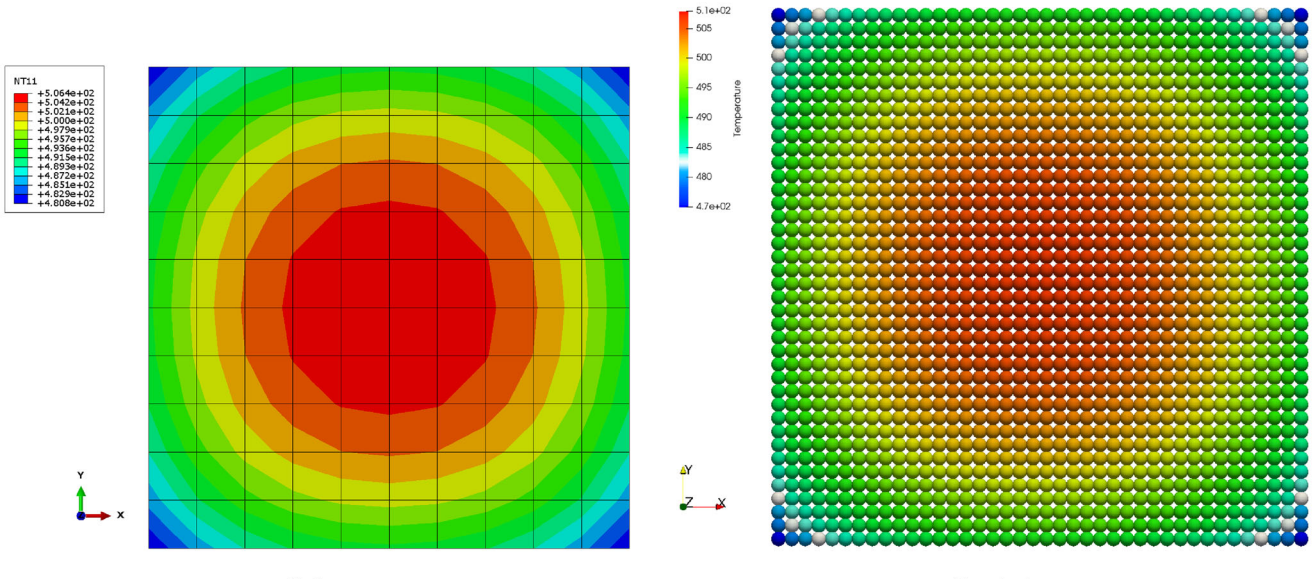
In the next section the printing process is verified.

## 4. Printing Process

This section describes how the PD mesh is generated based on the G-code and the corresponding CAD model. Furthermore, a printed L Profile is shown and analyzed as an example.

### 4.1. Printing Process

In order to simulate an additive material and its thermo-mechanical behavior with `PeriLab` a discretized geometrical




(a) Abaqus

(b) `PeriLab`

**Figure 7.** Geometry and discrete model for the verification of the cooling process.

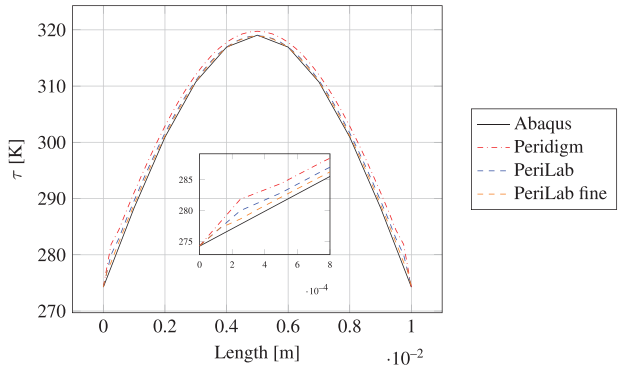



**Figure 8.** Comparison of the temperature distribution at final time $t = 100\,\mathrm{s}$ at position $y = 5 \times 10^{-3}$ m between Abaqus, `Peridigm` `PeriLab` and `PeriLab` with a finer mesh. The structured discretization is given as $dx = 2.5 \times 10^{-4}$ m and $dx = 1.25 \times 10^{-4}$ m for the fine mesh, as well as $\delta = 6.35 \times 10^{-4}$ m, cf. Figure 6a.

