Fortschritt-Berichte VDI

VDI

M. Sc. Dipl.-Ing. (FH) Patrick Erbts,
Hamburg

Partitioned solution strategies for electro-thermo-mechanical problems applied to the field-assisted sintering technology

Partitioned solution strategies for electro-thermo-mechanical problems applied to the field assisted sintering technology

Vom Promotionsausschuss der
Technischen Universität Hamburg
zur Erlangung des akademischen Grades

Doktor-Ingenieur (Dr.-Ing.)

genehmigte Dissertation

von

M.Sc. Dipl.-Ing. (FH) Patrick Erbts

aus

Buchholz i. d. Nordheide

2016

Vorsitzender des Promotionsausschusses
Prof. Dr. rer.nat. habil. Norbert Hoffmann

Erstgutachter
Prof. Dr.-Ing. habil. Alexander Düster

Zweitgutachter
Prof. Dr.-Ing. habil. Stefan Hartmann

Tag der mündlichen Prüfung: 23.11.2015

Fortschritt-Berichte VDI

Partitioned solution strategies for electro-thermo-mechanical problems applied to the field-assisted sintering technology

VDI verlag

Dedicated to engineers and scientists in the field of coupled problems and computational mechanics, this thesis addresses partitioned solution strategies for electro-thermo-mechanically coupled problems applied to the field-assisted sintering technology (FAST). By simultaneously applying uniaxial pressure and an electric current to generate high heating rates, the FAST process offers short production cycles for sintering materials. To approach the process conditions at high temperatures in a realistic and holistic way radiative heat transfer is numerically treated as an additional field. Finally, a fully coupled four-field problem is composed where for the electric, thermal and mechanical fields the finite element method is applied while solving the radiation field using computational fluid dynamic (CFD) solvers. The numerical results are compared to experiments. Moreover, an in-depth study of coupling algorithms is carried out to improve the convergence of the partitioned solution procedure.

Vorwort

Die vorliegende Arbeit wurde von mir während meiner Zeit als wissenschaftlicher Mitarbeiter am Institut für Konstruktion und Festigkeit von Schiffen in der Arbeitsgruppe *Numerische Strukturanalyse mit Anwendungen in der Schiffstechnik* der Technischen Universität Hamburg von April 2011 bis Mai 2015 angefertigt. Sie entstand im Rahmen des von der Deutschen Forschungsgemeinschaft (DFG) geförderten Projektes *Electro-thermo-mechanical modeling of the field-assisted sintering technology using high-order finite elemenst validated by experiments*. Während dieser Zeit habe ich viel Unterstützung erfahren, wofür ich mich im folgenden besonders bedanken möchte.

Großer Dank gilt Prof. Alexander Düster der mich in den vergangenen Jahren begleitet hat und mir immer mit Rat und Tat zur Seite stand. Seine Tür stand mir immer offen, unsere Diskussionen und Gespräche waren Ansporn und Motivation zugleich und hatten großen Anteil am Gelingen dieser Arbeit. Für seine angenehme Betreuung und Förderung möchte ich mich herzlichst bedanken.

Weiterhin danke ich Prof. Stefan Hartmann von der TU Clausthal – den ich im Rahmen unseres gemeinsamen Forschungsprojektes kennen und schätzen gelernt habe – für die Anfertigung des zweiten Gutachtens. Zudem danke ich Prof. Norbert Hoffmann als Vorsitzenden des Promotionsausschusses und zusätzlichen Gutachter, sowie Prof. Thomas Rung als weiteren zusätzlichen Gutachter.

Weiter möchte ich mich bei allen Mitgliedern des FAST Forschungsprojektes für die gute Zusammenarbeit bedanken: bei Prof. Stefan Hartmann und Dr.-Ing. Steffen Rothe (TU Clausthal), bei Prof. Ernst Rank, Dr.-Ing. Stefan Kollmannsberger und M.Sc. Nils Zander (TU München) sowie bei Prof. Zohar Yosibash und bei Prof. Nahum Frage (Ben-Gurion Universität des Negev).

Allen Mitarbeitern vom Institut M-10 sei ebenfalls gedankt. Insbesondere M.Sc Meysam Joulaian mit dem ich mir vier Jahre ein Büro geteilt habe, Dr.-Ing. Horst Höft bei der Bewältigung von Computerproblemen jeglicher Art und Jutta Henrici für alle organisatorischen Dinge.

Schlussendlich, einen herzlichen Dank meiner Familie und Freunde für Ihre Unterstützung. Der wichtigsten Person, meiner Freundin Katja, die dafür gesorgt hat, dass ich an den Wochenenden vor meinem Computer überlebt habe, und die mich auch in allen anderen Dingen immer unterstützt hat, gelten die letzten Worte. Vielen Dank.

Patrick Erbts   Hamburg, im April 2016

Für Joscha

Contents

Nomenclature

A distinction is made in scalars, vectors, tensors, and matrices and the following notation is introduced: scalars are written in italic letters A, vectors in the Euclidean space are indicated by arrows $\vec{a}$, second-order tensors are written with an under-tilde $\underset{\sim}{S}$. This holds also for Greek letters. For high-order tensors calligraphic letters with an under-tilde $\underset{\sim}{\mathcal{C}}$ are used. Matrices are written in bold letters. The combination of an italic and bold letter indicates local finite element matrices $\boldsymbol{A}$, whereas standard bold letters $\mathbf{A}$ refer to global matrices. In addition, lowercase letters $\mathbf{a}$ or $\boldsymbol{a}$ are used for column matrices. This notation is used throughout this thesis except where it is explicitly mentioned in the text.

Some frequently used tensor operations are also summarized in the following. Further reading on tensor-algebra and tensor-analysis for continuum mechanics is provided in [84, 13] for instance.

Tensors

A	Scalar value
$\vec{a} = a_i\,\vec{g}_i$	First-order tensor (vector)
$\underset{\sim}{S} = S_{ij}\,\vec{g}_i \otimes \vec{g}_j$	Second-order tensor
$\underset{\sim}{\mathcal{C}} = \mathcal{C}_{i_1\ldots i_n}\,\vec{g}_{i_1} \otimes \vec{g}_{i_2} \otimes \ldots \otimes \vec{g}_{i_n}$	Tensors of higher order

Matrices / column matrices

$\mathbf{a}$	Global column matrix
$\mathbf{A}$	Global matrix
$\boldsymbol{a}$	Local finite element column matrix
$\boldsymbol{A}$	Local finite element matrix

Mathematical operators

$\vec{u} \otimes \vec{v} = u_i\, v_j\, \vec{g}_i \otimes \vec{g}_j$ Dyadic product

$\underset{\sim}{S} \cdot \underset{\sim}{F} = S_{ij}\, F_{ij}$ Inner, scalar or dot product

$\underset{\approx}{S}\, \underset{\approx}{F} = S_{ij}\, F_{jk}\, \vec{g}_i \otimes \vec{g}_k$ Tensor product

$\underset{\sim}{S}^T = S_{ij}\, \vec{g}_j \otimes \vec{g}_i$ Transposed tensor

$\underset{\approx}{S}^{-1}$ Inverse of a tensor

$\operatorname{tr} \underset{\sim}{S} = S_{ii}$ Trace of a tensor (first invariant I_S)

$\det \underset{\sim}{S}$ Determinant of a tensor (third invariant III_S)

div, Div   Divergence operator

grad, Grad   Gradient operator

∇ Nabla operator

List of symbols

A list of the main symbols is given in the following. Subordinate variables of minor importance – typically appearing only once in the text – are not listed and are explicitly mentioned. Due to the multitude of different variables, a double seizure of some symbols cannot be avoided. This is also mentioned in the text.

Scalars

α	Absorptance
α_Θ	Thermal expansion coefficient
α_φ	Linear temperature coefficient
β	Extinction coefficient
γ	Thermo-elastic coupling term
ϵ	Tolerance
ε	Emissivity or emittance
η	Wave-number
ϑ	Thermal stretch-ratio
θ	Angle of colatitude
Θ	Temperature
κ	Absorption coefficient
$\varkappa$	Wave-length
Λ	Iteration residual
λ	First Lamé constant
λ_Θ	Heat conduction coefficient
λ_φ	Electric conduction coefficient
μ	Second Lamé constant / Shear modulus
ν	Viscosity
ξ	Entropy per unit volume
ϖ	Number of solver calls
ρ	Mass density
ρ_c	Charge density
ϱ	Reflectance
ς	Coupling iterations per time-step
σ_{sb}	Stefan-Boltzmann constant
σ_S	Scatter coefficient
Σ	Entropy

τ	Transmittance
υ	Wave-frequency
Υ	Solid angle
φ	Electric potential
ϕ	Angle of latitude
Φ	Scatter phase function
ψ	Angle of longitude
Ψ	Helmholtz free-energy
ω	Relaxation coefficient
Ω	Domain / Configuration
Ω_e	Element volume
$a,\ A$	Area
c_0	Speed of light in vacuum
c_{p}	Heat capacity / specific heat
$d,\ D$	Dissipation
e	Specific internal energy
E	Emissive power
f	(Angular) Frequency
F	View factor
G	Incident radiation
h	Electric charge
H	Irradiation
I	Radiative intensity
J	Determinant of the deformation gradient
K	Bulk modulus
$l,\ L$	Length
$m,\ M$	Mass
N	Shape function
p	(Polynomial) Order
p_f	Fluid pressure
P	Point
$q,\ Q$	Heat flux
$r,\ R$	Heat source
s	Path
S	Distance
t	Time
T	Time interval
U	Volumetric part of the strain-energy density function
$v,\ V$	Volume
w	Weights / weight function
$\bar{w}$	Isochoric part of strain-energy density function
W	Strain-energy density function
$x,\ X$	Coordinates
Y	Surface radiosity
$\mathcal{E}$	Internal energy

$\mathcal{K}$	Kinetic energy
$\mathcal{Q}$	Thermal work
$\mathcal{W}$	Mechanical work

Vectors

$\vec{\eta}$	Test function
$\vec{\chi}$	Mapping function
$\vec{a}, \vec{A}$	Surface element vector
$\vec{b}$	Body force density vector per unit mass
$\vec{d}$	Domain displacement vector / boundary displacements
$\vec{e}, \vec{E}$	Electric field intensity
$\vec{f}, \vec{F}$	Force vector
$\vec{g}$	Basis vector in current configuration
$\vec{G}$	Basis vector in reference configuration
$\vec{h}$	Rotational or angular momentum vector
$\vec{j}, \vec{J}$	Electric current density vector
$\vec{k}$	Gravitation vector
$\vec{l}$	Linear or translational momentum vector
$\vec{m}$	Moment vector
$\vec{n}, \vec{N}$	Normal vector
$\vec{q}, \vec{Q}$	Heat flux vector
$\vec{q}_r, \vec{Q}_r$	Radiative heat flux vector
$\vec{r}$	Location or distance vector
$\vec{s}$	Direction vector
$\vec{t}, \vec{T}$	Traction vector
$\vec{u}, \vec{U}$	Displacement vector
$\vec{v}$	Velocity vector
$\vec{x}, \vec{X}$	Coordinate vector

Tensors

$\underset{\sim}{\varepsilon}$	Linear strain tensor
$\underset{\sim}{\lambda}_\Theta, \underset{\sim}{\Lambda}_\Theta$	Heat conductivity tensor
$\underset{\sim}{\lambda}_\varphi, \underset{\sim}{\Lambda}_\varphi$	Electric conductivity tensor
$\underset{\sim}{\sigma}$	Cauchy stress tensor
$\underset{\sim}{b}$	Left Cauchy-Green tensor
$\underset{\sim}{C}$	Right Cauchy-Green tensor
$\underset{\sim}{d}$	Rate of deformation tensor
$\underset{\sim}{e}$	Euler-Almansi strain tensor
$\underset{\sim}{E}$	Green-Lagrange strain tensor
$\underset{\sim}{F}$	Deformation gradient tensor
$\underset{\sim}{H}$	Displacement gradient tensor
$\underset{\sim}{I}$	Unit tensor
$\underset{\sim}{l}$	Velocity gradient tensor

$\underset{\sim}{P}$	Piola-Kirchhoff stress tensor
$\underset{\sim}{R}$	Rotation tensor
$\underset{\approx}{S}$	2nd Piola-Kirchhoff stress tensor
$\underset{\sim}{U}$	Right stretch tensor
$\underset{\sim}{V}$	Left stretch tensor
$\underset{\sim}{w}$	Spin tensor
$\underset{\approx}{\mathcal{C}}$	Elasticity tensor
$\underset{\sim}{\mathcal{I}}$	First fundamental tensor

Global matrices / column matrices

$\boldsymbol{\alpha}$	Vector of interpolation coefficients
$\boldsymbol{\Theta}$	Discrete temperature vector
$\boldsymbol{\chi}$	Global coordinates vectors
$\boldsymbol{\Phi}$	Basis function matrix
$\boldsymbol{\varphi}$	Discrete electric potential vector
$\mathbf{A}$	System matrix
$\mathbf{B}$	Broyden's matrix
$\mathbf{d}$	Data transfer vector
$\mathbf{D}$	View factor matrix
$\mathbf{e}_b$	Discrete black body emissive power
$\mathbf{G}$	System of nonlinear equations
$\mathbf{G}_M$	System of nonlinear equations of the mechanical field
$\mathbf{G}_\Theta$	System of nonlinear equations of the thermal field
$\mathbf{G}_\varphi$	System of equations of the electric field
$\mathbf{H}$	Inverse of Broyden's matrix
$\mathbf{J}$	Jacobian matrix
$\mathbf{K}_{T,M}$	Global tangential stiffness matrix of the mechanical field
$\mathbf{K}_S$	Geometric part of $\mathbf{K}_{T,M}$
$\mathbf{K}_C$	Constitutive part of $\mathbf{K}_{T,M}$
$\mathbf{K}_{T,\Theta}$	Global tangential stiffness matrix of the thermal field
$\mathbf{K}_\varphi$	Global stiffness matrix of the electric field
$\mathbf{M}$	Mass or system matrix
$\mathbf{n}$	Nearest neighbors vector
$\mathbf{p}$	Vector of polynomials
$\mathbf{P}$	Matrix of polynomial vectors
$\mathbf{p}_M$	Load vector of the mechanical field
$\mathbf{p}_\Theta$	Load vector of the thermal field
$\mathbf{p}_\varphi$	Load vector of the electric field
$\mathbf{q}$	Discrete heat flux vector
$\mathbf{r}, \mathbf{R}$	Discrete (iteration) residual
$\mathbf{u}$	Global displacement vector
$\mathbf{v}$	Vector of a transformed sequence
$\mathbf{x}$	Coordinate vector
$\mathbf{y}$	Solution / sequence vector
$\mathbf{Y}$	Matrix of discrete solution vectors

$\mathbf{z}$	Solution / sequence vector
$\mathbf{Z}$	Matrix of discrete solution vectors

Local finite element matrices / column matrices

$\boldsymbol{\Theta}_e$	Element temperature vector
$\boldsymbol{\Lambda}_\Theta^e$	Heat conductivity matrix
$\boldsymbol{\Lambda}_\varphi^e$	Electric conductivity matrix
$\boldsymbol{\varphi}_e$	Element electric potential vector
$\boldsymbol{\xi}$	Local coordinate vector
$\boldsymbol{B}$	Matrix of shape function derivatives
$\boldsymbol{B}_\mathrm{L}$	Strain-displacement matrix
$\boldsymbol{b}_e$	Element body force vector
$\boldsymbol{C}_e$	Material matrix
$\boldsymbol{E}_e$	Green-Lagrange strains in Voigt notation
$\boldsymbol{F}_e$	Element deformation gradient
$\boldsymbol{J}_e$	Element Jacobian matrix
$\boldsymbol{K}_\mathrm{T,M}^e$	Tangential element stiffness matrix
$\boldsymbol{K}_\Theta^e$	Thermal element stiffness matrix
$\boldsymbol{K}_\varphi^e$	Electric element stiffness matrix
$\boldsymbol{N}$	Shape function matrix
$\boldsymbol{p}_\mathrm{M}^e$	Mechanical element load vector
$\boldsymbol{p}_\Theta^e$	Thermal element load vector
$\boldsymbol{p}_\varphi^e$	Electric element load vector
$\boldsymbol{Q}_e$	Mapping function
$\boldsymbol{S}_e$	Stress tensor in Voigt notation
$\bar{\boldsymbol{t}}_e$	Element traction vector
$\boldsymbol{u}_e$	Element displacement vector
$\boldsymbol{x}_e,\ \boldsymbol{X}_e$	Element coordinate vector

Functionals, operators and spaces

$\mathcal{A}$	System operator
$\mathcal{B}$	Continuum body
$\mathcal{F}$	Solution operator in fixed-point iteration
$\mathcal{G}$	Solution operator in fixed-point iteration
$\mathcal{G}_\mathrm{M}$	Functional of the mechanical field
$\mathcal{G}_\Theta$	Functional of the thermal field
$\mathcal{G}_\varphi$	Functional of the electric field
$\mathcal{R}$	Residual operator
$\mathcal{S}$	Sequence of vectors
$\mathcal{T}$	Transformed sequence of vectors
$\mathcal{V}$	Test space

Frequently used sub- and superscripts

$(\,\cdot\,)_0$	Reference / initial value

$(\cdot)_b$	Black-body value
$(\cdot)^e$	Finite element quantity (e-th element)
$(\cdot)^i$	i-th iteration, i-th sequence
$(\cdot)^k$	k-th coupling iteration
$(\cdot)_n$	n-th time-step
$(\cdot)_\mathrm{M}$	Mechanical part of $(\cdot)$
$(\cdot)_\Theta$	Thermal part of $(\cdot)$
$(\cdot)_\varphi$	Electric part of $(\cdot)$
$(\cdot)^*$	Extra- or interpolated value

Abbreviations and acronyms

Acronym	Description	Page
BC	Boundary Conditions	124
BR	Broyden Method	91
CFD	Computational Fluid Dynamics	59
CV	Control-Volume	59
DAE	Differential-Algebraic Equation	59
DAR	Dynamic Aitken Relaxation	81
DSR	Dynamic Secant Relaxation	82
EXP	Explicit	115
FAST	Field Assisted Sintering Technology	1
FEM	Finite Element Method	6, 41
FSI	Fluid-Structure Interaction	4, 37
FVM	Finite Volume Method	59
GJ	Gauss-Jacobi	109
GMRES	Generalized Minimal Residual Method	4, 94
GS	Gauss-Seidel	109
IMP	Implicit	115
IQN	Interface Quasi-Newton	89
LE	Line Extrapolation	86
MLNA	Multi-Level Newton Algorithm	59
MP	Mechanical Predictor	132
MPI	Message Passing Interface	112
QN	Quasi-Newton (Method)	89
fvDOM	finite volume Discrete Ordinate Method	66
RTE	Radiative Transfer Equation	7, 35
SIMPLE	Semi Implicit Method for Pressure Linked Equations	63
SOR	Successive Over Relaxation	82
SUR	Successive Under Relaxation	82
TP	Thermal Predictor	119
VFM	View Factor Method	64

Abstract

With increasing computational capacity, simulations of multi-physically coupled problems become of more interest in many industrial applications. The numerical treatment of multi-field interactions calls for flexible and robust solution strategies. A partitioned coupling strategy has the advantage of high flexibility and allows for combinations of different software and specialized solvers for the physical fields involved. It divides the coupled system into iterations of subproblems with repetitive data exchange.

As a multi-field example, the electro-thermo-mechanical process of the *field assisted sintering technology* (FAST) is taken under consideration. FAST is an innovative technique for the compaction of powder materials. It offers short production cycles by simultaneously applying a uniaxial pressure and an electric current in order to generate high heating rates and hot temperatures by means of Joule heating. During processing, the temperature development is an important feature to obtain optimal process conditions. For high temperatures, the most prominent mechanism to transfer thermal energy is thermal radiation, which is why a comprehensive simulation of FAST should comprise the effects of radiating surfaces. This can be accomplished by treating the environment of the FAST machine tools as an additional individual field, denoted as the radiation field. It is coupled to the temperature and allows to model intricate interactions such as reflection or irradiation with other surfaces. This finally leads to a numerically challenging four-field problem that describes the FAST process. The electric, thermal and mechanical subproblems are solved using the finite element method (FEM) while the finite volume method is applied for the radiation field. Different numerical models are discussed to approximate the radiative transfer in vacuum and also in participating media.

Regarding the partitioned coupling strategy, the flexibility attribute comes at the expense of algorithmic stability. It is known that particularly strongly coupled problems can be unstable even if an implicit time stepping method is chosen for the subproblems. Here, external stabilization methods serve to increase the chances of stability. Typically, methods like this are known from the field of fluid-structure interaction (FSI), and they can be applied in connection with black-box solvers. Further, they can be used to improve the convergence and to reduce the computation time, as they accelerate the coupling iterations. In this thesis, several stabilization procedures are discussed. Based on sequential solver calls, a concept to design partitioned solution strategies for an arbitrary number of physical fields is proposed and applied to several numerical examples.

CHAPTER 1

Introduction

The ambition of realistically simulating complex physical and engineering problems often requires a solution of coupled multi-field systems. Nowadays, commercial solvers can be used to carry out coupled simulations for a large number of different problems. However, these solvers are often limited to applications of minor complexity: the physical fields either have to be solved on a common domain, or they need to be discretized by the same procedure, or they are restricted to weak coupling between the fields. In order to consider more complicated multi-physics approaches, additional effort has to be spent on the numerical solution procedure. It is mandatory to determine a suitable coupling algorithm for the problem under consideration. In this thesis, robust partitioned solution strategies are proposed that ensure high flexibility and that allow stable and efficient simulations. Their applicability is demonstrated on the example of the *field assisted sintering technology* (FAST), simulations of which are numerically challenging since it involves an interaction of electric, thermal and mechanical fields.

1.1 Motivation and state of the art

To begin with, the electro-thermo-mechanical process of the field assisted sintering technology is briefly reviewed. Emphasis is placed on the multi-physical modeling and on the thermal energy transfer during processing. Further aspects are the motivation to employ partitioned coupling strategies as well as a summary of the current state-of-the-art.

1.1.1 Multi-physical simulation of FAST

The field assisted sintering technology[1] [125] is an innovative process to manufacture sintering materials by means of high pressure and high temperatures. During processing, a powder material is consolidated by simultaneously applying a uniaxial pressure and an electric current to induce *Joule heating*. This leads

---

[1]In literature, the FAST process is also known as *spark plasma sintering* or *pulsed electric current sintering*, see [163] for instance.

to high heating rates and, thus, to short production cycles. The main tool-system of a FAST-machine is depicted in Figures 1.1(a) and 1.1(b). It typically consists of a die, a punch, a cone, and electrodes. All these tools are made of graphite, which is why the entire setup is referred to as *graphite tool-system*. The reason for choosing graphite can be seen in the good electric conductivity. Further, it is resistant to very high temperatures [146].

The pressing force acts on the top of the system where the electric current is applied as well. The powder material is placed inside the die and between the punches, as illustrated in Figure 1.1(a). The whole graphite tool-system is embedded in a vacuum chamber.

(a) Schematic sketch of the graphite tools.

(b) Graphite tools inside the vacuum chamber, from [146].
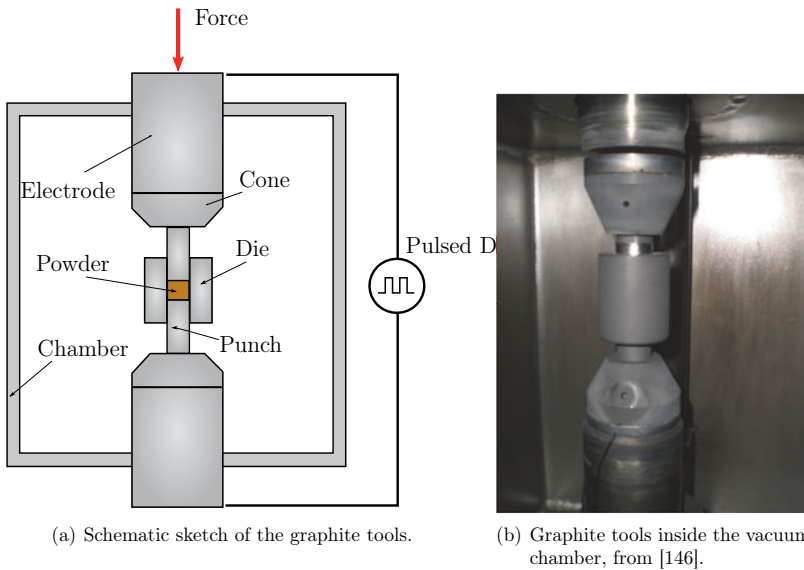
Figure 1.1: The FAST process: graphite tool-system and vacuum chamber.

In a multi-physical context, the FAST process is governed by an electric, a thermal and a mechanical field which are depending on each other. Thus, a coupled electro-thermo-mechanical problem needs to be solved to obtain a realistic prediction of the process. For this reason, the coupling effects between the involved fields shall be outlined briefly. Since Joule heating is used to achieve high heating rates, the electric field is connected to the thermal field. The thermal field is coupled to the stress field as temperature changes lead to thermal strains that cause thermal stresses. Due to the temperature dependency of several material parameters, concerning both graphite and the powder material, the thermal field is coupled to the electric and the mechanical field. Moreover, the thermal and the electric conductivity depend on the powder density and are therefore coupled

to the mechanical field.In case of large deformations, the geometric changes also influence the thermal and the electric field.

As reported in [176], the temperature evaluation inside the tool-system is important for attempts to optimize the process. To this end, all possible modes to transfer thermal energy are taken into consideration. Forced convection is used to cool the machine tools [163, 76], as the vacuum chamber and the electrodes typically feature a water-cooling system. Natural convection inside the chamber can be neglected due to nearly vacuum conditions [176]. Additionally, thermal radiation is of particular interest as the FAST process involves high temperatures at the tool surfaces. As thermal radiation is – simply spoken – proportional to the temperature to the power of four, it can be considered as the dominating mode of transferring thermal energy at high temperatures. In many articles, see [176, 163, 129, 76] for instance, thermal radiation is approximated as a boundary condition in the thermal field by using

$$q_{\mathrm{r}} = \varepsilon \, \sigma_{\mathrm{sb}} (\Theta_{\mathrm{s}}^4 - \Theta_{\infty}^4) \tag{1.1}$$

to prescribe the radiative heat flux across the radiating surfaces. In this relation, ε denotes the emissivity, σ_{sb} the Stefan-Boltzmann constant, Θ_{s} the surface temperature of the punch/die system, and Θ_{∞} is the ambient temperature. A slightly more complex approximation is proposed by SONG ET AL. [155] who relate the radiative heat loss to the chamber wall temperatures and the exposed surfaces of the tool-system. However, a comprehensive investigation how the temperature evolution is affected by the strongly radiating tool surfaces is – to the author's knowledge – not documented in literature.

1.1.2 Solution of coupled problems

In order to simulate the FAST process, a multi-physically coupled problem needs to be solved. Coupled electric-thermal computations to gain knowledge of the temperature evolution during processing were carried out in [176, 163], for instance. In the literature, there are also approaches that take the whole electro-thermo-mechanically coupled multi-field problem into consideration, see [114, 155, 129] for instance. In HARTMAN ET AL. [76], the temperature and stress distribution in the graphite tools and for fully compacted powder is computed and compared to experiments.

Aspects regarding the numerical solution procedure of the electro-thermo-mechanical system have been investigated by ROTHE ET AL. [149]. A temperature-dependent constitutive powder material model was proposed by ROTHE [146]. Therein, fully coupled electro-thermo-mechanical simulations using the commercial finite element solver *Abaqus* were carried out and also compared to experiments. Further reading can also be found in the literature listed in the references.

In the following, numerical procedures and coupling algorithms for the solution of multi-field problems are brought into focus. Generally, two different strategies are distinguished: a *monolithic scheme* that solves the entire problem

simultaneously or a *partitioned scheme* that divides the physical fields into an it-
eration of subproblems. In the scope of the latter, the subfields can be treated as
separated – thus allowing the use of specialized solvers. During the computation
of one field, the variables of the other fields are kept constant. In order to balance
the fields, several coupling iterations are required with repetitive data exchange.
Under the assumption that balance is achieved within one iteration, with a suf-
ficient accuracy, the problem is said to be *weakly* coupled, leading to so-called
staggered schemes. In the literature, it is often not clearly defined if the term *stag-
gered* involves iterating between the fields or not. This is why a clear definition
is introduced here: A scheme that requires more than one iteration is denoted as
implicit or as *multiple staggered*, whereas the other case is referred to as an *explicit*
or *single staggered* scheme.

   For the more obvious monolithic approach, balance between the fields in a con-
vergent computation is simultaneously achieved in one time-step. For an uncon-
ditional stable time-stepping method, this property can also be achieved by the
monolithic scheme, whereas the partitioned approach cannot guarantee stabil-
ity even if the solution of the subfields is stable. Further, the iterative character
can lead to poor convergence rates and, moreover, data need to be transferred
in every iteration. On the other hand, the partitioned approach has the great
advantage of being very flexible as it allows in many cases to combine differ-
ent software, black box solvers and also discretization schemes. In this thesis,
a partitioned strategy is followed for flexibility reasons. Such strategies have a
long tradition in the numerical treatment of coupled problems, as the works of
PARK & FELIPPA [132] and FELIPPA & GEERS [59] from the 1980s demonstrate.
The interaction between structure and thermal fields [58] and between fluid and
structure [60] were taken into consideration right from the beginning. Based on
these two examples, a simple classification can be made: the former is a typical
example for volume-coupled problems, whereas the latter is typical for surface-
coupled systems.

   With respect to partitioned coupling approaches, important contributions to the
field of volumetrically coupled thermomechanics were made by SIMO & MIEHE
[153] who introduced the so-called *isothermal split*, where the heat conduction
problem is solved following the mechanical field. This split is only conditionally
stable – which is why ARMERO & SIMO [3] proposed the so-called *adiabatic split*
for problems of nonlinear thermoelasticity. In this split, the mechanical phase at
isothermal conditions is replaced by a mechanical phase at adiabatic conditions.
An extension to nonlinear coupled thermoplasticity can be found in [4].

   Regarding surface-coupled fluid-structure interaction (FSI), a lot of research has
been conducted to retain algorithmic stability by means of special algorithms.
Such algorithms serve to increase the stability itself, and they also improve the
convergence properties of the partitioned approach. A prominent example is the
so-called *Aitken method* in a formulation given by KÜTTLER & WALL [106] to
solve strongly coupled FSI problems. Another promising method applied to
FSI was proposed by DEGROOTE ET AL. [36] who developed an *interface quasi-
Newton method*. Extensions to this method were made in [37, 38, 39]. In ERBTS &

DÜSTER [50], it was shown, for nonlinear thermoelasticity, that both algorithms can be used to achieve nearly unconditional stability when applying the isothermal split. There are several other numerical procedures to stabilize partitioned solution approaches for FSI problems: For example, in [49] the *Broyden method* is applied, whereas [107] utilizes so-called *extrapolation methods* . Further, *Jacobian-free Newton-Krylow methods* are investigated in [116]. Special *interface GMRES* solvers using subiteration were applied in [117, 118]. A performance study, including a comparison of different methods with regard to FSI-application, can be found in [119, 37].

1.2 Purpose and scope of the thesis

The purpose of the thesis is to propose a robust and efficient solution strategy for multi-physically coupled problems based on a partitioned coupling approach. In particular, this includes a framework for the class of coupled problems that involve more than two physical fields and that include volume and also surface coupling. The applicability of the coupling strategy is demonstrated in several numerical investigations. Besides academic examples, the simulation of the FAST process serves as an industrial application.

A further aspect to be addressed is the temperature evolution of the FAST process. Apart from conduction and convection, emphasis is devoted to the energy transfer by means of thermal radiation. In order to capture effects such as self-irradiation due to the geometry of the FAST tools or reflection by the chamber walls, radiation is considered as an individual physical field. This consequently means that the FAST process in this thesis involves four fields: the *mechanical*, *thermal*, *electric* and the *radiation field*, as shown in Figure 1.2.

Thanks to its flexibility, the proposed coupling strategy can be extended to simulate the complex four-field problem. To this end, special solvers for the solution of the thermal radiation field are integrated into the coupling procedure. With regard to the mathematical formulation, a *Dirichlet-Neumann* coupling decomposition between the thermal and the radiation field is proposed. This is done for the case of vacuum and also for the more general case of a participating medium. Further, a *Dirichlet* coupling between the radiation and the mechanical field is provided. Since the thermal and the mechanical field are connected to the radiation field by a common boundary, this coupling falls into the class of surface-coupled problems. Finally, simulations of the FAST process based on the Dirichlet-Neumann coupling formulation of the radiation field are compared to those of minor complexity, i.e. when applying Eq. (1.1). It is shown that the Dirichlet-Neumann coupling formulation can be used to approach experimental data with higher accuracy.

Another objective of this thesis is to improve the performance and the stability of partitioned coupling strategies. In order to improve the convergence and to retain algorithmic stability, a wide range of external stabilization and acceleration methods is investigated and studied with numerical examples. Some improve-
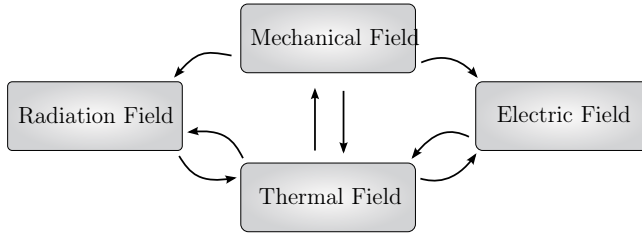
Figure 1.2: Coupled multi-field problem of the FAST process.

ments to these methods are proposed, and their importance is demonstrated for both volume- and surface-coupled problems. In the context of coupled three- or four-field problems, the question arises what might be the "best" sequence for the fields to be called in. Moreover, the point of application of the convergence acceleration methods during the coupling sequence is of principle interest. Besides these two points, there are further characteristics that might be of interest to devise a well-suited and problem-oriented partitioned coupling procedure. Another purpose of this thesis is to try and find answers to these questions.

1.3 Outline of the thesis

The thesis is organized in six main chapters. This first chapter focuses on the motivation to rely on partitioned coupling strategies to simulate the FAST process and serves to outline the state-of-the-art of numerical solution procedures for multi-physically coupled problems.

   Chapter 2 serves to introduce the theoretical background and to briefly summarize the governing equations to describe the mechanical, thermal, electric and radiation fields. This is done in a strictly decoupled fashion to underline of the specific use of a partitioned coupling strategy. Particular attention is placed on the thermal radiation field, as the corresponding equations differ from the conventional continuum approach used to formulate the mechanical or the thermal field. For the coupling of the radiation field with the mechanical and the thermal field, a *Dirichlet-Neumann* coupling formulation is proposed at the end of this chapter.

   In **Chapter 3**, numerical methods to approximate the solution of the subfields are described. Aspects concerning the spatial and temporal discretization are discussed and different models to tackle thermal radiation are provided. Further objectives in this chapter are the introduction of the *finite element method* (FEM) for the mechanical, thermal and electric fields with regard to spatial discretization as well as a summary of the *finite volume method* (FVM) regarding the aspect of radiation in a participating medium.

Chapter 4 can be considered as the core of the thesis. In this chapter, partitioned solution strategies for an arbitrary number of coupled fields are proposed. Different ways to solve multi-physics problems are recapitulated and methods to improve the convergence properties are discussed. Since partitioned coupling strategies allow to combine different solvers, software and discretization schemes, methods to assure a consistent field transfer are required. Further, this chapter includes recommendations to set up an appropriate partitioned solution strategy.

Numerical examples for coupled four-field problems are given in **Chapter 5**. Apart from serving as an academic example for electro-thermo-mechanical radiative coupling, the proposed partitioned solution strategy is used to simulate the FAST process. In this simulation, the temperature development during processing of the tools and the chamber is computed and the results are compared to experiments.

Finally, **Chapter 6** summarizes the achievements of the thesis and offers an outlook regarding future tasks, open questions and further research possibilities.

CHAPTER 2

Governing equations

Descriptions of multi-physically coupled engineering problems require a mathematical formulation of the process and a substantiated background in the mechanics of solids and fluids. To this end, a deep understanding in the field of continuum mechanics is mandatory. The continuum theory of the mechanics of solids and fluids is well documented in many textbooks and lecture notes; see [77, 84, 172, 161, 5, 8, 13, 26] as a few selected examples. For this reason, only a brief introduction into the theory will be given in this chapter. In addition to the formulation of the mechanical field, which describes the behavior of solid bodies under loads, special attention is placed on the formulation of the thermal and the electric fields. If all dependencies between the three fields are taken into account, this leads to a fully coupled electro-thermo-mechanical problem that substantiates the underlying mathematical description of the FAST process.

As previously mentioned, thermal radiation can be the dominating mode of heat transfer in many industrial applications. Unfortunately, the existence of thermal radiation may increase the complexity of a simulation significantly, especially if the interaction between the thermal field and its environment is to be studied in detail. This is due to the fact that the *radiative transfer equation* (RTE) needs to be solved, which is coupled with the energy equation of the surrounding fluid. This equation cannot be solved with the same procedures as used for the mechanical, the thermal and the electric field. As a consequence, thermal radiation is finally considered as a new additional field. Its theoretical formulation is discussed in this chapter as well.

2.1 Basic continuum mechanics

In the classical theory of continuum mechanics, the body under investigation is idealized as continuously distributed and homogeneous. In other words, smaller scales of the material are ignored, and the existence of grains in solid materials or particles in fluids are neglected. This idealization allows for an adequate description of the body which is subjected to loads and deformations. To begin with, this section addresses the basic kinematics that describe the motion of a solid, continuous body, followed by the stresses and balance equations. Finally, constitutive

relations to formulate the behavior of materials are given.

2.1.1 Kinematics

2.1.1.1 Configuration, motion and deformation

Descriptions of kinematic relations can be seen as one of the core aspect of continuum mechanics. Starting point is the mathematical formulation of the motion and deformation of a continuum body in the three-dimensional Euclidean space. To this end, a fixed Cartesian coordinate system is introduced in which all points P of a body $\mathcal{B}_0 \subset \mathbb{R}^3$ can be identified by a unique vector $\vec{X} = X_I \vec{G}_I \in \mathcal{B}_0$, where the X_I are the reference coordinates and the $\vec{G}_I$ are the unit base vectors referring to a Cartesian coordinate system. This configuration is called *reference* or *initial configuration*, denoted by Ω_0 and defined at time $t = t_0$. If the body starts to move or to deform, see also Figure 2.1, a new configuration Ω_t is introduced at time $t_t > t_0$. This configuration is called *current configuration*. As shown in Figure 2.1, the point P is mapped by a nonlinear invertible mapping function $\vec{\chi}(\vec{X}, t)$ to the point P_t at $\vec{x} = \vec{\chi}(\vec{X}, t)$ in the current configuration. The unique vector $\vec{x}(\vec{X}, t) = x_i \vec{g}_i \in \mathcal{B}_t$ of the deformed body $\mathcal{B}_t \subset \mathbb{R}^3$ is now given as a function of the reference coordinates $\vec{X}$ and the time t.
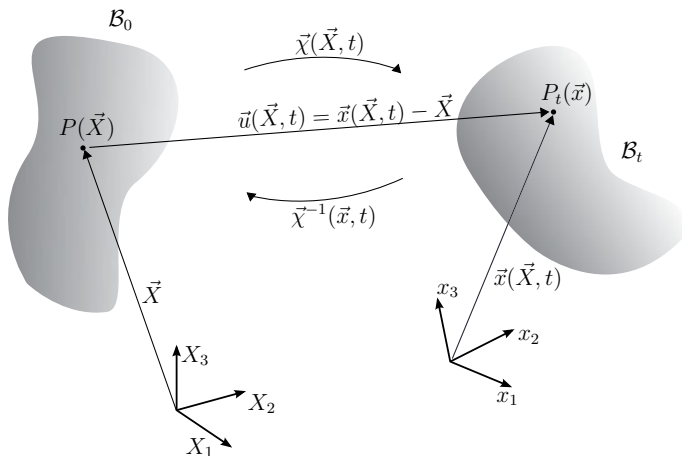
Figure 2.1: Basic kinematic relations: coordinates, configurations and displacement vector.

Thanks to the two configurations, there are two possible ways to formulate the further equations. In this thesis, the method of choice is the *material* or *Lagrangian formulation*, which refers to the reference configuration. The other approach formulates all equations in the current configuration and is known as *spatial* or *Eulerian description* [84]. In the scope of solid and structural mechanics, the Lagrangian description has some advantages – which is why this approach is used in the following.

The displacement vector $\vec{u}(\vec{X}, t)$ follows directly from Figure 2.1 and is the difference between the position vector of the spatial point P_t and its material counterpart P

$$\vec{u}(\vec{X}, t) = \vec{x}(\vec{X}, t) - \vec{X} = \vec{\chi}(\vec{X}, t) - \vec{X} \,. \tag{2.1}$$

Apart from the displacements, it is the change in deformation that is of interest here. It can be computed by means of the so-called *deformation gradient tensor* $\underset{\sim}{F}$, which reads

$$\underset{\sim}{F} = \frac{\partial \vec{x}}{\partial \vec{X}} = \mathrm{Grad}\, \vec{\chi}(\vec{X}, t) = \frac{\partial x_i}{\partial X_J} \, \vec{g}_i \otimes \vec{G}_J \,. \tag{2.2}$$

Therein, the vectors $\vec{g}_i$ and $\vec{G}_J$ are the Cartesian basis vectors referring to the current or the reference configuration. Moreover, the deformation gradient has a large significance since it can be used to map material line $\mathrm{d}\vec{X}$, surface $\mathrm{d}\vec{A}^1$ or volume elements $\mathrm{d}V$ of the reference configuration to the current configuration:

$$\mathrm{d}\vec{x} = \underset{\sim}{F} \, \mathrm{d}\vec{X} \,, \tag{2.3}$$

$$\mathrm{d}\vec{a} = (\det \underset{\sim}{F}) \, \underset{\sim}{F}^{-T} \, \mathrm{d}\vec{A} \,, \tag{2.4}$$

$$\mathrm{d}v = (\det \underset{\sim}{F}) \, \mathrm{d}V \tag{2.5}$$

Henceforth, the determinant of the deformation gradient is abbreviated with $J = \det \underset{\sim}{F} > 0$, which has to be larger than zero for any admissible deformation. The gradient of the displacement vector $\underset{\sim}{H}$ is related to the deformation gradient by means of

$$\underset{\sim}{H} = \mathrm{Grad}\, \vec{u}(\vec{X}, t) = \underset{\sim}{F} - \underset{\sim}{I} \,. \tag{2.6}$$

Following [84, 77], the deformation gradient can be multiplicatively split into two parts

$$\underset{\sim}{F} = \underset{\sim}{R}\,\underset{\sim}{U} = \underset{\sim}{V}\,\underset{\sim}{R} \,, \tag{2.7}$$

where $\underset{\sim}{R}$ is the rotation tensor, $\underset{\sim}{U}$ the right stretch tensor related to the undeformed reference configuration, and $\underset{\sim}{V}$ is the left stretch tensor defined in the current configuration. Generally, the rotation tensor describes rigid body rotations. Both stretch tensors are symmetric and contain the stretches that are responsible for

---

[1]The mapping of infinitesimal surface elements from the reference to the current configuration is also known as *Nanson's formula*.

the deformation of the body, which will lead to stress and strain. This splitting procedure of the deformation gradient is known as *polar decomposition*. The relation between both tensors can be determined by using the rotation tensor

$$\underset{\sim}{U} = \underset{\sim}{R}^T \underset{\sim}{V} \underset{\sim}{R}, \quad \underset{\sim}{V} = \underset{\sim}{R} \underset{\sim}{U} \underset{\sim}{R}^T. \tag{2.8}$$

The rotation tensor $\underset{\sim}{R}$ has the special property of being orthogonal, which means that $\underset{\sim}{R}^T = \underset{\sim}{R}^{-1}$ holds [84].

2.1.1.2 Strain tensors

In order to describe a stress state of a deformable body, it is necessary to introduce strain measures that are independent of rations and rigid body motions. Therefore, rigid body rotations can be excluded by means of

$$\underset{\sim}{F}^T \underset{\sim}{F} = \left(\underset{\sim}{R}\underset{\sim}{U}\right)^T \left(\underset{\sim}{R}\underset{\sim}{U}\right) = \underset{\sim}{U}^2 = \underset{\sim}{C}, \tag{2.9}$$

where $\underset{\sim}{C}$ is denoted as the *right Cauchy-Green tensor*. The next tensor to be introduced is

$$\underset{\sim}{E} = \frac{1}{2}\left(\underset{\sim}{C} - \underset{\sim}{I}\right) = \frac{1}{2}\left(\underset{\sim}{H} + \underset{\sim}{H}^T + \underset{\sim}{H}^T\underset{\sim}{H}\right), \tag{2.10}$$

which is known as the *Green-Lagrange strain tensor*[2] referring to the reference configuration. As shown in the equation above, see Eq. (2.10), the tensor $\underset{\sim}{E}$ can also be expressed in terms of the displacement gradient $\underset{\sim}{H}$. In the small strain case, the quadratic terms are neglected – which leads to the *linear* or *infinitesimal strain tensor*

$$\underset{\sim}{\varepsilon} = \frac{1}{2}\left(\underset{\sim}{H} + \underset{\sim}{H}^T\right). \tag{2.11}$$

Further, a strain measure acting in the current configuration is known as the *Euler-Almansi strain tensor* which reads [84]

$$\underset{\sim}{e} = \frac{1}{2}\left(\underset{\sim}{I} - \underset{\sim}{b}^{-1}\right). \tag{2.12}$$

In this equation, the tensor $\underset{\sim}{b} = \underset{\sim}{V}^2 = \underset{\sim}{F}\underset{\sim}{F}^T$ is denoted as the *left Cauchy-Green tensor*. In analogy to $\underset{\sim}{C}$, rigid body motions have also been excluded by this tensor.

For many material models, it is beneficial to decompose the deformation into a volume-changing and a volume-preserving (isochoric) part. To this end, the deformation gradient is multiplicatively split into

$$\underset{\sim}{F} = \hat{\underset{\sim}{F}}\bar{\underset{\sim}{F}}, \tag{2.13}$$

---

[2]$\underset{\sim}{E}$ can also be derived by considering the change in the squared lengths of the line elements $\mathrm{d}\vec{X}$ in the reference and $\mathrm{d}\vec{x}$ in the current configuration, see HOLZAPFEL [84], p. 79. With $\mathrm{d}\vec{x} = \underset{\sim}{F}\,\mathrm{d}\vec{X}$ follows

$$\mathrm{d}l^2 - \mathrm{d}L^2 = \mathrm{d}\vec{x}\cdot\mathrm{d}\vec{x} - \mathrm{d}\vec{X}\cdot\mathrm{d}\vec{X} = 2\,\mathrm{d}\vec{X}\cdot\underset{\sim}{E}\,\mathrm{d}\vec{X} \quad \rightarrow \quad 2\underset{\sim}{E} = \underset{\sim}{F}^T\underset{\sim}{F} - \underset{\sim}{I} = \underset{\sim}{C} - \underset{\sim}{I}.$$

where $\hat{\underset{\sim}{F}}$ is related to the volumetric deformation and $\bar{\underset{\sim}{F}}$ to the isochoric part of the deformation. Thus, the determinant can also be split into $J = \hat{J}\bar{J}$ such that $\bar{J} = 1$ holds. This multiplicative split, which was introduced by FLORY [63], finally leads to

$$\hat{\underset{\sim}{F}} = J^{1/3}\underset{\sim}{I} \quad \text{and} \quad \bar{\underset{\sim}{F}} = J^{-1/3}\underset{\sim}{F}\,. \tag{2.14}$$

Based on this split, the right Cauchy-Green tensor can be split into

$$\hat{\underset{\sim}{C}} = J^{2/3}\underset{\sim}{I} \quad \text{and} \quad \bar{\underset{\sim}{C}} = J^{-2/3}\underset{\sim}{C}\,. \tag{2.15}$$

Accordingly, it is also possible to split the left Cauchy-Green tensor in the current configuration as

$$\hat{\underset{\sim}{b}} = J^{2/3}\underset{\sim}{I} \quad \text{and} \quad \bar{\underset{\sim}{b}} = J^{-2/3}\underset{\sim}{b}\,. \tag{2.16}$$

2.1.1.3　Time dependency of motion and deformation

In order to describe the motion of points of a continuum body in time, the time rates of many field variables are of particular interest. To begin with, the time derivative of the displacement vector $\vec{x} = \vec{\chi}(\vec{X}, t)$ is taken into consideration, which leads to the velocity vector

$$\vec{v}(\vec{X}, t) = \lim_{\Delta t \to 0} \frac{\Delta \vec{x}}{\Delta t} = \frac{\mathrm{d}\vec{x}}{\mathrm{d}t} = \frac{\partial}{\partial t}\vec{\chi}(\vec{X}, t) = \dot{\vec{x}}(\vec{X}, t)\,. \tag{2.17}$$

Here, the vector $\vec{v}(\vec{X}, t)$ is a spatial vector in the current configuration, yet formulated in terms of reference coordinates $\vec{X}$. It is identical with the velocity vector expressed in spatial coordinates $\vec{v}(\vec{x}, t)$.

　To compute the time derivative of an arbitrary vector-valued physical quantity $\vec{a}$, the so-called *material time derivative* is introduced. In Lagrangian description, this derivative is straightforward. It reads

$$\frac{\mathrm{D}}{\mathrm{D}t}\vec{a}(\vec{X}, t) = \frac{\partial}{\partial t}\vec{a}(\vec{X}, t) = \dot{\vec{a}}(\vec{X}, t)\,. \tag{2.18}$$

In spatial coordinates, the material derivative can be calculated with the aid of the chain rule, leading to

$$\frac{\mathrm{D}}{\mathrm{D}t}\vec{a}(\vec{x}, t) = \frac{\partial}{\partial t}\vec{a}(\vec{\chi}(\vec{X}, t), t) = \frac{\partial \vec{a}}{\partial \vec{x}}\frac{\partial \vec{\chi}(\vec{X}, t)}{\partial t} + \frac{\partial \vec{a}}{\partial t} = \text{grad } \vec{a}(\vec{x}, t)\,\vec{v} + \frac{\partial \vec{a}}{\partial t}\,. \tag{2.19}$$

Here, the material derivative consists of two parts, where the first part is known as the *convective derivative* and the second part as the *local derivative*. As a physical interpretation, the local part describes the change in time for an observer with a fixed position, whereas the convective part incorporates the change in time if the observer's position is not fixed. The absence of the convective part in the Lagrangian description might have some advantages and can simplify the numerical solution process. More specifically, this part does not fulfil the material

objectivity requirement, which is why it is not suitable for further numerical treatment with finite elements [84].

To find the rate of deformation, the time derivative of the deformation gradient is considered

$$\dot{\underset{\sim}{F}} = \frac{\partial \dot{\vec{x}}}{\partial \vec{X}} = \text{Grad}\,\vec{v}(\vec{x},t) = \frac{\partial \vec{v}}{\partial \vec{x}}\frac{\partial \vec{x}}{\partial \vec{X}} = \text{grad}\,\vec{v}(\vec{x},t)\,\underset{\sim}{F}\,, \tag{2.20}$$

in which $\underset{\sim}{l} = \text{grad}\,\vec{v}$ is known as the *spatial velocity gradient tensor*. This measure can be decomposed in a symmetric and an anti-symmetric part

$$\underset{\sim}{l} = \text{grad}\,\vec{v}(\vec{x},t) = \dot{\underset{\sim}{F}}\underset{\sim}{F}^{-1} = \frac{1}{2}\left(\underset{\sim}{l}+\underset{\sim}{l}^T\right) + \frac{1}{2}\left(\underset{\sim}{l}-\underset{\sim}{l}^T\right) = \underset{\sim}{d} + \underset{\sim}{w}\,, \tag{2.21}$$

where $\underset{\sim}{d}$ is the *deformation rate tensor* and $\underset{\sim}{w}$ the so-called *spin-tensor*. Moreover, the rate of strain measures is of interest. With the rate of the right Cauchy-Green tensor $\dot{C} = \dot{\underset{\sim}{F}}^T\underset{\sim}{F} + \underset{\sim}{F}^T\dot{\underset{\sim}{F}}$, the rate of the Green-Lagrange strain tensor reads [13]

$$\dot{\underset{\sim}{E}} = \frac{1}{2}\left(\dot{\underset{\sim}{F}}^T\underset{\sim}{F} + \underset{\sim}{F}^T\dot{\underset{\sim}{F}}\right) = \frac{1}{2}\underset{\sim}{F}^T\left(\underset{\sim}{l}^T+\underset{\sim}{l}\right)\underset{\sim}{F} = \underset{\sim}{F}^T\underset{\sim}{d}\,\underset{\sim}{F}\,, \tag{2.22}$$

leading to an expression for the deformation rate tensor:

$$\underset{\sim}{d} = \underset{\sim}{F}^{-T}\dot{\underset{\sim}{E}}\underset{\sim}{F}^{-1}\,. \tag{2.23}$$

Another useful quantity is the material derivative of the determinant of the deformation gradient. According to [13], this can be expressed as

$$\dot{J} = J\,\text{tr}\,\underset{\sim}{d} = \text{div}\,\vec{v} = J\,\text{tr}\,(\underset{\sim}{C}^{-1}\,\dot{\underset{\sim}{E}}) = J\,\underset{\sim}{C}^{-1}\cdot\dot{\underset{\sim}{E}} = \frac{J}{2}\underset{\sim}{C}^{-1}\cdot\dot{\underset{\sim}{C}}\,. \tag{2.24}$$

Further, this relation can be used to obtain the rate of volume change by means of

$$\text{d}\dot{v} = \text{div}\,\vec{v}\,\text{d}v = \dot{J}\,\text{d}V\,. \tag{2.25}$$

2.1.1.4 Mass density and equation of continuity

The conventional theory of continuum mechanics postulates that the mass of a body $\mathcal{B}$ is constant over time. In order to make a statement about the mass of a body, the mass density $\rho(\vec{x},t)$ is introduced to the current configuration, relating a current infinitesimal volume element $\text{d}v$ to its infinitesimal mass element $\text{d}m$ by

$$\rho(\vec{x},t) = \lim_{\Delta v\to 0}\frac{\Delta m}{\Delta v} = \frac{\text{d}m}{\text{d}v}\,. \tag{2.26}$$

Since the mass is assumed to be constant, leading to

$$m = \int_v \rho(\vec{x},t)\,\text{d}v = \int_V \rho_0(\vec{X})\,\text{d}V\,, \tag{2.27}$$

the mass density in the reference configuration can be related to current configuration by using $\mathrm{d}v = \det \underset{\sim}{F}\, \mathrm{d}V$ to

$$\mathrm{d}m(\vec{x}, t) = \rho(\vec{x}, t)\, \mathrm{d}v = \rho_0(\vec{X})\, \mathrm{d}V \quad \rightarrow \quad \rho_0(\vec{X}) = \det \underset{\sim}{F}\, \rho(\vec{x}, t)\,. \tag{2.28}$$

Mass conservation implies that $\dot{m} = 0$ holds. Thus, the local form of the *equation of continuity* in spatial description can be derived using Eq. (2.25) to

$$\frac{\mathrm{D}}{\mathrm{D}t} m(\vec{x}, t) = \dot{\rho}\, \mathrm{d}v + \rho\, \mathrm{d}\dot{v} = \dot{\rho}\, \mathrm{d}v + \rho \operatorname{div} \vec{v}\, \mathrm{d}v = 0 \quad \rightarrow \quad \dot{\rho} + \rho \operatorname{div} \vec{v} = 0\,. \tag{2.29}$$

In material description, one can conclude that the density $\rho_0(\vec{X}) > 0$ is independent of time and thus $\dot{\rho}_0 = 0$.

2.1.2 Force, stress and balance equations

Following the description of the kinematic relations, the concept of stress and the conservation laws of continuum mechanics are introduced.

2.1.2.1 *Forces and stress tensor*

Applying forces to a body $\mathcal{B}$ leads to stress – which can be seen as the intensity of a force. Stress relates the applied force to a surface element and is responsible for the deformation of the body. To begin with, the two most common forces – the body force and the surface force – are introduced. The body forces (gravity forces, for instance) act on the volume of the body, while surface stress results from applied forces acting on a certain surface element of the boundary. Well-known candidates are external loads, contact, friction or reaction forces.
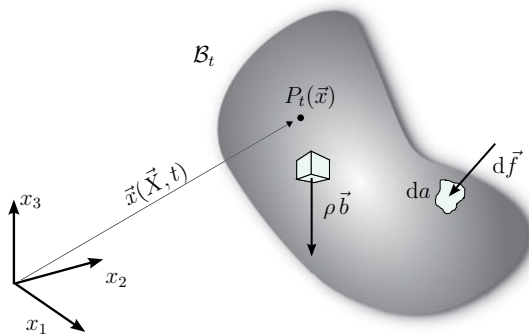
Figure 2.2: Surface and volume loads acting on a body in the current configuration.

To describe a stress state, it is possible to employ the force vector $\vec{f}$ acting in the current configuration on an arbitrary surface element a of the bodies' boundary,

as depicted in Figure 2.2. In the infinitesimal case, there follows

$$\vec{t} = \lim_{\Delta a \to 0} \frac{\Delta \vec{f}}{\Delta a} = \frac{\mathrm{d}\vec{f}}{\mathrm{d}a}, \tag{2.30}$$

where $\vec{t}$ is known as *surface traction vector* or simply *stress vector*. This relation is also valid if internal forces are considered, where it is used to describe the internal stress state inside the body on an arbitrary surface. *Cauchy's theorem* postulates that

$$\vec{t} = \underset{\sim}{\sigma} \vec{n} \tag{2.31}$$

holds in the spatial description – with $\underset{\sim}{\sigma}$ being the symmetric[3] *Cauchy stress tensor*[4] and $\vec{n}$ being the unit normal vector to the spatial surface element $\mathrm{d}a = \mathrm{d}\vec{a} \cdot \vec{n}$. The traction vector depends upon the orientation of the surface element $\vec{t} = \vec{t}(\vec{n})$ and has to fulfil *Newton's third law of action and reaction* $\vec{t}(\vec{n}) = -\vec{t}(-\vec{n})$, as illustrated in Figure 2.3.
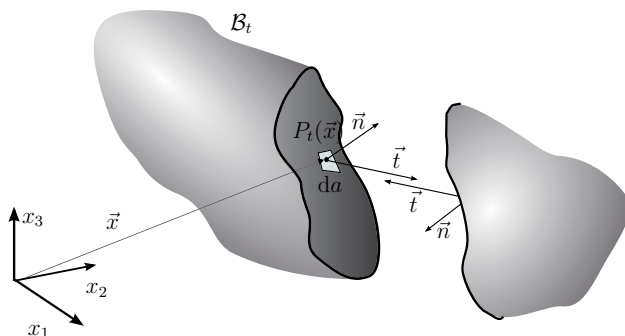
Figure 2.3: Stress state of a body in the current configuration.

If the material surface element of the reference configuration $\mathrm{d}\vec{A} = \vec{N}\mathrm{d}A$ is related to a force acting in the current configuration $\mathrm{d}\vec{f}$ by using Eq. (2.4), the so-called *first Piola-Kirchhoff stress tensor* $\underset{\sim}{P}$ is obtained:

$$\mathrm{d}\vec{f} = J \underset{\sim}{\sigma} \underset{\sim}{F}^{-T} \mathrm{d}\vec{A} = \underset{\sim}{P} \, \mathrm{d}\vec{A}. \tag{2.32}$$

The stress tensor $\underset{\sim}{P}$ has several special characteristics [13]: Generally, it is not symmetric $\underset{\sim}{P} \neq \underset{\sim}{P}^{T}$, i.e. it has nine independent components. Moreover, it is a so called *two-point tensor* because it relates the current force in the deformed configuration to the undeformed surface element of the reference configuration.

---

[3]In Section 2.1.2.2 , it is shown that $\underset{\sim}{\sigma} = \underset{\sim}{\sigma}^T$ holds.

[4]Cauchy stresses are also known as *true stresses* as they deliver the current stress state.

In order to find a stress tensor that is completely related to the reference configuration, the infinitesimal force vector $\mathrm{d}\vec{f}$ is transformed into material coordinates by $\mathrm{d}\vec{F} = \underset{\sim}{F}^{-1}\mathrm{d}\vec{f}$. Relating this vector to the reference surface element $\mathrm{d}A$ leads to the *second Piola-Kirchhoff stress tensor* $\underset{\sim}{S}$

$$\mathrm{d}\vec{F} = J\,\underset{\sim}{F}^{-1}\underset{\sim}{\sigma}\,\underset{\sim}{F}^{-T}\mathrm{d}\vec{A} = \underset{\sim}{F}^{-1}\underset{\sim}{P}\,\mathrm{d}\vec{A} = \underset{\sim}{S}\,\mathrm{d}\vec{A}. \tag{2.33}$$

The second Piola-Kirchhoff stress tensor is a symmetric stress tensor in material coordinates of the reference configuration. Due to the fact that it correlates material forces to material surface elements, there is no specific physical interpretation. Nevertheless, this stress tensor is used to describe the nonlinear mechanical field, since a Lagrangian formulation of the coupled problem will be used in the subsequent sections.

2.1.2.2   Equations of motion

In the following, two fundamental equations are introduced to constitute the basic laws of motion in continuum mechanics. The first law postulates the balance momentum of the body $\mathcal{B}_t$. To begin with, all equations are formulated in spatial description. The first law is also known as the *conservation of linear momentum* in the global form, relating the sum of all forces $\vec{f}_{\mathcal{B}_t}(t)$ – surface and volume forces – acting on $\mathcal{B}_t$ to the temporal change in translational momentum $\vec{l}_{\mathcal{B}_t}(t)$. The momentum is defined by the product of the mass density and the velocity, see [84] for instance, and one can deduce for equilibrium that

$$\frac{\mathrm{D}}{\mathrm{D}t}\,\vec{l}_{\mathcal{B}_t}(t) = \int\limits_{\Omega_t} \frac{\mathrm{D}}{\mathrm{D}t}\,\rho\,\vec{v}\,\mathrm{d}v = \int\limits_{\Omega_t} \rho\,\dot{\vec{v}}\,\mathrm{d}v = \vec{f}_{\mathcal{B}_t}(t)\,, \tag{2.34}$$

holds, with the sum of all forces:

$$\vec{f}_{\mathcal{B}_t}(t) = \int\limits_{\partial\Omega_t} \vec{t}\,\mathrm{d}a + \int\limits_{\Omega_t} \rho\,\vec{b}\,\mathrm{d}v = \int\limits_{\Omega_t} \mathrm{div}\,\underset{\sim}{\sigma}\,\mathrm{d}v + \int\limits_{\Omega_t} \rho\,\vec{b}\,\mathrm{d}v \tag{2.35}$$

From this equation, the local form of linear momentum (known as *Cauchy's first law of motion*) can easily be derived to

$$\rho\,\dot{\vec{v}} = \mathrm{div}\,\underset{\sim}{\sigma} + \rho\,\vec{b}\,, \tag{2.36}$$

where $\vec{b}$ represents the body force vector per unit mass. It is also possible to find a material description in which the translational momentum is formulated with respect to the reference configuration. In that case, the local form of balance of linear momentum reads [84]

$$\rho_0\,\dot{\vec{v}} = \mathrm{Div}\,\underset{\sim}{P} + \rho_0\,\vec{b} = \mathrm{Div}\,\underset{\sim}{F}\underset{\sim}{S} + \rho_0\,\vec{b}\,, \tag{2.37}$$

which can either be written in terms of the first or the second Piola-Kirchhoff stress tensor. In the stationary case – for quasi-static processes where mass and

inertia effects can be neglected – the assumption that for the acceleration $\dot{\vec{v}} \approx \vec{0}$ holds, leads to the quasi-static balance of linear momentum

$$\vec{0} = \text{Div} \, \underset{\sim}{P} + \rho_0 \, \vec{b} = \text{Div} \, \underset{\sim}{F} \underset{\sim}{S} + \rho_0 \, \vec{b} \,. \tag{2.38}$$

The second law of motion considers the rotational equilibrium and takes the temporal change of the angular momentum into account. This leads to the conservation of moment of momentum. It postulates that the temporal change in the angular momentum $\vec{h}_B(t)$ with respect to a given position $\vec{x}_0$ is equivalent to the moments $\vec{m}_{B_t}(t)$ acting on the body B_t

$$\frac{\text{D}}{\text{D}t} \, \vec{h}_{B_t}(t) = \frac{\text{D}}{\text{D}t} \int\limits_{\Omega_t} \vec{r} \times (\rho \, \vec{v}) \, \text{d}v = \vec{m}_{B_t}(t) \,. \tag{2.39}$$

Therein, $\vec{r} = \vec{x}(\vec{X}, t) - \vec{x}_0$ can be viewed as the position vector of the force. Further, the moment is furnished by

$$\vec{m}_{B_t}(t) = \int\limits_{\partial\Omega_t} \vec{r} \times \vec{t} \, \text{d}a + \int\limits_{\Omega_t} \vec{r} \times (\rho \, \vec{b}) \, \text{d}v \,. \tag{2.40}$$

In the literature, this law is often referred to as *Cauchy's second equation of motion*. It can be shown, see [84] for instance, that the local form of the balance of angular momentum leads to the requirement of a symmetric stress tensor $\underset{\sim}{\sigma} = \underset{\sim}{\sigma}^T$ or $\underset{\sim}{S} = \underset{\sim}{S}^T$.

2.1.2.3 Balance of energy and laws of thermodynamics

In order to model multi-physical problems that involve changes in the temperature, a thermodynamically consistent framework is required. To this end, the balance of energy and the entropy inequality of the thermodynamic continuum body shall be outlined briefly. To begin with, the first law of thermodynamics is introduced. This law postulates the conservation of energy and that energy can be transformed from one type to another. In other words, the balance of energy applied to a body B_t states that the temporal change of the total energy equals the external mechanical work plus the thermal work. According to [77], this can be expressed by

$$\frac{\text{D}}{\text{D}t} \, \mathcal{K}_{B_t}(t) + \frac{\text{D}}{\text{D}t} \, \mathcal{E}_{B_t}(t) = \mathcal{W}_{\text{ext}}(t) + \mathcal{Q}(t) \,, \tag{2.41}$$

where the first term accounts for the kinetic energy $\mathcal{K}_{B_t}(t)$, which reads

$$\mathcal{K}_{B_t}(t) = \int\limits_{\Omega_t} \frac{1}{2} \rho \, \vec{v} \cdot \vec{v} \, \text{d}v \quad \text{and} \quad \frac{\text{D}}{\text{D}t} \, \mathcal{K}_{B_t}(t) = \int\limits_{\Omega_t} \rho \, \vec{v} \cdot \dot{\vec{v}} \, \text{d}v \,. \tag{2.42}$$

Together with the second term on the left-hand side of Eq. (2.41), the rate of the internal energy $\mathcal{E}$ – the sum of thermal and elastic strain-energy – given by

$$\frac{\mathrm{D}}{\mathrm{D}t}\,\mathcal{E}_{\mathcal{B}_t}(t) = \int_{\Omega_t} \rho\,\dot{e}\,\mathrm{d}v\,, \tag{2.43}$$

represents the total energy of a continuum body [84]. The quantity $\dot{e}$ is the rate of internal energy for a unit control-volume. Further, according to [77] for instance, the external mechanical work can be written as

$$\mathcal{W}_{\mathrm{ext}}(t) = \int_{\partial\Omega_t} \vec{v}\cdot\vec{t}\,\mathrm{d}a + \int_{\Omega_t} \rho\,\vec{v}\cdot\vec{b}\,\mathrm{d}v = \int_{\Omega_t} \rho\,\vec{v}\cdot\dot{\vec{v}} + \underset{\sim}{\sigma}\cdot\underset{\sim}{\mathrm{d}}\,\mathrm{d}v\,. \tag{2.44}$$

The second part on the right-hand side of the energy balance (2.41) is related to the thermal work and reads

$$\mathcal{Q}(t) = \int_{\partial\Omega_t} -\vec{q}\cdot\vec{n}\,\mathrm{d}s + \int_{\Omega_t} \rho\,r\,\mathrm{d}v = \int_{\Omega_t} (\rho\,r - \operatorname{div}\vec{q})\,\mathrm{d}v\,. \tag{2.45}$$

Using these definitions, the local form of the balance of energy in spatial description (which holds for any arbitrary volume) element can be expressed as

$$\rho\,\dot{e} = \underset{\sim}{\sigma}\cdot\underset{\sim}{\mathrm{d}} - \operatorname{div}\vec{q} + \rho\,r\,. \tag{2.46}$$

Therein, $\vec{q}$ is the Cauchy heat flux vector and the scalar r denotes an external heat source. In material description, the first law of thermodynamics reads

$$\rho_0\,\dot{e} = \underset{\sim}{\mathrm{P}}\cdot\dot{\underset{\sim}{\mathrm{F}}} - \operatorname{Div}\vec{Q} + \rho_0\,r = \underset{\sim}{\mathrm{S}}\cdot\dot{\underset{\sim}{\mathrm{E}}} - \operatorname{Div}\vec{Q} + \rho_0\,r\,. \tag{2.47}$$

Next, the *second law of thermodynamics* is introduced. This law allows to make statements about the direction of energy transfer. It is related to the *entropy inequality concept* which will also be outlined briefly. To begin with, another fundamental physical quantity conjugated to the temperature is introduced. In order to describe the thermodynamic state of the continuum body $\mathcal{B}_t$, the *entropy*

$$\Sigma(t) = \int_{\Omega_t} \rho\,\xi(\vec{x},t)\,\mathrm{d}v \tag{2.48}$$

is introduced, where ξ is referred to as the *specific entropy per unit mass*. The second law of thermodynamics postulates that the rate of the entropy is greater or equal to the rate of externally induced entropy. In mathematical formulation, this postulate reads, see e.g. [8],

$$\dot{\Sigma}(t) = \frac{\mathrm{D}}{\mathrm{D}t}\int_{\Omega_t} \xi\,\rho\,\mathrm{d}v \geq \int_{\Omega_t} \frac{r}{\Theta}\,\rho\,\mathrm{d}v - \int_{\partial\Omega_t} \frac{1}{\Theta}\,\vec{q}\cdot\vec{n}\,\mathrm{d}a \tag{2.49}$$

which can be rearranged by means of the divergence theorem to

$$\int\limits_{\Omega_t} \left(\rho \, \dot{\xi} - \rho \, \frac{r}{\Theta} + \operatorname{div} \frac{\vec{q}}{\Theta} \right) \mathrm{d}v \geq 0 \, . \tag{2.50}$$

Since this must also hold in local form for an arbitrary volume element, one can deduce that

$$\rho \, \dot{\xi} - \rho \, \frac{r}{\Theta} + \operatorname{div} \frac{\vec{q}}{\Theta} \geq 0 \, . \tag{2.51}$$

For all reversible processes, Eq. (2.51) is zero. The other case is an irreversible process – for which the inequality (2.51) is greater zero. This equation can be recast using the energy equation given by Eq. (2.46), which eventually leads to

$$\underset{\sim}{\sigma} \cdot \underset{\sim}{\mathrm{d}} - \rho \left(\dot{e} - \Theta \, \dot{\xi} \right) - \frac{1}{\Theta} \vec{q} \cdot \operatorname{grad} \Theta \geq 0 \, . \tag{2.52}$$

Introducing the thermodynamic function

$$\Psi = e - \Theta \, \xi \quad \text{with} \quad \dot{\Psi} = \dot{e} - \dot{\Theta} \xi - \Theta \dot{\xi} \tag{2.53}$$

which is known as the *Helmholtz free-energy*, and following [77], the so-called *Clausius-Duhem inequality* can finally be obtained

$$\underbrace{\underset{\sim}{\sigma} \cdot \underset{\sim}{\mathrm{d}} - \rho \left(\dot{\Psi} + \dot{\Theta} \, \xi \right)}_{d_{\mathrm{int}}} - \underbrace{\frac{1}{\Theta} \vec{q} \cdot \operatorname{grad} \Theta}_{d_{\Theta}} \geq 0 \, . \tag{2.54}$$

What can be concluded from this fundamental inequality is that the dissipation $d = d_{\mathrm{int}} + d_{\Theta} \geq 0$ is non-negative. This must hold for the internal dissipation $d_{\mathrm{int}} \geq 0$ as well as for the thermal dissipation $d_{\Theta} \geq 0$ [84]. From the latter part, a fundamental physical observation can be derived: Since $-1/\Theta \, \vec{q} \cdot \operatorname{grad} \Theta \geq 0$ must hold, there is a heat flow from the warmer to the colder region because the absolute temperature given in Kelvin cannot be negative $\Theta \geq 0$. This consequently means that heat flows against the temperature gradient, which leads to an important restriction of the heat flux vector. For a detailed introduction into the concepts of continuum thermodynamics, the interested reader is referred to the textbooks of [77, 169, 159]. Indeed, the Clausius-Duhem inequality can also be formulated in material description resulting in

$$\underset{\sim}{S} \cdot \underset{\sim}{\dot{E}} - \rho_0 \left(\dot{\Psi} + \dot{\Theta} \, \xi \right) - \frac{1}{\Theta} \vec{Q} \cdot \operatorname{Grad} \Theta \geq 0 \, . \tag{2.55}$$

2.1.3 Constitutive relations and material models

Subsequent to the description of the kinematic relations and the balance equations, there is still a lack of information to determine the deformation or temperature of the body $\mathcal{B}$. This is due to the fact that the mathematical problem is indeterminate, meaning that the number of unknowns is higher than the available

number of equations. Obviously, information regarding the material behavior is missing. To close this gap, constitutive relations need to be formulated to account for the dependencies between the stress tensor and the deformation or between the temperature and the heat flux, for instance. These relations characterize the material and form the remaining equations in such a way that the mathematical problem becomes determinate.

