Noname manuscript No.
(will be inserted by the editor)

hp-finite element simulation of thermoelastic deformation in NC-milling

Removal-dependent mesh refinements and a fictitious domain approach

Andreas Schröder · Andreas Byfut · Raffael Joliet · Tobias Surmann

Received: date / Accepted: date

Abstract This paper presents an *hp*-finite element approach for the simulation of heat diffusion and thermoelastic deformation in the NC milling process. Varying complex shapes resulting from the material removal are taken into account via a removal-dependent mesh refinement that relies on the isotropic bisection of hexahedrons intersecting the geometry of the milling tool in contact. To represent the cutting surface more accurately, the hexahedrons at the cutting surface are further subdivided into tetrahedrons, pyramids and prisms. To avoid the involved application of higher-order shape funtions on these shapes, and to enable the exclusive use of tensor product shape functions on hexahedrons, a fictitious domain approach is used. Numerical experiments show the applicability of the proposed method.

Keywords NC milling · mesh refinement · hp-finite elements

PACS 65.40.De

Mathematics Subject Classification (2000) 74F05 · 74S05 · 65M50 · 65M60 · 65M85

support by the DFG priority programme SPP 1480, SU 59911-1, CA 151/20-1

A. Schröder
Humboldt-Universität zu Berlin, Unter den Linden 6, 10099 Berlin
Tel.: +49-30-2093-2630
Fax: +49-30-2093-5859
E-mail: schroder@math.hu-berlin.de

A. Byfut
Humboldt-Universität zu Berlin, Unter den Linden 6, 10099 Berlin
Tel.: +49-30-2093-5436
Fax: +49-30-2093-5859
E-mail: byfut@math.hu-berlin.de

R. Joliet
Technische Universität Dortmund, Baroper Straße 301, 44227 Dortmund
Tel.: +49-231-755-2364
Fax: +49-231-755-5141
E-mail: joliet@isf.de

T. Surmann
Technische Universität Dortmund, Baroper Straße 301, 44227 Dortmund
Tel.: +49-231-755-4390
Fax: +49-231-755-5141
E-mail: surmann@isf.de

1 Introduction

In the roughing process of the NC milling (see Figure 1), there is a significant amount of heat induced into the workpiece due to the conversion of energy during the chip formation process, which results in a global thermoelastic deformation. These distortions remain present in the subsequent finishing process. As the NC paths are usually generated purely geometrically, the tool position is independent from the expansion state of the workpiece which may consequently cause a strong deviation of the workpiece surface from its designed shape after cooling down. Consequently, critical fabrication tolerances may be exceeded. In order to take countermeasures, it is necessary to know the thermomechanical deformation. Purely geometric simulation systems that predict collision situations and general geometric errors are already commercially available, but do not solve this kind of problem.

The aim of this paper is to present a finite element approach for the simulation of heat diffusion and thermoelastic deformation during the NC milling process for varying geometrically complex shapes resulting from the material removal during the NC milling. This allows for predictions on possible deviations between the

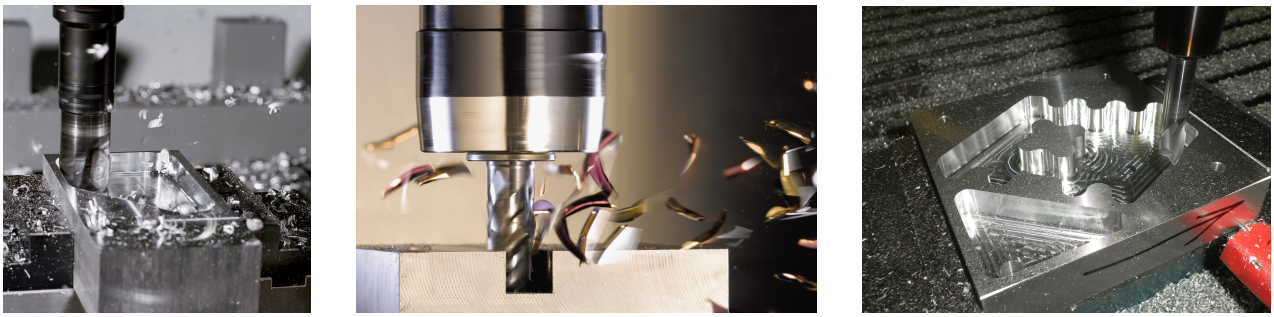


Fig. 1 Workpiece and milling cutter during the roughing process.

designed and the produced shape. The finite element approach is based on a removal-dependent mesh refinement for the workpiece. The mesh refinement relies on the isotropic bisection of hexahedrons intersecting the geometry of the milling tool in contact. The hexahedrons at the cutting surface are further subdivided to represent this surface more accurately. This is done by the removal of vertices from hexahedrons that are captured by the milling tool. The resulting polyhedrons are then further subdivided into tetrahedrons, pyramids and prisms. Hexahedrons completely captured by the milling tool are removed from the workpiece mesh.

Heat diffusion and thermoelastic deformation are often modeled by linear thermoelasticity [1,10,19] which is also used in this paper. The varying geometry of the workpiece is taken into account in presented model of thermoelasticity, more precisely, in its variational formulation. Furthermore, it is assumed that heat is only induced by the milling tool. It diffuses into the interior and eventually balances with the ambient temperature at the boundary. Well-established Newmark and Crank-Nicholson finite difference schemes are applied to discretize the hyperbolic and parabolic contributions of the variational equations in temporal direction, cf. [13]. A finite element approach is used to discretize the resulting semi-discrete system in spatial direction. This is directly done on the removal-dependent mesh of the workpiece. To increase the quality of the approximation, *hp*-finite elements are applied [2,5,11,17]. To avoid the involved application of higher-order shape funtions on tetrahedrons, prisms and in particular pyramids, and to enable the exclusive use of tensor product shape functions on hexahedrons, a fictitious domain approach is used which is similar to that proposed in [7,12] for the finite cell method. In the approach presented here, the computational domain is covered by a hexahedron mesh and is directly used as the finite element mesh. Similar to the XFEM, the cutting surface is directly incorporated into the finite element discretization space. This is done by the multiplication of the tensor product shape functions with a characteristic function which is defined elementwise employing the subtriangulation of hexahedrons into tetrahedrons, prisms and pyramids. This method is equivalent to the use of a weighted L^2-scalar product with the characteristic function as a weight function.

The covering mesh is locally refined and contains hanging nodes resulting from the refinement of hexahedrons without the refinement of neighboring hexahedrons. Using a conforming discretization approach, we have to ensure the finite element functions of higher-order to be continuous across edges and faces. In the presence of hanging nodes, this is done through the contraint of the local basis functions via the concept of constrained approximation and the use of connectivity matrices in the assembly process. For tensor product shape functions on hexahedrons, this can be done very efficiently, cf. [15,16]. Apart from the efficient incorporation of hanging nodes, tensor product shape functions allow for an efficient evaluation of the functions themselves as well as their derivatives. Consequently, the element matrices can be computed very efficiently, which is particularly true for paraxial hexahedral meshes. This further motivates the use of the fictitious domain approach for the proposed modeling of the NC milling process.

The approximation quality of the finite element discretization is increased by a uniform degree distribution with degree greater one (p-method) or by using an appropriate *hp*-adaptive degree distribution (*hp*-method). A canonical *hp*-distribution on the mesh refinements is to use polynomial degree one on smallest hexahedrons and, then, to increase the polynomial degree layerwise so that the highest polynomial degree is used on the largest hexahedrons. This leads to a significant increase of the approximation quality on large hexahedrons since it is expected that the solution's regularity is high away from the contact area of the milling tool.

In each time step of the time stepping schemes, the workpiece mesh is adjusted to the position of the milling

tool and, in the case of *hp*-finite elements, the polynomial degree distribution is adapted as well. The data transfer from the previous to the current finite element mesh is done via an L^2-projection. Again, a weighted L^2-scalar product based on the characteristic function of the workpiece mesh is used. This approach allows for the removal of heat due to the removal of material from the workpiece in a natural way.

The remainder of this paper is organized as follows: In Section 2, the removal-dependent mesh refinement for the workpiece is presented. The modeling of heat diffusion and thermoelastic deformations of the workpiece with linear thermoelasticity is described in Section 3. The modeling of the heat input at cutting surface is introduced in Section 4. The spatial discretization with *hp*-finite element methods and a fictitious domain approach as well as the L^2-projection with weighted L^2-scalar product are described in Section 5. Numerical results show the applicability of the approach in Section 6.