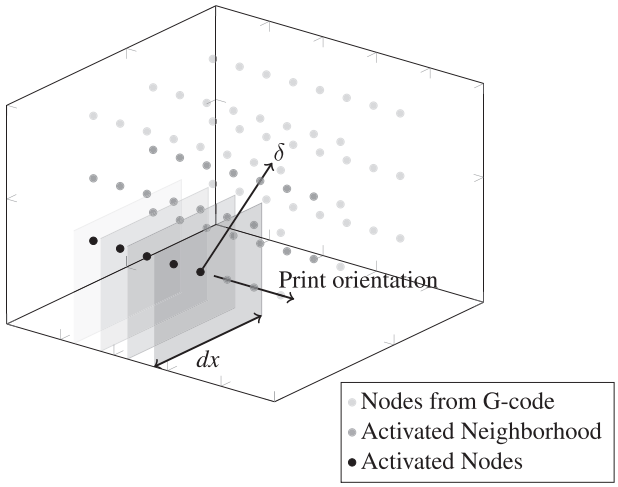



**Figure 9.** Conversion schema.

model is needed. As pointed out in ref. [12] a list of points, their corresponded volumes and material for the mesh free approximation is needed. To obtain them, it starts with the CAD model of the structure to be printed. With a commercial slicer software, the path to print this CAD model is derived. Typically, the machines for the printing process are controlled by the most widely used computer numerical control programming language G-code.[28] It describes the path and process speed of the printing process. This numerical process information has to be interpreted and translated into the `PeriLab` mesh representation. Each point gets extra information about its status, namely an activation time. If the simulation time exceeds this activation time the point is activated. Following steps are necessary to write a discretized mesh.

- Read parameters from G-code
- Read printing path from G-code
- Extracting EventSeries
- Create a PD reference mesh
- Find PD nodes within a bounding box around the printing path
- Calculating activation time and orientation of nodes
- Assign activation time and orientation to the mesh points
- Write converted mesh

The first step is to create a series of events from the tool-path planning. The EventSeries is an `Abaqus` specific data format describing the position of the tool-center point at all time points. Next the information of the EventSeries will be used to create a discretized model. The `PeriLab` mesh free material point based approach utilizes points with corresponding volumes representing the center of gravity and volume of a geometrical element (sphere, block or finite element). With this mesh an analysis of the whole structure can be applied, by calculating the interaction between these material points. As stated earlier, for additive manufacturing additionally the activation information needs to be provided.

The printing path and depositing rate is then used to obtain the activation time for each material point and the orientation of these points based on the path direction. Within the G-code the movement of the nozzle is mainly described by linear interpolations using a fixed start- and endpoint and a path velocity based on the feed-rate. In order to obtain the `PeriLab` mesh the G-code path is followed. The maximum step width is defined by the minimum distance between two material points of the `PeriLab` mesh. Because the path is only a line a bounding box with the above-mentioned parameters is placed around the print path to find the material points which have to be activated. They get the specific activation time when it lies the first time inside the virtual nozzle bounding box. As shown in **Figure 9** every `PeriLab` node which is crossed by the bounding box will receive the corresponding activation time and orientation. This strategy allows mesh refinements, because multiple points can be activated simultaneously.

During a simulation, if the assigned time-value of nodes is reached by the simulation time, these nodes and their neighbors will be activated. Additional boundary conditions, e.g., printing temperatures can be applied.

The user has the option to choose if every time step for each node a surface-check will take place. Especially for process with heated materials, the identification of a surface or non-surface node is important, in order to calculate the correct heat flow as shown in Equation (2.4). This leads to a cooling of the structure. During the printing process this parameter might change.

### 4.2. L-Angle

In the following a 3D printed L-angle structure is virtually printed. The L-angle is a simple test geometry suitable for the analysis of part distortions by measuring the angle. Moreover, the large ratio of contact area to total height allows to differ the effects of heat conduction and convective heat transfer. In the vicinity of the printing bead, heat conduction dominates the thermal transport mechanism, whereas as the height increases, convective heat transfer assumes greater importance. Thus, the L-angle geometry provides a trade-off between complexity and representativeness. For verification purposes of `PeriLab` the model is calculated with FEM in `Abaqus` and PD in `PeriLab`. In order to get reasonable calculation times only the thermal model is solved. This allows larger stable time steps provided by Equation (28).