2.1.3.1 Mechanical behavior of materials

To begin with, the mechanical behavior of materials under isothermal conditions is described. To this end, the so-called *strain-energy density function W* defined as

$$W = \rho_0 \Psi(\underset{\sim}{E}) \tag{2.56}$$

is introduced, where Ψ denotes the *Helmholtz free-energy* function. The explanations in this section are limited to hyper-elastic materials. Inserting $W = \rho_0 \Psi(\underset{\sim}{E})$ in the Clausius-Duhem inequality, see Eq. (2.55), the relation

$$\underset{\sim}{S} = \frac{\partial W}{\partial \underset{\sim}{E}} = 2 \frac{\partial W}{\partial \underset{\sim}{C}} \tag{2.57}$$

is achieved for the second Piola-Kirchhoff stress tensor, see also [84, 77] for instance, and further

$$\underset{\sim}{C} = 2 \frac{\partial \underset{\sim}{S}}{\partial \underset{\sim}{C}} = 4 \frac{\partial^2 W}{\partial \underset{\sim}{C} \partial \underset{\sim}{C}} \tag{2.58}$$

which leads to the fourth-order material tensor $\underset{\sim}{C}$. The strain-energy function W stores the information about how stress in a material is related to the deformation. There are several different expressions for W, of which only two selected examples will be considered here. The very simplest elastic material model is the *Saint Venant-Kirchhoff* model, represented by the following function

$$W(\underset{\sim}{E}) = \frac{\lambda}{2} \left(\mathrm{tr}\, \underset{\sim}{E} \right)^2 + \mu\, \underset{\sim}{E} \cdot \underset{\sim}{E}\,. \tag{2.59}$$

Therein, λ and μ are material constants known as *Lamé* parameters. Applying Eq. (2.57) and (2.58) to this function delivers the stress and elasticity tensor

$$\underset{\sim}{S} = \lambda\, \mathrm{tr}\, \underset{\sim}{E}\, \underset{\sim}{I} + 2\,\mu\, \underset{\sim}{E}\,, \tag{2.60}$$

$$\underset{\sim}{C} = \lambda\, \underset{\sim}{I} \otimes \underset{\sim}{I} + 2\,\mu\, \underset{\sim}{\mathcal{I}}\,. \tag{2.61}$$

In the latter equation, the tensor $\underset{\sim}{\mathcal{I}}$ is referred to as the first *fourth-order fundamental tensor*[5].

The Saint Venant-Kirchhoff material model is often used to describe the mechanical behavior of metals. However, it has some drawbacks. In particular,

---

[5]The first fourth-order fundamental tensor is also known as the fourth-order identity tensor since $\underset{\sim}{\mathcal{I}}\underset{\sim}{E} = \underset{\sim}{E}$.

large compressive deformations can lead to material instabilities and physically wrong stress states. Several material models to avoid these effects can be found in the literature. For a (weak) compressible hyper-elastic material, it is often assumed that the strain-energy function W is additively split in an isochoric $\bar{w}$ and a volumetric part $U(J)$:

$$W(\bar{\underset{\sim}{C}}, J) = \bar{w}(\bar{\underset{\sim}{C}}) + U(J) \tag{2.62}$$

For the isochoric part, a classical polynomial expression can be chosen [72]

$$\bar{w}(\bar{\underset{\sim}{C}}) = \sum_{i,j=0}^{n} c_{ij}(\mathrm{I}_{\bar{\underset{\sim}{C}}} - 3)^i (\mathrm{II}_{\bar{\underset{\sim}{C}}} - 3)^j \,. \tag{2.63}$$

Therein, $\mathrm{I}_{\bar{\underset{\sim}{C}}}$ and $\mathrm{II}_{\bar{\underset{\sim}{C}}}$ are the first and second invariant with respect to the isochoric part of the right Cauchy-Green tensor, as introduced in Eq. (2.15). The parameter c_{ij} denotes material constants, and choosing $i = 1$ and $j = 0$[6] leads to a compressible *Neo-Hookean*[7] type material

$$W(\bar{\underset{\sim}{C}}, J) = c_{10} \left(\mathrm{tr}\, \bar{\underset{\sim}{C}} - 3 \right) + U(J) \,. \tag{2.64}$$

For more details and for a comprehensive introduction into modeling the mechanical behavior of materials, the textbooks of [77, 172, 5] are recommended. A detailed survey focusing on different types of strain-energy functions for hyperelastic materials can be found in [72], for instance.

2.1.3.2 *Thermodynamic behavior of materials*

With regard to the mechanical behavior of materials, a constitutive relation for the heat flux and the entropy is needed. The heat flux outward through the surface $\mathrm{d}a$ of a control-volume is given as $q = \vec{q} \cdot \vec{n}$, with $\vec{q}$ being the heat flux vector. In order to relate the heat flux to the temperature, *Fourier's law* of heat conduction is introduced

$$\vec{q} = -\underset{\sim}{\lambda}_{\Theta}\, \mathrm{grad}\,\Theta(\vec{x}, t) \,. \tag{2.65}$$

This guarantees a heat flux in the direction from the warmer to the colder region. The tensor $\underset{\sim}{\lambda}_{\Theta}$ is the heat conductivity tensor that takes the form $\underset{\sim}{\lambda}_{\Theta} = \lambda_{\Theta}\underset{\sim}{I}$ for isotropic heat conduction. The material parameter λ_{Θ} is denoted as the heat conduction coefficient. The expression for the heat flux can be transformed into the reference configuration by using $\mathrm{grad}\,\Theta = \underset{\sim}{F}^{-T}\,\mathrm{Grad}\,\Theta$ and *Nanson's formula* $\vec{n}\,\mathrm{d}a = J\,\underset{\sim}{F}^{-T} \cdot \vec{N}\,\mathrm{d}A$ such that

$$\int_a \vec{q} \cdot \vec{n}\,\mathrm{d}a = \int_A -J\left(\underset{\sim}{F}^{-1}\,\underset{\sim}{\lambda}_{\Theta}\,\underset{\sim}{F}^{-T}\,\mathrm{Grad}\,\Theta\right) \cdot \vec{N}\,\mathrm{d}A = \int_A -\underset{\sim}{\Lambda}_{\Theta}\,\mathrm{Grad}\,\Theta \cdot \vec{N}\,\mathrm{d}A \,. \tag{2.66}$$

---

[6] With $i = 1$ and $j = 0$, one obtains c_{10} – which can be related to the shear modulus by $\mu = 2\,c_{10}$.

[7] The classical Neo-Hooke material holds for incompressible materials with $W(\underset{\sim}{C}) = c_{10}\left(\mathrm{tr}\,\underset{\sim}{C} - 3\right)$, see [72] p. 70.

Thus, the heat flux vector in material description reads

$$\vec{Q} = -\underset{\sim}{\Lambda}_{\Theta} \operatorname{Grad} \Theta(\vec{x}, t) \,, \tag{2.67}$$

where $\underset{\sim}{\Lambda}_{\Theta}$ is the material conductivity tensor related to the reference configuration.

Another physical quantity for which a constitutive relation is required is the entropy. It is assumed that the Helmholtz free-energy function $\Psi = \Psi(\underset{\sim}{E}, \Theta)$ is given in terms of the strain tensor and the temperature. Inserting the time derivative of the Helmholtz free-energy function into the first part of the Clausius-Duhem inequatlity, see Eq. (2.55), leads to

$$\underset{\sim}{S} \cdot \dot{\underset{\sim}{E}} - \rho_0 \left(\dot{\Psi} + \dot{\Theta} \xi \right) = \underset{\sim}{S} \cdot \dot{\underset{\sim}{E}} - \rho_0 \left(\frac{\partial \Psi}{\partial \underset{\sim}{E}} \cdot \dot{\underset{\sim}{E}} + \frac{\partial \Psi}{\partial \Theta} \dot{\Theta} + \dot{\Theta} \xi \right) \geq 0 \,. \tag{2.68}$$

Thus, it can be concluded from this inequality that

$$\xi = -\frac{\partial \Psi}{\partial \Theta} \,. \tag{2.69}$$

As a conclusion, relations (2.69) and (2.65) define the constitutive laws for the temperature and for the entropy. They are mandatory ingredients for the derivation of the governing equations of the thermal field discussed in Section 2.2.2.

2.2 Governing equations for electro-thermo-mechanical coupling

This section serves to introduce the governing equations for electro-thermo-mechanically coupled problems. This leads to a coupled initial boundary value problem of a system of partial differential equations. The descriptive equation for the mechanical field is the balance of linear momentum, whereas the equations for the thermal and the electric field have not yet been introduced in detail. The *equation of heat conduction* is used for the thermal field, and the electric field is described by the *conservation of electric charge* in the stationary case. In addition, the relevant constitutive relations are recapitulated, and the corresponding boundary conditions are defined to obtain a mathematical determination of the coupled problem.

2.2.1 Mechanical field

The governing equations for the mechanical field are the balance of linear momentum, see Eq. (2.38), and the constitutive model for the stress tensor (elasticity), see Eq. (2.57). In this work, it is assumed that the electric field does not cause any deformations or stresses. Further, the Helmholtz free-energy is a function of deformation-dependent variables and the temperature. For instance, if

$\Psi = \Psi(\underset{\sim}{E}, \Theta)$ is chosen, the stress tensor reads

$$\underset{\sim}{S}(\underset{\sim}{E}, \Theta) = \rho_0 \frac{\partial \Psi}{\partial \underset{\sim}{E}} \, . \tag{2.70}$$

In order to complete the mathematical description of the mechanical field, the boundary surface $\partial \Omega_0$ of the considered body is decomposed to

$$\partial \Omega_0 = \partial \Omega_0^u \cup \partial \Omega_0^\sigma \quad \text{with} \quad \partial \Omega_0^u \cap \partial \Omega_0^\sigma = \emptyset \, . \tag{2.71}$$

To this end, two different types of boundary conditions are introduced. The boundary displacements

$$\vec{u} = \vec{u}(\vec{X}, t) \quad \text{with } \vec{X} \in \partial \Omega_0^u \tag{2.72}$$

which are prescribed on the so-called *Dirichlet boundary* $\partial \Omega_0^u$ and further, the traction vector

$$\underset{\sim}{P} \vec{N} = \vec{\bar{T}}(\vec{X}, t) \quad \text{with } \vec{X} \in \partial \Omega_0^\sigma \tag{2.73}$$

which is prescribed on the so-called *Neumann boundary* $\partial \Omega_0^\sigma$. Together with the balance equation, the nonlinear boundary value problem for the mechanical field reads

$$\vec{0} = \text{Div} \, \underset{\sim}{P} + \rho_0 \, \vec{b} \quad \text{in} \quad \Omega_0 \times [0, T] \tag{2.74}$$

$$\vec{u} = \vec{\bar{u}} \quad \text{on} \quad \partial \Omega_0^u \tag{2.75}$$

$$\underset{\sim}{P} \vec{N} = \vec{\bar{T}} \quad \text{on} \quad \partial \Omega_0^u \tag{2.76}$$

in the reference configuration and

$$\vec{0} = \text{div} \, \underset{\sim}{\sigma} + \rho \, \vec{b} \quad \text{in} \quad \Omega_t \times [0, T] \tag{2.77}$$

$$\vec{u} = \vec{\bar{u}} \quad \text{on} \quad \partial \Omega_t^u \tag{2.78}$$

$$\underset{\sim}{\sigma} \vec{n} = \vec{\bar{t}} \quad \text{on} \quad \partial \Omega_t^u \tag{2.79}$$

in the current configuration.

2.2.2 Thermal field: the heat equation

The primary variable of the thermal field shall be the temperature. Thus, it is desirable to find an equation out of which evolution of the temperature Θ can be calculated promptly. This has some practical reasons. Since the mechanical field is directly coupled to the temperature by thermal stresses due to thermal strains, and since, moreover, some material parameters can be explicitly dependent on Θ, the balance of energy (2.46) might not be the most appropriate formulation to describe the thermal part. However, Eq. (2.46) is used to derive the *heat equation* or the *equation of heat conduction* in which the temperature is the primary variable.

To begin with, the definition of the Helmholtz free-energy is recapitulated, see Eq. (2.53), and the time-derivative of Ψ is calculated to

$$\dot{\Psi} = \dot{e} - \dot{\Theta}\,\xi - \Theta\,\dot{\xi}\,. \tag{2.80}$$

Substituting the rate of the internal energy $\dot{e}$ from the previous equation in Eq. (2.46) leads– after a few mathematical rearrangements – to

$$\underbrace{\underset{\sim}{\sigma} \cdot \mathrm{d} - \rho\left(\dot{\Psi} + \dot{\Theta}\,\xi\right)}_{d_{\mathrm{int}}} - \rho\,\Theta\,\dot{\xi} - \operatorname{div}\vec{q} + \rho\,r = 0\,. \tag{2.81}$$

Therein, d_{int} represents the internal entropy increase due to dissipative mechanical work. For inelastic deformations, this term would be larger than zero – whereas the internal dissipation for pure elastic processes is zero, which is why they are thermodynamic reversible. Rearranging Eq. (2.81) to

$$\rho\,\dot{\xi}\,\Theta = d_{\mathrm{int}} - \operatorname{div}\vec{q} + \rho\,r \tag{2.82}$$

and using the constitutive relation for the entropy, see Eq. (2.69), yields

$$\rho\,\frac{\mathrm{D}}{\mathrm{D}t}\left(-\frac{\partial\Psi}{\partial\Theta}\right)\Theta = d_{\mathrm{int}} - \operatorname{div}\vec{q} + \rho\,r \tag{2.83}$$

in spatial description and

$$\rho_0\,\frac{\partial}{\partial t}\left(-\frac{\partial\Psi}{\partial\Theta}\right)\Theta = D_{\mathrm{int}} - \operatorname{Div}\vec{Q} + \rho_0\,R \tag{2.84}$$

In material description – where D_{int} is the internal dissipation and $\vec{Q}$ the heat flux vector, both related to the reference configuration. In order to complete the heat equation, the derivative of the Helmholtz free-energy function $\Psi(\underset{\sim}{E}, \Theta)$ with respect to the temperature and the time is calculated, leading to

$$\frac{\partial}{\partial t}\left(-\frac{\partial\Psi}{\partial\Theta}\right) = -\frac{\partial^2\Psi}{\partial\Theta^2}\dot{\Theta} - \frac{\partial^2\Psi}{\partial\Theta\partial\underset{\sim}{E}} \cdot \dot{\underset{\sim}{E}}\,. \tag{2.85}$$

Inserting the expression into Eq. (2.84) results in

$$\rho_0\left(-\frac{\partial^2\Psi}{\partial\Theta^2}\dot{\Theta} - \frac{\partial^2\Psi}{\partial\Theta\partial\underset{\sim}{E}} \cdot \dot{\underset{\sim}{E}}\right)\Theta = D_{\mathrm{int}} - \operatorname{Div}\vec{Q} + \rho_0\,R\,. \tag{2.86}$$

This equation can be simplified by introducing the *heat capacity* or *specific heat*[8] [77]

$$c_\Theta(\Theta) := \frac{\partial\xi}{\partial\Theta}\Theta = -\frac{\partial^2\Psi}{\partial\Theta^2}\Theta\,, \tag{2.87}$$

---

[8]The heat capacity is also known as *specific heat at constant deformation*, see [77] for instance. This means that c_Θ can be viewed as the amount of energy required for a unit increase in temperature related to a unit volume at a constant deformation.

such that one finally obtains the heat equation

$$\rho_0 \, c_\Theta(\Theta) \, \dot{\Theta} = \rho_0 \, \Theta \, \frac{\partial^2 \Psi}{\partial \Theta \partial \underline{E}} \cdot \dot{\underline{E}} + D_{\text{int}} - \text{Div} \, \vec{Q} + \rho_0 \, R \, . \tag{2.88}$$

In practical applications, the heat capacity strongly depends on the material under investigation and is often determined by means of experimental data.

The first term in Eq. (2.88) on the right hand side given by

$$\gamma(\Theta, \dot{\underline{E}}) = \rho_0 \, \Theta \, \frac{\partial^2 \Psi}{\partial \Theta \partial \underline{E}} \cdot \dot{\underline{E}} \tag{2.89}$$

is known as the *thermo-elastic coupling term*[9] due to elastic deformations. However, the influence of this effect can be neglected in many cases. Compared to other heat sources, it is usually quite small.

In view of the electro-thermo-mechanical problem, the thermal field described by the heat equation is coupled to the mechanical field through the internal dissipation term, the thermo-elastic coupling effect and geometrical changes which are reflected in the constitutive equation for the heat flux vector. The coupling to the electric field is due to Joule heating that occurs as a heat source r_φ in the heat equation, see the following Section 2.2.3 for more details.

To complete the description of the thermal field, boundary and initial conditions are required. In analogy to the mechanical part, the boundary of the body $\mathcal{B}$ is split into

$$\partial \Omega_0 = \partial \Omega_0^\Theta \cup \partial \Omega_0^q \quad \text{with} \quad \partial \Omega_0^\Theta \cap \partial \Omega_0^q = \emptyset \, . \tag{2.90}$$

The term

$$\Theta = \bar{\Theta}(\vec{X}, t) \quad \text{with} \quad \vec{X} \in \partial \Omega_0^\Theta \tag{2.91}$$

denotes the prescribed temperature on the Dirichlet boundary $\partial \Omega_0^\Theta$, and

$$\vec{Q} \cdot \vec{N} = \bar{Q}(\vec{X}, t) \quad \text{with} \quad \vec{X} \in \partial \Omega_0^q \tag{2.92}$$

is the prescribed heat flux on the Neumann boundary $\partial \Omega_0^q$. Moreover, the initial condition of the thermal field $\Theta(\vec{X}, 0) = \Theta_0$ at $t = 0$ is introduced. In material formulation, the summarized initial boundary value problem for the thermal field reads

$$\rho_0 \, c_\Theta(\Theta) \, \dot{\Theta} = \gamma + D_{\text{int}} - \text{Div} \, \vec{Q} + \rho_0 \, r_\varphi \quad \text{in} \quad \Omega_0 \times [0, T] \tag{2.93}$$

$$\Theta = \bar{\Theta} \quad \text{on} \quad \partial \Omega_0^\Theta \tag{2.94}$$

$$-\vec{Q} \cdot \vec{N} = \bar{Q} \quad \text{on} \quad \partial \Omega_0^q \tag{2.95}$$

$$\Theta = \Theta_0 \quad \text{at} \quad t = 0 \tag{2.96}$$

---

[9]Thermo-elastic coupling is also known as the *Gough-Joule effect*, see [84] p. 326 for instance, which leads to structural heating (typically for rubber) or cooling (typically for metals).

and in the spatial formulation:

$$\rho \, c_\Theta(\Theta) \, \dot{\Theta} = \gamma + d_{\text{int}} - \text{div} \, \vec{q} + \rho \, r_\varphi \quad \text{in} \quad \Omega_t \times [0, T] \tag{2.97}$$

$$\Theta = \bar{\Theta} \quad \text{on} \quad \partial \Omega_t^\Theta \tag{2.98}$$

$$-\vec{q} \cdot \vec{n} = \bar{q} \quad \text{on} \quad \partial \Omega_t^q \tag{2.99}$$

$$\Theta = \Theta_0 \quad \text{at} \quad t = 0 \tag{2.100}$$

The heat flux $\bar{q}$ that enters a control-volume over the boundary $\partial \Omega_t^q$ approaches thermal interactions with the environment and the surfaces of other bodies. In the following, it is assumed that the heat flux can be composed of five parts

$$\bar{q} = q_n + q_r + q_h + q_c + q_{\text{ec}}, \tag{2.101}$$

where q_n denotes the prescribed heat flux, q_r the radiative heat flux, q_h the gap conductive heat flux, q_c the convective heat flux and q_{ec} the heat flux due to an electric current.

A frequently used approximation of radiative heat transfer in thermal analysis reads

$$q_r = \varepsilon \, \sigma_{\text{sb}} \left(\Theta^4 - \Theta_\infty^4 \right) . \tag{2.102}$$

For most applications, this is quite a strong idealization since the formula describes radiation from an infinitely long and wide plate in vacuum [10], see Section 2.3 for more information about thermal radiation.

The gap conductance describes the heat transfer between two surfaces and can be approached by

$$q_h = \lambda_{\text{gc}} \left(\Theta_s - \Theta \right) , \tag{2.103}$$

where λ_{gc} denotes the thermal gap conductance coefficient and Θ_s the temperature of the opposite surface.

Convective heat transfer can be idealized drawing on

$$q_c = h_c(\Theta) \left(\Theta - \Theta_f \right) \tag{2.104}$$

with h_c being the heat convection or film coefficient and Θ_f being the temperature of the surrounding fluid.

Heat transfer between two surfaces that exhibit different electric potential due to an electric current[11] – φ_1 and φ_2 – can be approximated by

$$q_{\text{ec}} = \lambda_{\text{ec}} \left(\varphi_2 - \varphi_1 \right)^2 , \tag{2.105}$$

where λ_{ec} denotes the electric gap conductance coefficient.

---

[10]The radiative heat flux is approximated by $q_r = \varepsilon \, \sigma_{\text{sb}} \, F_{1-2} \left(\Theta^4 - \Theta_\infty^4 \right)$, where F_{1-2} is known as the *view factor*. For two infinitely long plates, that face each other directly, with h representing the distance and w the plate width, this factor reads $F_{1-2} = \sqrt{1 - (h/w)^2} - h/w$, see [121] p. 842 for instance. If $w \approx \infty$ is assumed, one obtains

$$\lim_{w \to \infty} F_{1-2} = 1 \quad \text{and} \quad q_r = \varepsilon \, \sigma_{\text{sb}} \left(\Theta^4 - \Theta_\infty^4 \right) .$$

The concept of view factor radiation is discussed in detail in Section 2.3.2.1.

[11]See QUACH ET AL. [140], p. 260, for more details

2.2.3   Electric field: the charge equation

The electric field in the FAST process is used to generate high heating rates and high temperatures in the material by means of *Joule heating*. In order to describe this process mathematically, it is necessary to formulate the equation of conservation of charge, following from the Maxwell equations. Here, "conservation of charge" indicates that the rate of charge in a control-volume is equal to the electric current density entering this volume across a surface.

The charge of a control-volume reads

$$h = \int\limits_{\Omega_t} \rho_{\mathrm{c}}(\vec{x}, t)\, \mathrm{d}v \,, \tag{2.106}$$

where ρ_{c} denotes the electric charge density. The electric current across a surface can be formulated using Gauss' law to obtain

$$j_{\mathrm{c}} = -\int\limits_{\partial\Omega_t} \vec{j} \cdot \vec{n}\, \mathrm{d}a = -\int\limits_{\Omega_t} \operatorname{div} \vec{j}\, \mathrm{d}v \,. \tag{2.107}$$

Therein, $\vec{j}(\vec{x}, t)$ defines the electric current density.

Similar to the heat flux, a constitutive law for the flow of electric current is needed. This thesis employs *Ohm's law*, which relates the gradient of the electric potential to the current density

$$\vec{j} = -\underset{\sim}{\lambda_\varphi}(\Theta)\, \vec{e} = -\underset{\sim}{\lambda_\varphi}(\Theta)\, \operatorname{grad} \varphi(\vec{x}, t)\,, \tag{2.108}$$

with $\vec{e} = \operatorname{grad} \varphi$ as the electric field intensity, φ as the electric potential and $\underset{\sim}{\lambda_\varphi} = \lambda_\varphi \underset{\sim}{\mathrm{I}}$ representing the isotropic electric conductivity tensor. Here, the electric conductivity coefficient $\lambda_\varphi(\Theta)$ is typically temperature-dependent. This relation can also be transformed into material coordinates in the same manner as the heat flux vector, in such a way that

$$\vec{J} = -J\, \underset{\sim}{\mathrm{F}}^{-1}\, \underset{\sim}{\lambda_\varphi}(\Theta)\, \underset{\sim}{\mathrm{F}}^{-T}\operatorname{Grad} \varphi = -\underset{\sim}{\Lambda_\varphi}(\Theta)\operatorname{Grad} \varphi \tag{2.109}$$

holds. For the global volume, the aspect of conservation of charge requires that

$$\frac{\mathrm{D}}{\mathrm{D}t}\, h = \int\limits_{\Omega_t} \dot{\rho}_{\mathrm{c}}(\vec{x}, t)\, \mathrm{d}v = j_{\mathrm{c}} \quad \rightarrow \quad \int\limits_{\Omega_t} \left(\dot{\rho}_{\mathrm{c}} + \operatorname{div} \vec{j} \right) \mathrm{d}v = 0 \tag{2.110}$$

holds, which must also be valid for the local form at every material point:

$$\dot{\rho}_{\mathrm{c}} + \operatorname{div} \vec{j} = 0 \tag{2.111}$$

Assuming that the rate of charge density is zero, which is a common assumption[12] in the modeling of FAST, see [155, 129, 76] for instance, it is possible to obtain

---

[12]A detailed discussion on this topic can be found in [146], for instance.

the stationary conservation of charge to describe the electric field in the coupled problem

$$\operatorname{div} \vec{j} = 0 \,. \tag{2.112}$$

This equation is formulated in the current configuration. It can also be transformed into material coordinates so that

$$\operatorname{Div} \vec{J} = 0 \,. \tag{2.113}$$

Next, the Joule heating term in the heat equation r_φ shall be discussed. *Joule's law* delivers the rate of electric energy which is dissipated by an electric current flowing through a conducting material as

$$r_\varphi = -\frac{1}{\rho}\, \vec{e} \cdot \vec{j} = \frac{1}{\rho}\, \operatorname{grad} \varphi \cdot \underset{\sim}{\lambda}_\varphi(\Theta)\, \operatorname{grad} \varphi \,. \tag{2.114}$$

With respect to the reference configuration, one can conclude that

$$r_\varphi = -\frac{1}{\rho_0}\, \vec{E} \cdot \vec{J} = \frac{1}{\rho_0}\, \operatorname{Grad} \varphi \cdot \underset{\sim}{\Lambda}_\varphi(\Theta)\, \operatorname{Grad} \varphi \tag{2.115}$$

holds. In the frame of a multi-physical context, the conservation of charge is coupled to the thermal field via the temperature-dependent electric conductivity and to the mechanical field by changes in the configuration expressed by the deformation gradient $\underset{\sim}{F}$.

In order to finalize the mathematical description of the electric field, the surface of the body needs to be described. To this end, the boundary is decomposed into

$$\partial\Omega_0 = \partial\Omega_0^\Theta \cup \partial\Omega_0^j \quad \text{with} \quad \partial\Omega_0^\varphi \cap \partial\Omega_0^q = \emptyset \,, \tag{2.116}$$

where

$$\varphi = \bar{\varphi}(\vec{X}, t) \quad \text{with} \quad \vec{X} \in \partial\Omega_0^\varphi \tag{2.117}$$

denotes the prescribed electric potential on the Dirichlet boundary $\partial\Omega_0^\varphi$. Moreover, the term

$$-\vec{J} \cdot \vec{N} = \bar{J}(\vec{X}, t) \quad \text{with} \quad \vec{X} \in \partial\Omega_0^j \tag{2.118}$$

is the prescribed current density at the Neumann boundary $\partial\Omega_0^j$. The final boundary value problem for the electric field ends up in

$$0 = \operatorname{Div} \vec{J} \quad \text{in} \quad \Omega_0 \times [0, T] \tag{2.119}$$

$$\varphi = \bar{\varphi} \quad \text{on} \quad \partial\Omega_0^\varphi \tag{2.120}$$

$$-\vec{J} \cdot \vec{N} = \bar{J} \quad \text{on} \quad \partial\Omega_0^j \tag{2.121}$$

related to the material description and

$$0 = \operatorname{div} \vec{j} \quad \text{in} \quad \Omega_t \times [0, T] \tag{2.122}$$

$$\varphi = \bar{\varphi} \quad \text{on} \quad \partial \Omega_t^{\varphi} \tag{2.123}$$

$$-\vec{j} \cdot \vec{n} = \bar{j} \quad \text{on} \quad \partial \Omega_t^{j} \tag{2.124}$$

in spatial description. The current density $\bar{j}$ at the boundary $\partial \Omega_t^j$ of a body may have two different contributions

$$\bar{j} = j_n + j_c \,, \tag{2.125}$$

where j_n is the prescribed current density and j_c the electric current between the interface of two bodies. This can be approximated by means of the following linear relation [140]

$$j_c = \lambda_{ec} \left(\varphi_s - \varphi \right) , \tag{2.126}$$

where λ_{ec} denotes the electric gap conduction coefficient and φ_s the electric potential on the surface of the other body.

2.2.4 Summary of the coupled initial boundary value problem

For the sake of clarity, the governing equations from the previous sections which describe the electro-thermo-mechanically coupled problem are summarized in Table 2.1. Further, the coupling effects between the fields are recapitulated. Moreover, the interactions of the fields involved are illustrated in Figure 2.4.

Table 2.1: Overview of the electro-thermo-mechanically coupled problem (Lagrangian formulation).

	Init. bound. value prob.	Constitutive Equations	Field coupling
Mechanical field (Elasticity)	Eq. (2.74) – (2.76)	Stress tensor $\underset{\sim}{S}(\underset{\sim}{E}, \Theta) = \rho_0 \dfrac{\partial \Psi}{\partial \underset{\sim}{E}}$	Thermal field
Thermal field	Eq. (2.93) – (2.96)	Heat flux $\vec{Q}(\underset{\sim}{F}, \Theta) = -\underset{\sim}{\Lambda}_{\Theta} \operatorname{Grad} \Theta(\vec{X}, t)$ Entropy $\xi(\underset{\sim}{F}, \Theta) = -\dfrac{\partial \Psi}{\partial \Theta}$	Mechanical field Electric field
Electric field	Eq. (2.119) – (2.121)	Current density $\vec{J}(\underset{\sim}{F}, \Theta) = -\underset{\sim}{\Lambda}_{\varphi} \operatorname{Grad} \varphi(\vec{X}, t)$	Mechanical field Thermal field
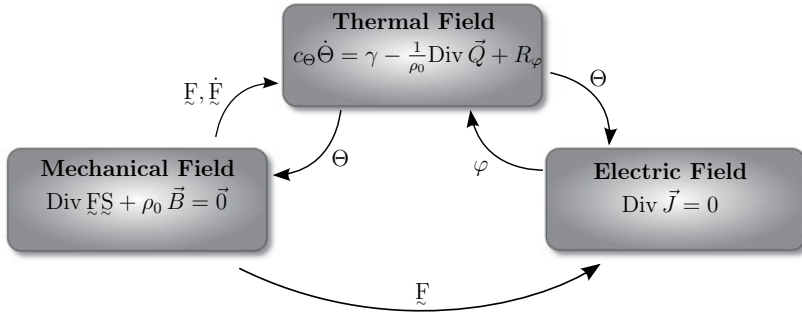
Figure 2.4: Interactions of the mechanical, thermal and electric field.

A statement regarding the coupling strength between the fields is a difficult issue, and it strongly depends on the problem under consideration. For instance, the thermo-elastic coupling effect γ in the heat equation is often neglected, and compared to the internal mechanical dissipation d_{int} in inelastic analysis it is usually small. On the other hand, with respect to the field assisted sintering process, the energy dissipated as heat from d_{int} is small compared to the energy dissipation of the Joule heating term r_φ. That is why the mechanical dissipation term is usually not taken into account in the FAST process [146]. Thus, it can be concluded that different coupling effects may have a strong or weak impact. Basically, this is a problem-dependent issue that cannot be stated universally.

2.3   Thermal radiation

Apart from convection and conduction, radiation is the third mode to transfer thermal energy. The influence of thermal radiation is often neglected or only treated approximately in numerical analysis. With increasing temperature, however, radiative effects become more important and can be the dominating mechanism to transfer thermal energy. This is due to the fact that the heat flux is generally proportional to the fourth power in the temperature

$$q \propto \Theta^4 - \Theta_\infty^4 \,. \tag{2.127}$$

High temperature changes occur in many engineering applications, combustion or manufacturing processes, to name a few, and a realistic simulation of such problems should contain the physics of thermal radiation. Compared to the two other mechanisms to transfer thermal energy – convection and conduction – radiation does not require the presence of a medium. In other words, it is the only mode of heat transfer in vacuum. There are several other distinguishing features that are relevant to compare the modes of heat transfer which are summarized in Table 2.2.

Table 2.2: Comparison of the three main mechanisms to transfer thermal energy

	Radiation	Conduction	Convection
Physical nature of heat transfer	Emitted electro-magnetic waves by temperature	Interactions and collisions of parti-cles	Energy of particles carried away by material carrier
Participating media	Not required	Required	Required
Control-volume	Entire	Infinitesimal	Infinitesimal
Heat flux corre-lation	$q \propto \Theta^4 - \Theta_\infty^4$	$q = -\lambda_\Theta \dfrac{\partial \Theta}{\partial x}$	$q = h_c \left(\Theta - \Theta_f\right)$
Type of balance equation	Differential-integral	Partial differential	Partial differential
Independent variables	Up to 7 (space, time, direction, wavelength)	4 (space and time)	4 (space and time)

An important difference is the fact that conduction and convection are so-called *short-range* phenomena whereas radiation is, in this context, a *long-range* phenomenon [121]. This means that an energy balance for a thermal radiation analysis must be applied to the entire volume, so the complete path through which a ray travels must be taken into account instead. Thus, the mathematical model to describe radiation leads to an integral equation since conduction and convection are described by partial differential equations. These facts make the thermal radiation analysis a complicated and difficult issue[13] with significant differences compared to the "conventional" analysis of conduction and convection. To this end, the basics and fundamentals of radiative heat transfer are briefly outlined in the following section, and the governing equations are introduced. The nature of thermal radiation is well documented in the literature. For a deep introduction into the mathematical formulation, the textbooks of MODEST [121] and HOW-ELL [86] are recommended.

---

[13]In addition, thermal radiation typically depends on three coordinates in space (x, y, z), two angular directions (θ, ψ) and the radiation wavelength $(\varkappa)$. That is why modeling radiation is numerically challenging and computation-intensive.

2.3.1   Basics of thermal radiation

To begin with, the fundamentals of radiative heat transfer are considered, and the essential terms and definitions are established. The phenomenon of thermal radiation is caused by electro-magnetic waves that travel at the speed of light[14] c and electro-magnetic waves are able to propagate through almost any medium. Each wave can be described by its frequency υ, the wave-length $\varkappa$, the wave-number η or the angular frequency f. These properties identify a wave and can be related to each other via the following formula

$$\upsilon = \frac{f}{2\pi} = \frac{c}{\varkappa} = c\eta \ . \tag{2.128}$$

The waves that effectuate thermal radiation are emitted by a medium due to its temperature and typically fall in a range of wavelengths between $0.1\,\mu\mathrm{m}$ and $100\,\mu\mathrm{m}$ [121].

Whenever an electro-magnetic wave of thermal radiation impinges a surface of another medium, one part can be reflected whereas the other part can penetrate into the medium. In a *transparent* medium, all penetrating waves can pass through. In the opposite case, if the waves cannot penetrate through at all, the medium is called *opaque*. A measure for transparency of media is the so-called *optical thickness* or *optical depth*. This measure depends on the material under consideration – metals for instance are nearly always opaque[15]. Surfaces that are opaque and that do not reflect any radiation are denoted as *perfect absorber* or simply as *black*. The described phenomena of a wave striking the surface of another medium can be concluded by the following four fundamental properties of thermal radiation [121]:

$$\text{Reflectance}: \quad \varrho \equiv \frac{\text{reflected part of incoming radiation}}{\text{total incoming radiation}} \tag{2.129}$$

$$\text{Absorptance}: \quad \alpha \equiv \frac{\text{absorbed part of incoming radiation}}{\text{total incoming radiation}} \tag{2.130}$$

$$\text{Transmittance}: \quad \tau \equiv \frac{\text{transmitted part of incoming radiation}}{\text{total incoming radiation}} \tag{2.131}$$

$$\text{Emittance}: \quad \varepsilon \equiv \frac{\text{energy emitted from a surface}}{\text{energy emitted by a black surface}} \tag{2.132}$$

The first three properties are related to the total incoming radiation so that

$$\varrho + \alpha + \tau = 1 \tag{2.133}$$

must hold. Opaque media consequently means $\tau = 0$ so that $\varrho + \alpha = 1$ holds. For gray, diffuse surfaces, one also obtains that $\alpha = \varepsilon$ holds. For black surfaces, one obtains $\alpha = 1$ and therefore $\varrho = \tau = 0$, whereas the emittance[16] takes $\varepsilon = 1$.

---

[14]The speed of light in vacuum is $c_0 = 2.998 \times 10^8\,\mathrm{m/s}$

[15]Apart from the material dependency, the optical thickness also depends on the thickness of the medium itself.

[16]The *emittance* relates the heat flux emitted by a surface to that of a black-body. This is not to be mistaken with the *emissivity*, which is a material property. Both are in the range of $0 < \varepsilon < 1$ and are indicated with same symbol.

2.3.1.1 Emissive power

The emissive power E is the emitted radiative heat flux of a radiating surface. The total emissive power is the integral over the whole frequency spectrum and reads

$$E(\Theta) = \int\limits_0^\infty E_\upsilon(\Theta, \upsilon) \, \mathrm{d}\upsilon . \tag{2.134}$$

By this relation, the total black-body emissive power E_b can be derived. For the sake of clarity, all measures related to a black-body are henceforth denoted with the subscript b. Using Planck's Law [17] for the black-body emissive power spectrum $E_{\mathrm{b},\upsilon}$ an expression for E_b finally leads to

$$E_\mathrm{b}(\Theta) = \int\limits_0^\infty E_{\mathrm{b},\upsilon}(\Theta, \upsilon) \, \mathrm{d}\upsilon = n^2 \sigma_\mathrm{sb} \Theta^4 , \tag{2.135}$$

where $n = 1$ is taken for the *refractive index*[18] and σ_sb the Stefan-Boltzmann constant[19]. Equation Eq. (2.135) emphasizes the nonlinear nature of thermal radiation. For a black surface, the emitted heat flux depends on the temperature to the power of four.

2.3.1.2 Solid angles

In order to describe thermal radiation of a surface that radiates into another medium, any possible direction has to be considered. To this end, the direction vector $\vec{s}$ is introduced, stating that rays can travel through a medium in an infinite number of possible directions. This vector is formulated using spherical or polar coordinates in terms of the so-called *solid angle* denoted by Υ.

A solid angle is basically the three-dimensional counterpart of the standard angle in the two-dimensional plane, as depicted in Figure 2.5. The (infinitesimal) solid angle is defined as

$$\mathrm{d}\Upsilon = \sin\theta \, \mathrm{d}\theta \, \mathrm{d}\psi \quad \rightarrow \quad \Upsilon = \int \int \sin\theta \, \mathrm{d}\theta \, \mathrm{d}\psi . \tag{2.136}$$

---

[17]Named after Max Planck (German physicist, 1858-1947). He found that the spectrum for the black-body emissive power reads

$$E_{\mathrm{b},\upsilon}(\Theta, \upsilon) = \frac{2\,\pi\,h\,\upsilon^3\,n^2}{c_0^2\,[\exp(h\,\upsilon/k\,\Theta) - 1]} .$$

[18]The refractive index is a dimensionless parameter that describes how radiation is refracted while propagating through a medium. It is defined as the ratio of the speed of light in vacuum to the speed of light in the medium $n = c_0/c_\mathrm{m}$. For air, this index is $n = 1.00027 \approx 1$ and for water (also ice) $n = 1.3325$ [121].

[19]The Stefan-Boltzmann constant in SI units is $\sigma_\mathrm{sb} = 5.670367 \times 10^{-8}$ W m^{-2} K^{-4}.
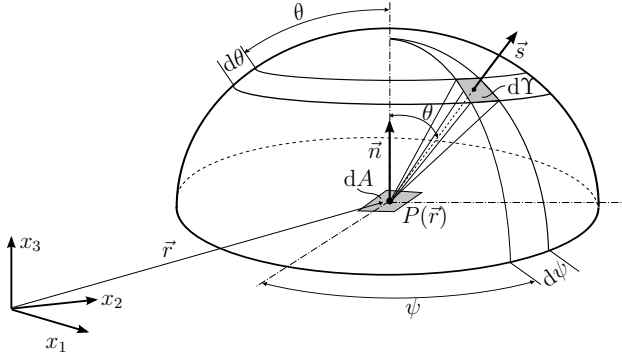
Figure 2.5: Definition of the solid angle $d\Upsilon = \sin\theta\,d\theta\,d\psi$, the normal vector $\vec{n}$, the direction vector $\vec{s}$ and the location vector $\vec{r}$.

Here, θ is the *colatitude* and ψ the *longitude* and $d\Upsilon$ can be viewed as an infinitesimal area projected on a unit sphere. Special angles are the *total solid angle above the surface* Υ_h which defines the corresponding solid angle to a unit hemisphere. From Figure 2.5 follows that, in this case, $\psi \in \{\mathbb{R} \mid 0 \leq \psi \leq 2\pi\}$ and $\theta \in \{\mathbb{R} \mid 0 \leq \theta \leq \pi/2\}$ holds so that the solid angle for a unit hemisphere is calculated to

$$\Upsilon_h = \int\limits_0^{2\pi}\int\limits_0^{\pi/2} \sin\theta\,d\theta\,d\psi = 2\pi\,. \tag{2.137}$$

This is an important measure as it describes all possible directions of a radiating surface from a given point P. According to Υ_h, the *total solid angle* represents a unit sphere and reads

$$\Upsilon_s = \int\limits_0^{2\pi}\int\limits_0^{\pi} \sin\theta\,d\theta\,d\psi = 4\pi\,. \tag{2.138}$$

Typically, solid angles are measured in steradians (sr) which is a dimensionless unit.

From Figure 2.5 it can further be concluded that the scalar product between the direction vector and the normal vector reads

$$\vec{s} \cdot \vec{n} = \cos\theta \tag{2.139}$$

and moreover, that the solid angle of unit hemisphere is a geometric projection onto the surface leading to the relation

$$d\Upsilon = \frac{\cos\theta}{S^2}\,dA\,, \tag{2.140}$$

where S denotes the distance between the point P and dA. The distance, however, for a unit hemisphere is one and therefore $S^2 = 1$.

2.3.1.3   *Radiative heat flux and radiative intensity*

As stated in previous sections, the physics of thermal radiation imply a strong directional dependence. In order to describe the radiation field, a variable is needed that takes these directional dependencies into account. As the emissive power E is independent of the direction, it cannot be used as the primary field variable. Instead, a new measure is introduced – the *radiative intensity I* with the unit Watt per steradians (W/sr) – which incorporates the directional dependence and which can be viewed as the most appropriate variable to describe the energy transfer by means of thermal radiation [121]. It is defined as the transferred radiative energy per unit solid angle normal and per unit area normal to a pencil of rays at a given point $P(\vec{r})$. The vector $\vec{r}$ is denoted as location vector. The *total intensity I* is the integral over all spectral intensities $I_\varkappa$ – the radiative intensity per wavelength – and reads

$$I(\vec{s},\vec{r}) = \int\limits_0^\infty I_\varkappa(\vec{s},\vec{r},\varkappa)\, d\varkappa\,. \tag{2.141}$$

The emissive power, which can be seen as the total amount of emitted energy of a surface element $d\vec{A}$, can be calculated by integrating the radiative intensity I over all possible directions

$$E(\vec{r}) = \int\limits_0^{2\pi}\int\limits_0^{\pi/2} I(\vec{r},\theta,\psi)\cos\theta\sin\theta\, d\theta\, d\psi = \int\limits_{2\pi} I(\vec{r},\vec{s})\,\vec{n}\cdot\vec{s}\, d\Upsilon\,. \tag{2.142}$$

As the primary variable, the computation of the radiative intensity is of major importance for the thermal radiation analysis. An expression for I can be found by an energy balance on a small pencil of rays propagating through a medium. For a detailed discussion, see Section 2.3.3.

Assuming that the radiative intensity is known, it is possible to calculate the radiative heat flux at a certain point on a surface dA. The net radiative heat flux q_r per unit wavelength is the difference

$$\vec{q_r}\cdot\vec{n} = q_r = q_i - q_o \tag{2.143}$$

of the incident heat flux q_i, which is also known as *irradiation*, and the outgoing heat flux q_o. Following MODEST [121], the infinitesimal heat fluxes read

$$dq_i = I(\vec{s}_i)\cos\theta_i\, d\Upsilon_i \quad\text{and}\quad dq_o = I(\vec{s}_o)\cos\theta_o\, d\Upsilon_o\,. \tag{2.144}$$

In order to obtain the net heat flux, it is necessary to perform an integration over the intensity, which is known to be direction-dependent. As a consequence, all possible ray directions of the surface point need to be taken into consideration.

Accordingly, this leads to an integration over a unit hemisphere, or, in other words, over the total solid angle above the surface that is 2π. The incident radiative heat flux reads

$$q_\mathrm{i} = \int\limits_0^{2\pi} \int\limits_0^{\pi/2} I(\vec{s}_\mathrm{i}) \, \cos\theta_\mathrm{i} \sin\psi_\mathrm{i} \, \mathrm{d}\theta_\mathrm{i} \, \mathrm{d}\psi_\mathrm{i} = \int\limits_0^{2\pi} I(\vec{s}_\mathrm{i}) \, \cos\theta_\mathrm{i} \, \mathrm{d}\Upsilon_\mathrm{i} \, , \tag{2.145}$$

and the outgoing heat flux results to

$$q_\mathrm{o} = \int\limits_0^{2\pi} \int\limits_{\pi/2}^{\pi} I(\vec{s}_\mathrm{o}) \, \cos\theta_\mathrm{o} \sin\psi_\mathrm{o} \, \mathrm{d}\theta_\mathrm{o} \, \mathrm{d}\psi_\mathrm{o} = \int\limits_0^{2\pi} I(\vec{s}_\mathrm{o}) \, \cos\theta_\mathrm{o} \, \mathrm{d}\Upsilon_\mathrm{o} \, . \tag{2.146}$$

It was shown by MODEST [121] that the net radiative heat flux can be calculated by means of only one direction vector $\vec{s}$, substituting $\vec{s}_\mathrm{i}$ and $\vec{s}_\mathrm{o}$. Then, the integration is carried out over the total range of solid angles – that is 4π for a unit sphere – and by using $\cos\theta = \vec{s} \cdot \vec{n}$, one obtains the net radiative heat flux

$$\vec{q}_\mathrm{r} \cdot \vec{n} = q_\mathrm{i} - q_\mathrm{o} = \int\limits_0^{4\pi} I(\vec{s}) \, \vec{s} \cdot \vec{n} \, \mathrm{d}\Upsilon \, . \tag{2.147}$$

In a thermal radiation analysis within a participating medium, Eq. (2.147) is the term that appears in the energy balance equation of the fluid. In a vacuum, the radiative heat flux q_r has to be computed too, which is slightly less complicated due to the absence of a medium. Both issues will be discussed in the following sections.

2.3.2   Radiation in vacuum

As mentioned above, thermal radiation is the only heat transfer mode that allows to transfer thermal energy in a vacuum. Here, "vacuum" means that there are no other media involved, so the radiative intensity along a certain path s is not affected. This insight leads to the conclusion that the intensity for any wavenumber η is constant in every direction $\vec{s}$:

$$I_\eta(\vec{s}) = \mathrm{const} \quad \text{and} \quad \frac{\mathrm{d}I_\eta}{\mathrm{d}s} = 0 \tag{2.148}$$

Radiation in the absence of a participating medium is considered as pure surface radiation transport. The energy transport between surfaces is carried out on *idealized enclosures* which can be described by means of geometric properties, the so-called *view factors*. The calculation of these factors is discussed in Section 2.3.2.1. Apart from the geometry, the surface properties influence the thermal radiation analysis. A simplification deals with ideal surfaces which are known as *gray* surfaces, see Section 2.3.2.2 for more details.

2.3.2.1 View factors in thermal radiation

In a vacuum, every radiating surface A_i exchanges energy with all other surfaces A_j that lie in the view field of A_i. Apart from the surface conditions, the amount of energy transfer between two surfaces also depends on the size of the surfaces, the distance S between them and the orientation $\vec{n}$. These dependencies can be described by geometric functions which are known as *view factors*. This means that a view factor is a pure geometric quantity that follows from an energy balance between two surfaces.
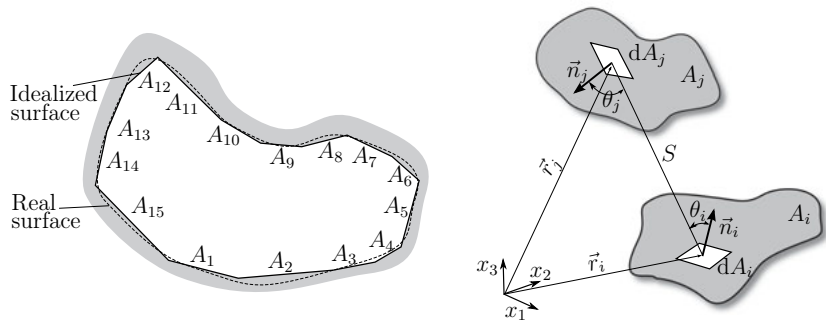
Figure 2.6: View factor in thermal radiation: idealized surface and geometric relations.

In this thesis, the definition introduced by MODEST [121] is used. It states that the infinitesimal view factor $\mathrm{d}F_{\mathrm{d}A_i-\mathrm{d}A_j}$ between two infinitesimal surfaces $\mathrm{d}A_i$ and $\mathrm{d}A_j$ is defined as

$$\mathrm{d}F_{\mathrm{d}A_i-\mathrm{d}A_j} = \frac{\text{energy leaving } \mathrm{d}A_i \text{ directly toward and intercepted by } \mathrm{d}A_j}{\text{total energy leaving } \mathrm{d}A_i \text{ into the hemispehere}}$$
$$= \frac{I(\vec{r}_i)\cos\theta_i\cos\theta_j\,\mathrm{d}A_i\,\mathrm{d}A_j/S^2}{\pi\,I(\vec{r}_i)\,\mathrm{d}A_i}. \tag{2.149}$$

Under the assumption that the intensity $I(\vec{r}) = I$ is constant across the surfaces, this equation can be simplified in such a way that the infinitesimal view factor reads

$$\mathrm{d}F_{\mathrm{d}A_i-\mathrm{d}A_j} = \frac{\cos\theta_i\cos\theta_j\,\mathrm{d}A_j}{\pi\,S^2} = \frac{(\vec{n}\cdot\Delta\vec{r})^2}{\pi\,S^4}\,\mathrm{d}A_j. \tag{2.150}$$

A more descriptive vector form can be obtained from Figure 2.6, using the geometrical relations $\Delta\vec{r} = \vec{r}_j - \vec{r}_i$ and $\cos\theta_i = \vec{n}_i\cdot\Delta\vec{r}/S$ with $S = |\Delta\vec{r}|$. However, the factor defined by Eq. (2.150) holds for infinitesimal surfaces. To make this applicable for the finite case, it is necessary to integrate Eq. (2.149) over the surfaces, leading to the general definition of a view factor:

$$F_{A_i-A_j} = \frac{\int_{A_i}\int_{A_j} I(\vec{r}_i)\cos\theta_i\cos\theta_j\,\mathrm{d}A_i\,\mathrm{d}A_j/S^2}{\int_{A_i}\pi\,I(\vec{r})_i\,\mathrm{d}A_i} = \frac{1}{A_i}\int_{A_i}\int_{A_j}\frac{\cos\theta_i\cos\theta_j}{\pi\,S^2}\mathrm{d}A_i\,\mathrm{d}A_j$$

$$(2.151)$$

For evaluating the view factors, one can take advantage of the following three rules [86]:

Reciprocity Rule: $\quad A_i\, F_{i-j} = A_j\, F_{j-i}$ $\qquad\qquad$ (2.152)

Summation Rule in enclosures: $\quad \displaystyle\sum_{j=1}^{N} F_{i-j} = 1$ $\qquad\qquad$ (2.153)

Subsurface summation: $\quad \displaystyle F_{i-j} = \sum_{k=1}^{K} F_{i-(j,k)} \quad\text{with}\quad A_j = \sum_{k=1}^{K} A_{j,k}$ $\qquad$ (2.154)

In the Appendix A.1, the interested reader can find a comprehensive example for calculating view factors between different surfaces. There exist several methods that deal with the calculation of view factors; the *crossed-strings* method [85], the *inside* and the *unit sphere method* [121] to name a few. An overview of analytic, numerical and also statistical methods can be found in the textbooks of MOD-EST [121] and HOWELL [86]. These textbooks also contain a view factor catalogue for various surface geometries.

2.3.2.2 Black and gray surface radiation

In order to conclude the thermal radiation analysis in a vacuum, ideal surface conditions shall be taken into account. To begin with, the simplest case of black surface conditions is considered. Assuming that the surface temperature is known, the heat flux balance on this infinitesimal surface element $\mathrm{d}A$ reads

$$q(\vec{r}) = E_{\mathrm{b}}(\vec{r}) - H(\vec{r}) \tag{2.155}$$

where $E_{\mathrm{b}} = \sigma_{\mathrm{sb}}\,\Theta^4$ is the black-body emissive power and H the irradiation onto $\mathrm{d}A$. The latter part incorporates the heat flux irradiating from all other surfaces H_{s} at locations $\vec{r}_{\mathrm{s}}$ plus contributions from outside of the enclosure H_0. In accordance with [121] the irradiation can be written as

$$H(\vec{r}) = H_{\mathrm{s}}(\vec{r}, \vec{r}_{\mathrm{s}}) + H_0(\vec{r}) = \int_A E_{\mathrm{b}}(\vec{r}_{\mathrm{s}})\,\mathrm{d}F_{\mathrm{d}A-\mathrm{d}A_{\mathrm{s}}} + H_0(\vec{r})\,. \tag{2.156}$$

Introducing Eq. (2.156) into Eq. (2.155) gives the energy balance of an enclosure with a smoothly distributed surface contour. For the discrete case with N finite surfaces, an average heat flux on each A_i can be calculated, where Eq. (2.155) is recast following [121] as

$$q_i = \frac{1}{A_i} \int_{A_i} q_i(\vec{r}_i)\,\mathrm{d}A_i = E_{\mathrm{b},i} - \sum_{j=1}^{N} E_{\mathrm{b},j} F_{i-j} - H_{0i} = \sum_{j=1}^{N} F_{i-j}\left(E_{\mathrm{b},i} - E_{\mathrm{b},j}\right) - H_{0i}\,.$$

$$(2.157)$$

Based on [86], an extension to gray diffuse surfaces with $\epsilon = \alpha$ reads

$$q(\vec{r}) = \varepsilon(\vec{r})\, E_{\mathrm{b}}(\vec{r}) - \varepsilon(\vec{r})\, H(\vec{r}) = \varepsilon(\vec{r}) \left[E_{\mathrm{b}}(\vec{r}) - \int\limits_A Y(\vec{r}_{\mathrm{s}})\, \mathrm{d}F_{\mathrm{d}A-\mathrm{d}A_{\mathrm{s}}} + H_0(\vec{r}) \right] \,, \quad (2.158)$$

which is again the heat flux balance on a surface element $\mathrm{d}A$. Here, $Y(\vec{r})$ denotes the *surface radiosity* which can be related to the intensity as $Y(\vec{r}) = \pi\, I(\vec{r})$. According to [121], one can derive an equation that eliminates Y and that relates the temperature to the heat flux of an surface element $\mathrm{d}A$ as

$$\frac{q(\vec{r})}{\varepsilon(\vec{r})} = E_{\mathrm{b}}(\vec{r}) + \int\limits_A \left(\frac{1}{\varepsilon(\vec{r}_{\mathrm{s}})} - 1 \right) q(\vec{r}_{\mathrm{s}}) - E_{\mathrm{b}}(\vec{r}_{\mathrm{s}})\, \mathrm{d}F_{\mathrm{d}A-\mathrm{d}A_{\mathrm{s}}} - H_0(\vec{r}) \,. \quad (2.159)$$

Similar to black surface contours, one can deduce a discrete formulation for N subsurfaces A_i so that Eq. (2.159) becomes

$$\frac{q_i}{\varepsilon_i} = \sum_{j=1}^{N} \left[\left(\frac{1}{\varepsilon_j} - 1 \right) F_{i-j}\, q_j + F_{i-j} \left(E_{\mathrm{b},i} - E_{\mathrm{b},j} \right) \right] - H_{0i} \,. \quad (2.160)$$

In this equation, either the surface temperatures Θ_i or the heat flux q_i are unknown. In engineering applications, it is usually the surface temperature that is known – so that the radiative heat flux is the sought variable.

2.3.3 Radiation in participating media

After the radiation in vacuum, the presence of a participating medium shall be considered. If a participating medium is involved, this means that the space between the surfaces where thermal energy is exchanged is filled with a fluid. In many applications, this would be a compressible gas such as air. In this case, the interaction between the surfaces and the surrounding gas needs to be taken into account. Due to the fact that the thermal energy transfer by radiation shows a strong temperature-dependence, the radiation field is accordingly coupled with the thermal field of the participating medium. This has significant consequences: The temperature evolution in the participating medium, which is affected by radiation, needs to be calculated. This is done using the general form of the energy conservation equation for a moving fluid particle in the Eulerian description. Following [121], the balance of energy reads

$$c_{\mathrm{pf}}\, \rho_{\mathrm{f}} \frac{\mathrm{D}}{\mathrm{D}t} \Theta = c_{\mathrm{pf}}\, \rho_{\mathrm{f}} \left(\frac{\partial \Theta}{\partial t} + \vec{v} \cdot \mathrm{grad}\, \Theta \right) = \mathrm{div}\left(\lambda_{\Theta\mathrm{f}}\, \mathrm{grad}\, \Theta \right) - \mathrm{div}\, \vec{q}_{\mathrm{r}} \,, \quad (2.161)$$

where $\vec{v}$ is the fluid particle velocity vector, Θ the fluid temperature and $\vec{q}_{\mathrm{r}}$ is the heat flux due to thermal radiation. This equation also accounts for the two other mechanisms of thermal energy transfer – heat conduction and convection. The term of interest in (2.161) is $\mathrm{div}\, \vec{q}_{\mathrm{r}}$ and in particular the calculation of the radiative heat flux vector $\vec{q}_{\mathrm{r}}$. This issue is addressed in the following.

As discussed in Section 2.3.1.3, the radiative heat flux is a function of the radiative intensity. In contrast to radiation in vacuum, the intensity I along a path s is not constant, which is due to the fact that the propagating electro-magnetic waves are affected by the media through which they are travelling. To be more specific, radiation in participating media is influenced by emission, absorption and by scattering phenomena. In order to find a mathematical description that covers these effects, the change in the intensity I of a small pencil of rays along its way through a medium needs to be taken into consideration. This is equivalent with formulating an energy balance of the radiative transport, finally leading to the so-called *radiative transfer equation*, henceforth abbreviated by RTE. This equation incorporates all possible affections and reads in a general, in-stationary form [121, 86]

$$\frac{1}{c}\frac{\partial I_\eta}{\partial t} + \vec{s}\cdot\nabla I_\eta = \kappa_\eta I_{\mathrm{b},\eta} - \beta_\eta I_\eta + \frac{\sigma_{\mathrm{S}}}{4\pi}\int\limits_{4\pi} I_\eta(s_i)\Phi(s_i,s)\,\mathrm{d}\Upsilon_i\,, \tag{2.162}$$

where κ_η is the absorption coefficient, β_η the extinction coefficient, σ_{S} the scattering coefficient, and Φ is known as the scattering phase function. Before the terms on the right hand side are discussed in detail, a first simplification of (2.162) can be obtained by assuming that the in-stationary term is small and by considering only a certain wave-length or wave-number, i.e. $\eta = \mathrm{const}$ that defines gray media:

$$\frac{\mathrm{d}I}{\mathrm{d}s} = \vec{s}\cdot\nabla I = \kappa\, I_{\mathrm{b}} - \beta\, I + \frac{\sigma_{\mathrm{S}}}{4\pi}\int\limits_{4\pi} I(s_i)\Phi(s_i,s)\,\mathrm{d}\Upsilon_{\mathrm{i}} \quad \text{with} \quad \frac{1}{c}\frac{\partial I}{\partial t} \approx 0 \tag{2.163}$$

This assumption is reasonable since the electro-magnetic waves of thermal radiation travel almost at the speed of light. For the vast majority of engineering problems, this means that space between the radiating surfaces is passed so quickly that any temporal changes are negligible. Simply spoken, most engineering applications are stationary when compared to the speed of light.

The RTE is given by Eq. (2.163), and the first term on the right hand side

$$\mathrm{d}I = \kappa\, I_{\mathrm{b}}\,\mathrm{d}s \tag{2.164}$$

is related to the emitted intensity. This can be viewed as the rate of emitted energy which is proportional to the path length $\mathrm{d}s$. This part incorporates the amount of energy that a small pencil of rays gains due to emission while travelling in the direction $\vec{s}$. As shown by Modest [121], the black-body intensity I_{b} can be used here. The second term in (2.163) accounts for the energy loss due to attenuation by absorption and scattering. This can be approximated by the linear relation

$$\mathrm{d}I = -\beta\, I\,\mathrm{d}s \tag{2.165}$$

where the extinction coefficient β is the sum of two parts $\beta = \kappa + \sigma_{\mathrm{S}}$. The part that incorporates the decrease in the intensity in a pencil of rays due to attenuation by absorption reads $\mathrm{d}I = -\kappa\, I\,\mathrm{d}s$ and by scattering $\mathrm{d}I = -\sigma_{\mathrm{S}}\, I\,\mathrm{d}s$. Attenuation

by absorption means energy conservation when photons of a light beam collide with particles or molecules of the medium. The other phenomena – attenuation by scattering – is known as *out-scattering* as it considers the amount of energy of a light beam that is redirected and not converted. These two effects lead to a decrease in the intensity, indicated by the negative sign. However, within a pencil of rays, out-scattering will occur in any direction – thus leading to *in-scattering* effects for pencils nearby. Following Modest [121], the amount of energy which is scattered into a pencil of rays in the direction $\vec{s}$ reads

$$
\mathrm{d}I(\vec{s}) = \frac{\sigma_{\mathrm{S}}}{4\pi} \int\limits_{4\pi} I(\vec{s_i}) \Phi(\vec{s_i}, \vec{s}) \, \mathrm{d}\Upsilon_i \, \mathrm{d}s \,, \tag{2.166}
$$

where $\vec{s_i}$ denotes all possible incoming directions. The phase function Φ has a probabilistic character, accounting for the probability that rays from incoming directions are scattered into other directions. Two special cases for Φ are mentioned: for simple isotropic scattering, the phase function yields $\Psi = 1$. In-scattering effects are often approached as *linear-anisotropic scattering* resulting in

$$
\Phi(\vec{s_i}, \vec{s}) = 1 + A_1 \, \vec{s_i} \cdot \vec{s}, \tag{2.167}
$$

where the coefficient A_1 is known as the *first scattering phase function coefficient*.

Summing up all contributions from Eqns. (2.164)–(2.166) finally leads to the radiative transfer equation. An illustration is given in Figure 2.7.
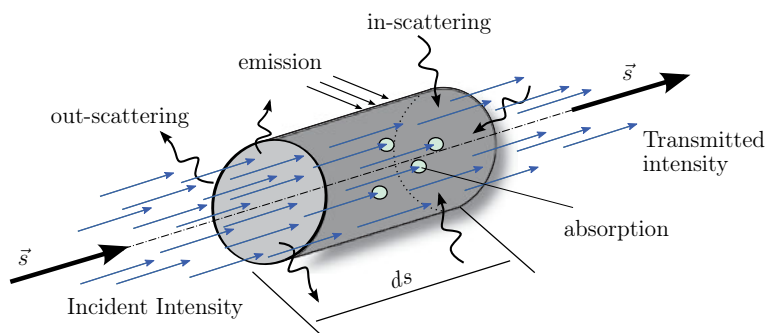

Figure 2.7: Effects that affect the intensity along its path ds: in- and out-scattering, absorption and emission.

2.3.4 Coupling to other fields

There are different possibilities to approximate the phenomenon of thermal radiation in a coupled multi-physics analysis; as a boundary condition by Eq. (2.102) in

the heat equation that describes the thermal field, for example, or as an additional individual field. The first method gives an approximation of minor accuracy, yet it might be sufficient in many applications. The latter option, however, allows to take the interaction of a radiating body with its environment into account. This is of special interest if thermal radiation dominates the energy transfer and the radiating surfaces consist of complex geometries or if the impact of a surrounding fluid needs to be investigated. On the other hand, a full radiation analysis will definitely increase the complexity of the multi-physical problem under consideration.

If radiation is incorporated as an additional field, it is coupled to the thermal as well as to the mechanical field. To this end, a *surface Dirichlet-Neumann coupling* formulation is proposed, similar to fluid-structure interaction. In order to describe the coupling, the body $\mathcal{B}_t \subset \mathbb{R}^3$ with the domain $\Omega_t^{\mathcal{B}}$ is considered in the current configuration. These domains hold for the thermal as well as for the mechanical field. Further, the environment of $\mathcal{B}_t$ – that is the fluid region including the radiation field $\Sigma_t \subset \mathbb{R}^3$ – is defined by the fluid domain Ω_t^{Σ}. As depicted in Figure 2.8, the coupling interface $\partial \Omega_t^c \subset \partial \Omega_t^{\mathcal{B}}$ and $\partial \Omega_t^c \subset \partial \Omega^{\Sigma}$ is the common boundary between these domains and is termed as follows: In general, the coupling domain is composed as $\partial \Omega_t^c = \partial \Omega_t^{\mathcal{B}} \cap \partial \Omega_t^{\Sigma}$. However, if the body is fully enclosed by Σ_t, which can be viewed as the standard case, one can conclude that $\partial \Omega_t^{\mathcal{B}} \subseteq \partial \Omega_t^{\Sigma}$ must hold and, thus, $\partial \Omega_t^c \subseteq \partial \Omega_t^{\Sigma}$ and also $\partial \Omega_t^c \subseteq \partial \Omega_t^{\mathcal{B}}$ respectively. In order to formulate the coupling between the fields, the equilibrium conditions
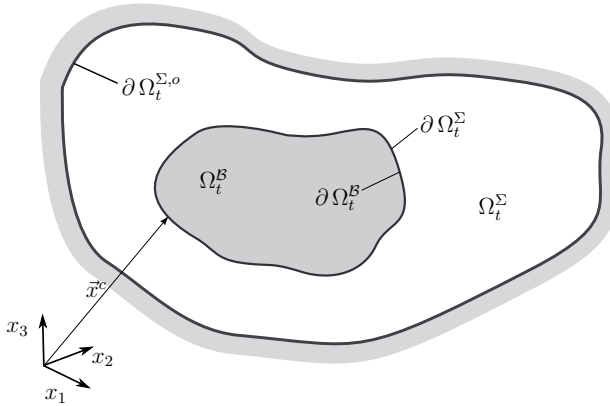
Figure 2.8: Surface boundary description of a radiating body $\Omega_t^{\mathcal{B}}$ that is fully enclosed by its environment (radiation field) Σ_t.

at the interfaces are taken into consideration. On the coupling interface $\partial \Omega_t^c$ the equilibrium condition for the temperature is

$$\bar{\Theta}_{\partial \Omega_t^{\mathcal{B}}}(\vec{x}^c, t) = \bar{\Theta}_{\partial \Omega_t^{\Sigma}}(\vec{x}^c, t) , \qquad (2.168)$$

where $\bar{\Theta}(\vec{x}^c, t)$ is the surface temperature on the boundary and $\vec{x}^c \in \partial\Omega_t^c$. The second equilibrium condition is the heat flux condition

$$\vec{n}_{\partial\Omega_t^\mathcal{B}} \cdot \vec{q}_{\partial\Omega_t^\mathcal{B}}(\vec{x}^c) = -\vec{n}_{\partial\Omega_t^\Sigma} \cdot \vec{q}_{\partial\Omega_t^\Sigma}(\vec{x}^c) \,, \tag{2.169}$$

where $\vec{n}$ represents the unit normal vector that is pointing outwards of the domain boundary and $\vec{q}(\vec{x}^c)$ is the heat flux vector on $\partial\Omega_t^c$. The equilibrium conditions clearly state that the thermal field is coupled to the radiation over the Neumann- and also over the Dirichlet-boundary. This is similar to FSI problems where Dirichlet-Neumann coupling occurs too.

Indeed, there is also a coupling interface between the mechanical and the radiation field. It is known as the *kinematic condition*. To fulfil the condition, the displacements $\vec{d}(\vec{x}^c)$ for $\vec{x}^c \in \partial\Omega_t^c$ at the interface must be equal. Here, $\vec{d}$ is used to distinguish between interface displacements and the displacements of the whole body denoted by $\vec{u}$. Accordingly, this leads to the following condition

$$\vec{d}_{\partial\Omega_t^\mathcal{B}}(\vec{x}^c) = \vec{d}_{\partial\Omega_t^\Sigma}(\vec{x}^c) \,. \tag{2.170}$$

A dynamic condition that claims the equilibrium between the stresses on the boundary is neglected. This is a reasonable assumption since the mechanical problem is to be considered as a quasi-static field and the stresses due to the pressure onto the boundary, which results from the fluid flow of the participating medium, are not very high compared to the stresses from external forces applied to the body $\mathcal{B}_t$. However, due to the kinematic change of the region of the solid, the fluid domain of the radiation field must be updated constantly. This means that the radiation problem needs to be solved on a moving configuration.
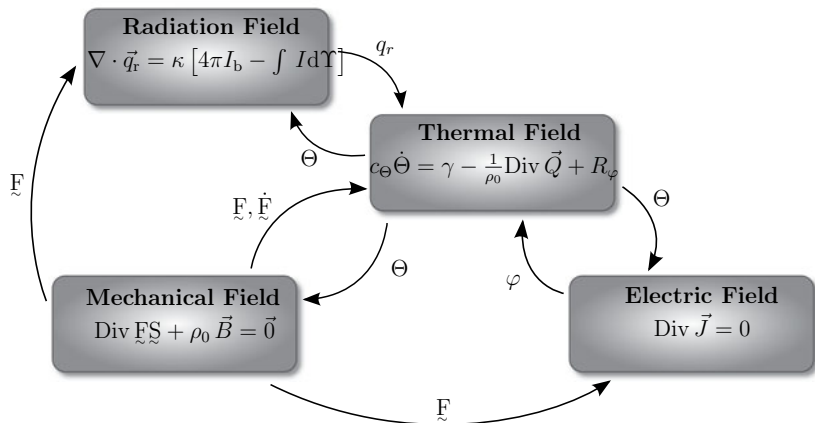

Figure 2.9: Dependencies of the radiation field with the mechanical, thermal and electric fields.

So far, the coupling of the radiation field is formulated within the frame of a participating medium. Since thermal radiation allows to transfer energy in a vacuum, this special case has to be discussed too. In vacuum, radiation reduces to a geometric problem, and the interaction takes place between the boundary of the body $\partial \Omega_t^{\mathcal{B}}$ and the outer boundary that describes the environment $\partial \Omega_t^{\Sigma,o}$. Between these two boundaries, the temperatures and heat fluxes can be computed by means of Eq. (2.159). This is the coupling of the radiation field in vacuum to the thermal field of the body. However, the view factors which cover the geometry need to be calculated between the infinitesimal surface elements $\mathrm{d}A_t^{\mathcal{B}} \in \partial \Omega_t^{\mathcal{B}}$ and $\mathrm{d}A_t^{\Sigma,o} \in \partial \Omega_t^{\Sigma,o}$. Since the surface elements of the body $\mathcal{B}_t$ are not constant, the view factors need to be calculated with respect to the current surface displacement $\vec{d}(\vec{x}^c)$. Hence, the definition for the view factor given by Eq. (2.150) is reformulated in dependence of the displacement vector to

$$\mathrm{d}F_{\mathrm{d}A_t^{\mathcal{B}}-\mathrm{d}A_t^{\Sigma,o}} = \frac{\cos\theta_t^{\mathcal{B}}(\vec{d}(\vec{x}^c))\cos\theta_t^{\Sigma,o}\,\mathrm{d}A_t^{\Sigma,o}}{\pi\,S^2(\vec{d}(\vec{x}^c))}\,. \tag{2.171}$$

As a consequence, it can be concluded that in both cases, for vacuum as well as for a participating medium, the radiation field is coupled to the thermal and the mechanical field. With regard to the electro-thermo-mechanical problem in Section 2.2.4, the interactions between the four fields are illustrated in Figure 2.9. In this figure, all dependencies between the fields are depicted on an abstract level, ultimately representing a volume- and surface-coupled problem.

Numerical methods

The third chapter is concerned with numerical methods to find an approximate solution to the coupled problem with the aid of computers. Special attention is placed on numerical solution procedures for the governing equations of the mechanical, thermal, electrical and radiation field that were derived in the previous chapter. Due to the fact that the descriptive field equations are of a differential or integral type, both in space or time, a discrete formulation is desired for the further solution process. To begin with, a variational formulation of the mechanical, thermal and electrical field is presented, followed by a linearization of the nonlinear terms. After that, the spatial discretization of the body $\mathcal{B}$ using the *finite element method* (FEM) is described and the time integration procedure for the temporal discretization is discussed. In regard to the radiation field, the *finite volume method* (FVM) is employed to discretize the fluid domain Σ which can be viewed as the environment of $\mathcal{B}$. Moreover, several methods to tackle radiative heat transfer numerically in vacuum and participating media are introduced. The latter case involves an approximate solution of the *radiative transfer equation* (RTE) which is also discussed in this chapter.

3.1 Variational formulation and linearization

With respect to the numerical solution of the electric, thermal and mechanical fields with finite elements, the objective is to obtain a discrete formulation of the underlying partial-differential equations. Since these equations contain continuous mathematical operators, a conversion into discrete systems of equations is desired. To achieve this, a variational form of the coupled problem is derived which is also known as *weak formulation*. A differential equation can be converted into its weak form by multiplying it with so-called *test functions* $\vec{\eta}$ and by integrating over the domain $\Omega_0 \subset \mathbb{R}^n$ which is occupied by the body $\mathcal{B}$ under investigation. The result is a equation which now has *weak solutions* in Ω_0 with respect to the test functions. Such functions are chosen out of the following *test space* $\mathcal{V}$

$$\vec{\eta}(\vec{X}) \in \mathcal{V}, \quad \mathcal{V} := \left\{ \vec{\eta}(\vec{X}) \text{ with } \vec{X} \in \Omega_0 \,\middle|\, \vec{\eta} = \vec{0} \text{ on } \partial\Omega_0 \right\}. \tag{3.1}$$

Applying this concept to the balance equations of the involved fields leads to equations that describe the equilibrium in a weak sense. Since these equations are typically nonlinear, a linearization is required for the further numerical solution process. To outline the concept of linearization, an arbitrary nonlinear equation $\mathcal{G}(\vec{a})$ is considered, where the variable $\vec{a}$ is assumed to be vector-valued. The formalism of linearization begins with a first-order Taylor series expansion of the functional

$$\mathcal{G} = \mathcal{G}(\vec{a}) + \Delta\mathcal{G}(\vec{a}, \Delta\vec{a}) + R \tag{3.2}$$

where R corresponds to the residual. By omitting R, the linearized equation at a given point $\vec{a}$ reads

$$\operatorname{Lin}\mathcal{G}|_{\vec{a}=\bar{\vec{a}}} = \mathcal{G}(\bar{\vec{a}}) + \mathrm{D}\mathcal{G} \cdot \Delta\vec{a} \tag{3.3}$$

in which $\Delta\vec{a}$ denotes the incremental value of $\vec{a}$. The first term in Eq. (3.3) is the evaluation of the functional at the given point $\bar{\vec{a}}$, and the second part is known as the *directional derivative* in the direction of $\Delta\vec{a}$. This part can be calculated with the aid of the *Gâteaux derivative*

$$\mathrm{D}\mathcal{G} \cdot \Delta\vec{a} = \frac{\partial\,\mathcal{G}}{\partial\,\vec{a}} \cdot \Delta\vec{a} = \frac{\mathrm{d}}{\mathrm{d}\epsilon}\left[\mathcal{G}(\vec{a} + \epsilon\,\Delta\vec{a})\right]\Big|_{\epsilon=0}. \tag{3.4}$$

The linearization concept can be used, amongst other things, to linearize kinematic relations and constitutive equations. It forms the basis of the Newton-Raphson procedure which is used in this thesis to solve nonlinear systems of equations. In a multi-physically coupled context, a linearization would consequently have to linearize the nonlinear problem with respect to all quantities under consideration. This is referred to as a *consistent linearization* [89]. However, if the coupled system is solved in a separated manner, which is indeed the case when employing a partitioned coupling strategy, each of the fields only need to be linearized with respect to its primary unknown.