2 Removal-dependent mesh refinement of the workpiece

In this section, we consider the mesh refinement of the NC-milled workpiece with respect to the material removal performed by the milling cutter. The mesh refinement algorithm introduced in this section computes a mesh $\mathcal{H}$ which approximately represents the difference $\Omega\backslash\Theta$ of two sets $\Omega, \Theta \subset \mathbb{R}^3$ with Lipschitz boundary. The set Ω is assumed to be polygonal in such a way that

$$\Omega = \bigcup_{H\in\tilde{\mathcal{H}}} \bigcup_{T\in\tilde{S}(H)} T. \tag{1}$$

where $\tilde{\mathcal{H}}$ is a mesh of hexahedrons and $\tilde{S}$ is a mapping which maps each $H \in \tilde{\mathcal{H}}$ onto a subdivision of a polyhedron resulting from the removal of one or more vertices or onto H itself. The refinement algorithm generates a new hexahedron mesh $\mathcal{H}$ and a new subdivision mapping S so that

$$\Omega\backslash\Theta \approx \bigcup_{H\in\mathcal{H}} \bigcup_{T\in S(H)} T.$$

Furthermore, it generates meshes $\mathcal{C}$ and $\mathcal{D}$ of faces (triangles or rectangles) which represent the cutting surface and that area of the cutting surface where the material removal takes place.

To formally describe the refinement algorithm, we need some preparations. To this end, let $\mathcal{V}(T)$ be the set of all vertices of a hexahedron, tetrahedron, prism or pyramid $T \subset \mathbb{R}^3$ and $\mathcal{U}(H)$ be the set of the 8 hexahedrons which are obtained from the symmetric bisection of the hexahedron $H \subset \mathbb{R}^3$. Let $\mathcal{V} := \mathcal{V}(H) \cap \Theta$ be the set of vertices of H which are cut. In Figure 4, all possible distributions of those vertices are depicted where symmetric vertex distributions are omitted. The removal of the vertices given by $\mathcal{V}$ from H is done in following way: A vertex $V \in \mathcal{V}$ is cut by the removal of the convex hull of this vertex and those vertices which are adjacent via an edge, see Figure 4a. We continue this procedure for the remaining polyhedron. This is done until a polyhedron is obtained which does not contain vertices indicated to be cut. The removal process is given by the Algorithm 1 where $\mathrm{adj}(V, P)$ denotes the adjacent vertices to $V \in \mathcal{V}$ in the polyhedron P and $\mathrm{conv}(M)$ denotes the convex hull of the set $M \subset \mathbb{R}^3$. The input data is a hexahedron H and the set $\mathcal{V}$. The output is a convex polyhedron $P = P(H, \mathcal{V})$.

Algorithm 1 Vertex removal.

```
P := H
R := V
A := ∅
while R ≠ ∅ do
    if R ∩ A ≠ ∅ then
        V ∈ R ∩ A
    else
        V ∈ R
    end if
    R := R\{V}
    P := (P\ conv({V} ∪ adj(V, P))) ∪ conv(adj(V, P))
    A := adj(V, P)
end while
```

Remark 1 Adding the convex hull of the adjacent vertices in Algorithm 1, we ensure that the face spanned by these vertices remains in the resulting polyhedron. Furthermore, the use of the set $\mathcal{A}$ of the adjacent vertices of V guarantees that $\mathrm{conv}(\mathrm{adj}(V, P))$ is always contained in a plane. An example can be seen in Figure 2(e) where Algorithm 1 would fail if this precaution is ignored.

We subdivide the resulting polyhedron $P(H, \mathcal{V})$ into tetrahedrons, pyramids and prisms as examplarily depicted in Figure 3 and set $\mathcal{S}(H, \mathcal{V})$ to this subdivision. Note that the subdivision is, of course, not unique. The particular cases $\mathcal{V} = \emptyset$ and $\mathcal{V} = \mathcal{V}(H)$ are specified by $\mathcal{S}(H, \emptyset) := \{H\}$ and $\mathcal{S}(H, \mathcal{V}(H)) := \emptyset$. Moreover, we define the set $\mathcal{C}(H, \mathcal{V})$ as the set of faces of the subdivision $\mathcal{S}(H, \mathcal{V})$ which are uncovered by the removal, cf. Figure 4. This set depends on the dimension of the

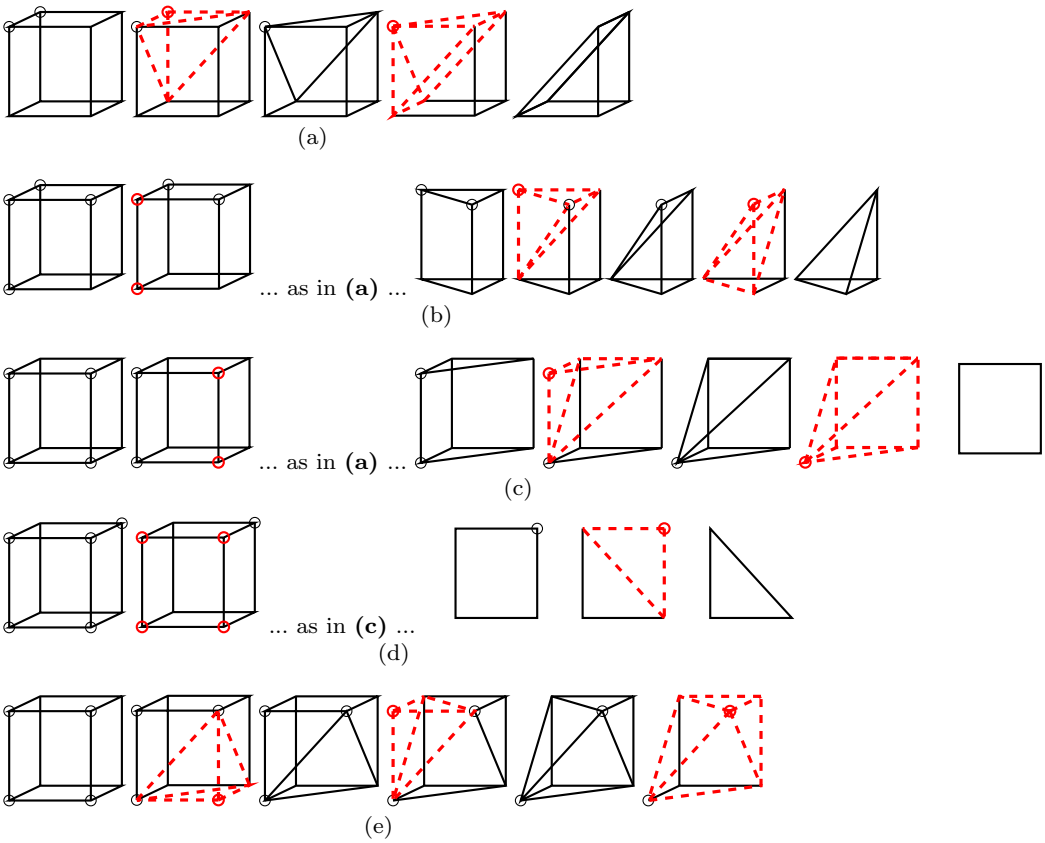


Fig. 2 Removal of vertices by Algorithm 1. Vertices to be cut are marked by a circle.

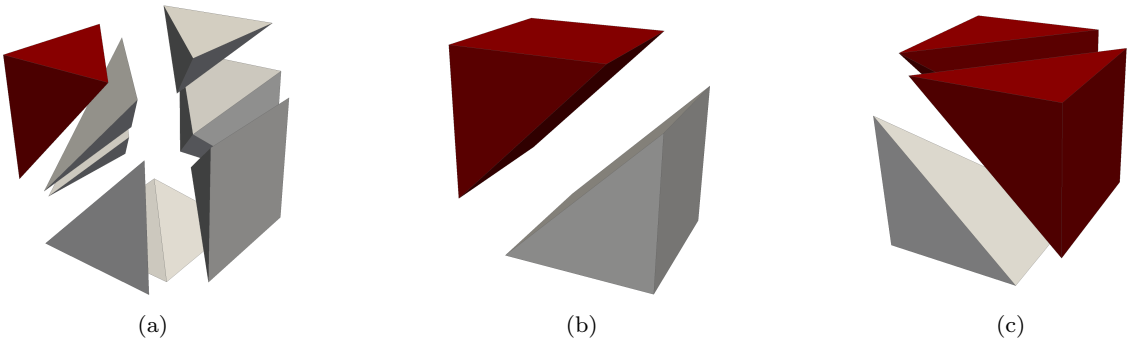


Fig. 3 Subdivisions $\mathcal{S}(H,\mathcal{V})$ of the hexahedron H into tetrahedrons, prisms and pyramids. The polyhedron marked by red color is removed. This corresponds to the removal of the following vertices given by $\mathcal{V}$: **(a)** the top, front, left vertex is removed, **(b)** both top, left vertices are removed, **(c)** all top vertices but the top, front, left vertex are removed.