**Table 3.** Material properties of the models.[30–33]

| Material property | Value | Unit |
|---|---|---|
| Density | 1240 | $kgm^{-3}$ |
| Specific heat | 1800 | $Jkg^{-1}\,K^{-1}$ |
| Young's modulus | 3450 | MPa |
| Poisson's ratio | 0.34 | — |
| Expansion coefficient | $7 \times 10^{-5}$ | $K^{-1}$ |
| Heat convection coefficient | 15 | $Wm^{-2}\,K^{-1}$ |
| Thermal conductance material | 0.12 | $Wm^{-1}\,K^{-1}$ |
| Thermal conductance print bed | 1.2 | $Wm^{-1}\,K^{-1}$ |

**Table 4.** Printing parameters of the L-Angle.

| Parameter | Value | Unit |
|---|---|---|
| Layer height | $0.2 \times 10^{-3}$ | m |
| Layer width | $0.45 \times 10^{-3}$ | m |
| Print bed temperature | 333 | K |
| Extrusion temperature | 463 | K |
| Print time | 1532 | s |

### 4.3. Geometry, Material and Process Data

The L-shaped angle geometry has legs of 50 mm length, 20 mm height, and a uniform thickness of 2 mm. The analysis is performed in 3D. The applied material parameters are taken from the data sheet of the Ingeo Biopolymer Polylactic Acid (PLA) from NatureWorks,[29] completed by literature values for typical 3D printing PLA materials. The properties and their sources are listed in **Table 3**.

The geometry is sliced with the PrusaSlicer[34] software. The L-angle is created with four contour strands (perimeter) and no infill is used. The angle is printed as shown in **Figure 10**.

The printing parameters are taken from the PrusaSclicer Software for an example PLA material and are summarized in **Table 4**.

Analogous to the `PeriLab` process, the generated G-code is transformed into an EventSeries for the FEM simulation. In `Abaqus` the geometry is discretized and material properties are assigned according to Table 3. Subsequently, the EventSeries is imported via the Simulia `Abaqus` AM Modeler plugin. Within this plugin, process parameters such as layer height and width are defined. In the simulation, the elements are then activated along the G-code path according to the defined layer thickness and width. Therefore, the G-code and the FEM mesh must be in alignment. Due to different reference coordinate systems during the slicing process, this is not often the case and it is necessary to realign the mesh and the part. In this case the mesh was aligned to the printing path. The reason for the differently oriented mesh and printing path is that the part is reoriented on the printing bed for the slicing process. The resulting model simulates the printing process through activating the elements when material is deposit. The simulation is carried out by a sequentially coupled temperature-displacement analyses. First, a thermal analysis is conducted. In the thermal analysis, the print bed is also modelled in order to consider the heat conduction between the print bed and the component which is a crucial boundary condition. In future the result of the thermal simulation can then be transferred as pre-defined field to a structural simulation to calculate the corresponding residual stresses and deformation. This was right now not possib le (**Figure 11**).

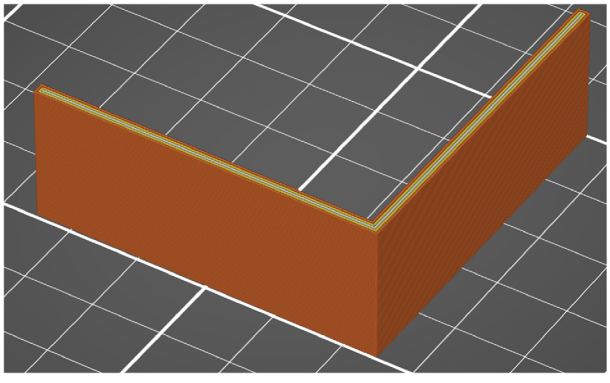


**Figure 10.** Sliced geometry in PrusaSlicer Software of the L-angle 50 × 20 × 2 mm.

### 4.4. Analysis

In **Figure 12** below the final temperature profile of the L-angle simulation along the z-axis for the `PeriLab` as well as for the `Abaqus` simulation for two different sections at the end of the printing process can be seen. The first section is positioned in the middle of the component, while the second is at the furthest end of the component.

Due to the heating of the lower part of the printed component, the temperature is equal to the 333 K print bed temperature for both sections. With an increasing distance the next layers are going to show a lower temperature due to the heat flow to the environment. The minimal temperature of approximately 294 K is then reached at 7.6 mm height. As the first section is encapsuled by a few layers of printed material, the core temperature is higher at those points, as we would expect.

Above 10 mm the influence of the of the most recently deposited layers is increasing and the temperature rises to 335 K. It can also be seen that the cooling process of the second section, which is located at the end of the components leg, is quicker, due to the higher heat convection. In summary, the results are highlighting the expected process related thermal behavior of an additive manufactured part.