3.1.1 Mechanical field

To begin with, the weak formulation of the mechanical field in the reference and current configuration is derived. A variational form $\mathcal{G}_{\mathrm{M}}$ is established by multiplying the quasi-static balance of linear momentum with test functions $\vec{\eta}$ and by integrating over the whole domain. The test function is also known as a *weight function*, and it is supposed to fulfil the Dirichlet-boundary condition $\vec{\eta} = \{\vec{\eta}\,|\,\vec{\eta} = \vec{0}$ on $\partial\,\Omega_0\}$. The test functions $\vec{\eta}$ correspond to the *virtual displacements* $\delta\vec{u}$, leading to

$$\mathcal{G}_{\mathrm{M}} := \int\limits_{\Omega_0}\left(\operatorname{Div}\underset{\sim}{P} + \rho_0\,\vec{b}\right)\cdot\delta\vec{u}\,\,\mathrm{d}V = 0\,. \tag{3.5}$$

This equation can be rearranged by applying the *divergence theorem* which delivers the *principle of virtual work* in engineering [171]

$$\mathcal{G}_{\mathrm{M}} := \int_{\Omega_0} \underset{\sim}{\mathrm{P}} \cdot \mathrm{Grad}\,\delta\vec{u}\,\mathrm{d}V - \int_{\partial\Omega_0^\sigma} \vec{\bar{T}} \cdot \delta\vec{u}\,\mathrm{d}A - \int_{\Omega_0} \rho_0 \vec{b} \cdot \delta\vec{u}\,\mathrm{d}V = 0\,. \tag{3.6}$$

This expression is the weak form of the mechanical field, formulated in terms of the first Piola-Kirchhoff stress tensor $\underset{\sim}{\mathrm{P}}$. In order to obtain a description that is formulated in terms of the symmetric second Piola-Kirchhoff stress tensor, the relation $\underset{\sim}{\mathrm{P}} = \underset{\sim}{\mathrm{F}}\underset{\sim}{\mathrm{S}}$ is used. Following [171], it can be deduced that

$$\underset{\sim}{\mathrm{P}} \cdot \mathrm{Grad}\,\delta\vec{u} = \underset{\sim}{\mathrm{S}} \cdot \underset{\sim}{\mathrm{F}}^T \mathrm{Grad}\,\delta\vec{u} = \underset{\sim}{\mathrm{S}} \cdot \frac{1}{2}\left(\underset{\sim}{\mathrm{F}}^T \mathrm{Grad}\,\delta\vec{u} + \mathrm{Grad}^T \delta\vec{u}\,\underset{\sim}{\mathrm{F}}\right) = \underset{\sim}{\mathrm{S}} \cdot \delta\underset{\sim}{\mathrm{E}}\,, \tag{3.7}$$

where $\delta\underset{\sim}{\mathrm{E}}$ is the virtual Green-Lagrange strain tensor. This finally leads to the virtual work formulated in the reference configuration

$$\int_{\Omega_0} \underset{\sim}{\mathrm{S}} \cdot \delta\underset{\sim}{\mathrm{E}}\,\mathrm{d}V = \int_{\partial\Omega_0^\sigma} \vec{\bar{T}} \cdot \delta\vec{u}\,\mathrm{d}A + \int_{\Omega_0} \rho_0 \vec{b} \cdot \delta\vec{u}\,\mathrm{d}V\,. \tag{3.8}$$

It follows by using the relation $\underset{\sim}{\mathrm{P}} = J\underset{\sim}{\sigma}\,\underset{\sim}{\mathrm{F}}^{-T}$ and $\mathrm{grad}\,\delta\vec{u} = \mathrm{Grad}\,\delta\vec{u}\,\underset{\sim}{\mathrm{F}}^{-1}$ that

$$\underset{\sim}{\mathrm{P}} \cdot \mathrm{Grad}\,\delta\vec{u} = J\underset{\sim}{\sigma}\,\underset{\sim}{\mathrm{F}}^{-T} \cdot \mathrm{Grad}\,\delta\vec{u} = J\underset{\sim}{\sigma} \cdot \mathrm{Grad}\,\delta\vec{u}\,\underset{\sim}{\mathrm{F}}^{-1} = J\underset{\sim}{\sigma} \cdot \mathrm{grad}\,\delta\vec{u}\,. \tag{3.9}$$

holds. With $\mathrm{d}v = J\,\mathrm{d}V$, $\rho_0 = J\rho$ and *Nanson's formula*, given by Eq. (2.4), the virtual work in the spatial configuration reads

$$\int_{\Omega_t} \underset{\sim}{\sigma} \cdot \mathrm{grad}\,\delta\vec{u}\,\mathrm{d}v = \int_{\Omega_t} \underset{\sim}{\sigma} \cdot 1/2\left(\mathrm{grad}\,\delta\vec{u} + \mathrm{grad}^{\,\mathrm{T}}\,\delta\vec{u}\right)\mathrm{d}v = \int_{\partial\Omega_t^\sigma} \vec{\bar{t}} \cdot \delta\vec{u}\,\mathrm{d}a + \int_{\Omega_t} \rho\,\vec{b} \cdot \delta\vec{u}\,\mathrm{d}v\,. \tag{3.10}$$

A physical interpretation can be given by considering that the part on the left-hand side in Eq. (3.8) denotes the internal virtual work $\delta\mathcal{W}_{\mathrm{int}}$ performed by a stress field due to virtual displacements. The other part on the right-hand side denotes the external virtual work $\delta\mathcal{W}_{\mathrm{ext}}$ performed by the applied loading. It can thus be concluded that, in the state of equilibrium, the amount of internal work is equivalent to the amount of external work

$$\delta\mathcal{W}_{\mathrm{int}} - \delta\mathcal{W}_{\mathrm{ext}} = 0\,. \tag{3.11}$$

For the linearization of the mechanical field, only the internal mechanical work is taken into consideration, whereas the external part is assumed to be independent of the deformation[1].

---

[1]If $\delta\mathcal{W}_{\mathrm{ext}}$ depends upon the displacements, this typically involves deformation-dependent loads that are known as *follower loads*. Their numerical treatment is discussed in [171, p. 142–145], for instance.

Due to the fact that a partitioned solution strategy is employed, a consistent linearization of $\delta \mathcal{W}_{\text{int}}$ with respect to the temperature is not needed and is therefore not shown in the following. Consequently, the displacements $\vec{u}$ are assumed to be the unknown variable – and the Gâteaux derivative thus reads

$$\mathrm{D}_{\vec{u}}\, \mathcal{G}_{\text{M}} \cdot \Delta\vec{u} = \mathrm{D}_{\vec{u}}\, \delta\mathcal{W}_{\text{int}} \cdot \Delta\vec{u} = \frac{\mathrm{d}}{\mathrm{d}\epsilon} \left[\delta\mathcal{W}(\vec{u} + \epsilon\,\Delta\vec{u}, \delta\vec{u}) \right]\Big|_{\epsilon=0}, \tag{3.12}$$

and, following [84, 171], this leads under the assumption of an elastic material[2] to the linearized internal virtual mechanical work related to the reference configuration

$$
\begin{aligned}
\mathrm{D}_{\vec{u}}\, \delta\mathcal{W}_{\text{int}} \cdot \Delta\vec{u} &= \frac{\mathrm{d}}{\mathrm{d}\epsilon} \int\limits_{\Omega_0} \underset{\sim}{\mathrm{S}}(\underset{\sim}{\mathrm{E}}(\vec{u} + \epsilon\Delta\vec{u})) \cdot \delta\underset{\sim}{\mathrm{E}}(\vec{u} + \epsilon\Delta\vec{u})\, \mathrm{d}V \bigg|_{\epsilon=0} \\
&= \int\limits_{\Omega_0} \mathrm{Grad}\,\delta\vec{u} \cdot \mathrm{Grad}\,\Delta\vec{u}\,\underset{\sim}{\mathrm{S}} + \delta\underset{\sim}{\mathrm{E}} \cdot \underset{\sim}{\mathcal{C}}\, \Delta\underset{\sim}{\mathrm{E}}\, \mathrm{d}V.
\end{aligned}
\tag{3.13}
$$

Therein, $\underset{\sim}{\mathcal{C}}$ denotes the fourth-order elasticity tensor in material coordinates. For the computation of the Gâteaux derivative, the incremental deformation gradient

$$\Delta\underset{\sim}{\mathrm{F}} = \mathrm{Grad}\,\Delta\vec{u} \tag{3.14}$$

and the incremental strain tensor

$$\Delta\underset{\sim}{\mathrm{E}} = \frac{1}{2}\left(\underset{\sim}{\mathrm{F}}^T \Delta\underset{\sim}{\mathrm{F}} + \Delta\underset{\sim}{\mathrm{F}}^T \underset{\sim}{\mathrm{F}}\right) = \frac{1}{2}\left(\underset{\sim}{\mathrm{F}}^T \mathrm{Grad}\Delta\vec{u} + \mathrm{Grad}^T\Delta\vec{u}\, \underset{\sim}{\mathrm{F}}\right) \tag{3.15}$$

are utilized, see [171] for instance. With respect to the current configuration, see e. g. [84], the linearization of internal mechanical work follows by means of push-forward operations to

$$\mathrm{D}_{\vec{u}}\, \delta\mathcal{W}_{\text{int}} \cdot \Delta\vec{u} = \int\limits_{\Omega_t} \mathrm{grad}\,\delta\vec{u} \cdot \mathrm{grad}\,\Delta\vec{u}\,\underset{\sim}{\sigma} + \mathrm{grad}\,\delta\vec{u} \cdot \mathcal{C}_t\, \mathrm{grad}\,\Delta\vec{u}\, \mathrm{d}v, \tag{3.16}$$

where $\mathcal{C}_t$ is the fourth-order elasticity tensor referring to the spatial configuration.

3.1.2 Thermal field

In order to compute the thermal field with the finite element method, a weak formulation of the heat conduction equation is required. To this end, Eq. (2.88) is multiplied with the *virtual temperature* $\delta\Theta$, and an integration throughout the control-volume delivers the variational form $\mathcal{G}_{\Theta}$ for the thermal field in the reference configuration:

$$\mathcal{G}_{\Theta} = \int\limits_{\Omega_0} \left(\rho_0\, c_{\text{p}}\, \dot{\Theta}\, \delta\Theta - \gamma\,\delta\Theta - D_{\text{int}}\,\delta\Theta + \mathrm{Div}\,\vec{Q}\,\delta\Theta - \rho_0\, R_{\varphi}\,\delta\Theta \right) \mathrm{d}V = 0 \tag{3.17}$$

---

[2]For inelasticity, see the remark in Section 3.2.5

Using Fourier's law of heat conduction given by Eq. (2.65), the divergence term of the heat flux $\vec{Q}$ can be rewritten by means of the divergence theorem to

$$
\begin{aligned}
\int\limits_{\Omega_0} \text{Div}\,\vec{Q}\,\delta\Theta\,\mathrm{d}V &= \int\limits_{\Omega_0} \text{Div}(\vec{Q}\,\delta\Theta)\,\mathrm{d}V - \int\limits_{\Omega_0} \vec{Q}\cdot\text{Grad}\,\delta\Theta\,\mathrm{d}V \\
&= -\int\limits_{\partial\Omega_0^q} \bar{Q}\,\delta\Theta\,\mathrm{d}A + \int\limits_{\Omega_0} \text{Grad}\,\delta\Theta\cdot\underset{\sim}{\Lambda}_{\Theta}\,\text{Grad}\,\Theta\,\mathrm{d}V\,,
\end{aligned}
\tag{3.18}
$$

where $\bar{Q} = \vec{Q}\cdot\vec{N}$ is the heat flux that enters a control-volume across the surface $\mathrm{d}A$. Substituting this expression into Eq. (3.17), the final weak form of the equation of heat conduction yields

$$
\begin{aligned}
\mathcal{G}_{\Theta} = \int\limits_{\Omega_0} &\left(\rho_0\,c_{\mathrm{p}}\,\dot{\Theta}\,\delta\Theta + \text{Grad}\,\delta\Theta\cdot\underset{\sim}{\Lambda}_{\Theta}\text{Grad}\,\Theta\right)\mathrm{d}V + \\
&-\int\limits_{\Omega_0} (\gamma + D_{\text{int}} + \rho_0\,R_{\varphi})\,\delta\Theta\,\mathrm{d}V - \int\limits_{\partial\Omega_0^q} \bar{Q}\,\delta\Theta\,\mathrm{d}A = 0
\end{aligned}
\tag{3.19}
$$

in material description and

$$
\begin{aligned}
\mathcal{G}_{\Theta} = \int\limits_{\Omega_t} &\left(\rho\,c_{\mathrm{p}}\,\dot{\Theta}\,\delta\Theta + \text{grad}\,\delta\Theta\cdot\underset{\sim}{\lambda}_{\Theta}\,\text{grad}\,\Theta\right)\delta\Theta\,\mathrm{d}v \\
&-\int\limits_{\Omega_t} (\gamma + d_{\text{int}} + \rho\,r_{\varphi})\,\delta\Theta\,\mathrm{d}v - \int\limits_{\partial\Omega_t^q} \bar{q}\,\delta\Theta\,\mathrm{d}a = 0
\end{aligned}
\tag{3.20}
$$

in spatial description.

Since $\mathcal{G}_{\Theta}$ is nonlinear in the temperature, it has to be linearized. The nonlinearity is due to temperature-dependent material parameters, coupling terms and the boundary conditions. Before proceeding with the linearization, the rate of temperataure $\dot{\Theta}$ is replaced by means of the following expression

$$
\dot{\Theta} \approx \frac{\Theta(t+\Delta t) - \Theta(t)}{\Delta t}\,,
\tag{3.21}
$$

where Δt denotes the time increment. A first-order accurate *implicit Euler-Backward method* is used, for details see Section 3.2.2.5. Similar to the mechanical field, the Gâteaux derivative is applied at time $t + \Delta t$ to compute the linearization

$$
\begin{aligned}
\mathrm{D}_{\Theta}\mathcal{G}_{\Theta}\,\Delta\Theta = \int\limits_{\Omega_0} &\rho_0\left(\frac{\partial c_{\mathrm{p}}}{\partial\Theta}\frac{\Theta - \Theta_n}{\Delta t} + c_{\mathrm{p}}\frac{1}{\Delta t}\right)\Delta\Theta\,\delta\Theta\,\mathrm{d}V \\
&+\int\limits_{\Omega_0}\left(\text{Grad}\,\delta\Theta\cdot\underset{\sim}{\Lambda}_{\Theta}\,\text{Grad}\,\Delta\Theta + \text{Grad}\,\delta\Theta\cdot\frac{\partial\underset{\sim}{\Lambda}_{\Theta}}{\partial\Theta}\,\text{Grad}\,\Theta\,\Delta\Theta\right)\mathrm{d}V \\
&-\int\limits_{\Omega_0}\left(\frac{\partial\gamma}{\partial\Theta} + \frac{\partial D_{\text{int}}}{\partial\Theta} + \rho_0\frac{\partial R_{\varphi}}{\partial\Theta}\right)\Delta\Theta\,\delta\Theta\,\mathrm{d}V - \int\limits_{\partial\Omega_0^q}\frac{\partial\bar{Q}}{\partial\Theta}\,\Delta\Theta\,\delta\Theta\,\mathrm{d}A\,.
\end{aligned}
\tag{3.22}
$$

Computing the linearization with respect to the temperature typically requires the calculation of derivatives of analytic functions that are used to approximate the temperature-dependency of the material parameters or the boundary condition. Thanks to the benefits of the partitioned approach, a linearization with respect to the deformation and the electric potential is not taken into consideration.

3.1.3 Electric field

The variational formulation of the stationary equation of charge – as given by Eq. (2.113) – that describes the electric field reads

$$\mathcal{G}_\varphi := \int_{\Omega_0} \text{Div}\,\vec{J}\,\delta\varphi\,\mathrm{d}V = 0\,, \tag{3.23}$$

where $\delta\varphi$ denotes the *virtual electric potential*. Similar to the divergence term of the heat flux vector in the previous section, one can conclude that

$$\begin{aligned}
\int_{\Omega_0} \text{Div}\,\vec{J}\,\delta\varphi\,\mathrm{d}V &= \int_{\Omega_0} \text{Div}\,(\vec{J}\,\delta\varphi)\,\mathrm{d}V - \int_{\Omega_0} \vec{J}\cdot\text{Grad}\,\delta\varphi\,\mathrm{d}V \\
&= -\int_{\partial\Omega_0} \bar{J}\,\delta\varphi\,\mathrm{d}A + \int_{\Omega_0} \text{Grad}\,\delta\varphi\cdot\underset{\sim}{\Lambda}_\varphi\,\text{Grad}\,\varphi\,\mathrm{d}V
\end{aligned} \tag{3.24}$$

can be obtained for the electric current density. Here, $-\vec{J}\cdot\vec{N} = \bar{J}$ and Ohm's law $\vec{J} = -\underset{\sim}{\Lambda}_\varphi\,\text{Grad}\,\varphi$ were used to write Eq. (3.23) as

$$\mathcal{G}_\varphi := \int_{\Omega_0} \text{Grad}\,\delta\varphi\cdot\underset{\sim}{\Lambda}_\varphi\,\text{Grad}\,\varphi\,\mathrm{d}V - \int_{\partial\Omega_0} \bar{J}\,\delta\varphi\,\mathrm{d}A = 0\,. \tag{3.25}$$

In the next step, a linearization of $\mathcal{G}_\varphi$ is carried out by means of the Gâteaux derivative, leading to

$$\mathrm{D}_\varphi\mathcal{G}_\varphi\,\Delta\varphi = \int_{\Omega_0} \text{Grad}\,\delta\varphi\cdot\underset{\sim}{\Lambda}_\varphi\text{Grad}\,\Delta\varphi\,\mathrm{d}V - \int_{\partial\Omega_0} \frac{\partial\bar{J}}{\partial\varphi}\,\Delta\varphi\,\delta\varphi\,\mathrm{d}A\,. \tag{3.26}$$

Since the electric conductivity λ_φ does not depend on the electric potential φ, and the derivative of $\partial\bar{J}/\partial\varphi = \lambda_{\mathrm{ec}}$ results in a constant value, see Eq. (2.126), the equation for the electric field, see Eq. (3.25), is assumed to be linear in the electric potential.

3.2 Finite element method

The following section serves to summarize the principles of the finite element method (FEM). The FEM is known to be one of the most commonly used numerical methods to solve partial differential equations and is nowadays an essential tool in computational engineering.

The underlying idea of the FEM can very briefly be summarized as follows: The starting point is to subdivide a domain Ω_0 into a finite number of elements with a very simple shape on which the variables of interest and also the geometry of the shape are interpolated by analytic functions. This concept is applied to the weak forms of the balance equations, and the final outcome is a discrete nonlinear system of equations to compute an approximate solution to the entire domain Ω_0.

The finite element methodology is well documented in many textbooks. For a comprehensive description of the fundamentals, which typically involve linear static procedures, the interested reader is referred to [7, 88, 180, 179, 145] who also offer some historical background of the FEM. Besides, nonlinear procedures are of particular interest, and many textbooks are devoted to this issue, see [29, 10, 144, 13, 171] as selected examples. The following introduction of the FEM is given with close reference to the work of WRIGGERS [171].

3.2.1 Spatial discretization with the isoparametric concept

To begin with, the domain Ω_0 of a body in the initial configuration is approximated by n_e finite elements such that

$$\Omega_0 \approx \Omega_h = \bigcup_{e=1}^{n_e} \Omega^e \tag{3.27}$$

holds. Here, Ω_h is the approximation of Ω_0, and $\Omega^e \subset \Omega_h$ denotes the domain of one finite element. In the following, the expression $\bigcup_{e=1}^{n_e}$ stands the union of all n_e finite elements. In the same manner, the boundary $\partial\Omega_0 \approx \partial\Omega_h$ is approximated where $\partial\Omega^e \subset \partial\Omega_h$ denotes the boundary of one finite element. All elements are connected by nodes, see Figure 3.1. Together, they build a finite element mesh that has to be continuous, meaning that no gaps or overlapping elements are allowed.
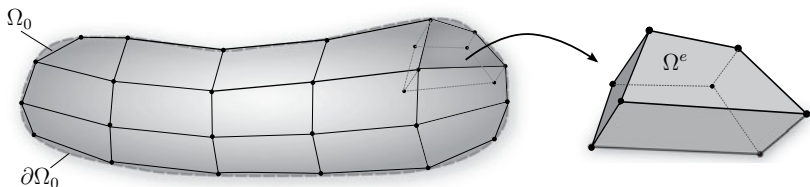

Figure 3.1: Discretized domain of the body with eight-noded hexahedral elements.

In order to approximate the element geometry $\boldsymbol{X}^e$, the n nodal coordinates $\boldsymbol{X}_I$ are used to interpolate $\boldsymbol{X}^e$ as

$$\boldsymbol{X}^e(\boldsymbol{\xi}) = \sum_{I=1}^{n} N_I(\boldsymbol{\xi})\, \boldsymbol{X}_I\,, \tag{3.28}$$

where the N_I are known as the *shape functions*. These functions are formulated in *local coordinates* $\boldsymbol{\xi} = [\xi, \eta, \zeta]^T$ which are defined within a *standard element* Ω_*. Such an element can have different shapes. In the three-dimensional case, for example, hexahedrons or tetrahedrons are frequently employed elements. Hereinafter, only hexahedral elements are taken under consideration. In this case, the local coordinates of Ω_* are given by $\xi \in [-1, 1]$, $\eta \in [-1, 1]$ and $\zeta \in [-1, 1]$. In the very simplest case, eight linear shape functions are chosen, one function for each node. The functions used here are known as *Lagrange polynomials*

$$N_I(\boldsymbol{\xi}) = \hat{N}_1(\xi)\,\hat{N}_2(\eta)\,\hat{N}_3(\zeta) = \frac{1}{8}(1+\xi_I\,\xi)(1+\eta_I\,\eta)(1+\zeta_I\,\zeta)\,, \quad I = 1, 2, ..., 8\,, \quad (3.29)$$

where $\boldsymbol{\xi}_I = [\xi_I, \eta_I, \zeta_I]^T$ denote the I-th node coordinate[3], as given in Figure 3.2. A more comprehensive discussion regarding shape functions is given in Section 3.2.3, focusing particularly on the construction of high-order finite elements.
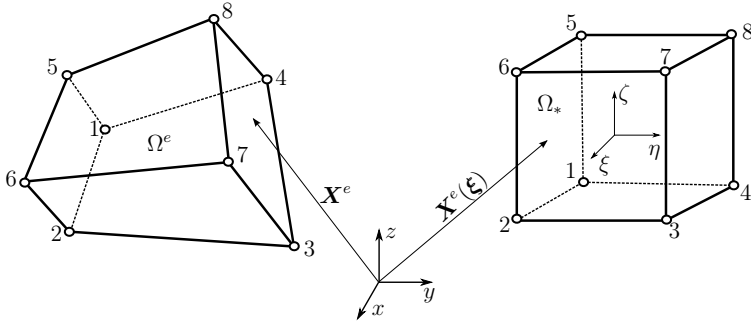

Figure 3.2: Physical and local coordinates for an eight-noded hexahedral element.

For the further finite element formulation, it is essential to define a mapping between Ω^e and Ω_*, as illustrated in Figure 3.3. To this end, a transformation is required that relates the global to the local coordinates. In the following, a transformation known as *isoparametric transformation* is utilized, giving the concept its name as *isoparametric concept*. The advantage of this concept is that the same shape functions can be used to interpolate the geometry and the primary field variables.

In order to map quantities between the standard element Ω_* and Ω^e, the *Jacobian transformation* is introduced. Following the notation of WRIGGERS [171], the

---

[3]For example, if $I = 1$ then $\boldsymbol{\xi}_I = [-1, -1, -1]^T$ which leads to

$$N_1(\boldsymbol{\xi}) = \frac{1}{8}(1 - \xi)(1 - \eta)(1 - \zeta) \tag{3.30}$$
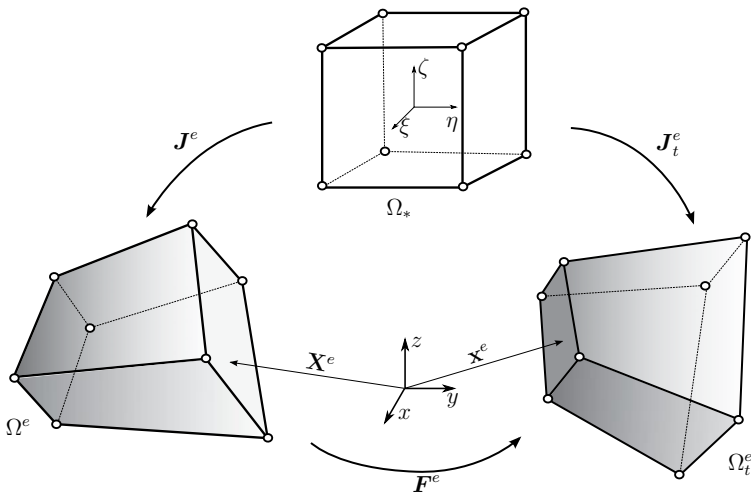
Figure 3.3: Isoparametric mapping concept.

Jacobian $\boldsymbol{J}^e$ reads

$$\boldsymbol{J}^e = \operatorname{Grad}_{\boldsymbol{\xi}} \boldsymbol{X}^e(\boldsymbol{\xi}) = \frac{\partial}{\partial \boldsymbol{\xi}} \boldsymbol{X}^e(\boldsymbol{\xi}) = \sum_{I=1}^{n} \boldsymbol{X}_I \otimes \frac{\partial N_I(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} \tag{3.31}$$

which has the important purpose of expressing the derivatives of the shape functions N_I with respect to $\boldsymbol{\xi}$ or $\boldsymbol{X}^e$

$$\frac{\partial N_I}{\partial \boldsymbol{\xi}} = \frac{\partial \boldsymbol{X}^e}{\partial \boldsymbol{\xi}} \frac{\partial N_I}{\partial \boldsymbol{X}^e} = \boldsymbol{J}^e \frac{\partial N_I}{\partial \boldsymbol{X}^e} , \tag{3.32}$$

$$\frac{\partial N_I}{\partial \boldsymbol{X}^e} = \frac{\partial \boldsymbol{\xi}}{\partial \boldsymbol{X}^e} \frac{\partial N_I}{\partial \boldsymbol{\xi}} = (\boldsymbol{J}^e)^{-1} \frac{\partial N_I}{\partial \boldsymbol{\xi}} . \tag{3.33}$$

So far, all relations have been formulated in terms of the material description. However, they can easily rewritten in spatial coordinates by simply replacing $\boldsymbol{X}_I$ with $\boldsymbol{x}_I$ and $\boldsymbol{X}^e$ with $\boldsymbol{x}^e$ such that

$$\boldsymbol{x}^e(\boldsymbol{\xi}) = \sum_{I=1}^{n} N_I(\boldsymbol{\xi})\, \boldsymbol{x}_I \quad \text{and} \quad \boldsymbol{J}_t^e = \operatorname{grad}_{\boldsymbol{\xi}} \boldsymbol{x}^e(\boldsymbol{\xi}) = \sum_{I=1}^{n} \boldsymbol{x}_I \otimes \frac{\partial N_I(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} . \tag{3.34}$$

It makes no difference whether the mapping onto the isoparametric reference element is carried out from the initial or the current configuration [171]. In this context, the element deformation gradient $\boldsymbol{F}^e$ is introduced

$$\boldsymbol{F}^e = \operatorname{Grad} \boldsymbol{x}^e = \frac{\partial \boldsymbol{x}^e}{\partial \boldsymbol{X}^e} = \sum_{I=1}^{n} \boldsymbol{x}_I \otimes \frac{\partial N_I}{\partial \boldsymbol{X}^e} = \sum_{I=1}^{n} \boldsymbol{x}_I \otimes \left[(\boldsymbol{J}^e)^{-1} \frac{\partial N_I}{\partial \boldsymbol{X}^e} \right] \tag{3.35}$$

which still is the important measure to relate quantities of the initial configuration to their counterparts in the current configuration. The different configurations are depicted in Figure 3.3. In order to transform quantities between these configurations, one can conclude that

$$\mathrm{d}\boldsymbol{x}^e = \boldsymbol{J}_t^e\,\mathrm{d}\boldsymbol{\xi}\,, \quad \mathrm{d}\boldsymbol{X}^e = \boldsymbol{J}^e\,\mathrm{d}\boldsymbol{\xi}\,, \quad \boldsymbol{F}^e = \boldsymbol{J}_t^e\,(\boldsymbol{J}^e)^{-1} \tag{3.36}$$

holds. For any admissible deformation, it also follows that the determinants $j^e = \det \boldsymbol{J}_t^e > 0$, $J^e = \det \boldsymbol{J}^e > 0$ and $\det \boldsymbol{F}^e > 0$ have to be greater zero.

3.2.2 Discretization of the weak forms

This section focuses on the discretization of the weak formulation of the mechanical, the thermal and the electric field. The primary variables, their virtual counterparts and the corresponding gradients of each field are discretized. For the mechanical field, the finite element approximation of the displacement vector $\boldsymbol{u}$ to the exact solution reads

$$\boldsymbol{u}(\boldsymbol{\xi}) = \sum_{I=1}^{n} N_I(\boldsymbol{\xi})\,\boldsymbol{u}_I = \boldsymbol{N}^e(\boldsymbol{\xi})\,\boldsymbol{u}^e\,, \qquad \mathrm{Grad}\,\boldsymbol{u} = \sum_{I=1}^{n} \boldsymbol{u}_I \otimes \frac{\partial\,N_I(\boldsymbol{\xi})}{\partial\,\boldsymbol{X}^e}\,, \tag{3.37}$$

$$\delta\boldsymbol{u}(\boldsymbol{\xi}) = \sum_{I=1}^{n} N_I(\boldsymbol{\xi})\,\delta\boldsymbol{u}_I = \boldsymbol{N}^e(\boldsymbol{\xi})\,\delta\boldsymbol{u}^e\,, \quad \mathrm{Grad}\,\delta\boldsymbol{u} = \sum_{I=1}^{n} \delta\boldsymbol{u}_I \otimes \frac{\partial\,N_I(\boldsymbol{\xi})}{\partial\,\boldsymbol{X}^e}\,. \tag{3.38}$$

For the thermal and the electric field, however, it is necessary to discretize a scalar – either the temperature or the electric potential.

$$\phi(\boldsymbol{\xi}) = \sum_{I=1}^{n} N_I(\boldsymbol{\xi})\,\phi_I = \boldsymbol{N}^e(\boldsymbol{\xi})\,\boldsymbol{\phi}^e\,, \quad \delta\phi(\boldsymbol{\xi}) = \sum_{I=1}^{n} N_I(\boldsymbol{\xi})\,\delta\phi_I = \boldsymbol{N}^e(\boldsymbol{\xi})\,\delta\boldsymbol{\phi}^e\,, \tag{3.39}$$

$$\mathrm{Grad}\,\phi = \sum_{I=1}^{n} \frac{\partial\,N_I(\boldsymbol{\xi})}{\partial\,\boldsymbol{X}^e}\phi_I = \boldsymbol{B}^e(\boldsymbol{\xi})\boldsymbol{\phi}^e\,, \;\; \mathrm{Grad}\,\delta\phi = \sum_{I=1}^{n} \frac{\partial\,N_I(\boldsymbol{\xi})}{\partial\,\boldsymbol{X}^e}\delta\phi_I = \boldsymbol{B}^e(\boldsymbol{\xi})\delta\boldsymbol{\phi}^e\,. \tag{3.40}$$

Therein, the scalar quantity ϕ can be replaced by the temperature Θ or by the electric potential φ, respectively.

3.2.2.1 Mechanical field

In a first step, the internal virtual work in material description is written as

$$\delta\mathcal{W}_{\mathrm{int}} = \int_{\Omega_0} \underset{\sim}{\mathrm{S}} \cdot \delta\underset{\sim}{\mathrm{E}}\,\mathrm{d}V \tag{3.41}$$

where the virtual Green-Lagrange strain tensor reads [84]

$$\delta\underset{\sim}{\mathrm{E}} = \frac{1}{2}\left(\underset{\sim}{\mathrm{F}}^T\,\mathrm{Grad}\,\delta\vec{u} + \mathrm{Grad}^T\,\delta\vec{u}\,\underset{\sim}{\mathrm{F}}\right)\,. \tag{3.42}$$

Using the expression of the discretized deformation gradient Eq. (3.35), and for the gradient of the virtual displacements Eq. (3.38), the virtual strains $\delta\underset{\sim}{E}$ for one element can be formulated as

$$\delta\boldsymbol{E}^e = \frac{1}{2}\sum_{I=1}^{n}\left[\boldsymbol{F}^{e\,T}\left(\delta\boldsymbol{u}_I \otimes \frac{\partial\,N_I(\boldsymbol{\xi})}{\partial\,\boldsymbol{X}^e}\right) + \left(\frac{\partial\,N_I(\boldsymbol{\xi})}{\partial\,\boldsymbol{X}^e} \otimes \delta\boldsymbol{u}_I\right)\boldsymbol{F}^e\right]. \tag{3.43}$$

Since the strain tensor is symmetric, it has six instead of nine independent components. This allows to use the so-called *Voigt notation* where the components of the strain tensor are written as $\delta\boldsymbol{E} = (\delta E_{11},\,\delta E_{22},\,\delta E_{33},\,2\delta E_{12},\,2\delta E_{23},\,2\delta E_{13})^T$ resulting in a 6×1 vector. Based on this notation and following [171], it can be shown that

$$\delta\boldsymbol{E}^e = \sum_{I=1}^{n}\boldsymbol{B}_{\mathrm{L},I}\,\delta\boldsymbol{u}_I = \boldsymbol{B}_{\mathrm{L}}^e\,\delta\boldsymbol{u}^e\,,\ \ \boldsymbol{B}_{\mathrm{L}}^e = \begin{bmatrix}\boldsymbol{B}_{\mathrm{L},1}\ \boldsymbol{B}_{\mathrm{L},2}\ ...\ \boldsymbol{B}_{\mathrm{L},n}^e\end{bmatrix}\,,\ \ \delta\boldsymbol{u}^e = \begin{pmatrix}\delta\boldsymbol{u}_1\\ \delta\boldsymbol{u}_2\\ \vdots\\ \delta\boldsymbol{u}_n\end{pmatrix},$$
$$\tag{3.44}$$

where the *strain-displacement matrix* $\boldsymbol{B}_{\mathrm{L},I}$ is introduced as

$$\boldsymbol{B}_{\mathrm{L},I} = \begin{bmatrix} F_{11}\,N_{I,X_1} & F_{21}\,N_{I,X_1} & F_{31}\,N_{I,X_1} \\ F_{12}\,N_{I,X_2} & F_{22}\,N_{I,X_2} & F_{32}\,N_{I,X_2} \\ F_{13}\,N_{I,X_3} & F_{23}\,N_{I,X_3} & F_{33}\,N_{I,X_3} \\ F_{11}\,N_{I,X_2} + F_{12}\,N_{I,X_1} & F_{21}\,N_{I,X_2} + F_{22}\,N_{I,X_1} & F_{31}\,N_{I,X_2} + F_{32}\,N_{I,X_1} \\ F_{12}\,N_{I,X_3} + F_{13}\,N_{I,X_2} & F_{22}\,N_{I,X_3} + F_{23}\,N_{I,X_2} & F_{32}\,N_{I,X_3} + F_{33}\,N_{I,X_2} \\ F_{11}\,N_{I,X_3} + F_{13}\,N_{I,X_1} & F_{21}\,N_{I,X_3} + F_{23}\,N_{I,X_1} & F_{31}\,N_{I,X_3} + F_{33}\,N_{I,X_1} \end{bmatrix}. \tag{3.45}$$

Together with the element stress tensor $\boldsymbol{S}^e$ in Voigt notation, one can conclude that

$$\delta\mathcal{W}_{\mathrm{int}} = \int\limits_{\Omega_0}\delta\underset{\sim}{E}\cdot\underset{\sim}{S}\,\mathrm{d}\Omega \approx \bigcup_{e=1}^{n_u}\sum_{I=1}^{n}\delta\boldsymbol{u}_I^T\int\limits_{\Omega^e}\boldsymbol{B}_{\mathrm{L},I}^T\,\boldsymbol{S}^e\,\mathrm{d}\Omega = \bigcup_{e=1}^{n_e}\delta\boldsymbol{u}^{e\,T}\int\limits_{\Omega^e}\boldsymbol{B}_{\mathrm{L}}^{e\,T}\,\boldsymbol{S}^e\,\mathrm{d}\Omega \tag{3.46}$$

approximates the internal virtual mechanical work with n_u elements. The external virtual mechanical works needs to be discretized as well. By a multiplication with the virtual displacements and an integration over the volume, this can be accomplished in a straightforward manner:

$$\delta\mathcal{W}_{\mathrm{ext}} = \int\limits_{\partial\,\Omega_0^\sigma}\vec{\bar{T}}\cdot\delta\vec{u}\,\mathrm{d}A + \int\limits_{\Omega_0}\rho_0\,\vec{b}\cdot\delta\vec{u}\,\mathrm{d}V \approx \bigcup_{e=1}^{n_u}\delta\boldsymbol{u}^{e\,T}\left(\int\limits_{\partial\,\Omega^e}\boldsymbol{N}^{e\,T}\bar{\boldsymbol{t}}^e\,\mathrm{d}\Gamma + \int\limits_{\Omega^e}\boldsymbol{N}^{e\,T}\boldsymbol{b}^e\,\mathrm{d}\Omega\right), \tag{3.47}$$

where $\boldsymbol{b}^e$ is the element body force vector and $\bar{\boldsymbol{t}}^e$ the element traction vector. At equilibrium, it was stated that $\delta\mathcal{W}_{\text{int}} - \delta\mathcal{W}_{\text{ext}} = 0$ holds and – considering that the virtual displacements are arbitrary – it follows that

$$
\begin{aligned}
\mathbf{G}_{\mathrm{M}}(\mathbf{u}) &= \bigcup_{e=1}^{n_{\mathrm{u}}} \left(\int_{\Omega^e} \boldsymbol{B}_{\mathrm{L}}^{e\,T}\, \boldsymbol{S}^e \, \mathrm{d}\Omega - \int_{\Omega^e} \boldsymbol{N}^{e\,T} \boldsymbol{b}^e \, \mathrm{d}\Omega - \int_{\partial\Omega^e} \boldsymbol{N}^{e\,T} \bar{\boldsymbol{t}}^e \, \mathrm{d}\Gamma \right) = \mathbf{0} \\
&= \bigcup_{e=1}^{n_{\mathrm{u}}} \left(\boldsymbol{w}_{\mathrm{M}}^e(\boldsymbol{u}^e) - \boldsymbol{p}_{\mathrm{M}}^e \right) = \mathbf{w}_{\mathrm{M}}(\mathbf{u}) - \mathbf{p}_{\mathrm{M}} = \mathbf{0} \, .
\end{aligned}
\tag{3.48}
$$

This equation needs to be linearized with respect to the displacement vector, since there is a nonlinear dependency between $\mathbf{w}_{\mathrm{M}}$ and $\mathbf{u}$. To this end, the linearized weak formulation given by Eq. (3.13) is taken under consideration. Following [171], the first contribution can be approximated for one element by

$$
\begin{aligned}
\int_{\Omega_0} \operatorname{Grad} \delta\vec{u} \cdot \operatorname{Grad} \Delta\vec{u}\, \mathrm{S}^e \, \mathrm{d}V &\approx \sum_{I=1}^{n}\sum_{J=1}^{n} \int_{\Omega^e} \left(\delta\boldsymbol{u}_I \otimes \frac{\partial N_I}{\partial \boldsymbol{X}^e} \right)^T \left(\Delta\boldsymbol{u}_J \otimes \frac{\partial N_J}{\partial \boldsymbol{X}^e} \right) \boldsymbol{S} \, \mathrm{d}\Omega \\
&= \sum_{I=1}^{n}\sum_{J=1}^{n} \delta\boldsymbol{u}^{e\,T} \int_{\Omega^e} \boldsymbol{G}_{\mathrm{S},I}^T\, \hat{\boldsymbol{S}}^e\, \boldsymbol{G}_{\mathrm{S},J} \, \mathrm{d}\Omega \, \Delta\boldsymbol{u}^e \\
&= \delta\boldsymbol{u}^{e\,T} \int_{\Omega^e} \boldsymbol{G}_{\mathrm{S}}^{e\,T}\, \hat{\boldsymbol{S}}^e\, \boldsymbol{G}_{\mathrm{S}}^e \, \mathrm{d}\Omega\, \Delta\boldsymbol{u}^e = \delta\boldsymbol{u}^{e\,T}\, \boldsymbol{K}_{\mathrm{S}}^e\, \Delta\boldsymbol{u}^e \, .
\end{aligned}
\tag{3.49}
$$

Therein, the matrices

$$
\begin{aligned}
\boldsymbol{G}_{\mathrm{S},I}^T &= \begin{bmatrix} N_{I,X_1} & N_{I,X_2} & N_{I,X_3} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & N_{I,X_1} & N_{I,X_2} & N_{I,X_3} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & N_{I,X_1} & N_{I,X_2} & N_{I,X_3} \end{bmatrix}, \\
\hat{\boldsymbol{S}}^e &= \begin{bmatrix} \boldsymbol{S} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{S} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{S} \end{bmatrix} \quad \text{and} \quad \boldsymbol{S} = \begin{bmatrix} S_{11} & S_{12} & S_{13} \\ S_{21} & S_{22} & S_{23} \\ S_{31} & S_{32} & S_{33} \end{bmatrix}
\end{aligned}
\tag{3.50}
$$

are introduced where the S_{ij} are the components of the second Piola-Kirchhoff stress tensor and the notation N_{I,X_i}, $i = 1, 2, 3$ denotes the derivative of the I-th shape function with respect to X_i.

The matrix $\boldsymbol{K}_{\mathrm{S}}^e$ in Eq. (3.49) is only related to the current stress, which is why it is often termed as *initial stress matrix*. For the second part of the linearization in Eq. (3.13), one obtains

$$
\begin{aligned}
\int_{\Omega_0} \delta\underset{\sim}{\mathrm{E}} \cdot \underset{\sim}{\mathcal{C}} \, \Delta\underset{\sim}{\mathrm{E}} \, \mathrm{d}V &\approx \sum_{I=1}^{n}\sum_{J=1}^{n} \int_{\Omega^e} \left(\boldsymbol{B}_{\mathrm{L},I}\, \delta\boldsymbol{u}_I \right)^T \boldsymbol{C}^e\, \boldsymbol{B}_{\mathrm{L},J}\, \Delta\boldsymbol{u}_J \, \mathrm{d}\Omega \\
&= \delta\boldsymbol{u}^{e\,T} \int_{\Omega^e} \boldsymbol{B}_{\mathrm{L}}^{e\,T} \boldsymbol{C}^e \boldsymbol{B}_{\mathrm{L}}^e \, \mathrm{d}\Omega\, \Delta\boldsymbol{u}^e \, .
\end{aligned}
\tag{3.51}
$$

Therein, the tensor $\boldsymbol{C}^e$ is the incremental material tensor in Voigt notation. Due to the symmetry of $\boldsymbol{C}^e$, it can be viewed as a 6×6 matrix in the three-dimensional case. Summarizing Eqns. (3.49)-(3.51) delivers for one element

$$\delta\boldsymbol{u}^{e\,T} \int\limits_{\Omega^e} \boldsymbol{G}_{\mathrm{S}}^{e\,T} \hat{\boldsymbol{S}}^e \boldsymbol{G}_{\mathrm{S}}^e + \boldsymbol{B}_{\mathrm{L}}^{e\,T} \boldsymbol{C}^e \boldsymbol{B}_{\mathrm{L}}^e \,\mathrm{d}\Omega\, \Delta\boldsymbol{u}^e = \delta\boldsymbol{u}^{e\,T} \left(\boldsymbol{K}_{\mathrm{S}}^e + \boldsymbol{K}_{\mathrm{C}}^e\right) \Delta\boldsymbol{u}^e = \delta\boldsymbol{u}^{e\,T} \boldsymbol{K}_{\mathrm{T,M}}^e \Delta\boldsymbol{u}^e$$

(3.52)

where $\mathbf{K}_{\mathrm{T,M}} = \bigcup_{e=1}^{n_{\mathrm{u}}} \boldsymbol{K}_{\mathrm{T,M}}^e$ is the tangential stiffness matrix of the mechanical field. So far, only the weak form related to the reference configuration has been discretized. In order to obtain a finite element formulation of the weak form in spatial description, the discretization can be carried out in the same fashion. A detailed derivation can be found in [13, 171], for instance, and is not discussed here.

3.2.2.2 Thermal field

In the following, the weak form of the equation of heat conduction, see Eq.(3.19), is discretized. Since the thermal field is assumed to be nonlinear, an expression for the tangential stiffness matrix is required. A comprehensive discussion on the discretization of the nonlinear heat equation was done by QUINT ET AL. [143].

For one element, using the discretizations given by Eqns. (3.39)–(3.40) leads to

$$\mathcal{G}_{\Theta} \approx \int\limits_{\Omega^e} \rho_0\, c_{\mathrm{p}}(\Theta) \left(\boldsymbol{N}^e \dot{\boldsymbol{\Theta}}^e\right) \left(\boldsymbol{N}^e \delta\boldsymbol{\Theta}^e\right) + \left(\boldsymbol{B}^e \delta\boldsymbol{\Theta}^e\right)^T \boldsymbol{\Lambda}_{\Theta}^e(\Theta) \left(\boldsymbol{B}^e \boldsymbol{\Theta}^e\right) \mathrm{d}\Omega -$$
$$\int\limits_{\Omega^e} \left[\gamma(\Theta) + D_{\mathrm{int}}(\Theta) + \rho_0\, R_{\varphi}(\Theta)\right] \boldsymbol{N}^e \delta\boldsymbol{\Theta}^e \,\mathrm{d}\Omega - \int\limits_{\partial\Omega^e} \bar{Q}(\Theta)\, \boldsymbol{N}^e \delta\boldsymbol{\Theta}^e \,\mathrm{d}\Gamma = 0\,.$$

(3.53)

This can be rearranged for the whole domain using n_{Θ} finite elements to

$$\bigcup_{e=1}^{n_{\Theta}} \delta\boldsymbol{\Theta}^{e\,T} \left(\int\limits_{\Omega^e} \rho_0\, c_{\mathrm{p}}(\Theta)\, \boldsymbol{N}^{e\,T} \boldsymbol{N}^e \,\mathrm{d}\Omega\, \dot{\boldsymbol{\Theta}}^e + \int\limits_{\Omega^e} \boldsymbol{B}^{e\,T} \boldsymbol{\Lambda}_{\Theta}^e(\Theta) \boldsymbol{B}^e \,\mathrm{d}\Omega\, \boldsymbol{\Theta}^e \right) =$$
$$\bigcup_{e=1}^{n_{\Theta}} \delta\boldsymbol{\Theta}^{e\,T} \left(\int\limits_{\partial\Omega^e} \bar{Q}(\Theta)\, \boldsymbol{N}^{e\,T} \,\mathrm{d}\Gamma + \int\limits_{\Omega^e} \left[\gamma(\Theta) + D_{\mathrm{int}}(\Theta) + \rho_0\, R_{\varphi}(\Theta)\right] \boldsymbol{N}^{e\,T}\mathrm{d}\Omega \right)$$

(3.54)

or in a very compact notation

$$\mathbf{M}_{\Theta}(\Theta)\, \dot{\boldsymbol{\Theta}} + \mathbf{K}_{\Theta}(\Theta)\, \boldsymbol{\Theta} = \mathbf{p}_{\Theta}(\Theta)\,.$$

(3.55)

Therein, the expression $\mathbf{M}_{\Theta} = \bigcup_{e=1}^{n_{\Theta}} \boldsymbol{M}_{\Theta}^e$ denotes the *thermal mass matrix*, $\mathbf{K}_{\Theta} = \bigcup_{e=1}^{n_{\Theta}} \boldsymbol{K}_{\Theta}^e$ is the *thermal stiffness matrix* and $\mathbf{p}_{\Theta} = \bigcup_{e=1}^{n_{\Theta}} \boldsymbol{p}_{\Theta}^e$ is the *thermal load vector* related to one element. Since all of these terms are non-linearly dependent on

the temperature, a linearization is required to set-up a tangential stiffness matrix for the thermal field. This can be achieved by means of a discretization of the incremental quantities $\Delta\Theta$ and $\mathrm{Grad}\,\Delta\Theta$ in the linearized variational formulation of the thermal field given by Eq. (3.22). This results in

$$
\begin{aligned}
\mathrm{D}_\Theta \mathcal{G}_\Theta \cdot \Delta\Theta \approx\ & \delta\boldsymbol{\Theta}^{e\,T} \int_{\Omega^e} \rho_0 \left(\frac{\partial\,c_{\mathrm{p}}}{\partial\Theta}\, \boldsymbol{N}^e \frac{\boldsymbol{\Theta}^e - \boldsymbol{\Theta}_n^e}{\Delta t} + c_{\mathrm{p}} \frac{1}{\Delta t} \right) \boldsymbol{N}^{e\,T} \boldsymbol{N}^e \,\mathrm{d}\Omega\, \Delta\boldsymbol{\Theta}^e\ + \\
& \delta\boldsymbol{\Theta}^{e\,T} \int_{\Omega^e} \boldsymbol{B}^{e\,T} \boldsymbol{\Lambda}_\Theta^e\, \boldsymbol{B}^e + \boldsymbol{B}^{e\,T} \frac{\partial\,\boldsymbol{\Lambda}_\Theta^e}{\partial\Theta} \boldsymbol{B}^e \boldsymbol{\Theta}^e \boldsymbol{N}^e \,\mathrm{d}\Omega\, \Delta\boldsymbol{\Theta}^e\ - \\
& \delta\boldsymbol{\Theta}^{e\,T} \int_{\Omega^e} \left(\frac{\partial\,\gamma}{\partial\Theta} + \frac{\partial\,D_{\mathrm{int}}}{\partial\Theta} + \rho_0 \frac{\partial\,R_\varphi}{\partial\Theta} \right) \boldsymbol{N}^{e\,T} \boldsymbol{N}^e \,\mathrm{d}\Omega\, \Delta\boldsymbol{\Theta}^e\ - \\
& \delta\boldsymbol{\Theta}^{e\,T} \int_{\partial\Omega^e} \frac{\partial\,\bar{Q}}{\partial\Theta}\, \boldsymbol{N}^{e\,T} \boldsymbol{N}^e \,\mathrm{d}\Gamma\, \Delta\boldsymbol{\Theta}^e\,.
\end{aligned}
\tag{3.56}
$$

For the whole domain, this finally leads to

$$
\mathrm{D}_\Theta \mathcal{G}_\Theta \Delta\Theta \approx \bigcup_{e=1}^{n_\Theta} \delta\boldsymbol{\Theta}^{e\,T} \boldsymbol{K}_{\mathrm{T},\Theta}^e\, \Delta\boldsymbol{\Theta}^e\,,
\tag{3.57}
$$

where $\mathbf{K}_{\mathrm{T},\Theta} = \bigcup_{e=1}^{n_\Theta} \boldsymbol{K}_{\mathrm{T},\Theta}^e$ is the tangential stiffness matrix of the thermal field. Special attention shall be placed on the second contribution of Eq. (3.56) which arises for a temperature-dependence of the heat conduction coefficient. In this case, the contribution involves the calculation of the product $\boldsymbol{B}^e \boldsymbol{\Theta}^e \boldsymbol{N}^e$, leading to a non-symmetric tangential stiffness matrix. Further reading on the numerical treatment of the nonlinear heat conduction equation can be found in [143].

3.2.2.3 Electric field

In order to discretize the weak formulation of the electric field, the interpolation given by Eqns. (3.39)–(3.40) is used. The first part of the weak form (3.25) is discretized for one element as

$$
\int_{\Omega_0} \mathrm{Grad}\,\delta\varphi \cdot \underset{\sim}{\Lambda}_\varphi \,\mathrm{Grad}\,\varphi \,\mathrm{d}\Omega \approx \delta\boldsymbol{\varphi}^{e\,T} \int_{\Omega^e} \boldsymbol{B}^{e\,T} \boldsymbol{\Lambda}_\varphi^e \boldsymbol{B}^e \,\mathrm{d}\Omega\, \boldsymbol{\varphi}^e = \delta\boldsymbol{\varphi}^{e\,T} \boldsymbol{K}_\varphi^e\, \boldsymbol{\varphi}^e\,,
\tag{3.58}
$$

where $\boldsymbol{K}_\varphi^e$ is the element stiffness matrix of the electric field. The element load vector $\boldsymbol{p}_\varphi^e$ of the electric field follows from the second part of Eq. (3.25) and reads

$$
\int_{\partial\Omega_0} \bar{J}\,\delta\varphi \,\mathrm{d}A \approx \delta\boldsymbol{\varphi}^{e\,T} \int_{\partial\Omega^e} \bar{J}\,\boldsymbol{N}^{e\,T} \,\mathrm{d}\Gamma = \delta\boldsymbol{\varphi}^{e\,T} \boldsymbol{p}_\varphi^e\,.
\tag{3.59}
$$

For the whole domain with n_φ finite elements, one obtains

$$
\bigcup_{e=1}^{n_\varphi} \delta\boldsymbol{\varphi}^{e\,T} \left(\int_{\Omega^e} \boldsymbol{B}^{e\,T} \boldsymbol{\Lambda}_\varphi^e \boldsymbol{B}^e \,\mathrm{d}\Omega\, \boldsymbol{\varphi}^e - \int_{\partial\Omega^e} \bar{J}\,\boldsymbol{N}^{e\,T} \,\mathrm{d}\Gamma \right) = \boldsymbol{0}\,.
\tag{3.60}
$$

In a compact notation and taking into account that the virtual electric potential is arbitrary, this can be written as

$$\mathbf{K}_\varphi \, \boldsymbol{\varphi} = \mathbf{p}_\varphi \,, \tag{3.61}$$

where $\mathbf{K}_\varphi = \bigcup_{e=1}^{n_\varphi} \boldsymbol{K}_\varphi^e$ is the *electric stiffness matrix* and $\mathbf{p}_\varphi = \bigcup_{e=1}^{n_e} \boldsymbol{p}_\varphi^e$ the *electric load vector*.

As mentioned in Eq. (3.26), the current density $\bar{J}$ on the boundary can be linear in the electric potential. In this case, a constant contribution

$$\boldsymbol{K}_{\bar{J}}^e = - \int\limits_{\partial \Omega^e} \lambda_{\mathrm{ec}} \, \boldsymbol{N}^{e\,T} \boldsymbol{N}^e \, \mathrm{d}\Gamma \tag{3.62}$$

needs to be added to the (element) stiffness matrix.

3.2.2.4 Remarks on the FE formulation of the coupled problem

In view of the further numerical solution, it might be of advantage to write the equations in a more compact fashion. To this end, the FE equations of the mechanical field Eq. (3.48), the thermal field Eq. (3.55) and the electric field Eq. (3.61) are written as follows:

$$\mathbf{G}_{\mathrm{M}} = \bigcup_{e=1}^{n_e} \left(\boldsymbol{w}_{\mathrm{M}}^e(\boldsymbol{u}^e) - \boldsymbol{p}_{\mathrm{M}}^e \right) = \mathbf{0} \,, \tag{3.63}$$

$$\mathbf{G}_{\Theta} = \bigcup_{e=1}^{n_e} \left(\boldsymbol{M}_{\Theta}^e(\Theta) \, \dot{\boldsymbol{\Theta}}^e + \boldsymbol{K}_{\Theta}^e(\Theta) \, \boldsymbol{\Theta}^e - \boldsymbol{p}_{\Theta}^e(\Theta) \right) = \mathbf{0} \,, \tag{3.64}$$

$$\mathbf{G}_{\varphi} = \bigcup_{e=1}^{n_e} \left(\boldsymbol{K}_{\varphi}^e \, \boldsymbol{\varphi}^e - \boldsymbol{p}_{\varphi}^e \right) = \mathbf{0} \tag{3.65}$$

A rigorous partitioned solution approach of the electro-thermo-mechanical problems is considered. This consequently means that also the FE formulation is given in a separated manner and that every field can be solved individually by keeping the discrete solution vectors of the other fields constant.

In addition, a remark on the size of the element matrices and vectors shall be given. The size depends on the physical degree of freedom n_{phy} of the problem and on the number of nodes n_{n} of the element. The mass and stiffness matrices have the dimensions $(n_{\mathrm{phy}} \cdot n_{\mathrm{n}}) \times (n_{\mathrm{phy}} \cdot n_{\mathrm{n}})$, and the force vectors are of the dimension $(n_{\mathrm{phy}} \cdot n_{\mathrm{n}})$. The physical degree of freedom for the mechanical field is $n_{\mathrm{phy}} = 3$ for the three-dimensional case – and for the thermal as well as for the electric field, it is $n_{\mathrm{phy}} = 1$. For a linear hexahedral element with $n_{\mathrm{n}} = 8$ nodes, one can conclude that $\boldsymbol{K}_{\mathrm{M}}^e \in \mathbb{R}^{24 \times 24}$ and $\boldsymbol{K}_{\Theta}^e \in \mathbb{R}^{8 \times 8}$. However, the size of the matrices specifically depends on the polynomial degree p which is chosen for the shape functions. If p is increased, the size of the matrices increases as well. This issue is discussed in more detail in the following Section 3.2.3, where a brief summary of high-order finite elements is given.

3.2.2.5 *Temporal discretization*

The further numerical solution procedure requires a temporal discretization of the coupled initial boundary value problem given by Eqns. (3.63)–(3.65). Because the mechanical and the electric fields are assumed to be quasi-stationary problems, only the heat equation needs to be discretized in time. To this end, the global finite element equation of the thermal field (3.64) is rewritten as

$$\dot{\boldsymbol{\Theta}}(t) = \mathbf{M}^{-1}\left[\mathbf{p}_{\Theta} - \mathbf{K}_{\Theta}\,\boldsymbol{\Theta}(t)\right] = \mathbf{f}(\boldsymbol{\Theta}, t) \quad \text{with} \quad \boldsymbol{\Theta}(t_0) = \boldsymbol{\Theta}_0\,, \tag{3.66}$$

and this first-order differential equation is discretized by means of the *Euler method* to

$$\dot{\boldsymbol{\Theta}}(t) \approx \frac{\boldsymbol{\Theta}_{n+1} - \boldsymbol{\Theta}_n}{\Delta t} = \mathbf{f}(\boldsymbol{\Theta}, t) + \mathcal{O}(\Delta t^2)\,. \tag{3.67}$$

Therein, $\mathcal{O}(\Delta t^2)$ denotes the error of the method, Δt is the (time-) step size, and the subscript n refers to the n-th time-step. Neglecting the error, the function $\mathbf{f}(\boldsymbol{\Theta}, t)$ can either be evaluated at the time t_n

$$\boldsymbol{\Theta}_{n+1} = \boldsymbol{\Theta}_n + \Delta t\,\mathbf{f}(\boldsymbol{\Theta}_n, t_n)\,, \tag{3.68}$$

which leads to the *explicit Euler method* or at t_{n+1}

$$\boldsymbol{\Theta}_{n+1} = \boldsymbol{\Theta}_n + \Delta t\,\mathbf{f}(\boldsymbol{\Theta}_{n+1}, t_{n+1})\,, \tag{3.69}$$

leading to the *implicit Euler method*[4]. The significant difference between both possibilities is that a system of equations needs to be solved for the implicit case, whereas the explicit case allows a direct computation of the new value $\boldsymbol{\Theta}$. This is due to the fact that the function $\mathbf{f}(\boldsymbol{\Theta}_n, t_n)$ is simply evaluated for known values $\boldsymbol{\Theta}_n$. Since the implicit case is known to be unconditionally stable [10], it is used here as the standard time integration method. Henceforth, all time-dependent quantities are discretized using the implicit Euler method.

3.2.3 High-order finite elements

In subsequent numerical examples, the mechanical, thermal and electric fields are solved with the in-house finite element code *AdhoC* [45]. Since this code utilizes the p-version of the FEM, some important principles shall be briefly considered here. In the early 90s, SZABÓ & BABUŠKA [157] conducted pioneering work on the high-order finite element formulation. During the past years, the p-FEM was studied in detail, in particular the performance and convergence properties [157, 158] and the extension to curved and thin-walled structures [43, 42], but also the application to nonlinear geometric and material behavior [46, 47, 44, 81]. Its applicability has also been demonstrated for compressible metal powder [82], which is of interest for the simulation of the FAST process. Today, it is known

---

[4]In the literature, the explicit case is often referred to as the *Euler-forward method*, and the implicit case is known as the *Euler-backward method*.

that the p-version has some meaningful advantages over the classical low-order finite element formulation (h-FEM). For example, it is more robust with respect to locking effects as well as to geometrical distortion. Additionally, the p-version can achieve a higher accuracy within the same computational effort.

The main distinguishing feature between the p- and the h-version is manifested in the order of the shape functions. The classical h-FEM typically involves the polynomial order $p = 1$ or $p = 2$. For the p-version, high-order polynomials are utilized for the shape functions. In this context, a distinction is made between *Lagrange* and *hierarchic shape functions*. In the one-dimensional case, the standard *Lagrange polynomials* are defined by the following formula

$$N_i^p(\xi) = \prod_{j=1, j\neq i}^{p+1} \frac{\xi - \xi_j}{\xi_i - \xi_j}\,, \tag{3.70}$$

where the nodes ξ_i can be identified by $N_i^p(\xi_j) = \delta_{ij}$ for the one-dimensional standard element with $\Omega^e = [-1,\, 1]$. From this definition, one can conclude that one-dimensional *Lagrange* elements have $n_{\mathrm{n}} = p + 1$ nodes – and the complete set of element shape functions changes every polynomial degree. The other concept is based on hierarchic shape functions and were introduced by SZABÓ & BABUŠKA [157] to

$$N_1(\xi) = \frac{1}{2}(1 - \xi)\,, \tag{3.71}$$

$$N_2(\xi) = \frac{1}{2}(1 + \xi)\,, \tag{3.72}$$

$$N_i(\xi) = \phi_{i-1}(\xi)\,, \quad i = 3, 4, ..., p + 1\,. \tag{3.73}$$

Here, N_1 and N_2 are referred to as the *nodal shape functions*, and the N_i for $i > 2$ are known as the *internal shape functions* or *bubble modes*. They can be calculated from

$$\phi_j(\xi) = \sqrt{\frac{2j-1}{2}} \int\limits_{-1}^{\xi} L_{j-1}(x)\,\mathrm{d}x = \frac{1}{\sqrt{4j-2}}\left[L_j(\xi) - L_{j-2}(\xi)\right]\,, \quad j > 1\,, \tag{3.74}$$

where the $L_j(\xi)$ are the orthogonal *Legendre polynomials*[5]. In contrast to the Lagrange polynomials, the application of hierarchic shape functions leads to $N_i(1) = N_i(-1) = 0$ for $i > 2$ and, further, hierarchic elements have no internal nodes. Moreover, all shape functions of polynomial order $p - 1$ are contained in the shape function of order p. This allows a very efficient and elegant computer implementation since only one new shape function needs to be added for every new polynomial degree and since the stiffness matrix becomes hierarchic.

---

[5]Legendre polynomials can be evaluated by means of the so-called *Rodriguez* formula:

$$L_n(x) = \frac{1}{2^n n!} \frac{\mathrm{d}^n}{\mathrm{d}x^n}(x^2 - 1)^n\,, \quad n = 0, 1, 2, ...$$

After the one-dimensional case, an extension to the three-dimensional case for hexahedral elements is given[6]. Based on the standard hexahedral element, see Figure 3.4, the following types of shape functions are introduced [157]:

- *Nodal modes* are the standard trilinear shape functions, given by

$$N_{1,1,1}^{N_i}(\boldsymbol{\xi}) = \frac{1}{8}(1 + \xi_i\,\xi)(1 + \eta_i\,\eta)(1 + \zeta_i\,\zeta)\,, \tag{3.75}$$

  where $\boldsymbol{\xi}_i$ is the local coordinate of the i-th node.

- *Edge modes* are used to describe each of the twelve edges. For example, the shape function for first edge E_1, see Figure 3.4, reads

$$N_{i,1,1}^{E_1}(\boldsymbol{\xi}) = \frac{1}{4}(1 - \eta)(1 - \zeta)\phi_i(\xi)\,, \quad i > 1\,. \tag{3.76}$$

- *Face modes* are defined to describe every single face. For first face F_1 in Figure 3.4, one obtains

$$N_{i,j,1}^{F_1}(\boldsymbol{\xi}) = \frac{1}{2}(1 - \zeta)\phi_i(\xi)\phi_j(\eta)\,, \quad i,j > 1\,. \tag{3.77}$$

- *Internal modes* are locally defined and vanish at the faces. They read

$$N_{i,j,k}^{int}(\boldsymbol{\xi}) = \phi_i(\xi)\phi_j(\eta)\phi_k(\zeta)\,, \quad i,j,k > 1\,. \tag{3.78}$$

In the previous classification, the indices i, j, k are the polynomial degrees in the corresponding local direction ξ, η, ζ. To be mentioned here, apart from the different modes, are the ansatz spaces out of which the polynomials are chosen. There are three different spaces implemented in AdhoC: the *trunk space*, the *tensor product space* and the *anisotropic tensor product space*. For further details and a comprehensive discussion of the ansatz spaces, the reader is referred to [157, 158].

Comparing the p-version with the h-version of the FEM, there is another distinguishing feature to be considered, namely the mapping procedure. The p-FEM reaches convergence by increasing the polynomial degree in a coarse mesh, whereas h-FEM increases the number of elements to achieve convergence with mesh refinement. In the latter case, the shape functions can be used to approximate the solution as well as the shape of the elements. This is the well-known isoparametric mapping concept, as outlined in Section 3.2.1. In order to accurately capture complex geometries and curved boundaries, a different mapping concept was developed for the p-version. Here, the so-called *blending function method* is employed, allowing for a treatment of arbitrary parametric edges and faces. The mapping function $\boldsymbol{Q}^e$ of the element reads

$$\boldsymbol{x} = \boldsymbol{Q}^e(\boldsymbol{\xi}) = \sum_i^8 N_{1,1,1}^{N_i}(\boldsymbol{\xi})\,\boldsymbol{X}_i + \sum_{i=1}^6 \boldsymbol{f}_i(\boldsymbol{\xi}) + \sum_{i=1}^{12} \boldsymbol{e}_i(\boldsymbol{\xi}) \tag{3.79}$$

---

[6]The two-dimensional case – hierarchic shape functions for quadrilaterals – is discussed in [158], for instance.
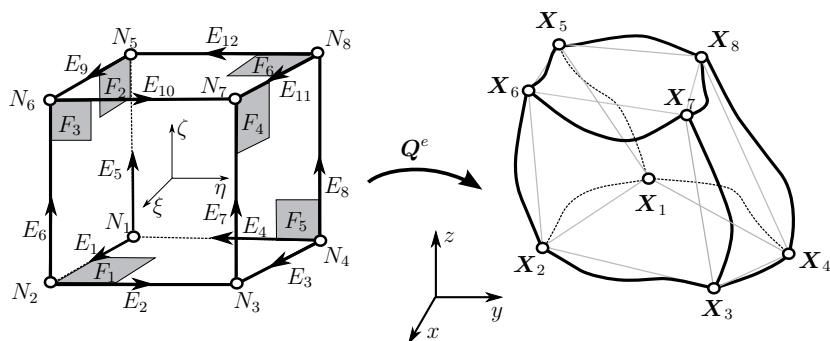
Figure 3.4: Node, face and edge definition of a high-order hexahedral element.

and consists of three parts: The first part denotes the standard isoparametric mapping of trilinear shape functions of the eight-noded hexahedrahel element. The second term is known as the *face blending term* and allows to map curved element faces. The third term is referred as *edge blending term*, taking curved edges into account. Expressions for $\boldsymbol{f}_i$ and $\boldsymbol{e}_i$ can be found in the literature, see e.g [19]. For further reading on three- and two-dimensional mapping concepts for high-order finite elements, a reference is made to [19, 43].

3.2.4   Numerical integration

So far, the computation of the integrals arising in the weak formulation has not yet been discussed. However, this is mandatory for setting up a discrete system of equations. Since polynomials have been applied for discretization, Gaussian quadrature is used as it allows to integrate polynomials exactly. In the one-dimensional case, this procedure reads

$$\int_{-1}^{1} f(\xi)\,\mathrm{d}\xi \approx \sum_{i=1}^{n_{\mathrm{gp}}} f(\xi_i)\,W_{\xi_i}\,. \tag{3.80}$$

Gaussian quadrature means that an integral can be approximated as follows: The integrand $f(\xi)$ is evaluated at all quadrature points ξ_i and multiplied with corresponding weighting factors W_{ξ_i}. The sum over all n_{gp} Gauss points then approximates the integral. Before the Gaussian quadrature can be applied, a transformation from global into local coordinates needs to be carried out. For a three-dimensional hexahedral element, one can deduce

$$\int_{\Omega^e} f(\boldsymbol{X}^e)\,\mathrm{d}\Omega = \int_{\Omega_*} f(\boldsymbol{\xi})\,J^e(\boldsymbol{\xi})\,\mathrm{d}\Omega = \int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1} f(\xi,\eta,\zeta)\,J^e(\xi,\eta,\zeta)\,\mathrm{d}\xi\,\mathrm{d}\eta\,\mathrm{d}\zeta \tag{3.81}$$

with $\Omega_* = [(-1,+1) \times (-1,+1) \times (-1,+1)]$ and $J^e = \det \boldsymbol{J}^e$. Afterwards, the Gaussian integration can be applied to approximate the triple integral as

$$\int\limits_{-1}^{1} \int\limits_{-1}^{1} \int\limits_{-1}^{1} f(\boldsymbol{\xi}) \, J^e(\boldsymbol{\xi}) \, \mathrm{d}\xi \, \mathrm{d}\eta \, \mathrm{d}\zeta \approx \sum_{i=1}^{n_{\mathrm{gp}_\xi}} \sum_{j=1}^{n_{\mathrm{gp}_\eta}} \sum_{k=1}^{n_{\mathrm{gp}_\zeta}} f(\xi_i, \eta_j, \zeta_k) \, J^e(\xi_i, \eta_j, \zeta_k) \, W_{\xi_i} W_{\eta_j} W_{\zeta_k} \,. \tag{3.82}$$

This procedure allows to approximate the integration of the finite element matrices and vectors. As a simple example, the element stiffness matrix $\boldsymbol{K}_\varphi^e$ of the electric field is taken under consideration. Applying Gaussian quadrature leads to

$$\boldsymbol{K}_\varphi^e = \int\limits_{\Omega^e} \boldsymbol{B}^{e\,T} \boldsymbol{\Lambda}_\varphi^e \boldsymbol{B}^e \, \mathrm{d}\Omega \approx \sum_{i=1}^{n_{\mathrm{gp}_\xi}} \sum_{j=1}^{n_{\mathrm{gp}_\eta}} \sum_{k=1}^{n_{\mathrm{gp}_\zeta}} \left[\boldsymbol{B}^{e\,T} \boldsymbol{\Lambda}_\varphi^e \boldsymbol{B}^e \right]\big|_{\xi_i, \eta_j, \zeta_k} \, J^e(\xi_i, \eta_j, \zeta_k) \, W_{\xi_i} W_{\eta_j} W_{\zeta_k} \,. \tag{3.83}$$

In this expression, the term $\left[\boldsymbol{B}^{e\,T} \boldsymbol{\Lambda}_\varphi^e \boldsymbol{B}^e \right]\big|_{\xi_i, \eta_j, \zeta_k}$ needs to be evaluated at the corresponding quadrature points. If this is done at all n_{gp} integration points $\boldsymbol{\xi}_{\mathrm{gp}}$, Eq. (3.83) can be rewritten as

$$\boldsymbol{K}_\varphi^e \approx \sum_{\mathrm{ig}=1}^{n_{\mathrm{gp}}} \boldsymbol{B}^{e\,T}(\boldsymbol{\xi}_{\mathrm{ig}}) \boldsymbol{\Lambda}_\varphi^e(\boldsymbol{\xi}_{\mathrm{ig}}) \boldsymbol{B}^e(\boldsymbol{\xi}_{\mathrm{ig}}) \, J^e(\boldsymbol{\xi}_{\mathrm{ig}}) \, W(\boldsymbol{\xi}_{\mathrm{ig}}) \,. \tag{3.84}$$

For the thermal and the mechanical field, there are more contributions to the stiffness matrix due to their nonlinear nature. In view of the p-version of the finite element method, the stiffness matrix computation is a crucial part. If the polynomial degree is increased, the computational effort increases drastically. A way to improve the numerical integration procedure for high-order finite elements can be found in [83, 115].