polyhedron $P(H,\mathcal{V})$ and is given by

$$\mathcal{C}(H,\mathcal{V}) := \begin{cases} \mathcal{F}(H^\circ \cap \partial P(H,\mathcal{V})), & \dim P(H,\mathcal{V}) = 3 \\ \mathcal{F}(\partial H \cap P(H,\mathcal{V})), & \dim P(H,\mathcal{V}) = 2 \\ \emptyset, & \text{otherwise} \end{cases}$$

where $\dim(M)$ denotes the dimension of the linear hull of the set $M \subset \mathbb{R}^3$ and $\mathcal{F}(M)$ is the set of faces of M, if M is a subset of the boundary of a polyhedron. Note that $\mathcal{C}(H,\emptyset) = \mathcal{C}(H,\mathcal{V}(H)) = \emptyset$. We define the set of faces $\mathcal{F}'(H)$ of the subdivision $S(H)$ of the hexahedron $H \in \tilde{\mathcal{H}}$ by

$$\mathcal{F}'(H) := \bigcup_{T \in S(H)} \mathcal{F}(\partial T). \tag{2}$$

The input data of the refinement algorithm 2 are the hexahedron mesh $\tilde{\mathcal{H}}$, the mapping $\tilde{S}$ and the set $\tilde{\mathcal{C}}$ containing the faces at the cutting surface as well as the

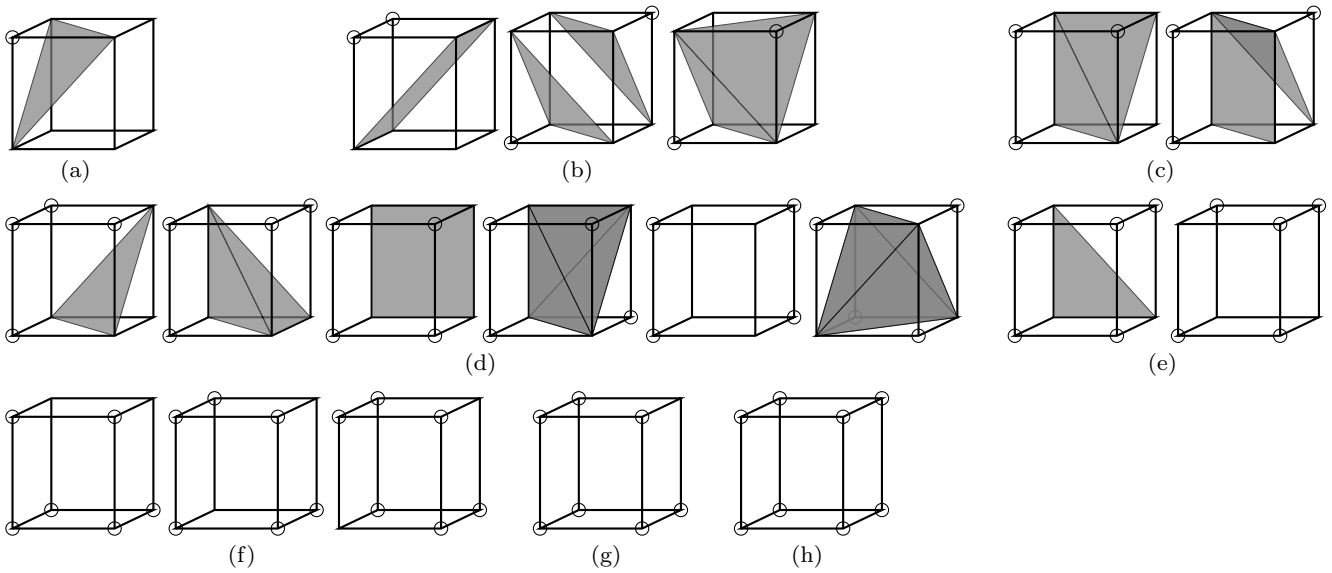


Fig. 4 Grey faces indicates the faces in $\mathcal{C}(H,\mathcal{V})$ of the subdivision $\mathcal{S}(H,\mathcal{V})$. The removed vertices are marked by a circle: **(a)** one vertex, **(b)** two vertices, ..., **(h)** eight vertices

set $\Theta \subset \mathbb{R}^3$ and a minimal hexahedron length $d > 0$. The maximum length of the hexahedron H is denoted by $\ell(H)$. The output is the new hexahedron mesh $\mathcal{H}$ and the set $\mathcal{D}$ containing the faces in the contact area. Moreover, the mapping $\tilde{S}$ and the set $\tilde{\mathcal{C}}$ are updated to S and $\mathcal{C}$, respectively.

Remark 2 The initial determination of the set $\mathcal{R}$ in Algorithm 2 can be done very efficiently by collision detection algorithms based on octree data structures. The introduction of the vertices set $\mathcal{N}$ prevents the use of vertices which are already cut.

Remark 3 Algorithm 1 is executed in a preprocessing step to determine all possible, but finite many subdivisions which are tabulated in advance so that the evaluation of $\mathcal{S}(H,\mathcal{V})$ and $\mathcal{C}(H,\mathcal{V})$ can be done efficiently.

In the subsequent sections, we only need to determine that an element is contained in the sets $\mathcal{C}$ or $\mathcal{D}$. For instance, we do not need an iteration through these sets. Hence, in the implementation of Algorithm 2, we only have to assign an attribute to a face $F \in \mathcal{C}$ or $F \in \mathcal{D}$ that indicates its containedness.

Algorithm 2 Mesh refinement.

$\mathcal{R} := \{H \in \tilde{\mathcal{H}} \mid \exists T \in \tilde{S}(H) :\ T \cap \Theta \neq \emptyset\}$
$\mathcal{N} := \{\mathcal{V}(H)\backslash\mathcal{V}(T) \mid H \in \tilde{\mathcal{H}},\ T \in S(H)\}$
$\mathcal{H} := \tilde{\mathcal{H}}\backslash\mathcal{R}$
$\mathcal{C} := \tilde{\mathcal{C}}$
$\mathcal{D} := \emptyset$
for all $H \in \mathcal{H}$ **do**
  $S(H) := \{H\}$
end for
for all $H \in \mathcal{R}$ **do**
  $\mathcal{C} := \mathcal{C}\backslash\mathcal{F}'(H)$
end for
while $\mathcal{R} \neq \emptyset$ **do**
  $H \in \mathcal{R}$
  $\mathcal{R} := \mathcal{R}\backslash\{H\}$
  $\mathcal{V} := \mathcal{V}(H) \cap \Theta \cap \mathcal{N}$
  if $\ell(H) < d$ **then**
    if $\mathcal{S}(H,\mathcal{V}) \neq \emptyset$ **then**
      $\mathcal{H} := \mathcal{H} \cup \{H\}$
      $S(H) := \mathcal{S}(H,\mathcal{V})$
    end if
    if $\mathcal{C}(H,\mathcal{V}) \neq \emptyset$ **then**
      $\mathcal{C} := \mathcal{C} \cup \mathcal{C}(H,\mathcal{V})$
      $\mathcal{D} := \mathcal{D} \cup \mathcal{C}(H,\mathcal{V})$
    end if
  else
    if $H \cap \Theta \neq \emptyset$ **then**
      $\mathcal{R} = \mathcal{R} \cup \mathcal{U}(H)$
    else
      $\mathcal{H} := \mathcal{H} \cup \{H\}$,
      $S(H) := \{H\}$
    end if
  end if
end while

3 The coupled system of thermoelasticity

We apply linear thermoelasticity to model the coupling of the deformation of the workpiece and the heat diffusion and refer to the monographs [9,10,14,19] for detailed overviews. We assume that the raw workpiece is given by a paraxial cuboid $\Omega^0 := [a_1, b_1] \times [a_2, b_2] \times [a_3, b_3]$ which is, of course, often the case in milling practice. The workpiece is clamped at the closed boundary