Additionally, the direct comparison with `Abaqus` shows that there is a slight temperature difference of approximately 2.4 K. The main reason for this offset between both simulation frameworks is the different time step size. The `Abaqus` simulation is using a five-time greater time increment compared to the `PeriLab` model. As a result, large portions of the layers are printed at ones. That decreases the time for cooling for certain parts of the problems (worst case is $5dt$). On the other side underlying layers are allowed to cool longer. They exist longer as outer layer and are allowed to transmit heat over the surface. This is supported by Figure 12. The inner temperature is much lower for the `Abaqus` solution, whereas the outer temperature is nearly equal between

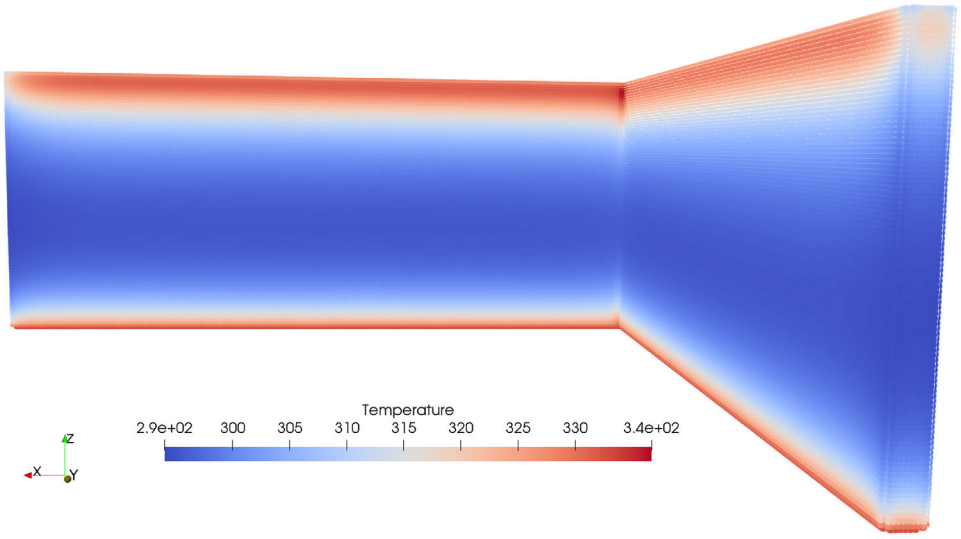



**Figure 11.** Temperature distribution of the PD L-angle simulation results.

Abaqus and PeriLab. Due to the numerical effort it was not feasible to run the analysis in smaller time increments in Abaqus.

The second difference is the surface approximation. PD has no surfaces. In the numerical approximation each point is a volume. Therefore, edges and corners are handled as surfaces as well, leading to errors.

To conclude, besides the differences of the models both lead to a comparable result. The included implementations are capable of encapsuling the additive manufacturing process of the L-Angle and for more complex geometrical models as well.

## 5. Conclusion

The paper shown here, presented the PD framework to analyze additive manufacturing processes using PeriLab. The process excludes right now the complex material behavior like solidification, which will be included in a next step based on the work of Hein et al.[35]. The included thermal and thermo-elastic models were verified against analytical, FEM and Peridigm models and are in good agreement. A novel boundary detection algorithm was introduced to enable the use of a PD mesh-free implementation. It is needed, because within a manufacturing process the outer surface area might change. This identification works well for the provided use cases.

For the more complex analysis of an L-shaped profile, the Abaqus and PeriLab solution differs only slightly from each other and further work needs to be done to clarify those differentiation. Further research could focus on clarifying the mentioned behavior while allowing a more sophisticated code basis.

Finally, it can be summarized that a PD framework such as PeriLab is capable of a highly detailed dynamic thermo-mechanical analysis, which is needed for additive manufacturing processes. In further research the methods presented here will have to be refined, especially the influence of the PD mesh and the correct calculation of the specific volume needs to be looked into. The refined methods can then be validated with printed parts which can be printed and measured with the infrastructure at the DLR Brunswick. For example a L-angle in an industrial MiniFactory printer shown in **Figure 13**.

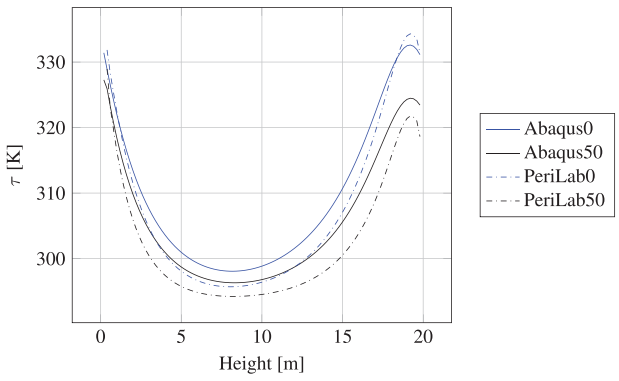



**Figure 12.** Comparison of the temperature distribution at final simulation time $t = 1550\,s$ at position $y = 25\,mm$ and $50\,mm$ between Abaqus and PeriLab.

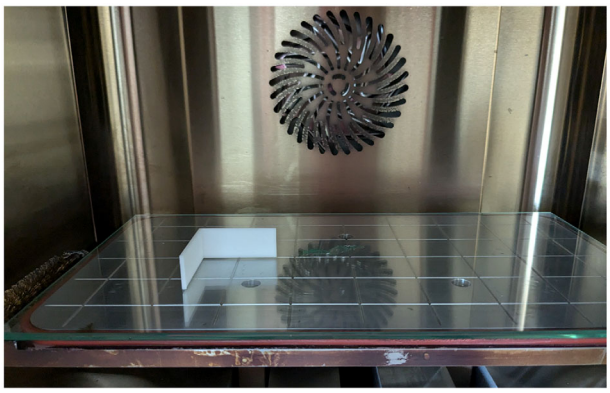


**Figure 13.** Printed L-angle in Minifactory printer.

Furthermore, the performance of a fully-coupled thermo-mechanical simulation for large components is still a huge obstacle for PD as well as existing FEM methods.

## Appendix A: Program Codes

**Listing 1.** Python code to solve 1D heat equation.

```
import numpy as np
from scipy.integrate import odeint
########################
N = 201 # number of points to discretize
# PD discretization
dx = 0.00005
# free length minus width of PD boundary
L = 0.02-4*dx condution
h = L / (N - 1)
X = np.arange(0, L+h, h)
k = 3
########################
def odefunc(u, t):
 dudt = np.zeros(X.shape)
 dudt[0] = 0 # constant at boundary condition
 dudt[-1] = 0
 # spatial second order differentiation
 # utilizing second order finite difference
 for i in range(1, N-1):
  dudt[i] = k * (u[i - 1] - 2*u[i] + u[i + 1]) / h**2
 return dudt
init = 0.0 * np.ones(X.shape) # initial temperature
init[0] = 10.0 # one boundary condition
init[-1] = 0 # the other boundary condition
tspan = np.arange(0,2e-5,3e-7)
sol = odeint(odefunc, init, tspan)
```

## Acknowledgements

The work was funded by the German Research Foundation funded project: "Gekoppelte Peridynamik-Finite-Elemente-Simulationen zur Schädigungsanalyse von Faserverbundstrukturen" Grant number: WI 4835/5-1 and the M-ERA.NET funded project Exploring Multi-Method Analysis of composite structures and joints under consideration of uncertainties engineering and processing (EMMA).

This measure was co-financed with tax funds on the basis of the budget passed by the Saxon state parliament. Grant number: 3028223. Federal Ministry for Economic Affairs and Climate Action funded project "Virtuelle Kennwertermittlung, Schadensprädiktion und Simulationsmethoden für geklebte Fügestellen eines LH2-Tanks in Faserverbundbauweise für die kommerzielle Luftfahrt". Grant number: 20W2214G. The authors like to thank for the funding.

Open access funding enabled and organized by Projekt DEAL.

## Conflict of Interest

The authors declare no conflict of interest.

## Author Contributions

C.W. wrote the paper, programed PD code and reviewed the paper, J.H. wrote the paper, programed PD code and reviewed the paper, F.W. provided the reference solutions and reviewed the paper, R.H. wrote the paper and reviewed it.

## Data Availability Statement

The data that support the findings of this study are openly available in https://github.com/PeriHub/PeriLab.jl at https://doi.org/10.5281/zenodo.10229386, reference number 1.

## Keywords

additive manufacturing, FEM, peridynamics, process simulation, thermal analysis

Received: August 7, 2024
Revised: October 16, 2024
Published online:

[1] P. Hartmann, C. Weißenfels, P. Wriggers, *Comput. Part. Mech.* **2020**.
[2] P. Penumakala, J. Santo, A. Thomas, *Composites, Part B* **2020**, *201*, 108336.
[3] M. Liebisch, R. Hein, T. Wille, *CEAS Aeronautical J.* **2018**, *9*, 545.
[4] S. A. Silling, M. Epton, O. Weckner, J. Xu, E. Askari, *J. Elastic.* **2007**, *88*, 151.
[5] F. Bobaru, M. Duangpanya, *Int. J. Heat Mass Transfer* **2010**, *53*, 4047.
[6] D. J. Littlewood, M. L. Parks, J. A. Mitchell, S. A. Silling, in *12th U.S. National Congress on Computational Mechanics, 22-25 July 2013, Raleigh, North Carolina, USA*. Albuquerque, New Mexico 87185 and Livermore, California 94550, USA, **2013**.
[7] J. A. Mitchell, S. A. Silling, D. J. Littlewood, *Mech. Mater. Struct.* **2015**, *10*, 539.
[8] X. Li, H. Ye, J. Zhang, in *2021 IEEE Int. Parallel Distr. Proc. Symp. (IPDPS)*, **2021**, pp. 433–443.
[9] C. Willberg, J.-T. Hesse, M. Garbade, M. Rädel, F. Heinecke, A. Schuster, A. Pernatii, *SoftwareX* **2023**.
[10] J.-T. Hesse, C. Willberg, R. Hein, F. Winkelmann, *PAMM* **2023**, *23*, e202300033.
[11] C. Willberg, J.-T. Hesse, Perilab - peridynamic laboratory v1.0, GitLab repository, **2023**, https://gitlab.com/dlr-perihub/perilab (accessed: December 2023).
[12] C. Willberg, J.-T. Hesse, A. Pernatii, *SoftwareX* **2024**, *26*, 101700.
[13] M. R. Tupek, Ph.D. Thesis, Massachusetts Intitute of Technology, **2014**.
[14] M. Breitenfeld, P. Geubelle, O. Weckner, S. A. Silling, *Comput. Methods Appl. Mech. Eng.* **2014**, *272*, 233.
[15] H. Chen, *Mech. Res. Commun.* **2018**, *90*, 34.
[16] P. Li, Z. Hao, W. Zhen, *Comput. Methods Appl. Mech. Eng.* **2018**, *339*, 262.
[17] M. R. Tupek, R. Radovitzky, *J. Mech. Phys. Solids* **2014**, *65*, Supplement C 82.
[18] C. Wu, *Comput. Mech.* **2014**, *54*, 1255.
[19] C. Wu, B. Ren, *Comput. Methods Appl. Mech. Eng.* **2015**, *291*, 197.
[20] J. Wan, Z. Chen, X. Chu, H. Liu, *Acta Mech. Sin.* **2019**, *35*, 1021.
[21] S. A. Silling, *Comput. Methods Appl. Mech. Eng.* **2017**, *332*, 42.
[22] T. Xue, X. Zhang, K. K. Tamma, *J. Comput. Phys.* **2018**, *374*, 1180.
[23] R. Brighenti, M. A. Zeleke, M. B. Ageze, *J. Eng.* **2021**, 20.
[24] X. Gu, Q. Zhang, E. Madenci, *Comput. Methods Appl. Mech. Eng.* **2019**, *36*, 2557.
[25] S. Oterkus, E. Madenci, A. G. Agwai, *J. Comput. Phys.* **2014**, *265*, 71.
[26] S. Oterkus, E. Madenci, A. G. Agwai, *J. Mech. Phys. Solids* **2014**, *64*, 1.
[27] D. Hillel, in *Encyclopedia of Physical Science and Technology (Third Edition)*, (Eds.: R. A. Meyers), Academic Press, New York, third edition edition, **2003**, pp. 77–97.
[28] https://reprap.org/wiki/G-code (accessed: December 2023).

[29] NatureWorks®, Ingeo 2003d, online, **2024**, https://catalog.ides.com/Datasheet.aspx?I=26793&E=150054s (accessed: January 2024).
[30] DEDietrich, Borosilicate glass properties, online, **2024**, https://za.dedietrich.com/products-solutions/borosilicate-glass-properties (accessed: January 2024).
[31] O. Zmeskal, L. Marackova, T. Lapcikova, P. Mencik, R. Prikryl, *AIP Conf. Proc.* **2020**, *2305*, 020022.
[32] R. Ferreira, I. Amatte, T. Assis Dutra, D. Bürger, *Composites, Part B* **2017**, *124*.
[33] Simplify3D, Filament properties table, online, **2024**.
[34] https://www.prusa3d.com/de/page/prusaslicer_424/ (accessed: January 2024).
[35] R. Hein, T. Wille, K. Gabtni, J. P. Dias, *Defect Diffus. Forum* **2015**, *362*, 224.