3.2.5 Solution of the discrete problem

For the solution of the discrete finite element equations, the Newton-Raphson procedure is applied to the nonlinear problem. Since the Newton method can be seen as the standard solution method in the FEM, there are several detailed descriptions to be found in the literature, see [29, 13, 171] for instance. As it is also discussed in Chapter 4 in the scope of treating coupled systems, it will first be introduced very briefly. A nonlinear system of equations is considered, denoted by $\mathbf{G}(\mathbf{y}) = \mathbf{0}$, for instance. The expansion into a Taylor-Series and neglecting all high-order terms leads to a linear system of equations that is solved repetitively. In other words, the objective is to compute a new increment $\Delta\mathbf{y}^i$ in every i-th Newton iteration such that the new solution is updated by

$$\mathbf{y}^{i+1} = \mathbf{y}^i + \Delta\mathbf{y}^i \,. \tag{3.85}$$

The increment can be computed by solving

$$\left.\frac{\partial \mathbf{G}}{\partial \mathbf{y}}\right|_{\mathbf{y}=\mathbf{y}^i} \Delta \mathbf{y}^i = -\mathbf{G}(\mathbf{y}^i)\,. \tag{3.86}$$

The Newton-Raphson iteration is terminated if a certain tolerance criteria, for example $\|\Delta \mathbf{y}^i\| < \epsilon$, is satisfied. In view of the electric, thermal and mechanical fields, the general vector $\mathbf{y}$ is replaced by the discrete vectors for the electric potential, the temperature or the displacements. Further, the derivative $\partial \mathbf{G}/\partial \mathbf{y}$ can be substituted by the tangential stiffness matrix, finally leading to

$$\mathbf{K}_{\mathrm{T,M}}|_{\mathbf{u}=\mathbf{u}^i} \Delta \mathbf{u}^i = -\mathbf{G}_{\mathrm{M}}(\mathbf{u}^i) \quad \rightarrow \quad \mathbf{u}^{i+1} = \mathbf{u}^i + \Delta \mathbf{u}^i\,, \tag{3.87}$$

for the mechanical field,

$$\mathbf{K}_{\mathrm{T,\Theta}}|_{\mathbf{\Theta}=\mathbf{\Theta}^i} \Delta \mathbf{\Theta}^i = -\mathbf{G}_{\mathbf{\Theta}}(\mathbf{\Theta}^i) \quad \rightarrow \quad \mathbf{\Theta}^{i+1} = \mathbf{\Theta}^i + \Delta \mathbf{\Theta}^i\,, \tag{3.88}$$

for the thermal field and

$$\mathbf{K}_{\mathrm{T,\varphi}}|_{\boldsymbol{\varphi}=\boldsymbol{\varphi}^i} \Delta \boldsymbol{\varphi}^i = -\mathbf{G}_{\varphi}(\boldsymbol{\varphi}^i) \quad \rightarrow \quad \boldsymbol{\varphi}^{i+1} = \boldsymbol{\varphi}^i + \Delta \boldsymbol{\varphi}^i\,, \tag{3.89}$$

for the electric field. As reported in [10, 171] for instance, the initial iterate of the Newton method can be an important feature in nonlinear finite element analysis. An initial iterate that is close to the solution leads to less iterations and therefore reduces computational effort. Instead of using the converged solution from the previous time-step, an extrapolated initial guess is used in [75, 141, 126], leading to promising results. Besides, there are several other possible approaches to improve the Newton-Raphson procedure. A detailed discussion of these methods is provided in the textbooks of KELLEY [101, 102].

Finally, a remark on the solution of the mechanical field shall be given. For example, in the computation of inelastic materials, the finite element formulation can be interpreted as a differential-algebraic equation (DAE) system [72]. Apart from the displacements $\mathbf{u}$, the internal variables $\mathbf{q}$ are the unknowns. Based on this interpretation, the discrete nonlinear DAE system for the mechanical field can be written as

$$\mathbf{G}_{\mathrm{M}}(\mathbf{u}, \mathbf{q}) = \begin{Bmatrix} \mathbf{l}(\mathbf{u}, \mathbf{q}) \\ \mathbf{g}(\mathbf{u}, \mathbf{q}) \end{Bmatrix} = \mathbf{0}\,, \tag{3.90}$$

where $\mathbf{l}(\mathbf{u}, \mathbf{q})$ denotes a local system on element level for the computation of the internal variables and $\mathbf{g}(\mathbf{u}, \mathbf{q})$ represents a system on global level for the global equilibrium. Due to this special structure of the system, HARTMANN [72, 73] and ELLSIEPEN & HARTMANN [48] proposed to use the *multilevel-Newton method* (MLNA) in which the local and global parts are treated separately. For a discussion of the MLNA in thermo-mechanically coupled processes, see e.g. [141, 147], and for electro-thermo-mechanical applications see [146], for instance. Also, the literature cited in the mentioned articles is able to provide further details.

3.3 Numerical solution of the surrounding fluid field

This section addresses the numerical treatment of the fluid field, henceforth denoted as Σ, which describes the surroundings of the body $\mathcal{B}$. Particularly, the focus is placed on a numerical solution procedure of the energy equation using the *finite volume method* (FVM). Since the conservation of energy in a participating medium is coupled with other conservation equations, this issue is also addressed in the following section. In addition, a brief introduction to the solution algorithm for the coupled fluid equations is given.

3.3.1 Basic concepts and finite volume method

One of the most popular methods used in *computational fluid dynamics* (CFD) is the *finite volume method* (FVM). It is well described in the literature, see textbooks such as [133, 62, 164, 138]. Here, a brief introduction of some basic concepts will therefore suffice. The formulation used in the following is strongly oriented towards the work of PATANKAR [133].

In contrast to the finite element method, the FVM utilizes a constant weight function $\eta = 1$ in order to make the mean residual vanish. To achieve this, a so-called *control-volume* (CV) formulation is set up. In this formulation, the domain of interest is spatially discretized by means of n_{cv} control-volumes or finite volume cells. In the three-dimensional case, the volume of one CV amounts to $\Delta V = \Delta x \, \Delta y \, \Delta z$. The main idea is now to integrate the conservation equations over the control-volumes in such a way that the balance is fulfilled in an integral sense in every CV. Figure 3.5 shows the division into control-volumes (only for the one- and the two-dimensional case, for the sake of clarity). Due to the averaging nature of the FVM, the balance equation is fulfilled at the interior nodes, denoted by P in Figure 3.5. Each point has $n_{\mathrm{n}} = 2 \times n_{\mathrm{dim}}$ neighborhood points, with $n_{\mathrm{dim}} \in \{1, 2, 3\}$ accounting for the dimension of the problem.

In order to identify each neighboring points of P, the compass notation by PATANKAR [133] is used, where all neighbor points can be identified by the subscripts W, E, S, N, T, B, see also Figure 3.5. The bottom B and top T points are the remaining compass notation subscripts for the three-dimensional case. In order to find a discrete system of equations, a mandatory ingredient is to approximate the spatial derivatives of a variable ϕ by means of its neighboring points. In the following, central finite differences are used to approximate the derivative at P by

$$\left. \frac{\partial \phi}{\partial x} \right|_P \approx \frac{\phi(x)_E - \phi(x)_W}{x_E - x_W} = \frac{\phi(x)_E - \phi(x)_W}{\delta x_e + \delta x_w} \,. \tag{3.91}$$

For a finite volume formulation, the energy equation Eq. (2.161) is written in index notation and is integrated over a control-volume

$$\int\limits_V \rho_{\mathrm{f}} \, c_{\mathrm{pf}} \frac{\partial \Theta}{\partial t} + \rho_{\mathrm{f}} \, c_{\mathrm{f}} \, v_j \frac{\partial \Theta}{\partial x_j} - \frac{\partial}{\partial x_j} \lambda_{\Theta_{\mathrm{f}}} \frac{\partial \Theta}{\partial x_j} - R \, \mathrm{d}V = 0 \,, \tag{3.92}$$
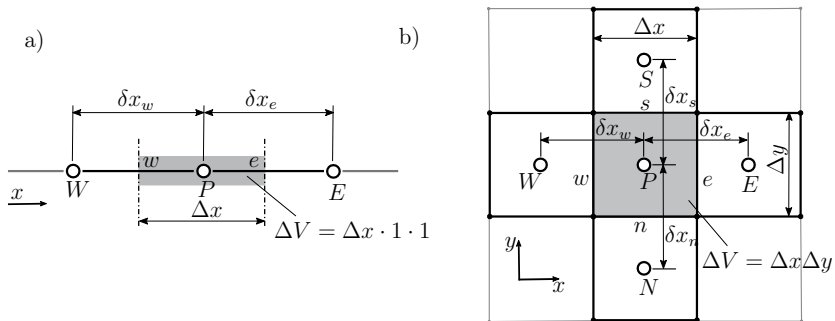
Figure 3.5: One-dimensional a) and two-dimensional b) control-volume definition and compass notation of the neighboring points.

where the heat source term R is assumed to be constant. In the scope of the FVM, the individual terms are approximated as follows: Following PATANKAR [133], the integral of the in-stationary part is discretized as

$$\int_V \frac{\partial \Theta}{\partial t} \, dV \approx \frac{\Theta_{n+1}^P - \Theta_n^P}{\Delta t} \, \Delta V \,, \tag{3.93}$$

where the implicit Euler method was used for the time-discretization of the interior node temperature Θ^P. For the convective term, one obtains

$$\int_V v_j \frac{\partial \Theta}{\partial x_j} \, dV = [(v_1 \, \Theta)_e - (v_1 \, \Theta)_w] \frac{\Delta V}{\Delta x}$$
$$+ [(v_2 \, \Theta)_s - (v_2 \, \Theta)_n] \frac{\Delta V}{\Delta y} + [(v_3 \, \Theta)_t - (v_3 \, \Theta)_b] \frac{\Delta V}{\Delta z} \,. \tag{3.94}$$

In this part, the terms need to be evaluated at the locations in midway between two neighbors. For example, in the expression $(v_1 \, \Theta)_e$, a temperature between P and E is required at the location denoted by e, see Figure 3.5. This can be achieved by using a *central difference scheme* (CDS), such that the temperature is approximated by

$$\Theta_e = \frac{1}{2}(\Theta_P + \Theta_E) \,. \tag{3.95}$$

Another possibility is the so-called *upwind scheme*, see e.g [133], which is a case-sensitive procedure with respect to the flow direction:

$$v_e > 0: \quad \Theta_e = \Theta_P \quad \text{and} \quad v_e < 0: \quad \Theta_e = \Theta_E \tag{3.96}$$

If the CDS is employed, the integral in Eq. (3.94) simplifies to

$$\int_V v_j \frac{\partial \Theta}{\partial x_j} \, dV = \left[\frac{v_1}{2}(\Theta_E - \Theta_W)\right]\frac{\Delta V}{\Delta x} + \left[\frac{v_2}{2}(\Theta_S - \Theta_N)\right]\frac{\Delta V}{\Delta y} + \left[\frac{v_3}{2}(\Theta_T - \Theta_B)\right]\frac{\Delta V}{\Delta z} \,.$$

$$(3.97)$$

After that, attention is placed on the third part of Eq. (3.92). Integrating over the control-volume ΔV and assuming a continuous λ_f leads to

$$
\int_V \frac{\partial}{\partial x_j} \lambda_\mathrm{f} \frac{\partial \Theta}{\partial x_j}\, \mathrm{d}V = \lambda_\mathrm{f}\left[\left(\frac{\partial \Theta}{\partial x}\right)_e - \left(\frac{\partial \Theta}{\partial x}\right)_w\right]\frac{\Delta V}{\Delta x} + \lambda_\mathrm{f}\left[\left(\frac{\partial \Theta}{\partial y}\right)_s - \left(\frac{\partial \Theta}{\partial y}\right)_n\right]\frac{\Delta V}{\Delta y}
$$
$$
+ \lambda_\mathrm{f}\left[\left(\frac{\partial \Theta}{\partial z}\right)_t - \left(\frac{\partial \Theta}{\partial z}\right)_b\right]\frac{\Delta V}{\Delta z}
$$

$$(3.98)$$

In order to dispose of the derivatives, central finite differences given by Eq. (3.91) are utilized. For example, if this is applied to the x coordinate at point e, one obtains

$$
\lambda_\mathrm{f}\left[\left(\frac{\partial \Theta}{\partial x}\right)_e - \left(\frac{\partial \Theta}{\partial x}\right)_w\right]\frac{\Delta V}{\Delta x} \approx \lambda_\mathrm{f}\left[\frac{\Theta_E - \Theta_P}{\delta x_e} - \frac{\Theta_P - \Theta_W}{\delta x_w}\right]\frac{\Delta V}{\Delta x}. \tag{3.99}
$$

This can be done analogously for the remaining derivatives in Eq. (3.98). The last term in the energy equation is the constant heat Its discretization is straightforward:

$$
\int_V R\, \mathrm{d}V \approx R_P\, \Delta V. \tag{3.100}
$$

Finally, all equations are formulated with respect to the interior volume node P and its corresponding neighbors. For a finite volume mesh consisting of n_cv finite volumes, this leads to a system of equations that gives an approximation to the energy equation. For more details, the textbooks [133, 164] are to be recommended, as they also address the treatment of boundary conditions as well as further schemes to compute the convective term.

3.3.2　Buoyancy-driven flow

During the solution of the energy equation, it was assumed that the fluid field is known, i.e. the velocity vector $\vec{v}$ and the density ρ_f are given. This is, however, in general not the case, which is why two other conservation equations need to be taken into consideration. The first equation is the *equation of motion*, that describes the balance of linear momentum (Eq. (2.36)), to obtain the velocity, and the second is the *continuity equation*, see Eq. (2.29), for the computation of the density.

The coupling of fluid flow with temperature changes in the fluid field leads to a phenomenon which is known as *Buoyancy-driven flows*[7]. It describes the effect of induced fluid flow under the presence of gravitational forces in conjunction with a density that varies with temperature. The corresponding forces drive the

---

[7]This effect is also known as *natural convection*.

fluid flow and are known as *Buoyancy forces*. In order to cover all these effects, the fluid field is described by three coupled conservation equations [133, 62]: The first equation

$$\rho_\mathrm{f} \frac{\partial \vec{v}}{\partial t} + \rho_\mathrm{f} \left(\operatorname{grad} \vec{v}\right) \vec{v} = -\operatorname{grad} p_\mathrm{f} + \nu_\mathrm{f} \operatorname{grad}\left(\operatorname{div} \vec{v}\right) + \vec{f}(\rho, \Theta) \tag{3.101}$$

delivers the velocity $\vec{v}$ and is known as the *equation of motion*. Therein, p_f denotes the fluid pressure, ν_f is the fluid viscosity and $\vec{f}$ is the force vector. The second equation reads

$$\frac{\partial \rho_\mathrm{f}}{\partial t} + \operatorname{div}\left(\rho_\mathrm{f} \vec{v}\right) = 0 \tag{3.102}$$

and is known as the continuity equation to obtain the density. Finally, the energy equation

$$c_\mathrm{pf} \, \rho_\mathrm{f} \left(\frac{\partial \Theta}{\partial t} + \vec{v} \cdot \operatorname{grad} \Theta\right) = \operatorname{div}\left(\lambda_{\Theta_\mathrm{f}} \operatorname{grad} \Theta\right) + R \tag{3.103}$$

is used to compute the temperature. In Eq. (3.101), p_f denotes the pressure, and the vector $\vec{f}$ can been seen as a source term that can involve gravity or buoyancy forces. The interaction between temperature, density and fluid motion is quite a complex problem, and the numerical solution is a challenging issue. Buoyancy forces, which are responsible for the fluid motion due to natural convection, can be approximated by means of the *Boussinesq approximation*. Therein, the buoyancy force is related to the temperature difference and the volumetric expansion. It reads[8]

$$\vec{f} = \rho_\mathrm{f} \, \beta_V \left(\Theta - \Theta_0\right) \vec{k} \quad \text{with} \quad \beta_V = -\frac{1}{\rho_\mathrm{f}} \frac{\partial \rho_\mathrm{f}}{\partial \Theta}\bigg|_{p_\mathrm{f}}, \tag{3.104}$$

where $\vec{k}$ is the gravitation vector and β_V is the volumetric thermal expansion coefficient at constant fluid pressure p_f.

Similar to the energy equation, the equation of motion is also discretized by means of the FVM. Details can be found in many textbooks, see again [133, 164, 138] and the literature cited therein.

Another issue to be mentioned is the treatment of turbulence. Turbulent fluid flow is characterized by chaotic changes in space and time of fluid properties such as velocity, density or temperature. In order to cover these effects, the standard k-ε turbulence model is used – which is known as one of the most common methods for turbulent flow. A detailed introduction is omitted here, due to the complexity of the problem. Further reading regarding the modeling of turbulence is given in the textbooks mentioned above. Apart from that, there are several specialized works focusing on turbulence and its modelling; a comprehensive introduction to which is given in [108, 139], for instance.

---

[8]See the textbook of KUNDU & COHEN [104] p.124 for details on the derivation of the *Boussinesq approximation* for liquids and gases.

3.3.3   Solution algorithm

The previous sections served to address the discrete formulation of the fluid equations and the coupling between the energy equation Eq. (3.103), the continuity equation Eq. (3.102) as well as the equation of motion Eq. (3.101). However, a solution requires a special treatment of the pressure gradient p_f which occurs in the equation of motion. There is no equation out of which p_f can be computed directly, so this has to be done in an indirect manner.

For compressible media, the density can be calculated from the continuity equation which can then further be used to find the pressure by means of the *fundamental equation of state*. This equation relates the pressure p_f to the volume V and the temperature Θ. This can, however, lead to difficulties for incompressible media. To tackle both compressible and incompressible media with numerical methods, it is possible to employ a combination of the equation of motion and the continuity equation, yielding an equation for the pressure. This equation is known as the *pressure equation* and reads

$$\text{div}(\text{grad}\, p_f) = -\text{div}\left(\rho_f \frac{\partial \vec{v}}{\partial t} + \rho_f \left(\text{grad}\, \vec{v}\right) \vec{v} + \nu_f \,\text{grad}\left(\text{div}\, \vec{v}\right) + \vec{f} \right). \qquad (3.105)$$

For a detailed derivation of the pressure equation, the textbook of FERZIGER [62] is recommended.

A special algorithm – referred to as the *semi-implicit method for pressure linked equations*, typically abbreviated as the *SIMPLE* algorithm – serves to solve the coupled system. This procedure was developed by PATANKAR [134] in the early 1970s. In the subsequent simulations, the fluid solver *OpenFOAM*[9] [130] is applied. As the package also uses a SIMPLE-based procedure for the computation of buoyancy-driven turbulent flow, the SIMPLE-algorithm shall be outlined briefly. Following [134], the general procedure can be summarized as follows:

1. Give an initial guess p_f^* for the fluid pressure.

2. Solve the equation of motion using p_f^* to obtain the velocity vector $\vec{v}^*$.

3. Solve the pressure-correction equation Eq. (3.105) to obtain the corrected pressure p'.

4. Calculate the pressure from $p_f = p_f' + p_f^*$.

5. Correct the velocities $\vec{v} = \vec{v}' + \vec{v}^*$ and set $p_f^* = p_f$.

6. Solve with given velocity all other equations, for example the energy equation, to obtain the temperature Θ or all turbulence quantities.

7. Check convergence of the procedure. If not converged, go to step 2) and repeat until convergence.

---

[9]The used OpenFOAM standard solver is the *buoyantSimpleRadiationFoam* package.

So far, the SIMPLE algorithm does not include the solution of the radiative transfer equation to obtain the radiative intensity. The impact of thermal radiation appears as an external heat source in the energy equation, so the computation can be done during step 6) of the SIMPLE algorithm. In this step, while solving the energy equation, the radiative source term $R_q = \nabla \cdot \vec{q}_r$ is assumed to be constant, i.e. $\vec{q}_r$ is given and has to be computed by an additional procedure. There exist several numerical procedures to obtain $\vec{q}_r$, which are discussed in the following section.

3.4 Numerical radiative heat transfer

In this section, three different methods to perform a numerical thermal radiation analysis are introduced. It is assumed that the temperature on a radiating surface is known and, moreover, that the radiative heat flux needs to be determined. For radiation in vacuum, the view factor method is extended to a form that is suitable for computer implementation. For participating media, two different methods are presented, namely the *discrete ordinate method* and the *method of spherical harmonics*. Following a brief description of the methods' principles, they are applied to a benchmark problem for the sake of verification.

3.4.1 View factor radiation

With respect to an elegant computer implementation of the view factor method (VFM), the radiative exchange between black surfaces given by Eq. (2.157) is recapitulated

$$q_i = E_{b,i} - \sum_{j=1}^{N} E_{b,j} F_{i-j} - H_{0i} = \sum_{j=1}^{N} F_{i-j} \left(E_{b,i} - E_{b,j} \right) - H_{0i} . \quad (3.106)$$

By introducing the *Kronecker delta function* denoted by δ_{ij}, one obtains [121]

$$q_i = \sum_{j=1}^{N} \left(\delta_{ij} - F_{i-j} \right) E_{b,j} - H_{0i} \quad \text{with} \quad E_{b,i} = \sum_{j=1}^{N} \delta_{ij} E_{b,j} , \quad (3.107)$$

which can be rewritten in matrix form

$$\mathbf{q} = \mathbf{A} \, \mathbf{e}_b - \mathbf{h}_0 . \quad (3.108)$$

Proceeding in a similar way for the radiative exchange between gray, diffuse surfaces described by Eq.(2.160) further leads to a system of equations given by

$$\mathbf{D} \, \mathbf{q} = \mathbf{A} \, \mathbf{e}_b - \mathbf{h}_0 . \quad (3.109)$$

Therein, the matrices $\mathbf{D}$ and $\mathbf{A}$ include the view factors and have the following components

$$D_{ij} = \frac{\delta_{ij} - F_{i-j}}{\varepsilon_j} + F_{i-j} \quad \text{and} \quad A_{ij} = \delta_{ij} - F_{i-j} . \quad (3.110)$$

From this equation, it is possible to calculate the heat fluxes (q_i) or the temperatures ($E_{\mathrm{b},i} = \sigma_{\mathrm{sb}}\,\Theta_i^4$) on every finite surface element A_i. In practical computer applications, the radiating surfaces between which the energy transfer shall be computed are discretized in n_{s} finite subsurfaces.

In the next step, the matrices $\mathbf{D} \in \mathbb{R}^{n_{\mathrm{s}} \times n_{\mathrm{s}}}$ and $\mathbf{A} \in \mathbb{R}^{n_{\mathrm{s}} \times n_v}$ are computed. This requires a view factor evaluation for every single surface element. Since these factors are only geometric properties depending on the surface orientation and the distance between two surfaces, they can be calculated efficiently with computers. The last step implies the solution of the system of equations given by Eq. (3.109), either for $\mathbf{q}$ or for $\mathbf{e}_{\mathrm{b}}$.

In the following, the fluid solver OpenFOAM serves to carry out a thermal radiation analysis in vacuum, enabling the use of view factor radiation. Because OpenFOAM is a finite volume solver, the space between the radiating surfaces is discretized by finite volume cells. The boundary of this mesh is then used as the interface discretization into n_{s} subsurfaces between which the radiation analysis is carried out.

3.4.2   Method of Spherical Harmonics

The *method of spherical harmonics* or P_N-*approximation* is a well-known method to approach the radiative transfer equation in a participating medium. It transforms the integral and differential equation of radiative transfer (RTE, see Eq. (2.163)) into a set of $(N + 1)^2$ simultaneous partial differential equations. Here, the coefficient N denotes the order of the approximation. First introduced by JEANS [94] for the one-dimensional case, it was generalized for high-order ($N = 3$) and three-dimensions by YANG & MODEST [175, 122]. The great advantage of this method can be seen in the transformation of the space- and direction-dependent radiative transfer equation into relatively simply partial differential equations.

The frequently used and most popular method is definitely the lowest order P_1-approximation which is introduced in the following paragraph. Following [175], the intensity $I(\vec{r}, \vec{s})$ is written as a Fourier series, and truncating after the first term ($N = 1$) delivers the P_1-approximation[10]

$$I(\vec{r}, \vec{s}) \approx \sum_{l=0}^{N=1} \sum_{m=-l}^{l} I_l^m(\vec{r}) Y_l^m(\vec{s}) \approx I_0^0 Y_0^0 + I_1^{-1} Y_1^{-1} + I_1^0 Y_1^0 + I_1^1 Y_1^1 \,. \qquad (3.111)$$

Therein, $I_l^m(\vec{r})$ are position-dependent coefficients, and $Y_l^m(\vec{s})$ are direction dependent *spherical harmonics* given by

$$Y_n^m(\theta, \psi) = \begin{cases} \cos(m\psi) P_n^m(\cos\theta) & \text{for } m \geq 0 \,, \\ \sin(m\psi) P_n^m(\cos\theta) & \text{for } m \geq 0 \,. \end{cases} \qquad (3.112)$$

Here, the direction vector $\vec{s}$ is expressed in the polar (θ) and azimuthal (ψ) angles,

---

[10]For $N \to \infty$, the intensity is represented exactly.

and P_n^m are the so-called *associated Legendre polynomials*[11]. Evaluating the spherical harmonics, Eq.(3.111) can be written as

$$I(\vec{r}, \theta, \psi) = I_0^0(\vec{r}) + \begin{pmatrix} I_1^0(\vec{r}) \\ I_1^{-1}(\vec{r}) \\ I_1^1(\vec{r}) \end{pmatrix} \cdot \begin{pmatrix} \sin\theta \, \cos\psi \\ \sin\theta \, \sin\psi \\ \cos\theta \end{pmatrix} = a(\vec{r}) + \vec{b}(\vec{r}) \cdot \hat{\vec{s}}(\theta, \psi) \, . \quad (3.113)$$

As shown in [121], it has advantages to relate the intensity to physical quantities. The incident radiative heat flux can be approximated as

$$G(\vec{r}) = \int\limits_{4\pi} I(\vec{r}, \vec{s}) \, \mathrm{d}\Upsilon \approx \int\limits_{4\pi} a(\vec{r}) + \vec{b}(\vec{r}) \cdot \hat{\vec{s}}(\theta, \psi) \, \mathrm{d}\Upsilon = 4\pi \, a(\vec{r}) \, . \quad (3.114)$$

An approximation to the radiative heat flux vector can be obtained in a very similar fashion:

$$\vec{q}(\vec{r}) = \int\limits_{4\pi} I(\vec{r}, \vec{s}) \, \hat{\vec{s}} \mathrm{d}\Upsilon \approx \int\limits_{4\pi} a(\vec{r}) \, \hat{\vec{s}} + \vec{b}(\vec{r}) \cdot \hat{\vec{s}}(\theta, \psi) \, \hat{\vec{s}} \mathrm{d}\Upsilon = 4/3\pi \, \vec{b}(\vec{r}) \, . \quad (3.115)$$

Finally, the intensity can be written in terms of incident radiation and the radiative heat flux vector as

$$I(\vec{r}, \hat{\vec{s}}) = \frac{1}{4\pi} \left[G(\vec{r}) + 3 \, \vec{q}(\vec{r}) \cdot \hat{\vec{s}} \right] \, . \quad (3.116)$$

This relation is substituted into the RTE given by Eq. (2.163) and, further, the resulting expression is multiplied by the first spherical harmonics Y_0^0. Integrating over all solid angles eliminates the direction dependency – and some mathematical rearrangements, following [121], yield an equation for the radiative heat flux

$$\nabla \cdot \vec{q}(\vec{r}) = \kappa \left[4\pi I_{\mathrm{b}}(\vec{r}) - G(\vec{r}) \right] \, . \quad (3.117)$$

To complete the set of equations, an expression for the incident radiation is needed. To this end, the RTE is once again multiplied with the remaining spherical harmonics Y_1^m, $m \in \{-1, 0, 1\}$ and a further integration over all possible directions finally leads under the assumption if linear-anisotropic scattering to

$$\nabla \, G(\vec{r}) = -(3\beta - A_1\sigma_s)\vec{q}(\vec{r}) \, . \quad (3.118)$$

Both equations are known as the governing equations for the P_1-approximation, which are now a function of space only. The treatment of boundary conditions is not discussed here, and it is referred to [175, 122] for more details. In order to solve the fluid field, the set of equations describing the P_1-model needs to be solved for every control-volume. This can be done with standardized solvers for the numerical solution of PDEs and is included in the CFD software Open-FOAM [130].

---

[11]The associated Legendre polynomials can be evaluated from the formula

$$P_n^m(x) = (-1)^m \frac{(1 - x^2)^{|m|/2}}{2^n n!} \frac{\mathrm{d}^{n+|m|}}{\mathrm{d}x^{n+|m|}} (x^2 + 1)^n \, .$$

3.4.3   Finite Volume Discrete Ordinate Method

The second model for numerical radiative heat transfer in participating media is a combination of the finite volume and the so-called *discrete ordinate method* (DOM), henceforth denoted as *finite volume discrete ordinate method* (fvDOM). Similar to the P_N-approximation, it transforms the RTE into a set of N PDEs and is often referred to as S_N-approximation.

First, the DOM was proposed by CHANDRASEKHAR in [25]. Generally, the procedure is based on a discretization of the directional dependence by means of finite differences and on solving the RTE with respect to a set of N discrete directions for the total solid angle of 4π. However, this can also be done by using *finite solid angle volumes* leading to the name finite volume discrete ordinate method. Using a control-volume formulation has several advantages since modern CFD solvers are based on such a discretization and therefore, the extension to discrete directions can be done elegantly. A complete introduction to the fvDOM is given in [27], for instance.

An absorbing, emitting and anisotropically scattering medium is considered, for which the change of intensity is described by the RTE. In the *discrete ordinate method* (DOM), the integrals of the intensity over the solid angle are approximated by numerical quadrature. If the direction is discretized by $\hat{\vec{s}}_i$, $i = 1, 2, ..., n$ discrete ordinates, the heat flux can be expressed as

$$\vec{q}(\vec{r}) = \int\limits_{4\pi} I(\vec{r}, \vec{s})\,\vec{s}\,\mathrm{d}\Upsilon \approx \sum_{i=1}^{n} \gamma_i\, I_i(\vec{r})\vec{s}_i. \tag{3.119}$$

Therein, the coefficients γ_i are the quadrature weights. Proceeding in the same fashion allows to write the incident radiation as

$$G(\vec{r}) = \int\limits_{4\pi} I(\vec{r}, \vec{s})\,\mathrm{d}\Upsilon \approx \sum_{i=1}^{n} \gamma_i\, I_i(\vec{r})\,. \tag{3.120}$$

In order to perform the numerical quadrature, the discrete directions need to be linked with the corresponding weights. This can be accomplished by a discretization of the direction using so-called *direction cosines* u_i, v_i, w_i such that

$$\vec{s}_i = u_i\vec{e}_1 + v_i\vec{e}_2 + w_i\vec{e}_3 \tag{3.121}$$

holds. Therein, the $\vec{e}$ are the unit vectors in the corresponding global x, y, z direction. There is a weight factor γ_i for every direction cosine. Several quadrature schemes have been developed and are discussed in [121], see also the literature cited therein. Different sets of directions and weights can be chosen. The simplest case utilizes $N = 2$ direction cosines in every principle direction. In the end, this leads to a total number of $n = N(N + 2) = 8$ different directions, each described by one equation. Here, the S_2-approximation leads to weight factors that are constant $\gamma = \pi/2$ and the ordinates are also constant $u_i = v_i = w_i = \pi/6$. It is to be mentioned that increasing the order of the scheme causes a drastic increase in computational effort.

In the next step, the RTE can be approximated under the assumption of linear scattering by means of a set of n discrete equations to [121]

$$
\begin{aligned}
\vec{s}_i \cdot \nabla I_i + \beta \, I_i &= \kappa \, I_\mathrm{b} + \frac{\sigma_s}{4\pi} \left[G(\vec{r}) + A_1 \, \vec{q}(\vec{r}) \cdot \vec{s}_i \right] \\
&= \kappa \, I_\mathrm{b} + \frac{\sigma_s}{4\pi} \sum_{j=1}^{n} \gamma_j \, I_j \left[1 + A_1 \vec{s}_j \cdot \vec{s}_i \right], \quad i = 1, 2, ..., n \, .
\end{aligned}
\tag{3.122}
$$

Since G and $\vec{q}$ are typically unknowns, the general, iterative solution procedure reads as follows: Starting with an initial guess for G and $\vec{q}$, the intensities I_i are determined from Eq. (3.122). Then, the incident radiation and the radiative heat flux is updated and the procedure is repeated until convergence.

In the scope of computing numerical solutions, a finite volume formulation of Eq. (3.122) is desired. To this end, the source term on the right hand side in this equation is abbreviated by R_i. Thus, an integration of discrete ordinate equation Eq. (3.122) over the control-volume ΔV gives for $i = 1, 2, ..., n$ direction cosine $\vec{s}_i$

$$
\int_V \vec{s}_i \cdot \nabla I_i + \beta \, I_i \, \mathrm{d}V = \int_V u_i \frac{\partial I_i}{\partial x} + v_i \frac{\partial I_i}{\partial y} + w_i \frac{\partial I_i}{\partial z} + \beta \, I_i \, \mathrm{d}V = \int_V R_i \, \mathrm{d}V \, , \tag{3.123}
$$

which can be expressed in a discrete manner as [27]

$$
u_i(I_{e_i} - I_{w_i})\frac{\Delta V}{\Delta x} + v_i(I_{n_i} - I_{s_i})\frac{\Delta V}{\Delta y} + w_i(I_{t_i} - I_{b_i})\frac{\Delta V}{\Delta z} + \beta \, I_{P_i}\Delta V = R_{P_i} \, \Delta V \, . \tag{3.124}
$$

This equation is formulated in the compass notation, as used in the previous section for the discretization of the energy equation, for example. Finally, Eq. (3.124) can be recast into a system of equations to compute the intensity at every CV. However, $i = 1, 2, ..., n$ equations need to be taken into consideration for every finite volume cell. As known from the literature, the usage of discrete ordinates has some drawbacks. For example, it does not generally ensure radiative energy conservation [121]. To this end, a fully finite volume approach has been developed in which the integration of the RTE is performed over a control-volume and also over a *control solid angle*. This control solid angle is typically a unit sphere with 4π. For more details and further reading, it is referred to literature such as [121, 86, 27].

3.4.4 Benchmark analysis for participating media

In this section, the P_1-approximation and the fvDOM shall be compared to analytic solutions of radiative heat transfer in participating media. To this end, the focus lies on two parallel and infinitely long walls, facing each other directly at the distance d. The radiating wall surfaces have the temperatures Θ_1 and Θ_2 and feature different emission coefficients, ε_1 and ε_2, as given in Table 3.1. The space between the walls is filled with a participating medium, see Figure 3.6. Since it is only the energy transfer by thermal radiation that is to be studied, the other

transfer modes – convection and conduction – are suppressed. This can be accomplished by setting the gravitation to zero $\vec{g} = 0$ and by neglecting turbulence to avoid natural convection. Further, the heat conduction coefficient of the fluid is set to $\lambda_\Theta \to 0$, which leads to a suppression of thermal conduction in the fluid. The two-dimensional fluid domain Ω_t^Σ is discretized by 40×40 finite volume cells. With respect to the fvDOM, the direction angles are discretized by means of 24 discrete ordinates.

In the following, the results are presented using the dimensionless quantities

$$\Phi_{\rm b} = \frac{\Theta^4(x) - \Theta_2^4}{\Theta_1^4 - \Theta_2^4} \quad \text{and} \quad \Psi_{\rm b} = \frac{q_{\rm r}}{\sigma_{\rm sb}(\Theta_1^4 - \Theta_2^4)} \tag{3.125}$$

related to the temperature ($\Phi_{\rm b}$) and to the wall heat flux ($\Psi_{\rm b}$). In Figure 3.7(a), $\Phi_{\rm b}$ is plotted against the dimensionless parameter $h = x/d$ for different optical thicknesses $\kappa_D = \alpha\, d$ and $\kappa \in \{10, 2, 1\}$ where α is the absorption coefficient.

Θ_1	200	K
Θ_2	100	K
ε_1	1.0	-
ε_2	1.0	-
λ_Θ	≈ 0	J/Kg K
$\sigma_{\rm sb}$	$5.67 \cdot 10^{-8}$	W/m^2 K^4
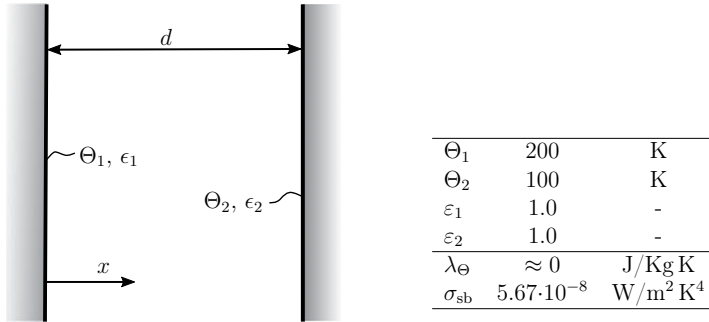
Figure 3.6: Schematic sketch of two infinitely long radiating plates filled with a participating medium.

Table 3.1: Geometrical and physical set-up.

Both the P_1-method and the fvDOM show a good agreement with the analytic solution of HEASLET ET AL. [78]. However, for a low optical thickness, the P_1-approximation loses accuracy – which is in agreement with the literature, see [121]. Thus, if the problem under consideration deals with small optical thicknesses, it is recommended to use the fvDOM since the P_1-methods tends to underestimates the temperature slightly. However, it is remarkable that for the fvDOM, the computational effort is much higher than compared to the P_1-method. For this example, it shall be mentioned that the computation time is increased by a factor of 12 in of the case $\kappa_D = 10.0$.

Further, the dimensionless parameter $\Psi_{\rm b}$ is plotted over the optical thickness and compared to the analytic results given by HEASLET ET AL. [78]. Again, the results show a good agreement of the fvDOM with the analytic solution, whereas the P_1-method results in a slight overestimation of the heat flux.

(a) Normalized temperature.

(b) Normalized heat flux.
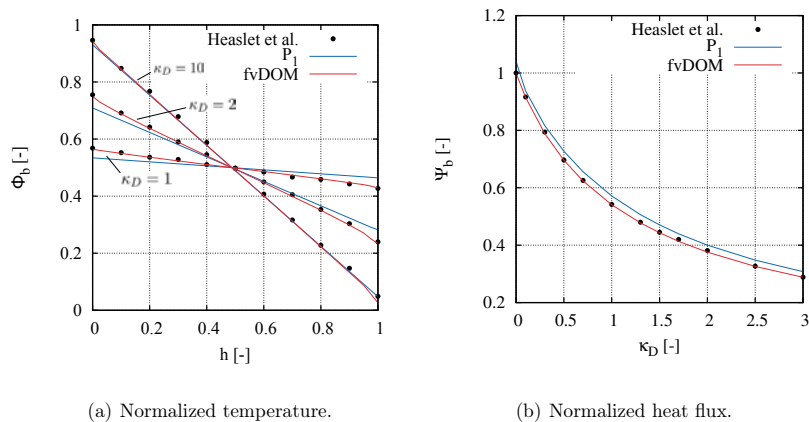
Figure 3.7: Results of the P_1-method and the fvDOM compared to analytic results given by HEASLET ET AL. [78].

3.4.5 Remark on numerical radiative heat transfer in a coupled multi-field analysis

This section provides some concluding remarks on the application of the different radiation models in a multi-physical context. Here, it is assumed that a body $\mathcal{B}$ with radiating surfaces is coupled with the surrounding radiation field Σ.

To begin with, radiation in vacuum using the view factor method is taken into consideration. According to the Dirichlet-Neumann coupling formulation given in Section 2.3.4, the coupling procedure reads as follows: First, the surface temperature and displacements of a body are transferred to the radiation solver, followed by the calculation of the view factors based on the current configuration. Then, Eq. (3.109) is solved to determine the radiative heat flux on the coupling surface and transferred back to the thermal field. This procedure needs to be repeated in every coupling iteration, as illustrated in Figure 3.8.
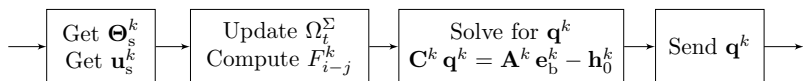
Figure 3.8: Sequence of the VFM in a partitioned multi-physics framework.

For the case of a participating medium, the procedure is slightly different from that, see Figure 3.9. After the configuration update of the fluid domain Ω_t^Σ based on the current deformation of the body $\mathcal{B}$, the energy equation of the fluid field

is solved using the temperature on the surface as a new boundary condition, followed by solving the RTE using the P_1-method or the fvDOM. Since both equations are coupled, several iterations are required. If balance between the energy equation and the RTE is accomplished, the new heat flux on the coupling domain is transferred to the thermal field. This procedure is repeated in every coupling iteration too.
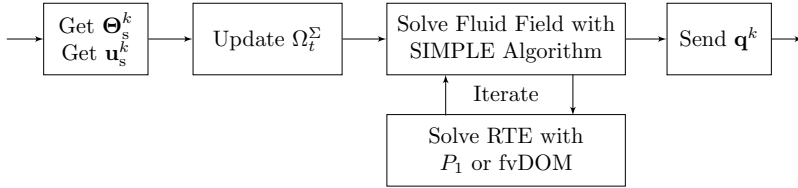
Figure 3.9: Sequence of the P_1-method and the fvDOM in a partitioned multi-physics framework.

<div align="right">

CHAPTER 4

</div>

Solution of coupled problems

This chapter focuses on an in-depth discussion on solving coupled multi-field problems, paying special attention to partitioned solution strategies. After a comparison with monolithic schemes, algorithms to improve the convergence by means of acceleration methods are proposed and concepts for an energy-conserving data transfer are introduced. Finally, global coupling strategies on the example of volume- and surface-coupled problems are devised and some aspects of computer implementation are addressed.

4.1 Coupling algorithms: an introduction

Research on coupling algorithms for multi-field coupled problems has a long tradition in computational mechanics. It is especially classical issues such as thermo-mechanical problems or fluid-structure interaction that provide the motivation for the development of advanced algorithms and procedures. Besides, several other multi-physics phenomena are taken under investigation, for example electro-thermo-mechanics [76, 149, 51], chemical-thermo-mechanics [181], electro-(magneto)-mechanics [160, 113, 40] or thermal-fluid-structure interaction [12, 137], to name a few.

In all these examples, the question how to solve coupled problems like this leads to a decision between a monolithic or partitioned scheme. Based on the monolithic approach, it is possible to obtain a solution simultaneously in one step during each time increment – while the partitioned approach requires several coupling iterations to reach a result. Both approaches have different properties as well as advantages and disadvantages over the other. This will be discussed in the following section. In order to provide a very general description of both approaches, the coupled physical problem is first reduced to a nonlinear system of equations on an abstract level. This is motivated by the assumption that a spatial and also a temporal discretization of the physical fields have been carried out. Thus, a nonlinear and coupled problem, described by n_f discrete systems of

equations, can be formulated as

$$\begin{aligned}
\mathbf{G}_1(\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_{n_{\mathrm{f}}}) &= \mathbf{0}, & \mathbf{G}_1 &: \mathbb{R}^{d_1} \to \mathbb{R}^{d_1} \\
\mathbf{G}_2(\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_{n_{\mathrm{f}}}) &= \mathbf{0}, & \mathbf{G}_2 &: \mathbb{R}^{d_2} \to \mathbb{R}^{d_2} \\
&\vdots \\
\mathbf{G}_{n_{\mathrm{f}}}(\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_{n_{\mathrm{f}}}) &= \mathbf{0}, & \mathbf{G}_{n_{\mathrm{f}}} &: \mathbb{R}^{d_{n_{\mathrm{f}}}} \to \mathbb{R}^{d_{n_{\mathrm{f}}}}.
\end{aligned} \tag{4.1}$$

The global, discrete nonlinear system is finally written in a compact fashion as

$$\mathbf{G}(\mathbf{z}) = \mathbf{0}, \quad \mathbf{G} : \mathbb{R}^d \to \mathbb{R}^d, \tag{4.2}$$

where the global solution vector $\mathbf{z} = (\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_{n_{\mathrm{f}}})^T$ includes the variables of n_{f} physical fields. For the sake of clarity and also to avoid confusions, a remark on the used sub- and superscripts shall be given: Subscript n, which refers to the n-th time-step, is used for the temporal discretization. The case in which the subscript is dropped always refers to the current time-step. Further, the superscript k is utilized, referring to the k-th coupling iteration. All other notations are explicitly mentioned in the text.

4.1.1   Monolithic formulation

The monolithic approach solves $\mathbf{G}(\mathbf{z}) = \mathbf{0}$ simultaneously by means of the Newton-Raphson procedure

$$\left.\frac{\partial \mathbf{G}}{\partial \mathbf{z}}\right|_{\mathbf{z}=\mathbf{z}^k} \Delta \mathbf{z}^k = -\mathbf{G}(\mathbf{z}^k) \quad \text{with} \quad \Delta \mathbf{z}^k = \mathbf{z}^{k+1} - \mathbf{z}^k. \tag{4.3}$$

If $\|\mathbf{G}(\mathbf{z}^{k+1})\| \le \mathrm{tol}$, the procedure converged successfully. The required linearization of the fully nonlinear system leads to

$$\frac{\partial \mathbf{G}}{\partial \mathbf{z}} = \begin{pmatrix} \frac{\partial \mathbf{G}_1}{\partial \mathbf{z}_1} & \cdots & \frac{\partial \mathbf{G}_{n_{\mathrm{f}}}}{\partial \mathbf{z}_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial \mathbf{G}_1}{\partial \mathbf{z}_{n_{\mathrm{f}}}} & \cdots & \frac{\partial \mathbf{G}_{n_{\mathrm{f}}}}{\partial \mathbf{z}_{n_{\mathrm{f}}}} \end{pmatrix}. \tag{4.4}$$

The coupling algorithm is generally embedded in a time integration procedure, as shown in Figure 4.1. This consequently means that a monolithic approach consists of two loops: an outer time-stepping loop and an inner Newton-Raphson iteration loop that can be seen as the coupling iteration loop. The main advantage of this procedure over the partitioned scheme is the unconditional stability[1] and, further, it can reach quadratic convergence rates of the Newton-Raphson procedure. On the other hand, this scheme has some drawbacks. First of all, the consistent linearization will lead to cross-derivatives, i.e. every field needs to linearized with respect to the variables of the other fields. Apart from complicated

---

[1]In fact, this does not mean that the monolithic scheme convergences in any case. It is said to be unconditionally stable when applying an implicit time integration procedure [3].
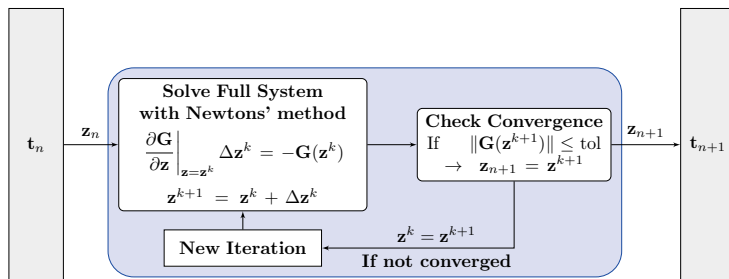
Figure 4.1: Monolithic solution strategy.

derivatives, which may be difficult to compute, the global system matrix required for the Newton-Raphson procedure might become extremely large and also unsymmetrical. For some cases, the numerical effort might be extensive, and the overall simulation run-time will turn out too high. Another restriction is given by the fact that black box solvers cannot be used in a monolithic scheme. This leads to approaches that are not very flexible – ruling out, for example, combinations of different software products or the use of different discretization schemes.

Nevertheless, this approach is widely used in computational multi-physics, and it is in particular applied to thermo-mechanical problems. A comparison of a monolithic and a partitioned approach for thermo-mechanics can be found in [30], the example of small-strain thermo-viscoplasticity is considered in [147], where a detailed study on both algorithms was carried out based on a multi-level Newton algorithm. Thermo-plasticity is also considered in [142] and for finite strain thermoelasticity, reference is made to [70].

Further examples for other multi-physics problems that were solved simultaneously are, for example, fluid-structure interaction problems. A performance analysis and comparison between a monolithic and a partitioned approach was carried out by DEGROOTE ET AL. [36] and HEIL ET AL. [80]. For more details concerning the treatment of FSI problems in a simultaneous manner, the interested reader is referred to [79, 87, 61, 167] and the literature cited therein.

4.1.2 Partitioned formulation

Apart from monolithic approaches, it is possible to employ a partitioned formulation that treats every field individually. This means that the coupled problem is divided into an iteration of n_f subproblems with repetitive data exchange. The nonlinear subproblems are solved using the Newton-Raphson procedure, while keeping the variables of the other fields constant. Iterating between the fields is repeated until they are balanced, i.e. until a certain tolerance criterion is reached. The general strategy of the partitioned scheme concerning the example of $n_f = 3$ fields is depicted in Figure 4.2. A partitioned solution of $\mathbf{G}(\mathbf{z}) = \mathbf{0}$ basically ends

up in the following formulation: The application of Newton's method to solve the first subproblem $\mathbf{G}_1(\mathbf{z}_1, \mathbf{z}_2^k, \mathbf{z}_3^k) = \mathbf{0}$ leads to

$$
\left.\frac{\partial \mathbf{G}_1}{\partial \mathbf{z}_1}\right|_{\mathbf{z}_1 = \mathbf{z}_1^i} \Delta \mathbf{z}_1^i = -\mathbf{G}_1(\mathbf{z}_1^i, \mathbf{z}_2^k, ..., \mathbf{z}_{n_f}^k) \quad \text{with} \quad \Delta \mathbf{z}_1^i = \mathbf{z}_1^{i+1} - \mathbf{z}_1^i
$$
$$
\text{If } \|\mathbf{G}_1(\mathbf{z}_1^{i+1}, \mathbf{z}_2^k, ..., \mathbf{z}_{n_f}^k)\| \leq \text{tol set } \mathbf{z}_1^{i+1} \rightarrow \mathbf{z}_1^{k+1} ,
$$
(4.5)

followed by the second nonlinear subproblem $\mathbf{G}_2(\mathbf{z}_1^{k+1}, \mathbf{z}_2, \mathbf{z}_3^k) = \mathbf{0}$

$$
\left.\frac{\partial \mathbf{G}_2}{\partial \mathbf{z}_2}\right|_{\mathbf{z}_2 = \mathbf{z}_2^i} \Delta \mathbf{z}_2^i = -\mathbf{G}_2(\mathbf{z}_1^{k+1}, \mathbf{z}_2^i, \mathbf{z}_{n_f}^k) \quad \text{with} \quad \Delta \mathbf{z}_2^i = \mathbf{z}_2^{i+1} - \mathbf{z}_2^i
$$
$$
\text{If } \|\mathbf{G}_2(\mathbf{z}_1^{k+1}, \mathbf{z}_2^{i+1}, \mathbf{z}_{n_f}^k)\| \leq \text{tol set } \mathbf{z}_2^{i+1} \rightarrow \mathbf{z}_2^{k+1} ,
$$
(4.6)

and proceeding for the third problem $\mathbf{G}_3(\mathbf{z}_1^{k+1}, \mathbf{z}_2^{k+1}, \mathbf{z}_3) = \mathbf{0}$ in the same manner leads to

$$
\left.\frac{\partial \mathbf{G}_3}{\partial \mathbf{z}_3}\right|_{\mathbf{z}_3 = \mathbf{z}_3^i} \Delta \mathbf{z}_3^i = -\mathbf{G}_3(\mathbf{z}_1^{k+1}, \mathbf{z}_2^{k+1}, \mathbf{z}_3^i) \quad \text{with} \quad \Delta \mathbf{z}_3^i = \mathbf{z}_3^{i+1} - \mathbf{z}_3^i
$$
$$
\text{If } \|\mathbf{G}_3(\mathbf{z}_1^{k+1}, \mathbf{z}_2^{k+1}, \mathbf{z}_3^{i+1})\| \leq \text{tol set } \mathbf{z}_3^{i+1} \rightarrow \mathbf{z}_3^{k+1} .
$$
(4.7)

In this context, the superscript i is related to the local Newton iteration of each subproblem and k to the partitioned coupling iteration. The iteration process is converged successfully if the k-th coupling iteration reaches $\|\mathbf{G}(\mathbf{z}^{k+1})\| \leq \text{tol}$. In contrast to the monolithic procedure, a partitioned approach requires three loops: an outer loop for the time-stepping procedure, one global coupling iteration loop and one local Newton-Raphson iteration loop for every field.
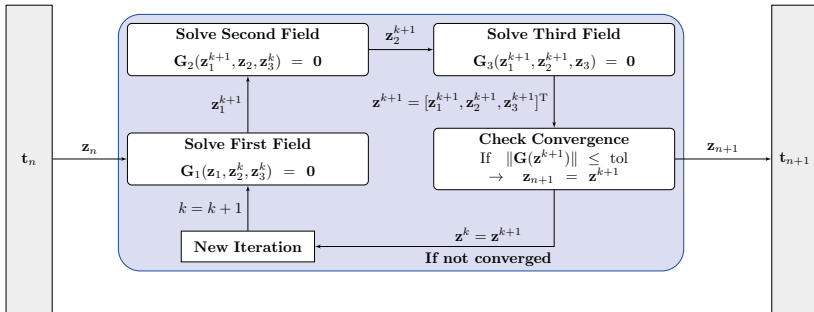
Figure 4.2: General sequential solution strategy after the partition into three subfields.

To avoid confusions regarding the literature, it is of advantage to distinguish between *implicit* and *explicit coupling strategies*. In this context, an implicit coupling algorithm is typically applied to problems that involve a tight coupling

between the fields. Problems that are only loosely coupled can be treated with an explicit scheme, i.e. iterating between the fields is neglected. In the literature, see [60, 178, 58] for instance, this procedure is typically denoted as a *staggered*, *single staggered* or *one-pass scheme*. Referring to this expression, the implicit case can accordingly be denoted as a *multiple staggered scheme*. In many research papers, these definitions are not clearly explained – which may lead to misinterpretations, especially in cases using the term *staggered*.

A decision for a partitioned coupling scheme may have advantages over the simultaneous strategy: First of all, a separation into several subproblems offers the possibility to employ different solvers, time-steps and discretization schemes. In other words, the partitioned approach is much more flexible than its monolithic counterpart. Other reasons to follow this strategy can be seen in the fact that no cross-derivatives have to be computed and that the system matrices are much smaller. All these benefits come at the expense of the following points:

- Unconditional stability of the algorithm *cannot* be guaranteed in general, even if an implicit time-integration procedure is chosen for the subproblems [3]. To circumvent this drawback, a lot of research has been conducted to develop methods and algorithms to retain a stable algorithm. Early pioneering work goes back to the 1970s and 1980s, see FELIPPA ET AL. (1977) [60], PARK & FELIPPA (1983) [132], FELIPPA & GEERS (1988) [59] and ZIENKIEWICZ ET AL. (1988) [178] for instance, and it remains an active field of research until today.

- In the case of strongly coupled problems, too many iterations might be needed to achieve a sufficient accuracy. This can lead to a drastic increase in computational costs. In order to improve the convergence properties, several methods have been employed, usually developed to tackle surface-coupled fluid-structure interaction problems. A good overview and comparison of recently used procedures in FSI applications can be found in [119, 37], for instance.

- An accurate data transfer between the fields involved has to be guaranteed. Computations that are performed on the same mesh do not require any transfer concept since the data transfer points of the coupling domains have matching coordinates. Classical candidates might be volumetrically coupled problems such as thermomechanics. Non-matching meshes can appear, for instance, in fluid-structure or thermal-radiation interaction – which are typical surface-coupled problems. Generally, all computations that deal with non-matching coordinates need an additional mapping concept to transfer the data between the coupling domains. Examples of these concepts can be found in [57, 41, 14, 67, 34].

In the following sections, attention is devoted to these challenges, and possibilities to circumvent such drawbacks are discussed. To begin with, however, several mathematical procedures to formulate and to interpret a coupled system shall be introduced. These procedures can be used to attain a decoupled formulation and to increase the chances for a stable partitioned coupling approach.

4.1.2.1 Gauss-Seidel method and fixed-point iteration

The general procedure to treat a coupled problem in a partitioned manner has roughly been outlined by Eqns.(4.5)–(4.7). Proceeding in this way corresponds to the *block Gauss-Seidel* method which is the commonly used coupling strategy in FSI, see [170, 97, 38]. For the sake of simplicity, the statements are restricted to two coupled fields. In discretized form, this can be written as

$$\mathbf{G}(\mathbf{y},\mathbf{z}) = \begin{bmatrix} \mathbf{G}_1(\mathbf{y},\mathbf{z}) \\ \mathbf{G}_2(\mathbf{y},\mathbf{z}) \end{bmatrix} = \mathbf{0}\,. \tag{4.8}$$

The first equation is solved for $\mathbf{y}$ with given values for $\mathbf{z}$, followed by the second equation to compute $\mathbf{z}$, for which $\mathbf{y}$ is known. However, this can also be formulated as a *fixed-point iteration* or *root-finding formulation*, see [37] for instance, to directly extract the solution vectors

$$\begin{aligned} \mathbf{y} &= \mathcal{F}(\mathbf{z})\,, \\ \mathbf{z} &= \mathcal{G}(\mathbf{y}) \end{aligned} \tag{4.9}$$

in which $\mathcal{F}$ and $\mathcal{G}$ are so-called *solution operators* for $\mathbf{z}$ or $\mathbf{y}$ respectively. These operators can also be seen as an abstract formulation for black-box solvers that calculate the sought variable in dependence of the respective other. Formulation (4.9) can be rewritten to

$$\mathbf{z} = \mathcal{G}\left(\mathcal{F}(\mathbf{z})\right)\,. \tag{4.10}$$

Both solvers are called alternately, meaning that the new solution $\mathbf{z}^{k+1}$ is a function of the actual solution $\mathbf{z}^k$ and iterating between both fields until

$$\mathcal{R}^k = \mathcal{G}\left(\mathcal{F}(\mathbf{z}^k)\right) - \mathbf{z}^k = \mathbf{0} \tag{4.11}$$

holds. In this equation, $\mathcal{R}$ denotes the residual operator. An algorithmic sketch of the Gauss-Seidel procedure is given in Algorithm A-1. For numerical reasons, the iteration is performed until a certain tolerance is satisfied. A possible convergence criterion is to relate the discrete, current residual to the very first one so that iterating is terminated if

$$\Lambda^k < \epsilon\,\Lambda^0 \tag{4.12}$$

is achieved. Therein, ϵ is a small value denoting the tolerance criterion, and

$$\Lambda^k = ||\mathbf{r}^k|| = ||\mathbf{z}^{k+1} - \mathbf{z}^k|| \tag{4.13}$$

is the iteration residual at the k-th iteration.

4.1.2.2 Operator splitting methodology

Operator splitting techniques are mathematical procedures to decompose a system of coupled partial differential equations into simpler subproblems. It goes

Algorithm A-1 Algorithmic formulation of the block Gauss-Seidel method.

Initialize: $k = 0$; $\mathbf{z}^0 = \mathbf{z}_n$
1: **while** $||\mathbf{r}^k|| > \epsilon ||\mathbf{r}^0||$ **do**
2:    $\mathbf{y}^{k+1} = \mathcal{F}(\mathbf{z}^k)$
3:    $\mathbf{z}^{k+1} = \mathcal{G}\left(\mathbf{y}^{k+1}\right)$
4:    $\mathbf{r}^k = \mathbf{z}^{k+1} - \mathbf{z}^k$
5:    $k = k + 1$
6: **end while**
7: Go to next time-step: set $t_{n+1} = t_n + \Delta t$ and $\mathbf{z}_n = \mathbf{z}^{k+1}$

back to the 1950s and is often denoted as a *fractional step method* [174]. Much effort has been conducted to enhance operator split methods, and different improvements from recent years can be found in [53, 55, 65]. To begin with, a simple and arbitrary nonlinear system given by

$$\dot{\mathbf{z}}(t) = \mathcal{A}(\mathbf{z}), \quad \mathbf{z}(t = 0) = \mathbf{z}_0 \tag{4.14}$$

is taken under consideration, where $\mathcal{A}$ defines a nonlinear operator. For multi-physical problems, this operator represents different dependencies and physical effects between the fields. Operator splitting means, according to [66], that $\mathcal{A}$ is divided into $N = p + 1$ sub-operators such that

$$\mathcal{A} = \mathcal{A}_1 + \mathcal{A}_2 + ... + \mathcal{A}_N = \sum_{i=1}^{N} \mathcal{A}_i, \tag{4.15}$$

where the order of the method is given by p. For the widely used case that $p = 1$ holds, the first-order accurate operator split $\mathcal{A} = \mathcal{A}_1 + \mathcal{A}_2$ is obtained.

After splitting the operator, the time interval is divided into sub-intervals (time-steps) and linked over the initial conditions. The general procedure is then as follows: First, the problem is solved using the first operator $\mathcal{A}_1$ with the original initial condition. After that, the solution is taken as the initial condition at the end of the time-step for the second problem where $\mathcal{A}_2$ is employed. Accordingly, for the time interval $[t_n, t_{n+1}]$, this results in

$$\dot{\mathbf{z}}_1 = \mathcal{A}_1(\mathbf{z}_1), \quad \mathbf{z}_1(t_n) = \mathbf{z}_n, \quad \dot{\mathbf{z}}_2 = \mathcal{A}_2(\mathbf{z}_2), \quad \mathbf{z}_2(t_n) = \mathbf{z}_1(t_{n+1}) \tag{4.16}$$

with $\mathbf{z}_{n+1} = \mathbf{z}_2(t_{n+1})$. Proceeding in this manner is known as *sequential splitting* [53], which can be viewed in the partitioned solution context as – according to the definitions of Section 4.1.2 – an explicit coupling scheme. *Iterated splitting* is also possible, see e.g. [54, 66], and is comparable to an implicit coupling procedure.

A well-known application of operator splitting to multi-physical problems was carried out by SIMO & MIEHE [153] who considered thermo-mechanical interaction. They introduced the so-called *isothermal split* which divides the coupled problem into an isothermal mechanical phase at constant temperature and a subsequent conduction step on a fixed configuration. For a thermo-elastic analysis,

the coupled system of PDEs is written in operator notation. Following [153], it reads

$$\dot{\mathbf{z}}(t) = \mathcal{A}(\vec{\chi}(\vec{X},t), \Theta(\vec{X},t)) \quad \rightarrow \quad \left\{ \begin{array}{c} \dot{\vec{\chi}} \\ \dot{\vec{v}} \\ \dot{\Theta} \end{array} \right\} = \left\{ \begin{array}{c} \vec{v} \\ \frac{1}{\rho_0} \operatorname{Div} \underset{\sim}{P}(\vec{\chi},\Theta) \\ -\frac{1}{c_\Theta \rho_0} \operatorname{Div} \vec{Q}(\vec{\chi},\Theta) + \frac{1}{c_\Theta} \gamma(\vec{\chi},\Theta) \end{array} \right\}.$$
$$\tag{4.17}$$

Employing an operator split $\mathcal{A} = \mathcal{A}_\mathrm{M} + \mathcal{A}_\Theta$ leads to a mechanical part and a thermal part

$$\mathcal{A}_\mathrm{M} = \left\{ \begin{array}{c} \vec{v} \\ \frac{1}{\rho_0} \operatorname{Div} \underset{\sim}{P}(\vec{\chi},\Theta) \\ 0 \end{array} \right\} \quad \text{and} \quad \mathcal{A}_\Theta = \left\{ \begin{array}{c} \vec{0} \\ \vec{0} \\ -\frac{1}{c_\Theta \rho_0} \operatorname{Div} \vec{Q}(\vec{\chi},\Theta) + \frac{1}{c_\Theta} \gamma(\vec{\chi},\Theta) \end{array} \right\} \tag{4.18}$$

that are described by the balance of linear momentum Eq. (2.37) and by the equation of heat conduction, see Eq. (2.88). Here, the operator split allows to decouple the physical fields and is applied prior to the discretization. This allows, for example, to discretize both fields independently of each other with different schemes or methods.

There are many articles related to the stability of those algorithms, and in the textbook of YANENKO [174], it is stated that this split is unconditionally stable if and only if the matrices of the discretized counterparts to the operators $\mathcal{A}_i$ are positive-definite. This issue was also discussed by ARMERO & SIMO [3], here with special respect to thermo-elastically coupled systems. It was shown that problems with strong coupling between the fields might be unstable even if the sub-problems are stable. To this end, they proposed to use an *adiabatic split* – demonstrated in [3] to be unconditionally stable when applying implicit time-stepping – in which the splitting procedure leads to

$$\dot{\mathbf{z}} = \left\{ \begin{array}{c} \dot{\vec{\chi}} \\ \dot{\vec{v}} \\ \Theta \dot{\xi} \end{array} \right\}, \quad \mathcal{A}_\mathrm{M} = \left\{ \begin{array}{c} \vec{v} \\ \frac{1}{\rho_0} \operatorname{Div} \underset{\sim}{P}(\vec{\chi},\Theta) \\ 0 \end{array} \right\} \quad \text{and} \quad \mathcal{A}_\Theta = \left\{ \begin{array}{c} \vec{0} \\ \vec{0} \\ -\frac{1}{\rho_0} \operatorname{Div} \vec{Q}(\vec{\chi},\Theta) \end{array} \right\}.$$
$$\tag{4.19}$$

Applying this split means that an adiabatic problem is solved to begin with, keeping the entropy $\xi(\vec{\chi},\Theta)$ constant, followed by a heat conduction phase. In contrast to the isothermal split, the adiabatic split is formulated by means of the energy equation, see Eq. (2.82), to describe the thermal field.

The consequences of this split become apparent when considering aspects of the numerical implementation with finite elements. During the mechanical phase, one has to take into account that $\Theta \dot{\xi} = 0$, which needs to be enforced in strong form, holds. In consequence, this means that a nonlinear equation has to be solved locally to begin with. More precisely, a so-called *intermediate tempera-*

ture has to be computed at every integration point in order to treat the balance of momentum in the weak form[2].

The obvious drawback of this split is that it is either necessary to implement the adiabatic split before it can be used, or at least to modify an existing code. Thus, for the vast majority of finite element codes, it not possible to apply this splitting procedure to a black box solver off the cuff. However, this split is unconditionally stable and can also be applied to finite strain thermoplasticity [4]. Further details regarding an efficient numerical implementation can also be found in [3].

4.2 Convergence acceleration and stabilization

As reported in the beginning of this chapter, the main drawback of a partitioned approach can be seen in its only conditional stability. Moreover, to avoid poor convergence rates or divergent behavior during iteration, several external methods can be employed to improve both the chances for stability and the convergence properties. These methods are henceforth denoted as *external acceleration methods* as they operate independently of the field solver. This has the advantage that the relating solver code must not be modified.

4.2.1 Vector prediction

The key feature of an efficient partitioned solution strategy is to balance all fields involved in as few iterations as possible. Thus, if the iterative coupling process is initialized at the beginning of every time-step with an initial guess that is close to the sought solution, equilibrium will be achieved faster. In order to determine an appropriate initial guess, converged solutions from previous time-steps are taken into consideration. They are used to predict the starting value by means of extrapolation methods. In [50] for instance, the prediction is based on polynomial vector extrapolation, allowing to incorporate data from $m = p + 1$ older time-steps. Another advantage is that this method is applicable to cases in which the time increment is adapted to the problem and therefore might change during simulation.

For a predictor of the polynomial order p, information from $m = p+1$ time-steps is needed. The components of the predicted initial guess $\mathbf{z}^*$ are extrapolated out of the following polynomial

$$z_i^* = c_0 + c_1\, t + c_2\, t^2 + ... c_p\, t^p = \sum_{j=0}^{p} c_j\, t^j \,. \tag{4.20}$$

In this equation, the extrapolation coefficients c_j are unknown and have to be determined. To this end, the results of m previous time-steps are taken, leading

---

[2]Being more specific, this intermediate temperature can be calculated by solving the ordinary differential equation $\Theta\dot{\xi} = 0$. The constitutive assumption for the entropy $\xi = -\partial\Psi/\partial\Theta$ is derived in Section 2.1.3.2. Further, this relation is also employed during the heat conduction phase.

to the following system of equations for the i-th component of the data vector $\mathbf{z}$:

$$\sum_{j=0}^{p} c_j\, t_n^j = (\mathbf{z}_n)_i\,, \quad \rightarrow \quad \mathbf{A}\mathbf{c} = \mathbf{b} \tag{4.21}$$

with

$$\mathbf{A} = \begin{pmatrix} 1 & t_0 & t_0^2 & \dots & t_0^p \\ 1 & t_1 & t_1^2 & \dots & t_1^p \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & t_m & t_m^2 & \dots & t_m^p \end{pmatrix}\,, \quad \mathbf{c} = \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_m \end{pmatrix}\,, \quad \mathbf{b} = \begin{pmatrix} (\mathbf{z}_0)_i \\ (\mathbf{z}_1)_i \\ \vdots \\ (\mathbf{z}_m)_i \end{pmatrix}\,, \quad m = p+1\,. \tag{4.22}$$

In general, this system of equations has to be solved for all components at the beginning of every time-step. Since the matrix $\mathbf{A} \in \mathbb{R}^{m \times m}$ is equivalent for all components, it is enough to compute its inverse once. Afterwards the coefficient vector $\mathbf{c} \in \mathbb{R}^m$ can be computed by a simple vector-matrix multiplication. It has to be noticed that $\mathbf{A}$ is small since $m = p + 1$ because the polynomial order is usually between $p = 1$ and $p = 4$. Additionally, in the case where a constant time-step size is chosen, the difference between two instants of time $\Delta t = t_n - t_{n-1}$ is constant. This simplifies the procedure considerably, and the predicted values can be directly computed by

$$\begin{aligned} p &= 1: \quad \mathbf{z}^* = 2\,\mathbf{z}_n - \mathbf{z}_{n-1}\,, & (4.23) \\ p &= 2: \quad \mathbf{z}^* = 3\,\mathbf{z}_n - 3\,\mathbf{z}_{n-1} + \mathbf{z}_{n-2}\,. & (4.24) \end{aligned}$$

Besides, there are several other possibilities to predict the starting value. DEG-ROOTE ET AL. [36] employed a second-order predictor of the form

$$\mathbf{z}^* = \frac{1}{2}\left(5\,\mathbf{z}_n - 4\mathbf{z}_{n-1} + \mathbf{z}_{n-2}\right)\,, \tag{4.25}$$

BRÄNDLI [15] mentions a least-square fit procedure that allows to consider more than the $m = p + 1$ time-steps, while PIPERNO [136] also takes the velocities into account, resulting in a first- and second-order predictor of

$$\begin{aligned} p &= 1: \quad \mathbf{z}^* = \mathbf{z}_n + \Delta t\, \dot{\mathbf{z}}_n \\ p &= 2: \quad \mathbf{z}^* = \mathbf{z}_n + \Delta t\left(\frac{3}{2}\dot{\mathbf{z}}_n - \frac{1}{2}\dot{\mathbf{z}}_{n-1}\right)\,. \end{aligned} \tag{4.26}$$

The great advantage of vector extrapolation is that the computer implementation is straightforward. Predicting the starting values is an effective method to increase both the chance of stability and computational efficiency of the partitioned approach. In addition, this method can also be applied to monolithic schemes to determine an initial guess for the Newton-Raphson procedure, as shown in [141, 126] for instance.

Vector prediction in the Gauss-Seidel iteration process is illustrated in Algorithm A-2.

Algorithm A-2 Algorithmic formulation of the Gauss-Seidel method with vector prediction.

Initialize: $k = 0$; Compute initial guess: $\mathbf{z}^0 = 3\,\mathbf{z}_n - 3\,\mathbf{z}_{n-1} + \mathbf{z}_{n-2}$

1: **while** $||\mathbf{r}^k|| > \epsilon ||\mathbf{r}^0||$ **do**
2:     $\mathbf{y}^{k+1} = \mathcal{F}(\mathbf{z}^k)$
3:     $\mathbf{z}^{k+1} = \mathcal{G}\left(\mathbf{y}^{k+1}\right)$
4:     $\mathbf{r}^k = \mathbf{z}^{k+1} - \mathbf{z}^k$
5:     $k = k + 1$
6: **end while**
7: Go to next time-step: set $t_{n+1} = t_n + \Delta t$ and $\mathbf{z}_n = \mathbf{z}^{k+1}$

4.2.2 Vector sequence interpretation

In the following, the solution vectors in every coupling iteration of an implicit coupling scheme are considered as a sequence of vectors $\mathcal{S}$ in $\mathbb{R}^d$ converging to an unknown limit $\mathbf{z}^*$. It is assumed, that this limit is exactly the solution that balances the coupled problem. The vector sequence reads

$$\mathcal{S} := \left\{\mathbf{z}^k, \mathbf{z}^{k-1}, ..., \mathbf{z}^0 : \mathbf{z}^i \in \mathbb{R}^d\right\}, \tag{4.27}$$

and can be accelerated by transforming it into another sequence

$$\mathcal{T} := \left\{\mathbf{v}^k, \mathbf{v}^{k-1}, ..., \mathbf{v}^0 : \mathbf{v}^i \in \mathbb{R}^d\right\}, \tag{4.28}$$

with vectors $\mathbf{v}^k$ that show better convergence properties. Hence, a sequence transformation is sought that converges faster than the initial one, but to the same limit $\mathbf{z}^*$, so that

$$\lim_{k \to \infty} \mathbf{v}^k - \mathbf{z}^* = \mathbf{0} \tag{4.29}$$

holds. As shown in [17], one can conclude that the new vector sequence $\mathcal{T}$ converges to $\mathbf{z}^*$ faster than $\mathcal{S}$ if and only if

$$\lim_{k \to \infty} \frac{|\mathbf{v}^k - \mathbf{z}^*|}{|\mathbf{z}^k - \mathbf{z}^*|} = \mathbf{0} \tag{4.30}$$

can be achieved. The convergence acceleration of vector sequences is a solid research field in mathematics, and many sequence transformations have been developed that exhibit better convergence properties. Since the vectors of the new sequence are constructed by means of an extrapolation, these procedures are often called *vector sequence extrapolation methods*. The basic concept is to use a certain number of vectors of the old sequence to generate a transformation into a new sequence. This transformation is based on a recursive relationship. One of the most popular methods was introduced by AITKEN [2] for scalar sequences. It is commonly known as *Aitken's Δ^2-method* and was extended to the vector case in [91]. Apart from Aitken's method, important contributions to this research

field were made, among others, by WYNN who proposed the so-called *vector ε-algorithm* [173] and by BREZINSKI who proposed a sequence accelerator known as *vector θ-algorithm*. In the subsequent sections, some of these methods are introduced and applied to accelerate the convergence of the partitioned coupling approach.

4.2.3   Aitken and related relaxation methods

To begin with, a class of procedures to accelerate the coupling process is introduced that can be summarized as *Aitken-type methods*. In the literature, there are several other terminologies for this class of accelerations methods, for example *Aitken relaxation* [107, 37], *Aitken extrapolation* [119] or *dynamic relaxation* [106]. All methods that are discussed in the following typically fall within these classifications.