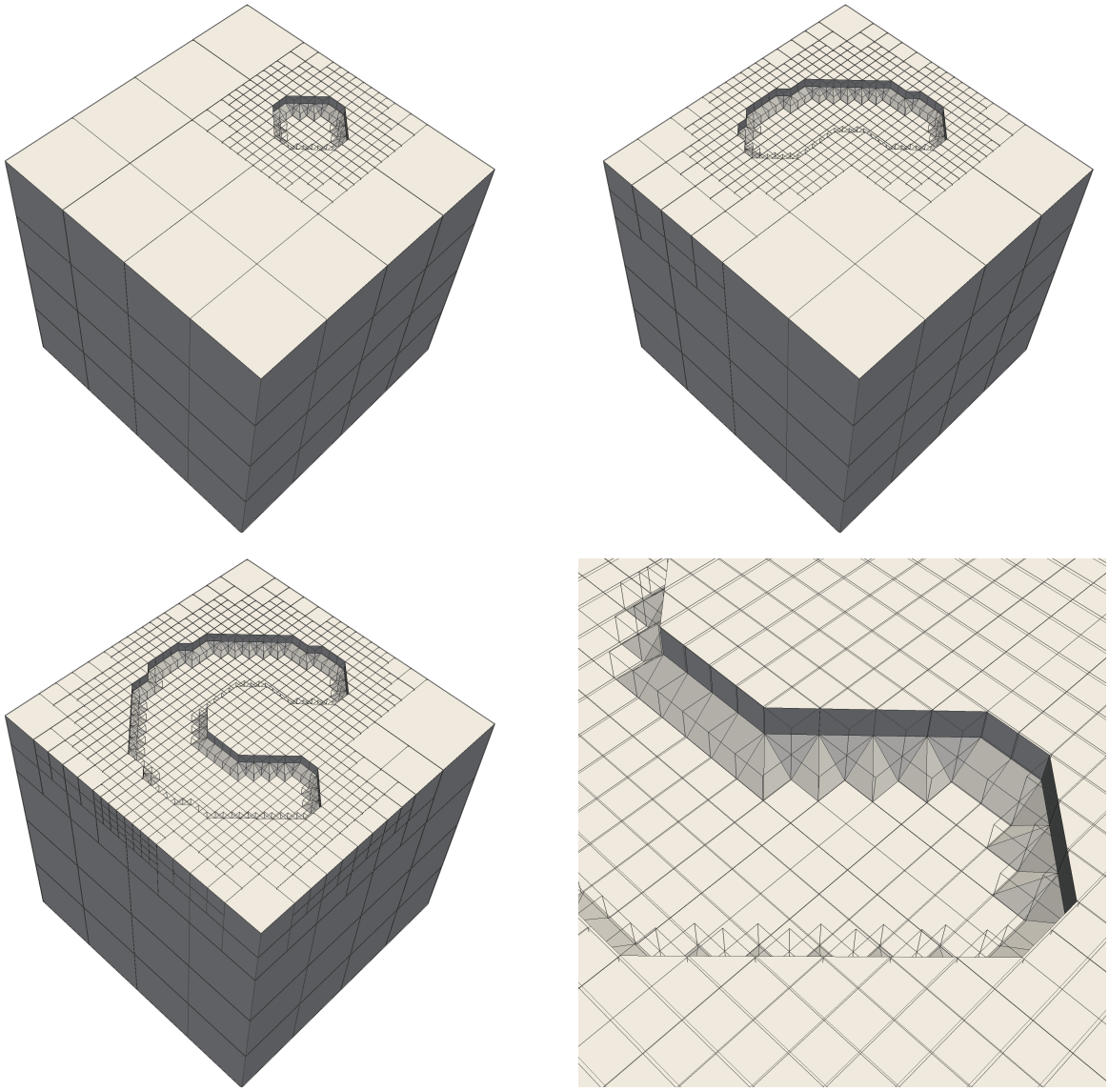


Fig. 5 Adaptive refinement for workpiece with isotropic hexahedron bisections for the finite element mesh and the subdivision into tetrahedrons, prism and pyramids for the subtriangulation of hexahedrons.

part $\Gamma_D \subset \partial\Omega^0$ assumed to have a positive measure. Furthermore, let $\Lambda^t \subset \mathbb{R}^3$ describe the milling cutter at time $t \in I := [0,T]$, $T > 0$, where $\Lambda^0 \cap \Omega^0 = \emptyset$. We assume $\Gamma_D \cap \Lambda^t = \emptyset$ for all $t \in I$, i.e. there is no removal at that boundary part where the workpiece is clamped. The hull of all milling cutter positions is given by $\Xi^t := \bigcup_{s=0}^t \Lambda^s$. Hence, the milled workpiece at $t \in I$ is $\Omega^t := \Omega^0 \backslash \Xi^t$.

Let $L^2(\Omega^0)$ and $H^1(\Omega^0)$ denote the usual Sobolev spaces and $H^1_D(\Omega^0) := \{v \in H^1(\Omega^0) \mid v = 0 \text{ on } \Gamma_D\}$. For a displacement field $v \in H^1(\Omega^0;\mathbb{R}^3)$ we specify the linearized strain tensor as $\varepsilon(v) := \frac{1}{2}(\nabla v + (\nabla v)^\top)$ and the stress tensor of Hooke's material law as $\sigma(v) := 2\mu\varepsilon(v) + 2\,\mathrm{tr}\,\varepsilon(v)\mathbb{I}$ with the identity tensor $\mathbb{I}$ and Lame's contants λ and μ. The heat flow is given by $-k\nabla\theta$ for a heat distribution $\theta \in H^1(\Omega^0)$ with heat conductivity $k > 0$.

We consider the following problem of thermoelasticity: Find a deformation field u and a heat distribution θ with $u(0) = \dot{u}(0) = 0$ and $\theta(0) = \theta_0$ in Ω^0 such that

$$\rho\ddot{u}(t) = \mathrm{div}(\sigma(u(t)) + m\theta(t)\mathbb{I}) \text{ on } \Omega^t, \tag{3}$$

$$c\dot{\theta}(t) = -k\,\mathrm{div}\,\nabla(\theta(t)) + m\theta_0\,\mathrm{tr}\,\varepsilon(\dot{u}(t)) \text{ on } \Omega^t, \tag{4}$$

$$\sigma_n(u(t)) = 0 \text{ on } \Gamma_N^t, \tag{5}$$

$$\partial_n(\theta(t)) = g(t) \text{ on } \Gamma_N^t \tag{6}$$

for almost all $t \in I$. Here, $\theta_0 \in \mathbb{R}$ is an initial constant heat distribution, m is the stress-temperature modulus, c is the specific heat and ρ is the density of the workpiece. Furthermore, n denotes the vector-valued function describing the outer unit normal vector with respect to the Neumann boundary part $\Gamma_N^t := \partial\Omega^t \backslash \Gamma_D$. We set $\sigma_{n,i} := \sigma_{ij} n_j$ and $\partial_n := (\partial/\partial x_i) n_i$.

The function $g(t) \in L^2(\Gamma_N^t; \mathbb{R}^3)$ model the heat input on Γ_N^t at time $t \in I$, cf. Section 4. Note that no volume or surface forces are included in (3)-(6) so that deformations only result from thermal stresses. Furthermore, no interior heat sources are assumed so that heat is only induced by $g(t)$ on the boundary Γ_N^t. For simplicity, we assume that m, c and ρ are positive constants.

In the case that the coupling of the heat distribution to the strains is neglectable, the coupling term $\theta_0 m \operatorname{tr} \varepsilon(\dot{u}(t))$ in (4) can be omitted. Hence, we obtain the parabolic heat equation

$$c\dot{\theta}(t) = -k \operatorname{div} \nabla(\theta(t)) \text{ on } \Omega^t \tag{7}$$

instead of (4). If acceleration effects are irrelevant, we obtain the quasi-static equilibrium equation

$$0 = \operatorname{div}(\sigma(u(t)) + m\theta(t)\mathbb{I}) \text{ on } \Omega^t \tag{8}$$

which substitutes the hyperbolic equation (3). These model simplifications lead to a decoupling of the heat distribution from the deformation field and facilitate the discretization of the problem since only the parabolic heat equation has to be discretized in time. We refer to the monograph [1] for more detailed explanations, in particular, with respect to further justifications of model reductions.

To apply discretization schemes based on finite elements, it is necessary to reformulate (3)-(6) in terms of variational equations. For this purpose, let $(\cdot,\cdot)_{0,\omega}$ denote the usual L^2-scalar product on $L^2(\omega)$, $L^2(\omega;\mathbb{R}^3)$ and $L^2(\omega;\mathbb{R}^{3\times 3})$, respectively, and define

$$V := W^{2,\infty}(I; L^2(\Omega^0;\mathbb{R}^3)) \cap L^\infty(I; H_D^1(\Omega^0;\mathbb{R}^3)),$$
$$W := W^{1,\infty}(I; L^2(\Omega^0)) \cap L^\infty(I; H_D^1(\Omega^0)).$$

Using integration by parts, we obtain the weak formulation: Find $(u,\theta) \in V \times W$ with $u(0) = \dot{u}(0) = 0$ and $\theta(0) = \theta_0$ such that

$$\begin{aligned}\rho(\ddot{u}(t), v)_{0,\Omega^t} &+ (\sigma(u(t)), \varepsilon(v))_{0,\Omega^t} \\ &= m(\operatorname{div}(\theta(t)\mathbb{I}), v)_{0,\Omega^t}\end{aligned} \tag{9}$$

$$\begin{aligned}c(\dot{\theta}(t), \varphi)_{0,\Omega^t} &+ k(\nabla(\theta(t)), \nabla\varphi)_{0,\Omega^t} \\ &= m\theta_0(\operatorname{tr}\varepsilon(\dot{u}(t)), \varphi)_{0,\Omega^t} + (g(t), \varphi)_{0,\Gamma_N^t}\end{aligned} \tag{10}$$

for all $(v,\varphi) \in H_D^1(\Omega^0;\mathbb{R}^3) \times H_D^1(\Omega^0)$ and almost all $t \in I$. We choose the trial spaces V and W for notational convenience. The existence of a solution is not clear, even in the case of an unvarying geometry, i.e. $\Omega^0 = \Omega^t$ for all $t \in I$, cf. [8].