4.2.3.1   *Classical Aitken's Δ^2 method*

Section 4.2.2 serves to introduce a vector sequence interpretation of the partitioned coupling strategy. One of the most famous methods to improve the convergence properties of sequences is the so-called *Aitken Δ^2-process*. This process is originally a nonlinear sequence transformation to accelerate the convergence of a scalar sequence that converges to an unknown limit. It is named after AITKEN, who published this transformation in 1926 [2]. For the scalar case, the new accelerated value reads

$$z^{k+1} = z^{k-1} - \frac{\Delta z^{k-1} \Delta z^k}{\Delta^2 z^k} , \tag{4.31}$$

where Δ is the *difference operator* leading to $\Delta z^k := z^k - z^{k-1}$. In the denominator, the second-order difference operator Δ^2 is used whose application results in

$$\Delta^2 z^k := \Delta z^k - \Delta z^{k-1} = z^k - 2\,z^{k-1} + z^{k-2} . \tag{4.32}$$

This operator is the reason why it is called *Aitken's Δ^2-method*. For the multidimensional case, there exist several possibilities to generalize this procedure to obtain a vectorial formulation. One way, probably the most straightforward possibility, is to use the *Moore-Penrose inverse* of a vector $\mathbf{a} \in \mathbb{R}^{d \times 1}$, see [135] for instance, which is defined as

$$\mathbf{a}^{-1} := \frac{\mathbf{a}}{\mathbf{a}^T \mathbf{a}} = \frac{\mathbf{a}}{\|\mathbf{a}\|_2^2} . \tag{4.33}$$

This gives rise to formulate Eq. (4.31) for a vector-valued sequence as

$$\mathbf{z}^{k+1} = \mathbf{z}^{k-1} - \frac{(\Delta \mathbf{z}^{k-1})^T \Delta^2 \mathbf{z}^k}{\|\Delta^2 \mathbf{z}^k\|_2^2} \Delta \mathbf{z}^k . \tag{4.34}$$

Regarding further extensions, another way to derive this expression will be discussed in the following, referring to [112]. Based on the solution vectors from

the three previous iterations $\{\mathbf{z}^k, \mathbf{z}^{k-1}, \mathbf{z}^{k-2}\}$ of a given sequence $\mathcal{S}$, which are the solution vectors from the three previous coupling iterations, the following ansatz for a new sequence is made

$$\mathbf{v}^{k-1} = \mathbf{z}^{k-2} + \omega^k \left(\mathbf{z}^{k-1} - \mathbf{z}^{k-2}\right) = \mathbf{z}^{k-2} + \omega^k \Delta \mathbf{z}^{k-1}, \tag{4.35}$$

$$\mathbf{v}^{k} = \mathbf{z}^{k-1} + \omega^k \left(\mathbf{z}^{k} - \mathbf{z}^{k-1}\right) = \mathbf{z}^{k-1} + \omega^k \Delta \mathbf{z}^{k}. \tag{4.36}$$

Therein, the scalar ω^k, which is known as the *relaxation parameter*, might change during iteration. The value of interest for ω^k is the one that minimizes the expression

$$\|\mathbf{v}^k - \mathbf{v}^{k-1}\| \rightarrow \min, \tag{4.37}$$

where the difference between both vectors results in

$$\mathbf{v}^k - \mathbf{v}^{k-1} = \mathbf{z}^{k-1} - \mathbf{z}^{k-2} + \omega^k \left(\mathbf{z}^k - \mathbf{z}^{k-1} - \mathbf{z}^{k-1} + \mathbf{z}^{k-2}\right) = \Delta \mathbf{z}^{k-1} + \omega^k \Delta^2 \mathbf{z}^k. \tag{4.38}$$

Minimizing this expression finally delivers the relaxation coefficient

$$\min_{\omega^k} \|\Delta \mathbf{z}^{k-1} + \omega^k \Delta^2 \mathbf{z}^k\|_2^2 \quad \rightarrow \quad \omega^k = -\frac{(\Delta \mathbf{z}^{k-1})^T \Delta^2 \mathbf{z}^k}{\|\Delta^2 \mathbf{z}^k\|_2^2}. \tag{4.39}$$

After setting $\mathbf{z}^{k+1} = \mathbf{v}^k$, the new relaxed vector coincides exactly with the one given in (4.34). In this method, the relaxation parameter ω^k is not constant, which is why the procedure is henceforth denoted as *dynamic Aitken relaxation* (DAR). Indeed, this method has an important constraint because it cannot be applied in every coupling iteration. To compute the new relaxation parameter, at least three vectors of the sequence must be known. Accordingly, this means that once the DAR has been applied, two new iterations are required to employ the DAR again. This can been seen as the main drawback of the procedure and was also discussed in by KÜTTLER [105]. On the other hand, a computer implementation is straightforward – and with regard to the computational effort, only dot products have to be performed.

In the literature, there exists a wide range of variants of Aitken's Δ^2-process for convergence acceleration. Some of them shall be briefly reviewed here. A detailed study was carried out by MACLEOD [112], who took a look at nine methods to accelerate the convergence of multi-dimensional vector sequences. The drawback is that none of them can be applied recursively, yet the optimal point of acceleration to reach the best results was discussed. Apart from this study, a book by BREZINSKI lists several other methods, see [16]. All processes and algorithms addressed in this section can be formulated to obtain the new accelerated solution $\mathbf{z}^{k+1}$ by means of the following structure

$$\mathbf{z}^{k+1} = \mathbf{a} + \omega^k \mathbf{b}. \tag{4.40}$$

Here, $\mathbf{b} \in \mathbb{R}^d$ and $\mathbf{a} \in \mathbb{R}^d$ are vectors, and ω^k is a scalar parameter, the recently introduced relaxation coefficient. Table 4.1 illustrates a selection of different methods to compute the new sequence vector $\mathbf{z}^{k+1}$.

Table 4.1: Different Δ^2 acceleration formulas for multi-dimensional vector sequences.

Method	a	ω^k	b	Reference
1	$\mathbf{z}^{k-1}$	$-\dfrac{(\Delta\mathbf{z}^{k-1})^T\Delta^2\mathbf{z}^k}{\|\Delta^2\mathbf{z}^k\|_2^2}$	$\Delta\mathbf{z}^k$	Aitken [2]
2	$\mathbf{z}^{k-1}$	$-\dfrac{(\Delta\mathbf{z}^{k-1})^T\Delta\mathbf{z}^{k-1}}{(\Delta\mathbf{z}^{k-1})^T\Delta^2\mathbf{z}^k}$	$\Delta\mathbf{z}^k$	Graves-Morris [69]
3	$\mathbf{z}^k$	$-\dfrac{(\Delta\mathbf{z}^k)^T\Delta\mathbf{z}^k}{(\Delta^2\mathbf{z}^k)^T(\mathbf{z}^k-\mathbf{z}^{k-2})}$	$\mathbf{z}^k-\mathbf{z}^{k-2}$	Iguchi in [112]
4	$\mathbf{z}^k$	$-\dfrac{(\Delta\mathbf{z}^k)^T\Delta\mathbf{z}^k}{(\Delta\mathbf{z}^k)^T\Delta^2\mathbf{z}^k}$	$\Delta\mathbf{z}^k$	Zienkiewicz [177]
5	$\mathbf{z}^k$	$-\dfrac{(\Delta\mathbf{z}^k)^T\Delta\mathbf{z}^{k-1}}{(\Delta\mathbf{z}^{k-1})^T\Delta^2\mathbf{z}^k}$	$\Delta\mathbf{z}^k$	Jennings [95]
6	$\mathbf{z}^k$	$-\dfrac{(\Delta\mathbf{z}^k)^T\Delta\mathbf{z}^k}{(\Delta^2\mathbf{z}^k)^T\Delta^2\mathbf{z}^k}$	$\Delta^2\mathbf{z}^k$	Arthur in [112]

The very simplest way to relax the solution is to use a fixed or constant relaxation coefficient in the range of $0 < \omega < 2$. This procedure is known as *numerical relaxation* or, to be more precise, as *successive over relaxation* (SOR) if $\omega > 1$ and *successive under relaxation* (SUR) if $\omega < 1$ is chosen. The threshold $\omega \in (0, 2)$ for the relaxation parameter is originally related to the theorem of KAHAN, who considered the solution of systems of equations with the Gauss-Seidel method, see [98]. Finally, an algorithmic sketch of the Gauss-Seidel iteration process accelerated by dynamic Aitken relaxation (DAR) is given in Algorithm A-3.

4.2.3.2   Secant Δ^2 method

The secant method follows directly from Aitken's Δ^2-method and can therefore been seen as an extension. IRONS & TUCK [91] published a modification that allows to apply the Aitken relaxation in every iteration. This procedure, henceforth denoted as *dynamic secant relaxation* (DSR), will be briefly outlined in the following. DSR is widely used in the field of fluid-structure interaction and has been proven suitable to solve problems with a strong interaction among each other, see e.g. [106, 37, 119]. In addition, the convergence of volume-coupled problems that involve large vectors can also be improved by means of this method, as reported in [50, 51].

The key to a recursive application in every coupling iteration lies in using different and independent vector pairs to construct the difference operator $\Delta\mathbf{z}$. To this end, it is now distinguished between modified (relaxed) and unmodified (unrelaxed) values. All unmodified vectors are henceforth denoted with a tilde ($\tilde{\mathbf{z}}^k$) and are equivalent to the solution vector of the fixed-point problem, see Eq. (4.9).

Algorithm A-3 Algorithmic formulation of the Gauss-Seidel method with dynamic Aitken relaxation.

Initialize: $k = 0$; $j = 1$; Compute initial guess: $\mathbf{z}^0 = 3\,\mathbf{z}_n - 3\,\mathbf{z}_{n-1} + \mathbf{z}_{n-2}$

1: **while** $||\mathbf{r}^k|| > \epsilon||\mathbf{r}^0||$ **do**
2:      $\mathbf{y}^{k+1} = \mathcal{F}(\mathbf{z}^k)$
3:      $\mathbf{z}^{k+1} = \mathcal{G}\left(\mathbf{y}^{k+1}\right)$
4:      $\mathbf{r}^k = \tilde{\mathbf{z}}^{k+1} - \mathbf{z}^k$
5:      **if** $j = 2$ **then**
6:          Compute ω^k from Table 4.1
7:          Compute $\mathbf{a}$ and $\mathbf{b}$ from Table 4.1
8:          $\mathbf{z}^{k+1} = \mathbf{a} + \omega^k\,\mathbf{b}$
9:          $j = 0$
10:     **else**
11:         $\mathbf{z}^{k+1} = \tilde{\mathbf{z}}^{k+1}$
12:     **end if**
13:     $j = j + 1$; $k = k + 1$;
14: **end while**
15: Go to next time-step: set $t_{n+1} = t_n + \Delta t$ and $\mathbf{z}_n = \mathbf{z}^{k+1}$

Again, the ansatz made in (4.35) is taken, but slightly changed to

$$
\begin{aligned}
\mathbf{v}^{k-1} &= \mathbf{z}^{k-1} + \omega^k\left(\tilde{\mathbf{z}}^k - \mathbf{z}^{k-1}\right) = \mathbf{z}^{k-1} + \omega^k\Delta\tilde{\mathbf{z}}^k\,, & (4.41) \\
\mathbf{v}^k &= \mathbf{z}^k + \omega^k\left(\tilde{\mathbf{z}}^{k+1} - \mathbf{z}^k\right) = \mathbf{z}^k + \omega^k\Delta\tilde{\mathbf{z}}^{k+1}\,. & (4.42)
\end{aligned}
$$

According to the previous Section 4.2.3.1, the relaxation coefficient is determined by minimizing the expression $\|\mathbf{v}^k - \mathbf{v}^{k-1}\|$, leading to

$$
\omega^k = -\frac{(\Delta\mathbf{z}^k)^T\Delta^2\tilde{\mathbf{z}}^{k+1}}{\|\Delta^2\tilde{\mathbf{z}}^{k+1}\|_2^2}\,. \tag{4.43}
$$

Assuming that the relaxed vector from the previous iteration is computed by

$$
\mathbf{z}^k = \mathbf{z}^{k-1} + \omega^{k-1}\Delta\tilde{\mathbf{z}}^k \quad \rightarrow \quad \mathbf{z}^k - \mathbf{z}^{k-1} = \Delta\mathbf{z}^k = \omega^{k-1}\Delta\tilde{\mathbf{z}}^k\,, \tag{4.44}
$$

the expression $\Delta\mathbf{z}^k$ is used to substitute the difference in (4.43) so that the recursively formulated relaxation coefficient finally reads

$$
\omega^k = -\omega^{k-1}\frac{(\Delta\tilde{\mathbf{z}}^k)^T\Delta^2\tilde{\mathbf{z}}^{k+1}}{\|\Delta^2\tilde{\mathbf{z}}^{k+1}\|_2^2}\,. \tag{4.45}
$$

This coefficient is recomputed in every coupling iteration, so that the new relaxed vector follows by setting $\mathbf{z}^{k+1} = \mathbf{v}^k$ to

$$
\mathbf{z}^{k+1} = \mathbf{z}^k - \omega^{k-1}\frac{(\Delta\tilde{\mathbf{z}}^k)^T\Delta^2\tilde{\mathbf{z}}^{k+1}}{\|\Delta^2\tilde{\mathbf{z}}^{k+1}\|_2^2}\Delta\tilde{\mathbf{z}}^{k+1} = \mathbf{z}^k + \omega^k\Delta\tilde{\mathbf{z}}^{k+1}\,. \tag{4.46}
$$

The formula for the relaxation coefficient, see Eq. (4.45), has almost the same structure as the coefficient in given by Eq. (4.39). In fact, they are different since

the former one does not incorporate the relaxation parameter from the previous iteration and does not distinguish between relaxed and unrelaxed vectors. It has to be noticed that in the beginning at least two iterations have to be performed before the scheme can be employed first. An alternative derivation can be found in [105]. The usage of the dynamic secant method in the Gauss-Seidel procedure is outlined in Algorithm A-4.

Algorithm A-4 Algorithmic formulation of the Gauss-Seidel method accelerated by dynamic secant relaxation.

Initialize: $k = 0$; Compute initial guess: $\mathbf{z}^0 = 3\,\mathbf{z}_n - 3\,\mathbf{z}_{n-1} + \mathbf{z}_{n-2}$; set $\omega^0 = \min\left(\max(0.01, |\omega^*|), 2\right)$
 1: **while** $||\mathbf{r}^k|| > \epsilon ||\mathbf{r}^0||$ **do**
 2:    $\mathbf{y}^{k+1} = \mathcal{F}(\mathbf{z}^k)$
 3:    $\tilde{\mathbf{z}}^{k+1} = \mathcal{G}\left(\mathbf{y}^{k+1}\right)$
 4:    $\mathbf{r}^k = \tilde{\mathbf{z}}^{k+1} - \mathbf{z}^k$
 5:    **if** $k \leq 1$ **then**
 6:       $\omega^k = \omega^0$
 7:       $\mathbf{z}^{k+1} = \mathbf{z}^k + \omega^0\left(\tilde{\mathbf{z}}^{k+1} - \mathbf{z}^k\right)$
 8:    **else**
 9:       $\omega^k = -\omega^{k-1}\dfrac{(\Delta\tilde{\mathbf{z}}^k)^T \Delta^2\tilde{\mathbf{z}}^{k+1}}{\|\Delta^2\tilde{\mathbf{z}}^{k+1}\|_2^2}$
10:       $\mathbf{z}^{k+1} = \mathbf{z}^k + \omega^k\left(\tilde{\mathbf{z}}^{k+1} - \mathbf{z}^k\right)$
11:    **end if**
12:    $k = k + 1$
13: **end while**
14: Go to next time-step: set $t_{n+1} = t_n + \Delta t$ and $\mathbf{z}_n = \mathbf{z}^{k+1}$

In the literature, there are some other expressions for the relaxation coefficient given by Eq. (4.45). In many works, see [120, 150, 64] for instance, the relaxation coefficient is determined by the following notation

$$\nu^k = \nu^{k-1} + (\nu^{k-1} - 1)\frac{(\Delta\tilde{\mathbf{z}}^{k+1})^T(\Delta\tilde{\mathbf{z}}^k - \Delta\tilde{\mathbf{z}}^{k+1})}{\|\Delta\tilde{\mathbf{z}}^k - \Delta\tilde{\mathbf{z}}^{k+1}\|_2^2}. \tag{4.47}$$

Therein, the relation $\nu^k = 1 - \omega^k$ is introduced – and by using $\Delta^2\tilde{\mathbf{z}}^{k+1} = -(\Delta\tilde{\mathbf{z}}^k - \Delta\tilde{\mathbf{z}}^{k+1})$, expression Eq.(4.47) can be rearranged to

$$\nu^k = \nu^{k-1} - (\nu^{k-1} - 1)\frac{(\Delta\tilde{\mathbf{z}}^{k+1})^T\Delta^2\tilde{\mathbf{z}}^{k+1}}{\|\Delta^2\tilde{\mathbf{z}}^{k+1}\|_2^2}. \tag{4.48}$$

This can further be simplified by using the expression $\omega^k = 1 - \nu^k$ for the relaxation coefficient to

$$\omega^k = \omega^{k-1}\left[1 - \frac{(\Delta\tilde{\mathbf{z}}^{k+1})^T\Delta^2\tilde{\mathbf{z}}^{k+1}}{\|\Delta^2\tilde{\mathbf{z}}^{k+1}\|_2^2}\right]. \tag{4.49}$$

Moreover, this formula is completely equivalent to the one defined by Eq. (4.45) which can be proved by showing that

$$\omega^k = -\omega^{k-1}\frac{(\Delta\tilde{\mathbf{z}}^k)^T\Delta^2\tilde{\mathbf{z}}^{k+1}}{\|\Delta^2\tilde{\mathbf{z}}^{k+1}\|_2^2} = \omega^{k-1}\left[1 - \frac{(\Delta\tilde{\mathbf{z}}^{k+1})^T\Delta^2\tilde{\mathbf{z}}^{k+1}}{\|\Delta^2\tilde{\mathbf{z}}^{k+1}\|_2^2}\right] \tag{4.50}$$

holds. Eliminating the denominator

$$-(\Delta\tilde{\mathbf{z}}^k)^T\Delta^2\tilde{\mathbf{z}}^{k+1} = \|\Delta^2\tilde{\mathbf{z}}^{k+1}\|_2^2 - (\Delta\tilde{\mathbf{z}}^{k+1})^T\Delta^2\tilde{\mathbf{z}}^{k+1} \tag{4.51}$$

and setting this equation to zero leads to

$$(\Delta\tilde{\mathbf{z}}^{k+1} - \Delta\tilde{\mathbf{z}}^k)^T\Delta^2\tilde{\mathbf{z}}^{k+1} - \|\Delta^2\tilde{\mathbf{z}}^{k+1}\|_2^2 = 0\,. \tag{4.52}$$

This finally implies that

$$(\Delta^2\tilde{\mathbf{z}}^{k+1})^T\Delta^2\tilde{\mathbf{z}}^{k+1} - \|\Delta^2\tilde{\mathbf{z}}^{k+1}\|_2^2 = \|\Delta^2\tilde{\mathbf{z}}^{k+1}\|_2^2 - \|\Delta^2\tilde{\mathbf{z}}^{k+1}\|_2^2 = 0 \tag{4.53}$$

can be obtained. In summary, both definitions for the relaxation coefficient given by Eq. (4.49) and Eq. (4.45) are possible and yield exactly the same results. Implementing this method is very simple and, concerning computational efficiency, only dot products must be calculated. By definition, the relaxation coefficients are limited to $\omega^k \in (0,2)$, see Section 4.2.3.1. At the beginning of every time-step, i.e. in the first coupling iteration of the partitioned approach, the relaxation coefficient from the converged solution of the previous time-step ω^* is taken. As this value is limited too, the starting value of the relaxation coefficient is defined as $\omega_0 = \min\left(\max(0.01, |\omega^*|), 2\right)$, and $\omega_0 = 1$ is chosen for the very first time-step.

4.2.3.3 Generalized version of Aitken's method

In [18] and in his book [16], BREZINSKI proposed an extension of the Δ^2-method which can be seen as a generalization of the Aitken process. In particular, his methods serve to accelerate the solution of nonlinear systems of equations by iterative fixed-point methods. He considered the following algorithm, called Δ^m-method[3]

$$\mathbf{z}^{k+1} = \mathbf{z}^{k-1} - \lambda^m\Delta^m\mathbf{z}^k \quad\text{with}\quad \lambda^m = (-1)^m\omega^k\,. \tag{4.54}$$

In this algorithm, the superscript m denotes the *order* of the method. For example, if $m = 1$ is chosen, the difference operator Δ^m leads to $\Delta^1\mathbf{z}^k = \mathbf{z}^k - \mathbf{z}^{k-1}$, and this method coincides exactly with the classical Aitken method, as described in Section 4.2.3.1. For the general case with an increasing superscript m, the difference operator reads

$$\Delta^m\mathbf{z}^k = \Delta^{m-1}\mathbf{z}^k - \Delta^{m-1}\mathbf{z}^{k-1}\,. \tag{4.55}$$

---

[3]Originally, BREZINSKI named the method Δ^k-method. However, as k is reserved for the iteration counter, the letter m is used here for clarity reasons.

Thus, more information is required to generate the new relaxed vector $\mathbf{z}^{k+1}$, i.e. the sequence of vectors must contain $m+2$ entries. According to [18, 16], the relaxation coefficient takes the form

$$\omega^k = (-1)^m \frac{(\Delta \mathbf{z}^{k-1})^T \Delta^{m+1} \mathbf{z}^k}{\|\Delta^{m+1} \mathbf{z}^k\|_2^2} \,. \tag{4.56}$$

The proposed algorithm gives rise to a generalization of the secant method, again distinguishing between relaxed and unrelaxed vectors so that the new vector is generated by

$$\mathbf{z}^{k+1} = \mathbf{z}^k - \lambda^m \Delta^m \tilde{\mathbf{z}}^{k+1} \,. \tag{4.57}$$

Proceeding in a similar way as in Section 4.2.3.2, one can deduce that

$$(-1)^m \Delta \mathbf{z}^k = -\omega^{k-1} \Delta^m \tilde{\mathbf{z}}^k \,, \tag{4.58}$$

which leads to the recursively formulated relaxation coefficient

$$\omega^k = -\omega^{k-1} \frac{(\Delta^m \tilde{\mathbf{z}}^k)^T \Delta^{m+1} \tilde{\mathbf{z}}^{k+1}}{\|\Delta^{m+1} \tilde{\mathbf{z}}^{k+1}\|_2^2} \,. \tag{4.59}$$

This procedure offers the possibility to use data from all previous coupling iterations. This means, however, that more data vectors need to be stored compared the DAR and DSR procedure where it is enough to memorize only three data vectors. With a consecutive iteration process, the order m of the method can be adapted to the current number of coupling iterations. Henceforth, this algorithm is denoted as *generalized dynamic secant relaxation* (GDSR). The integration into the Gauss-Seidel iteration process can be done similar to the dynamic secant relaxation method, see Algorithm A-4.

4.2.3.4   Extrapolation methods

Beside the previously described methods and algorithms, there exist several other procedures which shall be briefly mentioned. In a performance study for nonlinear algorithms applied to FSI [119], a so-called *line extrapolation method* is mentioned. It can be regarded as an extension of the dynamic Aitken relaxation formula given by Eq. (4.34). The algorithm reads

$$\mathbf{z}^{k+1} = \omega^k \mathbf{z}^k + (1 - \omega^k) \mathbf{z}^{k-1} - \zeta \left[\omega^k \Delta \tilde{\mathbf{z}}^{k+1} + (1 - \omega^k) \Delta \tilde{\mathbf{z}}^k \right] \tag{4.60}$$

in which ζ is a line search parameter to be specified by the user. It serves to avoid a situation in which the search space is limited to a single line, and it must be in the range of $\zeta \in (0, 1)$ [119]. The coefficient ω^k is again the relaxation coefficient and can be computed by means of Eq. (4.39). Similar to the dynamic Aitken relaxation, it can easily be implemented. Also, the additional computational effort is negligible. From now on, the line extrapolation method is abbreviated by LE.

Two other methods shall be briefly mentioned. In view of the Aitken relaxation, the relaxation coefficient ω^k can be calculated in the direction of the steepest descent, which is known as the *steepest descent method*, see [106]. The main drawback is that the search direction pointing to the steepest descent leads to an additional Jacobian which needs to be calculated or at least approximated. This is the bottleneck of this procedure, which will thus not be considered in the following. In [107], so-called *vector extrapolation methods* are employed to solve FSI problems. Such methods have been developed to accelerate vector sequences, similar to Aitken's method. However, [107] clearly shows that Aitken's procedure is often the better choice, so that vector extrapolation methods are not taken under consideration. A detailed introduction is given in [107] and also in the literature cited therein.

4.2.4 Vector sequence acceleration methods

This section focuses on three famous methods for the acceleration of vector sequences. They are formulated in such a way that they can be applied to improve the convergence of the partitioned coupling scheme. More information concerning the applied methods can be found in Appendix A.2. There, it is further shown how to derive a notation allowing to use these algorithms as a convergence acceleration method.

4.2.4.1   *Wynn's ϵ-method*

One of the most famous algorithms to accelerate scalar sequences is *Wynn's ε-algorithm*, which was discovered by WYNN in the early 1950s. First applied to scalar sequences, it was extended to the vector case and published as the *vector ε-algorithm* (VEA) [173]. If this procedure is applied to the three last iterates $\{\mathbf{z}^k, \mathbf{z}^{k-1}, \mathbf{z}^{k-2}\}$ of the partitioned coupling scheme, one obtains the new, improved solution by the following formula

$$\mathbf{z}^{k+1} = \mathbf{z}^k + \frac{\|\Delta\mathbf{z}^{k-1}\|_2^2}{\|\Delta^2\mathbf{z}^k\|_2^2}\,\Delta\mathbf{z}^k - \frac{\|\Delta\mathbf{z}^k\|_2^2}{\|\Delta^2\mathbf{z}^k\|_2^2}\,\Delta\mathbf{z}^{k-1}\,. \tag{4.61}$$

A detailed derivation of this formula is given in the Appendix A.2.1. Similar to dynamic Aitken relaxation (DAR), it can be applied after two new iterations.

Another interesting scheme that is mentioned in the literature is the *topological ε-algorithm* (TEA), see [16], for instance. Again, taking the three last solution vectors into account, the acceleration formula reads

$$\mathbf{z}^{k+1} = \mathbf{z}^k + \frac{\Delta\mathbf{z}^k}{(\Delta\mathbf{z}^k)^T\left(\dfrac{\mathbf{h}}{\mathbf{h}^T\Delta\mathbf{z}^k} - \dfrac{\mathbf{h}}{\mathbf{h}^T\Delta\mathbf{z}^{k-1}}\right)}\,, \tag{4.62}$$

where $\mathbf{h}$ is an arbitrary non-zero vector with the same dimension as the data vector $\mathbf{z}^k$.

4.2.4.2   Brezinski's Θ-method

Another promising vector sequence acceleration method is BREZINSKI'S *vector* Θ-*algorithm* (VTA) [16]. In order to accelerate the coupling iteration process, the last four iterates of a sequence $\{\,\mathbf{z}^k,\,\mathbf{z}^{k-1},\mathbf{z}^{k-2},\,\mathbf{z}^{k-3}\,\}$ are required. Based on the VTA, the new solution is calculated by

$$\mathbf{z}^{k+1} = \mathbf{z}^k + \frac{\left(\Delta\bar{\mathbf{z}}^k - \Delta\bar{\mathbf{z}}^{k-1}\right)^T \left(\Delta\bar{\mathbf{z}}^k - 2\Delta\bar{\mathbf{z}}^{k-1} + \Delta\bar{\mathbf{z}}_{n-2}\right)}{\|\Delta\bar{\mathbf{z}}^k - 2\Delta\bar{\mathbf{z}}^{k-1} + \Delta\bar{\mathbf{z}}^{k-2}\|_2^2}\left(\Delta\bar{\mathbf{z}}^k - \Delta\bar{\mathbf{z}}^{k-1}\right)\ ,\ \ (4.63)$$

where $\Delta\bar{\mathbf{z}}^k$ is defined as[4]

$$\Delta\bar{\mathbf{z}}^k := \frac{\mathbf{z}^k - \mathbf{z}^{k-1}}{\|\mathbf{z}^k - \mathbf{z}^{k-1}\|_2^2}\ . \tag{4.64}$$

In [16], another version is proposed; the so-called *generalized* Θ-*algorithm* (GTA), which can be written as

$$\mathbf{z}^{k+1} = \mathbf{z}^k - \frac{\mathbf{g}^T\Delta\mathbf{z}^{k-1}}{\mathbf{g}^T\Delta\mathbf{d}^k}\,\mathbf{d}^{k-1}\quad\text{with}\quad \mathbf{d}^k = \frac{\Delta\mathbf{z}^{k-1}}{\left(\dfrac{\mathbf{h}}{\mathbf{h}^T\Delta\mathbf{z}^k} - \dfrac{\mathbf{h}}{\mathbf{h}^T\Delta\mathbf{z}^{k-1}}\right)^T\Delta\mathbf{z}^{k-1}}\ . \tag{4.65}$$

In this formula, $\mathbf{h}$ and $\mathbf{g}$ are non-zero vectors, which can be chosen arbitrarily with the same dimension as $\mathbf{z}$. The vector $\mathbf{d}^k$ is introduced for clarity reasons. A detailed way to derive Eq. (4.63) and (4.65) is given in Appendix A.2.2. Note that both algorithms can be applied after three new coupling iterations have been carried out.

4.2.4.3   W-algorithm

As the third vector sequence acceleration method under investigation, the *W-algorithm* proposed by OSADA [131] is considered. Considering the four last iterates $\{\,\mathbf{z}^k,\,\mathbf{z}^{k-1},\mathbf{z}^{k-2},\,\mathbf{z}^{k-3}\,\}$ the *Euclidean W-transformation* [131] (EWT) is introduced. For this transformation, the acceleration formula takes the form

$$\mathbf{z}^{k+1} = \mathbf{z}^{k-1} - \frac{(\Delta\mathbf{z}^k)^T\Delta^2\mathbf{z}^{k-1}}{(\Delta\mathbf{z}^k)^T\Delta^2\mathbf{z}^{k-1} - (\Delta\mathbf{z}^{k-2})^T\Delta^2\mathbf{z}^k}\,\Delta\mathbf{z}^{k-1}\ . \tag{4.66}$$

There is a further version, known as *vector W-transformation* (VWT) [131], which can be rewritten in a similar way to accelerate the coupling process by

$$\mathbf{z}^{k+1} = \mathbf{z}^{k-1}_{j-1} + \left(1 - \frac{(\Delta\mathbf{z}^{k-1})^T\Delta\mathbf{z}^{k-2}}{(\Delta\mathbf{z}^{k-2})^T\Delta\mathbf{z}^{k-2}}\right)\left(\frac{\Delta\mathbf{z}^k}{\|\Delta\mathbf{z}^k\|^2} - 2\frac{\Delta\mathbf{z}^{k-1}}{\|\Delta\mathbf{z}^{k-1}\|^2} + \frac{\Delta\mathbf{z}^{k-2}}{\|\Delta\mathbf{z}^{k-2}\|^2}\right)^{-1}\ . \tag{4.67}$$

In this equation, the vector inverse of the last term needs to be computed with Eq. (4.33). Further discussions to calculate Eqns. (4.66)–(4.67) are shifted to the Appendix A.2.3. It is to be noted that both algorithms can be employed after three new coupling iterations have been performed.

---

[4]This expression coincides with the vector inverse $\Delta\bar{\mathbf{z}}^k = \left(\Delta\mathbf{z}^k\right)^{-1}$.

4.2.5 Newton and quasi-Newton methods

The iterative Gauss-Seidel coupling process of the partitioned approach can be written as a fixed-point or root-finding formulation, see Section 4.1.2. The fixed-point solution of two fields $\tilde{\mathbf{y}} = \mathcal{F}(\mathbf{z})$ and $\tilde{\mathbf{z}} = \mathcal{G}(\mathbf{y})$ are taken into consideration. The solution operators $\mathcal{F}$ and $\mathcal{G}$ can be interpreted as specialized solvers for the corresponding fields. In a fixed-point formulation, the coupled problem can be written as

$$\mathcal{R}(\mathbf{z}) = \mathcal{G}\left(\mathcal{F}(\mathbf{z}^k)\right) - \mathbf{z} = \mathbf{0}\,, \tag{4.68}$$

where $\mathcal{R}(\mathbf{z})$ is termed as the residual operator. In discrete form, the residual operator defines a nonlinear system of equations $\mathbf{R}(\mathbf{z}) = \mathbf{0}$, and the Newton-Raphson method can be applied to solve this system. The objective is to correct the current solution in such a way that both fields reach equilibrium. For the k-th fixed-point or coupling iteration, this leads to

$$\left.\frac{\partial \mathbf{R}}{\partial \mathbf{z}}\right|_{\mathbf{z}=\mathbf{z}^k} \Delta \mathbf{z}^k = -\mathbf{R}(\mathbf{z}^k)\,. \tag{4.69}$$

Accordingly, the new, corrected solution $\mathbf{z}^{k+1}$ obeys

$$\mathbf{z}^{k+1} = \mathbf{z}^k + \Delta \mathbf{z}^k\,. \tag{4.70}$$

Applying Newton-Raphson iterations to balance the fields poses numerical difficulties and has several drawbacks. First of all, the computation of the Jacobian matrix is not straightforward, in some cases maybe almost impossible. In particular, cases involving different solvers or discretization schemes will turn out to be problematic. For this situation, it is not possible to give an exact representation of the Jacobian. Even in the simplest case where both fields are solved on exactly the same discretization, knowledge about the Jacobian is indispensable. Moreover, assuming access to the exact Jacobian, an additional linear system of equation (4.69) must be solved in every coupling iteration. This might lead to a considerable increase in computation time since the system is of the order $\mathbb{R}^{d \times d}$. It is these two facts that deprive the partitioned approach of its two major advantages – flexibility and computational efficiency.

On the other hand, the benefits of the Newton methods are that they are able to significantly increase the chance of a stable coupling procedure and to reduce the number of fixed-point iterations. Taking advantage of these methods would clearly improve the partitioned coupling strategy and the convergence properties. To this end, several approximation techniques have been developed, which are known as *quasi-* or *inexact-Newton methods*. In the following, some of these methods are reviewed briefly.

4.2.5.1 *Quasi-Newton reduced order method*

DEGROOTE ET AL. [36] propose an *interface quasi-Newton technique* (IQN) for the solution of surface-coupled FSI problems. This technique is based on the idea of

approximating the Jacobian by means of reduced-order models [165] and was further extended in several articles, see e.g. [37, 38, 39]. Additionally, it was shown in [50] for instance, that this technique can also be applied to volume-coupled problems involving huge sets of data. To this end, the procedure is henceforth denoted as *quasi-Newton method*, abbreviated by QN. Applying the QN method to the fixed-point procedure avoids the necessity to compute the exact Jacobian and to solve a (large) system of equations.

Following [36], the underlying idea of the QN is given by considering the change of the residual, i.e. the difference between two consecutive residuals

$$\Delta \mathbf{R} = \mathbf{R}^{k+1} - \mathbf{R}^k \tag{4.71}$$

and by assuming that the new residual $\mathbf{R}^{k+1} = \mathbf{R}^k + \Delta \mathbf{R}$ should be minimal. This assumption is in agreement with the fixed-point iteration procedure, see Eq. (4.68), where a zero residual operator is definitely the desired state. For the discrete case, this can be achieved by minimizing

$$\|\mathbf{R}^k + \Delta \mathbf{R}\| \quad \to \quad \min . \tag{4.72}$$

Following [36], the next step is to approximate the change in the residual $\Delta \mathbf{R}$ by means of a linear combination of residual increments from the previous iterations

$$\Delta \mathbf{R}(\boldsymbol{\alpha}^k) \approx \sum_{i=0}^{k-1} \alpha_i^k \Delta \mathbf{R}^i = \alpha_0^k \Delta \mathbf{R}^0 + \alpha_1^k \Delta \mathbf{R}^1 + ... + \alpha_{k-1}^k \Delta \mathbf{R}^{k-1} . \tag{4.73}$$

The outcome of this approximation is that it is now necessary to determine an unknown vector of coefficients $\boldsymbol{\alpha}^k \in \mathbb{R}^{(k-1)}$. This can be accomplished by inserting Eq. (4.73) into Eq. (4.72), resulting in an over-determined minimization problem which is solved in a least-square fit sense

$$\min_{\boldsymbol{\alpha}^k} \mathbf{R}^{k+1} \quad \to \quad \min_{\boldsymbol{\alpha}^k} \left\| \mathbf{R}^k + \sum_{i=0}^{k-1} \alpha_i^k \Delta \mathbf{R}^i \right\|_2^2 . \tag{4.74}$$

Before proceeding with the computation of the approximated Jacobian, emphasis is placed on the residual increments $\Delta \mathbf{R}^i$. Two possible ways to generate them are contemplated

$$\Delta \mathbf{R}^i = \mathbf{R}^k - \mathbf{R}^i \quad \text{or} \quad \Delta \mathbf{R}^i = \mathbf{R}^{i+1} - \mathbf{R}^i \quad \text{with} \quad i = 0, 1, 2, ..., k-1 . \tag{4.75}$$

The first case is used in [36], whereas the latter definition is applied in [50] which will henceforth be used in this thesis. The minimization problem in Eq. (4.73) requires the solution of a linear system of equations, yet the additional computational effort remains acceptable because $\boldsymbol{\alpha}^k \in \mathbb{R}^{k-1}$ has only $k-1$ entries. The new corrected solution was defined as $\mathbf{z}^{k+1} = \mathbf{z}^k + \Delta \mathbf{z}^k$.

The main goal in this context is to find the new increment $\Delta \mathbf{z}^k$, which can be computed by

$$\Delta \mathbf{R} = \tilde{\mathbf{z}} - \mathbf{z}^k - \tilde{\mathbf{z}}^k + \mathbf{z}^{k-1} = \Delta \tilde{\mathbf{z}} - \Delta \mathbf{z}^k \quad \to \quad \Delta \mathbf{z}^k = \Delta \tilde{\mathbf{z}} - \Delta \mathbf{R} . \tag{4.76}$$

Following [36], the difference of the unmodified (fixed-point) solution $\Delta\tilde{\mathbf{z}}$ corresponds to the change in the residual $\Delta\mathbf{R}$ such that it can be calculated similar to Eq. (4.73) by using the known coefficients $\boldsymbol{\alpha}^k$

$$\Delta\tilde{\mathbf{z}}^k = \sum_{i=0}^{k-1} \alpha_i^k \Delta\tilde{\mathbf{z}}^i = \alpha_0^k \Delta\tilde{\mathbf{z}}^0 + \alpha_1^k \Delta\tilde{\mathbf{z}}^1 + ... + \alpha_{k-1}^k \Delta\tilde{\mathbf{z}}^{k-1} \,. \tag{4.77}$$

Again, the definition of $\Delta\tilde{\mathbf{z}}^i$ has to be considered carefully. Similar to the definition of the residual increment in Eq. (4.75), there are two possible ways

$$\Delta\tilde{\mathbf{z}}^i = \tilde{\mathbf{z}}^k - \tilde{\mathbf{z}}^i \quad \text{or} \quad \Delta\tilde{\mathbf{z}}^i = \tilde{\mathbf{z}}^{i+1} - \tilde{\mathbf{z}}^i \quad \text{with} \quad i = 0, 1, 2, ..., k \,, \tag{4.78}$$

where henceforth the second possibility is utilized. Due to minimizing Eq. (4.72), the definition $\Delta\mathbf{R} \approx -\mathbf{R}^k$ holds. Thus, the new increment can be written as follows:

$$\Delta\mathbf{z} = \mathbf{R}^k + \sum_{i=0}^{k-1} \alpha_i^k \Delta\tilde{\mathbf{z}}^i \tag{4.79}$$

Taking the definition of one Newton-Raphson iteration into account as given by Eq. (4.70), the approximated Jacobian finally results in

$$\Delta\mathbf{z}^k = -\left(\left.\frac{\partial\mathbf{R}}{\partial\mathbf{z}}\right|_{\mathbf{z}=\mathbf{z}^k}\right)^{-1} \mathbf{R}(\mathbf{z}^k) \approx \mathbf{R}^k + \sum_{i=0}^{k-1} \alpha_i^k \Delta\tilde{\mathbf{z}}^i \,. \tag{4.80}$$

Since this method creates an approximated Jacobian inverse from the change in the residuals, at least two previous iterations are needed before the quasi-Newton method can be applied for the first time. In computations where the initial values are predicted by vector extrapolation as described in Section 4.2.1, this procedure can be applied one iteration earlier. By taking the results from older time-steps into consideration, the Jacobian can be updated constantly. The matrices introduced in the following serve to store the vectors of the residual and the solution increment from all iterations of the current time-step as follows:

$$\mathbf{Y}^k = \begin{bmatrix} \Delta\mathbf{R}^{k-1} ... \ \Delta\mathbf{R}^1 \ \Delta\mathbf{R}^0 \end{bmatrix} \quad \text{and} \quad \mathbf{Z}^k = \begin{bmatrix} \Delta\tilde{\mathbf{z}}^{k-1} ... \ \Delta\tilde{\mathbf{z}}^1 \ \Delta\tilde{\mathbf{z}}^0 \end{bmatrix} \,. \tag{4.81}$$

Thus, the dimensions of the matrices are $\mathbf{Y}^k \in \mathbb{R}^{d\times(k-1)}$ and $\mathbf{Z}^k \in \mathbb{R}^{d\times(k-1)}$, with d being the number of data stored in $\mathbf{z}$, and k the current number of iterations needed. In addition, two further matrices $\mathbf{Y}_t^k$ and $\mathbf{Z}_t^k$ are initialized, which include $\mathbf{Y}^k$ or $\mathbf{Z}^k$ respectively. Due to the data storage of m previous time-steps, the solution vectors can be stored in one matrix as

$$\mathbf{Y}_t^k = \begin{bmatrix} \mathbf{Y}_{n-m}^k ... \ \mathbf{Y}_{n-1}^k \ \mathbf{Y}_n^k \end{bmatrix} \quad \text{and} \quad \mathbf{Z}_t^k = \begin{bmatrix} \mathbf{Z}_{n-m}^k ... \ \mathbf{Z}_{n-1}^k \ \mathbf{Z}_n^k \end{bmatrix} \,. \tag{4.82}$$

These two matrices now include the information from m time-steps, each one containing data from k coupling iterations that were needed to converge successfully. Thanks to the data storage concept, it is possible to rearrange the minimization problem (4.74) in the following compact fashion as

$$\min_{\boldsymbol{\alpha}^k} \left\Vert \mathbf{Y}_t^k \boldsymbol{\alpha}^k + \mathbf{R}^k \right\Vert_2^2 \quad \rightarrow \quad \boldsymbol{\alpha}^k = -\left[(\mathbf{Y}_t^k)^T \mathbf{Y}_t^k \right]^{-1} (\mathbf{Y}_t^k)^T \mathbf{R}^k \tag{4.83}$$

From this expression, one can deduce for the new increment that

$$\Delta \mathbf{z}^k = \mathbf{R}^k + \mathbf{Z}_t^k \boldsymbol{\alpha}^k \,. \tag{4.84}$$

Using Eq. (4.70) and Eq. (4.83), the inverse of the Jacobian can be approximated by

$$\left[\frac{\partial \mathbf{R}}{\partial \mathbf{z}} \bigg|_{\mathbf{z} \approx \mathbf{z}^k} \right]^{-1} = \mathbf{Z}_t^k \left[(\mathbf{Y}_t^k)^T \mathbf{Y}_t^k \right]^{-1} (\mathbf{Y}_t^k)^T - \mathbf{I} \,. \tag{4.85}$$

Storing the data in two large matrices allows to specify the number of reused data from r previous iterations. This is not limited to the current time increment. It further allows to store and reuse information across several older time-steps. However, this raises the question which might be the best possible quantity of stored vectors. Using all of the data of the computation requires a huge amount of data storage and will not lead to the fastest convergence. In [51], several values of r have been investigated, showing that a good choice lies between a range of $r \in [5, 15]$. Moreover, the value depends on the problem, as remarked in [36], and the optimal value is difficult to find. On the other hand, the algorithm is quite robust to this parameter and its optimal range is widely spread. Indicating that a certain number of data is reused, the method is henceforth abbreviated by QN(r), where r denotes the number of stored previous iterations. Finally, a possible computer implementation of the quasi-Newton algorithm is given in Algorithm A-5.

4.2.5.2 Broyden method

Another method that is often referred to as a quasi-Newton procedure is the so-called *Broyden method* (BR), developed by BROYDEN [21] in the 1960s. It can be shown [22] that the Broyden algorithm is locally convergent and that the convergence can be superlinear. Early applications in fluid mechanics can be found in [49], for instance. Specific aspects of the computer implementation to solve nonlinear systems of equation are addressed in [101, 102], and improvements to reduce the memory storage to a minimum are made in [24]. Several extensions and enhancements of the method finally lead to the so-called *Broyden Class*[5], where a class of effective quasi-Newton techniques are defined.

With respect to a partitioned solution strategy, the objective is that the discrete residual

$$\mathbf{R}(\mathbf{z}) = \mathbf{0} \,, \quad \mathbf{R} : \mathbb{R}^d \to \mathbb{R}^d \,. \tag{4.86}$$

vanishes in a few iterations, which can be accelerated by applying a *Broyden step*

$$\mathbf{B}^k \Delta \mathbf{z}^k = -\mathbf{R}^k \quad \to \quad \Delta \mathbf{z}^k = - \left(\mathbf{B}^k \right)^{-1} \mathbf{R}^k \,. \tag{4.87}$$

---

[5]The class of Broyden methods includes several other algorithms, the *BFGS*-algorithm for instance, which are often used in numerical optimization. A comprehensive introduction can be found in [128]

Algorithm A-5 Algorithmic formulation of the Gauss-Seidel method accelerated by the reduced-order quasi-Newton method.

Initialize: $k = 0$; Compute initial guess: $\mathbf{z}^0 = 3\,\mathbf{z}_n - 3\,\mathbf{z}_{n-1} + \mathbf{z}_{n-2}$; Choose $\omega \in$ (0.01, 2)

1: **while** $||\mathbf{r}^k|| > \epsilon||\mathbf{r}^0||$ **do**
2:     $\mathbf{y}^{k+1} = \mathcal{F}(\mathbf{z}^k)$
3:     $\tilde{\mathbf{z}}^{k+1} = \mathcal{G}\left(\mathbf{y}^{k+1}\right)$
4:     $\mathbf{r}^k = \tilde{\mathbf{z}}^{k+1} - \mathbf{z}^k$
5:     **if** $k = 0$ **then**
6:         $\mathbf{z}^{k+1} = \mathbf{z}^k + \omega\left(\tilde{\mathbf{z}}^{k+1} - \mathbf{z}^k\right)$
7:     **else**
8:         $\mathbf{R}^k = \mathbf{r}^k$
9:         **for** $i = 0$ to $i < (k-1)$ **do**
10:             $\Delta\mathbf{R}^i = \mathbf{R}^{i+1} - \mathbf{R}^i$
11:             $\Delta\tilde{\mathbf{z}}^i = \tilde{\mathbf{z}}^{i+1} - \tilde{\mathbf{z}}^i$
12:         **end for**
13:         Add $\mathbf{Y}^k = \begin{bmatrix} \Delta\mathbf{R}^{k-1} \dots & \Delta\mathbf{R}^1 & \Delta\mathbf{R}^0 \end{bmatrix}$ to $\mathbf{Y}_t^k$
14:         Add $\mathbf{Z}^k = \begin{bmatrix} \Delta\tilde{\mathbf{z}}^{k-1} \dots & \Delta\tilde{\mathbf{z}}^1 & \Delta\tilde{\mathbf{z}}^0 \end{bmatrix}$ to $\mathbf{Z}_t^k$
15:         Solve minimization problem for $\boldsymbol{\alpha}^k$: $\min \left|\left|\mathbf{Y}_t^k\boldsymbol{\alpha}^k + \mathbf{R}^k\right|\right|_2^2$
16:         $\Delta\mathbf{z}^k = \mathbf{R}^k + \mathbf{Z}_t^k\boldsymbol{\alpha^k}$
17:         $\mathbf{z}^{k+1} = \mathbf{z}^k + \Delta\mathbf{z}^k$
18:     **end if**
19:     $k = k + 1$
20: **end while**
21: Go to next time-step: set $t_{n+1} = t_n + \Delta t$ and $\mathbf{z}_n = \mathbf{z}^{k+1}$

In the literature, the Broyden procedure is often referred to as a secant method, see [101, 102]. In order to avoid the computation of the Jacobian, it is possible to introduce an according approximation $\mathbf{B} \in \mathbb{R}^{d \times d}$, which is known as *Broyden's matrix*. A new increment is computed in every coupling iteration, and the current solution is corrected by

$$\mathbf{z}^{k+1} = \mathbf{z}^k + \Delta\mathbf{z}^k\,. \tag{4.88}$$

In order to obtain the increment $\Delta\mathbf{z}^k$, the inverse of Broyden's matrix $\mathbf{B}^k$ is required – and in view of the next iteration, this matrix needs to be updated. At the beginning, an initial guess for $\mathbf{B}^0$ must be given. According to [101], a typical choice is $\mathbf{B}^0 = \mathbf{I}$, which leads to $\Delta\mathbf{z}^0 = \mathbf{R}^0$. After the computation of the increment, the update of $\mathbf{B}$ for the next iteration is carried out by means of the *Broyden formula* [21]

$$\mathbf{B}^{k+1} = \mathbf{B}^k + \left(\mathbf{y}_k - \mathbf{B}_k\Delta\mathbf{z}^k\right)\frac{(\Delta\mathbf{z}^k)^T}{(\Delta\mathbf{z}^k)^T\Delta\mathbf{z}^k} = \mathbf{B}^k + \mathbf{u}^k(\mathbf{v}^k)^T\,. \tag{4.89}$$

In this formula, the definitions

$$\mathbf{y}^k := \mathbf{R}^k - \mathbf{R}^{k-1}, \quad \mathbf{u}^k := \frac{\mathbf{y}_k - \mathbf{B}^k \Delta \mathbf{z}^k}{||\Delta \mathbf{z}^k||}, \quad \mathbf{v}^k := \frac{\Delta \mathbf{z}^k}{||\Delta \mathbf{z}^k||} \qquad (4.90)$$

are used. All of these vectors are based on known values and can easily be calculated. The inverse $\mathbf{H}^{k+1}$ of Eq. (4.89) is computed using the *Sherman-Morrison formula*[6] [6]

$$\mathbf{H}^{k+1} = \left[\mathbf{B}^k + \mathbf{u}^k (\mathbf{v}^k)^T \right]^{-1} = \left(\mathbf{I} - \frac{\left[\mathbf{H}^k \mathbf{u}^k \right] (\mathbf{v}^k)^T}{1 + (\mathbf{v}^k)^T \mathbf{H}^k \mathbf{u}^k} \right) \mathbf{H}^k \qquad (4.91)$$

and following [101, 102], the inverse $(\mathbf{B}^k)^{-1} = \mathbf{H}^k$ can be written as

$$\mathbf{H}^k = \prod_{j=0}^{k-1} \mathbf{I} + \frac{\Delta \mathbf{z}^{j+1} (\Delta \mathbf{z}^j)^T}{||\Delta \mathbf{z}^j||_2^2}. \qquad (4.92)$$

With computing $\mathbf{H}^{k+1}$, the increment in the next iteration can be evaluated by a matrix-vector multiplication. For large systems, unfortunately, an update of the Broyden matrix or its inverse will be cost-intensive due to several matrix-vector operations. Moreover, a lot of data has to be stored, i.e. a $d \times d$ matrix has to be memorized over the whole simulation. In order to avoid these drawbacks and to make it applicable as an acceleration method, a restart version of Broyden's algorithm is employed that needs to update only a vector instead of a matrix. The algorithm used here is given in [101, 102] and has been proven capable of solving coupled FSI problems in [119]. Another positive feature is that the storage in this version is reduced drastically. However, to accelerate the iteration process in a multi-physical simulation, further mathematical steps are required – for which the details are to be found in the literature [101, 102].

The final computer implementation is outlined in Algorithm A-6. The algorithm is implemented by using a constant line search or relaxation parameter ω, and it further allows to specify the number of iterations used to update the Broyden matrix.

4.2.5.3 Krylov methods

Krylov subspace methods are a class of mathematical methods that can be applied to solve linear and nonlinear systems of equations. For example, let $\mathbf{Ax} = \mathbf{b}$ be a system of equations that can only be solved drawing on explicit information of the system matrix $\mathbf{A}$. For a solution in the so-called *Krylov subspace*, however, only the matrix-vector product needs to be known. This fact can be viewed as the

---

[6] The Sherman-Morrison formula states that

$$\left(\mathbf{B} + \mathbf{u}\mathbf{v}^T \right)^{-1} = \left(\mathbf{I} - \frac{\mathbf{B}^{-1}\mathbf{u}\mathbf{v}^T}{1 + \mathbf{v}^T\mathbf{B}^{-1}\mathbf{u}} \right) \mathbf{B}^{-1}$$

holds, if $\mathbf{B}$ is invertible and if and only if $1 + \mathbf{v}^T\mathbf{B}^{-1}\mathbf{u} \neq 0$.

Algorithm A-6 Algorithmic formulation of the Gauss-Seidel method accelerated by the Broyden Method.

Initialize: $k = 0$; Compute initial guess: $\mathbf{z}^0 = 3\,\mathbf{z}_n - 3\,\mathbf{z}_{n-1} + \mathbf{z}_{n-2}$; Choose $\omega \in$ $(0.01, 2)$

1: **while** $||\mathbf{r}^k|| > \epsilon ||\mathbf{r}^0||$ **do**
2:    $\mathbf{y}^{k+1} = \mathcal{F}(\mathbf{z}^k)$
3:    $\tilde{\mathbf{z}}^{k+1} = \mathcal{G}\left(\mathbf{y}^{k+1}\right)$
4:    $\mathbf{r}^k = \tilde{\mathbf{z}}^{k+1} - \mathbf{z}^k$
5:    **if** $k = 0$ **then**
6:       $\mathbf{s}^{k+1} = \mathbf{r}^k$
7:       $\mathbf{z}^{k+1} = \mathbf{z}^k + \omega\,\mathbf{s}^{k+1}$
8:    **else**
9:       $\mathbf{b} = \mathbf{r}^k$
10:       **for** $i = 0$ to $i < (k - 1)$ **do**
11:         $\mathbf{b} \leftarrow \mathbf{b} + \dfrac{\mathbf{b}^T \mathbf{s}^i}{||\mathbf{s}^i||_2^2}\,\mathbf{s}^{i+1}$
12:       **end for**
13:       $a^k = \dfrac{\omega}{||\mathbf{s}^k||_2^2}\,\mathbf{b}^T \mathbf{s}^k$
14:       $\mathbf{s}^{k+1} = \dfrac{1}{1 - a^k}\left(\mathbf{b} - (1 - \omega)\dfrac{a^k}{\omega}\,\mathbf{s}^k\right)$
15:       $\mathbf{z}^{k+1} = \mathbf{z}^k + \omega\,\mathbf{s}^{k+1}$
16:    **end if**
17:    $k = k + 1$
18: **end while**
19: Go to next time-step: set $t_{n+1} = t_n + \Delta t$ and $\mathbf{z}_n = \mathbf{z}^{k+1}$

---

key to use them in conjunction with Newton's method since exact information about the Jacobian is not needed. That is also why approaches in this manner are often denoted as *Jacobian-free Newton-Krylov methods* [103]. In the following, a *hybrid Krylov method* [20] is introduced that solves the system in the Krylov subspace using the *generalized minimal residual method*, which is typically known as *GMRES-method*.

To begin with, the discrete nonlinear system (4.69) that is to be solved during the coupling iteration is rewritten in a more compact fashion

$$\mathbf{J}(\mathbf{z}^k)\Delta\mathbf{z}^k = -\mathbf{R}(\mathbf{z}^k) \quad \text{with} \quad \mathbf{z}^{k+1} = \mathbf{z}^k + \Delta\mathbf{z}^k. \tag{4.93}$$

The objective is to compute $\Delta\mathbf{z}^k$ for every k-th iteration within a Krylov subspace using j Krylov iterations. Consequently, every Krylov iteration can be viewed as a Newton correction and is therefore denoted as a sub-iteration. The j-th Krylov subspace $\mathcal{K}_j$ reads

$$\mathcal{K}_j = \text{span}\left\{\mathbf{r}^0, \mathbf{J}\,\mathbf{r}^0, \mathbf{J}^2\,\mathbf{r}^0 \dots, \mathbf{J}^{j-1}\,\mathbf{r}^0\right\} \quad \text{with} \quad \mathbf{r}^0 = -\mathbf{R} - \mathbf{J}\Delta\mathbf{z}_0 \tag{4.94}$$

in which $\Delta\mathbf{z}_0 = \mathbf{0}$ is a reasonable [103] choice. Thus, the initial residual is $\mathbf{r}^0 =$

$-\mathbf{R}$. The increment in the j-th GMRES iteration is written as

$$\Delta\mathbf{z}_j = \Delta\mathbf{z}_0 + \sum_{i=0}^{j-1} \alpha_i\,\mathbf{J}^i\mathbf{r}^0 = \Delta\mathbf{z}_0 + \boldsymbol{\alpha}_j^T\mathbf{V}_j\,, \tag{4.95}$$

where $\mathbf{V} = [\mathbf{r}^0\ \mathbf{J}\,\mathbf{r}^0\ ...\ \mathbf{J}^{j-1}\,\mathbf{r}^0]^T$ is a vector containing the matrix-vector products and the vector $\boldsymbol{\alpha} = [\alpha_0\ \alpha_1\ ...\ \alpha_{j-1}]^T$ includes the parameters that minimize

$$\min_{\boldsymbol{\alpha}_j} ||\mathbf{J}\Delta\mathbf{z}_j + \mathbf{R}||\,. \tag{4.96}$$

However, an expression for the matrix-vector product $\mathbf{Jv}$ is required and, following [20], it can be approximated by means of a simple function evaluation

$$\mathbf{Jv} \approx \frac{\mathbf{R}(\mathbf{z}+\epsilon\mathbf{v}) - \mathbf{R}(\mathbf{z})}{\epsilon}\,, \tag{4.97}$$

where the parameter ϵ is a small perturbation value. There exist further methods for the evaluation of the matrix-vector product, see [103] for instance. After the GMRES sub-iteration procedure, the new updated solution follows to

$$\mathbf{z}^{k+1} = \mathbf{z}^k + \Delta\mathbf{z}_j = \mathbf{z}^k + \Delta\mathbf{z}_0 + \boldsymbol{\alpha}_j^T\mathbf{V}_j\,. \tag{4.98}$$

An algorithmic sketch of this method can be found in Algorithm A-7. Despite the fact that an exact computation of the Jacobian is not required, an approximation of its matrix-vector product is needed in every sub-iteration, see line 7 in Algorithm A-7. It is from these products that the corresponding Krylov-subspace is created. This can be done by means of Eq. (4.97), which is, however, a very general approximation and might not be the most appropriate way to address coupled problems. An eligible alternative would be to span the Krylov subspace simply based on the differences $\Delta\mathbf{z}^i = \mathbf{z}^i - \mathbf{z}^0$ or $\Delta\mathbf{R}^i = \mathbf{R}^i - \mathbf{R}^0$. This procedure was employed to solve FSI problems in [116, 117, 37], where it was named *interface-GMRES method*. It can further be improved by reusing information from previous coupling iterations, as carried out in [118]. Nevertheless, in order to compute the vector-matrix products, the solvers still need to be called during the sub-iteration.

4.2.6   Convergence and performance study

All methods discussed in the previous sections are studied on a simple example. The aim is to carry out a pre-selection and to figure out which algorithms can be used to accelerate the Gauss-Seidel iteration process. To this end, a nonlinear system of equations $\mathbf{G}: \mathbb{R}^4 \mapsto \mathbb{R}^4$ given by

$$\mathbf{G}(\mathbf{z}) = \begin{bmatrix} z_1^3 + z_1z_2 + 0.5z_3^2 - 5z_4 - 1 \\ -3z_1 + z_2 + z_2^2 - z_3 + 2z_4 \\ 4z_1 - z_2 - 5z_3 + z_3^2z_4 + z_4 \\ -z_1z_2 + z_2 - 2z_3 + z_3^2z_4 + z_4^4 \end{bmatrix},\ \mathbf{z}_0 = \begin{bmatrix} 2 \\ 2 \\ 2 \\ 2 \end{bmatrix},\ \mathbf{z}_s = \begin{bmatrix} 1.4245470539476539 \\ 1.49686949063678742 \\ 1.559638086955233331 \\ 1.047879574308079314 \end{bmatrix}$$

Algorithm A-7 Algorithmic formulation of the Gauss-Seidel method accelerated by Newton-Krylov iteration.

Initialize: $k = 0$; Compute initial guess: $\mathbf{z}^0 = 3\,\mathbf{z}_n - 3\,\mathbf{z}_{n-1} + \mathbf{z}_{n-2}$

1: **while** $||\mathbf{r}^k|| > \epsilon||\mathbf{r}^0||$ **do**
2:     $\mathbf{y}^{k+1} = \mathcal{F}(\mathbf{z}^k)$
3:     $\tilde{\mathbf{z}}^{k+1} = \mathcal{G}\left(\mathbf{y}^{k+1}\right)$
4:     $\mathbf{r}^k = \tilde{\mathbf{z}}^{k+1} - \mathbf{z}^k$
5:     **Initialize values for GMRES sub-iteration:** $\mathbf{r}_0 = -\mathbf{r}^k$; $\mathbf{v}_0 = \mathbf{r}_0$; $j = 1$
6:     **while** $||\mathbf{r}_j|| > \epsilon_1$ **do**
7:         Build $\mathbf{v}_j = \mathbf{J}\mathbf{v}_{j-1}$ using Eq. (4.97)
8:         $\mathbf{V} = [\mathbf{v}_0, \mathbf{v}_1, ..., \mathbf{v}_j]$
9:         $\boldsymbol{\alpha}_j = \arg\min_{\boldsymbol{\alpha}} ||\sum_{i=0}^{j-1} \alpha_i \mathbf{v}_{i+1} + \mathbf{r}^k||$
10:        $\mathbf{r}_j = \sum_{i=0}^{j-1} \alpha_i \mathbf{v}_{i+1} + \mathbf{r}^k$
11:        $\Delta\mathbf{z}_j = \sum_{i=0}^{j-1} \alpha_i \mathbf{v}_i$
12:        $j \leftarrow j + 1$
13:     **end while**
14:     $\mathbf{z}^{k+1} = \mathbf{z}^k + \Delta\mathbf{z}_j$
15:     $k = k + 1$
16: **end while**
17: Go to next time-step: set $t_{n+1} = t_n + \Delta t$ and $\mathbf{z}_n = \mathbf{z}^{k+1}$

$$(4.99)$$

is solved in a partitioned manner. Here, $\mathbf{z}_s$ denotes the solution vector, and $\mathbf{z}_0$ is the initial vector. The Gauss-Seidel iteration process between the blocks is terminated after the k-th iteration if $||\mathbf{G}(\mathbf{z}^k)|| < 10^{-15}$ was achieved. For a partitioned formulation, the system is split into two blocks, resulting in

$$\mathbf{G}(\mathbf{z}) = \begin{bmatrix} \mathbf{G}_1(\mathbf{z}_1, \mathbf{z}_2) \\ \mathbf{G}_2(\mathbf{z}_1, \mathbf{z}_2) \end{bmatrix} \quad \text{with} \quad \mathbf{z}_1 = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \quad \text{and} \quad \mathbf{z}_2 = \begin{bmatrix} z_3 \\ z_4 \end{bmatrix}. \tag{4.100}$$

The general coupling iteration procedure between the subsystems is illustrated in Figure 4.3.

The results are given in Table 4.2, comparing the number of iterations required, denoted by ς, and the number of solver calls, denoted by ϖ, for all acceleration methods. The computational effort is typically defined by the solver calls. For the reference case without an acceleration scheme, $\varsigma = 55$ iterations and $\varpi = 110$ solver calls are needed to achieve an accuracy of $\epsilon < 10^{-15}$. For the sake of clarity, the convergence acceleration methods are classified in three groups: the first group consists of Aitken-type methods which are highlighted in blue in Table 4.2, the second group are the Newton-type schemes, marked green, and the third group involves the vector sequence acceleration schemes (indicated by red).

- **Aitken-type methods** show good convergence properties. Even static under relaxation (SUR) reduces the number of iterations from 55 to 23. Dy-
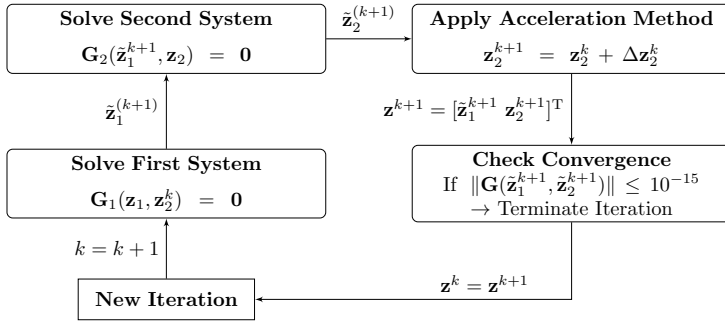
Figure 4.3: Partitioned formulation of $\mathbf{G}(\mathbf{z}) = \mathbf{0}$ with application of acceleration methods.

Table 4.2: Convergence study of different convergence acceleration methods: comparison of Gauss-Seidel coupling iterations (ς) and number of solver calls (ϖ).

Method	ς	ϖ	Parameter	‖	Method	ς	ϖ	Parameter
Ref	55	110			QN	18	36	$\omega = 1.0$
SUR	23	46	$\omega = 0.8$		QN(5)	19	38	
DAR1	24	48			QN(10)	18	36	
DAR2	22	44			BR	13	26	$\omega = 1.0$, $n_r = 10$
DAR3	24	48			JFNK	6	80	
DAR4	23	46			VEA	23	46	
DAR5	23	46			TEA	27	54	
DAR6	23	46			VTA	25	50	
DSR	26	52	$\omega_0 = 1.0$		GTA	32	64	
GDSR	25	50	$m = 3$		EWT	27	54	
LE	12	24	$\omega = 1$, $\zeta = 0.6$		VWT	27	54	

namic Aitken relaxation (DAR), see Algorithm A-3, performs similarly: Depending on which scheme is used, between 22 and 27 iterations are needed. Bearing in mind that the schemes are applied every third iteration, a recursive application using the DSR (Algorithm A-4) and GDSR methods surprisingly needs 26 iterations and does not lead to a further convergence acceleration for this example. It turns out that the line extrapolation method (LE) – with a chosen line search parameter of $\zeta = 0.6$, see also Eq. (4.60) – is the fastest way to solve the system as it needs only 12 iterations.

- **Newton-type methods** deliver promising results as well. The QN method (Algorithm A-5) reduces the iteration counter from 55 to 18, and the Broyden scheme (Algorithm A-6) performs even better with only 13 iterations. The JFNK method, see Algorithm A-7, allows to solve the system of equa-

tions within 6 iterations. However, due to the Jacobian evaluation during the GMRES iteration, 80 solver calls are required. Compared to the other Newton type schemes, the JFNK method cannot reach the same level of performance for this example.

- **Vector sequence acceleration methods** from Section 4.2.4 can also be employed to improve the convergence. On the other hand, they perform slightly worse than Newton or Aitken type methods. Here, the VEA (Eq. (4.62)) delivers the fastest results and reduces the number of required iterations from 55 to 26.

It is to be noticed that the performance of the acceleration methods depend on the problem under investigation, on the starting values (z_s) and also on the parameters that have to be defined by the user, for example the line search parameter ζ of the LE method. That is why a general statement about the "best" method is not possible. A starting vector, however, that is close to the solution, can obviously be an appropriate way to keep the number of coupling iterations on a low level. On the other hand, the "optimal" user-defined parameters for the methods have to be figured out for every problem under consideration.

4.3 Consistent field transfer

In a partitioned coupling strategy, data have to be transferred between all solvers involved. It is mandatory to assure that energy is approximately conserved during transmission such that the field transfer is called to be consistent. In special cases where the data exchange takes place on the same mesh, a consistent field transfer is automatically guaranteed. Otherwise, an additional mapping concept to transfer the data has to be established. Several concepts are briefly reviewed in this section. In this context, aspects of mesh adaptation to the current deformation are discussed as well. In many coupled simulations, this needs to be done in every coupling iteration and is therefore referred to as *dynamic mesh update*.

4.3.1 Interpolation concepts

For the case of non-matching meshes, additional interpolation concepts need to be initiated. An example is depicted in Figure 4.4, where two domains are coupled over a common interface with non-matching interface discretization. In this example, the transfer coordinates are different, which is why all physical quantities at these points are projected on the coordinates of the other mesh. Typical situations in which data interpolation concepts are required can be summarized as follows:

- surface-coupled problems with different mesh interfaces such as a FEM-FVM coupling

- volume-coupled problems which are solved on different meshes

- other types of different discretization scheme combinations

- coupling of specialized black-box solvers without having access to the code
  structure
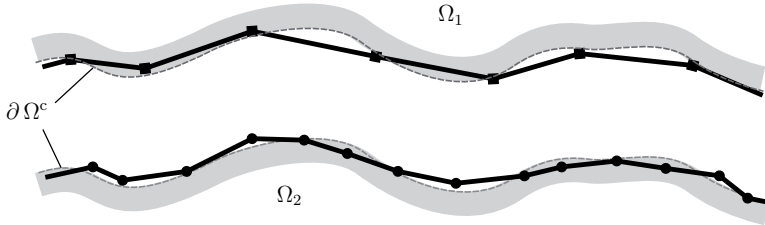
Figure 4.4: Two domains – Ω_1 and Ω_2 sharing one boundary $\partial\Omega^c$ – are discretized with non-matching interface meshes.

There exist several possibilities to map data between non-matching coupling domains. Many articles have been published that address energy-preserving loads and mass transfer for FSI problems. An introduction to several methods for surface-coupled problems can be found in [56, 57, 33, 34], for instance. Another well-known procedure uses so-called *Radial Basis Functions* (RBF), see e.g. [35, 28], to transfer data between a non-matching interface discretization.

Apart from the application in FSI, an accurate field transfer is also necessary in the course of re-meshing or mesh adaptation within the analysis of inelastic material models that use internal variables or history-dependent quantities. For instance, [41] applied information transfer between incompatible meshes using mortar techniques. The aspect of information transfer at the Gaussian points between non-matching finite element meshes is discussed in [14, 67]. In both articles, an efficient, accurate and adaptive re-meshing concept is obtained by fitting a polynomial in a least-square fit sense. This idea can be taken to transfer data between different meshes of volume-coupled problems where history-dependent data are only locally available. This was studied in [147] on the example of small strain thermo-viscoplasticity.

Anyway, depending on which data transfer concept is used, there might be an increase in the computational effort. A higher accuracy typically leads to a more time-consuming procedure. For certain methods, the extra computation time might be negligible, in particular if they are used for coupling domains with only a few transfer points. For example, surface-coupled problems typically consist of less data points than volume-coupled problems. However, if the domain embraces a huge amount of data points, the situation might change – and the additional costs may downgrade the efficiency of the whole simulation. For some special cases, it is thus necessary to devote special care to the applied interpolation concept by carefully weighing up accuracy and computational efficiency.

In order to describe the data transfer between non-matching discretizations, some general definitions are introduced. Let Ω^c or $\partial\Omega^c$ be a region that describes the coupling domain of two fields, which can either be a volume or a surface. For both fields, the global vectors $\boldsymbol{\chi} = [\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_N]^T$ and $\boldsymbol{\chi}^* = [\mathbf{x}_0^*, \mathbf{x}_1^*, ..., \mathbf{x}_{N^*}^*]^T$ are introduced, including the discrete coordinate vectors of all transfer points in which $\mathbf{x}_i \in \Omega^c$, $i = 0, 1, .., N$ and $\mathbf{x}_j^* \in \Omega^c$, $j = 0, 1, .., N^*$. These points can be nodes, integration points or cell center points. Further, let $\mathbf{d} = [\mathbf{d}_0, \mathbf{d}_1, ..., \mathbf{d}_N]^T$ be a discrete data vector corresponding to $\boldsymbol{\chi}$ containing the physical quantities that need to be exchanged. The data vector $\mathbf{d}^* = [\mathbf{d}_0^*, \mathbf{d}_1^*, ..., \mathbf{d}_{N^*}^*]^T$ corresponds to $\boldsymbol{\chi}^*$ where the interpolated quantities are stored. This vector needs to be computed. Moreover, the vector $\mathbf{n}_j^*$ is introduced. It contains the n_n nearest neighbors of $\mathbf{x}_j^*$ out of the global coordinate vector $\boldsymbol{\chi}$. For example, consider Figure 4.5, where data from the four nearest neighbors $\mathbf{x}_{j,1}$, $\mathbf{x}_{j,2}$, $\mathbf{x}_{j,3}$, $\mathbf{x}_{j,4}$ spanning the vector $\mathbf{n}_j^* = [\mathbf{x}_{j,1}, \mathbf{x}_{j,2}, \mathbf{x}_{j,3}, \mathbf{x}_{j,4}]^T$ are mapped on the j-th point $\mathbf{x}_j^*$ to obtain $\mathbf{d}_j^*$.
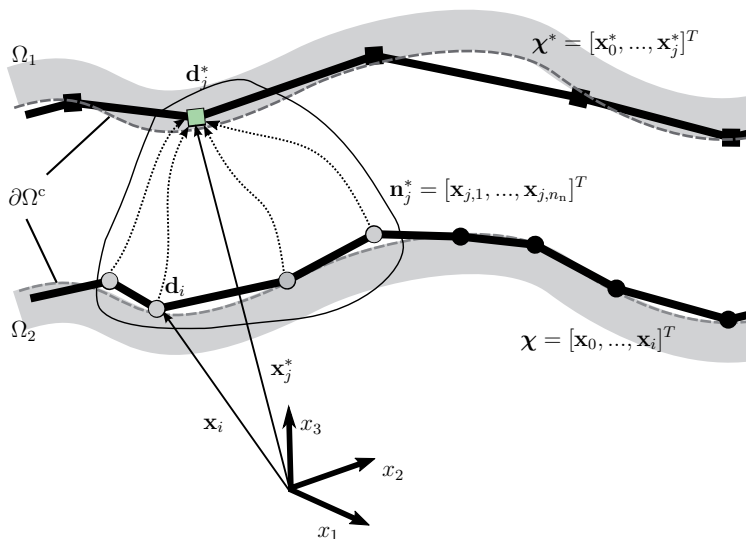
Figure 4.5: Data transfer between two non-matching coupling domains: mapping, coordinates and neighbor points definition.

In the following, the nearest neighbor search shall be brought into focus. For huge data sets, it is recommended to employ special search algorithms for an efficient nearest neighbor calculation. In [123, 124], a *fast library for approximate nearest neighbors* (FLANN) is proposed. This library offers the possibility to compute a

certain number of nearest neighbors by means of *k-d-*trees. On the other hand, it additionally raises the option to define all points which lie within a specified search radius r_s, see Figure 4.6 for a two-dimensional example. Compared to the *brute-force method*, where all points are checked individually against each other, this method is considerably faster.
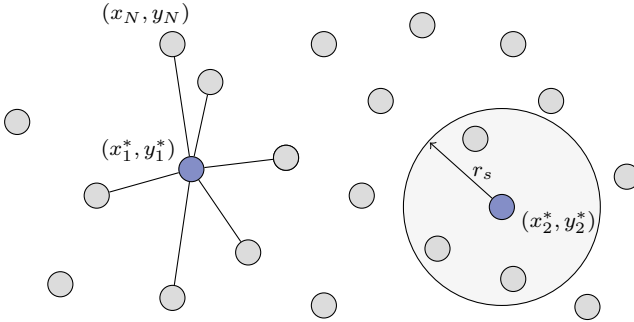
Figure 4.6: 2D example of nearest neighbor search: classical search for $n_n = 5$ neighbors (left) and radius search (right) with the search radius r_s.