The model reductions given by (7) and (8) lead to the variational formulation

$$\begin{aligned}&(\sigma(u(t)), \varepsilon(v))_{0,\Omega^t} = m(\operatorname{div}(\theta(t)\mathbb{I}), v)_{0,\Omega^t} \\ &c(\dot{\theta}(t), \varphi)_{0,\Omega^t} + k(\nabla(\theta(t)), \nabla\varphi)_{0,\Omega^t} = (g(t), \varphi)_{0,\Gamma_N^t}.\end{aligned} \tag{11}$$

instead of (9)-(10).

To discretize the variational systems (9)-(10) and (11) in time, we split the time interval I into N subintervals $I_n := (t_{n-1}, t_n]$ of length $k_n := t_n - t_{n-1}$ with $0 =: t_0 < t_1 < \ldots < t_{N-1} < t_N := T$.

We also discretize the workpiece domain Ω^{t_n}. This is done by Algorithm 2 where a mesh $\mathcal{H}^n$ is generated in each time step t_n, $1 \le n \le N$. Since Ω^0 is assumed to be a paraxial cuboid, we are able to use a mesh $\mathcal{H}^0$ of paraxial hexahedrons with $\Omega^0 := \bigcup_{H\in\mathcal{H}^0} H$ where the simplest choice for $\mathcal{H}^0$ is given by $\mathcal{H}^0 := \{\Omega^0\}$. Obviously, it would be very involved to create a completely new mesh at time step t_n by Algorithm 2 by setting Θ to the hull of all milling cutter positions Ξ^{t_n}. In order to reuse the refinements done in previous steps, Algorithm 2 can be applied in the following way: Compute the mesh $\mathcal{H}^n := \mathcal{H}$ as well as the mapping $S^n := S$ and the sets $\mathcal{C}^n := \mathcal{C}$ and $\mathcal{D}^n := \mathcal{D}$ by setting $\tilde{\mathcal{H}} := \mathcal{H}^{n-1}$, $\tilde{S} := S^{n-1}$, $\tilde{\mathcal{C}} := \mathcal{C}^{n-1}$ and $\Theta := \Lambda^{t_n}$ where $\mathcal{H}^{n-1}$, S^{n-1} and $\mathcal{C}^{n-1}$ are created at the time step t_{n-1}. For $t_0 = 0$, we take $\mathcal{H}^0$ as introduced above as well as $S^0(H) := H$ for $H \in \mathcal{H}^0$ and $\mathcal{C}^0 = \emptyset$. The milled workpiece at time step t_n is then approximately given by

$$\tilde{\Omega}^n := \bigcup_{H\in\mathcal{H}^n} \bigcup_{T\in S^n(H)} T. \tag{12}$$

Note that $\tilde{\Omega}^n \subset \tilde{\Omega}^{n-1}$ and $\tilde{\Omega}^n \subset \hat{\Omega}^n$ where $\hat{\Omega}^n$ is created by Algorithm 2 with $\tilde{\mathcal{H}} := \mathcal{H}^0$ and $\Theta := \Xi^{t_n}$. However, $\tilde{\Omega}^n = \hat{\Omega}^n$ does not hold in general.

Remark 4 To represent $\tilde{\Omega}^n$ within an appropriate data structure, it is sufficient to store $\mathcal{H}^n$. The access to the subdivisions at the cutting surface can easily be handled by the mapping S^n. The sets $\mathcal{C}^n$ and $\mathcal{D}^n$ are not used so far. In Section 4 we need these sets to define Neumann data on the cutting surface.

The temporal discretization of the hyperbolic equation (9) may be done by the Newmark scheme: Denoting the approximation of the function $w(t_n)$ by w^n, we approximate $v := \dot{u}$ and $a := \ddot{u}$ by

$$a^n := \frac{1}{\beta k_n^2}(u^n - u^{n-1}) - \frac{1}{\beta k_n} v^{n-1} - (\frac{1}{\beta} - 1)a^{n-1},$$
$$v^n := v^{n-1} + k_n((1-\alpha)a^{n-1} + \alpha a^n).$$

Here, α and β are parameters in the interval $[0,2]$. For unvarying geometries, the choice $\alpha = 1/2$ ensures second order convergence, whereas $2\beta \geq \alpha \geq 1$ implies unconditional stability, cf. [13].

The parabolic heat equation (10) can be discretized in temporal direction with the Crank-Nicholson scheme,

$$c(\theta^n,\varphi)_{0,\tilde{\Omega}^n} + \frac{k_n}{2}k(\nabla\theta^n,\nabla\varphi)_{0,\tilde{\Omega}^n} = \langle d^n,\varphi\rangle \tag{13}$$

with

$$\begin{aligned}\langle d^n,\varphi\rangle &:= \langle f^n,\varphi\rangle + \frac{k_n}{2}m\theta_0(\operatorname{tr}\varepsilon(v^n+v^{n-1}),\varphi)_{0,\tilde{\Omega}^n},\\ \langle f^n,\varphi\rangle &:= c(\theta^{n-1},\varphi)_{0,\tilde{\Omega}^n} - \frac{k_n}{2}(k(\nabla\theta^{n-1},\nabla\varphi)_{0,\tilde{\Omega}^n}\\ &\quad - (\tilde{g}^n + \partial_n\theta^{n-1},\varphi)_{0,\tilde{\Gamma}_N^n}).\end{aligned} \tag{14}$$

where the Neumann boundary condition is given by $\tilde{g}^n \in L^2(\tilde{\Gamma}_N^n)$ on $\tilde{\Gamma}_N^n := \partial\tilde{\Omega}^n\backslash\Gamma_D$. Applying this temporal discretization to the variational formulation (11) resulting from model reduction, we obtain

$$(\sigma(u^n),\varepsilon(v))_{0,\tilde{\Omega}^n} = m(\operatorname{div}(\theta^n\mathbb{I}),v)_{0,\tilde{\Omega}^n} \tag{15}$$

$$c(\theta^n,\varphi)_{0,\tilde{\Omega}^n} + \frac{k_n}{2}k(\nabla\theta^n,\nabla\varphi)_{0,\tilde{\Omega}^n} = \langle f^n,\varphi\rangle. \tag{16}$$

Again, the temporal discretization (15)-(16) of the reduced model is decoupled in the sense that u^n depends on θ^n, but θ^n does not depend on u^n. Thus, θ^n is computed via (15) first, then it is inserted in (16) and u^n is computed. The temporal discretization of the non-reduced model given by the combination of Newmark and Crank-Nicholson scheme is fully coupled. In each time step, a further iteration (for instance a fix point iteration) has to be executed to compute u^n and θ^n.

4 Modeling of heat input

The only source of heat is given by the material removal at the cutting surface. It is modeled by the Neumann data which also models the balancing of the workpiece's heat with the ambient temperature. The Neumann data are given by the function $g(t) \in L^2(\Gamma_N^t)$ for $t \in I$ or $\tilde{g}^n$, respectively. We assume that a certain fraction $\mu > 0$ of the mechanical cutting work W is induced into the milled workpiece at the contact area of the milling cutter. On the remaining part of the workpiece surface, the heat is balanced with the ambient temperature. This heat input is modeled in the non-discretized systems of Section 3 by the Neumann data

$$g(t) := \begin{cases}\mu(t)W(t), & \text{on } \partial\Omega^t \cap \partial\Lambda^t,\\ a(\theta(t)-\theta_a), & \text{on } \Gamma_N^t\backslash(\partial\Omega^t\cap\partial\Lambda^t)\end{cases}$$

where $\theta_a \in \mathbb{R}$ is the constant ambient temperature and a is a further constant modeling the heat transfer to air, cf. [18].

The Neumann data $\tilde{g}^n$ of the discretized systems of Section 3 are specified via

$$\tilde{g}^n := \begin{cases}\mu^n W^n, & \text{on } \tilde{\Gamma}_{\mathcal{D}}^n\\ a(\theta^n-\theta_a), & \text{on } \tilde{\Gamma}_N^n\backslash\tilde{\Gamma}_{\mathcal{D}}^n\end{cases}$$

where $\tilde{\Gamma}_{\mathcal{D}}^n := \bigcup_{F\in\mathcal{D}^n} F$.