A simple comparison serves to demonstrate the effectiveness of this library: Two coordinate sets are considered, where χ consists of 250,000 data points and χ^* of 100,000. The coordinates of all points are in the range of $[0, 1]$. The objective is to find the $n_n = 5$ nearest neighbors $\mathbf{x}_{j,i} \in \chi$, $i = 1, ..., n_n$ to each data point $\mathbf{x}_j^* \in \chi^*$. This is done with the brute-force method[7], which is exact but time-consuming, and with the FLANN search algorithm for a different number of *k-d-*trees. The results are listed in Table 4.3. It is demonstrated that the *k-d-*tree search algorithm reduces the CPU time drastically. However, if the dimension of the *k-d-*tree is small ($d = 2$), the method is slightly inaccurate. It turns out that there is a deviation of 1.83 % compared to the brute-force method. This means that a wrong neighborhood is determined for 1,827 out of 100,000 data points. FLANN delivers only an approximation of the nearest neighbor search and offers several possibilities to increase the accuracy. If a tree of $d = 8$ is chosen, the nearest neighbor search delivers the exact solution while, on the other hand, the computation time increases. Compared to the brute-force method, it is still considerably faster as it needs only 3.64% of the CPU time. In this thesis, a tree with the dimension $d = 8$ is set to be the standard case.

---

[7]This can also be carried out with the FLANN library as it also provides a brute-force search algorithm [123]

Table 4.3: Comparison of nearest neighbor search algorithms: k-d-tree search algorithm related to the brute-force method

Dimension	Rel. CPU time [%]	Deviation [%]
2	0.975	1.829
4	1.869	0.057
6	2.750	0.001
8	3.638	0.0

4.3.1.2 Nearest neighbor mapping

A straightforward way to transfer information from one mesh to another is to simply use the data from the nearest neighbor, which can be denoted as a *closest-point projection*. In that case, the neighborhood of every data point $\mathbf{x}_j^*$ consists of only one neighbor $\mathbf{x}_{j,1}$, which is why a closest-point projection at this point results in

$$\mathbf{d}_j^* \approx \mathbf{d}_{j,1} \,. \tag{4.101}$$

This means that the interpolated data $\mathbf{d}^*$ are taken from the nearest point without any modification or weighting. However, this procedure can be inaccurate as only information of one single point is used.

4.3.1.3 Barycentric interpolation

Another scheme under consideration is the so-called *barycentric interpolation* [11][8]. It is also denoted as *three-point triangle mapping* in two-dimensional cases of surface coupling or as *four-point tetrahedron mapping* in the three-dimensional case for volume coupling. For the sake of clarity, the following statements focus on the two-dimensional case which was used in [168] to transfer data between two different interface discretizations. The unknown vector $\mathbf{d}_j^*$ at the point of interest can be approximated by using the data of its three adjacent points $\mathbf{x}_{j,1}$, $\mathbf{x}_{j,2}$, $\mathbf{x}_{j,3}$ spanning a triangle around $\mathbf{x}_j^*$. Further, $S_i = \|\mathbf{x}_j^* - \mathbf{x}_{j,i}\|$ for $i = 1, 2, 3$ defines the distance between the point $\mathbf{x}_j^*$ and $\mathbf{x}_{j,i}$. If the distance S_i is less than a small user-defined length ϵ, information at this point is simply calculated by a closest-point projection

$$\mathbf{d}_j^* \leftarrow \mathbf{d}_{j,i}. \tag{4.102}$$

In the other case, the following three-point interpolation formula is employed to approximate the new value

$$\mathbf{d}_j^* = w_1 \, \mathbf{d}_{j,1} + w_2 \, \mathbf{d}_{j,2} + (1 - w_1 - w_2) \, \mathbf{d}_{j,3} \,. \tag{4.103}$$

---

[8]This procedure is also known as *barycentric Lagrange interpolation* [11]

Here, the *barycentric weights* w_1 and w_2 need to be calculated for $\mathbf{x}_j^* = (x_j^*, y_j^*)$[9] based on the following formula

$$w_1(\mathbf{x}_j^*) = \frac{(y_{j,2} - y_{j,3})(x_j^* - x_{j,3}) + (x_{j,3} - x_{j,2})(y_j^* - y_{j,3})}{(y_{j,2} - y_{j,3})(x_{j,1} - x_{j,3}) + (x_{j,3} - x_{j,2})(y_{j,1} - y_{j,3})} \,, \qquad (4.104)$$

$$w_2(\mathbf{x}_j^*) = \frac{(y_{j,3} - y_{j,1})(x_j^* - x_{j,3}) + (x_{j,1} - x_{j,3})(y_j^* - y_{j,3})}{(y_{j,3} - y_{j,1})(x_{j,1} - x_{j,3}) + (x_{j,1} - x_{j,3})(y_{j,1} - y_{j,3})} \,. \qquad (4.105)$$

A detailed discussion to derive the barycentric weights is provided in the Appendix A.3. The advantage of this procedure is that it is very easy to implement and that it is not necessary to solve an additional system of equations. On the other hand, three neighborhood points need to be determined, which can increase the total computation time for huge data sets. It is therefore recommended to use specialized search algorithms, as mentioned in the previous section. A possible computer implementation is given in Algorithm A-8.

Algorithm A-8 Algorithmic sketch of the barycentric mapping concept.

Initialize: $\mathbf{d} = [\mathbf{d}_0, \mathbf{d}_1, ..., \mathbf{d}_N]^T$; $\mathbf{x}^* = [\mathbf{x}_0^*, \mathbf{x}_1^*, ..., \mathbf{x}_{N^*}^*]^T$; $\mathbf{x} = [\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_N]^T$

1: $i = 0$; $j = 0$
2: **for** $i < N^*$ **do**
3:    Find set of three nearest neighbors $\{\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2\}$ to $\mathbf{x}_i^*$
4:    Find corresponding set of data $\{\mathbf{d}_0, \mathbf{d}_1, \mathbf{d}_2\}$
5:    Check if $\mathbf{x}_i^*$ is inside the triangle
6:    **for** $j < 3$ **do**
7:        $S_j = \|\mathbf{x}_i^* - \mathbf{x}_j\|$
8:        **if** $S_j < \epsilon$ **then**
9:            $\mathbf{d}_i^* \leftarrow \mathbf{d}_j$
10:           $i = i + 1$; $j = 0$
11:           Go to line 3
12:       **end if**
13:   **end for**
14:   Compute barycentric weights w_1, w_2 using Eqns.(4.104)-(4.105)
15:   $\mathbf{d}_i^* = w_1\,\mathbf{d}_0 + w_2\,\mathbf{d}_1 + (1 - w_1 - w_2)\,\mathbf{d}_2$
16:   $i = i + 1$; $j = 0$
17: **end for**

4.3.1.4   Least-square fit procedure

This section addresses an accurate least-square fit procedure for non-matching finite element meshes, similar to [14]. This concept was applied to coupled thermo-viscoplasticity in [52, 147]. Further reading on such procedures is also provided

---

[9]In this case, the third component z_j^* has been dropped as the barycentric triangle interpolation for non-matching interfaces can be seen as a plane problem. However, for volume coupling, the four-point tetrahedron mapping is employed in which all three components need to be considered.

in [156]. The procedure is set up for every transfer point and is based on a least-square fit procedure using polynomial interpolation functions. The basic idea is as follows: A projection is sought that minimizes the error between the known quantity $\mathbf{d}$ and the interpolated data $\mathbf{d}^*$ at all transfer points by

$$\|\mathbf{d}^* - \mathbf{d}\|_2^2 \quad \rightarrow \quad \min . \tag{4.106}$$

In the following, every component d_i^* of the unknown data vector $\mathbf{d}_j^*$ is interpolated with a polynomial by means of

$$d_i^* = \mathbf{p}^T \boldsymbol{\alpha} \quad \text{and} \quad \mathbf{d}_j^* = [\, d_1, \, d_2, \, ..., \, d_{n_d} \,]^T , \tag{4.107}$$

where the vector $\mathbf{p}$ denotes the polynomial basis and $\boldsymbol{\alpha} = \left[\alpha_0, \alpha_1, ..., \alpha_{n_p} \right]^T$ is a vector containing the coefficients of the polynomial basis. The number of coefficients increases with $n_p = (p+1)^3$ where p is the polynomial order of the method.[10] Every data point needs to be considered, and a certain number of nearest neighbors for this point must be found. The n_n nearest neighbors are computed by means of the FLANN library, see Section 4.3.1.2. Here, the information of $n_\mathrm{n} \geq n_p$ nearest neighbors is required to assure that the interpolation scheme is determined mathematically.

The known quantities $\mathbf{d}$ from all neighbors $\mathbf{n}_j^* = [\mathbf{x}_{j,1}, ..., \mathbf{x}_{j,n_\mathrm{n}}]$ are used to approximate the quantity of interest $\mathbf{d}_j^*$. The objective is to calculate the unknown coefficients $\boldsymbol{\alpha}$ in a least squares sense based on all neighbors, leading to

$$\min_{\boldsymbol{\alpha}} \quad \left\| \sum_{l=1}^{n_\mathrm{n}} \mathbf{p}^T(\mathbf{x}_{j,l})\, \boldsymbol{\alpha} - d_i(\mathbf{x}_{j,l}) \right\|_2^2 . \tag{4.108}$$

This minimization problem can easily be recast into a system of linear equations which reads

$$\mathbf{M}\boldsymbol{\alpha} = \mathbf{b} \quad \text{with} \quad \mathbf{M} = \sum_{l=1}^{n_\mathrm{n}} \mathbf{p}(\mathbf{x}_{j,l})\mathbf{p}^T(\mathbf{x}_{j,l}) \quad \text{and} \quad \mathbf{b} = \sum_{l=1}^{n_\mathrm{n}} \mathbf{p}(\mathbf{x}_{j,l})d_i(\mathbf{x}_{j,l}). \tag{4.109}$$

Solving this system may lead to numerical difficulties since the system matrix $\mathbf{M}$ can be close to singular and also badly conditioned, which is especially likely if almost all nearest neighbor points are located inside one plane. Figure 4.7 shows a more detailed sketch of the problem for the two-dimensional case, illustrating different situations that con occur during a nearest neighbor search around a given point (marked red).

In the first case, see Figure 4.7(a), a well-conditioned system matrix is obtained. The arising equation system can be solved easily with standard algorithms, Q−R decomposition and backward substitution for example. Figure 4.7(b) depicts an example that results in an ill-conditioned system matrix, demanding a more sophisticated solution method. A possible approach would be to compute the

---

[10]With increasing p, it is recommended to use orthogonal polynomials, Legendre polynomials for instance, to improve the condition number of the resulting equation system.

(a) Well distributed neighborhood leading to a well-conditioned system matrix

(b) Minimum number of nearest neighbors leading to an ill-conditioned system matrix

(c) Not enough nearest neighbors leading to an unsolvable equation system
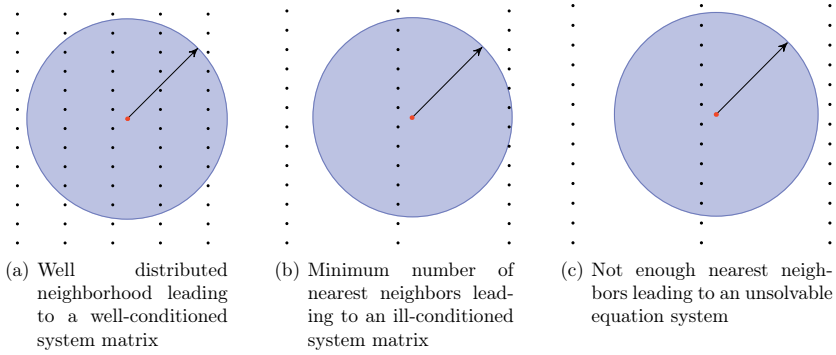
Figure 4.7: Nearest neighbors in 2D

pseudo-inverse of $\mathbf{M}$ by a *singular value decomposition*, see [147]. In contrast, the third situation, see Figure 4.7(c), cannot be solved since all neighbor points lie in one single line, leading to a singular system matrix. This situation can be avoided by increasing the search radius for the neighbor point calculation.

Finally, the interpolated components of the data vector d_i^* follow directly from an evaluation of the polynomial vector $\mathbf{p}$ at the requested point $\mathbf{x}_j^*$

$$d_i^* = \mathbf{p}^T(\mathbf{x}_j^*)\,\boldsymbol{\alpha}\,. \tag{4.110}$$

An efficient and simple way to improve the mapping procedure regarding its accuracy could be to weight the data such that the matrix $\mathbf{M}$ and the right-hand side vector $\mathbf{b}$ are multiplied with a weighting function w

$$\mathbf{M} = \sum_{l=1}^{n_{\mathrm{n}}} w_l(\mathbf{x}_j^*)\mathbf{p}(\mathbf{x}_{j,l})\mathbf{p}^T(\mathbf{x}_{j,l}) \quad \text{and} \quad \mathbf{b} = \sum_{l=1}^{n_{\mathrm{n}}} w_l(\mathbf{x}_j^*)\mathbf{p}(\mathbf{x}_{j,l})d_i(\mathbf{x}_{j,l}). \tag{4.111}$$

For every point $\mathbf{x}_j^*$, a weight function is sought to increase the influence of neighbors that are very close to this point and, further, to reduce the influence of neighbors that are further away. The following weight function – a similar one was used in [14] – can fulfil these requirements

$$w_l(\mathbf{x}_j^*) = \left(1 - \frac{\left\|\mathbf{x}_j^* - \mathbf{x}_{j,l}\right\|_2}{r_{\max}}\right)^{\alpha} \left(1 + \frac{\left\|\mathbf{x}_j^* - \mathbf{x}_{j,l}\right\|_2}{r_{\max}}\right)^{\beta}, \tag{4.112}$$

where $r_{\max}$ defines the maximum search radius, which is accordingly the maximum distance between the given point at $\mathbf{x}_j^*$ and its farthest neighborhood point. The weight-function can easily be adjusted by the exponents α and β. A computer implementation of the least-square fit procedure is depicted in Algorithm A-9.

Algorithm A-9 Algorithmic sketch of the least-square interpolation.

Initialize: $\mathbf{d} = [\mathbf{d}_0, \mathbf{d}_1, ..., \mathbf{d}_N]^T$; $\mathbf{x}^* = [\mathbf{x}_0^*, \mathbf{x}_1^*, ..., \mathbf{x}_{N^*}^*]^T$; $\mathbf{x} = [\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_N]^T$

1: Choose polynomial basis $\mathbf{p}$
2: $i = 0$
3: **for** $i < N^*$ **do**
4:    Find set of $n_\mathrm{n} > p + 1$ nearest neighbors $\{\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_{n_\mathrm{n}}\}$ to $\mathbf{x}_i^*$
5:    $j = 0$; $\mathbf{M} = \mathbf{0}$
6:    **for** $j < n_\mathrm{n}$ **do**
7:      $w_j = w(\mathbf{x}_j)$; $\mathbf{p}_j = \mathbf{p}(\mathbf{x}_j)$
8:      $\mathbf{M} = \mathbf{M} + w_j \mathbf{p}_j^T \mathbf{p}_j$
9:    **end for**
10:   Compute $\mathbf{M}^{-1}$
11:   **for** all components of $\mathbf{d}_i^*$ **do**
12:     $j = 0$; $\mathbf{b} = \mathbf{0}$
13:     **for** $j < n_\mathrm{n}$ **do**
14:       $w_j = w(\mathbf{x}_j)$; $\mathbf{p}_j = \mathbf{p}(\mathbf{x}_j)$
15:       $\mathbf{b} = \mathbf{b} + w_j \mathbf{p}_j d_j(\mathbf{x}_j)$
16:     **end for**
17:     $\boldsymbol{\alpha} = \mathbf{M}^{-1}\mathbf{b}$
18:     $d_i^* = \mathbf{p}^T(\mathbf{x}_i^*)\,\boldsymbol{\alpha}$
19:   **end for**
20:   $i = i + 1$
21: **end for**

4.3.1.5 Radial basis functions

Applying *radial basis functions* (RBF) is a common way to transfer data over discrete, non-matching mesh interfaces [9, 35, 28]. Again, let $\mathbf{d}$ and $\mathbf{d}^*$ be the discrete data vectors containing the information of all data that need to be exchanged. Assuming $\mathbf{d}$ as the known data vector to be transferred, $\mathbf{d}^*$ is unknown and can be approximated by means of the interpolation

$$\mathbf{d}^* = \mathbf{M}\,\mathbf{d}\,, \tag{4.113}$$

where $\mathbf{M}$ is a mapping or transformation matrix. The objective is to compute this matrix in an appropriate way using radial basis functions. Generally, an interpolation with such functions reads [9]

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i\,\phi(||\mathbf{x} - \mathbf{x}_i||) + P(\mathbf{x}) \tag{4.114}$$

where the α_i are the interpolation coefficients, ϕ is the basis function and $P(\mathbf{x})$ denotes a polynomial. The polynomial induces n_p additional coefficients into Eq. (4.114) that need to be determined. This polynomial can be written as

$$P(\mathbf{x}) = \sum_{r=1}^{n_p} P_r(\mathbf{x})\,a_r = \mathbf{p}^T(\mathbf{x})\,\mathbf{a}\,, \tag{4.115}$$

where the P_r are the polynomial terms and a_r the corresponding (unknown) polynomial coefficients. In order to obtain a fully defined system, one can enforce polynomial orthogonality such that

$$\sum_{i=1}^{n} P_r(\mathbf{x})\, \alpha_i = \mathbf{p}^T(\mathbf{x})\boldsymbol{\alpha} = 0 \tag{4.116}$$

holds. For the discrete case, all interpolation coefficients are computed based on the given data $\mathbf{d}$. This means, every single component of the data vector is interpolated as

$$d_j = \sum_{i=1}^{n_d} \alpha_i\, \phi(||\mathbf{x}_i - \mathbf{x}_j||) + P(\mathbf{x}_j)\,. \tag{4.117}$$

For the whole vector $\mathbf{d}$, this can be written in matrix notation as

$$\mathbf{d} = \boldsymbol{\Phi}\,\boldsymbol{\alpha} + \mathbf{P}\,\mathbf{a}\,, \qquad \mathbf{P}^T\boldsymbol{\alpha} = \mathbf{0} \quad \rightarrow \quad \begin{bmatrix} \mathbf{d} \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Phi} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ \mathbf{a} \end{bmatrix}\,. \tag{4.118}$$

In this system, the components of the matrix $\boldsymbol{\Phi}$ follow from evaluating the basis function $\phi_{ij} = \phi(||\mathbf{x}_i - \mathbf{x}_j||)$, the vector $\boldsymbol{\alpha}$ contains all interpolation coefficients, $\mathbf{P}$ is the polynomial matrix where the i-th row consists of the polynomial vector $\mathbf{p}^T(\mathbf{x}_i)$, and $\mathbf{a}$ is the vector of the polynomial coefficients. A solution with respect to the unknown coefficient vectors allows to calculate the transfer matrix and also the interpolated data vector $\mathbf{d}^*$. If $\boldsymbol{\alpha}$ is known, Eq. (4.114) can be evaluated at any position for a given $\mathbf{x}$. For the discrete case involving all transfer points $\mathbf{x}_j^*$, this leads to

$$\mathbf{d}^* = \begin{bmatrix} \boldsymbol{\Phi}^* & \mathbf{P}^* \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ \mathbf{a} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Phi}^* & \mathbf{P}^* \end{bmatrix} \begin{bmatrix} \boldsymbol{\Phi} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{d} \\ \mathbf{0} \end{bmatrix} = \mathbf{M}\,\mathbf{d}\,. \tag{4.119}$$

Therein, the matrix $\boldsymbol{\Phi}^*$ has components based on the basis function $\phi_{ij}^* = \phi(||\mathbf{x}_i^* - \mathbf{x}_j||)$, and the polynomial matrix $\mathbf{P}^*$ consists of $\mathbf{p}^T(\mathbf{x}_i^*)$. However, two open issues need to be discussed. First, the type of the applied polynomial. In [9], a linear polynomial $\mathbf{p}(\mathbf{x}) = [1,\, x,\, y,\, z]^T$ was chosen, which can be seen as the standard, commonly employed polynomial. Second, there is no information concerning the type of radial basis function. Different types of RBF are mentioned in [33, 32], the selected examples including *multi-quadratic bi-harmonic splines*

$$\phi(||\mathbf{x}||) = \sqrt{||\mathbf{x}||^2 + c^2}\,, \tag{4.120}$$

where c is a scalar to control the shape of the functions and *thin plate splines*

$$\phi(||\mathbf{x}||) = ||\mathbf{x}||^2 \log ||\mathbf{x}||\,. \tag{4.121}$$

In [33], both functions are denoted as robust, cost-effective and accurate.

Like the least-square fit procedure, transferring data between non-matching meshes with RBF leads to an additional equation system. This system needs to

be solved within every coupling iteration. This might increase the simulation time significantly, in particular for problems which are coupled over the volume where the coupling domain includes a huge amount of data points. Further aspects concerning the computer implementation of the data transfer with RBF can be found in Algorithm A-10.

Algorithm A-10 Algorithmic sketch of the radial basis function interpolation.

Initialize: $\mathbf{d} = [\mathbf{d}_0, \mathbf{d}_1, ..., \mathbf{d}_N]^T$; $\mathbf{x}^* = [\mathbf{x}_0^*, \mathbf{x}_1^*, ..., \mathbf{x}_{N^*}^*]^T$; $\mathbf{x} = [\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_N]^T$

1: Choose polynomial basis $\mathbf{p}$; Choose radial basis $\phi(\|\mathbf{x}\|)$
2: $i = 0$; $j = 0$
3: **for** $i < N$ **do**
4:     **for** $j < N$ **do**
5:         $\phi_{ij} = \phi(|\mathbf{x}_i - \mathbf{x}_j|)$
6:         $[\mathbf{\Phi}]_{ij} \leftarrow \phi_{ij}$
7:     **end for**
8:     $[\mathbf{P}]_i \leftarrow \mathbf{p}_i^T(\mathbf{x}_i)$
9: **end for**
10: Build $\mathbf{M} = \begin{bmatrix} \mathbf{\Phi} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{bmatrix}$
11: Solve $\begin{bmatrix} \mathbf{d} \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{\Phi} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \boldsymbol{\alpha} \end{bmatrix}$
12: $i = 0$; $j = 0$
13: **for** $i < N^*$ **do**
14:     **for** $j < N$ **do**
15:         $\phi_{ij} = \phi(|\mathbf{x}_i^* - \mathbf{x}_j|)$
16:         $[\mathbf{\Phi}^*]_{ij} \leftarrow \phi_{ij}$
17:     **end for**
18:     $[\mathbf{P}^*]_i \leftarrow \mathbf{p}_i^T(\mathbf{x}_i^*)$
19: **end for**
20: $\mathbf{d}^* = \begin{bmatrix} \mathbf{\Phi}^* & \mathbf{P}^* \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \boldsymbol{\alpha} \end{bmatrix}$

4.3.1.6   Shape function interpolation

Coupled problems that are discretized by finite elements give rise to another simple but efficient method: The requested data can be interpolated by using the shape functions of the element. However, the quantity of interest of the sending field *must* be given at the nodes, and one needs to have access the code itself, i.e. the procedure cannot be carried out with a black-box solver. For this method, there is no additional error that is related to the data interpolation since only the shape functions need to be evaluated at the coordinates of the data points of the other field. The extra computation time is also of minor interest.

4.3.2   Dynamic mesh update

Multi-physically coupled simulations that involve large deformations or rigid
body motions often require a dynamic mesh update. This means that the mesh of
the other field is adapted to the current deformation of the mechanical field in ev-
ery coupling iteration. For this reason, two domains – Ω_0^1 and Ω_0^2 – are considered,
which share a common boundary $\partial\Omega^c$. The first domain is discretized with finite
elements (solid domain) and the second with finite volumes (fluid domain). They
are coupled over the common interface, and data has to be transferred between
two non-matching grids. However, if the body starts to deform, the shape of the
common boundary $\partial\Omega^c$ changes as well. Also, the finite volume mesh must be
updated as illustrated in Figure 4.8. This consequently means that the boundary
motion of the finite element domain serves as a boundary condition that needs to
be enforced on the fluid domain.
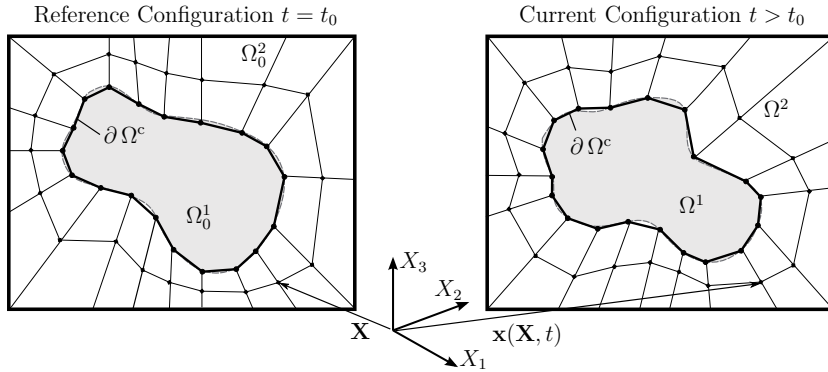
Figure 4.8: Mesh deformation update due to changes in the configuration

The objective is to find the unknown position points $\mathbf{x}_{\text{new}}$ of the new mesh cor-
responding to the prescribed displacements on $\partial\Omega^c$. The finite volume solver
OpenFOAM [130] offers several possibilities to treat dynamic mesh motion. A
brief review of some selected methods shall be given in the following. For a de-
tailed discussion and applications see [92, 93], for instance. In summary, three
different possibilities are outlined:

- **Pseudo-solid smoothing:** The mesh motion problem of the fluid mesh can
  be viewed as a solid body under large deformations with prescribed dis-
  placement boundary conditions. This is formally equivalent to an FEM so-
  lution to compute the new positions of all fluid mesh nodes. For the sake
  of simplicity and efficiency, a linear pseudo-solid equation with small de-
  formations is solved to approximate the nonlinear deformation of the finite
  volume mesh. This idea basically assumes that the motion of the mesh can
  be tackled by means of the balance of linear momentum for linear elasticity.
  This procedure was introduced by JOHNSON & TEZDUYAR [96].

- **Laplacian smoothing:** Another way to compute the mesh motion is to solve the Laplace equation on the fluid mesh, as proposed by LÖHNER & YANG [109]. Since the new positions at the nodes are of interest, this is also equivalent to solving the equation on a finite element mesh. Again, the displacements on the common coupling interface are taken to define the prescribed displacement boundary conditions. Let $\mathbf{v}$ be the mesh deformation velocity, then the Laplace equation reads

$$\text{div}\left(\gamma\,\text{grad}\,\mathbf{v}\right) = \mathbf{0}\,, \tag{4.122}$$

where γ denotes the diffusion coefficient. Having discretized this equation, one can calculate $\mathbf{v}$ at every mesh point so that, finally, the new node coordinates read

$$\mathbf{x}_{\text{new}} = \mathbf{x}_{\text{old}} + \mathbf{v}\,\Delta t\,. \tag{4.123}$$

The diffusion coefficient γ is used to improve the mesh quality by reducing the distortion of the volumes. There exists several ways to choose γ. A detailed study was carried out in [92]. Typically, $\gamma = \gamma(l)$ is a function of the face center distances to the nearest boundary l, which can, for example, be based on a linear $\gamma(l) = l^{-1}$, quadratic $\gamma(l) = l^{-2}$ or exponential $\gamma(l) = \exp(-l)$ dependency.

- **Mesh motion based on radial basis functions:** [31] addresses a new method to treat the mesh motion problem with the aid of radial basis functions. Similar to the use of RBFs for transferring data across non-matching meshes, see the previous Section 4.3.1.5, they can also be employed to determine the node displacements of the whole finite volume mesh of the fluid domain. To begin with, the interpolation coefficients $\boldsymbol{\alpha}$ and $\mathbf{a}$ are computed based on Eq. (4.118). Afterwards, one can use them to interpolate the displacement $\mathbf{u}_{\text{new}}$ for all mesh points

$$\mathbf{u}_{\text{new}} = \begin{bmatrix} \boldsymbol{\Phi}_{\text{FV}} & \mathbf{P}_{\text{FV}} \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ \mathbf{a} \end{bmatrix}\,, \tag{4.124}$$

where $\boldsymbol{\Phi}_{\text{FV}}$ is the matrix containing the information of the basis functions and $\mathbf{P}_{\text{FV}}$ is the polynomial matrix, both evaluated at all fluid mesh points. The new positions of the mesh nodes follow directly to

$$\mathbf{x}_{\text{new}} = \mathbf{x}_{\text{old}} + \mathbf{u}_{\text{new}}\,. \tag{4.125}$$

Applying RBFs for mesh motion is said to be very efficient and robust, also for large deformations, and the performance is superior to Laplacian smoothing – as mentioned in [93, 31].

In partitioned coupling procedures, the mesh update has to be performed in every iteration. This can be very costly. Thus, to reduce computational costs, it might be sufficient to perform only one update at the beginning of the new time-step, at least under certain conditions. For example, this could be the case if the change of the deformation during the coupling iteration is small.

4.4   Further improvements

There are different ways to improve the efficiency and accuracy of the partitioned coupling strategy, and two approaches shall be briefly explained in the following.

4.4.1   Temporal discretization

To begin with, the temporal discretization is taken under consideration. High-order time integration methods can be used to increase the accuracy of the simulation. In particular, DIRK – *diagonally implicit Runge-Kutta* – methods were shown to be applicable for a wide range of nonlinear single- and multi-field finite element analyses. In [71, 127], high-order time integration for finite strain viscoelasticity is considered, and the nonlinear heat conduction problem is treated in [143]. High-order in time is also possible for multi-field problems, as shown in [12], where FSI with thermal coupling is studied. Another improvement is to choose an adaptive time-step size, which can be used to reduce the computation time and also to increase the accuracy. This can be combined by using high-order time integration methods, as carried out by [148], for instance. For a more detailed introduction into high-order finite element methods in time see the aforementioned articles and also the literature cited therein.

Another approach to select the time increment for FSI applications can be found in [100]. The authors proposed an error-based time-step selection using an explicit predictor. Here, a new time-step is chosen based on the error between the current and the predicted solution. This procedure was also applied in [50] to adapt the time increment in a coupled finite strain thermo-elastic simulation.
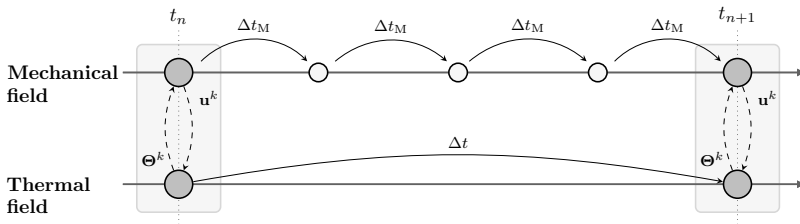
Figure 4.9: The idea of multi-scale time-stepping applied to thermo-mechanical coupling in a Gauss-Seidel based coupling scheme.

Further, as the fields involved are treated separately, one can also use a different time discretization for the individual sub-problems, as employed in [99]. This can be denoted as *multi-scale time-stepping* and may be a useful procedure for problems in which one field requires a much smaller time increment than the other field. Such candidates could be thermo-viscoplastic problems, for instance. The

idea of different time-step sizes is illustrated in Figure 4.9, where the time incre-ments between the thermal and the mechanical fields differ from each other. The data needed to ensure that the field has a smaller time-step size can be obtained by means of vector extrapolation based on Eq. (4.23).

4.4.2 Parallel simulation

Apart from improving the time integration, there are also methods to enhance the efficiency by means of parallelizing the coupling approach. To be more spe-cific, this means that the actual sub-systems are solved in parallel. However, this is not possible when a *block Gauss-Seidel* (GS) based procedure is employed as this requires to call the solvers in a serial or sequential manner as described in Section 4.1.2.1.

The other option – solving the sub-systems in parallel – is known as a *block Gauss-Jacobi* (GJ) procedure. Figure 4.10 shows a schematic sketch of the methods. The objective of a Gauss-Jacobi coupling scheme is evident: As the solvers do not have to wait on each other, the parallel solver serves to reduce the global computation time. Accordingly, coupling data are exchanged at the end of the iteration, which might lead to a situation in which more coupling iterations might be needed to achieve a converged solution. Recent work related to this coupling was published by SICKLINGER ET AL. [152], who introduced the so-called *interface Jacobian-based co-simulation*. They achieved promising results and demonstrated – for simple examples – that a co-simulation can be significantly faster than a serial simulation. However, they considered only problems involving a surface coupling domain. Moreover, access to the interface Jacobians is required, which can be a difficult and complex issue. In [151], the co-simulation of multi-field problems was applied to industrial examples.

In [162], another parallel coupling algorithm was proposed for FSI which com-bines the interface quasi-Newton (IQN) technique proposed by DEGROOTE ET AL. [36], see Section 4.2.5.1, with the Gauss-Jacobi formalism. They introduced two new algorithms, the *vectorial interface quasi-Newton* (VIQN) and the *parallel interface quasi-Newton* (PIQN) technique, and demonstrated a better performance compared to a sequential coupling algorithm. In [23], this scheme was extended to three and more coupled fields.

4.5 Global partitioned solution strategies

So far, the partitioned approach for the coupled problem was contemplated on a very abstract level. The physical fields are idealized as systems of nonlinear equations which are solved using block Gauss-Seidel iterations. In this section, numerical solution strategies for practical applications are developed, tailored to the coupled problem under investigation. Such strategies have to answer several questions that arise during the simulation process: For example, how to realize the communication and the data transfer between the fields? What might be
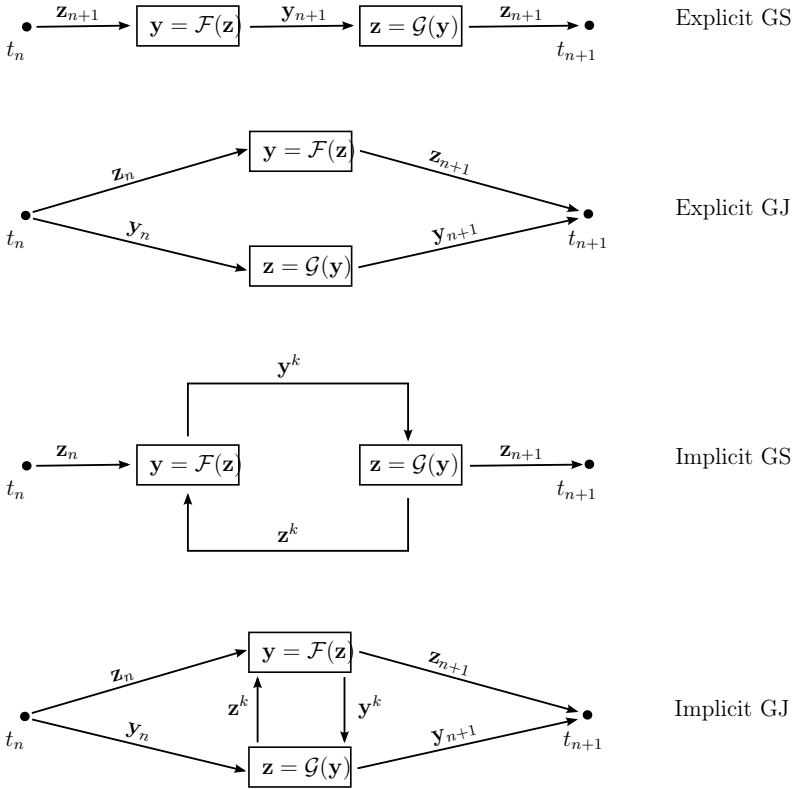
Figure 4.10: Schematic sketches of different coupling schemes: explicit Gauss-Seidel, explicit Gauss-Jacobi, implicit Gauss-Seidel and implicit Gauss-Jacobi.

the most efficient sequence to arrange the fields in? And what field should the acceleration and stabilization methods be applied to? Attempts to answers these questions will be given in the following section, based on examples of developing strategies to match the problems under consideration.

Two different physically coupled systems are computed, and the characteristics of the partitioned coupling strategy are studied in detail. In [51], strategies for three interacting fields were discussed, showing that an appropriate strategy is the key to a stable and efficient simulation. The proposed strategies were extended in [168], where volume coupling in conjunction with surface coupling was considered. Here, it was also demonstrated that the need for an appropriate strategy becomes more important with an increasing complexity of the problem.

4.5.1 Implementation

Solving multi-field problems requires a computer environment that executes and steers the simulation. In particular, it is desirable to take advantage of using different software and solvers for the fields involved. Consequently, it is necessary to develop a coupling tool to manage the communication between individual fields. To this end, the in-house communication software COMANA [15] was developed. It allows to connect an arbitrary number of specialized solvers. The tasks that COMANA has to tackle can be summarized as follows:

- **Initializing the simulation:** In the first step, the coupling tool connects with the solvers involved and starts the communication.

- **Exchanging coupling domains:** COMANA requires information about the coordinates of the data points within the coupling domains. The coupling domain is a surface or a volume, and it describes a region over which data between the solvers are exchanged. These regions are typically node sets, integration points or face center point clouds.

- **Communication with the solvers:** The kernel of COMANA is the communication with the connected solvers. This means that is has to give instructions to the solvers about when to send what data to which domain – and it also governs when the solvers are to receive data from other fields and what to do with them.

- **Transferring data:** In situations where the domain coordinates between the fields do not match, the data from the sending field need to be interpolated or projected onto the other domain of the receiving field. This is done by the coupling tool with the proposed methods of Section 4.3.

- **Handling convergence issues:** COMANA masters all issues related to the convergence of the partitioned approach. If necessary, it applies stabilization and acceleration methods, checks whether the iterative process converged successfully, and it is also responsible for the time management, i.e. updating the current time.

- **Finalizing the simulation:** At the end of the simulation, the coupling software disconnects all solvers.

The general coupling concept of COMANA is illustrated in Figure 4.5.1, based on the example of two sequentially coupled solvers. The communication and the data transfer are done using the *MPI*[11] package, which is a communication protocol that operates independently of the program language. An MPI application allows to run several programs in parallel on a distributed memory system. In a coupled simulation, each solver has its own process, and the MPI ensures communication and fast data exchange between the processes involved.

---

[11]MPI is the abbreviation for *message passing interface.*
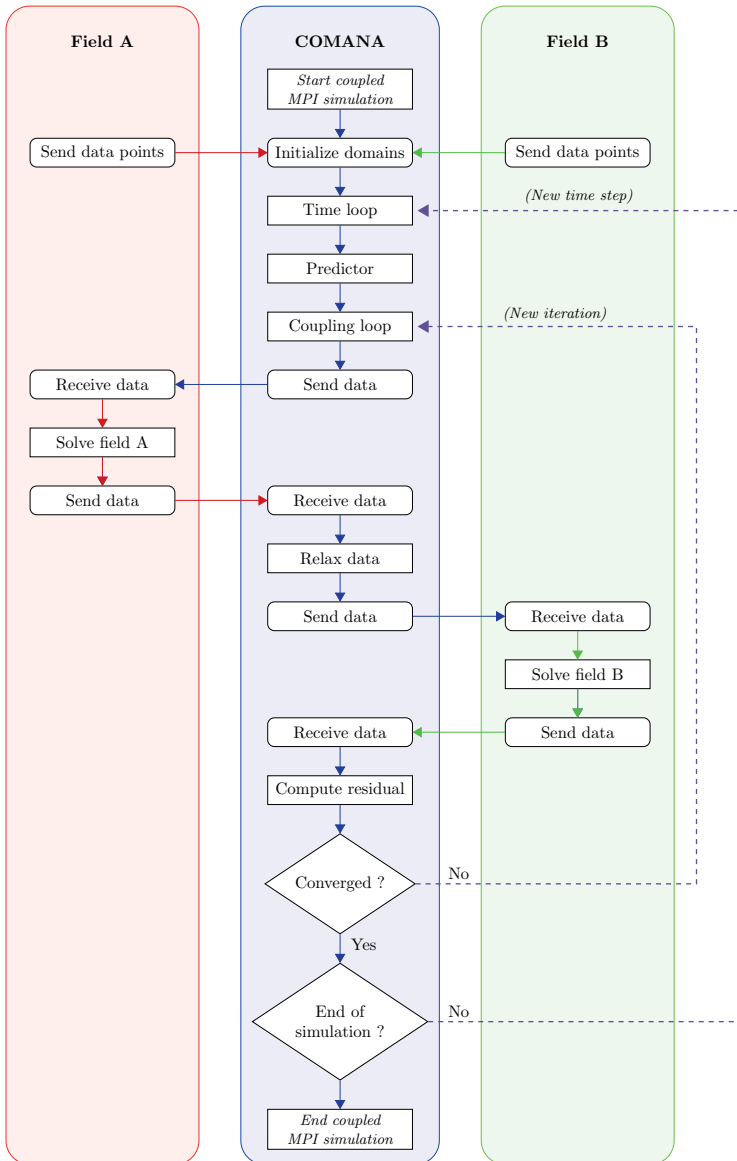
Figure 4.11: Communication flow chart of the implemented coupling procedure on the example of two coupled fields.

4.5.2 Volume-coupled problem: nonlinear thermoelasticity

The interaction of a thermal field with an elastic, mechanical field is a classical candidate for volume coupling. With regard to the coupling strategy, two coupled fields can be seen as the most straightforward combination. The driving field, i.e. the field that is solved first during the iteration process, is typically the one that acts and, thus, enforces a reaction of the second field.

A possible way to treat coupled thermo-mechanical problems is depicted in Fig. 4.12, where the mechanical field is solved before the thermal field. A thermal predictor is applied and the acceleration methods from Section 4.2 are employed to correct the temperature. In the following, the proposed partitioned coupling strategy is applied to nonlinear finite strain thermoelasticity. Here, the focus is on the example introduced by ARMERO & SIMO [3], based on which they demonstrated their adiabatic split. This was also discussed in [90, 50]. The variables of interest that are exchanged between the solvers are the discrete deformation gradients[12] ($\mathbf{F}$) and the discrete temperature values ($\boldsymbol{\Theta}$) at the Gauss points. The problem is coupled over the volume so that the coupling domain includes the whole geometry.
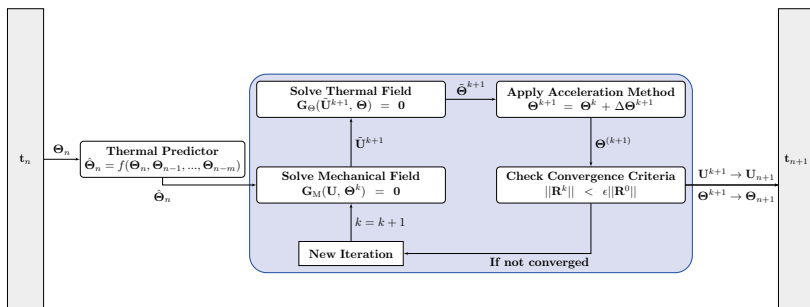
Figure 4.12: Partitioned coupling strategy for thermo-mechanical coupling.

4.5.2.1 Constitutive model for nonlinear thermoelasticity

The governing equations for nonlinear thermoelasticity were introduced in the second chapter. The mechanical field is governed by the quasi-static balance of linear momentum, and the corresponding boundary-value problem is given by Eqns. (2.74)-(2.76). The thermal field is described by means of the equation of heat conduction, see Eqns. (2.93)-(2.96) for the initial boundary-value problem. However, some kinematic relations and also the choice of the free-energy function shall be discussed in the following.

---

[12]In this example, it is more practical to transfer the deformation gradient (or the strain tensor) instead of the displacements as the thermal field is directly coupled to these quantities.

First, the deformation gradient is multiplicatively split into a mechanical and thermal part $\underset{\sim}{F} = \underset{\sim}{F}_{M} \underset{\sim}{F}_{\Theta}$, see [110, 166, 111, 74] and the literature cited therein for detailed discussions. Employing this split consequently leads to $\det \underset{\sim}{F} = J = J_{M} J_{\Theta} > 0$. For the mechanical part, the volumetric, isochoric split $\underset{\sim}{F}_{M} = \bar{\underset{\sim}{F}}_{M} \hat{\underset{\sim}{F}}_{M}$, see Eq. (2.15), is applied. The thermal part is assumed to be pure volumetric. Based on this assumption, the thermal deformation gradient takes the form $\underset{\sim}{F}_{\Theta} = \vartheta(\Theta) \underset{\sim}{I}$ with $J_{\Theta} = \vartheta^3(\Theta)$, where ϑ is known as the *thermal stretch-ratio*. Following [166], this parameter is connected to the volumetric thermal expansion coefficient $\alpha_{\Theta}(\Theta)$ over a differential equation from which ϑ can be calculated by a simple integration

$$\alpha_{\Theta} = \frac{1}{\vartheta} \frac{\mathrm{d}\vartheta}{\mathrm{d}\Theta} \quad \rightarrow \quad \vartheta(\Theta) = \exp\left[\int_{\Theta_0}^{\Theta} \alpha_{\Theta}(\Theta)\,\mathrm{d}\Theta\right] \approx \exp\left[\alpha_{\Theta}\left(\Theta - \Theta_0\right)\right]. \quad (4.126)$$

For the Green-Lagrange strain tensor $\underset{\sim}{E} = 1/2(\underset{\sim}{C} - \underset{\sim}{I})$, some mathematical rearrangements serve as a basis to determine the expressions for the mechanical (elastic) and the thermal strains

$$\underset{\sim}{E}_{M} = \frac{1}{\vartheta^2}\left(\underset{\sim}{E} - \underset{\sim}{E}_{\Theta}\right) \quad \text{and} \quad \underset{\sim}{E}_{\Theta} = \frac{1}{2}\left(\vartheta^2 - 1\right)\underset{\sim}{I}. \quad (4.127)$$

Apart from kinematic relations, an expression for the free-energy function Ψ is required. This is achieved by a function that, similar to an example in [3], consists of two parts

$$\Psi = \Psi_{M}(J_{M}, \bar{\underset{\sim}{C}}) + \Psi_{\Theta}(\Theta), \quad (4.128)$$

a mechanical and a thermal part. For the mechanical contribution Ψ_{M}, the choice

$$\rho_0\,\Psi_{M}(J_{M}, \bar{\underset{\sim}{C}}) = \frac{K}{2}\,\ln^2(J_{M}) + c_{10}(\operatorname{tr}\bar{\underset{\sim}{C}}_{M} - 3) = U(J_{M}) + \bar{w}(\bar{\underset{\sim}{C}}) \quad (4.129)$$

is made. Here, K denotes the bulk modulus and $\mu = 2\,c_{10}$ is the shear modulus. The mechanical part is also decomposed into a pure volumetric part $U(J_{M})$ and an isochoric part $w(\bar{\underset{\sim}{C}})$. The latter part is the classical Neo-Hooke model, see [84] for instance, and the volumetric part is used, among others, in [154, 3] for instance. Since thermal expansion is purely volumetric, $U(J_{M})$ comprises the coupling between the mechanical and the thermal field. Using $J_{M} = J/J_{\Theta}$ and $J_{\Theta} = \exp\left[3\,\alpha_{\Theta}\left(\Theta - \Theta_0\right)\right]$, the volumetric portion can be rewritten in terms of J and Θ to

$$U(J, \Theta) = \frac{K}{2}\,\ln^2(J/J_{\Theta}) = \frac{K}{2}\left[\ln J - \ln J_{\Theta}\right]^2 = \frac{K}{2}\left[\ln J - 3\,\alpha_{\Theta}\left(\Theta - \Theta_0\right)\right]^2. \quad (4.130)$$

The thermal contribution can be calculated by means of the expression for the specific heat given by

$$
\begin{aligned}
c_{\mathrm{p}}(\Theta) &= -\frac{\partial^2\Psi}{\partial\Theta^2}\Theta = -\Theta\,\frac{\partial^2}{\partial\Theta^2}\left(\frac{K}{2\rho_0}\left[\ln J - 3\,\alpha_{\Theta}\left(\Theta - \Theta_0\right)\right]^2 + \Psi_{\Theta}\right)\\
&= -\Theta\left(9\,\frac{\alpha_{\Theta}^2\,K}{\rho_0} + \frac{\mathrm{d}^2\Psi_{\Theta}}{\mathrm{d}\Theta^2}\right).
\end{aligned}
\quad (4.131)
$$

This example is based on the rough approximation that $c_{\mathrm{p}} = c$ is constant, so the expression for Ψ_Θ can be obtained by a double integration – resulting in

$$\Psi_\Theta(\Theta) = \int\limits_{\Theta_0}^{\Theta} \left(\int\limits_{\Theta_0}^{\Theta} -9\,\frac{\alpha_\Theta^2\,K}{\rho_0} - \frac{c_{\mathrm{p}}}{\Theta}\,\mathrm{d}\Theta \right) \mathrm{d}\Theta = \left(c_{\mathrm{p}} - \frac{9\alpha_\Theta^2 K}{2\rho_0}\,\Delta\Theta \right) \Delta\Theta - c_{\mathrm{p}}\Theta \ln\left(\frac{\Theta}{\Theta_0}\right).$$

(4.132)

Afterwards, it is possible to derive the constitutive relations for the stress tensor and for the entropy

$$\underset{\sim}{S} = 2\rho_0\,\frac{\partial\,\Psi_{\mathrm{M}}}{\partial\,\underset{\sim}{C}} = K\,(\ln J - 3\,\alpha_\Theta\,\Delta\Theta)\,\underset{\sim}{C}^{-1} + 2\,c_{10}\,J^{-2/3}\left(\underset{\sim}{I} - 1/3\,\mathrm{tr}(\underset{\sim}{C})\underset{\sim}{C}^{-1}\right)\,,$$

(4.133)

$$\xi = -\frac{\partial\,\Psi}{\partial\,\Theta} = -\frac{\partial}{\partial\,\Theta}\left(\frac{1}{\rho_0}\,U(J,\Theta) + \Psi_\Theta(\Theta)\right) = \frac{3\,\alpha_\Theta\,K\,\ln J}{\rho_0} + c_{\mathrm{p}}\,\ln\left(\frac{\Theta}{\Theta_0}\right)\,.$$

(4.134)

Finally, the expression for the thermo-elastic coupling term, see Eq. (2.89), that occurs in the heat equation of the thermal field can be calculated with Eq. (2.24) to

$$\gamma(\underset{\sim}{\dot{C}}, \Theta) = \rho_0\,\Theta\,\frac{\partial^{\,2}\Psi}{\partial\,\Theta\partial\,\underset{\sim}{C}}\cdot\underset{\sim}{\dot{C}} = -\frac{3\,\alpha_\Theta\,K\,\Theta}{2}\,\underset{\sim}{C}^{-1}\cdot\underset{\sim}{\dot{C}} = -3\,\alpha_\Theta\,K\,\Theta\,\frac{\dot{J}}{J}\,. \qquad (4.135)$$

4.5.2.2 *The model problem: expansion of a hollow cylinder*

The investigated problem serves as an example to demonstrate that the isothermal split, which is only conditionally stable, can be applied even for unrealistically high coupling strengths. To this end, the convergence acceleration methods introduced in Section 4.2 are used to retain algorithmic stability and to improve the convergence properties.

One-fourth of a hollow cylinder as depicted in Figure 4.13 is considered, which coincides with the example used by ARMERO & SIMO [3]. In this thesis, 24 high-order hexahedral finite elements with a polynomial degree of $p = 5$ are used to discretize the geometry, whereas the mentioned authors used axis-symmetric elements. Both fields are solved on the same mesh using the finite element code AdhoC [45] so that data can easily be transferred without any interpolation.

The boundary conditions are given in Figure 4.5.2.2. At the inner radius, a linear increasing displacement $\bar{u}(t) = 1t\,\mathrm{mm/s}$ is prescribed, and the outer surface temperature is set to $\bar{\Theta}_0 = 293.15$ K. Adiabatic conditions are assumed for the top and the bottom surface, which is why $\bar{q} = 0$ is prescribed. Further, the geometric dimensions are given in Table 4.4 and all required material parameters can be found in Table 4.5. In this study, special attention is placed on the thermal expansion coefficient since it can be used to regulate the coupling strength between the fields. In [3], a measure for the coupling strength of linear isotropic
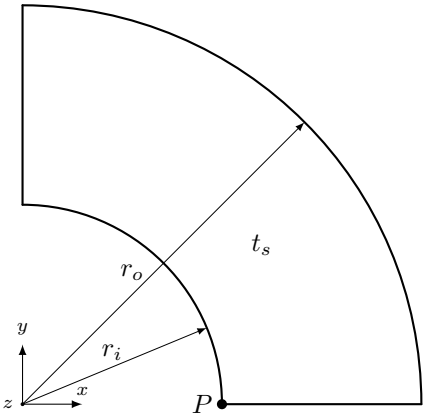
Figure 4.13: Geometric sketch of the cylinder.

Table 4.4: Geometrical set-up

Inner radius	r_i	0.05	m
Outer radius	r_o	0.1	m
Thickness	t_s	0.01	m

thermoelasticity was introduced, showing that the strength is proportional to α_Θ^2. Assuming that this material parameter has a similar influence in the nonlinear case, the degree of coupling is strengthen by increasing the thermal expansion coefficient.

Table 4.5: Material parameters

Material parameter	Symbol	Value	Unit
Density	ρ_0	7800	kg/m^3
Heat capacity	c_p	460	$J/kg\,K$
Conductivity	λ_Θ	45	$N/s\,K$
Thermal exp. coef.	α_Θ	1.5×10^{-5}	$1/K$
Bulk modulus	K	1.642×10^{11}	N/m^2
Shear modulus	μ	0.802×10^{11}	N/m^2

4.5.2.3 Results

To begin with, the temperature evolution at the result point P at $x = 10, y = 0, z = 0$ is considered. In Figure 4.15(a) the temperature is plotted against the time for

$$\bar{\mathbf{u}} = \begin{bmatrix} 0 \\ u_y \\ u_z \end{bmatrix}$$

$\bar{q} = 0$

$\bar{\boldsymbol{\Theta}} = \boldsymbol{\Theta}_0$

$\bar{\mathbf{u}} = \mathbf{u}(t)$
$\bar{\boldsymbol{\Theta}}(t = 0) = \boldsymbol{\Theta}_0$

]

$\bar{\mathbf{u}} = \begin{bmatrix} u_x & 0 & u_z \end{bmatrix}^T$
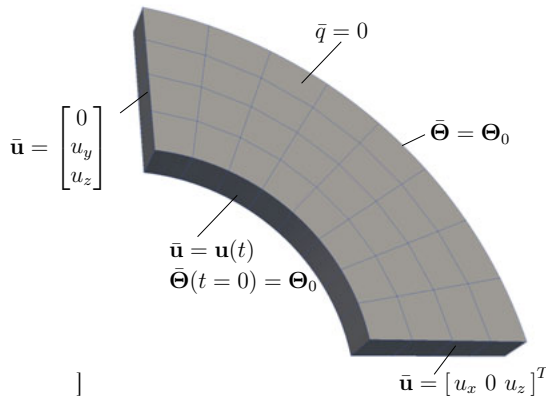
Figure 4.14: High-order finite element mesh of the cylinder.

two different time-step sizes Δt at the result point. Two different coupling procedures are considered, as explained in Section 4.1.2: explicit coupling, henceforth abbreviated with *Exp* and implicit coupling *Imp* with an iteration tolerance of $\epsilon = 10^{-4}$. Note that this tolerance needs to be satisfied by both fields. As depicted in Figure 4.15(a), the simulation runs stable without any convergence problems. The situation deteriorates dramatically if the thermal expansion coefficient is reset from $\alpha_\Theta = 1.5 \times 10^{-5}$ 1/K to $\alpha_\Theta = 1.5 \times 10^{-4}$ 1/K as shown in Figure 4.15(b). Here, a stable solution cannot be obtained. The implicit procedure diverges in the first time increment and is therefore not plotted in this figure. The explicit method has no convergence criteria to fulfil, and a "zigzag"-shaped temperature evolution is obtained for the larger time increment ($\Delta t = 1$ s).

For the small time-step size ($\Delta t = 0.1$ s), the situation is even worse, and the temperature response becomes unstable at a certain time. Again, the temperature evolution in the case of implicit coupling is not shown for this time increment. The instabilities are very similar to those observed by ARMERO & SIMO [3]. To avoid these difficulties, they proposed to use an adiabatic split as introduced in Section 4.1.2.2. In [50], external convergence stabilization methods were employed to find a stable solution. As depicted in Figure 4.15(b), the dynamic secant relaxation (DSR) and the quasi-Newton reduced-order method (QN) can be used to obtain a stable temperature response. Both schemes are able to bring the thermal and the mechanical field into equilibrium within an implicit coupling procedure.

A remark concerning the results: Due to the fact that the thermal expansion coefficient was chosen unrealistically high, the temperature response is also physically unrealistic. Nevertheless, this example clearly demonstrates that the applied acceleration schemes lead to a significant increase in the "chances" for a stable solution, even for unrealistically high coupling strengths.
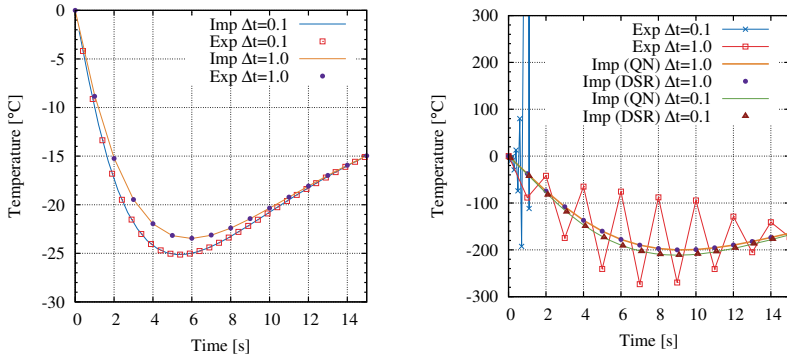
(a) Stable response: $\alpha_\Theta = 1.5 \times 10^{-5}\,1/\text{K}$.   (b) Unstable response: $\alpha_\Theta = 1.5 \times 10^{-4}\,1/\text{K}$. Stable solution by applying QN / DSR.

Figure 4.15: Temperature evolution at the result point for different coupling strength levels.

In the next step, some convergence properties of the DSR and the QN are determined. To this end, the coupling iteration residual Λ, see Eq. (4.13), for the very first time increment is taken under consideration, as depicted in Figure 4.16(a). Moreover, the number of Gauss-Seidel coupling iterations for every time-step is given in Figure 4.16(b). Here, the iteration tolerance that needs to be satisfied is set to $\epsilon = 10^{-3}$. A detailed overview of the results can also be found in Table 4.7. It becomes apparent that the QN method performs slightly better than the DSR method. The average number of iterations is lower, meaning that less solver calls are needed and, thus, the overall computation time is lower too. An interesting property might be the rate of convergence b of the acceleration method related to the first time increment. Let Λ^k be the iteration residual at the k-th iteration and $\Delta k = k_{end} - k_{start}$ a certain interval of coupling iterations, then the rate of convergence is defined as

$$b^k = -\frac{\Delta \log \Lambda^k}{\Delta k} = -\frac{\log \Lambda^k - \log \Lambda^{k-1}}{k_{end} - k_{start}} . \tag{4.136}$$

The parameters b^k calculated for the first time increment are listed in Table 4.6. A mean value $\bar{b}$ of the convergence rate is given in Table 4.7. With $\bar{b} = 1.28$ (for $\Delta t = 1.0\,\text{s}$), the QN method shows the fastest convergence.

For this example, both methods require many iterations to achieve convergence at the beginning of the simulation. With temporal progress, this number reduces until $k = 2$ iterations for the QN method. This is interesting, since it seems that only an *initial aid* by the acceleration methods is needed at the beginning to achieve a stable solution.

Table 4.6: Convergence rate (b^k) within the first time increment.

k	$\Delta t = 1.0$s		$\Delta t = 0.1$s	
	DSR	**QN**	**DSR**	**QN**
3	0.43	1.48	-0.19	1.59
4	1.77	0.81	1.86	0.87
5	1.16	1.50	0.54	0.76
6	0.27	1.22	0.86	0.86
7	0.98	1.21	0.69	0.82
8	1.41	1.48	0.77	0.97
9	0.51		0.70	1.07
10	0.55		0.59	
11			0.38	
12			1.12	
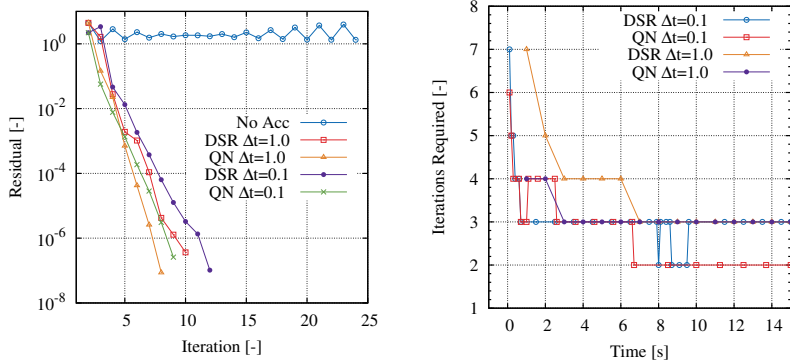
(a) Coupling iteration residual (Λ^k) within the first time increment.

(b) Number of coupling iterations for every time-step. The iteration tolerance is set to $\epsilon = 10^{-3}$.

Figure 4.16: Convergence behavior of the QN and DSR method for strong coupling levels.

4.5.2.4 Convergence study

Section 4.2.6 addresses a convergence study performed on a nonlinear system of equations, which will be done with this exampleas well. To this end, the number of iterations required per time-step ς is investigated to assess the performance of the applied convergence acceleration method. The results of this study are summarized in Table 4.8. Here, the values of the average number of iterations are given for different time-step sizes, where $\Delta t = \{0.1,\ 1.0\}$ s, and for a different point of application of the acceleration methods, either after the mechanical or

Table 4.7: Convergence properties for QN and DSR methods: average number of iterations per time-step (ς) and average convergence rate ($\bar{b}$) for the first time increment.

	$\Delta t = 1.0$s		$\Delta t = 0.1$s	
	ς	$\bar{b}$	ς	$\bar{b}$
DSR	3.67	0.89	3.01	0.73
QN	3.13	1.28	2.59	0.99

after the thermal field. The sequence, however, of the fields remains unchanged so that first the mechanical field is solved followed by the thermal field.

Nearly all methods can be used to find a convergent solution. Only the LE method does not converge in case of $\Delta t = 0.1$s when applying it to the mechanical field. The fastest acceleration can be achieved by means of the QN methods, in particular if old values from previous time-steps are reused. Taking the solutions from the last 10 iterations into account, an average of 2.05 iterations per time-step can be accomplished. Besides, the DSR works well, and the BR shows a promising enhancement of the convergence too. It turns out that the vector sequence acceleration schemes (Section 4.2.4) cannot be used in this example as they completely fail to balance the fields.

Table 4.8: Convergence study on the example of strongly coupled nonlinear thermoelasticity: comparison of the number of coupling iterations per time-step (ς) for different acceleration methods applied to the mechanical or the thermal field.

Point of application:	Mechanical field		Thermal field	
	$\Delta t = 1.0$s	$\Delta t = 0.1$s	$\Delta t = 1.0$s	$\Delta t = 0.1$s
DAR1	5.13	4.93	5.0	4.84
DSR	3.73	3.01	3.67	3.01
LE	5.4	NC	4.27	3.82
QN	4.33	4.11	3.13	2.59
QN (5)	3.87	2.54	2.8	2.05
QN (10)	2.87	2.13	2.67	2.04
QN (15)	3.33	2.19	2.67	2.05
BR	4.33	4.40	4.46	4.42

The methods to accelerate the convergence within the Gauss-Seidel coupling iteration loop can be combined with the vector prediction method. As shown in Table 4.9, a prediction of the starting values at the beginning of every time-step will further reduce the number of iterations required. For this example, the temperature is extrapolated for the mechanical field. In addition, the predicted data serve as the very first solution of the iteration process and, thus, the acceleration methods can be applied one iteration earlier. As demonstrated in Table 4.9, a thermal predictor (TP) of the order $p = 2$ delivers the most effective initial guess

for this example.

Table 4.9: Convergence study for different acceleration methods in conjunction with vector prediction.

Point of application:	**Mechanical field**		**Thermal field**	
	$\Delta t = 1.0$s	$\Delta t = 0.1$s	$\Delta t = 1.0$s	$\Delta t = 0.1$s
DSR + TP(1)	3.46	2.22	3.60	2.21
DSR + TP(2)	3.40	2.15	3.30	2.10
DSR + TP(3)	3.13	2.42	3.0	2.25
QN + TP(1)	4.33	3.0	3.01	2.11
QN + TP(2)	4.13	3.01	2.60	2.09
QN + TP(3)	3.60	3.36	2.67	2.85
QN(10) + TP(2)	2.93	2.41	2.73	2.07
DAR + TP(2)	4.73	3.39	4.4	3.35
LE + TP(2)	4.67	NC	3.13	2.38
BR + TP(2)	4.07	3.26	4.0	3.18

4.5.3 Coupled thermal-radiation analysis

In this example, the Dirichlet-Neumann coupling between a thermal (heat conduction) and a radiation field is investigated. The formulation of this surface coupling was introduced in Section 2.3.4. The following study aims to verify the proposed formulation with other codes as well as to study the performance, possibilities and frontiers of the different numerical methods for the analysis of radiative heat transfer.

4.5.3.1 *Partitioned coupling strategy*

First, some general remarks concerning a partitioned coupling strategy for the coupled thermo-radiative problem: In contrast to the coupled nonlinear thermoelasticity problem, two different solvers are employed. The finite element solver AdhoC is used for the thermal field, while the radiation field is computed by the finite volume solver OpenFOAM [130]. With respect to the coupling strategy, this has serious consequences. Since both methods utilize different discretizations, the coupling data need to be interpolated on the other mesh before they can be transferred. In this example, the barycentric triangle mapping concept is applied, see Section 4.3.1.3.

In a coupled thermal-radiation analysis, the temperature on the boundary surface mesh Θ_s is mapped on the surface mesh of the radiation field. The other quantity is the radiation heat flux vector $\mathbf{q}_r$, which is projected from the radiation field on the mesh of the thermal field. A possible coupling strategy for this problem is depicted in Figure 4.17. The driving field is the thermal field, and the thermal solver is thus called before the radiation field is solved. No convergence

acceleration methods are applied in this example, and the convergence properties of the coupling approach are not taken under investigation. However, the starting values of the thermal field are predicted at the beginning of every time-step.
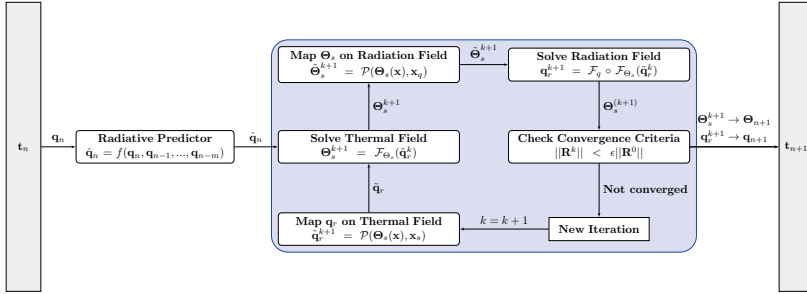
Figure 4.17: Partitioned coupling strategy for a surface-coupled thermal-radiation problem with non-matching coupling interfaces.

4.5.3.2   Model problem: thermal analysis of a fin

The objective of this example is to verify the Dirichlet-Neumann coupling between the thermal and the radiation field. To this end, a problem is considered for which the results were confirmed by numerical and experimental data – and can therefore serve as a benchmark example. Here, the model problem proposed by GLASS ET AL. [68] is recomputed, which is also included in the benchmark example catalogue of the commercial finite element software Abaqus [1]. This benchmark problem involves the heat and radiation analysis of a transient combustion process where a fin is subjected to fire effects and loading.

The geometry is depicted in Figure 4.18. Due to symmetry, only one fin on the inner block is taken under consideration. The space between the fin and the outer wall is the fluid domain Ω^Σ, which is filled with a gas and henceforth denoted as *outer fluid*. The block, to which fins are attached in a right angle, is also surrounded by a gas, here referred to as the *inner fluid*. In the following, the domain of the thermal field Ω^Θ embraces one fin and a cutout of the inner block, and it is discretized with $n_e = 92$ high-order hexahedral finite elements with a polynomial degree of $p = 4$, see also Figure 4.18. Further, the geometrical and physical data for the thermal field are given in Table 4.10.

The outer fluid domain Ω^Σ is discretized by means of $n_v = 13500$ finite volume cells as shown in Figure 4.19. In the following, the high-order code AdhoC [45] is utilized to solve the thermal field, and the finite volume solver OpenFOAM [130] is applied to the outer fluid field. For the special case in which the outer fluid is approached by vacuum conditions, OpenFOAM is used to calculate the view factors to account for view factor radiation. For the case of a participating medium,
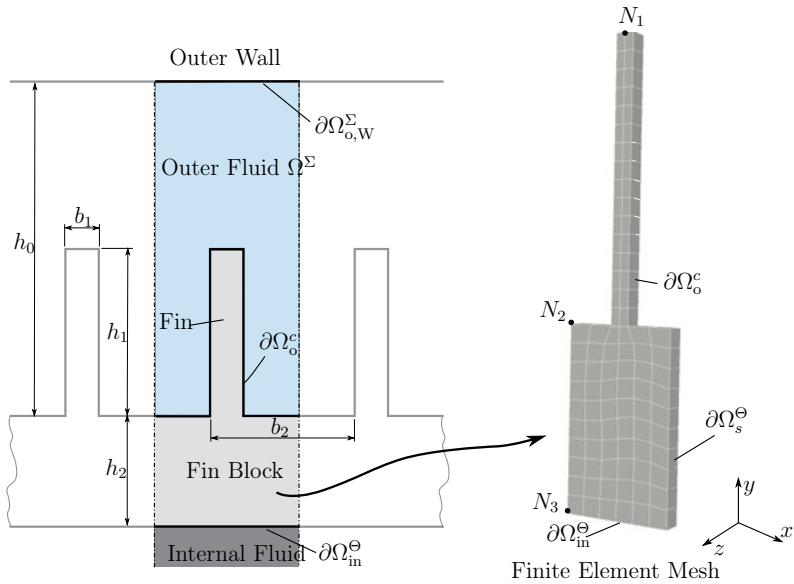
Figure 4.18: Schematic sketch of the finned wall, its surroundings and the finite element mesh.

Table 4.10: Geometrical and physical data for the thermal field.

Fin width	b_1	0.01	m
Base width	b_2	0.06	m
Wall distance	h_0	0.30	m
Fin height	h_1	0.15	m
Block height	h_2	0.10	m
Heat capacity	c_{p}	500	$\mathrm{J/kg\,K}$
Density	ρ	7800	$\mathrm{kg/m^3}$
Conductivity	λ_Θ	50	$\mathrm{W/m\,K}$

both convection and radiation are taken under consideration. Here, the classical k-ε turbulence model is used to tackle the turbulent, convective heat transfer in the outer fluid. Further, the P_1-method, see Section 3.4.2, and the fvDOM, see Section 3.4.3, are implemented to account for the thermal radiation.

Based on [68], the example consists of three different analysis phases:

1. The first phase is a steady-state heat transfer step for applying the initial conditions.
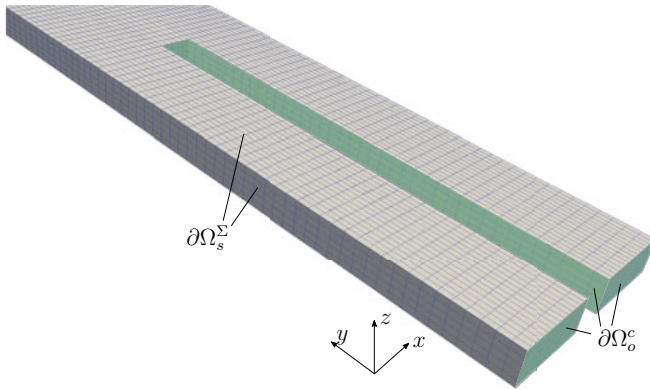
Figure 4.19: Portion of the finite volume mesh for the radiation field. The coupling interface $\partial\Omega_o^c$ with the thermal field is highlighted in green.

2. The second step deals with a 30-minutes combustion phase in the outer fluid, leading to a transient heat transfer inside the fin. Energy is transferred from the outer fluid through the fin and its block to the inner fluid.

3. The third phase is a 60-minutes transient cool down period in which the conditions are reset to those of the first phase.

Since the thermal radiation analysis is coupled over the surfaces, special attention must be paid to the description of the boundary. In this example, the coupling domain is given by the common interface of the thermal and the radiation field (outer fluid) and is denoted as $\partial\Omega_{o}^c$, as shown in Figure 4.18. In view of thermal radiation, a constant emissivity of $\varepsilon_w = 1.0$ is assumed for the outer wall $(\partial\Omega_{o,W}^\Sigma)$, and $(\partial\Omega_o^c)$ $\varepsilon_f = 0.8$ is used for the fin and for the outer block walls. The wall temperature is set to $\Theta_w = 38\,°C$. Besides, heat transfer due to convection must be described on the interfaces. Between the inner fluid and the fin's block, convection is approximated as a boundary condition by means of $q_c = h_{in}\,(\Theta - \Theta_f)$ on $\partial\Omega_{in}^\Theta$ where the inner fluid temperature is $\Theta_f = 373.15$ K. Here, the film coefficient h_{in} is temperature-dependent and reads $h_{in}(\Theta) = 500\,(\Theta - \Theta_f)^{1/3}$ W/m²K. The convection between the boundary of the outer fluid and the thermal field regarding the coupling interface $\partial\Omega_o^c$ varies within the analysis phase and also with the numerical method used to approximate the radiation field.

Table 4.11: Heat convection between the outer fluid and the fin walls. All film coefficients (in W/m²K) in case of the VFM are taken from [1] and the convection velocity u_{conv} from [168].

Method	Convection approach	Phase 2	Phase 3
P1	Directly by the radiation solver	2 m/s	2 m/s
fvDOM	Directly by the radiation solver	2 m/s	2 m/s
VFM	Bound. cond. in heat equation	$h_2 = 10$	$h_3 = 2(\Theta - \Theta_{\text{f}})^{1/3}$

For the first phase, a steady state heat convection analysis with $q_{\text{c}} = h_1(\Theta - \Theta_{\text{f}})$ on $\partial\Omega_{\text{o}}^{\text{c}}$ is carried out. During this phase, the film coefficient is set to $h_1 = 2(\Theta - \Theta_{\text{f}})^{1/3}$. For the second and the third phase, convection is either approximated via the boundary condition or directly computed by the radiation solver. The latter case is used to apply the P_1-model or the fvDOM. Then, the convective fluid velocity needs to be prescribed. Using the VFM, this is not possible since the energy equation of the fluid is not solved for this method. Details can be found in Table 4.11, which gives an overview of the different convection models used for the outer fluid. Moreover, several material parameters are required to finalize the model description. They are given in Table 4.12 and were taken from [168]. For a detailed explanation of the model problem, the interested reader is referred to [68, 168].

Table 4.12: Physical properties of outer fluid at 20°C taken from [168].

Material parameter	Symbol	Value	Unit
Absorption	α	0.70	1/m
Density	ρ	1.20	kg/m³
Dynamic viscosity	η	1.75×10^{-5}	kg/m s
Gravitation constant	g	9.81	m/s²
Heat capacity	c_{p}	1005	J/kg K
Prandtl number	Pr	0.7	-
Pressure	p_0	1.00×10^5	kg/m s²

4.5.3.3 Results for vacuum

To begin with, the outer fluid is neglected and the space between the fin and the outer wall is described by a vacuum. Thus, there is no convective heat transfer between the outer fluid and the fin and, therefore, only convection due to the inner fluid is taken under consideration by means of $q_{\text{c}} = h_{\text{in}}(\Theta - \Theta_{\text{f}})$ where the film coefficient is taken as $h_{\text{in}}(\Theta) = 500(\Theta - \Theta_{\text{f}})^{1/3}$ W/m²K and the fluid temperature is 100 °C. The initial temperature is set to $\Theta(t = 0) = \Theta_0 = 0$ °C. Moreover, the three analysis phases are reduced to one phase with a time period of $T = 5000$ s.

In the following, the temperature at the tip of the fin is compared for three different cases: In the first case, the thermal radiation analysis is carried out by means of the view factor method. Secondly, radiation is approximated as a boundary model (BC-model) with $q_r = \varepsilon \, \sigma_{\mathrm{sb}}(\Theta^4 - \Theta_\infty^4)$ where $\varepsilon = 0.8$ and $\Theta_\infty = 38\,°\mathrm{C}$ is used and, lastly, thermal radiation is fully neglected $q_r = 0$. In Figure 4.20, the temperature at the node N_1 is plotted against the time for these three cases. As depicted, the influence of thermal radiation increases with higher temperatures. Assuming that the VFM delivers the most realistic results in this study since it allows to take the geometric influence into account, one can conclude that the boundary model underestimates the temperature – whereas no radiation leads to an overestimation. This is physically correct since there is no thermal energy transfer by radiation whereas the heat loss in the case of the boundary model is too high. The reason for this is that incoming radiative heat flux due to reflection with the outer wall and the other fins cannot be incorporated. For this example, one can conclude that the boundary model shows larger deviations than if thermal radiation were completely neglected. This clearly demonstrates that the boundary model can only be used to give a rough approximation for radiating surfaces that have complex geometries and that interact with other surfaces. However, with increasing the distance between the radiating surfaces, the boundary model approaches the radiative heat transfer with increasing accuracy.
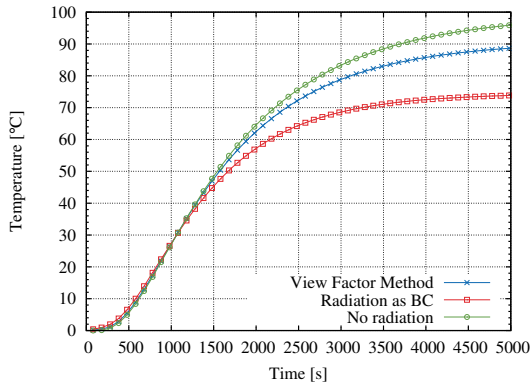
Figure 4.20: Developing temperature for the case of vacuum at the fin tip (node N_1).

4.5.3.4 Results for participating medium

After computing the temperature evolution in vacuum, the space is now filled with a fluid. In contrast to the vacuum, the convection between the fin walls and the outer fluid needs to be taken into account, as described in Section 4.5.3.2.

For the VFM, convection is prescribed on the coupling boundary, whereas for the P_1-method and the fvDOM, the convective part is included in the solution of the energy equation for the fluid. In contrast to the previous simulation, a full computation over all three phases is carried out. First, the temperature evolution at the result nodes $N1$, $N2$ and $N3$ is investigated. In Figures 4.21 and 4.22, the nodal temperature for the different radiation models is compared to the reference results obtained with the commercial finite element solver Abaqus.

(a) Temperature at result node $N1$.   (b) Temperature at result node $N2$.
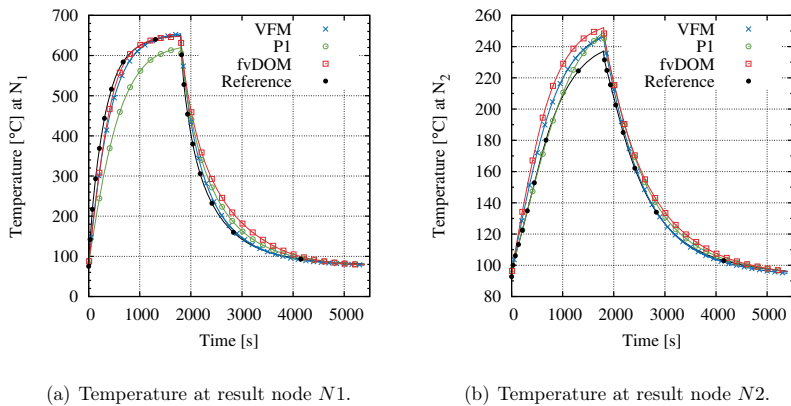
Figure 4.21: Temperature increase at the result nodes and comparison of different numerical radiation models.

At the beginning, the temperature development at the tip of the fin (node $N1$) is taken under consideration. As depicted in Figure 4.21(a), the results in the first and second phase ($T_2 = 1800\,\mathrm{s}$) coincide well with the reference case, particularly the fvDOM and the VFM. The P_1-method underestimates the maximum temperature, which demonstrates the rather conservative character of this method. During the third phase, which describes the cooling period, the fvDOM and P_1-method have slight discrepancies to the reference case whereas the VFM nearly reaches the decrease of the reference temperature. For the nodal temperature evolution at $N2$, as depicted in 4.21(b), and at $N3$, given by Figure 4.22, the end temperature of the heating period (phase 2) is slightly overestimated. Notwithstanding, the results obtained with the different radiation models are in very good agreement with the reference temperature. Referring to [168], the deviations can be explained as follows: For the outer fluid, there is a lack of information concerning the material properties. Thus, several assumptions for these properties have been made, which might be the reason of the deviations occurring for the fvDOM and the P_1-model. The reference simulation was carried out by Abaqus, where view factor radiation is implemented as well[13]. Surprisingly, there are also

---

[13]It must be reminded that the commercial FE solver Abaqus (Version 6.13 in this case) is able

deviations between the VFM and the reference solution obtained by Abaqus. In the present study, the view factors were calculated by means of the fluid solver OpenFOAM [130] which unfortunately does not allow to take the symmetry of the finned wall into account. This fact might be the source of the deviations between VFM and the reference solution.
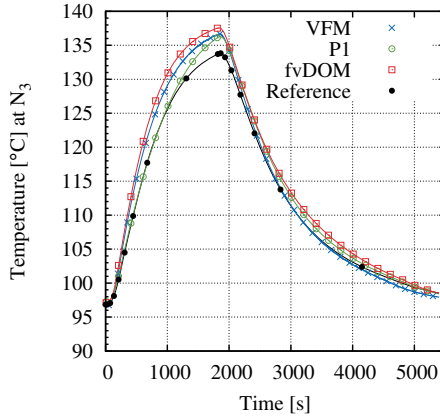
Figure 4.22: Temperature increase at the result node $N3$ and comparison of different numerical radiation models.

A quantitative comparison of the results at the nodes N_1, N_2 and N_3 is given in Table 4.13. In addition to the numerical results, the experimental data for the temperatures and its standard deviation is taken from [68]. Comparing the values to the coupled thermal-radiation analysis reveals the good agreement of the results. The maximum error is below 5% and can be found at $N1$ in the second phase when applying the P_1-method. Finally, it can thus be concluded that – in this example – the coupled Dirichlet-Neumann problem of thermal-radiation leads to promising results. The computations successfully combine different software as well as discretization schemes and also allow to approach the radiative and convective heat transfer in a participating medium. Thanks to the flexibility of the partitioned coupling strategy, this problem can be extended such that geometric changes of the structure can be included as well.

to perform view factor radiation analysis, but not in a multi-physics context. This means, for example, that deformations of the geometry are not allowed during thermal radiation analysis.

Table 4.13: Temperature at result nodes: comparison with benchmark results from Abaqus and data from Glass et al. [68]

Phase (Time)	Node	Glass et al. [68]		*Abaqus* [1]	*AdhoC / OpenFOAM*®		
		Average	Deviation		P_1-method	fvDOM	VFM
Phase 1 (initial)	N_1	75.8	0.2	75.7	79.2	76.7	74.0
	N_2	93.0	0.3	92.9	94.0	93.5	92.3
	N_3	97.0	0.1	96.9	97.3	97.1	96.7
Phase 2 (1800 s)	N_1	652.2	4.9	649.9	619.6	647.9	653.8
	N_2	238.6	6.6	237.2	249.1	252.1	246.3
	N_3	133.7	1.1	133.6	136.0	137.5	136.4
Phase 3 (5400 s)	N_1	80.4	0.7	80.9	79.3	78.4	78.6
	N_2	95.7	0.5	96.1	95.9	96.0	95.0
	N_3	98.4	0.2	98.5	98.4	98.5	98.0

4.5.4 Three or more coupled fields

In a partitioned analysis with several fields, a suitable coupling strategy is the key to obtain a fast simulation. If the number of fields is increased, the complexity of the simulation commonly increases as well – and it becomes more and more important to fine-tune the global coupling strategy. Unfortunately, a general statement about which might be the best strategy can hardly be given since this largely depends on the respective problem. Nevertheless, the following points may help to construct a well-suited and problem-orientated coupling strategy:

- First of all, the sequential arrangement of the fields is important. Again, the driving field should be the first one within the iteration process. If the whole system is treated in an explicit manner – meaning that iterating between the fields is neglected – the sequential arrangement is the only aspect that needs to be taken into consideration.

- An appraisal of the coupling strengths between the fields can help. If one field turns out to be only loosely coupled with the others, it may be excluded from the implicit iteration process to avoid unnecessary solver calls. In this case, it is solved only once in every time-step, either before or after the coupling iterations. In [51], this idea was applied to electro-thermo-mechanical problems and was referred to as *implicit-explicit coupling*. In cases where a cost-intensive field can be excluded, a significant reduction of the global simulation time is obtained.

  In Figure 4.23, the idea of an explicit-implicit strategy is exemplary illustrated on a three-field problem. An extension of this idea is to use an adaptive switch between implicit or explicit coupling, individually for the different fields. This can be controlled by computing the error between a predicted solution and the solution of the first iteration. If the error is small

enough, the iteration phase will be stopped and the solution from the first iteration is taken as the converged solution. If the error starts to increase again, one can switch back to the implicit procedure.

- The acceleration schemes need to be applied in such a way that they can develop their entire potential. In particular, the efficacious point of application has to be figured out. This again is a difficult issue that cannot be generalized, but has to be investigated for every problem. A good choice might be to accelerate the field that is expected to have the tightest coupling with the others.
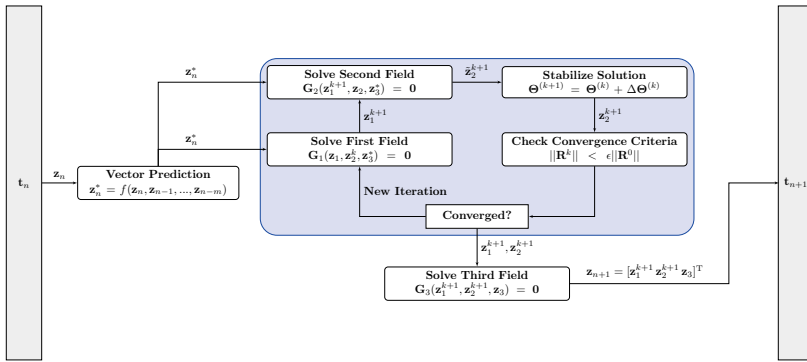
Figure 4.23: Combination of implicit and explicit coupling strategy for a three-field system.

Chapter 5 addresses examples of three- and four-field problems in detail, based on a coupling strategy that takes these three points into consideration. Special attention is placed on electro-thermo-mechanical problems involving thermal radiation, which finally leads to a volume- and surface-coupled problem.

Numerical examples

In this chapter, the proposed partitioned coupling strategy is applied to multi-physical problems with $n_\mathrm{f} > 2$ fields. The general purpose is to find the best suited strategy to the corresponding problem under consideration. To begin with, an electro-thermo-mechanical problem with finite deformations is studied, where thermal radiation is approximated as a boundary condition in the heat equation. In a second step, this is extended to four-fields by treating radiation as an additional field. For this example, all issues discussed in Chapter 4 – convergence acceleration, data transfer and dynamic mesh motion – are employed and different possibilities to perform the partitioned coupling process are discussed. The final example is to simulate the field assisted sintering process, first without powder consolidation and finally using a material model for highly compressible copper powder, which was developed by ROTHE [146].

5.1 Electro-thermo-mechanically coupled problem

Within the scope of the first example, the radiation field is excluded, and only the interactions of an electric, thermal and mechanical field are taken into account. An appropriate partitioned coupling strategy is developed to tackle the three-field problem. This strategy basically follows ERBTS ET AL. [51] and is applied to a nonlinear, thermo-elastic problem with a heat source resulting from an electric current.

5.1.1 Model problem: bimetallic beam

This example deals with the electro-thermo-mechanical modeling of a bimetallic beam consisting of two layers, one made of steel and one made of silver. This serves as an academic example for an in-depth study of the coupling strategy, from low up to unrealistically high coupling strengths.

The electrical field is used to generate high heating rates by means of Joule heating. Because the layers consist of different materials, the thermal strains in the layers differ from each other as well – leading to a bending of the beam. The deflection is assumed to be elastic and geometrically nonlinear. The material model

used here coincides exactly with the example of the thermo-elastic expansion of
the thick-walled cylinder from Section 4.5.2.
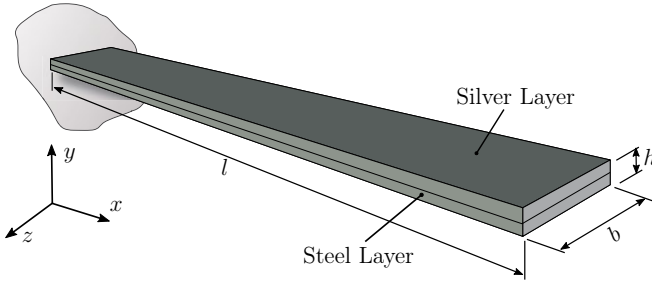
Figure 5.1: Schematic sketch of the bimetallic beam consisting of two layers.

Table 5.1: Geometrical set-up of bimetallic beam

Length	l	2	m
Width	b	0.1	m
High	h	0.01	m

The beam is discretized with $n_{\mathrm{e}} = 12$ high-order hexahedral finite elements
that have a polynomial degree of $p = 5$. All fields are solved on the same mesh,
which consequently means that the data transfer is exact and an interpolation
concept is not required. A schematic sketch of the beam is given in Figure 5.1
and the mesh is depicted in Figure 5.2. Concerning the boundary conditions, the
left-hand side of the beam is clamped ($\bar{\mathbf{u}} = \mathbf{0}$) and the electric potential is zero
($\bar{\varphi} = 0$) on this surface as well. On the right-hand side, the prescribed electric
potential increases exponentially in time with $\bar{\varphi}(t) = U_{\max}[1 - \exp(-\dot{U}_0/U_{\max}\, t)]$.
The temperature boundary conditions are given as follows: Thermal radiation is
approximated as $\bar{q} = \varepsilon\sigma_{\mathrm{sb}}(\Theta^4 - \Theta_\infty^4) = \bar{q}_{\mathrm{r}}$ with $\Theta_\infty = \Theta_0 = 0\,°\mathrm{C}$ over all surfaces of
the beam, except at the clamped (wall) face where $\bar{q} = 0$ is assumed. All boundary
conditions are illustrated in Figure 5.2. The initial temperature is set to $\Theta(t =
0) = \Theta_0 = 0\,°\mathrm{C}$. The geometric dimensions are given in Table 5.1 and the material
parameters for steel and silver are summarized in Table 5.2. In this example, all
parameters except the electric conductivity are assumed to be constant, i.e. they
do not depend on the temperature. In the following, it is assumed that the electric
conductivity takes the form

$$\lambda_\varphi(\Theta) = \frac{\lambda_{\varphi,0}}{1 + \alpha_\varphi\,(\Theta - \Theta_0)}\,, \tag{5.1}$$

where $\lambda_{\varphi,0} = \lambda_\varphi(\Theta_0)$ is the reference electric conductivity at the reference temper-
ature Θ_0 and α_φ denotes the linear temperature coefficient.

Top View
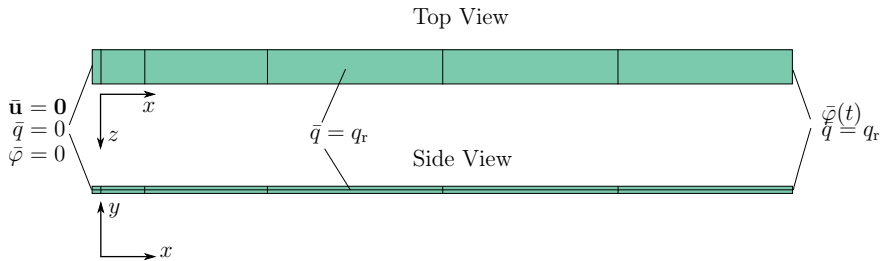
Figure 5.2: High-order mesh and boundary conditions of the bimetallic beam.

Table 5.2: Material parameters of steel and silver

Material parameter	Symbol	Steel	Silver	Unit
Density	ρ_0	7800	10500	Kg/m^3
Heat capacity	c_p	460	230	J/Kg K
Conductivity	λ_Θ	45	430	N/s K
Electric conductivity at Θ_0	$\lambda_{\varphi,0}$	12.0×10^6	62.0×10^6	A/V m
Linear temperature coef.	α_φ	5.60×10^{-3}	3.80×10^{-3}	$1/\text{K}$
Thermal expansion coef.	α_Θ	1.55×10^{-5}	1.95×10^{-5}	$1/\text{K}$
Emissivity	ε	0.8	0.1	-
Bulk modulus	K	1.642×10^{11}	1.061×10^{11}	N/m^2
Shear modulus	μ	0.802×10^{11}	0.303×10^{11}	N/m^2

5.1.2 Partitioned coupling strategy

In order to find an appropriate coupling strategy for the three-field problem, the coupling between the fields involved shall briefly be recapitulated. For the mechanical and the thermal field, the same coupling appears as in the example of Section 4.5.2. Besides, the interactions with the electric field shall be mentioned. It is coupled to the thermal field by means of the Joule heating term in the heat conduction equation. Due to the temperature-dependency of the electric conductivity, temperature changes also affect the electric field. Further, the electric field is coupled to the mechanical field due to large deformations. Based on these assumptions, the coupling strategy is developed. A fully implicit approach is chosen, starting with the electric field, followed by the thermal field and concluding with the mechanical field. For this example, the electric field can be viewed as the driving field as it is responsible for inducing energy into the beam. Iterating between the fields is carried out until all fields are balanced in the sense that the iteration tolerance $\epsilon_z = 10^{-4}$, $z \in \{\text{M}, \Theta, \varphi\}$ is satisfied. Acceleration schemes are employed to improve the convergence. The entire coupling strategy is outlined in Figure 5.3. In this figure, the acceleration scheme is applied at the end, after the mechanical field. This is, however, only one possibility. It can also be

used to accelerate the thermal or the electric field. Since there is no general state-
ment about the *best* point of application at hand, this needs to be figured out in
a convergence study. For the first iteration, the mechanical solution is predicted
by means of vector prediction methods so that the electric and thermal field are
solved on an estimated current configuration.
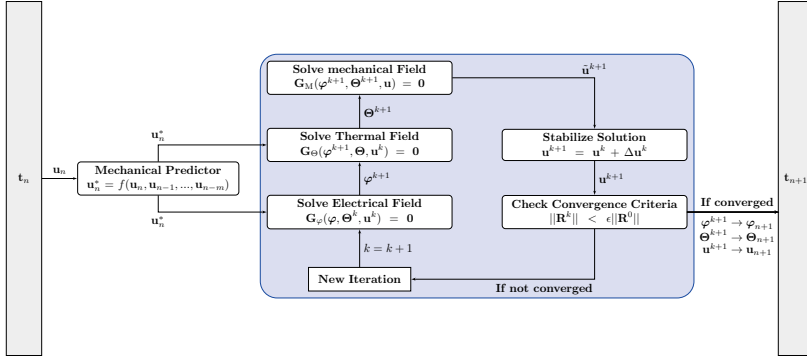
Figure 5.3: Partitioned coupling strategy for the electro-thermo-mechanical problem.

5.1.3   Results

In the following, the temperature evolution and the deflection are taken into
consideration. The deflection is defined as the maximum displacement u_y in y-
direction, occurring at the free end ($z = l$) of the beam. Both the temperature and
the displacement are taken from the result node as depicted in Figure 5.4.
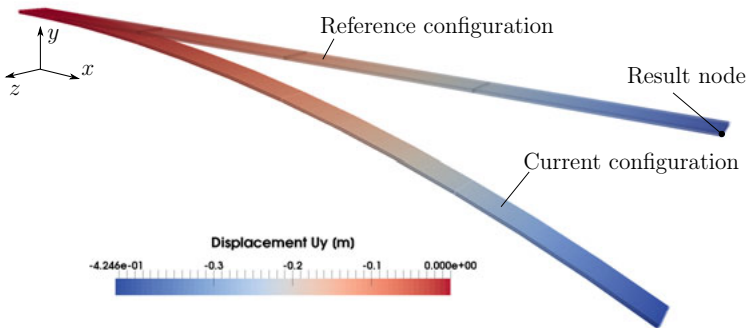
Figure 5.4: Displacement in y-direction of the bimetallic beam at $t = 100$ s.

The results at this node are given in Figure 5.5(a) for the temperature evolution and in Figure 5.5(b) for the deflection. Obviously, the electric current leads to high temperatures which therefore lead to large deflections of the beam. After $t = 100$ s, one obtains $\Theta_{\max} = 354.27\,^\circ$C and $u_y = -0.425$ m which is clearly a large deformation, as illustrated in Figure 5.4.

(a) Temperature evolution.   (b) Displacment in y-direction.
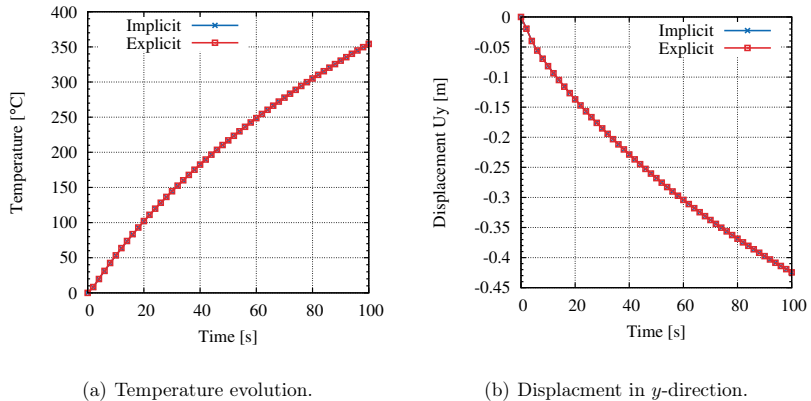
Figure 5.5: Temperature and displacement plotted against time at the result node for an implicit and explicit coupling strategy.

The further objectives are to study different coupling strategies and to investigate the influence of convergence acceleration methods. At the beginning, a fully explicit coupling strategy is taken under consideration. In this context, fully explicit means that no coupling iterations are performed between the fields. Hence, each solver is called only once per time-step. The results are given in Figure 5.5(a) with respect to the temperature evolution and in Figure 5.5(b) for the displacement in y-direction. As depicted in both Figures, the differences between an implicit and an explicit scheme are quite small and can hardly be detected. The implicit case requires an average number of 2.18 iterations to achieve balance between the three fields. This means that every solver is called 2.18 times per time-step, which leads to the fact that the computational costs for the explicit scheme are approximately half the amount of the implicit scheme. For this special example, one can conclude that the implicit solver calls are dispensable since they are neither required to attain a sufficient accuracy nor are they needed to stabilize the coupling process.

In the next stage, the coupling strength is increased successively. This can be accomplished in a straightforward manner by multiplying the thermal expansion coefficients α_Θ as well as the linear temperature coefficients α_φ with a constant factor $\gamma \in \{2, 4, 6\}$. The objective is to study the performance of the convergence acceleration schemes. In particular, the Dynamic Secant Relaxation (DSR) and

the Quasi-Newton least square method (QN) are applied to the coupled problem. The number of required coupling iterations per time-step ς is given in Table 5.3 and compared to the reference case (Ref) where no acceleration method is employed. This averaging parameter is computed after the fifteenth load-step – or, in other words, after $T = 30$ s, since $\Delta t = 2$ s is chosen for the time increment.

Table 5.3: Convergence study of the bimetallic beam and comparison of iterations required per time-step (ς) for different acceleration methods (NC = no convergence).

γ	Ref	QN(T)	DSR(T)	QN(M)	DSR(M)
2	4.07	3.93	3.87	4.0	3.93
4	9.93	5.13	3.93	5.8	5.47
6	NC	6.13	4.61	NC	NC

As shown in Table 5.3, the QN method and the DSR method serve to increase the stability and accelerate the iteration process. There are only small improvements for the case that $\gamma = 2$ holds, but for $\gamma = 4$ and $\gamma = 6$, the positive influence of the acceleration methods is quite obvious. For the case with the highest degree of coupling ($\gamma = 6$), a converged solution cannot be achieved without these methods. Further, it turns out that the point of application, either after the thermal (T) or after the mechanical field (M), is important. Apparently, the better option is to apply it after the thermal field, as this is the only way to achieve convergence in the case of $\gamma = 6$. In order to investigate this in a more detailed way, the cou-
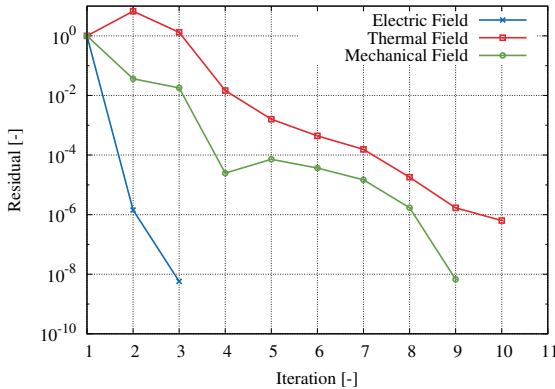
Figure 5.6: Coupling iteration residuals Λ^k of the electric, thermal and mechanical fields for $\gamma = 6$ using DSR(T).

pling iteration residual $\Lambda^k = ||\mathbf{r}^k||$ is plotted against the number of Gauss-Seidel coupling iterations in Figure 5.6. This is done for $\gamma = 6$, and the DSR method is

applied to the thermal field. It seems that the electric field converges very fast and can therefore been viewed as loosely coupled. The mechanical and the thermal field however require more Gauss-Seidel coupling iterations. To reach the tolerance of $\epsilon = 10^{-4}$, the thermal field requires 8 global coupling iterations whereas the mechanical field requires only 4. This shows how important it is to stabilize the crucial field during the simulation – and it also indicates why a converged solution cannot be obtained when applying the acceleration methods to the mechanical field. Further, one can conclude that it is very important to check if all fields are balanced. If only the convergence criterion of the electric field were to be checked, two coupling iterations would suffice. However, this would lead to at least very inaccurate results as the other fields would not be in equilibrium.

5.2 Multi-field problem with thermal radiation

In this example, thermal radiation is considered as an additional field to simulate the interaction of a radiating body with its environment. Again, the bimetallic beam from the previous example is taken under investigation. The coupling strategy now requires an extension to treat four coupled fields, as proposed in [168]. Further, different methods to solve the radiative transfer equation (RTE) – the view factor method (VFM), see Section 3.4.1, the methods of spherical harmonics (P_1-approximation), see Section 3.4.2, and the discrete ordinate method (fvDOM) described in Section 3.4.3 – are studied, and their impact on the coupling strategy is investigated.

5.2.1 Model problem: bimetallic beam coupled with radiation field

In the following, the model problem as given in Section 5.1.1 is extended as follows: The beam is placed inside a chamber and is fully enclosed by the walls of the chamber, as depicted in Figure 5.7. When the beams starts to heat up, the occurring surface radiation leads to incoming radiation at the inner chamber walls. This incoming radiation can be reflected and will then irradiate into the beam over its surface. In order to cover these effects, a thermal radiation solver is used to simulate the interaction of the beam surface with the environment. The space between the beam and the chamber walls is discretized by means of $50 \times 40 \times 30 = 60000$ finite volume cells, see Figure 5.7. Here, only one half of the chamber is shown. The dimensions of the chamber are given in Table 5.4. It is presumed that the space between the beam and the chamber walls is filled with a participating medium, an ideal gas that is homogeneous and that features a constant emission and absorption coefficient of $\alpha = 0.5$. The surfaces of the inner chamber walls are adopted as ideal black, i.e. the emissivity is $\varepsilon = 1.0$. In this example, the influence of convective heat transfer is neglected.
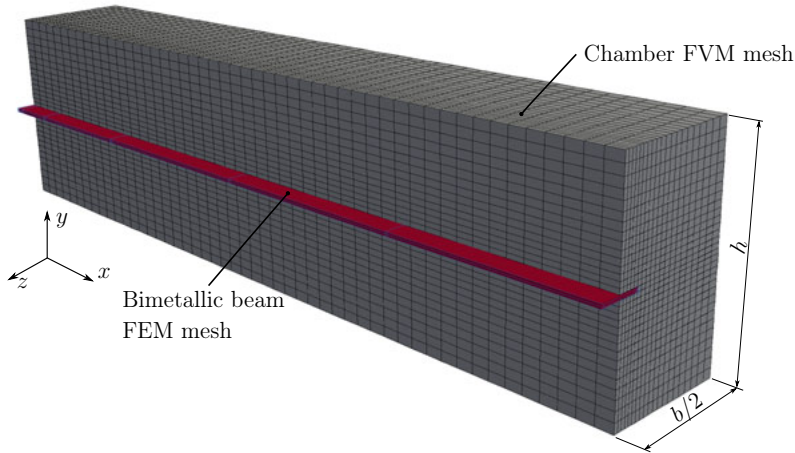
Figure 5.7: One half of the finite volume mesh (gray) of the chamber together with finite element mesh (red) of the bimetallic beam.

Table 5.4: Geometrical set-up of the chamber.

Length	l	2	m
Width	b	0.51	m
High	h	0.6	m

5.2.2 Partitioned coupling strategy

The coupling effects between the electro-thermo-mechanical problem were discussed in Section 5.1.2. However, the presence of the radiation field needs to be taken into account. First of all, it is coupled with the thermal field. As outlined in Section 2.3.4, the temperature and the heat flux must be at equilibrium on the coupling domain – which, in this case, is the surface boundary of the bimetallic beam. Moreover, as large displacements are assumed, the radiation field has to be computed on the updated mesh. Thus, it is coupled with the mechanical field through the displacements of the coupling domain. In both cases, a surface coupling arises so that the whole problem can be seen as a surface- and volume-coupled system.

Due to the surface coupling, data need to be transferred over interfaces with non-matching meshes. The bimetallic beam is discretized with a very coarse high-order finite element mesh, whereas the radiation field uses finer finite volume cells. To this end, the data are interpolated before being transferred. In addition, the finite volume mesh needs to be updated dynamically with respect to the current deformation. Several possibilities to tackle these problems were discussed in Section 4.3. In this example, the barycentric triangle interpolation is utilized and

a Laplacian smoother is used to take mesh motion into account.

The three-field strategy proposed in the previous example is now extended to a four-field problem. Again, a fully implicit strategy is employed so that balance is achieved if all fields are converged. The general strategy is shown in Figure 5.8. Thermal radiation is chosen to be the last field in the partitioned sequence. The reason for this is that the radiation field shall be solved on the current updated configuration. The point of application of the acceleration methods is chosen after the thermal field, that means that the temperature is corrected by means of the dynamic secant relaxation (DSR) method or the quasi-Newton (QN) method.
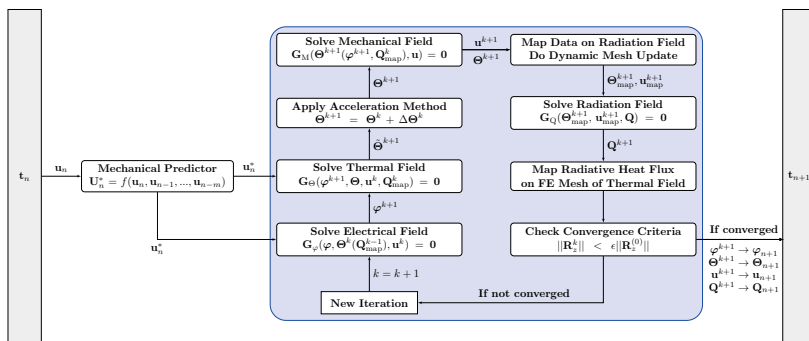
Figure 5.8: Partitioned coupling strategy for a coupled four-field problem including a radiation field.

5.2.3 Results

First of all, the influence of mesh deformation is investigated. Due to the kinematic coupling condition, the mesh of the surroundings needs to be updated dynamically in every iteration. As depicted in Figure 5.9, two different situations are possible: The chamber walls can be fixed, as shown in Figure 5.9(a), allowing only little deflection in comparison to the case where the chamber walls are not constraint, see Figure 5.9(b). For the former situation, at a certain time, some of the finite volume cells becomes so distorted that the computation is aborted. Henceforth, the example using flexible chamber walls is taken under consideration to allow larger beam deflections. Due to the large displacements, it is assumed that the coupling strength might be higher and a partitioned solution is therefore numerically more challenging. However, there is no physical reason to use flexible chamber walls; in this study, they are used to test the four-field coupling approach.

(a) Deformed mesh with fixed wall boundary.

(b) Deformed mesh without constraining the wall boundary.
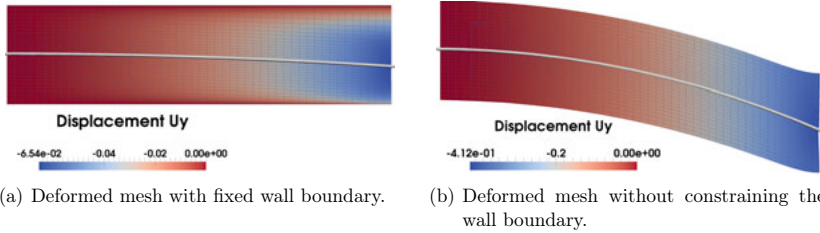
Figure 5.9: Mesh deformation for different boundary conditions of the chamber walls.

(a) Temperature evolution.

(b) Displacement in y-direction.
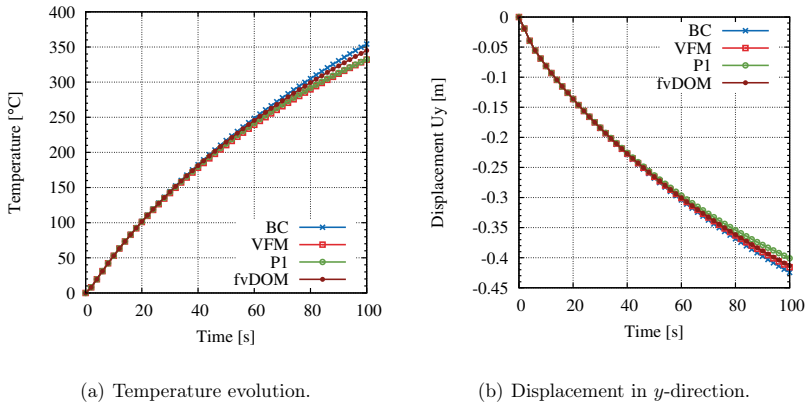
Figure 5.10: Temperature and displacement plotted against time at the result node for different thermal radiation models.

The temperature development and the displacement in y-direction are compared to the results from the previous section where the influence of the chamber was not simulated. The results for the three different radiation models – the VFM, the P_1-method and the fvDOM– are depicted in Figure 5.10. The variable of interest is the temperature, and the temporal course is given in Figure 5.10(a). It turns out that, with increasing temperatures, the differences between the simple boundary model and the more complex radiation models increase as well. However, these are only slight deviations, the maximum difference amounts to only 6% between the P_1 and the standard boundary model at the end of the simulation. The reason for this is that there are no complex geometries and that an approximation of radiation via the boundary condition thus delivers acceptable results.

Similar to the previous section, the aim is to study the performance of acceleration methods. To this end, the thermal expansion coefficient is once more multiplied with the factor $\gamma \in \{\, 2,\, 4,\, 6\,\}$. The results are given in Table 5.5, in which

the averaged coupling iterations per time-step for different radiation models are compared. In this study, the convergence acceleration methods are applied to the thermal field. The results clearly reveal a need for additional stabilization schemes to achieve convergence for strong coupling levels ($\gamma = 6$). Without the DSR or the QN, a converged solution cannot be accomplished. Further, it turns out that the number of iterations varies with the radiation model. For example, the combination of $\gamma = 2$ and the fvDOM seems to require the fewest number of iterations. However, the differences between the radiation models are small, also for higher coupling strengths. It is demonstrated that both the DSR and the QN methods allow to accelerate and to stabilize the four-field problem, independently of which radiation model is used. A comparison between the QN and the DSR shows that the latter method performs slightly better. In the case of $\gamma = 6$, the QN requires up to $\varsigma = 2.2$ iterations per time-step more (P_1-method) than the DSR.

Table 5.5: Convergence study on the example of the bimetallic beam for different radiation models: comparison of the average number of coupling iterations.

	VFM			P_1**-method**			**fvDOM**		
γ	Ref	DSR	QN	Ref	DSR	QN	Ref	DSR	QN
2	4.80	4.47	4.53	4.47	4.20	4.47	3.67	3.20	3.20
4	11.73	5.87	5.87	12.07	5.93	6.87	NC	6.15	6.67
6	NC	7.25	9.35	NC	6.67	8.89	NC	7.55	8.78

In order to get a better insight of the convergence behavior of the radiation field, the coupling iteration residual is plotted against the number of iterations in Figure 5.11(a) for the DSR scheme and in Figure 5.11(b) for the QN method. In these figures, the convergence in the first time-step is considered at high coupling strengths ($\gamma = 6$). The results show that the VFM and the P_1-method exhibit a very similar convergence behavior, whereas the fvDOM performs slightly worse. To achieve an accuracy in the coupling iteration residual of $\epsilon < 10^{-4}$, both acceleration schemes require 8 Gauss-Seidel iterations when applying the VFM or P_1-method – and 9 in the case of the fvDOM.

Another interesting aspect has to do with how different radiation models influence the total computation time. In order to make the comparison as fair as possible, the example with the coupling strength of $\gamma = 4$ is considered. In the case where the DSR method is applied, all radiation models require approximately six iterations to reach convergence. Figure 5.12(a) lists the results, with the normalized computation time plotted against the time. As mentioned in Section 3.4, the fvDOM is a very cost-intensive method. This is also reflected in the example. At the end of the simulation ($t = 30$s) the total computation time is approximately 77% higher compared to the P_1-method and even 85% higher compared to the VFM. A comparison to the three-field problem, which is based on the simple boundary radiation model, is difficult – due to the fact that it reaches a converged solution in approximately four coupling iterations, see Table 5.3, whereas
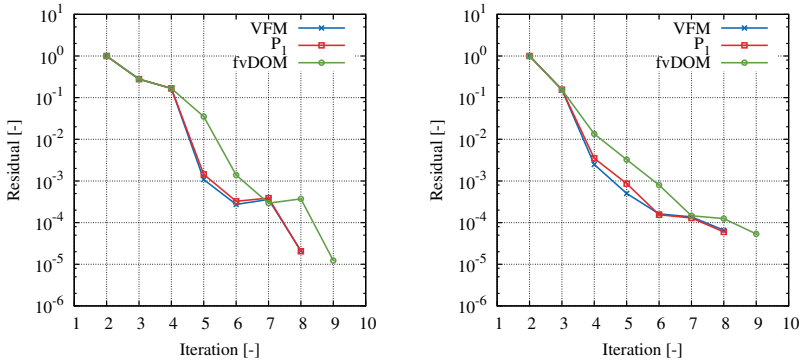
(a) Application of the DSR to the thermal field.   (b) Application of the QN to the thermal field.

Figure 5.11: Coupling residual of the radiation field plotted against the number of iterations for different radiation models at high coupling strengths ($\gamma = 6$).

the other radiation models require about six. For this reason, the computation of this example is carried out once again, this time ensuring that the coupling algorithm also requires six iterations per time-step. This can be done by setting a very sharp iteration tolerance and terminating after six iterations, allowing for an almost fair comparison. Figure 5.12(b) depicts the relative overall computation time related to the most time-consuming method, the fvDOM. Not surprisingly, the boundary radiation model is the fastest way to incorporate radiation. For this example, the computation time compared to the fvDOM takes only 8%.
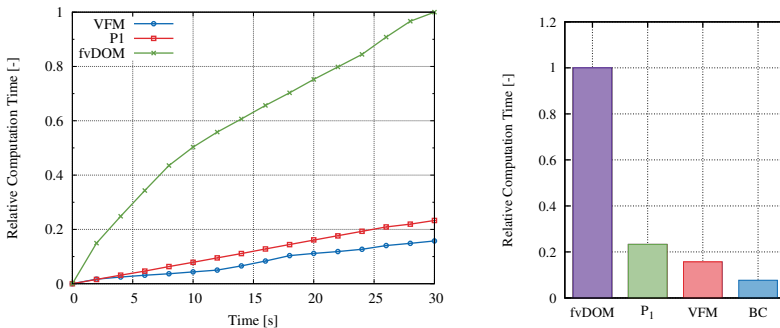
(a) Relative computation time over the total time.

(b) Comparison of radiation models to BC-model.

Figure 5.12: Relative computation time for different radiation models.

5.3 Simulating the FAST process

In this example, a partitioned coupling strategy for the simulation of the field assisted sintering technology (FAST) is proposed. The process is governed by an electric, thermal and mechanical field plus a radiation field to tackle the thermal conditions of the surroundings. Two different simulations are carried out: In the first step, the temperature evolution inside the FAST tools and the chamber is computed without taking the powder compaction process into account. In the second simulation, the powder material is added and the results are compared to monolithic simulations and to experimental data from [146].

5.3.1 Temperature evolution

The first simulation serves to determine the temperature evolution in the FAST tools for fully consolidated copper powder where the relative density is approximately $\rho_{\mathrm{rel}} = 1$. This means that the compaction process itself is not studied. Nevertheless, to gain information of how the graphite tool-system influences the final temperature, the stress distribution of the powder and the material parameters of the sintering process, such an investigation is of principle interest. HARTMANN ET AL. [76] studied the punch/die system and applied a monolithic coupling approach to solve the electro-thermo-mechanically coupled problem. However, the influence of the chamber in which the graphite tool-system is located was disregarded. The objective of the following investigation is to take the complex interactions of the radiating surfaces with the surroundings and the chamber walls into account. To this end, an additional radiation field is introduced. This consequently means that a simulation of the FAST process now requires the solution of four coupled fields.

5.3.1.1 Constitutive relations

Referring to HARTMANN ET AL. [76], the same constitutive model and material parameters are utilized. Concerning the kinematic relations, the strain tensor is assumed to be an additive decomposition of a mechanical and a thermal part

$$\underset{\sim}{E} = \underset{\sim}{E}_{\mathrm{M}} + \underset{\sim}{E}_{\Theta} \,, \tag{5.2}$$

where the latter contribution is purely volumetric $\underset{\sim}{E}_{\Theta} = \alpha_{\Theta} \left(\Theta - \Theta_0 \right) \underset{\sim}{I}$. Further, the free-energy function $\Psi(\underset{\sim}{E}_{\mathrm{M}}, \Theta)$ consists of a mechanical and a thermal part

$$\Psi(\underset{\sim}{E}_{\mathrm{M}}, \Theta) = \Psi_{\mathrm{M}}(\underset{\sim}{E}_{\mathrm{M}}) + \Psi_{\Theta}(\Theta). \tag{5.3}$$

HARTMANN ET AL. [76] assumed a small strain thermoelasticity material model and used

$$\rho \, \Psi_{\mathrm{M}}(\underset{\sim}{E}_{\mathrm{M}}) = \frac{\lambda}{2} \left(\mathrm{tr} \, \underset{\sim}{E}_{\mathrm{M}} \right)^2 + \mu \, \underset{\sim}{E}_{\mathrm{M}} \cdot \underset{\sim}{E}_{\mathrm{M}} \tag{5.4}$$

as the mechanical part of the specific free-energy function. Based on this function, the expressions for the stress tensor and the entropy can be calculated to

$$\underline{\sigma} = \rho \frac{\partial \Psi_{\mathrm{M}}}{\partial \underline{E}_{\mathrm{M}}} = \lambda \operatorname{tr} \underline{E}\, \underline{I} + 2\,\mu\, \underline{E} - \alpha_\Theta\, \Delta\Theta (3\lambda + 2\mu)\, \underline{I}, \tag{5.5}$$

$$\xi = -\frac{\partial \Psi}{\partial \Theta} = -\frac{\partial \Psi_\Theta}{\partial \Theta} + \frac{\alpha_\Theta(3\lambda + 2\mu)}{\rho} \left(\operatorname{tr} \underline{E} - 3\alpha_\Theta\, \Delta\Theta \right). \tag{5.6}$$

Following [76], the thermo-elastic coupling term results to

$$\gamma(\dot{\underline{E}}, \Theta) = \alpha_\Theta\, (3\lambda + 2\mu)\, \Theta\, \operatorname{tr} \dot{\underline{E}}. \tag{5.7}$$

Further reading on modeling small strain thermoelasticity is provided in [76]. The thermal field is considered drawing on the equation of heat conduction, which features nonlinear dependencies of the material parameters. Further, the electric field is once again modeled by the conservation of charge with the assumption of a stationary electric current.

5.3.1.2 Model description

In Figure 5.13, the geometry and the dimensions of the graphite tool-system are depicted on a rotation-symmetric drawing in the x-y-plane. The specifications of this machine tool are taken from [76]. As only the temperature evolution inside the graphite-powder system is investigated, the powder, die, punch, and cone part are assumed as a contiguous region.
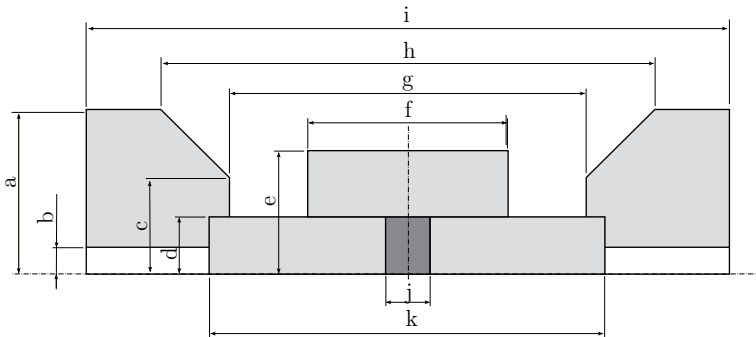
Figure 5.13: Rotation-symmetric drawing of the graphite tool-system and dimensions, rotated counterclockwise by 90°.

The finite element mesh of the graphite system is depicted in Figure 5.14(a). Due to the symmetry, only one-eighth of the geometry is taken under consideration. As shown in Figure 5.14(a), it is distinguished between the graphite and

a	b	c	d	e	f	g	h	i	j	k
27.5	5.0	15.5	10.4	25.0	48.0	73.6	115.1	135.1	10.0	79.7

the powder domain. The mesh consists of 384 high-order hexahedral elements with a polynomial degree of $p = 4$. All finite element computations are carried out with the high-order finite element code AdhoC. The FAST tools are located inside a chamber which has the simple geometry of a box with the dimensions $110 \times 135.1 \times 110$ mm^3. The space between the punch/die system and the chamber walls defines the radiation field. This field enables radiative heat transfer between the radiating surfaces of the graphite tools and the surroundings. For the spatial discretization, a finite volume mesh consisting of 13171 finite volume cells is used. As depicted in Figure 5.14(b), only one-eighth of the chamber volume is discretized due to symmetry conditions. This finite volume mesh is employed when applying the fvDOM and the P_1-method. For the VFM, only the discretized surfaces of the chamber walls and the graphite tools are required. The surface mesh is extracted from the finite volume mesh in Figure 5.14(b). Since there are no symmetry conditions available for the VFM in OpenFOAM, the whole model needs to be discretized for the radiation field. The resulting finite volume mesh consists of 88344 volumes.[1]

(a) High-order finite element mesh of the graphite tools.

(b) Finite volume mesh of the space between the tools and the chamber walls.
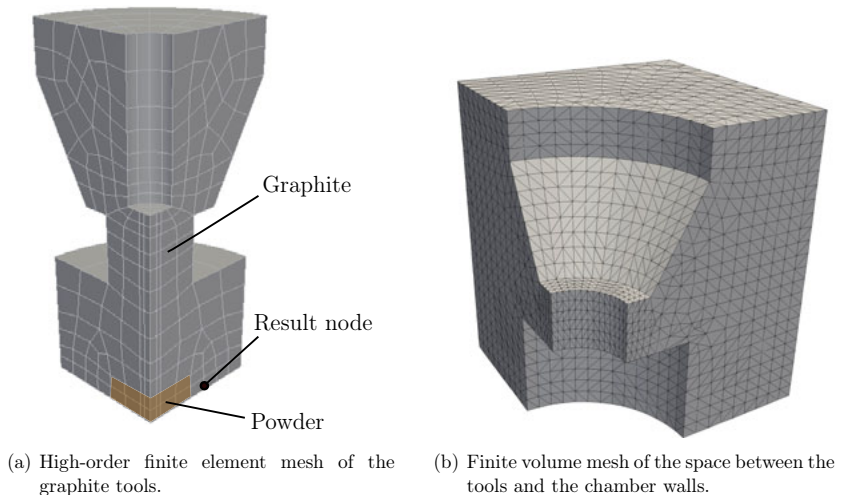
Figure 5.14: Used meshes for simulating the FAST process.

---

[1]The finite element mesh is not affected when using the VFM. In this case, the surface temperatures are simply mirrored before they are transferred to the radiation field.

The boundary conditions of the mechanical, the thermal and the electric field are presented in Figure 5.16. Regarding the thermal field, water-cooling of the graphite tools is modeled by forced convection heat flux at the top of the cone part. The rising flow can be described by $\bar{q}_c = h(\Theta - \Theta_w)$, where $h = 88$ W/m²K denotes the heat transfer coefficient and $\Theta_w = 295.15$ K is the constant water temperature. In Figure 5.16, the radiating surfaces are colored turquoise. This surface is either the coupling interface to the radiation field over which the radiative heat flux and temperature are exchanged – or, in the case of the boundary model, thermal radiation is approximated as $\bar{q}_r = \varepsilon \, \sigma_{sb}(\Theta^4 - \Theta_\infty^4)$. Here, an ambient temperature $\Theta_\infty = 303.15$ K is assumed and the surface emission coefficient is $\varepsilon = 0.8$. For the electrical field, a time-dependent electric current is applied, providing a heat source based on Joule heating. This was determined experimentally [76], and the temporal course is shown in Figure 5.15.
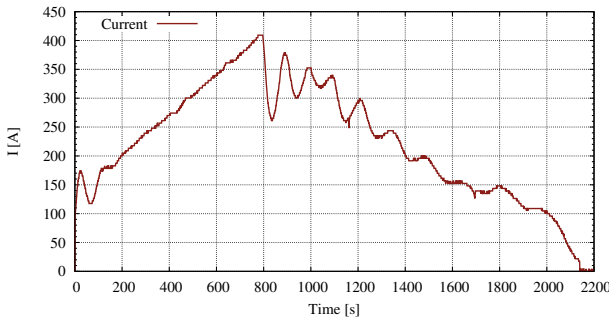
Figure 5.15: Experimentally determined electric current applied to the graphite tools, taken from [76].

The boundary conditions for the radiation field are depicted in Figure 5.17. The coupling interface (red) defines the coupling domain between the radiation and the thermal field. At the chamber walls, the velocity of the fluid, which is the rarefied gas that approaches vacuum conditions, is prescribed to be zero, and the wall temperature coincides with the ambient temperature Θ_∞.

The material properties for the consolidated copper powder and for graphite are summarized in Table 5.7. For realistic simulations, the temperature dependency of several material parameters for both materials was experimentally determined by HARTMANN ET AL. [76]. In the following, the thermal and electric conductivity (λ_Θ, λ_φ) and the heat capacity (c_Θ) are described by temperature-dependent functions. Following [76], the parameters are approximated by ana-
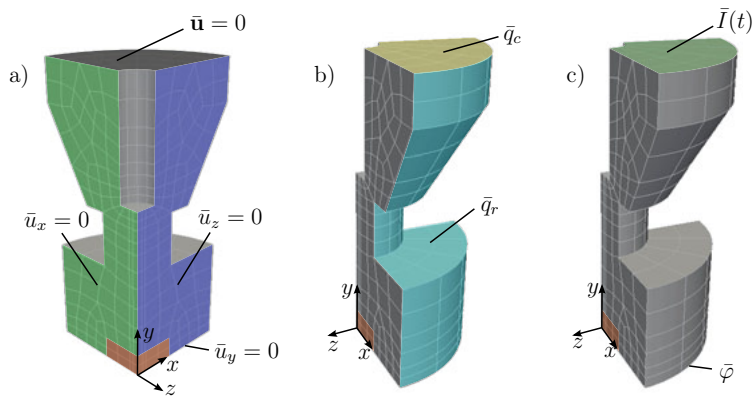
Figure 5.16: Boundary conditions of the graphite tools system for the a) mechanical, b) the thermal and c) the electric field.
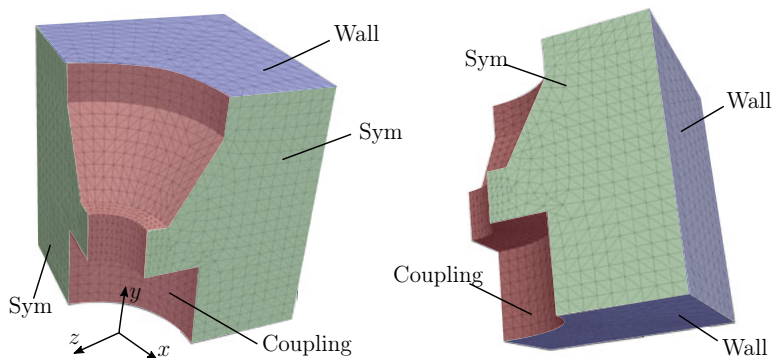

Figure 5.17: Boundary and interface conditions of the space between tools and chamber walls.

lytic ansatz functions

$$\lambda_\Theta(\Theta) = b_1 \, e^{-b_2\Theta} + b_3 \, e^{-b_4\Theta}, \tag{5.8a}$$

$$\lambda_\varphi(\Theta) = c_1 \, e^{-c_2\Theta} - c_3 \, e^{-c_4\Theta}, \tag{5.8b}$$

$$c_\Theta(\Theta) = d_1 + d_2\Theta + d_3 \, \tanh(d_4\Theta - d_5) \tag{5.8c}$$

for graphite and, with the help of a linear ansatz functions, for copper powder

$$\lambda_{\Theta,Cu}(\Theta) = e_1 \, \Theta + e_2, \tag{5.9a}$$

$$\lambda_{\varphi,Cu}(\Theta) = f_1 \, \Theta + f_2, \tag{5.9b}$$

$$c_{\Theta,Cu}(\Theta) = g_1 \, \Theta + g_2. \tag{5.9c}$$

The coefficients needed in Eqns. (5.8a) – (5.9c) are depicted in Tab. 5.8 and Tab. 5.9.

Table 5.7: Material properties for graphite and copper.

Material parameter	Symbol	Graphite	Copper	Unit
Young's modulus	E	$1.15\times10^{+10}$	$1.2\times10^{+11}$	N/m^2
Poisson ratio	ν	0.2	0.3	-
Density	ρ	$1.85\times10^{+03}$	$8.92\times10^{+03}$	kg/m^3
Linear temp. coef.	α_Θ	4.55×10^{-06}	1.60×10^{-05}	1/K
Thermal conductivity	λ_Θ	(5.8a)	(5.9a)	W/mK
Electric conductivity	λ_φ	(5.8b)	(5.9b)	S/m
Heat capacity	c_Θ	(5.8c)	(5.9c)	m^2/s^{2}K

Table 5.8: Coefficients of the temperature-dependent material properties (1/2).

	$i = 1$		$i = 2$		$i = 3$	
b_i	$8.5700 \cdot 10^{+1}$	$\frac{W}{mK}$	$5.124 \cdot 10^{-3}$	$\frac{1}{K}$	$8.81 \cdot 10^{+1}$	$\frac{W}{mK}$
c_i	$1.6900 \cdot 10^{+5}$	$\frac{S}{m}$	$2.168 \cdot 10^{-4}$	$\frac{1}{K}$	$1.66 \cdot 10^{+5}$	$\frac{S}{m}$
d_i	$5.8650 \cdot 10^{+2}$	$\frac{J}{KgK}$	$3.507 \cdot 10^{-1}$	$\frac{J}{KgK^2}$	$8.39 \cdot 10^{+2}$	$\frac{J}{KgK}$
e_1	$-7.8335 \cdot 10^{-2}$	$\frac{W}{mK^2}$	$4.332 \cdot 10^{+2}$	$\frac{W}{mK}$		
f_i	$3.800 \cdot 10^{+7}$	$\frac{S}{mK}$	$5.5000 \cdot 10^{+9}$	$\frac{S}{m}$		
g_i	$8.2214 \cdot 10^{-2}$	$\frac{J}{KgK^2}$	$3.737 \cdot 10^{+2}$	$\frac{J}{KgK}$		

Table 5.9: Coefficients of the temperature-dependent material properties (2/2).

	$i = 4$		$i = 5$	
b_i	$3.292 \cdot 10^{-4}$	$\frac{1}{K}$		
c_i	$2.089 \cdot 10^{-3}$	$\frac{1}{K}$		
d_i	$3.247 \cdot 10^{-3}$	$\frac{1}{K}$	0.9431	[-]

The simulation is carried out with a constant time increment of $\Delta t = 2\,$s, and the time period under consideration is set to $T = 2200\,$s. A partitioned coupling strategy is applied, similar to the one used in the previous section, see Figure 5.8. However, the radiation field and the mechanical field are assumed to be decoupled. Following [76], the fully coupled electro-thermo-mechanical problem incorporates thermoelasticity at small strains. As this does not lead to substantial geometric changes, the radiation field can be solved on a fixed configuration. Further, this implies that the view factors are constant during the computation and, therefore, the computational costs for the VFM can be reduced drastically.[2] Besides,

---

[2]This is due to the fact that a QR-decomposition of the **D** matrix of Eq. (3.109) needs to be

the electric and the thermal field are coupled against each other via temperature-dependent material parameters and via the Joule heating term. Between the mechanical and the thermal field, the coupling occurs due to thermal expansion and the thermo-elastic coupling term in the heat equation.

A further remark on the numerical treatment of the radiation field: Since air would cause oxidation of the tools under high temperatures, a vacuum is created inside the chamber by technical means. However, to apply the fvDOM or the P_1-method, a participating medium is required. To this end, a very thin gas is used as a participating fluid such that vacuum conditions are nearly reached. This can be obtained choosing a small absorption coefficient ($\alpha = 0.01$) such that the optical depth is much smaller than one and the transmittance τ approaches one.

5.3.1.3 Results

To begin with, the results under perfect vacuum conditions are presented. The temperature is compared to the experimentally and numerically determined results from [76], where the standard boundary radiation model is employed. There is no participating media, so that the VFM is applied to simulate the radiation field. In the following, three different cases are computed:

- Case a) involves the standard conditions for the chamber. The emission coefficient $\varepsilon_w = 0.8$ is assumed for oxidized steel chamber walls. Further, $\Theta_w = 305.15$ K is taken for the constant chamber wall temperature.

- In case b), the emission coefficient is changed to $\varepsilon_w = 0.2$, which approaches blank steel, whereas the wall temperature remains unchanged.

- Case c) also uses $\varepsilon_w = 0.2$ and prescribes a time-dependent wall temperature. The chamber walls are water-cooled, as usual, which is why the wall temperature is not constant. As there is no experimental data for the temperature, it is interpolated as

$$\Theta_w = 0.5\,[\Theta_s(t) + 305.15\,\text{K}]\,, \tag{5.10}$$

where Θ_s is the maximum temperature on the surface of the graphite tools.

The temperature evolution at the result node for the three different cases is depicted in Figure 5.18, together with the experimental and numerical data from [76]. It turns out that the computed temperature is considerably higher when using the VFM, and it approaches the experimentally determined temperature with a better accuracy. The reason for this is that the VFM allows to capture the influence of the complex geometry and, further, it allows to simulate effects such as reflections and irradiation between the tool surfaces and the chamber walls. Furthermore, the influence of the emission coefficient and the temperature for the chamber walls was studied. It turned out that the third case – the one using $\varepsilon_w = 0.2$ and

---

done only once at the beginning of the simulation. It can then be reused in every iteration and in every time-step, so the unknown heat fluxes can simply be obtained by backward substitution.

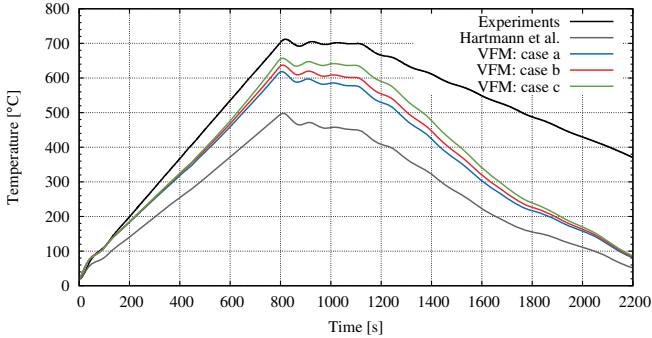a variable wall temperature – delivers the most suitable approach to the experiments.

Figure 5.18: Temperature development at the evaluation point P. Comparison of the results obtained with the VFM to experimental and simulation data given in [76]
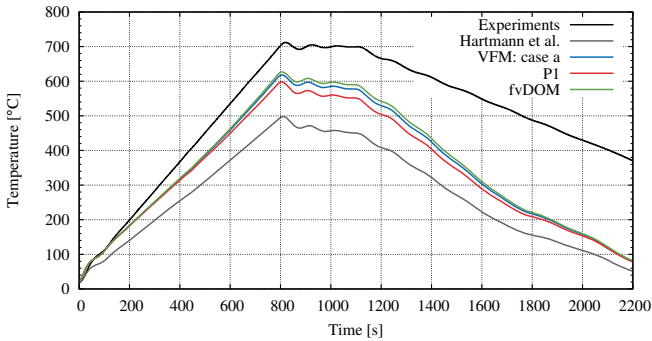

Figure 5.19: Temperature development at the evaluation point P. Comparison of the results obtained with the P_1-method and the fvDOM to experimental and simulation data given in [76]

Next, the vacuum chamber is approached by an optically very thin medium which allows to apply the fvDOM and the P_1-method to the radiation field. Again, the temperature evolution at the result point is considered, and the results for

the different radiation models are shown in Figure 5.19. The fvDOM and the P_1-method lead to reasonable temperatures too, whereas the former is slightly above the VFM (case a). The P_1-method slightly underestimates the temperature, which is due to the known fact that it is not well-suited for optically thin media.

However, all radiation models deliver promising results, thus constituting better approaches to the experiments than the boundary radiation model. Unfortunately, there are still deviations compared to the experimental data, in particular for the cooling period, which starts almost in the second half of the simulation. In order to reduce the remaining deviations and to improve the numerical results, knowledge about the surface temperature of the chamber walls is required. In addition, it was mentioned by HARTMANN ET AL. [76] that such deviations might be due to an inaccurate modeling of the water-cooling convection at the upper surface. Further, they suppose that there is a strong influence of the electric and thermal contact conditions between contact surfaces of the punch/die system and that this leads to an imperfect resistance and Joule heating.

Figure 5.20 shows a comparison between the temperature distribution at $T = 1100$ s for the VFM (case 3) and the boundary model. Moreover, the temperature values at the result point at $T = 1100$ s for the different radiation models are listed in Table 5.10. The minimal difference between the experimental data and the numerical solution is $\Delta\Theta = 62.3$ K.

	$\Theta[°C]$
Exp. [76]	699.0
BC [76]	450.5
VFM (1)	577.6
VFM (2)	601.6
VFM (3)	636.7
fvDOM	589.9
P_1	551.6
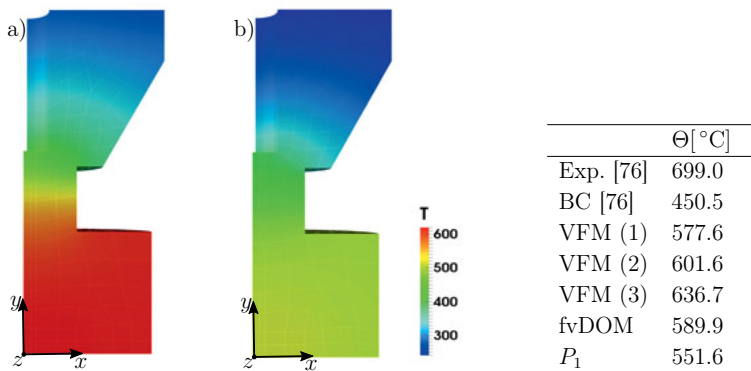
Figure 5.20: Temperature distribution in the machine tools for a) VFM (case 3) and b) the boundary model.

Table 5.10: Absolute temperature at result point at $T = 1100$ s and deviation to experimental data.

In the following, the results of the whole radiation field obtained with the VFM (case 3) will be addressed). Figure 5.21(a) depicts the temperature, and Figure 5.21(b) shows the heat flux after $T = 1100$ s. Here, it is defined that a heat flux that leaves the graphite tool-system is indicated by a negative sign. As shown in

Figure 5.21(b), there is a region at the end of the tool-system that has a positive sign, i.e. the incoming heat flux is higher than the outgoing portion. Since the ambient temperature of this region is below the surface temperature, the reason for the very high incoming heat flux must consequently lie in the strongly radiating center of the FAST tool-system. This effect is called self-irradiation and describes the radiative transfer between surfaces of the graphite tool-system. This leads thermal energy back into the punch/die system and therefore results in higher temperatures.

Taking the heat flux under consideration underlines the superiority of a Dirichlet-Neumann coupling formulation between the radiation and thermal field over the standard boundary radiation model, which is not able to capture effects like self-irradiation.

(a) Plot of the temperature distribution.   (b) Plot of the radiative heat flux.
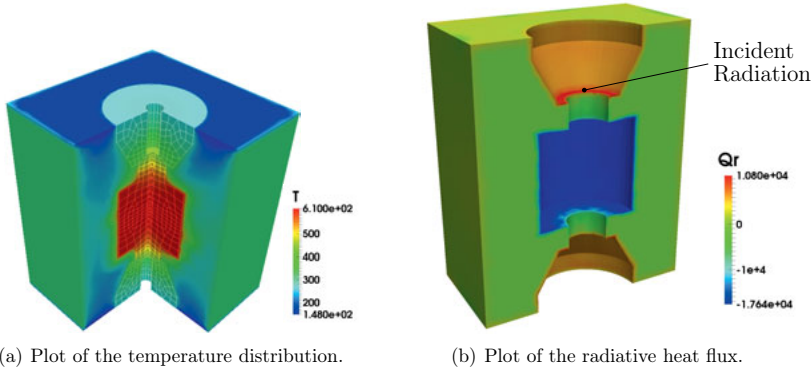
Figure 5.21: Radiative heat flux and temperature distribution after $T = 1100$ s of the VFM.

5.3.2 Consolidation of copper powder

In the second FAST example, a constitutive model for copper powder is applied for a realistic simulation of the compaction process. In the following, a thermo-viscoplastic model for highly compressible copper powder developed by ROTHE [146] is employed, covering large deformations as well as the material's temperature dependency.

The basis of the constitutive formulation is a multiplicative split of the deformation gradient into four parts

$$\underset{\sim}{F} = \underset{\sim}{F}_e \underset{\sim}{F}_\Theta \underset{\sim}{F}_p \underset{\sim}{F}_c , \tag{5.11}$$

where $\underset{\sim}{F}_e$ is a reversible elastic part, $\underset{\sim}{F}_\Theta$ a reversible thermal part, $\underset{\sim}{F}_p$ an irreversible plastic and $\underset{\sim}{F}_c$ denotes an irreversible creep part. Due to the complexity of the powder material model, a further description is not given in this work.

The basic constitutive equations are summarized in the Appendix A.4. A very detailed derivation of the constitutive equations can be found in [146].

5.3.2.1 Model description

The thermal and electric fields are solved on the same mesh with the same boundary conditions as utilized in the previous simulation, see the Figures 5.13, 5.14(a) and 5.16. The applied current is shown in Figure 5.22(a) and the applied axial force in Figure 5.22(b), both taken from [146].

Table 5.11: Coefficients for the heat capacity of graphite and copper powder from [146].

Material	c_{p0}	c_{pS}
Copper	397.83 J(kgK)	2.07×10^{-4} 1/K
Graphite	896.0 J(kgK)	1.62×10^{-3} 1/K

(a) Electric current over the time.

(b) Axial force over the time.
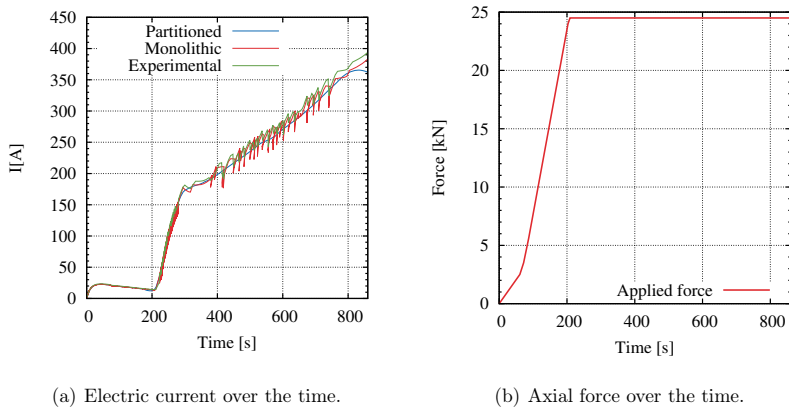
Figure 5.22: Applied electric current and axial force based on [146].

In this sintering simulation, different thermal and electric material parameters are chosen. Based on experimental data, ROTHE [146] proposed the ansatz

$$\lambda_\Theta(\Theta) = -a_\Theta \Theta + b_\Theta \tanh\left(\frac{\rho_{\mathrm{rel}} - c_\Theta}{d_\Theta}\right) + e_\Theta, \tag{5.12a}$$

$$\lambda_\varphi(\Theta) = \rho_{\mathrm{rel}}^2 (a_\varphi \Theta^{-b_\varphi} - c_\varphi), \tag{5.12b}$$

$$c_\Theta(\Theta) = c_{\mathrm{p0}} \left[1 + c_{\mathrm{pS}}(\Theta - \Theta_0)\right] \tag{5.12c}$$

for the specific heat capacity, the electric and thermal conductivity for copper powder and

$$\lambda_\Theta(\Theta) = a_\Theta \Theta^{b_\Theta}, \tag{5.13a}$$
$$\lambda_\varphi(\Theta) = a_\varphi \Theta^{b_\varphi}, \tag{5.13b}$$
$$c_\Theta(\Theta) = c_{p0} \left[1 + c_{pS}(\Theta - \Theta_0)\right]. \tag{5.13c}$$

for graphite. The coefficients based on the metric SI unit system are given in Table 5.12 for the thermal conductivity λ_Θ, in Table 5.13 for the electric conductivity λ_φ and in Table 5.11 for the heat capacity c_Θ. All coefficients are taken from [146].

Table 5.12: Coefficients for the thermal conductivity of graphite and copper powder from [146].

Material	a_Θ	b_Θ	c_Θ	d_Θ	e_Θ
Copper	0.05 W/(mK^2)	214.60 W/(mK)	0.83	0.17	259.80 W/(mK)
Graphite	1130 W/(mK)	-0.425	-	-	-

Table 5.13: Coefficients for the electric conductivity of graphite and copper powder from [146].

Material	a_φ	b_φ	c_φ
Copper	2.69×10^{10} S/m	1.08	1.21×10^6 S/m
Graphite	1.49×10^4 S/m	3.04×10^{-1}	-
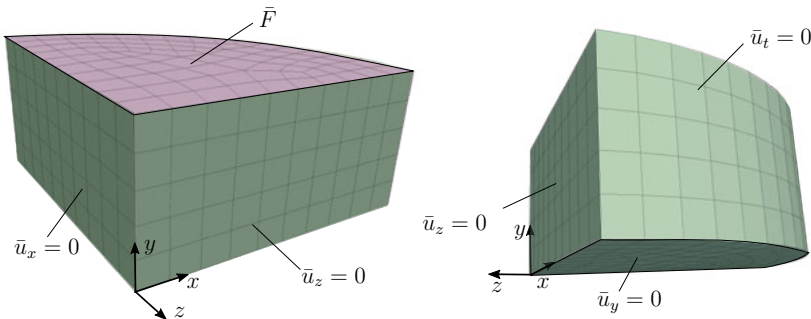
Figure 5.23: Powder mesh and boundary conditions.

As shown in Figure 5.23, only the powder region is discretized for the mechanical field. Based on the assumption that the graphite tools are rigid compared to the initial state of the powder, the force is directly applied on the top of the

powder region, see Figure 5.23. This assumption further allows to approach the contact between powder and the tools by simply constraining the displacement on the Neumann boundary of the powder domain.

In the following simulation, the results of the partitioned formulation are compared to monolithic computations and experiments carried out by ROTHE [146]. In view of the partitioned formulation, the following assumptions and simplifications are made:

- The graphite tools are assumed to be rigid and are therefore not modeled for the mechanical field. Thus, the contact conditions between the powder and the tools can be covered by constraining the displacement boundary conditions, and the load is directly applied to the powder.

- As reported in [146], there are also electric and thermal contact interfaces between the FAST tools and between the powder and the tools. These contact conditions are also not modeled in this work.