Note that the boundary conditions modeling the balance with the ambient temperature is a Robin boundary condition. This has to be taken into account in the Crank-Nicolson scheme (13), which is modified to

$$\begin{aligned}c(\theta^n,\varphi)_{0,\tilde{\Omega}^n} + \frac{k_n}{2}k(\nabla\theta^n,\nabla\varphi)_{0,\tilde{\Omega}^n} - \frac{k_n}{2}a(\theta^n,\varphi)_{0,\tilde{\Gamma}_N^n\backslash\tilde{\Gamma}_{\mathcal{D}}^n}\\ = \langle d^n,\varphi\rangle\end{aligned} \tag{17}$$

where d^n is defined as in (14) and f^n is redefined as

$$\begin{aligned}\langle f^n,\varphi\rangle &:= c(\theta^{n-1},\varphi)_{0,\tilde{\Omega}^n} - \frac{k_n}{2}(k(\nabla\theta^{n-1},\nabla\varphi)_{0,\tilde{\Omega}^n}\\ &-(\partial_n\theta^{n-1},\varphi)_{0,\tilde{\Gamma}_N^n}+a(\theta_a,\varphi)_{0,\tilde{\Gamma}_N^n\backslash\tilde{\Gamma}_{\mathcal{D}}^n}-(\mu^nW^n,\varphi)_{0,\tilde{\Gamma}_{\mathcal{D}}^n}).\end{aligned}$$

These modifications are also applied to the reduced variational formulation (15)-(16).

5 Spatial discretization with *hp*-finite elements and a fictitious domain approach

We apply *hp*-finite elements to discretize the deformation u^n and the temperature θ^n in space. For this purpose, let $p^n : \mathcal{H}^n \to \mathbb{N}$ be a polynomial degree distribution, $\mathbb{P}_q$ be the space of polynomials with degree q on $[-1,1]^3$ and, finally, $\Psi_H : [-1,1]^3 \to H \in \mathcal{H}^n$ be a bijective affine transformation. The *hp*-finite element space $\mathbb{S}_{hp}^n$ is defined as

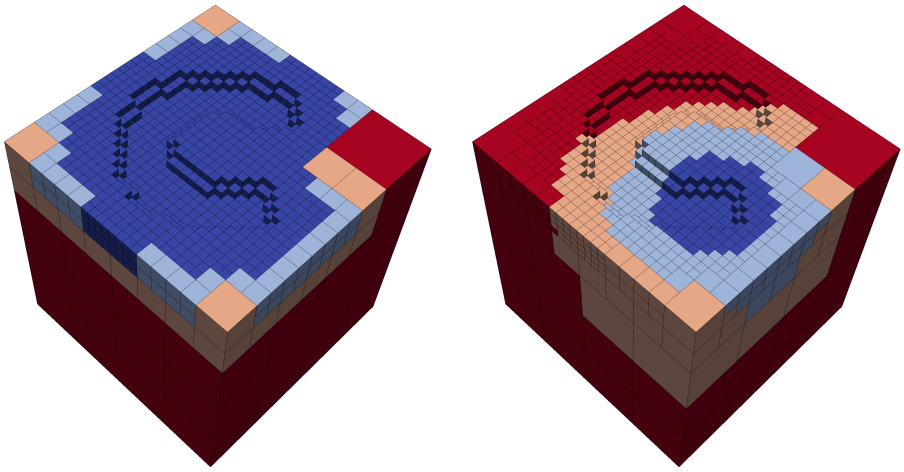
$$\mathbb{S}_{hp}^n := \{v \in H_D^1(\bar{\Omega}^n) \mid \forall H \in \mathcal{H} :\ v\circ\Psi_H \in \mathbb{P}_{p^n}\}$$

where $\bar{\Omega}^n := \bigcup_{H\in\mathcal{H}} H$.

Let $(\zeta_i)_{i=1}^s$ be a basis of $\mathbb{S}_{hp}^n$ with $s := \dim\mathbb{S}_{hp}^n$. For each $H \in \mathcal{H}^n$, there exists a matrix $\pi_H \in \mathbb{R}^{s\times r}$ so that

$$\zeta_{|H} = \pi_H\xi\circ\Psi_H^{-1}$$

with the shape functions $\xi := (\xi_i)_{i=1}^r$ as a basis of $\mathbb{P}_q$, $r := \dim\mathbb{P}_q = (q+1)^3$ and $q := \max_{H\in\mathcal{H}^n} p^n(H)$. The matrices π_H are called connectivity matrices; their construction for several basis functions with varying polynomial degree distribution is described in [15,16] where, in particular, nodal basis function with Lagrange polynomials as well as hierarchical basis functions with integrated Legendre polynomials are discussed. Note that



(a) (b)

Fig. 6 Polynomial degree distributions for a given time step, where **(a)** the polynomial degrees are chosen depending on the size of the individual hexahedrons and **(b)** on the distance of the individual hexahedrons from the milling cutter.

(multilevel) hanging nodes can not be avoided in the mesh refinement proposed in Section 2. They can also efficiently be handled with connectivity matrices. We refer to the cited references where symmetric and, additionally, unsymmetric subdivisions of the faces are concerned.

The deformation u^n is approximated by $u^n_{hp} \in (\mathbb{S}^n_{hp})^3$ with

$$u^n_{hp,i|H} := x_i^\top(\pi_H \xi \circ \Psi_H^{-1}),\ i = 1,2,3$$

and the temperature θ^n is approximated by $\theta^n_{hp} \in \mathbb{S}^n_{hp}$ with

$$\theta^n_{hp} := y^\top(\pi_H \xi \circ \Psi_H^{-1})$$

and coefficient vectors $x = (x_1, x_2, x_3) \in \mathbb{R}^{3s}$ and $y \in \mathbb{R}^s$.

In the following, we focus on the discretization of the reduced model as given in (15)-(16). But, all considerations can be carried over to the fully coupled system (9)-(10) temporally discretized by the Newmark and Crank-Nicholson scheme.

Using the connectivity matrices π_H, we obtain from the modifications of (15)-(16) as introduced in (17) the following algebraic system:

$$\begin{aligned} Kx &= F, \\ Ay &= B. \end{aligned} \tag{18}$$

The stiffness matrix $K \in \mathbb{R}^{3s\times 3s}$ and load vector $L \in \mathbb{R}^{3s}$ are defined as

$$K := \sum_{H\in\mathcal{H}^n} \hat{\pi}_H K_H \hat{\pi}_H^\top, \quad L := \sum_{H\in\mathcal{H}^n} \hat{\pi}_H L_H \tag{19}$$

with $\hat{\pi}_H := \mathrm{diag}(\pi_H, \pi_H, \pi_H) \in \mathbb{R}^{3s\times 3r}$ and the element matrices $K_H \in \mathbb{R}^{3r\times 3r}$ as well as the element vectors $L_H \in \mathbb{R}^{3r}$,

$$\begin{aligned} K_{H,kl} &:= (\sigma(\xi_{i'} \circ \Psi_H^{-1} e_{j'}), \varepsilon(\xi_i \circ \Psi_H^{-1} e_j))_{0,\Delta_H}, \\ L_{H,k} &:= m(\mathrm{div}(\theta^n_{hp}\mathbb{I}), \xi_i \circ \Psi_H^{-1} e_j)_{0,\Delta_H}. \end{aligned}$$

Here, $k = 3(j-1)+i$, $l = 3(j'-1)+i'$, $i,j = 1,\dots,r$, $i',j' = 1,2,3$. The unit cartesian vectors is denoted by $e_j \in \mathbb{R}^3$, and Δ_H is a subset of H which is specified below. The matrix $A \in \mathbb{R}^{s\times s}$ and the vector $B \in \mathbb{R}^s$ are given by

$$A := \sum_{H\in\mathcal{H}^n} \pi_H A_H \pi_H^\top, \quad B := \sum_{H\in\mathcal{H}^n} \pi_H B_H \tag{20}$$

with the element matrices $A_H \in \mathbb{R}^{r\times r}$ and the element vectors $B_H \in \mathbb{R}^r$ defined as

$$\begin{aligned} A_{H,ij} &:= c(\xi_j \circ \Psi_H^{-1}, \xi_i \circ \Psi_H^{-1})_{0,\Delta_H} \\ &\quad + \frac{k_n}{2} k(\nabla(\xi_j \circ \Psi_H^{-1}), \nabla(\xi_i \circ \Psi_H^{-1}))_{0,\Delta_H} \\ &\quad - \frac{k_n}{2} a(\xi_j \circ \Psi_H^{-1}, \xi_i \circ \Psi_H^{-1})_{0,\Upsilon_{R,H}} \\ B_{H,i} &:= c(\theta^{n-1}_{hp}, \xi_i \circ \Psi_H^{-1})_{0,\Delta_H} \\ &\quad - \frac{k_n}{2}(k(\nabla\theta^{n-1}_{hp}, \nabla(\xi_i \circ \Psi_H^{-1}))_{0,\Delta_H} \\ &\quad - (\partial_n \theta^{n-1}_{hp}, \xi_i \circ \Psi_H^{-1})_{0,\Upsilon_H} \\ &\quad + a(\theta_a, \xi_i \circ \Psi_H^{-1})_{0,\Upsilon_{R,H}} \\ &\quad - (\mu W^n, \xi_i \circ \Psi_H^{-1})_{0,\Upsilon_{\mathcal{D},H}}). \end{aligned}$$

and the subsets $\Upsilon_H, \Upsilon_{\mathcal{D},H}, \Upsilon_{R,H} \subset H$.

The fictitious domain approach is given via the definition of the sets $\Delta_H, \Upsilon_H, \Upsilon_{\mathcal{D},H}, \Upsilon_{R,H} \subset H$. Using the

mappings S^n, we set

$$\Delta_H := \bigcup_{T \in S^n(H)} T.$$

The sets $\Upsilon_H, \Upsilon_{\mathcal{D},H}, \Upsilon_{R,H}$ are subsets of the boundary $\tilde{\Gamma}_N^n$ and are defined by using the sets $\mathcal{C}^n$ and $\mathcal{D}^n$ which are generated by Algorithm 2. Again, $\mathcal{F}'(H)$ contains all faces of the subdivision of H given by the mapping S^n, see (2). With

$$\mathcal{F}^0(H) := \{F \in \mathcal{F}'(H) \mid F \subset \Gamma_N^0\}$$

we define

$$\Upsilon_{\mathcal{D},H} := \bigcup_{F \in \mathcal{F}'(H) \cap \mathcal{D}^n} F,$$

$$\Upsilon_{R,H} := \bigcup_{F \in (\mathcal{F}'(H) \cap (\mathcal{C}^n \setminus \mathcal{D}^n)) \cup \mathcal{F}^0(H)} F,$$

$$\Upsilon_H = \Upsilon_{\mathcal{D},H} \cup \Upsilon_{R,H}.$$

If one of the sets Δ_H or $\Upsilon_H, \Upsilon_{\mathcal{D},H}, \Upsilon_{R,H}$ are empty, the corresponding L^2-scalar products in the definition of the element matrices and vectors K_H, L_H, A_H and B_H are omitted.

Since $\tilde{\Omega}^n \subset \tilde{\Omega}^{n-1}$, it is possible to evaluate θ_{hp}^{n-1} on $\tilde{\Omega}^n$. But this necessitates the use of the different meshes $\mathcal{H}^n$ and $\mathcal{H}^{n-1}$. It is essentially more efficient to use the mesh $\mathcal{H}^n$ only. For this purpose, we transfer θ_{hp}^{n-1} to the mesh $\mathcal{H}^n$ via an L^2-projection yielding $\tilde{\theta}_{hp}^{n-1} \in \mathbb{S}_{hp}^n$ which is given by

$$(\tilde{\theta}_{hp}^{n-1} - \theta_{hp}^{n-1}, \varphi_{hp})_{0,\tilde{\Omega}^n} \tag{21}$$

for all $\varphi_{hp} \in \mathbb{S}_{hp}^n$. The L^2-projection (21) leads to the algebraic system

$$Mz = D \tag{22}$$

with the mass matrix $M \in \mathbb{R}^{s \times s}$ and the vector $D \in \mathbb{R}^s$ defined as

$$M := \sum_{H \in \mathcal{H}^n} \pi_H M_H \pi_H^T, \quad D := \sum_{H \in \mathcal{H}^n} \pi_H D_H. \tag{23}$$

The element matrices $M_H \in \mathbb{R}^{r \times r}$ and the element vectors $D_H \in \mathbb{R}^r$ are given by

$$M_{H,ij} := (\xi_j \circ \Psi_H^{-1}, \xi_i \circ \Psi_H^{-1})_{0,\Delta_H},$$
$$D_{H,i} := (\theta_{hp}^{n-1}, \xi_i \circ \Psi_H^{-1})_{0,\Delta_H}.$$

The main advantage of the proposed fictitious domain approach is that the integrals defined on Δ_H and $\Upsilon_H, \Upsilon_{\mathcal{D},H}, \Upsilon_{R,H}$ have to be computed on simple shapes given by hexahedrons, tetrahedrons, prisms and pyramids as well as rectangles and triangles. This can easily be done by using a Gauss quadrature on $[-1,1]^k$ with $k = 2, 3$ and a transformation from $[-1,1]^k$ to $T \in S^n(H)$ or $F \in \mathcal{F}'(H)$, respectively. In the case of hexehedrons and rectangles, this transformation is affine linear. For tetrahedrons, prisms, pyramids and triangles, a Duffy transformation can be used which, however, is non-linear, cf. [6,11]. To integrate over a pyramid, it is also convenient to use a subdivision into two tetrahedrons.

The exclusive use of hexahedron meshes allows to solely apply tensor product shape functions to span the space of polynomials $\mathbb{P}_q$, i.e.

$$\xi_{\alpha(i,j,k)}(x_1, x_2, x_3) = \eta_i(x_1)\eta_j(x_2)\eta_k(x_3)$$

with polynomials η_i of degree i and $i, j, k = 0, \ldots, q$ as well as a suitable numbering α. This structure can be exploited for an efficient evaluation and integration of the shape functions, cf. [11]. The element matrices of the paraxial hexahedrons, which do not intersect the cutting surface, can be computed by a simple scaling of a reference element matrix on $[-1,1]^3$ which is computed in advance in a preprocessing step. This leads to an efficient assembling of the matrices K, A and M, since no further integrations have to be performed on the hexahedrons which do not intersect the cutting surface. Moreover, the representations of K, A and M with (19), (20) and (23) in conjunction with the proposed scaling of reference element matrices enable to solve the algebraic systems (18) and (22) by an iterative scheme without storing these matrices. Only the element matrices corresponding to hexahedrons H at the cutting surface have to be stored and, if necessary, reassembled if the polynomial degree changes, i.e. if $p^{n-1}(H) \neq p^n(H)$.

The Neumann data as given in Section 4 leads to a local increase of stresses in the contact area of the milling cutter and the workpiece. We expect that the solution u and θ have low regularity there and high regularity away from this area. In order to obtain high (or even exponential) convergence with hp-methods, it is well known, that one should use small mesh sizes and small polynomial degrees in areas with low regularity, in contrast large mesh sizes and large polynomial degrees should be used in areas with high solution regularity. Algorithm 2 generates meshes with small mesh sizes at the cutting surface, whereas the mesh size of the hexahedrons is increased away from the cutting surface. Since the contact area of the milling cutter is a part of the cutting surface, the low regularity properties are automatically captured by the mesh refinement. However, it is supposable that high regularity also occurs at the cutting surface away from the contact area where, in principle, a large mesh size with high polynomial degree is preferable. Hence, we have a conflict resulting

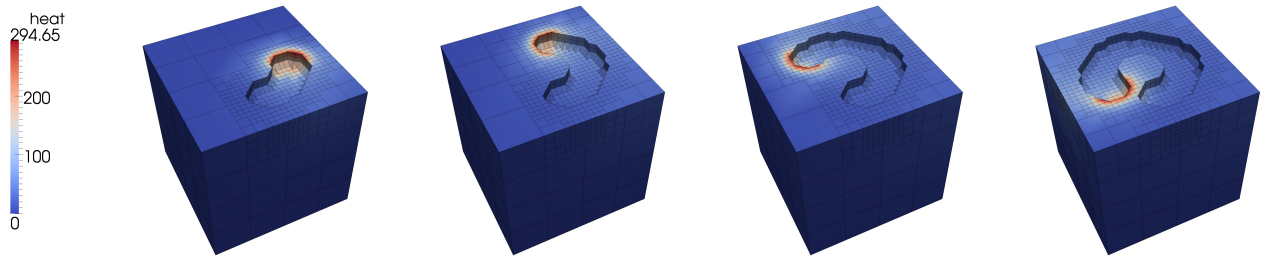


Fig. 7 Heat distribution in the milled workpiece

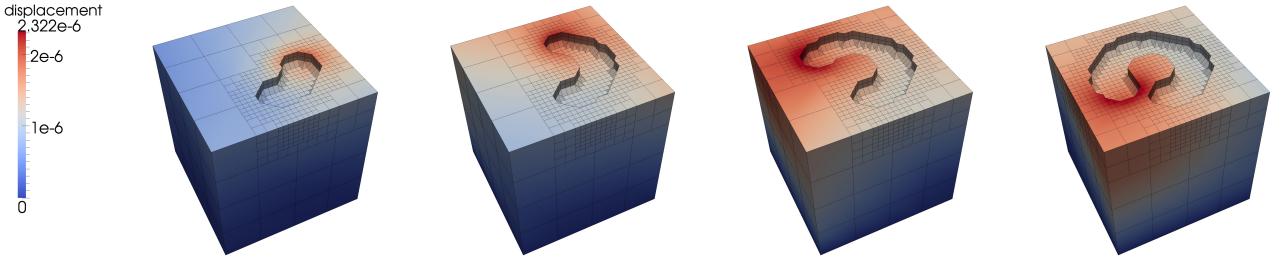


Fig. 8 Euclidian norm of the displacement field.

from the approximation quality of the surface on the one hand and from the mesh requirements of the finite element method on the other hand. This conflict may be solved by coarsening the hexahedrons at the cutting surface away from the contact area and by preserving their subdivisions which are still used to define the sets Δ_H and $\Upsilon_H, \Upsilon_{\mathcal{D},H}, \Upsilon_{R,H}$.

An obvious choice for the polynomial degree distribution p^n is to choose the degree according to the mesh size as depicted in Figure 6(a). In order to capture the probably high regularity of the solutions away from the contact area of the milling cutter, one may also choose the degree as a function of the distance to the contact area, see 6(b). In both cases, it is not clear whether these polynomial degree distributions together with the mesh refinement done by Algorithm 2 is appropriate to obtain a desired accuracy of the finite element solutions u_{hp}^n and θ_{hp}^n. Using an adaptive *hp*-refinement strategy based on error control, one may essentially improve the accuracy of the finite element approximation. In literature, several *hp*-strategies are proposed. However, in three dimensions, there are only few strategies, in particular, dealing with anisotropic mesh refinement and degree distributions. Here, we refer to the *hp*-refinement strategies relying on projection based interpolation techniques, [3,4,2,5].

6 Numerical experiments

In the numerical experiments, we study the heat diffusion in the workpiece and the deformation resulting from thermal stresses. The raw workpiece is represented by $\Omega^0 := [-20\,\mathrm{mm}, 20\,\mathrm{mm}]^3$; it is clamped at the bottom side $\Gamma_D := [-20\,\mathrm{mm}, 20\,\mathrm{mm}]^2 \times \{-20\,\mathrm{mm}\}$. The milling cutter is described by its cylindric hull with radius 4 mm. The cutter is moved along a circular path with radius 10 mm and center $(0,0)$. The depth of engagement is 3 mm in the top side $[-20\,\mathrm{mm}, 20\,\mathrm{mm}]^2 \times \{20\,\mathrm{mm}\}$ of the workpiece. Algorithm 2 generates the mesh $\mathcal{H}^n$ with minimal hexahedron length $d = 1.2\,\mathrm{mm}$. It is shown in Figure 5. In particular, the hexahedrons at the cutting surface as well as their subdivisions are depicted. We apply Hooke's material law with Young's modulus $E := 69 \cdot 10^6\,\mathrm{kg\,s^{-2}\,mm^{-1}}$ and Poisson number $\nu := 0.34$ which correspond to aluminium AL7075 with density $\rho := 2810 \cdot 10^{-9}\,\mathrm{kg\,mm^{-3}}$. The initial temperature and ambient temperature is set to $\theta_0 := \theta_a := 293\,\mathrm{K}$. For simplicity, we set the fraction of the mechanical cutting work μW defining the heat input at the contact area to $\mu W := 300\,\mathrm{N\,mm^{-2}}$. We refer to [18] for the computation of the mechanical work based on Kienzle's equation and the determination of the chip thickness in normal direction of the cut. The heat conductivity is set to $k := 115 \cdot 10^3\,\mathrm{W\,mm^{-1}\,K^{-1}}$ and the specific heat is $c := 862 \cdot 10^6\,\mathrm{mm\,s^{-2}\,K^{-1}}$. The constant describing the heat transfer to air is given by $a := 145\,\mathrm{kg\,s^{-3}\,K^{-1}}$.

In Figure 7, the heat distribution given by θ_{hp}^n is shown during the NC milling process. We find that the heat is induced at the contact area as expected and diffuses into the workpiece. The loss of heat resulting from the material removal is done automatically via the removal of hexahedrons and the L^2-projection which

transfers the previous heat distribution θ^{n-1} onto the current mesh given by $\mathcal{H}^n$ and S^n. In Figure 8, the Euclidian norm of the deformation $|u^n_{hp}|_2$ is depicted. Again, note that the deformation results from thermal stresses only. We observe that deformations particularly arise in the inner circle at the top side of the milled workpiece. A subsequent finishing process in this area before cooling down would lead to essential deviation of the workpiece surface from its designed shape.

Acknowledgements We would like to thank Friederike Hellwig for her Latex support and for proofreading this paper.

References

1. Boley, B.A., Weiner, J.: Theory of thermal stresses. New York and London: John Wiley & Sons, Inc. XVI, 586 p. (1960)
2. Demkowicz, L.: Computing with *hp*-adaptive finite elements. Vol. 1: One- and two-dimensional elliptic and Maxwell problems. Chapman & Hall/CRC (2007)
3. Demkowicz, L., Buffa, A.: H^1, $H(\mathrm{curl})$ and $H(\mathrm{div})$-conforming projection-based interpolation in three dimensions: Quasi-optimal p-interpolation estimates. Comput. Methods Appl. Mech. Eng. **194**(2-5), 267–296 (2005)
4. Demkowicz, L., Kurtz, J.: Projection-based interpolation and automatic hp-adaptivity for finite element discretizations of elliptic and Maxwell problems. ESAIM, Proc. **21**, 1–15 (2007)
5. Demkowicz, L., Kurtz, J., Pardo, D., Paszyński, M., Rachowicz, W., Zdunek, A.: Computing with *hp*-adaptive finite elements. Vol. II: Frontiers: Three-dimensional elliptic and Maxwell problems with applications. Chapman & Hall/CRC (2008)
6. Duffy, M.G.: Quadrature over a pyramid or cube of integrands with a singularity at a vertex. SIAM J. Numer. Anal. **19**, 1260–1262 (1982). DOI 10.1137/0719090
7. Düster, A., Parvizian, J., Yang, Z., Rank, E.: The finite cell method for three-dimensional problems of solid mechanics. Comput. Methods Appl. Mech. Eng. **197**(45-48), 3768–3782 (2008). DOI 10.1016/j.cma.2008.02.036
8. Evans, L.C.: Partial differential equations. Graduate Studies in Mathematics. 19. Providence, RI: American Mathematical Society (AMS). xvii, 662 p. (1998)
9. Fluegge, S.e.: Handbuch der Physik. Band VIa/2: Festkörpermechanik II. Bandherausgeber: C. Truesdell. Encyclopedia of physics. Vol. VIa/2: Mechanics of solids II. Editor: C. Truesdell. Berlin-Heidelberg-New York: Springer-Verlag (1972)
10. Haupt, P.: Continuum mechanics and theory of materials. Transl. from German by Joan A. Kurth. 2nd ed. Berlin: Springer (2002)
11. Karniadakis, G., Sherwin, S.: Spectral/*hp* element methods for computational fluid dynamics. 2nd ed. Numerical Mathematics and Scientific Computation. Oxford: Oxford University Press. (2005). DOI 10.1093/acprof:oso/9780198528692.001.0001
12. Parvizian, J., Düster, A., Rank, E.: Finite cell method. h- and p-extension for embedded domain problems in solid mechanics. Comput. Mech. **41**(1), 121–133 (2007). DOI 10.1007/s00466-007-0173-y
13. R., H.T.J.: The finite element method. Linear static and dynamic finite element analysis. Reprint of the 1987 original ed. Mineola, NY: Dover Publications. xxii, 682 p. (2000)
14. Salençon, J.: Handbook of continuum mechanics. General concepts. Thermoelasticity. Transl. from the French by Stephen Lyle. Berlin: Springer. (2001)
15. Schröder, A.: Constraints coefficients in *hp*-FEM. In: K. Kunisch (ed.) Numerical mathematics and advanced applications. Proceedings of ENUMATH 2007, pp. 183–190. Springer Berlin / Heidelberg (2008)
16. Schröder, A.: Constrained approximation in *hp*-FEM: Unsymmetric subdivisions and multilevel hanging nodes. In: Lecture Notes in Computational Sciences and Engineering (ICOSAHOM Proceedings), vol. 76, pp. 317–326 (2010)
17. Solin, P., Segeth, K., Delezel, I.: Higher-order finite element methods. Studies in Advanced Mathematics. CRC Press, Boca Raton (2004)
18. Surmann, T., Ungemach, E., Zabel, A., Joliet, R., Schröder, A.: Simulation of the Temperature Distribution in NC-Milled Workpieces. Advanced Materials Research **223**, 222–230 (2011)
19. Wilmanski, K.: Continuum thermodynamics. Part I: Foundations. Series on Advances in Mathematics for Applied Sciences 77. Hackensack, NJ: World Scientific. (2008)