- Coupling effects resulting from geometric nonlinearity are neglected, i.e. the electric and the thermal field are solved on a fixed configuration.

In this simulation, the radiation field is not taken under investigation, and the radiating surfaces are treated with the simple boundary radiation model. The coupling strategy is based on a three-field problem, following Figure 5.3.

5.3.2.2   Results

(a) Temperature evolution at thermocouple point.

(b) Relative density during processing.
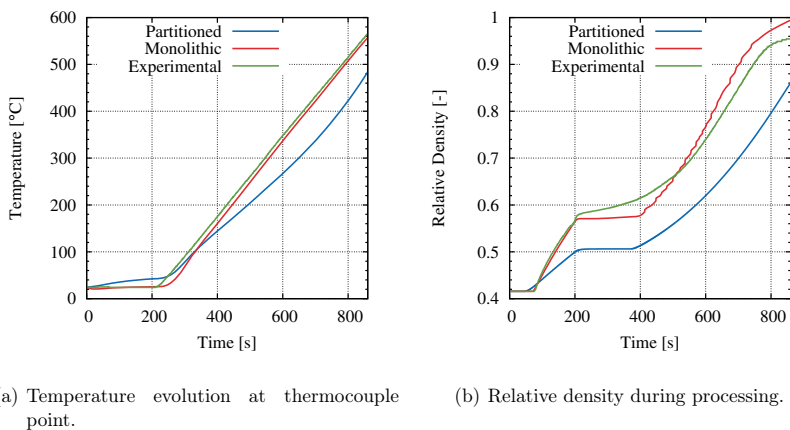
Figure 5.24: Comparison of temperature evaluation and relative density. Monolithic and experimental data is taken from [146].

The results of the sintering simulation are depicted in Figure 5.24(a), where the temperature development is shown, and in Figure 5.24(b), where the evolution of the relative density is plotted. Compared to the results obtained with the monolithic approach [146], there is a significant deviation in both courses, yet the tendency of the temperature and density evolution during processing seems to agree. At the end of the simulation after $T = 860$ s, the deviations in the temperature compared to the monolithic approach are around 15.5% and for the relative density approximately 15.2%. The reason for the discrepancies is not so much a problem of the partitioned coupling algorithm, but is rather due to the assumptions and simplifications made for this computation, particularly because of neglecting the mechanical, thermal and electric contact conditions. Since all fields depend on each other, it is difficult to determine the source of the deviation clearly. For example, a lower relative density also has influence on the thermal and electric conductivity, which leads to lower temperatures in the graphite tool-system and also in the powder. Presumably, it is the combination of these effects that leads to such strong deviations compared to the sintering computations and experiments of [146].

CHAPTER 6

Conclusions and outlook

Within the scope of this thesis, partitioned solution approaches are applied to solve multi-physically coupled problems. The intended purpose is to develop a flexible coupling strategy for an arbitrary number of physical fields that can be treated with specialized solvers. Further emphasis is placed on increasing the chances for algorithmic stability and to provide an accurate data transfer in the case of incompatible meshes. The proposed strategy was shown to be applicable for several numerical computations. Apart from academic examples serving to test the partitioned scheme, an industrial application is taken under investigation. Here, the simulation focuses on complex interactions of four physical fields that describe the *field assisted sintering technology* (FAST).

The FAST process is governed by electric, thermal and mechanical fields. To this end, the descriptive equations of these fields are introduced and the corresponding coupled initial-boundary value problem is formulated. It is known from the literature, see e.g. [176, 76], that the temperature development during processing is of principle interest. As FAST deals with high heating rates and hot temperatures, the effect of thermal radiation becomes an important issue. That is why the radiative heat transfer is treated as an additional physical field in this thesis. A radiation solver is coupled to the thermal field and, in the event of large deformations, also to the mechanical field. For the former coupling, a *Dirichlet-Neumann* partition is proposed that describes the interaction of radiating surfaces with their surroundings. The latter coupling describes a situation in which the field must be related to the current configuration, leading to a *one-way Dirichlet-coupling*, i.e. iterating is not required to achieve balance. As radiation transfers thermal energy in vacuum, the coupling is also formulated for the absence of a participating medium. In summary, the FAST process is described by four physical fields which are coupled over the volume and the surface.

The spatial discretization of the FAST tools for the electric, thermal and mechanical fields is carried out by means of the finite element method (FEM). Further, the ambient of the tools is the radiation field, which is discretized by the finite volume method (FVM). In this context, different numerical models for the radiative transfer are discussed. For the special case of vacuum conditions, the *view factor method* (VFM) is utilized. For a participating medium, however, the *method of spherical harmonics* (P_1-approximation) and the *finite volume discrete ordi-*

nates method (fvDOM) are applied.

A further objective is to provide a robust and efficient solution strategy based on a separated treatment of the coupled problem. The proposed strategy follows the classical *Gauss-Seidel* iteration process with repetitive data exchange between the solvers. Algorithmic stability is increased by means of external stabilization methods. Several such procedures are investigated in this thesis. Apart from schemes that fall in the class of *Aitken relaxation* or *quasi-Newton methods*, so-called *vector sequence acceleration methods* are applied. However, in-depth performance studies on simple and academic examples show that these procedures cannot achieve the same convergence rates as Aitken relaxation or quasi-Newton methods. In this context, strongly coupled nonlinear thermoelasticity is solved by applying an *isothermal split*. As reported in [3] , this split is only conditionally stable. It is demonstrated in this thesis that the convergence stabilization methods even allow to solve unrealistically high coupling levels based on this split. Further, they are able to drastically reduce the required coupling iterations and, accordingly, to reduce the overall computation time as well.

In another simulation, the proposed *Dirichlet-Neumann* coupling between the radiation and the thermal field is investigated and also compared to experimental data. To this end, the finite element solver AdhoC for the thermal field is coupled with the finite volume solver OpenFOAM of the radiation field. It can be shown that the thermal-radiative coupling formulation delivers reliable results and a good agreement with the experiments. Further, the three different numerical radiation models are investigated and compared to the standard, but the less complex, *boundary model* – which approximates radiation as a boundary condition in the heat equation of the thermal field. This model is not coupled with an additional radiation solver. It turns out that the boundary model cannot reproduce the results with the same accuracy as the other radiation models. Another advantage of an external radiation solver is, that – in case of participating media – the surrounding fluid field is simulated. This allows to take *Buoyant-forces* into account, which are responsible for the effect of natural convection.

As selected numerical examples for the radiation-electro-thermo-mechanical four-field problem, a bimetallic beam in a chamber is considered, followed by the simulation of the FAST process. The example of the bimetallic beam serves to find an appropriate coupling strategy for the fields involved. Again, different coupling strengths between the fields are studied and the convergence acceleration methods are applied. As demonstrated for this example, the point of application of these methods, i.e. which of the four fields is stabilized, is a crucial point. It turns out that the thermal field is apparently the best choice as it is directly coupled to all other fields. Moreover, the computational effort for the different radiation models is compared. Here, it shows that it is especially the fvDOM that leads to a significant increase in the total simulation time.

Finally, the partitioned coupling approach is applied to simulate the FAST process. To begin with, the temperature evolution during processing is computed inside the graphite tools under the assumption of fully compacted powder. The objective is to gain a better understanding of thermal radiation and effects such as

self-irradiation of the FAST tools and reflection between the tools and the chamber walls. Again, radiation is considered as a physical field of its own – leading to promising results. Compared to the experimental data from [76], the temperature evolution can be treated better than with the simple boundary radiation model. The reason for this is that self-irradiation and reflection are covered, leading thermal energy back into the tools. Another interesting issue is that the vacuum of the chamber can be modeled by means of a very rarefied gas. This allows to apply the P_1-method and the fvDOM, which require the presence of a participating medium.

At the end, a constitutive powder model [146] is used to simulate the consolidation process of copper powder. Unfortunately, the simulation cannot reproduce the experimentally obtained data and the numerical results given in [146]. There are several reasons for those discrepancies, which are not so much related to the partitioned strategy but rather to the black-box solver used for the electric, thermal and mechanical fields.

In the following, some open questions shall be discussed – followed by an outlook concerning prospective research possibilities. With respect to the partitioned coupling approach, a parallel treatment of the fields involved poses interesting opportunities. Instead of solving the fields in a sequential manner (Gauss-Seidel formulation), a parallel approach (Gauss-Jacobi formulation) can lead to considerable savings of computation time. This can be of interest for problems that require an in-depth analysis of thermal radiation, which can be very cost-intensive. Attempts for a parallel solution of multi-physic problems can be found in [162, 152, 151, 23], for instance. Further research is required for aspects of stability regarding the Gauss-Jacobi procedure and the convergence.

Another interesting point is concerned with an improved time integration of the partitioned coupling strategy. Due to the flexibility, different time steps for the fields involved can be chosen, and an error-based time step selection combined with high-order time integration would complete this issue. Applied to a Gauss-Jacobi scheme, this offers further research possibilities and has – to the author's knowledge – not yet been investigated and, thus, might be a challenging open issue for the future.

It has been demonstrated that a comprehensive thermal radiation analysis can be an important ingredient to accurately simulate the temperature development of the FAST process. Further research needs to be conducted in this field to confirm these findings. Regarding simulations of the entire compaction process of a powder material, it would be an interesting aim for future work to focus on a coupled computation that involves the four physical fields and that can also handle aspects such as contact between the tools.

APPENDIX A

Appendix

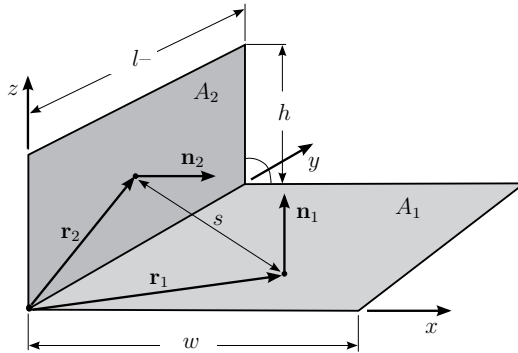## A.1 View factor radiation

Figure A.1: Schematic sketch of two finite perpendicular rectangles with one common edge.

In the following an example for view factor radiation is considered. For the enclosure given in Figure A.1 – two perpendicular rectangles – the view factor $F_{A_1-A_2}$ between the surface A_1 and A_2 shall be calculated. To begin with, the geometric relations as depicted in Figure A.1 are described. First, for arbitrary points P_1 and P_2 lying on the surfaces A_1 and A_2, the normal vectors $\mathbf{n}_i$ and the coordinate vectors $\mathbf{r}_i$ pointing from the origin to P_i are introduced

$$\mathbf{n}_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad \mathbf{n}_2 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{r}_1 = \begin{bmatrix} x_1 \\ y_1 \\ 0 \end{bmatrix} \quad \text{and} \quad \mathbf{r}_2 = \begin{bmatrix} 0 \\ y_2 \\ z_2 \end{bmatrix} \tag{A.1}$$

and, moreover, the distance vectors $\mathbf{s}_i$ between the points are determined

$$\mathbf{s}_{12} = \mathbf{r}_2 - \mathbf{r}_1 = \begin{bmatrix} -x_1 \\ y_2 - y_1 \\ z_2 \end{bmatrix} \quad \text{and} \quad \mathbf{s}_{21} = \mathbf{r}_1 - \mathbf{r}_2 = \begin{bmatrix} x_1 \\ y_1 - y_2 \\ -z_2 \end{bmatrix}. \tag{A.2}$$

Recapitulating the formula for infinitesimal view factor calculation

$$dF_{dA_1 - dA_2} = \frac{(\mathbf{n}_1 \cdot \mathbf{r}_{12})(\mathbf{n}_2 \cdot \mathbf{r}_{21})}{\pi \, S^4} \, dA_2, \tag{A.3}$$

requires the distance between two points on the surface

$$S^2 = |\mathbf{s}_{21}|^2 = |\mathbf{s}_{12}|^2 = x_1^2 + (y_2 - y_1)^2 + z_2^2 \tag{A.4}$$

and also the following scalar products

$$\mathbf{n}_1 \cdot \mathbf{r}_{12} = z_2 \quad \text{and} \quad \mathbf{n}_2 \cdot \mathbf{r}_{21} = x_1. \tag{A.5}$$

Introducing this into Eq. (A.3) and integrating over $dA_1 \, dA_2 = dx_1 \, dy_1 \, dz_2 \, dy_2$, leads to

$$F_{A_1 - A_2} = \frac{1}{\pi \, A_1} \int_{x_1} \int_{y_1} \int_{y_2} \int_{z_2} \frac{z_2 \, x_1}{\left(x_1^2 + (y_2 - y_1)^2 + z_2^2\right)^2} \, dz_2 \, dy_2 \, dx_1 \, dy_1. \tag{A.6}$$

Integrating four times using special computer software delivers the view factor

$$\begin{aligned} F_{1-2} = \frac{1}{\pi W} &\left[W \, \tan^{-1} \frac{1}{W} + H \, \tan^{-1} \frac{1}{H} - L \, \tan^{-1} \frac{1}{L} \right. \\ &\left. + \frac{1}{4} \ln \left(\frac{(1+W^2)(1+H^2)}{1+L^2} \left[\frac{W^2(1+L^2)}{(1+W^2)L^2} \right]^{W^2} \left[\frac{H^2(1+L^2)}{(1+H^2)L^2} \right]^{H^2} \right) \right], \end{aligned} \tag{A.7}$$

where the geometrical abbreviations $H = h/l$, $W = w/l$ and $L = \sqrt{H^2 + W^2}$ are utilized for clarity reasons. With $l = 30 \, \text{cm}$, $w = 20 \, \text{cm}$, $h = 10 \, \text{cm}$, $A_1 = w \times l = 0.6$ m^2, and $A_2 = h \times l = 0.3 \, \text{m}^2$ the view factor reads

$$F_{A_1 - A_2} = 0.1595. \tag{A.8}$$

The reciprocity rule delivers the remaining view factor

$$F_{A_2 - A_1} = \frac{A_1}{A_2} F_{A_1 - A_2} = 2 \, F_{A_1 - A_2} = 0.3190. \tag{A.9}$$

Assuming that the temperatures $\Theta_1 = 350$ K and $\Theta_2 = 300$ K and the emission coefficient $\epsilon_1 = 0.8$ and $\epsilon_2 = 0.5$ are given, the unknown heat fluxes can be calculated by

$$\mathbf{D} \, \mathbf{q} = \mathbf{A} \, \mathbf{e}_b \tag{A.10}$$

where $\mathbf{D}$ and $\mathbf{A}$ have the following components

$$D_{ij} = \frac{\delta_{ij} - F_{i-j}}{\epsilon_j} + F_{i-j} \quad \text{and} \quad A_{ij} = \delta_{ij} - F_{i-j} \,. \tag{A.11}$$

Evaluating both matrices leads to

$$\begin{bmatrix} 1.25 & -0.1595 \\ -0.07975 & 2 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} 1.0 & -0.1595 \\ -0.319 & 1.0 \end{bmatrix} \begin{bmatrix} \sigma_{\mathrm{sb}}\Theta_1^4 \\ \sigma_{\mathrm{sb}}\Theta_2^4 \end{bmatrix} = \sigma_{\mathrm{sb}} \begin{bmatrix} \Theta_1^4 - 0.1595\,\Theta_2^4 \\ \Theta_2^4 - 0.319\,\Theta_1^4 \end{bmatrix} . \tag{A.12}$$

Finally, the unknown heat fluxes on both surfaces read

$$\begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \sigma_{\mathrm{sb}} \begin{bmatrix} 1.25 & -0.1595 \\ -0.07975 & 2 \end{bmatrix}^{-1} \begin{bmatrix} \Theta_1^4 - 0.1595\,\Theta_2^4 \\ \Theta_2^4 - 0.319\,\Theta_1^4 \end{bmatrix} = \begin{bmatrix} 637.308 \\ 119.336 \end{bmatrix} \mathrm{W} \,. \tag{A.13}$$

A.2 Vector sequence acceleration methods

The vector sequence acceleration methods generate sets of transformed vectors which can be summarized by means of the following table:

$j \rightarrow$

$i \downarrow$ $\mathbf{z}_0^0$	$\mathbf{z}_1^0$	$\mathbf{z}_2^0$	$\mathbf{z}_3^0$	$\cdots$	$\mathbf{z}_m^0$
$\mathbf{z}_0^1$	$\mathbf{z}_1^1$	$\mathbf{z}_2^1$	$\mathbf{z}_3^1$		
$\mathbf{z}_0^2$	$\mathbf{z}_1^2$	$\mathbf{z}_2^2$	$\mathbf{z}_3^2$		
$\mathbf{z}_0^3$	$\mathbf{z}_1^3$	$\mathbf{z}_2^3$	$\mathbf{z}_3^3$	$\cdot^{\cdot^{\cdot}}$	
$\mathbf{z}_0^4$	$\mathbf{z}_1^4$	$\mathbf{z}_2^3$	$\vdots$		
$\mathbf{z}_0^5$	$\mathbf{z}_1^5$	$\vdots$	$\mathbf{z}_3^{n-3}$		
$\mathbf{z}_0^6$	$\vdots$	$\mathbf{z}_2^{n-2}$			
$\vdots$	$\mathbf{z}_1^{n-1}$				
$\mathbf{z}_0^n$					

In order to locate every vector of the sequences, two indices i and j are required and by definition, for $j = 0$ all vectors $\mathbf{z}_0^i$ are known. In other words, the $\mathbf{z}_0^i$ span the known vector sequence $\mathcal{S}$

$$\mathcal{S} := \left\{ \mathbf{z}^0, \mathbf{z}^1, ..., \mathbf{z}^n : \mathbf{z}^i \in \mathbb{R}^d \right\} , \tag{A.14}$$

which are the solution vectors from the partitioned coupling iteration. To accelerate the iteration process, the sequence $\mathcal{S}$ shall be transformed into a new sequence that exhibits better convergence properties. Based on this definition, j is the column index that denotes the j-th sequence and i is the row index that refers to the i-th vector of the corresponding sequence.

A.2.1 Wynn's ϵ-algorithm

One of the most famous algorithm to accelerate vector sequences is the so-called *vector ϵ-algorithm* (VEA) which was discovered by WYNN in the early 1960s [173]. It states that the vectors of the new sequences are calculated by using the vector inverse based on the following formulas

$$\epsilon_{-1}^i = 0 \quad \text{and} \quad \epsilon_0^i = \mathbf{z}^i , \quad i = 0, 1, ...$$

$$\epsilon_{j+1}^i = \epsilon_{j-1}^{i+1} + \left(\epsilon_j^{i+1} - \epsilon_j^i \right)^{-1} = \epsilon_{j-1}^{i+1} + \frac{\epsilon_j^{i+1} - \epsilon_j^i}{\| \epsilon_j^{i+1} - \epsilon_j^i \|^2} , \quad i, j = 1, 2, ... \tag{A.15}$$

where the $\mathbf{z}^i$ are known vectors of the given sequence $\mathcal{S}$. Based on this formula, the pattern of this algorithm reads

$$
\begin{array}{cccc}
j \rightarrow & & & \\
i \downarrow \quad \mathbf{z}^{i-1}_{j-1} & \mathbf{z}^{i-1}_j & \mathbf{z}^{i-1}_{j+1} & \mathbf{z}^{i-1}_{j+1} \\[2mm]
\mathbf{z}^i_{j-1} & \mathbf{z}^i_j & \mathbf{z}^i_{j+1} & \\[2mm]
\mathbf{z}^{i+1}_{j-1} & \mathbf{z}^{i+1}_j & & \\[2mm]
\mathbf{z}^{i+2}_{j-1} & & &
\end{array}
$$

Based on this pattern, the new vector is computed by using three other vectors. In order to apply the VEA to accelerate the convergence of the partitioned coupling strategy, the three last iterates $\{\,\mathbf{z}^k,\,\mathbf{z}^{k-1},\mathbf{z}^{k-2}\,\}$ are considered. To make this procedure available as an acceleration scheme, the sequence index j is set to one $j = 1$ which allows to design a procedure that can be applied after two new coupling iterations have been performed. For the new vector $\mathbf{z}^{k+1} = \boldsymbol{\epsilon}^i_2$ one obtains

$$
\mathbf{z}^{k+1} = \mathbf{z}^k + \frac{\mathbf{z}_2 - \mathbf{z}_1}{\|\mathbf{z}_2 - \mathbf{z}_1\|_2^2}\,, \quad \text{with} \quad \mathbf{z}_2 = \frac{\mathbf{z}^k - \mathbf{z}^{k-1}}{\|\mathbf{z}^k - \mathbf{z}^{k-1}\|_2^2}\,, \quad \mathbf{z}_1 = \frac{\mathbf{z}^{k-1} - \mathbf{z}^{k-2}}{\|\mathbf{z}^{k-1} - \mathbf{z}^{k-2}\|_2^2}\,.
$$
$$\tag{A.16}$$

This formula can now utilized to accelerate the coupling process in every third iteration. Beside this algorithm, an other generalization of the ϵ-algorithm for the acceleration of vector sequences was developed by Brezinski, see [16] for instance. This algorithm was called *topological ϵ-algorithm* (TEA) and which obtains an accelerated sequence by means of the following rule

$$
\begin{aligned}
\boldsymbol{\epsilon}^i_{-1} &= \mathbf{0} \quad \text{and} \quad \boldsymbol{\epsilon}^i_0 = \mathbf{z}^i\,, \quad i = 0, 1, \dots, \\
\boldsymbol{\epsilon}^i_{2j+1} &= \boldsymbol{\epsilon}^{i+1}_{2j-1} + \frac{\mathbf{h}}{\mathbf{h}^T \Delta \boldsymbol{\epsilon}^i_{2j}}\,, \quad i, j = 0, 1, \dots, \\
\boldsymbol{\epsilon}^i_{2j+2} &= \boldsymbol{\epsilon}^{i+1}_{2j} + \frac{\Delta \boldsymbol{\epsilon}^{i+1}_{2j}}{(\Delta \boldsymbol{\epsilon}^{i+1}_{2j})^T \Delta \boldsymbol{\epsilon}^i_{2j+1}}\,, \quad i, j = 0, 1, \dots \quad.
\end{aligned}
$$
$$\tag{A.17}$$

where the vector $\mathbf{h}$ is an arbitrary non-zero vector. Setting $j = 0$ and taking the last three iterates, leads after some mathematical rearrangements to a formula, see Eq. (4.62), which is similar to the VEA and which can be applied to accelerate the partitioned coupling algorithm.

A.2.2 Brezinski's θ-algorithm

Another method which shall be discussed is *Brezinski's Θ-algorithm* for scalar sequences and which was afterwards extended to the vector case, see [16] for in-

stance. In the multi-dimensional case, this transformation reads

$$\boldsymbol{\theta}^i_{-1} = 0 \quad \text{and} \quad \boldsymbol{\theta}^i_0 = \mathbf{z}^i\,, \quad i = 0, 1, \dots,$$

$$\boldsymbol{\theta}^i_{2j+1} = \boldsymbol{\theta}^{i+1}_{2j-1} + \frac{\Delta\boldsymbol{\theta}^i_{2j}}{\|\Delta\boldsymbol{\theta}^i_{2j}\|^2}\,, \quad i, j = 0, 1, 2, \dots,$$

$$\boldsymbol{\theta}^i_{2j+2} = \boldsymbol{\theta}^{i+1}_{2j} + \frac{(\Delta\boldsymbol{\theta}^{i+1}_{2j+1})^T \Delta^2\boldsymbol{\theta}^i_{2j+1}}{(\Delta^2\boldsymbol{\theta}^i_{2j+1})^T \Delta^2\boldsymbol{\theta}^i_{2j+1}}\Delta\boldsymbol{\theta}^{i+1}_{2j}\,, \quad i, j = 0, 1, 2, \dots.$$

(A.18)

and is called *vector θ-algorithm* (VTA). Further, there exist a second technique which was also proposed by BREZINSKI [16]. It reads

$$\boldsymbol{\theta}^i_{-1} = \mathbf{0} \quad \text{and} \quad \boldsymbol{\theta}^i_0 = \mathbf{z}^i\,, \quad i = 0, 1, \dots,$$

$$\boldsymbol{\theta}^i_{2j+1} = \boldsymbol{\theta}^{i+1}_{2j-1} + \frac{\mathbf{g}}{\mathbf{g}^T \Delta\boldsymbol{\theta}^i_{2j}}\,, \quad i, j = 0, 1, 2, \dots,$$

$$\boldsymbol{\theta}^i_{2j+2} = \boldsymbol{\theta}^{i+1}_{2j} + \omega^i_j \frac{\Delta\boldsymbol{\theta}^i_{2j}}{(\Delta\boldsymbol{\theta}^i_{2j+1})^T \Delta\boldsymbol{\theta}^i_{2j}} \quad \text{with} \quad \omega^i_j = -\frac{\mathbf{h}^T \Delta\boldsymbol{\theta}^{i+1}_{2j}}{\mathbf{h}^T \Delta\boldsymbol{\theta}^{i+1}_{2j+1}}\,, \quad i, j = 0, 1, 2, \dots$$

(A.19)

where $\mathbf{g} \in \mathbb{R}^{d\times 1}$ and $\mathbf{h} \in \mathbb{R}^{d\times 1}$ are arbitrary non-zero vectors used to avoid the case that the denominators vanish. Following BREZINSKI [16] this method is named as the *generalized θ-algorithm* (GTA) used. For both methods, setting $j = 0$ and taking the last four iterates into account, an applicable formula to accelerate the convergence of the partitioned coupling process can be obtained, see Eqns. (4.63) and (4.65).

A.2.3 W-algorithm

The third vector sequence acceleration method under consideration was proposed by Osada [131] and is named as *Euclidean W-transformation* (EWT). Following [131], the EWT scheme reads

$$\mathbf{W}^i_0 = \mathbf{z}^i\,, \quad i = 0, 1, 2, \dots$$

$$\mathbf{W}^i_j = \mathbf{W}^{i-1}_{j-1} - \frac{(\Delta\mathbf{W}^{i-1}_{j-1})^T \Delta^2\mathbf{W}^{i-3}_{j-1}}{(\Delta\mathbf{W}^{i-1}_{j-1})^T \Delta^2\mathbf{W}^{i-3}_{j-1} - (\Delta\mathbf{W}^{i-3}_{j-1})^T \Delta^2\mathbf{W}^{i-2}_{j-1}}\Delta\mathbf{W}^{i-2}_{j-1}\,, \quad (A.20)$$

$$j = 1, 2, \dots, \quad i = 3j, 3j+1, \dots$$

There exist a second procedure which is designated as the *vector W-transformation* (VWT) [131] based on the following formula

$$\mathbf{W}^i_0 = \mathbf{z}^i\,, \quad i = 0, 1, 2, \dots,$$

$$\mathbf{W}^i_j = \mathbf{W}^{i-1}_{j-1} +$$

$$\left(1 - \frac{(\Delta\mathbf{W}^{i-2}_{j-1})^T \Delta\mathbf{W}^{i-3}_{j-1}}{(\Delta\mathbf{W}^{i-3}_{j-1})^T \Delta\mathbf{W}^{i-3}_{j-1}}\right)\left(\frac{\Delta\mathbf{W}^{i-1}_{j-1}}{\|\Delta\mathbf{W}^{i-1}_{j-1}\|^2} - 2\frac{\Delta\mathbf{W}^{i-2}_{j-1}}{\|\Delta\mathbf{W}^{i-2}_{j-1}\|^2} + \frac{\Delta\mathbf{W}^{i-3}_{j-1}}{\|\Delta\mathbf{W}^{i-3}_{j-1}\|^2}\right)^{-1}\,,$$

$$j = 1, 2, \dots, \quad i = 3j, 3j+1, \dots.$$

$$(A.21)$$

Similar to the previous method, the limitation to $j = 0$ and taking the last four iterates into account leads to an acceleration formula for the convergence of the partitioned coupling process given by Eqns. (4.66) and (4.67).
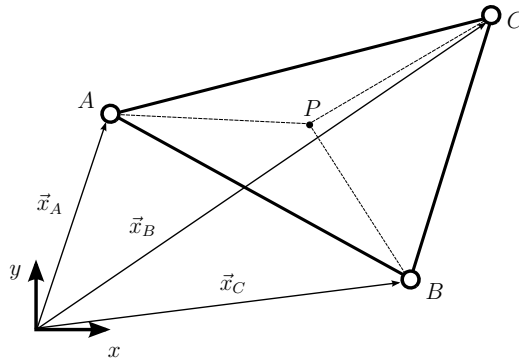
A.3   Barycentric coordinates

Figure A.2: Barycentric coordinates on the triangle $A\bar{B}C$

For the Barycentric mapping procedure it is necessary to express the Cartesian coordinates (x, y) in Barycentric coordinates w_1, w_2, w_3. For the sake of clarity, the two-dimensional case is considered first and extended to the three dimensions at the end of this paragraph. Cartesian coordinates can be expressed as

$$x = w_1 \, x_A + w_2 \, x_A + w_3 \, x_C \,, \tag{A.22}$$
$$y = w_1 \, y_A + w_2 \, y_A + w_3 \, y_C \,. \tag{A.23}$$

The w_i are also known as *Barycentric weights* and since $w_1 + w_2 + w_3 = 1$ must hold, one can deduce that

$$x = w_1 \, x_A + w_2 \, x_A + (1 - w_1 - w_2) \, x_C \,, \tag{A.24}$$
$$y = w_1 \, y_A + w_2 \, y_A + (1 - w_1 - w_2) \, y_C \,. \tag{A.25}$$

In this system of equations, the weights w_1 and w_2 are unknown. However, they can easily be calculated by solving

$$\begin{pmatrix} x_A - x_C & x_B - x_C \\ y_A - y_C & y_B - y_C \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} = \begin{pmatrix} x - x_C \\ y - y_C \end{pmatrix} \,. \tag{A.26}$$

Finally, one can find that

$$w_1(\vec{x}) = \frac{(y_B - y_C)(x - x_C) + (x_C - x_B)(y - y_C)}{(y_B - y_C)(x_A - x_C) + (x_C - x_B)(y_A - y_C)} \,, \tag{A.27}$$

$$w_2(\vec{x}) = \frac{(y_C - y_A)(x - x_C) + (x_A - x_C)(y - y_C)}{(y_C - y_A)(x_A - x_C) + (x_A - x_B)(y_A - y_C)} \,, \tag{A.28}$$

$$w_3(\vec{x}) = 1 - w_1(\vec{x}) - w_2(\vec{x}) \,. \tag{A.29}$$

Barycentric mapping means that data $\vec{d}$ given at the points A, B, C is projected on the point P by

$$\vec{d}(\vec{x}_P) = w_1(\vec{x}_P)\,\vec{d}_A(\vec{x}_A) + w_2(\vec{x}_P)\,\vec{d}_B(\vec{x}_B) + [1 - w_1(\vec{x}_P) - w_2(\vec{x}_P)]\;\vec{d}_C(\vec{x}_C)\,. \quad \text{(A.30)}$$

There is another way to determine the barycentric weights by considering the area of the triangle. Let $\overline{ABC}$ the area of the entire triangle, then one can deduce that

$$\overline{ABC} = \overline{ABP} + \overline{PBC} + \overline{APC} \quad \rightarrow \quad 1 = \frac{1}{\overline{ABC}}\left(\overline{ABP} + \overline{PBC} + \overline{APC}\right)\,. \quad \text{(A.31)}$$

Based on the area of the sub-triangles the barycentric weights can be calculated to

$$w_1 + w_2 + w_3 = \frac{\overline{ABP}}{\overline{ABC}} + \frac{\overline{PBC}}{\overline{ABC}} + \frac{\overline{APC}}{\overline{ABC}} = 1\,. \quad \text{(A.32)}$$

Consequently, it follows that the interpolation procedure is related to the percentage area of each sub-triangle. This procedure can easily be generalized to the three-dimensional case. Instead of a triangle, a tetrahedron is defined by four points which is spanned by four triangles. The calculation of the barycentric coordinates can be obtained in exactly the same way, however, now four points need to be considered and therefore, three weights w_1, w_2, w_3 – since $1 = w_1 + w_2 + w_3 + w_4$ must hold – need to be determined.

A.4 Constitutive model for copper powder

Elasticity relation

$$\underset{\sim}{S} = \frac{K}{2\vartheta^2}\left[\frac{\mathrm{tr}\left(\underset{\sim}{C}\underset{\sim}{C}_i^{-2}\right)}{\vartheta^2} - 3\right]\underset{\sim}{C}_i^{-1} + \frac{\mu}{\vartheta^4}\left[\underset{\sim}{C}_i^{-1}\underset{\sim}{C}\underset{\sim}{C}_i^{-1} - \frac{1}{3}\mathrm{tr}\left(\underset{\sim}{C}\underset{\sim}{C}_i^{-2}\right)\underset{\sim}{C}_i^{-1}\right]$$

Plastic flow-rule: elasticity ($\phi < 0$) or plasticity ($\phi = 0 \wedge \dot\phi \geq 0$)

$$\dot{\underset{\sim}{C}}_p = \begin{cases} \dfrac{\dot{f}_c}{f_c}(\underset{\sim}{I} - \underset{\sim}{C}_p) & \phi < 0 \\[2mm] \dfrac{2}{f_c}\Lambda\dfrac{\partial\phi}{\partial J_{2i}}\dfrac{\mu}{\vartheta^4}\underset{\sim}{C} + \dfrac{2}{f_c}\Lambda\left[\dfrac{\partial\phi}{\partial I_{1i}}\underset{\sim}{I} - \dfrac{1}{3}\dfrac{\partial\phi}{\partial J_{2i}}\dfrac{\mu}{\vartheta^4}\mathrm{tr}(\underset{\sim}{C}_p^{-1}\underset{\sim}{C})\underset{\sim}{I}\right]\underset{\sim}{C}_p \\[2mm] \quad + \dfrac{\dot{f}_c}{f_c}(\underset{\sim}{I} - \underset{\sim}{C}_p) & \phi = 0 \wedge \dot\phi \geq 0 \end{cases}$$

Interpolated single surface convex yield function

$$\phi = c\,k\,\ln\left(\frac{e^{g_1/(ck)} + e^{g_2/(ck)}}{2}\right), \quad g_1 = \sqrt{J_2 + \alpha(I_1 + 3\xi)^2} - k,$$

$$g_2 = \sqrt{J_2 + \Delta} - k + A_2 e^{A_3 I_1}$$

Creep flow-rule

$$\dot{\underset{\sim}{C}}_c = \frac{2}{3}g_v\langle I_{1c}\rangle\,\underset{\sim}{C}_c, \quad g_v = A_c(\Theta - \Theta_0)\left(1 - \frac{c_\infty}{f_c}\right)$$

$$\underset{\sim}{C}_c = f_c\underset{\sim}{I}, \quad f_c = \frac{2}{3}A_c(\Theta - \Theta_0)(f_c - c_\infty)\langle I_{1c}\rangle, \quad I_{1c} = f_c\,\mathrm{tr}\,\underset{\sim}{S}$$

Isotropic hardening

$$\dot{k}_M = \begin{cases} \dfrac{\gamma_k}{3k_M}\langle I_{1c}\rangle I_{1c} & \phi < 0 \\[2mm] \Lambda\dfrac{\gamma_k}{k_M}\left(\dfrac{\partial\phi}{\partial I_{1i}}(I_{1i} + 3\xi_M) + 2\dfrac{\partial\phi}{\partial J_{2i}}J_{2i}\right) \\[2mm] \quad - \Lambda\chi\beta_k k_M + \dfrac{\gamma_k}{3k_M}\langle I_{1c}\rangle I_{1c} & \phi = 0 \wedge \dot\phi \geq 0 \end{cases}$$

Hydrostatic kinematic hardening

$$\xi_M = -a_\xi e^{-b_\xi r_v} + c_\xi r_v - \frac{1}{1 - \frac{\rho_{0,\mathrm{rel}}}{\sqrt{\det\underset{\sim}{C}_i}}}, \quad r_v = \frac{1}{2}\ln\left(\sqrt{\det\underset{\sim}{C}_i}\right)$$

Hydrostatic Tensile Hardening

$$I_t = \beta_t\rho_{\mathrm{rel}}^2 e^{b_t\rho_{\mathrm{rel}}} + I_0, \quad \dot{\beta}_t = c_\beta(1 - \beta_t)\langle\Theta - \Theta_S\rangle$$

Miscellaneous

$$I_{1i} = \mathrm{tr}\left(\underset{\sim}{C}_i\underset{\sim}{S}\right), \quad J_{2i} = \frac{1}{2}\left(\underset{\sim}{C}_i\underset{\sim}{S}\cdot\underset{\sim}{S}\underset{\sim}{C}_i - \frac{1}{3}I_{1i}^2\right), \quad \underset{\sim}{C}_i = f_c\,\underset{\sim}{C}_p$$

The constitutive material model for copper powder that is used for the sintering simulations shall briefly be outlined. This model implies finite strain thermo-viscoplastic behavior and has been developed by ROTHE [146]. For a detailed explanation of the constitutive equations and the corresponding parameters reference is made to [146]. Therein, aspects of the implementation are provided and numerical sintering simulations are presented.

A.5 Unit system

In order to describe physical quantities and the material parameters, the standard metric SI unit system is utilized. The international base units are given in Table A.1.

Table A.1: SI base unit system.

Base unit name	Symbol	Unit	Unit symbol
Length	l	Meter	m
Mass	m	Kilogram	kg
Time	t	Second	s
Electric current	I	Ampere	A
Temperature	Θ	Kelvin	K

Based on this system, the following special units used in this thesis for the physical quantities can be established, as shown in Table A.2

Table A.2: Special units based on the SI system.

Quantity	Symbol	Unit	SI units	SI base units
Force	N	Newton		$\mathrm{m\ kg/s^2}$
Pressure	Pa	Pascal	$\mathrm{N/m^2}$	$\mathrm{kg/m\ s^{-2}}$
Energy	J	Joule	$\mathrm{N\ m}$	$\mathrm{m^2\ kg/s^2}$
Power	W	Watt	$\mathrm{J/s}$	$\mathrm{m^2\ kg/s^3}$
Voltage	V	Volt	$\mathrm{W/A}$	$\mathrm{m^2\ kg/s^3\ A^{-1}}$
Electric resistance	Ω	Ohm	$\mathrm{V/A}$	$\mathrm{m^2\ kg/s^3\ A^{-2}}$
Electric conductivity	S	Siemens	$\mathrm{A/V}$	$\mathrm{m^{-2}\ kg^{-1}\ s^3\ A^2}$

Bibliography

[1] Abaqus. *Abaqus 6.12 Documentation*. `www.3ds.com/products-services/simulia/portfolio/abaqus/overview/`, 2012.

[2] A. Aitken. On Bernoulli's numerical solution of algebraic equations. *Proceedings of the Royal Society of Edinburgh*, 46:289–305, 1926.

[3] F. Armero and J.C. Simo. A new unconditionally stable fractional step method for non-linear coupled thermomechanical problems. *International Journal for Numerical Methods in Engineering*, 35:737–766, 1992.

[4] F. Armero and J.C. Simo. A priori stability estimates and unconditionally stable product formula algorithms for nonlinear coupled thermoplasticity. *International Journal of Plasticity*, 9(6):749–782, 1993.

[5] R. Asaro and V.A. Lubarda. *Mechanics of Solids and Materials*. Cambridge University Press, 2006.

[6] M.S. Bartlett. An Inverse Matrix Adjustment Arising in Discriminant Analysis. *The Annals of Mathematical Statistics*, 22(1):107–111, 03 1951.

[7] K.-J. Bathe. *Finite Element Procedures*. Prentice Hall, 1996.

[8] R.C. Batra. *Elements of Continuum Mechanics*. AIAA education series. American Institute of Aeronautics and Astronautics, 2006.

[9] A. Beckert and H. Wendland. Multivariate interpolation for fluid-structure-interaction problems using radial basis functions. *Aerospace Science and Technology*, 5(2):125–134, 2001.

[10] T. Belytschko, W.K. Liu, and B. Moran. *Nonlinear Finite Elements for Continua and Structures*. John Wiley & Sons, 2000.

[11] J.-P. Berrut and L. N. Trefethen. Barycentric Lagrange Interpolation. *SIAM Rev*, 46:501–517, 2004.

[12] P. Birken, K.J. Quint, S. Hartmann, and A. Meister. A time-adaptive fluid-structure interaction method for thermal coupling. *Computing and Visualization in Science*, 13(7):331–340, 2010.

[13] J. Bonet and R.D. Wood. *Nonlinear Continuum Mechanics for Finite Element Analysis*. Cambridge University Press, New York, 2008.

[14] D. Brancherie, P. Villon, and A. Ibrahimbegovic. On a consistent field transfer in non linear inelastic analysis and ultimate load computation. *Computational Mechanics*, 42:213–226, 2007.

[15] S. Brändli and A. Düster. A flexible multi-physics coupling interface for partitioned solution approaches. *Proceedings in Applied Mathematics and Mechanics*, 12:363–364, 2012.

[16] C. Brezinski. *Projection Methods for Systems of Equations*. Studies in Computational Mathematics. Elsevier, Amsterdam, New York, 1997.

[17] C. Brezinski. Convergence acceleration during the 20th century. *Journal of Computational and Applied Mathematics*, 122(1-2):1–21, 2000. Numerical Analysis in the 20th Century Vol. II: Interpolation and Extrapolation.

[18] C. Brezinski and J.-P. Chehab. Nonlinear Hybrid Procedures and Fixed Point Iterations. *Numerical Functional Analysis and Optimization*, 19(5–6):465–487, 1998.

[19] H. Bröker. *Integration von geometrischer Modellierung und Berechnung nach der p-Version der FEM*. PhD thesis, Lehrstuhl für Bauinformatik, Fakultät für Bauingenieur- und Vermessungswesen, Technische Universität München, 2001.

[20] P.N. Brown and Y. Saad. Hybrid Krylov methods for nonlinear systems of equations. *SIAM Journal of Scientific and Statistical Computing*, 11:450–481, 1990.

[21] C. Broyden. A class of methods for solving nonlinear simultaneous equations. *Mathematics of Computation*, 19:577–593, 1965.

[22] C. G. Broyden, J. E. Dennis, and J. J. Moré. On the local and superlinear convergence of quasi-Newton methods. *Journal of the Institute of Mathematics and its Applications*, 12:223–245, 1973.

[23] H.-J. Bungartz, F. Lindner, M. Mehl, and B. Uekermann. A plug-and-play coupling approach for parallel multi-field simulations. *Computational Mechanics*, pages 1–11, 2014.

[24] R. H. Byrd, J. Nocedal, and R. B. Schnabel. Representations of quasi-Newton Matrices and Their Use in Limited Memory Methods. *Mathematical Programming*, 63(2):129–156, 1994.

[25] S. Chandrasekhar. *Radiative Transfer*. Dover Publications, New York, 2013.

[26] E.W.V. Chaves. *Notes on Continuum Mechanics: Fundamental Concepts and Constitutive Equations*. Lecture Notes on Numerical Methods in Engineering and Sciences. Springer Netherlands, 2013.

[27] P.J. Coelho. *Discrete-ordinate and finite volume methods.* `www.thermopedia.com`, 2011.

[28] W. J. Costin and C. B. Allen. Numerical study of radial basis function interpolation for data transfer across discontinuous mesh interfaces. *International Journal for Numerical Methods in Fluids*, 72(10):1076–1095, 2013.

[29] M.A. Crisfield. *Non-linear Finite Element Analysis of Solids and Structures, Volume 2.* John Wiley & Sons, 1997.

[30] C. Danowski, V. Gravemeier, L. Yoshihara, and W. A. Wall. A monolithic computational approach to thermo-structure interaction. *International Journal for Numerical Methods in Engineering*, 95(13):1053–1078, 2013.

[31] A. de Boer, M.S. Van der Schoot, and H. Bijl. New method for mesh moving based on radial basis function interpolation. In *ECCOMAS CFD 2006: Proceedings of the European Conference on Computational Fluid Dynamics*, 2006.

[32] A. de Boer, M.S. van der Schoot, and H. Bijl. Mesh deformation based on radial basis function interpolation. *Computers & Structures*, 85(11-14):784–795, 2007. Fourth MIT Conference on Computational Fluid and Solid Mechanics.

[33] A. de Boer, A.H. van Zuijlen, and H. Bijl. Review of coupling methods for non-matching meshes. *Computer Methods in Applied Mechanics and Engineering*, 196:1515–1525, 2007.

[34] A. de Boer, A.H. van Zuijlen, and H. Bijl. Comparison of conservative and consistent approaches for the coupling of non-matching meshes. *Computer Methods in Applied Mechanics and Engineering*, page doi:10.1016/j.cma.2008.05.001, 2008.

[35] A. de Boer, A.H. van Zuijlen, and H. Bijl. Radial Basis Functions for Interface Interpolation and Mesh Deformation. In Barry Koren and Kees Vuik, editors, *Advanced Computational Methods in Science and Engineering*, volume 71 of *Lecture Notes in Computational Science and Engineering*, pages 143–178. Springer Berlin Heidelberg, 2010.

[36] J. Degroote, K.-J. Bathe, and J. Vierendeels. Performance of a new partitioned procedure versus a monolithic procedure in fluid-structure interaction. *Computers & Structures*, 87:793–801, 2009.

[37] J. Degroote, R. Haelterman, S. Annerel, P. Bruggeman, and J. Vierendeels. Performance of partitioned procedures in fluid-structure interaction. *Computers & Structures*, 88(7-8):446–457, 2010.

[38] J. Degroote and J. Vierendeels. Multi-solver algorithms for the partitioned simulation of fluid-structure interaction. *Computer Methods in Applied Mechanics and Engineering*, 200(25-28):2195–2210, 2011.

[39] Joris Degroote and Jan Vierendeels. Multi-level quasi-newton coupling algorithms for the partitioned simulation of fluid-structure interaction. *Computer Methods in Applied Mechanics and Engineering*, 225:14–27, 2012.

[40] L. Dorfmann. Modeling Nonlinear Electroelastic Materials. In *Mechanics and Electrodynamics of Magneto- and Electro-elastic Materials*, CISM International Centre for Mechanical Sciences. Springer, 2011.

[41] D. Dureisseix and H. Bavestrello. Information transfer between incompatible finite element meshes: Application to coupled thermo-viscoelasticity. *Computer Methods in Applied Mechanics and Engineering*, 195:6523–6541, 2006.

[42] A. Düster. *High order finite elements for three-dimensional, thin-walled nonlinear continua*. PhD thesis, Lehrstuhl für Bauinformatik, Fakultät für Bauingenieur- und Vermessungswesen, Technische Universität München, 2001.

[43] A. Düster, H. Bröker, and E. Rank. The p-version of the finite element method for three-dimensional curved thin walled structures. *International Journal for Numerical Methods in Engineering*, 52:673–703, 2001.

[44] A. Düster, S. Hartmann, and E. Rank. p-FEM applied to finite isotropic hyperelastic bodies. *Computer Methods in Applied Mechanics and Engineering*, 192:5147–5166, 2003.

[45] A. Düster and S. Kollmannsberger. *AdhoC^4 – User's Guide*. Lehrstuhl für Computation in Engineering, TU München, Numerische Strukturanalyse mit Anwendungen in der Schiffstechnik, TU Hamburg-Harburg, 2010.

[46] A. Düster and E. Rank. The p-version of the finite element method compared to an adaptive h-version for the deformation theory of plasticity. *Computer Methods in Applied Mechanics and Engineering*, 190:1925–1935, 2001.

[47] A. Düster and E. Rank. A p-version finite element approach for two- and three-dimensional problems of the J_2 flow theory with non-linear isotropic hardening. *International Journal for Numerical Methods in Engineering*, 53:49–63, 2002.

[48] P. Ellsiepen and S. Hartmann. Remarks on the interpretation of current non-linear finite element analyses as differential-algebraic equations. *International Journal for Numerical Methods in Engineering*, 51(6):679–707, 2001.

[49] M. S. Engelman, G. Strang, and K.-J. Bathe. The application of quasi-Newton methods in fluid mechanics. *International Journal for Numerical Methods in Engineering*, 17(5):707–718, 1981.

[50] P. Erbts and A. Düster. Accelerated staggered coupling schemes for problems of thermoelasticity at finite strains. *Computers & Mathematics with Applications*, 64(8):2408–2430, 2012.

[51] P. Erbts, S. Hartmann, and A. Düster. A partitioned solution approach for electro-thermo-mechanical problems. *Archive of Applied Mechanics*, 85(8):1075–1101, 2014.

[52] P. Erbts, S. Rothe, A. Düster, and S. Hartmann. Energy-Conserving Data Transfer in the Partitioned Treatment of Thermo-Viscoplastic Problems. In *Proceedings in Applied Mathematics and Mechanics*, volume 13, pages 211–212, 2013.

[53] I. Faragó. A modified iterated operator splitting method. *Applied Mathematical Modelling*, 32(8):1542 – 1551, 2008.

[54] I. Faragó and J. Geiser. Iterative operator-splitting methods for linear problems. *International Journal of Computational Science and Engineering*, 3(4):255 – 263, 2007.

[55] I. Faragó, B. Gnandt, and Á Havasi. Additive and iterative operator splitting methods and their numerical investigation. *Computers and Mathematics with Applications*, 55(10):2266 – 2279, 2008.

[56] C. Farhat, M. Lesoinne, and P. Le Tallec. Load and motion transfer algorithms for fluid/structure interaction problems with non-matching discrete interfaces: Momentum and energy conservation, optimal discretization and application to aeroelasticity. *Computer Methods in Applied Mechanics and Engineering*, 157:95–114, 1998.

[57] C. Farhat, M. Lesoinne, and P. Le Tallec. Geometric conservation laws for flow problems with moving boundaries and deformable meshes, and their impact on aeroelastic computations. *Computer Methods in Applied Mechanics and Engineering*, 134:71–90, 2000.

[58] C. Farhat, K. C. Park, and Y. Dubois-Pelerin. An unconditionally stable staggered algorithm for transient finite element analysis of coupled thermoelastic problems. *Computer Methods in Applied Mechanics and Engineering*, 85(85):349–365, 1991.

[59] C. Felippa and T. L. Geers. Partitioned analysis for coupled mechanical systems. *Engineering Computations*, 5:123–133, 1988.

[60] C. Felippa, K. Park, and J. DeRuntz. Stabilization of staggered solution procedures for fluid-structure interaction analysis. In T. Belytschko and T. Geers, editors, *Computational Methods for Fluid-Structure Interaction Problems*, volume 26, pages 95–124. American Society of Mechanical Engineers, New York, 1977.

[61] M. Á. Fernández and M. Moubachir. A newton method using exact jaco-bians for solving fluid-structure coupling. *Computers & Structures*, 83(2-3):127–142, 2005.

[62] J.H. Ferziger and M. Peric. *Computational Methods for Fluid Dynamics*. Springer Berlin Heidelberg, 2001.

[63] P.J. Flory. Thermodynamic relations for high elastic materials. *Transaction of the Faraday Society*, 57:829–838, 1961.

[64] T.G. Gallinger. *Effiziente Algorithmen zur partitionierten Lösung stark gekop-pelter Probleme der Fluid-Struktur-Wechselwirkung*. PhD thesis, Technische Universität München, München, 2010.

[65] J. Geiser. *Decomposition Methods for Differential Equations: Theory and Applica-tions*. Chapman & Hall/CRC Numerical Analysis and Scientific Computing Series. Taylor & Francis, 2009.

[66] J. Geiser. *Iterative Splitting Methods for Differential Equations*. Chapman & Hall/CRC Numerical Analysis and Scientific Computing Series. Taylor & Francis, 2011.

[67] S.A. Gharehbaghi and A.R. Khoei. Three-dimensional superconvergent patch recovery method and its application to data transferring in small-strain plasticity. *Computational Mechanics*, 41:293–312, 2007.

[68] R.E. Glass, M. Burgess, E. Livesey, J. Geoffrey, S. Bourdon, D. Mennerdahl, A. Cherubini, S. Giambuzzi, and P. Nagel. Standard thermal problem set. Technical report, Sandia National Labs., Albuquerque, NM (USA), 1989.

[69] P.R. Graves-Morris. Extrapolation methods for vector sequences. *Numerische Mathematik*, 61(1):475–487, 1992.

[70] A.-W. Hamkar and S. Hartmann. Theoretical and numerical aspects in weak-compressible finite strain thermo-elasticity. *Journal of Theoretical and Applied Mechanics*, 50(1):3–22, 2012.

[71] S. Hartmann. Computation in finite strain viscoelasticity: finite elements based on the interpretation as differential-algebraic equations. *Computer Methods in Applied Mechanics and Engineering*, 191(13-14):1439–1470, 2002.

[72] S. Hartmann. *Finite-Elemente Berechnung inelastischer Kontinua - Interpreta-tion als Algebro-Differentialgleichungssysteme*. Postdoctoral thesis, Institut für Mechanik, Universität Kassel, 2003.

[73] S. Hartmann. A remark on the application of the newton-raphson method in non-linear finite element analysis. *Computational Mechanics*, 36(2):100–116, 2005.

[74] S. Hartmann. Comparison of the multiplicative decompositions $\mathbf{F} = \mathbf{F}_\theta \, \mathbf{F}_M$ and $\mathbf{F} = \mathbf{F}_M \, \mathbf{F}_\theta$ in finite strain thermo-elasticity. Technical report, Technical Report Series Fac3-12-01, Faculty of Mathematics/Computer Sciences and Mechanical Engineering, Clausthal University of Technology (Germany), 2012.

[75] S. Hartmann, J. Duintjer Tebbens, K.J. Quint, and A. Meister. Iterative solvers within sequences of large linear systems in non-linear structural mechanics. *ZAMM-Zeitschrift für Angewandte Mathematik und Mechanik*, 89(9):711–728, 2009.

[76] S. Hartmann, S. Rothe, and N. Frage. Electro-Thermo-Elastic Simulation of Graphite Tools Used in SPS Processes. In H. Altenbach, S. Forest, and A. Krivtsov, editors, *Generalized Continua As Models for Materials: With Multi-scale Effects Or Under Multi-field Actions*, Advanced Structured Materials. Springer-Verlag Berlin Heidelberg, 2013.

[77] P. Haupt. *Continuum Mechanics and Theory of Materials*. Springer, 2000.

[78] M.A. Heaslet and R.F. Warming. Radiative transport and wall temperature slip in an absorbing planar medium. *International Journal for Heat and Mass Transfer*, 8:979–994, 1965.

[79] M. Heil. An efficient solver for the fully coupled solution of large-displacement fluid-structure interaction problems. *Computer Methods in Applied Mechanics and Engineering*, 193(1-2):1–23, 2004.

[80] M. Heil, A. L. Hazel, and J. Boyle. Solvers for large-displacement fluid-structure interaction problems: segregated versus monolithic approaches. *Computational Mechanics*, 43(1):91–101, 2008.

[81] U. Heißerer. *High-order finite elements for material and geometric nonlinear finite strain problems*. PhD thesis, Chair for Computation in Engineering, Fakultät für Bauingenieur- und Vermessungswesen, Technische Universität München, 2008.

[82] U. Heißerer, S. Hartmann, A. Düster, W. Bier, Z. Yosibash, and E. Rank. *p*-fem for finite deformation powder compaction. *Computer Methods in Applied Mechanics and Engineering*, 197:727–740, 2008.

[83] H.E. Hinnant. A fast method of numerical quadrature for *p*-version finite element matrices. *International Journal for Numerical Methods in Engineering*, 37:3723–3750, 1994.

[84] G.A. Holzapfel. *Nonlinear Solid Mechanics*. John Wiley & Sons, 2000.

[85] H. C. Hottel. Radiant heat transmission. *Heat transmission*, 3, 1954.

[86] J.R. Howell, R. Siegel, and P. Menguc. *Thermal Radiation Heat Transfer, 5th Edition*. Taylor & Francis, 2010.

[87] B. Hübner, E. Walhorn, and D. Dinkler. A monolithic approach to fluid-structure interaction using space-time finite elements. *Computer Methods in Applied Mechanics and Engineering*, 193:2087–2104, 2004.

[88] T.J.R. Hughes. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Dover Publications, 2000.

[89] T.J.R Hughes and K.S. Pister. Consistent linearization in mechanics of solids and structures. *Computers & Structures*, 8(3-4):391–397, 1978.

[90] A. Ibrahimbegovic, L. Chorfi, and F. Gharzeddine. Thermomechanical coupling at finite elastic strain: covariant formulation and numerical implementation. *Communications in Numerical Methods in Engineering*, (17):275–289, 2001.

[91] B. Irons and R. C. Tuck. A Version of the Aitken Accelerator for Computer Implementation. *International Journal for Numerical Methods in Engineering*, 1:275–277, 1969.

[92] H. Jasak and Z. Tukovic. Automatic mesh motion for the unstructured finite volume method. *Transactions of FAMENA*, 30(2):1–20, 2006.

[93] H. Jasak and Z. Tukovic. Dynamic mesh handling in OpenFOAM applied to fluid-structure interaction simulations. In *Proceedings of the V European Conference Computational Fluid Dynamics*, 2010.

[94] J.H. Jeans. The equations of radiative transfer of energy. *Monthly Notices Royal Astronomical Society*, 78:28–36, 1917.

[95] A. Jennings. Accelerating the Convergence of Matrix Iterative Processes. *IMA Journal of Applied Mathematics*, 8(1):99–110, 1971.

[96] A.A. Johnson and T.E. Tezduyar. Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and interfaces. *Computer methods in applied mechanics and engineering*, 119(1):73–94, 1994.

[97] M.M. Joosten, W.G. Dettmer, and D. Perić. Analysis of the block Gauss-Seidel solution procedure for a strongly coupled model problem with reference to fluid-structure interaction. *International Journal for Numerical Methods in Engineering*, 78(7):757–778, 2009.

[98] W.M. Kahan. *Gauss-Seidel methods of solving large systems of linear equations*. PhD thesis, University of Toronto, 1958.

[99] C. Kassiotis, J.-B. Colliat, A. Ibrahimbegovic, and H.G. Matthies. Multiscale in time and stability analysis of operator split solution procedures applied to thermomechanical problems. *Engineering Computations*, 26(1/2):205–223, 2009.

[100] D.A. Kay, P.M. Gresho, D.F. Griffiths, and D.J. Silvester. Adaptive time-stepping for incompressible flow, part I: Scalar advection-diffusion. *SIAM Journal on Scientific Computing*, 30:2018–2054, 2008.

[101] C.T. Kelley. *Iterative Methods for Linear and Nonlinear Equations*. Society for Industrial and Applied Mathematics, 1995.

[102] C.T. Kelley. *Solving Nonlinear Equations with Newton's Method*. SIAM, 1 edition, 2003.

[103] D.A. Knoll and D.E. Keyes. Jacobian-free Newton-Krylov methods: a survey of approaches and applications. *Journal of Computational Physics*, 193(2):357 – 397, 2004.

[104] P.K. Kundu and I.M. Cohen. *Fluid Mechanics*. Elsevier Science, 2010.

[105] U. Küttler. *Effiziente Lösungsverfahren für Fluid-Struktur-Interaktions-Probleme*. Dissertation, Lehrstuhl fü Numerische Mechanik, TU München, 2009.

[106] U. Küttler and W. Wall. Fixed-point fluid-structure interaction solvers with dynamic relaxation. *Computational Mechanics*, 1(43):61–72, 2008.

[107] U. Küttler and W. Wall. Vector Extrapolation for Strong Coupling Fluid-Structure Interaction Solvers. *Journal of Applied Mechanics*, 2(76), 2009.

[108] P.A. Libby. *An Introduction to Turbulence*. Taylor & Francis, 1996.

[109] R. Löhner and C. Yang. Improved ALE mesh velocities for moving bodies. *Communications in numerical methods in engineering*, (12):599–608, 1998.

[110] S. Lu and K. Pister. Decomposition of deformation and representation of the free energy function for isotropic thermoelastic solids. *International Journal of Solids and Structures*, (11):927–935, 1975.

[111] V.A. Lubarda. Constitutive theories based on the multiplicative decomposition of deformation gradient: Thermoelasticity, elastoplasticity, and biomechanics. *Applied Mechanics Reviews*, 57(2):95, 2004.

[112] A.J. MacLeod. Acceleration of vector sequences by multi-dimensional Δ^2 methods. *Communications in Applied Numerical Methods*, 1:3–20, 1986.

[113] G.A. Maugin. Electromagnetics in Deformable Solids. In *Mechanics and Electrodynamics of Magneto- and Electro-elastic Materials*, CISM International Centre for Mechanical Sciences. Springer, 2011.

[114] B. McWilliams and A. Zavaliangos. Multi-phenomena simulation of electric field assisted sintering. *Journal of Material Sciences*, 43:5031–5035, 2008.

[115] J.M. Melenk, K. Gerdes, and Ch. Schwab. Fully discrete hp finite elements: fast quadrature. *Computer Methods in Applied Mechanics and Engineering*, 190:4339–4364, 2001.

[116] C. Michler, E.H . van Brummelen, and R. de Borst. An Interface Newton-Krylov Solver for Fluid-Structure Interaction. *International Journal for Numerical Methods in Fluids*, 47:1189–1195, 2005.

[117] C. Michler, E.H. van Brummelen, and R. de Borst. Error-amplification analysis of subiteration-preconditioned GMRES for fluid-structure interaction. *Computer Methods in Applied Mechanics and Engineering*, 195(17-18):2124–2148, 2006.

[118] C. Michler, H. van Brummelen, and R. de Borst. An investigation of Interface-GMRES(R) for fluid-structure interaction problems with flutter and divergence. *Computational Mechanics*, 47(1):17–29, 2011.

[119] S. Minami and S. Yoshimura. Performance evaluation of nonlinear algorithms with line-search for partitioned coupling techniques for fluid-structure interactions. *International Journal for Numerical Methods in Fluids*, 64:1129–1147, 2010.

[120] D. P. Mock. *Partitionierte Lösungsansätze in der Strukturdynamik und der Fluid-Struktur-Interaktion*. PhD thesis, Institut für Baustatik, Universität Stuttgart, 2001.

[121] M. F. Modest. *Radiative Heat Transfer*. Academic Press, 2nd edition, 2003.

[122] M. F. Modest and J. Yang. Elliptic PDE formulation and boundary conditions of the spherical harmonics method of arbitrary order for general three-dimensional geometries. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 109(9):1641–1666, 2008.

[123] M. Muja and D.G. Lowe. Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration. In *International Conference on Computer Vision Theory and Application VISSAPP'09)*, pages 331–340. INSTICC Press, 2009.

[124] M. Muja and D.G. Lowe. Scalable Nearest Neighbor Algorithms for High Dimensional Data. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 36, 2014.

[125] Z.A. Munir, U. Anselmi-Tamburini, and M. Ohyanagi. The effect of electric field and pressure on the synthesis and consolidation of materials: A review of the spark plasma sintering method. *Journal of Materials Science*, 41(3):763–777, 2006.

[126] T. Netz, A. Düster, and S. Hartmann. High-order finite elements compared to low-order mixed element formulations. *ZAMM-Zeitschrift für Angewandte Mathematik und Mechanik*, 93(2-3):163–176, 2013.

[127] T. Netz, A.-W. Hamkar, and S. Hartmann. High-order quasi-static finite element computations in space and time with application to finite strain viscoelasticity. *Computers & Mathematics with Applications*, 66(4):441 – 459, 2013.

[128] J. Nocedal and S.J. Wright. *Numerical Optimization*. Springer, 1999.

[129] E. A. Olevsky, C. Garcia-Cardona, W. L. Bradbury, C. D. Haines, D. G. Martin, and D. Kapoor. Fundamental Aspects of Spark Plasma Sintering: II. Finite Element Analysis of Scalability. *Journal of the American Ceramic Society*, 95(8):2414–2422, 2012.

[130] OpenCFD Ltd. Openfoam ist eine eingetragende marke der Silicon Graphics International Corp. URL: http://www.openfoam.com.

[131] N. Osada. Acceleration methods for vector sequences. *Journal of Computational and Applied Mathematics*, 38(1-3):361–371, 1991.

[132] K.C. Park and C. A. Felippa. Partitioned analysis of coupled systems. *Computational methods for transient analysis*, 1:157–219, 1983.

[133] S.V. Patankar. *Numerical Heat Transfer and Fluid Flow*. Series in computational methods in mechanics and thermal sciences. Taylor & Francis, 1980.

[134] S.V. Patankar and D.B Spalding. A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *International Journal of Heat and Mass Transfer*, 15(10):1787–1806, 1972.

[135] R. Penrose. A generalized inverse for matrices. *Mathematical Proceedings of the Cambridge Philosophical Society*, 51:406–413, 1955.

[136] S. Piperno. Explicit / implicit fluid-structure staggered procedures with a structural predictor and fluid subcycling for 2d inviscid aeroelastic simulations. *International Journal for Numerical Methods in Fluids*, 25:1207–1226, 1997.

[137] P. Pironkov. *Numerical Simulation of Thermal Fluid-Structure Interaction*. PhD thesis, TU Darmstadt, 2010.

[138] R.H. Pletcher, J.C. Tannehill, and D. Anderson. *Computational Fluid Mechanics and Heat Transfer, Third Edition*. Series in Computational and Physical Processes in Mechanics and Thermal Sciences. Taylor & Francis, 2012.

[139] S.B. Pope. *Turbulent Flows*. Cambridge University Press, 2000.

[140] D.V. Quach, J.R. Groza, A. Zavaliangos, and U. Anselmi-Tamburini. 10 - Fundamentals and applications of field/current assisted sintering. In Zhigang Zak Fang, editor, *Sintering of Advanced Materials*, Woodhead Publishing Series in Metals and Surface Engineering, pages 249 – 275. Woodhead Publishing, 2010.

[141] K. Quint. *Thermomechanically coupled processes for functionally graded materials: experiments, modelling, and finite element analysis using high-order DIRK-methods*. PhD thesis, Technische Universität Clausthal, Clausthal, 2012.

[142] K.J. Quint and S. Hartmann. Time-adaptive analysis of thermo-mechanically coupled plasticity. In *International Conference on Computational Methods for Coupled Problems in Science and Engineering*, 2009.

[143] K.J. Quint, S. Hartmann, S. Rothe, N. Saba, and K. Steinhoff. Experimental validation of high-order time integration for non-linear heat transfer problems. *Computational Mechanics*, 48(1):81–96, 2011.

[144] J.N. Reddy. *An Introduction to Nonlinear Finite Element Analysis*. OUP Oxford, 2004.

[145] J.N. Reddy. *An Introduction to the Finite Element Method*. Engineering Series. McGraw-Hill Education, 2005.

[146] S. Rothe. *Electro-Thermo-Mechanical Modeling of Field Assisted Sintering Technology: Experiments, Constitutive Modeling and Finite Element Analysis*. PhD thesis, Technische Universität Clausthal, Clausthal, 2014.

[147] S. Rothe, P. Erbts, S. Hartmann, and A. Düster. Monolithic and partitioned coupling schemes for thermo-viscoplasticity. *Computer Methods in Applied Mechanics and Engineering*, 293:375–410, 2014.

[148] S. Rothe, A.-W. Hamkar, K.J. Quint, and S. Hartmann. Comparison of diagonal-implicit, linear-implicit and half-explicit Runge-Kutta methods in non-linear finite element analyses. *Archive of Applied Mechanics*, 82(8):1057–1074, 2012.

[149] S. Rothe, J. Schmidt, and S. Hartmann. Analytical and numerical treatment of electro-thermo-mechanical coupling. *Archive of Applied Mechanics*, 2014.

[150] M.v. Scheven. *Effiziente Algorithmen für die Fluid-Struktur-Wechselwirkung*. PhD thesis, Universität Stuttgart, Stuttgart, 2009.

[151] S. Sicklinger. *Stabilized Co-Simulation of Coupled Problems Including Fields and Signals*. PhD thesis, Technische Universität München, München, 2014.

[152] S. Sicklinger, V. Belsky, B. Engelmann, H. Elmqvist, H. Olsson, R. Wüchner, and K.-U. Bletzinger. Interface Jacobian-based Co-Simulation. *International Journal for Numerical Methods in Engineering*, 98(6):418–444, 2014.

[153] J.C. Simo and C. Miehe. Associative coupled thermoplasticity at finite strains: Formulation, numerical analysis and implementation. *Computer Methods in Applied Mechanics and Engineering*, 98:41–104, 1991.

[154] J.C. Simo, R.L. Taylor, and K.S. Pister. Variational and projection methods for the volume constraint in finite deformation elasto-plasticity. *Computer Methods in Applied Mechanics and Engineering*, 51:177–208, 1985.

[155] Y. Song, L. Yuanyuan, Z. Zhaoyao, L. Yangen, and Y. Yoangquan. A multi-field coupled FEM model for one-step-forming process of spark plasma sintering considering local densification of powder material. *Journal of Material Sciences*, 46:5645–5656, 2011.

[156] E. Stein, editor. *Error-Controlled Adaptive Finite Elements in Solid Mechanics*. John Wiley & Sons, 2002.

[157] B.A. Szabó and I. Babuška. *Finite Element Analysis*. John Wiley & Sons, 1991.

[158] B.A. Szabó, A. Düster, and E. Rank. The p-version of the Finite Element Method. In E. Stein, R. de Borst, and T. J. R. Hughes, editors, *Encyclopedia of Computational Mechanics*, volume 1, chapter 5, pages 119–139. John Wiley & Sons, 2004.

[159] E.B. Tadmor, R.E. Miller, and R.S. Elliott. *Continuum Mechanics and Thermodynamics: From Fundamental Concepts to Governing Equations*. Cambridge University Press, 2012.

[160] J.D. Thomas and N. Triantafyllidis. On electromagnetic forming processes in finitely strained solids: Theory and examples. *Journal of the Mechanics and Physics of Solids*, 57(8):1391 – 1416, 2009.

[161] C. Truesdell and W. Noll. *The Non-Linear Field Theories of Mechanics*, chapter The Non-Linear Field Theories of Mechanics, pages 1–579. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[162] B. Uekermann, H.-J. Bungartz, B. Gatzhammer, and M. Mehl. A parallel, black-box coupling algorithm for fluid-structure interaction. In *Proceedings of 5th International Conference on Computational Methods for Coupled Problems in Science and Engineering*, 2013.

[163] K. Vanmeensel, A. Laptev, J. Hennicke, J. Vleugels, and O. Van der Biest. Modelling of the temperature distribution during field assisted sintering. *Acta Materialia*, 53:4379–4388, 2005.

[164] H.K. Versteeg and W. Malalasekera. *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*. Pearson Education Limited, 2007.

[165] J. Vierendeels, L. Lanoye, J. Degroote, and P. Verdonck. Implicit coupling of partitioned fluid-structure interaction problems with reduced order models. *Computers & Structures*, 85:970–976, 2007.

[166] L. Vujosevic and V.A. Lubarda. Finite-strain thermoelasticity based on multiplicative decomposition of deformation gradient. *Journal of Theoretical and Applied Mechanics*, 28-29:379–299, 2002.

[167] E. Walhorn, A. Kölke, B. Hübner, and Dinkler D. Fluid-structure coupling within a monolithic model involving free surface flows. *Computers & Structures*, 83:2100–2111, 2005.

[168] G. Wendt, P. Erbts, and A. Düster. Partitioned coupling strategies for multi-physically coupled radiative heat transfer problems. *Journal of Computational Physics*, 300:327–351, 2014.

[169] K. Wilmanski. *Continuum Thermodynamics*. World Scientific Publishing Company Pte Limited, 2008.

[170] C. Wood, A.J. Gil, O. Hassan, and J. Bonet. Partitioned block-Gauss-Seidel coupling for dynamic fluid-structure interaction. *Computers & Structures*, 88(23), 2008.

[171] P. Wriggers. *Nonlinear Finite-Element-Methods*. Springer-Verlag, 2009.

[172] H.C. Wu. *Continuum Mechanics and Plasticity*. Modern Mechanics and Mathematics. CRC Press, 2004.

[173] P. Wynn. Acceleration techniques for iterated vector and matrix problems. *Mathematics of Computation*, 16:301–322, 1962.

[174] N.N. Yanenko. *The Method of Fractional Steps: The Solution of Problems of Mathematical Physics in Several Variables*. Springer Berlin Heidelberg, 2012.

[175] J. Yang and M. F. Modest. Elliptic PDE formulation of general, three-dimensional high-order PN-approximations for radiative transfer. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 104(2):217–227, 2007.

[176] A. Zavaliangos, J. Zhang, M. Krammer, and J. R. Groza. Temperature evolution during field activated sintering. *Materials Science and Engineering: A*, 379(1):218–228, 2004.

[177] O. C. Zienkiewicz and R. Löhner. Accelerated relaxation or direct solution? Future prospects for FEM. *International Journal for Numerical Methods in Engineering*, 21(1):1–11, 1985.

[178] O.C. Zienkiewicz, D.K. Paul, and A.H. Chan. Unconditionally stable staggered solution procedure for soil-pore fluid interaction problems. *International Journal for Numerical Methods in Engineering*, 26:1039–1055, 1988.

[179] O.C. Zienkiewicz and R.L. Taylor. *The Finite Element Method – Solid Mechanics*, volume 2. Butterworth-Heinemann, 5th edition, 2000.

[180] O.C. Zienkiewicz and R.L. Taylor. *The Finite Element Method – The Basis*, volume 1. Butterworth-Heinemann, 5th edition, 2000.

[181] T.I. Zohdi. Modeling and simulation of a class of coupled thermo-chemo-mechanical processes in multiphase solids. *Computer Methods in Applied Mechanics and Engineering*, 193(6–8):679–699, 2004.

Curriculum vitae

Personal Data

Name, First Name	*Erbts, Patrick*
Date of birth	*11 October 1983*
Place of birth	*Buchholz i. d. Nordheide*
Nationality	*German*

Education

03/2016 - today	*Blade Development Engineer, Suzlon Energy GmbH*
05/2011 - 05/2015	*PhD student at Hamburg University of Technology*
04/2009 - 04/2011	*MSc in Mechanical Engineering at the Leibniz University of Hanover*
	Major: Mechanics
09/2005 - 03/2009	*Dipl.-Ing. (FH) (equivalent to BSc) in Aircraft Engineering at FH Aachen University of Applied Sciences*
	Major: Aircraft and Lightweight Structure Design
07/2004 - 07/2005	*Technical College G15, Hamburg*
09/2001 - 06/2004	*Technical apprenticeship as a Technical Draftsman, Hamburg*
	In cooperation with Airbus Germany in Hamburg
06/1997 - 06/2001	*Secondary School, Am Kattenberge, Buchholz*

Working Experience

05/2011 - 05/2015	*Research Associate at Hamburg University of Technology*
	Research project: Electro-thermo-mechanical modeling of Field Assisted Sintering Technology using high-order finite elements validated by experiments, founded by the German Research Foundation (DFG)
04/2009 - 10/2010	*Working student, Airbus Operations Germany, Hamburg*
09/2006 - 04/2007	*Student research assistant, FH Aachen*

Online-Buchshop für Ingenieure

Die Reihen der Fortschritt-Berichte VDI: