

TECHNISCHE UNIVERSITÄT MÜNCHEN

Lehrstuhl für Numerische Mechanik

Mortar Methods for Computational Contact Mechanics and General Interface Problems

Alexander Popp

Vollständiger Abdruck der von der Fakultät für Maschinenwesen der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

Vorsitzender: Univ.-Prof. Dr.-Ing. Klaus Drechsler

Prüfer der Dissertation:

1. Univ.-Prof. Dr.-Ing. Wolfgang A. Wall
2. Univ.-Prof. Dr.-Ing. Michael W. Gee
3. Prof. Tod A. Laursen, Ph.D.

Khalifa University, Abu Dhabi / UAE

Die Dissertation wurde am 28. Juni 2012 bei der Technischen Universität München eingereicht und durch die Fakultät für Maschinenwesen am 17. September 2012 angenommen.

Abstract

There exists a broad range of applications ranging from machine parts and tribology to crash-worthiness assessment and biomedical engineering, which all demand very general and powerful simulation tools for computational contact mechanics. Contact interaction of deformable bodies and its associated effects, for example frictional sliding, introduce significant additional complexities into the typical numerical modeling procedures of solid mechanics. This is mainly due to the inherent nonlinearity and lack of smoothness of the underlying physical principles. Several challenges have to be met on the way towards a truly general purpose simulation tool for computational contact mechanics, with the two most important aspects being given by robust space and time discretization as well as accurate and efficient constraint enforcement.

In this thesis, the applicability of mortar finite element methods for problems of computational contact mechanics is explored. In particular, a novel mortar approach for contact interaction in the fully nonlinear realm (i.e. including finite deformations and nonlinear material behavior) is presented, which draws its superior effectiveness from a sound mathematical foundation based on findings in the fields of domain decomposition and non-conforming discretization. Advantageous properties of mortar finite element methods include numerical stability with easily constructible discrete Lagrange multiplier spaces and preservation of the optimal spatial convergence rates of conforming discretizations, though with the added benefit of being able to deal with arbitrary non-matching interface meshes.

Within the present work, mortar methods are first investigated for mesh tying in nonlinear solid mechanics before considering the actual unilateral contact case. In this regard, an emphasis is put on the so-called dual Lagrange multiplier approach, where the discrete coupling variables are defined based on a biorthogonality relationship with the primary unknowns (i.e. displacements). As compared with standard Lagrange multiplier techniques, a localization of the interface coupling conditions is achieved, and thus the dual Lagrange multiplier approach significantly facilitates the resulting algorithms without impinging upon the optimality of mortar methods. The comprehensive framework presented here is the first one including feasible dual Lagrange multipliers for all types of first-order and second-order mortar finite elements.

The main contribution of this thesis, however, lies in the consistent extension of a mortar method with dual Lagrange multiplier interpolation to nonlinear unilateral contact scenarios. The first fully consistent linearization of the dual mortar approach in the context of implicit time integration is presented, from which very efficient nonlinear solution methods can be derived. Moreover, the inequality nature of contact constraints is accommodated with an enhancement of so-called primal-dual active set strategies or semi-smooth Newton methods, thus combining the search for the active contact constraints and all other sources of nonlinearities within one single iteration scheme. As a result, the accuracy of Lagrange multiplier methods and the robustness of mortar-based discretization are paired with an arguably unrivaled numerical efficiency drawn from the dual Lagrange multiplier concept and tailored nonlinear solution methods. A special emphasis is also put on creating a flexible simulation framework, which allows for the straightforward incorporation of closely related physical effects such as frictional sliding, abrasive wear and adhesion. Furthermore, all devised algorithms are designed to meet the requirements of high performance computing systems with regard to parallel efficiency and scalability.

Finally, the present work demonstrates that dual mortar methods also provide a convenient framework for many other single-field and multi-field problems of computational engineering

beyond solid and contact mechanics. Exemplarily, mortar finite element discretization in computational fluid dynamics, mortar-based interface treatment in fluid-structure interaction and a coupled fluid-structure-contact interaction approach are outlined. Altogether, the results obtained give rise to the hope that the numerical methods and algorithms developed in this thesis will not only prove helpful for the investigation of highly nonlinear contact phenomena, but also that they may lay the foundation for a general purpose analysis tool providing all capabilities of non-conforming discretization and interface coupling for complex multiphysics simulations.

Zusammenfassung

Es existiert eine große Bandbreite an Anwendungen, angefangen bei Maschinenelementen und tribologischen Systemen bis hin zur Beurteilung von Crashesicherheit oder biomedizinischen Fragestellungen, die allesamt sehr allgemeingültige und leistungsfähige Simulationswerkzeuge aus dem Bereich der numerischen Kontaktmechanik benötigen. Kontaktvorgänge deformierbarer Körper und weitere eng damit verknüpfte Effekte, wie beispielsweise das reibungsbehaftete Gleiten, bedeuten ein erhebliches Maß an zusätzlicher Komplexität für die typischerweise zum Einsatz kommenden numerischen Modellierungsverfahren der Strukturmechanik. Der Hauptgrund hierfür ist die inhärente Nichtlinearität und mangelnde Glattheit der zugrundeliegenden physikalischen Wirkprinzipien. Zur Entwicklung eines wirklich allgemeingültigen Simulationswerkzeugs für die numerische Kontaktmechanik müssen daher eine ganze Reihe von Herausforderungen bewältigt werden, wobei die beiden wichtigsten Aspekte in der robusten räumlichen und zeitlichen Diskretisierung sowie in der präzisen und effizienten Einbringung von Nebenbedingungen zu sehen sind.

Im Rahmen der vorliegenden Arbeit wird die Anwendbarkeit von Mortar-basierten Finite-Elemente-Methoden für Problemstellungen aus der numerischen Kontaktmechanik erforscht. Insbesondere wird ein neuartiger Mortar-Ansatz für vollständig nichtlineare Kontaktvorgänge (d.h. unter Berücksichtigung großer Deformationen und nichtlinearen Materialverhaltens) vorgestellt, der seine Leistungsfähigkeit aus einem tragfähigen mathematischen Fundament mit Erkenntnissen aus den Bereichen der Gebietszerlegungsmethoden sowie der nicht-konformen Diskretisierungsverfahren bezieht. Die vorteilhaften Eigenschaften von Mortar-basierten Finite-Elemente-Methoden umfassen deren numerische Stabilität für einfach zu definierende diskrete Lagrange-Multiplikator-Räume und die Erhaltung der optimalen räumlichen Konvergenzraten von konformen Diskretisierungsverfahren. Zusätzlich wird jedoch auch der Umgang mit beliebigen nicht-passenden Interface-Netzen ermöglicht.

Mortar-Methoden werden hier zunächst zur Netzkopplung (Mesh Tying) in der nichtlinearen Strukturmechanik untersucht, bevor dann der eigentliche Kontaktfall betrachtet wird. Dabei liegt ein Hauptaugenmerk auf dem sogenannten dualen Lagrange-Multiplikator-Ansatz, bei dem die diskreten Kopplungsvariablen basierend auf einer Biorthogonalitätsbedingung mit den primären Unbekannten (d.h. Verschiebungen) definiert sind. Im Vergleich zu Standard-Lagrange-Multiplikator-Ansätzen erreicht man eine lokale Beschränkung der Kopplungsbedingungen am Interface. Somit kann der duale Lagrange-Multiplikator-Ansatz eine erhebliche Vereinfachung der resultierenden Algorithmen bewirken ohne sich dabei negativ auf die Optimalität von Mortar-Methoden auszuwirken. Die hier präsentierte Methodik und Implementierung umfasst zum ersten Mal geeignete duale Lagrange-Multiplikator-Ansätze für sämtliche linearen und quadratischen Mortar-basierten Elementtypen.

Der wichtigste Beitrag der vorliegenden Dissertation liegt jedoch in der konsistenten Erweiterung einer Mortar-Methode mit dualen Lagrange-Multiplikatoren auf nichtlineare Kontaktvorgänge. Dabei wird zum ersten Mal eine vollständige Linearisierung des dualen Mortar-Ansatzes im Rahmen von impliziten Zeitintegrationsverfahren vorgestellt, mit deren Hilfe sehr effiziente nichtlineare Lösungsmethoden abgeleitet werden können. Des Weiteren wird dem Ungleichungs-Charakter der Kontaktbedingungen durch eine entsprechende Ergänzung von sogenannten primal-dualen Aktive-Mengen-Strategien oder nichtglatten Newton-Methoden Rechnung getragen, wodurch eine Kombination der Suche nach den aktiven Kontaktbedingungen

sowie aller anderer Quellen von Nichtlinearität innerhalb eines einzigen nichtlinearen Iterationschemas ermöglicht wird. Infolgedessen zeichnen sich die vorgestellten Ansätze neben der Lösungsgenauigkeit von Lagrange-Multiplikator-Methoden und der Robustheit Mortar-basierter Diskretisierung auch durch eine wohl einzigartige numerische Effizienz aus, die sich insbesondere aus dem Konzept der dualen Lagrange-Multiplikatoren sowie aus den für Kontaktvorgänge maßgeschneiderten nichtlinearen Lösungsmethoden ergibt. Ein besonderes Augenmerk wird außerdem auf die Flexibilität des entstandenen Simulationswerkzeugs gelegt, so dass eine Erweiterung auf andere kontaktspezifische Phänomene wie beispielsweise Reibung, abrasiven Verschleiß und Adhäsion vereinfacht wird. Darüber hinaus sind alle entwickelten Algorithmen so angelegt, dass sie den Anforderungen von Höchstleistungsrechnern bezüglich paralleler Ausführbarkeit und Skalierbarkeit genügen.

Abschließend wird in der vorliegenden Arbeit gezeigt, dass duale Mortar-Methoden neben der Struktur- und Kontaktmechanik auch einen geeigneten Rahmen zur Simulation zahlreicher weiterer Einfeld- und Mehrfeldprobleme in den computergestützten Ingenieurwissenschaften bieten. Exemplarisch werden die Grundzüge der Mortar-basierten Finite-Elemente-Diskretisierung in der numerischen Strömungsmechanik, die Interface-Behandlung mit Mortar-Ansätzen bei der Fluid-Struktur-Interaktion sowie ein gekoppelter Ansatz zur Simulation von Fluid-Struktur-Kontakt-Wechselwirkungen erläutert. Zusammenfassend besteht aufgrund der erzielten Ergebnisse Grund zur Hoffnung, dass sich die im Rahmen dieser Dissertation entwickelten Methoden und Algorithmen nicht nur für die Simulation nichtlinearer Kontaktphänomene als hilfreich erweisen, sondern dass darüber hinaus auch der Grundstein gelegt wurde für ein allgemeingültiges Simulationswerkzeug, das alle wichtigen Aspekte der nicht-konformen Diskretisierung sowie Interface-Kopplung für komplexe Mehrfeldprobleme bereitstellt.

Danksagung

Die vorliegende Arbeit entstand in den Jahren 2007 bis 2012 während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Lehrstuhl für Numerische Mechanik der Technischen Universität München (TUM). Während dieser Zeit wurde ich von vielen Personen begleitet, die entscheidend zum Erfolg der Arbeit beigetragen haben. An dieser Stelle möchte ich deshalb all diesen Personen meinen ganz herzlichen Dank aussprechen.

Der größte Dank gebührt meinem Doktorvater, Prof. Dr. Wolfgang A. Wall, für das entgegengebrachte Vertrauen, die fachliche und persönliche Unterstützung und ganz besonders auch für den großen Freiraum, den er mir von Anfang an bei meinen Aktivitäten am Lehrstuhl gewährt hat. All das war für mich und meine akademische Entwicklung von unschätzbarem Wert. Seine Art und Weise den Lehrstuhl für Numerische Mechanik zu führen und seine im positivsten Sinne ansteckende Begeisterung für die Wissenschaft haben mich sehr geprägt und werden mir immer als Vorbild dienen.

Ein weiteres großes Dankeschön möchte ich an Prof. Dr. Michael W. Gee vom Fachgebiet für Mechanik auf Höchstleistungsrechnern der TUM richten. Zum einen bedanke ich mich für die Übernahme des ersten Mitberichts, zum anderen für die geduldige und unermüdliche Unterstützung in allen Phasen meiner Promotion. Das Gefühl, bei kleineren und größeren Fragen zur Kontaktmechanik und zu Mortar-Methoden stets eine offene Bürotür und ein offenes Ohr vorzufinden, hat mir sehr geholfen und mich immer wieder aufs Neue motiviert.

Weiterhin bedanke ich mich bei Prof. Dr. Tod A. Laursen, Präsident der Khalifa University of Science, Technology and Research (KUSTAR) in Abu Dhabi, für die Übernahme des zweiten Mitberichts. Sein großes Interesse an meiner Arbeit ist mir eine ganz besondere Ehre. Für die Übernahme des Vorsitzes meiner Prüfungskommission danke ich Prof. Dr. Klaus Drechsler vom Lehrstuhl für Carbon Composites der TUM.

Ein besonderer Dank gilt auch Prof. Dr. Barbara I. Wohlmuth vom Lehrstuhl für Numerische Mathematik der TUM für die sowohl fachlich äußerst bereichernde als auch persönlich angenehme Zusammenarbeit in den vergangenen Jahren.

Bei den derzeitigen und ehemaligen Kollegen am Lehrstuhl für Numerische Mechanik möchte ich mich für viele spannende, fachliche (und nicht-fachliche) Diskussionsrunden, für die große Hilfsbereitschaft und ganz einfach für die vielen Freundschaften bedanken, die über die Jahre entstanden sind. Ich empfand es stets als außerordentliches Privileg, in einer herausragenden Arbeitsgruppe mit internationalem Renommee arbeiten zu dürfen, in der aber zugleich ein menschlich so positives Klima herrscht. Vor allem bei Dr. Moritz Frenzel, Dr. Ulrich Küttler, Dr. Lena Yoshihara, Dr. Thomas Klöppel, Dr. Christian Cyron, Markus Gitterle und Christoph Meier bedanke ich mich für die umfassende Unterstützung. Auch meinen zahlreichen Studenten danke ich für die tolle Zusammenarbeit und die wertvollen Beiträge, die sie in Form von Studienarbeiten und HiWi-Tätigkeiten zum Gelingen dieser Arbeit geleistet haben.

Schließlich danke ich von ganzem Herzen meiner Familie und meiner Freundin Julia für das in mich gesetzte Vertrauen und für so manchen Rat und Zuspruch in schwierigen Situationen. Ohne ihren Rückhalt wäre diese Arbeit undenkbar.

München, im Oktober 2012

Alexander Popp

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Fundamental approaches to contact simulations	2
1.3	Research objective	5
1.3.1	Specification of requirements	5
1.3.2	Proposal for a mortar finite element contact approach	7
1.4	Outline	8
2	Governing Equations and Finite Element Formulations	11
2.1	Solid mechanics	11
2.1.1	Kinematics	11
2.1.2	Stresses and constitutive laws	14
2.1.3	Balance equations	16
2.1.4	Initial boundary value problem	19
2.2	Contact mechanics	20
2.2.1	Contact kinematics	20
2.2.2	Tied contact constraints	23
2.2.3	Normal contact constraints	24
2.2.4	Frictional contact constraints	25
2.2.5	Further interface phenomena	26
2.3	Finite element formulations and solution schemes	27
2.3.1	From strong formulation to weak formulation	27
2.3.2	Space discretization	28
2.3.3	Time discretization	31
2.3.4	Spatial and temporal convergence	33
2.3.5	Linearization and solution techniques for nonlinear equations	33
2.3.6	Solution techniques for linear equations	35
2.4	Methods of constraint enforcement	36
2.4.1	Lagrange multiplier method	37
2.4.2	Penalty method	38
2.4.3	Augmented Lagrange method	39
2.4.4	Treatment of inequality constraints	40
2.5	Discretization techniques based on domain decomposition	43
2.5.1	General concepts of domain decomposition	43
2.5.2	Mortar finite element methods	44
2.5.3	Application to mesh tying in solid mechanics	45
2.5.4	Application to contact mechanics	45

3	Mortar Methods for Finite Deformation Solid Mechanics	47
3.1	Mesh tying problem statement	47
3.1.1	Strong formulation	48
3.1.2	Weak formulation	48
3.2	Discretization and numerical integration	50
3.2.1	Finite element discretization	50
3.2.2	Evaluation of mortar integrals in 2D	53
3.2.3	Evaluation of mortar integrals in 3D	56
3.2.4	Time discretization	61
3.2.5	Conservation laws	62
3.3	Discrete Lagrange multiplier spaces	64
3.3.1	Standard Lagrange multipliers	66
3.3.2	Dual Lagrange multipliers	69
3.3.3	Crosspoints and Dirichlet boundary conditions	80
3.4	Solution methods	81
3.4.1	Linearization and standard Newton algorithm	81
3.4.2	Standard Lagrange multiplier approach	82
3.4.3	Dual Lagrange multiplier approach	83
3.4.4	Penalty approach	84
3.5	Numerical examples	85
3.5.1	Consistency – Patch tests	86
3.5.2	Spatial convergence – Bending structure	89
3.5.3	Conservation properties – L-shaped block	92
4	Mortar Methods for Finite Deformation Contact Mechanics	95
4.1	Unilateral contact problem statement	95
4.1.1	Strong formulation	96
4.1.2	Weak formulation	97
4.2	Discretization and numerical integration	99
4.2.1	Finite element discretization	99
4.2.2	Evaluation and linearization of mortar integrals in 2D	102
4.2.3	Evaluation and linearization of mortar integrals in 3D	103
4.2.4	Evaluation and linearization of weighted gaps	104
4.2.5	Time discretization	105
4.2.6	Conservation laws	107
4.3	Discrete Lagrange multiplier spaces	109
4.3.1	Special requirements for contact mechanics	110
4.3.2	Linearization of dual Lagrange multipliers	111
4.3.3	Consistency of dual Lagrange multipliers	112
4.4	Active set strategy and semi-smooth Newton methods	115
4.4.1	General concepts of the primal-dual active set strategy	115
4.4.2	Application to frictionless contact	116
4.4.3	Extension to Coulomb friction	118
4.5	Solution methods	119
4.5.1	Linearization and semi-smooth Newton algorithm	120

4.5.2	Algebraic representation	123
4.5.3	Standard Lagrange multiplier approach	124
4.5.4	Dual Lagrange multiplier approach	125
4.5.5	Penalty approach	126
4.6	Aspects of implementation and parallel computing	129
4.6.1	Parallel redistribution and dynamic load balancing	129
4.6.2	Search algorithm for two body contact	132
4.6.3	Search algorithm for self contact and multiple bodies	135
4.7	Numerical examples	137
4.7.1	Consistency – Patch tests	137
4.7.2	Spatial convergence – Hertzian contact	139
4.7.3	Finite deformations – Ironing example	143
4.7.4	Coulomb friction – Sliding contact	145
4.7.5	Second-order interpolation – Torus impact	148
4.7.6	Conservation properties – Two rings	150
4.7.7	Self contact – Folding structure	154
4.7.8	Multiple bodies – Elastic particle system	155
4.7.9	Large-scale simulations – Two torus impact	157
5	Mortar Methods for Fluid Mechanics and Fluid-Structure Interaction	161
5.1	Subdomain coupling in fluid mechanics	161
5.1.1	Problem statement	162
5.1.2	Residual-based variational multiscale approach	164
5.1.3	Numerical example	166
5.2	Interface coupling in fluid-structure interaction	169
5.2.1	Problem statement	170
5.2.2	ALE-based moving-grid approach	173
5.2.3	Numerical example	173
5.3	Fluid-structure-contact interaction	176
5.3.1	Problem statement	178
5.3.2	XFEM-based fixed-grid approach	179
5.3.3	Numerical example	181
5.3.4	Extension towards elastohydrodynamic lubrication	182
6	Summary and Outlook	185
A	Details Concerning Consistent Linearization	189
A.1	Linearization of 2D mortar contact	189
A.2	Linearization of 3D mortar contact	195
	Bibliography	201

Nomenclature

Representation of scalars, tensors and other quantities

q, Q	Scalar quantity
\mathbf{q}	Vector
\mathbf{Q}	Second-order tensor
\mathcal{Q}	Higher-order tensor
\mathbf{q}	Discrete vector
\mathbf{Q}	Discrete matrix
\mathcal{Q}, \mathcal{Q}	Scalar or vector-valued function space

Operators and Symbols

$(\cdot)^T$	Transpose of a tensor
$(\cdot)^{-1}$	Inverse of a tensor or mapping
$(\cdot)^{-T}$	Transpose of the inverse of a tensor
$(\dot{\cdot}), (\ddot{\cdot})$	First and second time derivatives at a fixed reference position
$\hat{(\cdot)}$	Prescribed quantity
$\tilde{(\cdot)}$	Modified or transformed quantity
$\langle \cdot \rangle$	Macauley bracket
$\langle \cdot, \cdot \rangle_\Gamma$	Duality pairing on Γ
(\cdot, \cdot)	L^2 -inner product in Ω
$(\cdot, \cdot)_\Gamma$	L^2 -inner product on Γ
\mathbf{I}, \mathbf{I}	Identity tensor, identity matrix
det	Determinant
tr	Trace operator
log, ln	Logarithm, natural logarithm
Grad, ∇	Material gradient operator
Div, $\nabla \cdot$	Material divergence operator
div	Spatial divergence operator
Lin	Linearization operator
$\Delta(\cdot)$	Increment, change of (\cdot) in one time step
$\Delta(\cdot)$	Directional derivative of (\cdot) with respect to the discrete displacements \mathbf{d}
\otimes	Dyadic product
$\delta(\cdot)$	Virtual quantity
δ_{jk}	Kronecker delta

Superscripts and Subscripts

$(\cdot)^{(i)}$	Subdomain
$(\cdot)^{(e)}$, $(\cdot)^e$, $(\cdot)_e$	Element
$(\cdot)^{(1)}$, $(\cdot)_s$	Slave
$(\cdot)^{(2)}$, $(\cdot)_m$	Master
$(\cdot)_{(s,m)}$	Slave and master element pair
$(\cdot)_h$	Discretized in space with the FEM
$(\cdot)_g$, $(\cdot)_{gp}$	Gauss point
$(\cdot)_{int}$	Internal
$(\cdot)_{ext}$	External
$(\cdot)_{kin}$	Kinetic
$(\cdot)_{effdyn}$	Effective dynamic
$(\cdot)_{LM}$	Lagrange multiplier method
$(\cdot)_P$	Penalty method
$(\cdot)_{AL}$	Augmented Lagrange method
$(\cdot)_c$, $(\cdot)_{co}$	Contact
$(\cdot)_{mt}$	Mesh tying
$(\cdot)_\lambda$	Constraint
$(\cdot)_{seg}$	Segment
$(\cdot)_{cell}$	Integration cell

Domains and boundaries

Ω_0	Reference (material) configuration
Ω_t	Current (spatial) configuration
$\partial\Omega_0$	Boundary in reference configuration
$\partial\Omega_t$	Boundary in current configuration
Γ_u	Dirichlet partition of boundary in reference configuration
γ_u	Dirichlet partition of boundary in current configuration
Γ_σ	Neumann partition of boundary in reference configuration
γ_σ	Neumann partition of boundary in current configuration
Γ_c	Potential contact partition of boundary in reference configuration
γ_c	Potential contact partition of boundary in current configuration
Γ_a, Γ_i	Active and inactive contact zones in reference configuration
Ω	Spatial domain (for fluid mechanics and FSI)
Γ_D	Dirichlet partition of boundary (for fluid mechanics and FSI)
Γ_N	Neumann partition of boundary (for fluid mechanics and FSI)
Γ_I	Internal fluid interface (for fluid mechanics)
Γ	Fluid-structure interface (for FSI)
Γ_{FSI}	Fluid-structure interface (for FSCI)

Kinematics

e_1, e_2, e_3	Basis vectors of Cartesian coordinate system
\mathbf{X}	Position in reference configuration
\mathbf{x}	Position in current configuration
Φ_t	Mapping between reference and current configuration
\mathbf{u}	Displacement
$\dot{\mathbf{u}}$	Velocity
$\ddot{\mathbf{u}}$	Acceleration
\mathbf{F}	Deformation gradient
J	Jacobian determinant, determinant of \mathbf{F}
V_0, V	Reference and current volume
A_0, A	Reference and current surface area
\mathbf{N}, \mathbf{n}	Unit normal vector in reference and current configuration
$\mathbf{R}, \mathbf{U}, \mathbf{V}$	Rotation tensor, material and spatial stretch tensors
\mathbf{C}	Right Cauchy–Green deformation tensor
\mathbf{E}	Green–Lagrange strain tensor
\mathbf{L}	Material velocity gradient

Stresses and constitutive laws

\mathbf{t}	Traction vector in current configuration
$\boldsymbol{\sigma}$	Cauchy stress tensor
\mathbf{P}	First Piola–Kirchhoff stress tensor
\mathbf{S}	Second Piola–Kirchhoff stress tensor
\mathcal{P}_{int}	Internal power
Ψ	Strain energy function
\mathbf{C}_m	Constitutive tensor, material elasticity tensor
λ, μ	Lamé parameters
E	Young’s modulus
ν	Poisson’s ratio
$\boldsymbol{\epsilon}$	Linearized strain tensor
\mathbb{C}	Constitutive tensor for linear elasticity

Governing equations

t	Time
T	Total simulation time
m	Mass
ρ_0, ρ	Reference and current mass density
$\hat{\mathbf{b}}_0$	Body force in reference configuration
$\hat{\mathbf{t}}_0$	Prescribed pseudo-traction in reference configuration
$\hat{\mathbf{b}}$	Body force in current configuration

$\hat{\mathbf{t}}$	Prescribed traction in current configuration
$\hat{\mathbf{u}}_0$	Initial displacement at time $t = 0$
$\hat{\mathbf{u}}_0$	Initial velocity at time $t = 0$
\mathcal{W}	Work
\mathcal{P}	Power
\mathbf{w}	Weighting function vector
$\boldsymbol{\lambda}$	Lagrange multiplier vector

Contact mechanics

g_n	Gap function
\mathbf{g}	Gap vector
\mathbf{n}_c	Unit normal vector on contact surface
$\boldsymbol{\tau}_c^\xi, \boldsymbol{\tau}_c^\eta$	Unit tangent vectors on contact surface
χ	Contact mapping from slave to master surface
ξ, η	Local (convective) surface coordinates
$\mathbf{v}_{\tau, \text{rel}}$	Relative tangential velocity vector
\mathbf{t}_c	Contact traction vector in current configuration
p_n	Normal contact traction (pressure) in current configuration
\mathbf{t}_τ	Tangential contact traction in current configuration
\mathfrak{F}	Coefficient of friction

FE space discretization

nele	Number of elements
nnod	Number of nodes
ndim	Number of spatial dimensions
ndof	Number of degrees of freedom
N_k	FE shape function of node k
$\boldsymbol{\xi}$	Position in FE parameter space
\mathbf{X}	Discrete nodal positions in reference configuration
\mathbf{x}	Discrete nodal positions in current configuration
\mathbf{d}	Discrete nodal displacements
$\dot{\mathbf{d}}$	Discrete nodal velocities
$\ddot{\mathbf{d}}$	Discrete nodal accelerations
h	Characteristic element size
p	Polynomial degree of finite element interpolation
\mathbf{M}	Mass matrix
\mathbf{C}	Damping matrix
\mathbf{f}	Discrete force vector
\mathbf{K}_0	Initial tangent stiffness matrix
c_M, c_K	Parameters for Rayleigh damping

Time discretization

Δt	Time step size
n	Time step index
\mathbf{d}_n	Discrete nodal displacements at time t_n
\mathbf{v}_n	Discrete nodal velocities at time t_n
\mathbf{a}_n	Discrete nodal accelerations at time t_n
β, γ	Parameters of Newmark's method
α_f, α_m	Parameters of generalized- α method
ρ_∞	Spectral radius in the high-frequency limit

Solution techniques

i	Nonlinear iteration index
\mathbf{r}	Discrete residual vector
\mathbf{K}	Stiffness matrix
\mathbf{K}_T	Tangent stiffness matrix

Constraint enforcement

\mathbf{G}	Constraint vector
nco	Number of constraints
λ	Discrete Lagrange multiplier vector
\mathbf{C}	Constraint stiffness matrix
ϵ	Penalty parameter
k	Uzawa iteration index
\mathcal{A}	Active set of constraints
\mathcal{I}	Inactive set of constraints
c	Complementarity parameter

Mortar methods

$n^{(1)}$	Number of slave nodes
$n^{(2)}$	Number of master nodes
$m^{(1)}$	Number of slave nodes carrying discrete Lagrange multipliers
Φ_j	Lagrange multiplier shape function of node j
$N_k^{(1)}$	FE shape function of slave node k restricted to the contact surface
$N_l^{(2)}$	FE shape function of master node l restricted to the contact surface
$\xi^{(1)}, \xi^{(2)}$	Position in slave and master element parameter space
\mathbf{D}	First mortar coupling matrix
\mathbf{M}	Second mortar coupling matrix

\mathcal{S}	Set of all slave nodes or degrees of freedom
\mathcal{M}	Set of all master nodes or degrees of freedom
\mathcal{N}	Set of all nodes or degrees of freedom not related to \mathcal{S} and \mathcal{M}
\mathbf{B}_{mt}	Discrete mesh tying operator
\mathbf{g}_{mt}	Discrete mesh tying constraints
\mathbf{n}_k	Averaged unit normal vector at slave node k
$\boldsymbol{\tau}_k^\xi, \boldsymbol{\tau}_k^\eta$	Averaged unit tangent vectors at slave node k
n_k^{adj}	Number of slave elements adjacent to slave node k
n_s^e	Number of nodes of slave element e
n_m^e	Number of nodes of master element e
η	Position in segment parameter space (2D mortar coupling)
ξ_a, ξ_b	Segment end point coordinates (2D mortar coupling)
J_{seg}	Jacobian determinant of segment (2D mortar coupling)
J	Jacobian determinant of slave element
w_g	Weighting factor for Gauss point
$\mathbf{x}_0^{(1)}$	Position of slave element center (3D mortar coupling)
\mathbf{n}_0	Unit normal vector of slave element center (3D mortar coupling)
$\tilde{\mathbf{x}}$	Projected node position in auxiliary plane (3D mortar coupling)
n_{cell}	Number of integration cells (3D mortar coupling)
$\tilde{\boldsymbol{\eta}}$	Position in integration cell parameter space (3D mortar coupling)
$\tilde{\mathbf{x}}_v^{\text{cell}}$	Position of vertex v of an integration cell (3D mortar coupling)
J_{cell}	Jacobian determinant of integration cell (3D mortar coupling)
$\boldsymbol{\xi}^{\text{sub}}$	Position in subsegment parameter space (3D mortar coupling)
$\mathbf{f}^{(1)}, \mathbf{f}^{(2)}$	Sum of discrete slave and master force vectors
$\mathbf{m}^{(1)}, \mathbf{m}^{(2)}$	Sum of discrete slave and master moment vectors
\mathbf{P}	Discrete mortar projection operator
r	Reproduction order
$m_e^{(1)}$	Number of Lagrange multiplier nodes of slave element e
\mathbf{A}_e	Coefficient matrix for dual shape functions
$\mathbf{D}_e, \mathbf{M}_e$	Auxiliary coefficient matrices for dual shape functions
\mathbf{T}	Transformation matrix for quadratic dual shape functions
α	Transformation parameter for quadratic dual shape functions

Mortar methods for unilateral contact

\mathbf{B}_{co}	Discrete contact operator
$(\tilde{g}_n)_j$	Discrete weighted gap of slave node j
$(\tilde{\mathbf{v}}_{\tau, \text{rel}})_j$	Discrete relative tangential velocity of slave node j
e_{int}	Contact integration area of slave element e
\mathcal{A}	Set of active contact nodes
\mathcal{I}	Set of inactive contact nodes
C_j	Nonlinear complementarity function of slave node j in normal direction

c_n	Complementarity parameter in normal direction
\mathcal{G}	Set of slip nodes for frictional contact
\mathcal{H}	Set of stick nodes for frictional contact
\mathbf{D}_j	Nonlinear complementarity function of slave node j for Coulomb friction
c_τ	Complementarity parameter in tangential direction
$n^{(a)}$	Number of active nodes
\mathbf{K}_{co}	Contact stiffness matrix
$\tilde{\mathbf{K}}_{\text{effdyn,co}}$	Effective dynamic stiffness matrix including contact
$\tilde{\mathbf{r}}_{\text{effdyn,co}}$	Effective dynamic residual vector including contact
\mathbf{r}_{tot}	Total residual vector
$\mathbf{N}_S, \mathbf{N}_M$	Contact constraint matrices for non-penetration condition
$\mathbf{F}_S, \mathbf{T}_A$	Contact constraint matrices for frictionless sliding
$\mathbf{K}_{\text{co,P}}$	Contact stiffness matrix for penalty approach
n_{proc}	Number of processes / processors
r	Load balance limit for parallel redistribution
T^{\min}, T^{\max}	Minimum and maximum computation time of one individual processor
N, M	Number of slave and master elements

Fluid mechanics

\mathbf{u}	Fluid velocity
p	Fluid pressure
μ	Dynamic viscosity
ν	Kinematic viscosity
ϵ	Strain rate tensor
$\hat{\mathbf{f}}$	Fluid body force vector per unit volume
$\hat{\mathbf{h}}$	Prescribed fluid traction vector per unit surface area
$\hat{\mathbf{u}}_0$	Initial fluid velocity at time $t = 0$
\mathbf{v}	Fluid velocity weighting function
q	Fluid pressure weighting function
$\boldsymbol{\mu}$	Lagrange multiplier weighting function on Γ_1
c_L, c_D	Lift and drag coefficients

Variational multiscale approach

$\hat{\mathbf{u}}, \hat{p}$	Unresolved scales of fluid velocity and pressure
τ_C, τ_M	Fluid stabilization parameters
$\mathcal{R}_{C,h}$	Discrete residual of continuity equation
$\mathcal{R}_{M,h}$	Discrete residual of momentum equation
$\boldsymbol{\lambda}_u$	Lagrange multiplier vector for fluid velocity \mathbf{u} on Γ_1
λ_p	Lagrange multiplier vector for fluid pressure p on Γ_1
$\boldsymbol{\mu}_u, \mu_p$	Lagrange multiplier weighting functions on Γ_1

Fluid-structure interaction

$(\cdot)^S$	Solid field
$(\cdot)^F$	Fluid field
$(\cdot)^G$	ALE field
Γ	Fluid-structure interface
\mathbf{d}^G	ALE grid displacements
\mathbf{u}^G	ALE grid velocity
\mathbf{c}	ALE convective velocity
\mathbf{d}^S	Solid displacements
$\hat{\mathbf{f}}^S$	Solid body force vector per unit volume
$\hat{\mathbf{h}}^S$	Prescribed solid traction vector per unit surface area

Fluid-structure-contact interaction

Ω^+	Physical fluid domain
Ω^-	Fictitious fluid domain
Γ^+	Fluid side of fluid-structure interface
$\bar{\boldsymbol{\sigma}}$	Additional stress tensor for embedded Dirichlet conditions
$\bar{\gamma}$	Weighting functions associated with additional stress tensor
$\bar{\boldsymbol{\epsilon}}$	Strain rate tensor associated with additional stress tensor
Ψ	Enrichment function (XFEM)

Abbreviations

AABB	Axis-aligned bounding box
ALE	Arbitrary Lagrangian Eulerian
AMG	Algebraic multigrid
C	Conforming
CG	Conjugate gradient
CM	Conforming with mortar method
CPP	Closest point projection
CPU	Central processing unit
DD	Domain decomposition
DEM	Discrete element method
DNS	Direct numerical simulation
DOF	Degree of freedom
DOP	Discretized orientation polytope
EHL	Elastohydrodynamic lubrication
EMM	Energy-momentum method
FE	Finite element
FEM	Finite element method

FSCI	Fluid-structure-contact interaction
FSI	Fluid-structure interaction
GEMM	Generalized energy-momentum method
GMRES	Generalized minimal residual
GP	Gauss point
HSM	Hertz–Signorini–Moreau
HPC	High performance computing
IBVP	Initial boundary value problem
ILU	Incomplete lower upper triangular matrix
LBB	Ladyzhenskaya–Babuška–Brezzi
LM	Lagrange multiplier
KKT	Karush–Kuhn–Tucker
NCP	Nonlinear complementarity
NC	Non-conforming
NM	Non-matching
NTS	Node-to-segment
NURBS	Non-uniform rational B-splines
ODE	Ordinary differential equation
PDASS	Primal-dual active set strategy
PDE	Partial differential equation
PMTPE	Principle of minimum of total potential energy
PSPG	Pressure stabilizing Petrov–Galerkin
PTC	Pseudo-transient continuation
PVW	Principle of virtual work
SIMPLE	Semi-implicit method for pressure-linked equations
STS	Segment-to-segment
SUPG	Streamline upwind Petrov–Galerkin
VUM	Velocity update method
XFEM	Extended finite element method

1 Introduction

1.1 Motivation

Contact phenomena are virtually omnipresent in nature and biological systems. The associated length and time scales cover the entire spectrum from the nanoscale to the macroscopic level and from hypervelocity impact to quasi-static contact interaction, respectively. For example, the plate tectonics process of the continental drift, the simple motion sequence when walking or the flow of red blood cells (erythrocytes) through blood vessels are all representatives of processes largely dominated by contact and associated physical effects. Beyond that, science and engineering have exploited the principles of contact mechanics to develop processes, such as deep-drawing or extrusion-molding, as well as technical systems and machine parts, including car tires, fluid bearings, gears, shafts and splines or elastomeric seals.

Contact mechanics can be looked at from several different perspectives. For some scenarios, e.g. in nanotribology, it is helpful or even mandatory to investigate contact interaction at an atomistic level. For most contact applications, however, a purely macroscopic viewpoint based on classical continuum assumptions is sufficient. Throughout this thesis, a continuum approach will be followed, mainly considering contact mechanics as a particularly challenging subclass of solid and structural mechanics. The geometrical constraint of non-penetration of different solid bodies can then easily be identified as the most important underlying principle of contact interaction. In addition, the overall contact phenomenon is commonly also influenced by one or several closely related interface effects, for example sticking and sliding friction, adhesion, elastohydrodynamic lubrication and wear. Altogether, contact and its associated phenomena introduce strong additional nonlinearities into solid mechanics problems, where contact itself can basically be interpreted as a set of complex boundary conditions, possibly changing over time. Together with the already typical nonlinearities inherent in general solid mechanics, i.e. large deformations and nonlinear constitutive (material) behavior, this evinces the challenges and difficulties of mathematically describing and solving contact interactions, even if the given problem setup is quite simple. Due to this complexity, only very few contact problem settings exist, where analytical solution techniques are actually applicable. The early work conducted by Hertz [70] on pressure distributions between contacting elastic bodies more than a century ago, is commonly considered to be the origin of modern contact analysis. A comprehensive overview of the basic principles of contact mechanics, together with the most important analytical solution techniques can be found in the textbooks by Johnson [91] and Timoshenko and Goodier [161].

With general contact problems being hardly accessible for mathematical analysis, experimental procedures and numerical modeling are naturally becoming the focus of attention. Physical experiments are a convenient way of gaining information about certain aspects of contact mechanics, e.g. for determining coefficients of friction related to different material pairings. However, for the majority of contact scenarios, the applicability of experimental procedures is either

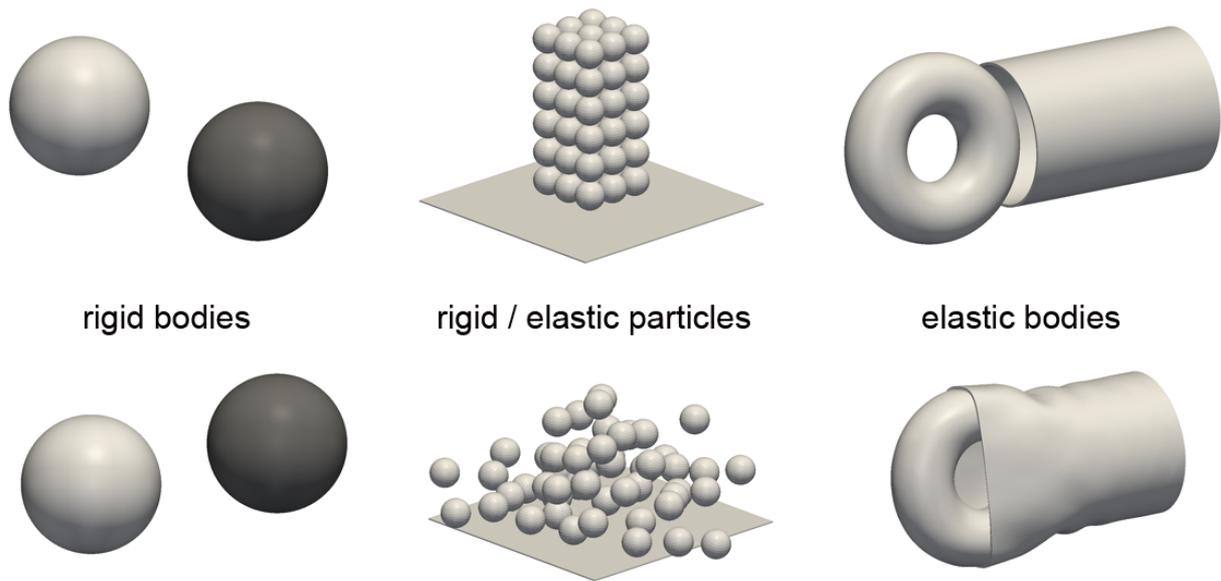


Figure 1.1: A possible classification of mechanical systems involving contact interaction.

limited or practically impossible. As a prominent example, experimental crashworthiness assessment, in accordance with safety regulations and consumer protection tests, causes considerable costs in the automotive industry. Complex contact phenomena in patient-specific surgery planning or during the design of medical devices, e.g. guaranteeing the optimal placement and minimum leakage of arterial stents, do not even allow for meaningful experimental tests at all. Thus, combining the aforementioned exemplary arguments, it becomes obvious that there is a very high and ever-growing demand for powerful numerical modeling and simulation techniques in the field of contact mechanics. What makes improved contact simulation approaches even more promising and likely to generate significant impact is the fact that the resulting numerical algorithms can typically be employed for a very broad range of scientific and technical interests. In fundamental physical, chemical or biological research, as well as in the applied sciences, novel methods and tools of computational contact mechanics allow for a better understanding of complex systems, which are influenced by contact phenomena. On the other hand, many aspects of engineering practice and product development (e.g. minimizing the frictional loss in gear transmissions, optimizing the structural integrity of car bodies in crash situations) also heavily benefit from improvements in contact modeling and simulation.

One possible, very general classification of contact problems is illustrated in Figure 1.1. The focus of interest throughout this thesis will be on contact interaction in nonlinear elastodynamics and to some extent also on agglomerations of particles, but not on rigid body dynamics.

1.2 Fundamental approaches to contact simulations

All ideas and methods of computational contact mechanics will be exclusively discussed in the context of the finite element method (FEM) throughout this thesis. Since the 1960s, the FEM has gradually evolved as the dominating numerical approximation technique for the solution of

partial differential equations (PDEs) in various fields, especially solid and structural mechanics including contact mechanics, but also in fluid mechanics, thermodynamics and for the treatment of coupled problems. The general FEM literature is abundant, exemplarily the interested reader is referred to the monographs by Bathe [6], Belytschko et al. [10], Hughes [86], Reddy [139], Zienkiewicz et al. [197] and Zienkiewicz and Taylor [196]. Other approaches for the numerical simulation of contact mechanics are only mentioned very briefly here for the sake of completeness. Multibody dynamics are a fitting tool when analyzing contact and impact phenomena of rigid bodies (see Figure 1.1, left), with possible extensions to elastic multibody dynamics allowing for a certain degree of deformation of the contacting bodies. Moreover, particle methods such as the discrete element method (DEM) are frequently used for investigating granular and particulate materials (see Figure 1.1, middle), whose mechanical behavior is largely dominated by contact interaction. While finite elements would not be the method of choice for such applications, this thesis is mainly related to contact of elastic solid bodies (e.g. thin-walled structures, see Figure 1.1, right), possibly including very large deformations. In this context, the FEM undoubtedly provides a very convenient framework for numerical modeling and simulation. Furthermore, there is an increasing interest in the interplay of contact mechanics with other physical phenomena, such as thermomechanics, fluid-structure interaction (FSI) and the lubrication behavior of thin fluid films, where finite elements are also an eligible approach, e.g. due to their generality and geometrical flexibility.

First contributions to the treatment of contact mechanics within the FEM can be traced back to the 1970s and 1980s. In Francavilla and Zienkiewicz [44] and Hughes et al. [87], contact conditions are formulated based on a very simple, purely node-based approach, which requires node-matching finite element meshes at the contact interface and is restricted to small deformations. Subsequently, a different idea was expedited, typically denoted as node-to-surface or node-to-segment (NTS) approach and characterized by a discrete, point-wise enforcement of the non-penetration condition at the finite element nodes. This NTS approach could readily be applied to the case of finite deformations and large sliding motions, therefore soon becoming the standard procedure in computational contact mechanics. Without claiming that the following listing is exhaustive, the reader is referred to Bathe and Chaudhary [7], Benson and Hallquist [15], Hallquist et al. [64], Laursen [106], Laursen and Simo [110], Simo and Laursen [149] and Wriggers et al. [184] for a comprehensive overview. An important basis for the methods to be proposed in this thesis is formed by the first investigations on the so-called segment-to-segment (STS) approach in Papadopoulos and Taylor [121] and Simo et al. [151]. In contrast to the purely point-wise procedure typical of NTS methods, the STS approach is based on a thorough sub-division of the contact surface into individual segments for numerical integration together with an independent approximation of the contact pressure. Thereby, the STS approach can be interpreted as precursor of mortar finite element methods for computational contact mechanics, which will be the main topic of this thesis.

Before reviewing the literature on mortar methods, however, an overview of other important aspects of computational contact mechanics aside from the discretization approach (NTS, STS, mortar) is given. One main focus of attention has been set on different procedures for the enforcement of contact constraints, with the most prominent representatives being penalty methods, Lagrange multiplier methods and Augmented Lagrange methods, see Alart and Curnier [3] for an excellent overview and discussion. Further questions related to contact modeling within a finite element framework comprise efficient search algorithms (cf. e.g. Williams and O'Connor

[174]), mesh adaptivity (cf. e.g. Carstensen et al. [21], Hübner and Wohlmuth [82], Wriggers and Scherf [182]), covariant surface description (cf. e.g. Laursen and Simo [110], Schweizerhof and Konyukhov [145]), surface smoothing (cf. e.g. Puso and Laursen [130], Wriggers et al. [185]), the treatment of contact on enriched and embedded interfaces (cf. e.g. Laursen et al. [111]), modeling of interface effects other than friction (cf. e.g. Sauer [143], Yang and Laursen [188]), beam contact (cf. e.g. Wriggers and Zavarise [183], Zavarise and Wriggers [193]) and energy conservation in the context of contact dynamics (cf. e.g. Hager et al. [63], Hesch and Betsch [71], Laursen and Chawla [108], Laursen and Love [109]), among others. Apart from numerous original papers, a comprehensive introduction to most of these topics can be found in the textbooks by Laursen [107] and Wriggers [180].

Nevertheless, novel robust discretization techniques for finite deformation contact problems, and especially mortar finite elements adapted for this purpose, have arguably received most attention in the field of computational contact mechanics in recent years. Mortar methods, which were originally introduced as an abstract domain decomposition technique (cf. e.g. Ben Belgacem [11], Bernardi et al. [16], Seshaiyer and Suri [147]), are characterized by an imposition of the occurring interface constraints in a weak sense and by the possibility to prove their mathematical optimality. In the context of contact analysis, this allows for a variationally consistent treatment of non-penetration and frictional sliding conditions despite the inevitably non-matching interface meshes for finite deformations and large sliding motions. Early applications of mortar finite element methods for contact mechanics can, for example, be found in Ben Belgacem et al. [13], Hild [75] and McDevitt and Laursen [116], though limited to small deformations. Gradually, restrictions of mortar-based contact formulations with respect to nonlinear kinematics have been removed, leading to the implementations given in Fischer and Wriggers [39, 40], Hesch and Betsch [71], Puso and Laursen [132, 133], Tur et al. [163] and Hesch and Betsch [74].

An alternative choice for the discrete Lagrange multiplier space, so-called dual Lagrange multipliers, was proposed in Wohlmuth [175, 176] and, in contrast to the standard mortar approach, generates interface coupling conditions that are much easier to realize without impinging upon the optimality of the method. Applications of this approach to small deformation contact problems can be found in Brunssen et al. [20], Flemisch and Wohlmuth [41], Hübner and Wohlmuth [80] and Hübner et al. [84], and some first steps towards finite deformations have been undertaken in Hartmann [65] and Hartmann et al. [66]. Another interesting feature of dual Lagrange multiplier interpolation is that it naturally fits together with so-called primal-dual active set strategies for constraint enforcement. It is well-known from the mathematical literature on constrained optimization problems and also from applications in computational contact mechanics, that primal-dual active set strategies can equivalently be interpreted as semi-smooth Newton methods (cf. e.g. Alart and Curnier [3], Christensen [22], Christensen et al. [23], Hintermüller et al. [77], Qi and Sun [135]), thus allowing for the design of very efficient global solution algorithms, especially in the context of nonlinear material behavior and finite deformations. While a few different discretization approaches have been suggested (see e.g. the contact domain method proposed in Hartmann et al. [67] and Oliver et al. [119]), and while NTS methods are still very popular in engineering practice, mortar-based contact formulations have become quite well-established in the meantime and can arguably be seen as state-of-the-art method for computational contact mechanics. However, there still remains a multitude of unresolved questions, especially in the context of the dual Lagrange multiplier version of the mortar approach and its application to fully nonlinear contact problems, which marks the point of origin for this thesis.

1.3 Research objective

The methods developed in this thesis and to be outlined in the following are aiming at very general, fully nonlinear contact problems, for which mortar finite element methods feature very desirable properties. These properties involve a sound mathematical foundation based upon mixed variational formulations, easy-to-construct Lagrange multiplier interpolations satisfying the necessary inf-sup conditions and mathematical optimality in the sense that the discretization error is bounded by the sum of the individual subdomain errors, see e.g. Wohlmuth [176]. Dual Lagrange multiplier spaces, in particular, possess a huge potential for mortar-based contact algorithms due to their inherent facilitation of the coupling constraints and their natural compatibility with efficient primal-dual active set strategies. However, a truly general and robust mortar-based contact formulation for finite deformations using dual Lagrange multipliers is still missing.

1.3.1 Specification of requirements

Based upon the previous explanations, the most important requirements for the development of a general mortar finite element approach with dual Lagrange multipliers in the context of fully nonlinear contact dynamics are listed and illustrated in the following.

Condensation of discrete Lagrange multipliers Depending on the size of the contact interface, mortar finite element methods based on standard Lagrange multiplier interpolation lead to an undesirable increase in the size of the global linear systems of equations. Moreover, the resulting algebraic representation has a typical saddle point structure, which may generate difficulties for the performance of iterative solvers and preconditioners. These aspects are rightly considered as serious drawbacks of Lagrange multiplier methods, and thus are commonly cited as an example why a penalty-based approach is more favorable for contact simulations. However, dual Lagrange multipliers offer a very convenient way to circumvent the disadvantages of mortar methods based on standard Lagrange multipliers, while at the same time not introducing penalty regularization. Based on a specific biorthogonality condition (see e.g. Flemisch and Wohlmuth [41] and Wohlmuth [175, 176]), the dual Lagrange multiplier variant allows for an easy elimination of the discrete Lagrange multiplier degrees of freedom by condensation. This property makes dual Lagrange multipliers the preferred discretization approach for the mortar-based contact formulations presented throughout this thesis.

First-order and second-order finite element interpolation To meet the requirement of universality, the complete range of first-order and second-order finite element interpolation in both 2D and 3D has to be considered. Besides an accurate algorithmic realization of the numerical integration of mortar coupling terms at the interfaces, this implies that adequate discrete Lagrange multiplier spaces need to be defined for all mentioned cases. In the context of dual Lagrange multipliers, especially an extension of the biorthogonality condition to second-order interpolation is quite difficult to achieve, see e.g. Lamichhane [103], Lamichhane et al. [105] and Lamichhane and Wohlmuth [104] for related investigations, however restricted to the classical domain decomposition case. Combining higher-order duality with the special demands of unilateral contact constraints is a hitherto completely unanswered question.

Fully consistent linearization Contact interaction is considered in the context of nonlinear finite element methods and implicit time integration schemes throughout this thesis. Thus, the proposed mortar-based contact formulation has to be consistently linearized to achieve the desired quadratic convergence rates within typical nonlinear solution methods of Newton–Raphson type. Apart from the usual nonlinearities stemming from the constitutive behavior (material nonlinearity) and finite deformations (geometrical nonlinearity), the search for the correct active contact set and stick or slip regions in the case of frictional contact introduce additional complexities here. While consistent linearization is, of course, well-known to be a prerequisite for efficient nonlinear solution algorithms, this topic has only rudimentarily been dealt with in the framework of dual Lagrange multiplier interpolation for finite deformation contact so far, see e.g. Hartmann [65] and Hartmann et al. [66]. In contrast to the established penalty-based treatment of standard mortar methods (cf. e.g. Puso and Laursen [132, 133]), a fully consistent linearization of dual mortar contact and the associated semi-smooth Newton solution strategies still needs further exploration.

Exact constraint enforcement strategy In general, any finite element representation of a given contact problem will inevitably introduce a certain discretization error. However, in order to achieve the best possible contact solution accuracy for this particular discretization, an exact satisfaction of the discrete contact constraints is required for the methods developed in this thesis. This precludes the application of simple penalty methods, which need an unphysical, user-defined penalty parameter affecting the accuracy of the numerical solution. An Augmented Lagrange version based on Uzawa’s algorithm (see e.g. Laursen [107]) removes this shortcoming, however at the additional computational cost of introducing nested nonlinear iteration loops. Thus, the focus will be set on efficient active set strategies for contact constraint enforcement here, which allow for a beneficial interpretation as semi-smooth Newton methods as e.g. described in Alart and Curnier [3], Christensen et al. [23] and Hintermüller et al. [77]. While not yet having been applied to mortar-based formulations for finite deformation contact, semi-smooth Newton methods seem particularly promising for such applications, allowing for an efficient treatment of all occurrent nonlinearities including the active set decisions for contact and friction within one single nonlinear iteration scheme.

Suitability for high performance parallel computing The developments in this thesis are aiming at challenging large-scale contact simulations, as exemplified for instance by rolling contact of car tires or by biomechanical and biomedical applications. Corresponding finite element models typically contain several million degrees of freedom, and may require an implicit solution within hundreds of time increments. Thus, the integration of all proposed methods into a high performance computing (HPC) framework is absolutely mandatory, as only that way the necessary parallel scalability of large contact simulations can be achieved. Special focus will be set on efficient parallel algorithms throughout this thesis, for instance with regards to contact search, dynamic load balancing in a multiprocessor environment and the applicability of state-of-the-art iterative linear solvers and associated preconditioning techniques.

Flexibility and extensibility of the mortar approach While the fundamentals of computational contact mechanics doubtlessly are the main focus of attention here, another requirement is to achieve a certain flexibility of the developed mortar finite element methods, and especially

the possibility to extend them to other single-field or multi-field problems. Within this thesis, the high flexibility of the dual mortar approach is to be demonstrated in two steps. Firstly, certain model extensions within contact mechanics, namely the inclusion of frictional sliding according to Coulomb's law and the treatment of self contact are investigated. Secondly, the implementation has to be done in a way that allows for transferring the proposed methods and ideas to other physical fields (e.g. fluid mechanics) and coupled problems (e.g. fluid-structure interaction).

1.3.2 Proposal for a mortar finite element contact approach

This thesis describes a novel mortar finite element approach for finite deformation contact problems in 2D and 3D using dual Lagrange multipliers, addressing all of the aforementioned requirements. To the author's knowledge, this distinguishes the proposed contact formulation from all existing approaches available in the literature to date. Most often, standard Lagrange multiplier interpolation in combination with a penalty or Augmented Lagrange (Uzawa) treatment of the contact constraints is employed, which may undoubtedly yield robust and efficient numerical algorithms, however with the well-known deficiencies of such constraint enforcement techniques. The most important ingredients and new scientific contributions of the presented approach are given in the following:

- the first successful implementation of a dual mortar formulation for finite deformation contact with consistent linearization in 2D and 3D, see also Popp et al. [123, 124, 125].
- an extension of the idea of primal-dual active set strategies and semi-smooth Newton methods to the fully nonlinear realm, see also Gitterle et al. [55], Popp et al. [123, 124, 125].
- the derivation of feasible dual Lagrange multiplier spaces for second-order interpolation in 3D, meeting the special requirements of unilateral contact as compared with classical domain decomposition, see also Popp et al. [128], Wohlmuth et al. [179].
- successful applications of the developed dual mortar scheme to single-field and multi-field problems other than pure computational contact mechanics, e.g. fluid dynamics and FSI, see also Ehrl et al. [35], Klöppel et al. [96], Mayer et al. [115].

Summing up, the methods proposed in this thesis bring together existing ideas in the fields of computational contact mechanics and mortar finite element methods and extend them towards a comprehensive and more accurate and efficient treatment of finite deformation contact problems than possible to date, thus for the first time tapping the full potential of dual Lagrange multiplier interpolation for such scenarios.

All methods and models devised as part of this work are implemented in the in-house finite element software package BACI (cf. Wall and Gee [171]), developed at the Institute for Computational Mechanics at Technische Universität München. This multi-purpose parallel research code is written in C++ and integrates open-source libraries of the Trilinos Project conducted by Sandia National Laboratories (cf. Heroux et al. [69]). Consequently, existing features such as common finite element element formulations, time integration schemes or iterative solution techniques are reused within this thesis and supplemented with various algorithmic extensions as well as new code modules.

1.4 Outline

The methods and algorithms presented here are organized with increasing complexity towards the main focus of attention, which is given by challenging applications of computational contact mechanics and by the inclusion of mortar methods as discretization approach for various single-field and coupled problems. Thus, the remainder of this thesis is organized as follows.

In **Chapter 2**, the relevant governing equations of nonlinear solid mechanics and contact mechanics are outlined. In addition, the basic concepts of weak formulations, finite element discretization, time integration and nonlinear and linear solution techniques are reviewed in a very general style. Typical methods of constraint enforcement and domain decomposition, especially mortar finite element methods, are explained in more detail due to the decisive role they play within the scope of this thesis.

Before considering the actual unilateral contact case, **Chapter 3** is devoted to an introduction to the imposition of interface coupling conditions using mortar methods in a classical domain decomposition framework. Exemplarily, the mortar-based mesh tying of dissimilar triangulations for nonlinear solid mechanics applications has been chosen for presentation, as this allows for temporarily putting aside some of the complexities associated with contact modeling. Nevertheless, many important aspects of mortar methods including several new findings for dual Lagrange multipliers can already be explained with the help of this simplified framework. The numerical integration procedure for the discrete mortar coupling terms in 2D and 3D is outlined, and suitable (dual) Lagrange multiplier interpolations are presented for all relevant cases. Finally, algebraic representations of the resulting linearized systems of equations are derived and the proposed methods are validated with several numerical examples.

Chapter 4 extends the aforementioned ideas to finite deformation contact mechanics, which is the main research direction of this thesis. Building upon a detailed problem formulation including the contact typical inequality constraints, the issues of mortar finite element discretization and dual Lagrange multiplier interpolation are taken up again, where the focus then is on additional requirements and enhancements as compared with the mesh tying case considered before. Furthermore, the question of how to enforce the unilateral contact constraints is addressed with a solution algorithm based on semi-smooths for finding the correct active set. Similar to **Chapter 3**, the final linearized systems of equations are derived for both dual Lagrange multiplier interpolation and competing approaches. Several aspects of parallel computing and efficient implementation are highlighted, before important properties of the developed methods are finally tested with various numerical examples ranging from simple benchmarks to large-scale contact simulations. Closely connected to this part of the thesis is **Appendix A**, which comprises all implementational details concerning the consistent linearization of mortar contact formulations in both two and three dimensions.

Further successful applications of mortar methods with dual Lagrange multipliers beyond pure solid and contact mechanics are outlined in **Chapter 5**. Exemplarily, mortar-based domain decomposition in computational fluid dynamics, coupling of non-matching interface meshes in the context of moving-grid fluid-structure interaction and the integration of the proposed contact algorithm into a fixed-grid FSI framework (for applications such as wet contact and elastohydrodynamic lubrication) are sketched. The mentioned applications demonstrate the versatility of the methods proposed in this thesis, thus already serving as a form of an early outlook towards promising future research directions.

Finally, the conclusion and outlook in **Chapter 6** summarize the most important results and accomplishments, but also point out which aspects of mortar finite element methods and computational contact mechanics still have room for improvement. Several extensions are suggested to further enhance the proposed formulations and numerical algorithms, so that dual mortar methods can be used for contact and multiphysics simulations even more efficiently.

2 Governing Equations and Finite Element Formulations

In this chapter, the basic concepts of nonlinear continuum mechanics are reviewed with a focus on the governing equations for solid dynamics and contact interaction required later. In addition, an overview of discretization techniques and especially finite element formulations along with the associated numerical solution methods is given. As contact and impact can be classified as constrained solid mechanics problems, the most important methods of constraint enforcement within the FEM are highlighted. Moreover, the fundamental concepts of domain decomposition and mortar methods are explained. Altogether, the aforementioned aspects are intended to provide an appropriate theoretical basis for the following chapters of this thesis.

2.1 Solid mechanics

A brief introduction to nonlinear solid dynamics serves as starting point for all further investigations. These remarks are not intended to give an exhaustive overview of the topic, but are rather geared towards outlining the necessary basics for contact mechanics. Throughout this thesis, a classical continuum perspective will be taken and structural mechanics models derived from special kinematic assumptions, such as beams, plates or shells, are not considered here. For more extensive reviews in the field of solid and structural dynamics, the reader is referred to the corresponding literature, e.g., Bonet and Wood [17], Gurtin [60], Holzapfel [78], Marsden and Hughes [113], Ogden [118] and Simo and Hughes [148].

2.1.1 Kinematics

In this section, the fundamental kinematic relationships describing the deformation of a homogeneous body are presented, see also Figure 2.1. The classical (Boltzmann) continuum model in a three-dimensional Euclidean space description is assumed. However, the derived concepts hold similarly in two dimensions. A common Cartesian coordinate system is considered here for all configurations, also if several bodies are involved, while the concept of curvilinear coordinates only becomes important for local surface descriptions within the finite element discretization. Two distinct observer frames are defined: the reference configuration $\Omega_0 \subset \mathbb{R}^3$ denotes the domain occupied by all material points \mathbf{X} at time $t = 0$, while the current configuration $\Omega_t \subset \mathbb{R}^3$ describes the changed positions \mathbf{x} at a certain time t . The motion and deformation from reference to current configuration are tracked with the bijective nonlinear deformation map

$$\Phi_t : \begin{cases} \Omega_0 \rightarrow \Omega_t \\ \mathbf{X} \mapsto \mathbf{x} \end{cases}, \quad (2.1)$$

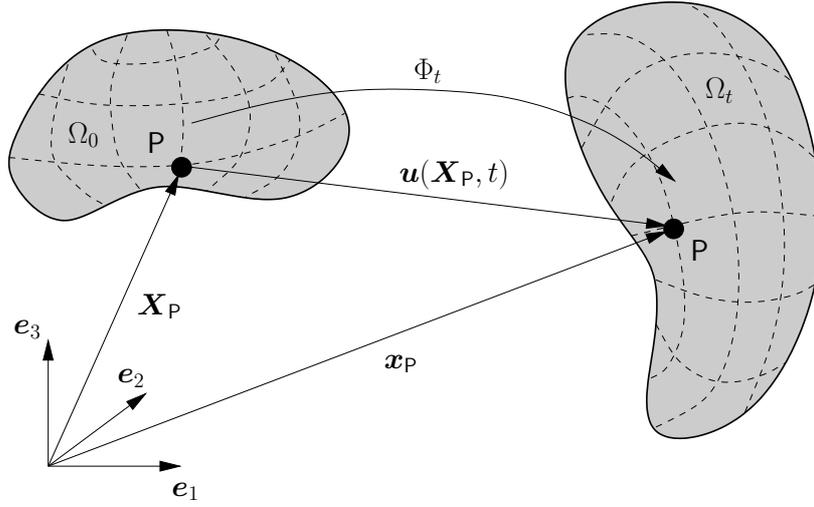


Figure 2.1: Cartesian coordinate system, reference configuration and current configuration for a total Lagrangian description of motion.

which also allows for the notations $\mathbf{x} = \Phi_t(\mathbf{X}, t)$ and $\mathbf{X} = \Phi_t^{-1}(\mathbf{x}, t)$. The absolute displacement of a material point (see again Figure 2.1) is then described as

$$\mathbf{u}(\mathbf{X}, t) = \mathbf{x}(\mathbf{X}, t) - \mathbf{X}. \quad (2.2)$$

Within the total Lagrangian approach employed throughout this thesis, kinematic relations and all derived quantities are described with respect to the material points in the reference configuration Ω_0 . Thus, the material point position \mathbf{X} plays the role of an independent variable for the problem formulation, while the primary unknown to be solved for is the time-dependent deformation map $\Phi_t(\mathbf{X}, t)$, or equivalently the displacement vector $\mathbf{u}(\mathbf{X}, t)$.

A fundamental measure for deformation and strain in the context of finite deformation solid mechanics is given by the deformation gradient \mathbf{F} , defined as partial derivative of the current configuration with respect to the reference configuration:

$$\mathbf{F} = \frac{\partial \mathbf{x}(\mathbf{X}, t)}{\partial \mathbf{X}} = \mathbf{I} + \frac{\partial \mathbf{u}(\mathbf{X}, t)}{\partial \mathbf{X}}, \quad (2.3)$$

where \mathbf{I} is the second-order identity tensor. Note, that the deformation gradient is a so-called two-point tensor, which has one basis defined in the current configuration and the other in the reference configuration. Geometrically interpreted, \mathbf{F} represents the mapping of an infinitesimal line element $d\mathbf{X}$ in the reference configuration to a corresponding line element $d\mathbf{x}$ in the current configuration, often denoted as *push-forward* operation, i.e.

$$d\mathbf{x} = \mathbf{F} \cdot d\mathbf{X}. \quad (2.4)$$

Assuming bijectivity and smoothness of the deformation map Φ_t , the inverse deformation gradient $\mathbf{F}^{-1} = \partial \mathbf{X} / \partial \mathbf{x}$ and the corresponding *pull-back* operation $d\mathbf{X} = \mathbf{F}^{-1} \cdot d\mathbf{x}$ are also well-defined, therefore guaranteeing a positive determinant $J = \det \mathbf{F} > 0$. This quantity, also

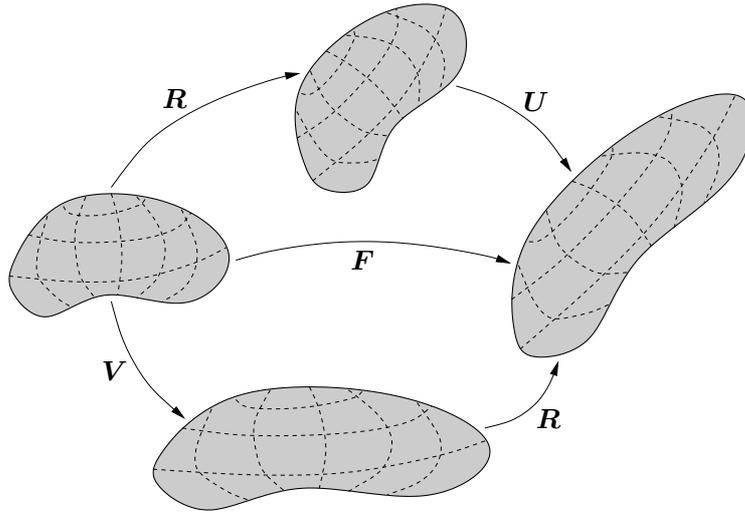


Figure 2.2: Polar decomposition of the deformation gradient \mathbf{F} .

commonly denoted as Jacobian determinant of the deformation, represents the transformation of an infinitesimal volume element between the two configurations:

$$dV = \det \mathbf{F} dV_0 = J dV_0. \quad (2.5)$$

Finally, the deformation gradient also allows for the mapping of an infinitesimal, oriented area element from reference to current configuration, yielding

$$d\mathbf{A} = J \mathbf{F}^{-\top} \cdot d\mathbf{A}_0, \quad (2.6)$$

which is commonly referred to as Nanson's formula. Herein, the infinitesimal area elements are interpreted as vectors $d\mathbf{A}_0 = dA_0 \mathbf{N}$ and $d\mathbf{A} = dA \mathbf{n}$, where \mathbf{N} and \mathbf{n} denote unit normal vectors of the area element in the reference and current configuration, respectively.

According to the polar decomposition theorem, any deformation characterized by the deformation gradient \mathbf{F} can be split into a volume-preserving rigid body motion part and a volume-changing stretch part, i.e.

$$\mathbf{F} = \mathbf{R} \cdot \mathbf{U}, \quad (2.7)$$

where \mathbf{R} is an orthogonal rotation tensor and \mathbf{U} the so-called right (or material) stretch tensor. Figure 2.2 illustrates the polar decomposition theorem and the interpretation of \mathbf{R} and \mathbf{U} as rigid body rotation and distortion, respectively. Based on these considerations, the right Cauchy–Green tensor \mathbf{C} is defined as

$$\mathbf{C} = \mathbf{F}^\top \cdot \mathbf{F} = \mathbf{U}^\top \cdot \mathbf{R}^\top \cdot \mathbf{R} \cdot \mathbf{U} = \mathbf{U}^\top \cdot \mathbf{U}, \quad (2.8)$$

where the orthogonality of the rotation tensor \mathbf{R} has been exploited. For the sake of completeness, it is pointed out that there also exists an alternative notation of (2.7), viz. $\mathbf{F} = \mathbf{V} \cdot \mathbf{R}$, based on the so-called left (or spatial) stretch tensor \mathbf{V} , see also Figure 2.2.

Comparing the deformation gradient \mathbf{F} and the right Cauchy–Green tensor \mathbf{C} as fundamental measures of deformation, three important differences become apparent. Firstly, \mathbf{C} is defined

purely with respect to the reference configuration, while the deformation gradient is a two-point tensor. Secondly, the right Cauchy–Green tensor is an objective measure meaning that any rigid body motion can be superimposed on the actual deformation without changing the definition of \mathbf{C} , while \mathbf{F} is not invariant under rigid body rotations. Thirdly, while \mathbf{F} correlates infinitesimal line elements, \mathbf{C} describes the mapping of their squares (including information on the enclosed angle) between reference and current configuration, i.e.

$$d\mathbf{x} \cdot d\mathbf{x} = d\mathbf{X} \cdot \mathbf{C} \cdot d\mathbf{X}. \quad (2.9)$$

The described properties make the right Cauchy–Green tensor \mathbf{C} an ideal candidate on the basis of which to formulate suitable strain measures for finite deformation solid mechanics. To ensure a consistent definition of zero strain in the undeformed configuration, an apparent choice is the so-called Green–Lagrange strain tensor \mathbf{E} defined in the material configuration as

$$\mathbf{E} = \frac{1}{2}(\mathbf{F}^T \cdot \mathbf{F} - \mathbf{I}) = \frac{1}{2}(\mathbf{C} - \mathbf{I}). \quad (2.10)$$

Although strain measures are never unique, the Green–Lagrange strain tensor is a very common choice in nonlinear solid mechanics, and can be considered particularly convenient if large deformations occur but only a moderate amount of stretch and compression. Other well-known strain measures in nonlinear solid mechanics include the so-called Euler–Almansi strain tensor, which is simply the spatial counterpart of the Green–Lagrange strain tensor, and the logarithmic strain tensor, which is the prevalent choice in large strain inelasticity.

The first and second time derivatives of the displacement vector $\mathbf{u}(\mathbf{X}, t)$ in material description, i.e. velocities $\dot{\mathbf{u}}(\mathbf{X}, t)$ and accelerations $\ddot{\mathbf{u}}(\mathbf{X}, t)$, are defined as follows:

$$\dot{\mathbf{u}}(\mathbf{X}, t) = \left. \frac{\partial \mathbf{u}(\mathbf{X}, t)}{\partial t} \right|_{\mathbf{X}} = \frac{d\mathbf{u}(\mathbf{X}, t)}{dt}, \quad (2.11)$$

$$\ddot{\mathbf{u}}(\mathbf{X}, t) = \left. \frac{\partial \dot{\mathbf{u}}(\mathbf{X}, t)}{\partial t} \right|_{\mathbf{X}} = \frac{d\dot{\mathbf{u}}(\mathbf{X}, t)}{dt} = \frac{d^2\mathbf{u}(\mathbf{X}, t)}{dt^2}. \quad (2.12)$$

Corresponding rate forms (i.e. time derivatives) of the deformation measures, such as the material velocity gradient $\mathbf{L} = \dot{\mathbf{F}}$ or the material strain rate tensor $\dot{\mathbf{E}} = \frac{1}{2}(\dot{\mathbf{F}}^T \cdot \mathbf{F} + \mathbf{F}^T \cdot \dot{\mathbf{F}}) = \frac{1}{2}\dot{\mathbf{C}}$ are readily defined, too.

2.1.2 Stresses and constitutive laws

The motion and deformation of an elastic body effects internal stresses. This is readily described by the traction vector \mathbf{t} in the current configuration:

$$\mathbf{t}(\mathbf{n}, \mathbf{x}, t) = \lim_{\Delta A \rightarrow 0} \frac{\Delta \mathbf{f}}{\Delta A}, \quad (2.13)$$

yielding the limit value of the resulting force \mathbf{f} acting on an arbitrary surface area ΔA characterized by its unit surface normal vector \mathbf{n} . The Cauchy theorem, commonly derived from equilibrium considerations using an infinitesimal tetrahedral volume element, then correlates tractions and stresses via

$$\mathbf{t} = \boldsymbol{\sigma} \cdot \mathbf{n}. \quad (2.14)$$

Herein, the symmetric Cauchy stress tensor $\boldsymbol{\sigma}$ represents the true internal stress state within a body in its a priori unknown current configuration, with diagonal and off-diagonal components being interpretable as normal stresses and shear stresses, respectively. A multitude of alternative stress definitions is also prevailing in nonlinear continuum mechanics. Exemplarily, the first Piola–Kirchhoff stress tensor \boldsymbol{P} , a two-point tensor, maps the *material* surface element $d\boldsymbol{A}_0 = dA_0 \boldsymbol{N}$ onto the spatial resulting force \boldsymbol{f} . Its definition is obtained from the Cauchy stress tensor $\boldsymbol{\sigma}$ by applying Nanson’s formula (2.6), yielding

$$\boldsymbol{P} = J \boldsymbol{\sigma} \cdot \boldsymbol{F}^{-\top}. \quad (2.15)$$

Consequently, it is possible to construct a stress tensor purely based on quantities in the reference configuration, too. By also transforming the resulting force vector \boldsymbol{f} according to (2.4), the symmetric second Piola–Kirchhoff stress tensor \boldsymbol{S} emerges as

$$\boldsymbol{S} = \boldsymbol{F}^{-1} \cdot \boldsymbol{P} = J \boldsymbol{F}^{-1} \cdot \boldsymbol{\sigma} \cdot \boldsymbol{F}^{-\top}. \quad (2.16)$$

In contrast to the Cauchy (or “true”) stress $\boldsymbol{\sigma}$, the first and second Piola–Kirchhoff stresses do not allow for a clear interpretation from an engineering point of view, i.e. the concept of normal and shear stress components is not directly transferable due to the fact that pull-back operations to the reference configuration are involved.

Nevertheless, the investigations above show that similar to strain measures also different stress measures are available in nonlinear continuum mechanics. However, stress and strain definitions cannot be combined arbitrarily in practice, but there rather exist natural connections based on the notion of energy conjugate (or work conjugate) pairs, see e.g. Bonet and Wood [17] for more details. Using the interrelations derived above, the following equivalent formulations of the internal power \mathcal{P}_{int} are possible:

$$\mathcal{P}_{\text{int}} = \int_{\Omega_t} \boldsymbol{\sigma} : (\boldsymbol{F}^{-\top} \cdot \dot{\boldsymbol{E}} \cdot \boldsymbol{F}^{-1}) d\Omega_t = \int_{\Omega_0} \boldsymbol{P} : \dot{\boldsymbol{F}} d\Omega_0 = \int_{\Omega_0} \boldsymbol{S} : \dot{\boldsymbol{E}} d\Omega_0, \quad (2.17)$$

where the spatial rate of deformation tensor $\boldsymbol{F}^{-\top} \cdot \dot{\boldsymbol{E}} \cdot \boldsymbol{F}^{-1}$ simply represents the spatial counterpart of the material strain rate tensor $\dot{\boldsymbol{E}}$. For pure rigid body motions the internal power vanishes due to the fact that $\dot{\boldsymbol{F}} = \dot{\boldsymbol{E}} = \mathbf{0}$ holds in that case.

With typical measures for both strains and stresses being established, constitutive relations or material models provide the missing link between kinematics and material response and will be presented next. Throughout this thesis, only homogeneous bodies undergoing purely elastic deformation processes without internal dissipation are considered. Moreover, the existence of a so-called strain energy function or elastic potential $\Psi(\boldsymbol{F})$ is assumed, which only depends upon the current state of deformation (*hyperelastic* material behavior). The requirement of objectivity implies that Ψ remains unchanged when an arbitrary rigid body rotation is applied to the current configuration. Thus, the strain energy function must be independent of the rotation component \boldsymbol{R} of the deformation gradient and can equivalently be expressed in terms of the stretch tensor \boldsymbol{U} , the right Cauchy–Green tensor \boldsymbol{C} or derived quantities, i.e.

$$\Psi(\boldsymbol{F}) = \Psi(\boldsymbol{U}) = \Psi(\boldsymbol{C}) = \Psi(\boldsymbol{E}). \quad (2.18)$$

A common formulation of hyperelastic materials in the reference frame then follows as

$$\boldsymbol{S} = 2 \frac{\partial \Psi}{\partial \boldsymbol{C}} = \frac{\partial \Psi}{\partial \boldsymbol{E}}. \quad (2.19)$$

The relation between \mathbf{S} and \mathbf{E} given by (2.19) will in general be nonlinear. Thus, it is possible (and necessary within typical finite element procedures, as will be seen later) to determine the fourth-order material elasticity tensor \mathcal{C}_m via repeated derivation, yielding

$$\mathcal{C}_m = \frac{\partial \mathbf{S}}{\partial \mathbf{E}} = \frac{\partial^2 \Psi}{\partial \mathbf{E} \partial \mathbf{E}}. \quad (2.20)$$

Exemplarily, two prevailing constitutive models are presented here. The St.-Venant–Kirchhoff material model is an isotropic, hyperelastic model based on a quadratic strain energy function

$$\Psi_{\text{SVK}} = \frac{\lambda}{2} (\text{tr } \mathbf{E})^2 + \mu \mathbf{E} : \mathbf{E}. \quad (2.21)$$

In this context, λ and μ represent the so-called Lamé parameters, which are correlated with the more common Young’s modulus E and Poisson’s ratio ν via

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)}, \quad \mu = \frac{E}{2(1+\nu)}. \quad (2.22)$$

Inserting (2.21) into (2.19) and (2.20), it can easily be observed that the St.-Venant–Kirchhoff material model defines a linear relationship between Green–Lagrange strains \mathbf{E} and second Piola–Kirchhoff stresses \mathbf{S} , and can therefore be interpreted as an objective generalization of Hooke’s law to the geometrically nonlinear realm.

Another widely used isotropic, hyperelastic constitutive law is the compressible Neo–Hookean model with the strain energy function

$$\Psi_{\text{NH}} = \frac{\mu}{2} (\text{tr } \mathbf{C} - 3) - \mu \ln J + \frac{\lambda}{2} (\ln J)^2. \quad (2.23)$$

In contrast to the simple St.-Venant–Kirchhoff model, a nonlinear stress-strain relationship is obtained when determining the partial derivative of (2.23) with respect to the Green–Lagrange strain tensor \mathbf{E} . Many other constitutive laws exist for miscellaneous applications (e.g. the well-known Mooney–Rivlin or Ogden models for rubber materials). However, with the focus of this thesis being on contact interaction rather than constitutive modeling, the interested reader is referred to the abundant literature on hyperelasticity, viscoelasticity or elastoplasticity for further details, e.g. in Holzapfel [78], Ogden [118] and Simo and Hughes [148]. The mesh tying and contact algorithms developed in Chapters 3 and 4 are applicable to any nonlinear solid mechanics problem, independent of the complexity of the employed constitutive model.

2.1.3 Balance equations

Apart from the kinematics, stresses and constitutive laws discussed in the previous sections, mechanical systems are commonly characterized by balance equations for mass, momentum and energy. There also exist associated conservation laws for these mechanical quantities, which are all valid under certain conditions. It is generally assumed here that no material degradation or growth occurs, and thus mass is always conserved. Momentum conservation, however, depends on whether external forces and moments are acting on the system or not. Conservation of mechanical energy is further reliant on the fact that the considered system is free of dissipative effects (e.g. due to inelastic material behavior). Typically, the balance equations can be

formulated globally, i.e. in integral form considering the whole body, or locally, i.e. at a specific material point. Moreover, it is again possible to take either a material or a spatial point of view.

Conservation of mass

The global form of the conservation of mass in spatial description is given as

$$\frac{dm}{dt} = \frac{d}{dt} \int_{\Omega_t} \rho dV = \int_{\Omega_t} (\dot{\rho} + \rho \operatorname{div} \dot{\mathbf{u}}) dV = 0. \quad (2.24)$$

Herein, the spatial mass density ρ of the spatial volume element dV has been introduced, and use has been made of Reynold's transport theorem as well as the spatial divergence operator $\operatorname{div}(\cdot)$. Consequently, the corresponding global form in material description follows as

$$\frac{dm}{dt} = \frac{d}{dt} \int_{\Omega_0} \rho_0 dV_0 = \int_{\Omega_0} \dot{\rho}_0 dV_0 = 0, \quad (2.25)$$

where the so-called reference mass density $\rho_0 = J\rho$ of the material volume element dV_0 is correlated with the spatial mass density ρ via the Jacobian determinant. The last conversion step in (2.25) simplifies significantly as compared with (2.24) due to the fact that the reference configuration does not depend on time. Obtaining the local formulations of mass conservation is straightforward, given the fact that (2.24) and (2.25) must also hold at any material point, i.e.

$$\dot{\rho} + \rho \operatorname{div} \dot{\mathbf{u}} = 0, \quad (2.26)$$

$$\dot{\rho}_0 = 0. \quad (2.27)$$

Balance of linear momentum

The global form of the balance of linear momentum in spatial description is obtained based on the requirement that the time derivative of linear momentum equals the sum of all external forces acting on a body. This yields

$$\frac{d}{dt} \int_{\Omega_t} \rho \dot{\mathbf{u}} dV = \int_{\Omega_t} \hat{\mathbf{b}} dV + \int_{\partial\Omega_t} \hat{\mathbf{t}} dA, \quad (2.28)$$

where $\hat{\mathbf{t}}$ represents external tractions on the boundary $\partial\Omega_t$ of the body in the current configuration, and $\hat{\mathbf{b}}$ refers to an external body force per current unit volume. After application of Reynold's transport theorem and local mass conservation (2.26) to the left-hand side and Gauss divergence theorem to the right-hand side of (2.28), a more convenient version of the global linear momentum balance in spatial description is obtained:

$$\int_{\Omega_t} \rho \ddot{\mathbf{u}} dV = \int_{\Omega_t} (\operatorname{div} \boldsymbol{\sigma} + \hat{\mathbf{b}}) dV. \quad (2.29)$$

Similarly, the global form in material description is first given by

$$\frac{d}{dt} \int_{\Omega_0} \rho_0 \dot{\mathbf{u}} dV_0 = \int_{\Omega_0} \hat{\mathbf{b}}_0 dV_0 + \int_{\partial\Omega_0} \hat{\mathbf{t}}_0 dA_0, \quad (2.30)$$

with the external volume and surface forces $\hat{\mathbf{b}}_0$ and $\hat{\mathbf{t}}_0$ now being defined on the undeformed unit volume and the boundary $\partial\Omega_0$, respectively. Analogous manipulations as carried out above lead to the following version of the global linear momentum balance in material description:

$$\int_{\Omega_0} \rho_0 \ddot{\mathbf{u}} \, dV_0 = \int_{\Omega_0} (\text{Div} \mathbf{P} + \hat{\mathbf{b}}_0) \, dV_0, \quad (2.31)$$

where the material divergence operator $\text{Div}(\cdot)$ has been introduced. Corresponding local formulations in spatial and material description are again readily obtained as

$$\rho \ddot{\mathbf{u}} = \text{div} \boldsymbol{\sigma} + \hat{\mathbf{b}}, \quad (2.32)$$

$$\rho_0 \ddot{\mathbf{u}} = \text{Div} \mathbf{P} + \hat{\mathbf{b}}_0, \quad (2.33)$$

also commonly denoted as Cauchy's first equation of motion. Together with the kinematic and constitutive relations introduced previously, the local form of the balance of linear momentum plays an important role in the mathematical formulation of solid mechanics problems as so-called initial boundary value problem (IBVP), as will be demonstrated in Section 2.1.4.

Balance of angular momentum

Similar to the balance of linear momentum, the balance of angular momentum can be deduced from the requirement that the time derivative of angular momentum with respect to a fixed point of origin equals the sum of all external moments acting on a body. Thus, the global form in spatial description is given as

$$\frac{d}{dt} \int_{\Omega_t} (\rho \mathbf{x} \times \dot{\mathbf{u}}) \, dV = \int_{\Omega_t} (\mathbf{x} \times \hat{\mathbf{b}}) \, dV + \int_{\partial\Omega_t} (\mathbf{x} \times \hat{\mathbf{t}}) \, dA. \quad (2.34)$$

Starting from (2.34) and mapping all domain and boundary integrals back to the reference configuration yields the balance of angular momentum in material description:

$$\frac{d}{dt} \int_{\Omega_0} (\rho_0 \mathbf{x} \times \dot{\mathbf{u}}) \, dV_0 = \int_{\Omega_0} (\mathbf{x} \times \hat{\mathbf{b}}_0) \, dV_0 + \int_{\partial\Omega_0} (\mathbf{x} \times \hat{\mathbf{t}}_0) \, dA_0. \quad (2.35)$$

Without presenting all mathematical details, it is pointed out that the local forms of the balance of angular momentum are again obtained by applying Gauss divergence theorem to the boundary integral term in (2.34) and (2.35), respectively. This eventually leads to the interesting observation that the local balance of angular momentum is equivalent to the requirement that the Cauchy stress tensor $\boldsymbol{\sigma}$ and the second Piola–Kirchhoff stress tensor \mathbf{S} are symmetric, i.e.

$$\boldsymbol{\sigma}^\top = \boldsymbol{\sigma}, \quad \mathbf{S}^\top = \mathbf{S}. \quad (2.36)$$

This result is also known as Cauchy's second equation of motion. Therefore, the balance of angular momentum usually does not enter the IBVP of nonlinear solid mechanics as independent equation, but rather implicitly by demanding symmetry of $\boldsymbol{\sigma}$ and \mathbf{S} .

Balance of mechanical energy

Throughout this thesis, only purely mechanical systems are considered and no other form of

energy (e.g. thermal, chemical, electrical) is taken into account. The balance of energy can generally be derived from the prior condition that the change in total energy equals the introduced external power, i.e.

$$\frac{d}{dt} \int_{\Omega_t} \frac{1}{2} \rho \dot{\mathbf{u}} \cdot \dot{\mathbf{u}} dV + \int_{\Omega_t} \boldsymbol{\sigma} : (\mathbf{F}^{-T} \cdot \dot{\mathbf{E}} \cdot \mathbf{F}^{-1}) dV = \int_{\Omega_t} \hat{\mathbf{b}} \cdot \dot{\mathbf{u}} dV + \int_{\partial\Omega_t} \hat{\mathbf{t}} \cdot \dot{\mathbf{u}} dA. \quad (2.37)$$

Herein, the first term on the left-hand side represents the rate of kinetic energy \mathcal{P}_{kin} whereas the second term is the internal mechanical power \mathcal{P}_{int} . The right-hand side comprises the external power \mathcal{P}_{ext} introduced by external volume and surface forces. For the sake of completeness, the material description of the global energy balance is given here, too:

$$\frac{d}{dt} \int_{\Omega_0} \frac{1}{2} \rho_0 \dot{\mathbf{u}} \cdot \dot{\mathbf{u}} dV_0 + \int_{\Omega_0} \mathbf{S} : \dot{\mathbf{E}} dV_0 = \int_{\Omega_0} \hat{\mathbf{b}}_0 \cdot \dot{\mathbf{u}} dV_0 + \int_{\partial\Omega_0} \hat{\mathbf{t}}_0 \cdot \dot{\mathbf{u}} dA_0, \quad (2.38)$$

where use has been made of the energy conjugate stress and strain rate pairs as defined in (2.17). While, in general, the balance of energy represents an independent postulate, it can easily be shown to reduce to a mere consequence of the balance of linear momentum in the case of purely mechanical systems considered here, see e.g. the lecture notes in Wall and Cyron [170]. Thus, local forms of the balance of mechanical energy do not provide any additional information as compared with the local forms of the balance of linear momentum.

2.1.4 Initial boundary value problem

The initial boundary value problem of nonlinear solid mechanics comprises the main results of the three previous sections within a set of coupled second-order partial differential equations, additionally satisfying a given set of initial conditions as well as boundary conditions. Exemplarily, the IBVP will be presented in the reference configuration here, however the spatial description is derived analogously. For the definition of suitable boundary conditions, $\partial\Omega_0$ is decomposed into two complementary sets in the absence of contact: Γ_σ represents the Neumann boundary, where the tractions $\hat{\mathbf{t}}_0$ are given, and Γ_u denotes the Dirichlet boundary, where displacements $\hat{\mathbf{u}}$ are prescribed. Neumann and Dirichlet boundaries are disjoint sets, i.e.

$$\Gamma_\sigma \cup \Gamma_u = \partial\Omega_0, \quad \Gamma_\sigma \cap \Gamma_u = \emptyset. \quad (2.39)$$

The initial boundary value problem in material description can be summarized as follows :

$$\text{Div} \mathbf{P} + \hat{\mathbf{b}}_0 = \rho_0 \ddot{\mathbf{u}} \quad \text{in } \Omega_0 \times [0, T], \quad (2.40)$$

$$\mathbf{u} = \hat{\mathbf{u}} \quad \text{on } \Gamma_u \times [0, T], \quad (2.41)$$

$$\mathbf{P} \cdot \mathbf{N} = \hat{\mathbf{t}}_0 \quad \text{on } \Gamma_\sigma \times [0, T]. \quad (2.42)$$

Herein, T denotes the end of the considered time interval. Due to the time dependency within the balance of linear momentum in (2.40), which contains second derivatives with respect to time t , a suitable set of initial conditions specifying the displacements $\hat{\mathbf{u}}_0(\mathbf{X})$ and velocities $\hat{\dot{\mathbf{u}}}_0(\mathbf{X})$ at time $t = 0$ is needed, viz.

$$\mathbf{u}(\mathbf{X}, 0) = \hat{\mathbf{u}}_0(\mathbf{X}) \quad \text{in } \Omega_0, \quad (2.43)$$

$$\dot{\mathbf{u}}(\mathbf{X}, 0) = \hat{\dot{\mathbf{u}}}_0(\mathbf{X}) \quad \text{in } \Omega_0. \quad (2.44)$$

The definition of a material model, such as for instance the ones given in (2.21) or (2.23), eventually rounds off the initial boundary value problem of finite deformation solid mechanics. The IBVP is also commonly referred to as *strong* formulation of nonlinear solid mechanics, as equations (2.40)–(2.44) are enforced at each individual point within the domain Ω_0 . Analytical solutions to the considered class of problems only exist in special cases, usually with a restriction to elementary geometries and under additional simplifying assumptions. However, the IBVP and especially its restatement as so-called *weak* formulation (see Section 2.3) represent the basis for numerical discretization techniques such as the finite element method.

2.2 Contact mechanics

In standard nonlinear solid mechanics, as considered above, both Dirichlet and Neumann boundaries are known a priori, which means that either the prescribed displacement or the external force is known on the entire boundary $\partial\Omega_0$ of the body. Contact interaction demands an extension of this framework based on the apparent observation that an additional contact boundary exists, which is *unknown* a priori and possibly also changes continuously over time. Moreover, neither the contact forces nor the displacements on the contact boundary are prescribed, which introduces a new type of nonlinearity into the previously described continuum mechanics problem formulation. Apart from nonlinear kinematic relations due to finite deformations and nonlinear constitutive behavior, contact interaction can in the first instance be mathematically interpreted as a set of nonlinear boundary conditions.

The following sections give a short introduction to the most important notions and concepts of contact mechanics, including contact kinematics, typical formulations of contact conditions and an overview of related physical phenomena, such as frictional sliding. For further details on these topics, the interested reader is referred to classical textbooks on contact mechanics, e.g. Johnson [91] and Kikuchi and Oden [94], or to more recent monographs on computational methods for contact mechanics, e.g. Laursen [107] and Wriggers [180].

2.2.1 Contact kinematics

From the viewpoint of mathematical problem formulation, contact and impact procedures can be classified into several different categories. A problem setup consisting of one single deformable body and a rigid obstacle is commonly referred to as Signorini contact, while the typical general problem formulation rests upon the assumption of two deformable bodies undergoing contact interaction. Moreover, self contact and contact involving multiple bodies represent well-known special cases. While it is usually advantageous or even essential to design specific numerical algorithms for the aforementioned special cases (see Section 4.6.3), all mathematical basics concerning contact kinematics and contact constraints can yet be perfectly derived for the case of two deformable bodies.

Hence, deformable-deformable contact of two bodies undergoing finite deformations, as illustrated in Figure 2.3, serves as prototype exclusively considered throughout this thesis. Notational details associated with contact interaction will be introduced first, thus extending the previously described problem formulation of nonlinear solid mechanics. Let the open sets $\Omega_0^{(1)}, \Omega_0^{(2)} \subset \mathbb{R}^3$ and $\Omega_t^{(1)}, \Omega_t^{(2)} \subset \mathbb{R}^3$ represent two bodies in the reference and current configuration, respectively.

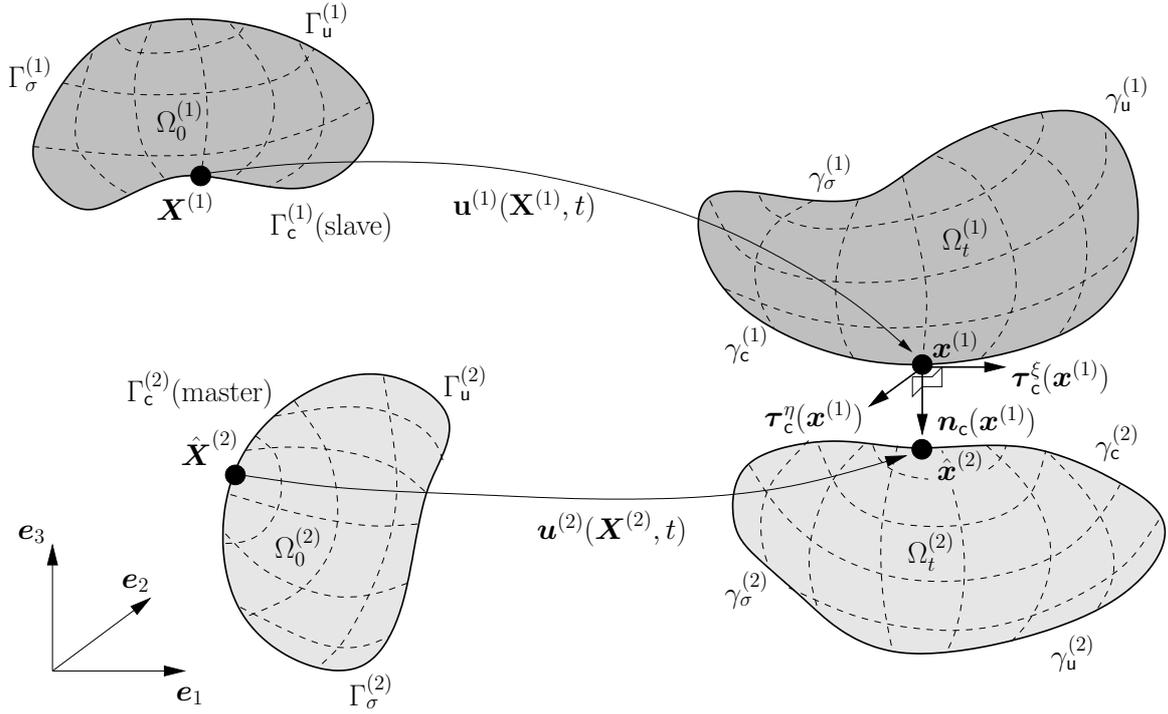


Figure 2.3: Kinematics and basic notation for a two body unilateral contact problem in 3D.

As the two bodies approach each other and may potentially come into contact on parts of their boundaries, the surfaces $\partial\Omega_0^{(i)}$, $i = 1, 2$, are now divided into three disjoint subsets, viz.

$$\begin{aligned} \partial\Omega_0^{(i)} &= \Gamma_u^{(i)} \cup \Gamma_\sigma^{(i)} \cup \Gamma_c^{(i)}, \\ \Gamma_u^{(i)} \cap \Gamma_\sigma^{(i)} &= \Gamma_u^{(i)} \cap \Gamma_c^{(i)} = \Gamma_\sigma^{(i)} \cap \Gamma_c^{(i)} = \emptyset, \end{aligned} \quad (2.45)$$

where $\Gamma_u^{(i)}$ and $\Gamma_\sigma^{(i)}$ are the well-known Dirichlet and Neumann boundaries, and $\Gamma_c^{(i)}$ represents the *potential* contact surface. The counterparts in the current configuration are denoted as $\gamma_u^{(i)}$, $\gamma_\sigma^{(i)}$ and $\gamma_c^{(i)}$. It is characteristic of contact problems that the *actual*, so-called active contact surface $\Gamma_a^{(i)} \subseteq \Gamma_c^{(i)}$ is unknown, possibly continuously changing over time and thus has to be determined as part of the nonlinear solution process. For the sake of completeness, and to be mathematically precise, the currently inactive contact surface $\Gamma_i^{(i)} = \Gamma_c^{(i)} \setminus \Gamma_a^{(i)}$ should technically be interpreted as part of the Neumann boundary $\Gamma_\sigma^{(i)}$.

A classical nomenclature in contact mechanics is retained throughout this thesis by referring to $\Gamma_c^{(1)}$ as the *slave* surface and to $\Gamma_c^{(2)}$ as the *master* surface, although the master-slave concept actually only makes sense in the context of finite element discretization and although its traditional meaning will not be entirely conveyed to the mortar FE approach proposed here.

Both bodies are required to satisfy the initial boundary value problem previously presented in Section 2.1.4, with the motion and deformation being described by the absolute displacement vectors $\mathbf{u}^{(i)} = \mathbf{x}^{(i)} - \mathbf{X}^{(i)}$. Moreover, a new fundamental geometric measure for proximity, potential contact and penetration of the two bodies is introduced with the so-called gap function $g_n(\mathbf{X}, t)$ in the current configuration. It is evident that the gap function and other contact-

related quantities need to be examined in a spatial description, even though the IBVP may still be formulated with respect to the reference configuration. The gap function is defined as

$$g_n(\mathbf{X}, t) = -\mathbf{n}_c \cdot \left[\mathbf{x}^{(1)}(\mathbf{X}^{(1)}, t) - \hat{\mathbf{x}}^{(2)}(\hat{\mathbf{X}}^{(2)}(\mathbf{X}^{(1)}, t), t) \right], \quad (2.46)$$

where some alternatives exist for the identification of the contact point $\hat{\mathbf{x}}^{(2)}$ on the master surface associated with each point $\mathbf{x}^{(1)}$ on the slave surface and also for the corresponding contact normal vector \mathbf{n}_c . The classical and perhaps most intuitive choice in contact mechanics is based on the so-called closest point projection (CPP), which determines $\hat{\mathbf{x}}^{(2)}$ as

$$\hat{\mathbf{x}}^{(2)} = \arg \min_{\mathbf{x}^{(2)} \in \gamma_c^{(2)}} \|\mathbf{x}^{(1)} - \mathbf{x}^{(2)}\|. \quad (2.47)$$

Consequently, \mathbf{n}_c is then chosen to be the outward unit normal to the current master surface $\gamma_c^{(2)}$ in $\hat{\mathbf{x}}^{(2)}$. A very comprehensive overview of the closest point projection, its mathematical properties and possible pitfalls due to non-uniqueness and certain pathological cases can be found in Konyukhov and Schweizerhof [97]. However, a slightly different approach is followed here, with the outward unit normal to the current *slave* surface $\gamma_c^{(1)}$ being considered as contact normal \mathbf{n}_c . Hence, the master side contact point $\hat{\mathbf{x}}^{(2)}$ is the result of a smooth interface mapping $\chi : \gamma_c^{(1)} \rightarrow \gamma_c^{(2)}$ of $\mathbf{x}^{(1)}$ onto the master surface $\gamma_c^{(2)}$ along \mathbf{n}_c , see Figure 2.3. Especially in the context of mortar finite element discretization, this choice has some practical advantages over the classical closest point projection common for node-to-segment discretization, as will be discussed in more detail in Sections 3.2.2 and 3.2.3.

Together with two vectors $\boldsymbol{\tau}_c^\xi$ and $\boldsymbol{\tau}_c^\eta$ taken from the tangential plane, \mathbf{n}_c forms a set of orthonormal basis vectors in the slave surface point $\mathbf{x}^{(1)}$. As these basis vector are attached to $\mathbf{x}^{(1)}$ and also move accordingly, they are commonly referred to as slip advected basis vectors. In this context, it is worth noting that the contact surface $\gamma_c^{(1)}$ is a two-dimensional manifold, which means that the tangential plane in each point $\mathbf{x}^{(1)}$ locally defines an \mathbb{R}^2 space embedded into the global \mathbb{R}^3 . Therefore, any quantity on $\gamma_c^{(1)}$ is readily parametrized with the two local coordinates $\xi(\mathbf{X}^{(1)}, t)$ and $\eta(\mathbf{X}^{(1)}, t)$. While the gap function characterizes contact interaction in normal direction, the primary kinematic variable for frictional sliding in tangential direction is given by the relative tangential velocity

$$\mathbf{v}_{\tau, \text{rel}} = (\mathbf{I} - \mathbf{n}_c \otimes \mathbf{n}_c) \cdot \left[\dot{\mathbf{x}}^{(1)}(\mathbf{X}^{(1)}, t) - \dot{\hat{\mathbf{x}}}^{(2)}(\hat{\mathbf{X}}^{(2)}(\mathbf{X}^{(1)}, t), t) \right]. \quad (2.48)$$

Note that this expression for $\mathbf{v}_{\tau, \text{rel}}$ is only exact in the case of perfect sliding and persistent contact, i.e. assuming $g_n = \dot{g}_n = 0$. Nevertheless, it is typically employed for quantifying the relative tangential movement of contacting bodies in all cases, even if the described prerequisites are not met exactly. To clarify the notation in (2.48), it is pointed out that $\dot{\hat{\mathbf{x}}}^{(2)}$ represents the current velocity of the material point $\hat{\mathbf{X}}^{(2)}$, viz. the material contact point associated with $\mathbf{X}^{(1)}$ at time t . Therefore, it does *not* include a change of the material contact point $\hat{\mathbf{X}}^{(2)}$ itself, or in other words, it does *not* include a change of the CPP of slave point $\mathbf{x}^{(1)}$. Based on the tangential plane defined above, $\mathbf{v}_{\tau, \text{rel}}$ can be decomposed into

$$\mathbf{v}_{\tau, \text{rel}} = v_\tau^\xi \boldsymbol{\tau}_c^\xi + v_\tau^\eta \boldsymbol{\tau}_c^\eta. \quad (2.49)$$

As already mentioned, the definition of the relative tangential velocity given above is only frame-indifferent when perfect sliding occurs ($g_n = 0$), see e.g. Laursen [107]. However, since an objective measure of the slip rate is essential for formulating frictional contact conditions in finite deformation formulations, an appropriate algorithmic modification of the slip rate is typically carried out later in the course of finite element discretization.

Similar to the kinematic measures g_n and $\mathbf{v}_{\tau,\text{rel}}$, the contact traction $\mathbf{t}_c^{(1)}$ on the slave surface $\gamma_c^{(1)}$ can be split into normal and tangential components, yielding

$$\mathbf{t}_c^{(1)} = p_n \mathbf{n}_c + \mathbf{t}_\tau = p_n \mathbf{n}_c + t_\tau^\xi \boldsymbol{\tau}_c^\xi + t_\tau^\eta \boldsymbol{\tau}_c^\eta. \quad (2.50)$$

Moreover, due to the balance of linear momentum on the contact interface, the traction vectors on slave side $\gamma_c^{(1)}$ and master side $\gamma_c^{(2)}$ are identical except for opposite signs, i.e.

$$\mathbf{t}_c^{(1)} = -\mathbf{t}_c^{(2)}. \quad (2.51)$$

These definitions conclude the contact kinematics section, with all mechanical quantities needed for the definition of contact conditions for nonlinear solid mechanics now introduced.

2.2.2 Tied contact constraints

While the main focus of this thesis is on unilateral contact problems, the integration of mesh tying or tied contact problems for connecting dissimilar meshes suggests itself due to the numerous conceptual similarities. Mesh tying applications are also closely connected to the notion of domain decomposition (see Section 2.5), sometimes simply being referred to as domain decomposition problems in nonlinear solid mechanics. Thus, throughout this thesis, mesh tying serves as simplified model problem through which many methodological and later also implementational aspects of computational contact mechanics can be clearly illustrated.

As will be seen in the following, mesh tying (or tied contact) perfectly fits into the framework of contact kinematics defined above and can simply be interpreted as a special case from now on. Figure 2.4 illustrates a typical tied contact problem in nonlinear solid mechanics. It can easily be observed that the fundamental kinematic measure for mesh tying is simply the relative displacement between the two bodies, sometimes also referred to as gap vector $\mathbf{g}(\mathbf{X}, t)$, viz.

$$\mathbf{g}(\mathbf{X}, t) = \mathbf{u}^{(1)}(\mathbf{X}^{(1)}, t) - \hat{\mathbf{u}}^{(2)}(\hat{\mathbf{X}}^{(2)}(\mathbf{X}^{(1)}, t), t). \quad (2.52)$$

Since it is typically assumed that the two bodies to be tied together share a common interface $\Gamma_c^{(1)} \equiv \Gamma_c^{(2)} \equiv \Gamma_c$ in the reference configuration, the gap vector is equivalently expressed as

$$\mathbf{g}(\mathbf{X}, t) = \mathbf{x}^{(1)}(\mathbf{X}^{(1)}, t) - \hat{\mathbf{x}}^{(2)}(\hat{\mathbf{X}}^{(2)}(\mathbf{X}^{(1)}, t), t), \quad (2.53)$$

thus demonstrating the similarity with the scalar gap function $g_n(\mathbf{X}, t)$ for unilateral contact defined in (2.46) even more clearly. As compared with unilateral contact, mesh tying firstly requires no distinction between normal and tangential directions at the interface, and secondly results in a simple vector-valued equality constraint:

$$\mathbf{g}(\mathbf{X}, t) = \mathbf{0}. \quad (2.54)$$

A considerable part of this thesis (see Chapter 3) will be exclusively devoted to the mortar finite element discretization of mesh tying problems in nonlinear solid mechanics, since this seems a very helpful intermediate step towards the actual topic of unilateral contact.

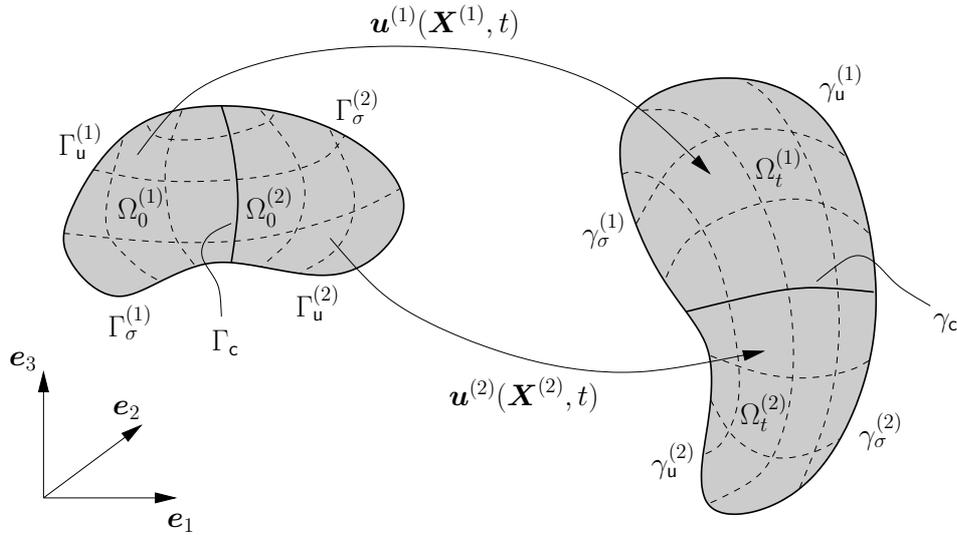


Figure 2.4: Kinematics and basic notation for a two body mesh tying problem in 3D.

2.2.3 Normal contact constraints

After the short interlude on mesh tying, the focus is now again set on unilateral contact conditions. Examining the gap function defined in (2.46) in more detail, it becomes obvious that a positive value $g_n(\mathbf{X}, t) > 0$ characterizes points currently not in contact, while a negative value $g_n(\mathbf{X}, t) < 0$ denotes the (physically non-admissible) state of penetration. Therefore, the classical set of Karush–Kuhn–Tucker (KKT) conditions, commonly also referred to as Hertz–Signorini–Moreau (HSM) conditions for frictionless contact on the contact boundary can be stated as

$$g_n(\mathbf{X}, t) \geq 0, \quad p_n(\mathbf{X}, t) \leq 0, \quad p_n(\mathbf{X}, t) g_n(\mathbf{X}, t) = 0. \quad (2.55)$$

As can be seen from Figure 2.5, the KKT conditions not only define a non-smooth and nonlinear contact law, but one that is multi-valued at $g_n(\mathbf{X}, t) = 0$. However, this set of inequality conditions also allows for a very intuitive physical interpretation. Due to the sign convention of the gap function introduced here, the first KKT condition simply represents the geometric constraint of non-penetration, whereas the second KKT condition implies that no adhesive stresses are allowed in the contact zone. Finally, the third KKT condition, well-known as complementarity condition, forces the gap to be closed when non-zero contact pressure occurs (contact) and the contact pressure to be zero when the gap is open (no contact). Note, that the type of KKT conditions defined in (2.55) also arise in many other problem classes of constrained optimization, and thus standard solution techniques (e.g. based on Lagrange multiplier methods and active set strategies) from optimization theory can readily be adapted for contact mechanics.

It is pointed out that the constraint formulation in (2.55) is based on a purely geometrical approach and takes a purely macroscopic perspective. This means that no micromechanical knowledge or constitutive equations taking into account the surface roughness or hardness of the contact interface are considered. Although not investigated in this thesis, such a micromechanical approach to normal contact is also possible, yielding a regularized version of the KKT conditions usually based on experimental data (e.g. flattening behavior of the surface asperities).

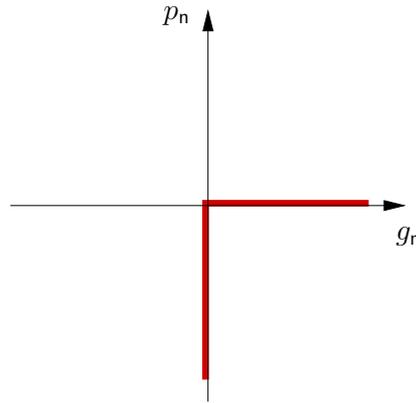


Figure 2.5: Karush–Kuhn–Tucker (KKT) conditions of non-penetration.

For the sake of completeness, the so-called persistency condition is also mentioned here. In the context of contact *dynamics*, the persistency condition is sometimes considered as an additional contact condition, requiring that

$$p_n(\mathbf{X}, t) \dot{g}_n(\mathbf{X}, t) = 0. \quad (2.56)$$

Herein, $\dot{g}_n(\mathbf{X}, t)$ represents the material time derivative of the gap function. Therefore, the persistency condition in combination with the KKT conditions in (2.55) basically demands that the contact pressure is only non-zero when the bodies are in contact and also remain so (persistent contact). On the contrary, the contact pressure is zero in the instant of bodies coming into contact and in the instant of separation. The persistency condition plays an important role in the design of energy conserving numerical algorithms for contact dynamics, see e.g. Laursen and Chawla [108], Laursen and Love [109], and bears a certain resemblance to the consistency condition in plasticity, see e.g. Simo and Hughes [148].

2.2.4 Frictional contact constraints

While frictionless response (i.e. $\mathbf{t}_\tau = \mathbf{0}$) is a common modeling assumption, and especially helpful for a thorough development of computational methods for contact mechanics, the real contact behavior of many technical systems is determined by the frictional response to tangential loading. The associated scientific field of tribology is extremely broad, also encompassing physical phenomena such as adhesion, wear or elastohydrodynamic lubrication. The following overview is restricted to a purely macroscopic observation of dry friction, classically described by Coulomb's law. One possible and widely used notation of Coulomb friction is given by

$$\Phi := \|\mathbf{t}_\tau\| - \mathfrak{F}|p_n| \leq 0, \quad \mathbf{v}_{\tau, \text{rel}} + \beta \mathbf{t}_\tau = \mathbf{0}, \quad \beta \geq 0, \quad \Phi \beta = 0. \quad (2.57)$$

Herein, $\|\cdot\|$ denotes the L^2 -norm in \mathbb{R}^3 , $\mathfrak{F} \geq 0$ is the friction coefficient and $\beta \geq 0$ is a scalar parameter. An intuitive physical interpretation of Coulomb's law as described in (2.57) is readily available, too. The first (inequality) condition, commonly referred to as slip condition, requires that the magnitude of the tangential stress \mathbf{t}_τ does not exceed a threshold defined by the coefficient of friction \mathfrak{F} and the normal contact pressure p_n . The frictional response is then

characterized by two physically distinct situations. The stick state, defined by $\beta = 0$, does not allow for any relative tangential movement in the contact zone, i.e. $\mathbf{v}_{\tau,\text{rel}} = \mathbf{0}$. In contrast, the slip state, defined by $\beta > 0$, implicates relative tangential sliding of the two bodies in accordance with the so-called slip rule given as second equation in (2.57). The last equation in (2.57) is again a complementarity condition, here separating the two independent solution branches of stick and slip. A commonly cited similarity of Coulomb's law exists with the most simple formulations of elastoplasticity, see e.g. Simo and Hughes [148]. This similarity is especially interesting in the course of developing numerical algorithms for friction, which usually reuse well-known methodologies from computational inelasticity.

Even more than for normal contact, detailed micromechanical investigations of the real frictional response can be conducted, e.g. concerning the coefficient of friction \mathfrak{F} . Throughout this thesis, \mathfrak{F} is assumed to be a constant reflecting the respective material pairings, and no influence whatsoever of the normal contact pressure, the surface roughness or temperature is taken into account. For further details on constitutive modeling of frictional sliding, the interested reader is exemplarily referred to Kikuchi and Oden [94] and Wriggers [180]. Finally, it is pointed out that frictional response in contact is a path-dependent process, thus introducing mechanical dissipation and making a system representation based on elastic potentials infeasible. Path-dependency can easily be observed in the fact that the tangential contact traction \mathbf{t}_τ depends on the velocity $\mathbf{v}_{\tau,\text{rel}}$ or on the rate of change of the tangential displacement if interpreted incrementally.

2.2.5 Further interface phenomena

Apart from frictional sliding, there exists a manifold of other physical effects taking place at contact interfaces within relevant biological and technical systems. Although these phenomena will not be considered in the numerical procedure developed in this thesis, an exemplary brief overview of lubrication, wear and adhesion is given in the following, along with references to the corresponding literature.

For many technical systems (e.g. certain machine parts), lubrication effects play an important role for the overall mechanical contact behavior, with the regime of elastohydrodynamic lubrication being the industrially most interesting. In such systems, a thin fluid film layer, the so-called lubricant, is established and kept up between the contacting bodies, thus allowing for efficient relative sliding with only very little dissipation. Mathematical and numerical modeling of the lubricant behavior is usually based on the Reynolds equation, which can be derived from the incompressible Navier–Stokes equations under certain simplifying assumptions. A comprehensive summary of elastohydrodynamics is provided in Gohar [56], while state-of-the-art computational approaches are presented in Yang and Laursen [188] and Stupkiewicz [154].

The term wear refers to a variety of phenomena, all related to degradation and removal of material in the context of contact with frictional sliding. Completely different mechanisms can be the driving force behind wear processes, therefore forming categories such as adhesive wear, abrasive wear or corrosive wear. A well-known representative of phenomenological wear models is given by the so-called Holm–Archard law, which allows to quantify the loss of material volume due to an adhesive or abrasive wear process. The physical and mathematical foundations of wear phenomena can be found in Archard [4] and Rabinowicz [137], whereas numerical procedures have been developed in Strömberg et al. [153] and Agelet de Saracibar and Chiumenti [2].

Adhesion describes a physical phenomenon that, at least to some extent, allows for a positive (tensile) contact traction in the normal direction of the interface. Again, this effect can be based on different mechanisms, such as dispersive adhesion or van-der-Waals adhesion based on the Lennard–Jones potential. For further details, the interested reader is referred to Frémond [46] and Raous et al. [138] as well as to the recent computational approach in Sauer [143].

Of course, there also exist other physical effects correlated with contact interaction, e.g. thinking of coupled thermomechanical contact models, which take into account dissipation due to frictional heating (see Hübner and Wohlmuth [81]). While implementations of the aforementioned phenomena are not presented here, it is nevertheless important to point out that the finite element methods for computational contact mechanics proposed within this thesis in principle allow for a quite straightforward integration of such effects. For instance, successful extensions of the FE framework developed herein towards thermomechanical frictional contact and wear processes can be found in Gitterle [54].

2.3 Finite element formulations and solution schemes

This section provides a brief introduction to the numerical treatment of nonlinear solid mechanics problems with finite element methods. Based on a weak formulation of the previously derived IBVP, the FEM for space discretization as well as typical implicit time stepping schemes for time discretization are presented. Furthermore, some associated numerical techniques concerning the solution of nonlinear and linear systems of equations are reviewed. For the sake of simplicity, finite element formulations are only derived for pure solid mechanics problems without contact interaction here. The discretization of mesh tying and unilateral contact problems follows in Chapters 3 and 4, respectively

2.3.1 From strong formulation to weak formulation

Many numerical methods for the solution of partial differential equations, and finite element methods in particular, require a transformation of the IBVP defined in (2.40)–(2.44) within a so-called weak or variational formulation. Although other variational principles exist, the well-known *principle of virtual work (PVW)* is derived exclusively here, with the starting point being a weighted residual notation of the balance equation (2.40) and the traction boundary condition (2.42), i.e.

$$\int_{\Omega_0} (\rho_0 \ddot{\mathbf{u}} - \text{Div} \mathbf{P} - \hat{\mathbf{b}}_0) \cdot \mathbf{w} \, dV_0 + \int_{\Gamma_\sigma} (\mathbf{P} \cdot \mathbf{N} - \hat{\mathbf{t}}_0) \cdot \mathbf{w} \, dA_0 = 0. \quad (2.58)$$

Herein, the weighting or test functions \mathbf{w} are initially arbitrary and can be interpreted as virtual displacements, i.e. $\mathbf{w} = \delta \mathbf{u}$. Since the solution for the displacements is known on the Dirichlet boundary Γ_u , it is required that

$$\mathbf{w} = \mathbf{0} \quad \text{on } \Gamma_u \times [0, T]. \quad (2.59)$$

Applying Gauss divergence theorem and inserting (2.59) and (2.16) yields

$$\int_{\Omega_0} (\rho_0 \ddot{\mathbf{u}} - \hat{\mathbf{b}}_0) \cdot \delta \mathbf{u} \, dV_0 + \int_{\Omega_0} (\text{Grad } \delta \mathbf{u})^\top : (\mathbf{F} \cdot \mathbf{S}) \, dV_0 - \int_{\Gamma_\sigma} \hat{\mathbf{t}}_0 \cdot \delta \mathbf{u} \, dA_0 = 0, \quad (2.60)$$

where the material gradient operator $\text{Grad}(\cdot)$ has been used. As the total variation of the Green–Lagrange strains can be expressed as $\delta \mathbf{E} = \frac{1}{2}((\mathbf{F}^\top \cdot \text{Grad } \delta \mathbf{u})^\top + \mathbf{F}^\top \cdot \text{Grad } \delta \mathbf{u})$ and \mathbf{S} is symmetric, the following simplified notation of the principle of virtual work in material description is obtained:

$$\delta \mathcal{W} := \underbrace{\int_{\Omega_0} \rho_0 \ddot{\mathbf{u}} \cdot \delta \mathbf{u} \, dV_0}_{-\delta \mathcal{W}_{\text{kin}}} + \underbrace{\int_{\Omega_0} \mathbf{S} : \delta \mathbf{E} \, dV_0}_{-\delta \mathcal{W}_{\text{int}}} - \underbrace{\int_{\Omega_0} \hat{\mathbf{b}}_0 \cdot \delta \mathbf{u} \, dV_0 - \int_{\Gamma_\sigma} \hat{\mathbf{t}}_0 \cdot \delta \mathbf{u} \, dA_0}_{-\delta \mathcal{W}_{\text{ext}}} = 0. \quad (2.61)$$

Three distinct contributions to the PVW can be identified. The first term in (2.61) represents the kinetic virtual work contribution $\delta \mathcal{W}_{\text{kin}}$, the second term denotes the internal virtual work contribution $\delta \mathcal{W}_{\text{int}}$, and the third and fourth term together form the virtual work of external loads $\delta \mathcal{W}_{\text{ext}}$. The PVW emerges as a very general principle of solid mechanics, as it does not require the existence of an associated potential \mathcal{W} . As an example, no constitutive assumptions whatsoever enter the weak formulation in (2.61), thus making it also valid and applicable for problems such as elastoplasticity, frictional sliding or non-conservative loading. If an associated potential \mathcal{W} actually exists, the principle of virtual work can alternatively be derived from the so-called principle of minimum of total potential energy (PMTPE), i.e. by total variation of an underlying functional.

It can easily be shown that solutions of the IBVP (i.e. of the strong formulation) also satisfy the weak formulation (2.61). As long as no restrictions are set on the choice of the weighting functions $\delta \mathbf{u}$, the two are formally identical, see e.g. Hughes [86]. However, due to the manipulations introduced above, the weak formulation poses weaker differentiability requirements to the solution functions \mathbf{u} , because only first derivatives of \mathbf{u} with respect to \mathbf{X} appear in (2.61) instead of second derivatives as in (2.40). Thus, the following solution and weighting spaces can be defined:

$$\mathcal{U} = \{ \mathbf{u} \in H^1(\Omega) \mid \mathbf{u}(\mathbf{X}, t) = \hat{\mathbf{u}}(\mathbf{X}, t) \text{ on } \Gamma_u \}, \quad (2.62)$$

$$\mathcal{V} = \{ \delta \mathbf{u} \in H^1(\Omega) \mid \delta \mathbf{u}(\mathbf{X}) = \mathbf{0} \text{ on } \Gamma_u \}. \quad (2.63)$$

Herein, $H^1(\Omega)$ denotes the Sobolev space of functions with square integrable values and first derivatives. While the solution space \mathcal{U} may in general depend on the time t due to a possible time dependency of the Dirichlet boundary conditions, the weighting space \mathcal{V} does not depend on the time t in any way. In conclusion, the weak formulation of the nonlinear solid mechanics problems at hand can be restated as follows: Find $\mathbf{u} \in \mathcal{U}$ such that

$$\delta \mathcal{W} = 0 \quad \forall \delta \mathbf{u} \in \mathcal{V}. \quad (2.64)$$

2.3.2 Space discretization

Space discretization is exclusively considered in the context of finite element methods here. However, as a detailed introduction to all important aspects of the FEM is beyond the scope of this thesis, only the basic ideas and notation will be highlighted. For a more elaborate survey of finite element methods, the reader is again referred to the corresponding literature, e.g. in Bathe [6], Belytschko et al. [10], Hughes [86], Reddy [139], Zienkiewicz and Taylor [196] and Zienkiewicz et al. [197].

Simply speaking, the concept of finite element discretization in this context is based on finding a numerical solution to (2.64) at discrete points, commonly referred to as nodes. The nodes are connected to form elements, which allows to formulate the following approximate partitioning of the domain Ω_0 into nele element subdomains:

$$\Omega_0 \approx \bigcup_{e=1}^{\text{nele}} \Omega_0^{(e)}. \quad (2.65)$$

The displacement solution $\mathbf{u}^{(e)}$ on element e is then typically approximated by local interpolation functions $N_k(\mathbf{X})$, yielding

$$\mathbf{u}^{(e)}(\mathbf{X}, t) \approx \mathbf{u}_h^{(e)}(\mathbf{X}, t) = \sum_{k=1}^{\text{nnod}^{(e)}} N_k(\mathbf{X}) \mathbf{d}_k(t), \quad (2.66)$$

where the discrete nodal values of the displacements $\mathbf{d}_k(t)$ have been introduced. Furthermore, the subscript \cdot_h signifies a spatially discretized quantity throughout this thesis and $\text{nnod}^{(e)}$ represents the number of nodes associated with the element e . The interpolation functions $N_k(\mathbf{X})$, commonly referred to as shape functions, are typically (but not exclusively) low-order polynomials, e.g. Lagrange polynomials, thus meeting the differentiability requirements of the weak form. Based on the so-called isoparametric concept, the element geometry in the reference configuration $\mathbf{X}^{(e)}$ and current configuration $\mathbf{x}^{(e)}$ is approximated using the same shape functions. Typically, $\Omega_0^{(e)}$ is mapped to a reference element geometry or parameter space $\boldsymbol{\xi} = (\xi, \eta, \zeta)$, e.g. the cube $[-1, 1] \times [-1, 1] \times [-1, 1]$, which defines an element Jacobian matrix $\mathbf{J}^{(e)} = \partial \mathbf{X}^{(e)} / \partial \boldsymbol{\xi}$. Thus, the interpolation of displacements, current geometry and reference geometry at the element level is alternatively expressed as

$$\mathbf{u}_h^{(e)}(\boldsymbol{\xi}, t) = \sum_{k=1}^{\text{nnod}^{(e)}} N_k(\boldsymbol{\xi}) \mathbf{d}_k(t), \quad (2.67)$$

$$\mathbf{x}_h^{(e)}(\boldsymbol{\xi}, t) = \sum_{k=1}^{\text{nnod}^{(e)}} N_k(\boldsymbol{\xi}) \mathbf{x}_k(t), \quad (2.68)$$

$$\mathbf{X}_h^{(e)}(\boldsymbol{\xi}) = \sum_{k=1}^{\text{nnod}^{(e)}} N_k(\boldsymbol{\xi}) \mathbf{X}_k, \quad (2.69)$$

with nodal positions \mathbf{X}_k and $\mathbf{x}_k(t)$ in the reference and current configuration, respectively. Depending on the geometric shape and chosen polynomial interpolation order, this results in several well-known finite element types. As an example, for 3D analysis, 8-node hexahedral (*hex8*) elements and 4-node tetrahedral (*tet4*) elements, which are characterized by (tri-)linear shape functions, represent the most popular choice. Finally, time derivatives of the displacements, e.g. the accelerations $\ddot{\mathbf{u}}$, and the weighting functions $\delta \mathbf{u}$ are also interpolated using the same shape functions. The latter convention is commonly referred to as Bubnov–Galerkin approach, as compared with a Petrov–Galerkin approach, where an independent set of shape functions is chosen for interpolating the weighting functions.

Examining (2.67) more closely, it becomes obvious that the finite element method basically introduces restrictions on the solution and weighting spaces defined in (2.62) and (2.63). In the discrete setting, these spaces only contain a finite number of solution and weighting functions, respectively, which is expressed mathematically in terms of finite dimensional subspaces $\mathcal{U}_h \subset \mathcal{U}$ and $\mathcal{V}_h \subset \mathcal{V}$. The limited selection of solution and weighting functions then serves as a basis for the numerical solution, i.e. the weak formulation is recast into a discrete form, which is no longer equivalent to strong and weak formulation, but rather represents an approximation.

The individual contributions to the discretized weak form are integrated element-by-element using Gauss quadrature and then sorted into global vectors based on the so-called assembly operator, which governs the arrangement of local vectorial quantities into global vectors, i.e.

$$\int_{\Omega_0} (\cdot) dV_0 \approx \mathbf{A} \int_{\Omega_{0,h}^{(e)}} (\cdot) dV_0. \quad (2.70)$$

After inserting the interpolations given by (2.67) into the weak formulation (2.61), the final spatially discretized formulation emerges as

$$\delta \mathbf{d}^T (\mathbf{M} \ddot{\mathbf{d}} + \mathbf{f}_{\text{int}}(\mathbf{d}) - \mathbf{f}_{\text{ext}}) = 0, \quad (2.71)$$

with the global mass matrix \mathbf{M} , the global vector of nonlinear internal forces \mathbf{f}_{int} and the global vector of external forces \mathbf{f}_{ext} . Moreover, $\delta \mathbf{d}$, $\ddot{\mathbf{d}}$ and \mathbf{d} are global vectors comprising all discrete nodal values of virtual displacements, accelerations and displacements. Due to the interpolation introduced above, all vectors in (2.71) are of the size $\text{ndof} = \text{ndim} \cdot \text{nnod}$, where nnod is the total number of nodes in the entire domain and ndim is the number of spatial dimensions. The variable name ndof refers to the fact that the discrete values of the nodal displacements \mathbf{d} are also denoted as degrees of freedom. Since (2.71) must hold for arbitrary virtual displacements $\delta \mathbf{d}$, it can equivalently be written as

$$\mathbf{M} \ddot{\mathbf{d}} + \mathbf{f}_{\text{int}}(\mathbf{d}) - \mathbf{f}_{\text{ext}} = \mathbf{0}. \quad (2.72)$$

This defines a system of ndof ordinary differential equations (ODEs), commonly referred to as semi-discrete equations of motion. So far, only space discretization with the finite element method has been established, but the system is still continuous with respect to time. For the sake of completeness, it is pointed out that the well-known and widely used Rayleigh model for viscous damping is readily introduced into (2.72) via

$$\mathbf{M} \ddot{\mathbf{d}} + \mathbf{C} \dot{\mathbf{d}} + \mathbf{f}_{\text{int}}(\mathbf{d}) - \mathbf{f}_{\text{ext}} = \mathbf{0}, \quad (2.73)$$

where $\dot{\mathbf{d}}$ is the global vector of discrete nodal velocities, again resulting from a finite element discretization similar to (2.67). In this context, \mathbf{C} represents the global damping matrix, which for the purely phenomenological Rayleigh damping model can be defined as

$$\mathbf{C} = c_M \mathbf{M} + c_K \mathbf{K}_0, \quad (2.74)$$

i.e. as a linear combination of the mass matrix \mathbf{M} and the initial tangent stiffness matrix \mathbf{K}_0 , which will be introduced in one of the next paragraphs. The scalars c_M and c_K are free parameters, weighting mass and stiffness proportional contributions to the damping matrix.

2.3.3 Time discretization

There exists a large variety of finite difference methods suitable for time discretization of the semi-discrete equations of motion (2.73). In doing so, time derivatives are approximated by their discrete counterparts, the difference quotients. Based on the introduction of a constant time step size Δt , the time interval of interest $t \in [0, T]$ is subdivided into several intervals $[t_n, t_{n+1}]$, where $n \in \mathbb{N}_0$ is the time step index, and thus the spatially discretized displacement solution $\mathbf{d}(t)$ is computed at a series of discrete points in time.

In principle, time integration methods can be divided into implicit and explicit schemes. While implicit methods lead to a fully coupled system of ndof nonlinear discrete algebraic equations for the unknown displacements $\mathbf{d}_{n+1} := \mathbf{d}(t_{n+1})$, explicit methods allow for a direct extrapolation towards \mathbf{d}_{n+1} without requiring a solution step. Throughout this thesis, only implicit schemes will be considered. They represent the method of choice for problems dominated by a low frequency response, while explicit methods are widely used in the context of high frequency responses and wave-like phenomena, e.g. in high velocity impact situations. In general, implicit time integration methods can be shown to be unconditionally stable, thus allowing for relatively large time step sizes as compared with explicit schemes. However, the implementation of implicit methods is more challenging due to the fact that nonlinear solution methods (see Section 2.3.5) including a linearization of the entire finite element formulation are required. For further details on time integration for nonlinear FEM, the reader is referred to Belytschko et al. [10].

Here, the presentation is restricted to one exemplary and widely used implicit time integration scheme, viz. the generalized- α method introduced by Chung and Hulbert [24]. This one-step time integration scheme is based on the well-known Newmark method, which allows for expressing the approximate discrete velocities $\mathbf{v}_{n+1} \approx \dot{\mathbf{d}}(t_{n+1})$ and accelerations $\mathbf{a}_{n+1} \approx \ddot{\mathbf{d}}(t_{n+1})$ at the end of the considered time interval $[t_n, t_{n+1}]$ solely in terms of already known quantities at time t_n and the unknown displacements \mathbf{d}_{n+1} , i.e.

$$\mathbf{v}_{n+1}(\mathbf{d}_{n+1}) = \frac{\gamma}{\beta \Delta t} (\mathbf{d}_{n+1} - \mathbf{d}_n) - \frac{\gamma - \beta}{\beta} \mathbf{v}_n - \frac{\gamma - 2\beta}{2\beta} \Delta t \mathbf{a}_n, \quad (2.75)$$

$$\mathbf{a}_{n+1}(\mathbf{d}_{n+1}) = \frac{1}{\beta \Delta t^2} (\mathbf{d}_{n+1} - \mathbf{d}_n) - \frac{1}{\beta \Delta t} \mathbf{v}_n - \frac{1 - 2\beta}{2\beta} \Delta t \mathbf{a}_n, \quad (2.76)$$

where $\beta \in [0, 1/2]$ and $\gamma \in [0, 1]$ are two parameters characterizing the behavior of the method. The generalized- α method introduces generalized mid-points $t_{n+1-\alpha_m}$ and $t_{n+1-\alpha_f}$ and shifts the evaluation of the individual terms in (2.73) from t_{n+1} to these midpoints. The following linear interpolation rules are commonly established for the generalized- α method:

$$\mathbf{d}_{n+1-\alpha_f} = (1 - \alpha_f) \mathbf{d}_{n+1} + \alpha_f \mathbf{d}_n, \quad (2.77)$$

$$\mathbf{v}_{n+1-\alpha_f} = (1 - \alpha_f) \mathbf{v}_{n+1} + \alpha_f \mathbf{v}_n, \quad (2.78)$$

$$\mathbf{a}_{n+1-\alpha_m} = (1 - \alpha_m) \mathbf{a}_{n+1} + \alpha_m \mathbf{a}_n, \quad (2.79)$$

$$\mathbf{f}_{\text{ext}, n+1-\alpha_f} = (1 - \alpha_f) \mathbf{f}_{\text{ext}, n+1} + \alpha_f \mathbf{f}_{\text{ext}, n}. \quad (2.80)$$

Eventually, the fully (i.e. space and time) discretized finite element formulation of nonlinear solid mechanics, also referred to as discrete linear momentum balance, is obtained as

$$\mathbf{M} \mathbf{a}_{n+1-\alpha_m} + \mathbf{C} \mathbf{v}_{n+1-\alpha_f} + \mathbf{f}_{\text{int}}(\mathbf{d}_{n+1-\alpha_f}) - \mathbf{f}_{\text{ext}, n+1-\alpha_f} = \mathbf{0}. \quad (2.81)$$

One important advantage of the generalized- α method is that it allows for introducing controllable numerical dissipation into the considered system, while at the same time retaining the important properties of unconditional stability and second-order accuracy. Controllable numerical dissipation in this context means that the parameters β , γ , α_m and α_f can be harmonized such that the desired damping effect is only achieved in the spurious high frequency modes, while damping in the low frequency domain is kept at a minimum. This procedure is usually united in the notion of a spectral radius ρ_∞ as the sole free parameter to choose for a generalized- α method. The other parameters then follow directly from the requirements of unconditional stability, second-order accuracy and optimized numerical dissipation as

$$\alpha_m = \frac{2\rho_\infty - 1}{\rho_\infty + 1}, \quad \alpha_f = \frac{\rho_\infty}{\rho_\infty + 1}, \quad \beta = \frac{1}{4}(1 - \alpha_m + \alpha_f)^2, \quad \gamma = \frac{1}{2} - \alpha_m + \alpha_f. \quad (2.82)$$

Note that no numerical dissipation is introduced into the system for the choice $\rho_\infty = 1$. Moreover, the generalized- α method also contains the classical Newmark method as a special case by setting $\alpha_m = \alpha_f = 0$. Finally, an important detail of the generalized- α method is the way the internal forces $\mathbf{f}_{\text{int}}(\mathbf{d}_{n+1-\alpha_f})$ are evaluated. The first alternative is based on a trapezoidal type of interpolation of the internal forces, i.e.

$$\mathbf{f}_{\text{int}}(\mathbf{d}_{n+1-\alpha_f}) = (1 - \alpha_f)\mathbf{f}_{\text{int}}(\mathbf{d}_{n+1}) + \alpha_f\mathbf{f}_{\text{int}}(\mathbf{d}_n), \quad (2.83)$$

while the second alternative is based on a midpoint type of approach, i.e.

$$\mathbf{f}_{\text{int}}(\mathbf{d}_{n+1-\alpha_f}) = \mathbf{f}_{\text{int}}((1 - \alpha_f)\mathbf{d}_{n+1} + \alpha_f\mathbf{d}_n). \quad (2.84)$$

Obviously, the two definitions coincide in the context of linear problems, i.e. for infinitesimal deformations and linear elastic material behavior. While there is no preferred choice in nonlinear solid mechanics, the first version defined in (2.83) will be employed throughout this thesis.

For the sake of completeness, a short outlook on energy-momentum conserving time integration schemes is also given here. While the conservation of all relevant quantities (i.e. energy, linear and angular momentum) cannot be guaranteed for most methods, including the presented generalized- α method, energy-momentum schemes provide an exact algorithmic conservation by construction. The most popular representative of this class of methods is the energy-momentum method (EMM) proposed by Simo and Tarnow [150]. The EMM was further analyzed in Gonzalez [57] and extended towards the so-called generalized energy-momentum method (GEMM) in Kuhl and Ramm [101] and Kuhl and Crisfield [100] with the possibility of optimized numerical dissipation. Basically, the only difference of the GEMM as compared with the generalized- α method explained above is the way the internal forces $\mathbf{f}_{\text{int}}(\mathbf{d}_{n+1-\alpha_f})$ are evaluated, cf. Simo and Tarnow [150] for the simple case of a Saint-Venant–Kirchhoff material model and Gonzalez [57] for a generalization to arbitrary hyperelastic materials.

Due to several reasons, time integration in general and energy-momentum conservation in particular is an intricate issue for systems including contact constraints. Although not in the focus of interest here, some remarks will be given in the context of mortar finite element discretization for mesh tying systems in 3.2.4 and contact systems in 4.2.5.

Moreover, it is pointed out that quasistatic problems, i.e. neglecting inertia effects, are also considered in the following. In that case, the time parameter t only plays the role of a pseudo-time and no time integration method is needed, but the quasistatic solution is rather computed as a series of static equilibrium states.

2.3.4 Spatial and temporal convergence

Finally, the important notions of spatial and temporal convergence rates of finite element formulations should be mentioned, i.e. the rate of convergence that is obtained when uniform mesh refinement is applied or when the time step Δt is reduced, respectively. Accuracy in space mainly depends on the polynomial order of the employed finite element interpolation. Measuring the discretization error of the displacements in the L^2 -norm, i.e.

$$\|\mathbf{u} - \mathbf{u}_h\| = \sqrt{\int_{\Omega_0} (\mathbf{u} - \mathbf{u}_h) \cdot (\mathbf{u} - \mathbf{u}_h) dV_0}, \quad (2.85)$$

one readily obtains the result that a polynomial expansion of degree p within finite elements of characteristic size h leads to a convergence order of $p + 1$ for the solid mechanics problems considered here, see e.g. Hughes [86]. Typically, the compact notation $\mathcal{O}(h^{p+1})$ is employed. Quite commonly, the discretization error is also measured in the H^1 -norm or the energy norm, the latter of which is defined for hyperelastic materials as

$$\|\mathbf{u} - \mathbf{u}_h\|_{\text{energy}} = \sqrt{\int_{\Omega_0} \Psi(\mathbf{E} - \mathbf{E}_h) dV_0}, \quad (2.86)$$

with a strain energy function $\Psi(\mathbf{E})$ as introduced in (2.18). For both H^1 -norm and energy norm, only a convergence rate of order $\mathcal{O}(h^p)$ can be expected due to the fact that their evaluation involves derivatives of the displacements. Thus, as an example, finite elements based on linear shape functions as introduced above show a convergence rate $\mathcal{O}(h^2)$ in the L^2 -norm and $\mathcal{O}(h)$ in the H^1 -norm and energy norm.

Accuracy in time is basically a characteristic property of the employed time integration scheme for the second-order ODEs resulting from finite element discretization. As mentioned before, the generalized- α method used throughout this thesis is second-order accurate in time, or in other words the resulting fully discrete system (2.81) features a temporal convergence rate $\mathcal{O}(\Delta t^2)$ for the displacements measured in the L^2 -norm.

2.3.5 Linearization and solution techniques for nonlinear equations

Within each time step, the system of ndof nonlinear discrete algebraic equations (2.81) needs to be solved for the unknown displacements \mathbf{d}_{n+1} . Throughout this thesis, the Newton–Raphson method is employed as an iterative nonlinear solution technique. Within each iteration step i , the residual of the discrete linear momentum balance can be defined as

$$\mathbf{r}_{\text{effdyn}}(\mathbf{d}_{n+1}^i) = \mathbf{M}\mathbf{a}_{n+1-\alpha_m}^i + \mathbf{C}\mathbf{v}_{n+1-\alpha_f}^i + \mathbf{f}_{\text{int}}(\mathbf{d}_{n+1-\alpha_f}^i) - \mathbf{f}_{\text{ext},n+1-\alpha_f}. \quad (2.87)$$

The Newton–Raphson method is based on repeated linearization of the residual in (2.87), solution of the resulting linearized system of equations and incremental update of the unknown

displacements until a user-defined convergence criterion is met. At first, the linearization is obtained from the truncated Taylor expansion of (2.87), viz.

$$\text{Lin } \mathbf{r}_{\text{effdyn}}(\mathbf{d}_{n+1}^i) = \mathbf{r}_{\text{effdyn}}(\mathbf{d}_{n+1}^i) + \underbrace{\left. \frac{\partial \mathbf{r}_{\text{effdyn}}(\mathbf{d}_{n+1})}{\partial \mathbf{d}_{n+1}} \right|}_{\mathbf{K}_{\text{effdyn}}(\mathbf{d}_{n+1}^i)}^i \Delta \mathbf{d}_{n+1}^{i+1}, \quad (2.88)$$

where the partial derivative of $\mathbf{r}_{\text{effdyn}}(\mathbf{d}_{n+1}^i)$ with respect to the displacements is commonly referred to as dynamic effective tangential stiffness matrix $\mathbf{K}_{\text{effdyn}}(\mathbf{d}_{n+1}^i)$ of size $\text{ndof} \times \text{ndof}$. In the context of the generalized- α method, the dynamic effective tangential stiffness matrix can be determined based on Newmark's approximation given in (2.75) and (2.76) and the generalized midpoints defined in (2.77)–(2.80), yielding

$$\begin{aligned} \mathbf{K}_{\text{effdyn}}(\mathbf{d}_{n+1}^i) &= \left. \frac{\partial \mathbf{r}_{\text{effdyn}}(\mathbf{d}_{n+1})}{\partial \mathbf{d}_{n+1}} \right|_i = \\ &= \left[\frac{1 - \alpha_m}{\beta \Delta t^2} \mathbf{M} + \frac{(1 - \alpha_f) \gamma}{\beta \Delta t} \mathbf{C} + (1 - \alpha_f) \mathbf{K}_T(\mathbf{d}_{n+1 - \alpha_f}) \right]^i, \end{aligned} \quad (2.89)$$

where $\mathbf{K}_T(\mathbf{d}_{n+1 - \alpha_f})$ is the tangential stiffness matrix associated with the internal forces as

$$\mathbf{K}_T(\mathbf{d}_{n+1 - \alpha_f}) = \frac{\partial \mathbf{f}_{\text{int}}(\mathbf{d}_{n+1 - \alpha_f})}{\partial \mathbf{d}_{n+1 - \alpha_f}}. \quad (2.90)$$

To sum up, the Newton–Raphson method provides an iterative procedure for finding the unknown solution \mathbf{d}_{n+1} for which the residual $\mathbf{r}_{\text{effdyn}}(\mathbf{d}_{n+1})$ vanishes. Within each iteration, it is required that

$$\text{Lin } \mathbf{r}_{\text{effdyn}}(\mathbf{d}_{n+1}^i) \stackrel{!}{=} \mathbf{0}, \quad (2.91)$$

or in other words, the following linear system of equations has to be solved:

$$\mathbf{K}_{\text{effdyn}}(\mathbf{d}_{n+1}^i) \Delta \mathbf{d}_{n+1}^{i+1} = -\mathbf{r}_{\text{effdyn}}(\mathbf{d}_{n+1}^i). \quad (2.92)$$

Having solved (2.92), the displacements \mathbf{d}_{n+1}^{i+1} at the end of the time step can be updated via

$$\mathbf{d}_{n+1}^{i+1} = \mathbf{d}_{n+1}^i + \Delta \mathbf{d}_{n+1}^{i+1}, \quad (2.93)$$

and the iteration counter is increased by one, i.e. $i \rightarrow i + 1$. The procedure in (2.92) and (2.93) is repeated until a certain user-defined convergence criterion, usually with regard to the L^2 -norm of the residual $\|\mathbf{r}_{\text{effdyn}}(\mathbf{d}_{n+1}^i)\|$, is met. The most advantageous property of the Newton–Raphson method is its local quadratic convergence. This means that if the start solution estimate \mathbf{d}_{n+1}^0 is sufficiently close to the actual solution \mathbf{d}_{n+1} , i.e. within the problem-dependent convergence radius, then the residual norm approaches zero with a quadratic convergence rate.

In this thesis, only exact Newton–Raphson methods are considered as described above or later also their semi-smooth variants for the inclusion of contact constraints. However, the computational cost associated with such an approach can be considerable for nonlinear solid mechanics problems, bearing in mind that it requires a consistent linearization and thus a determination

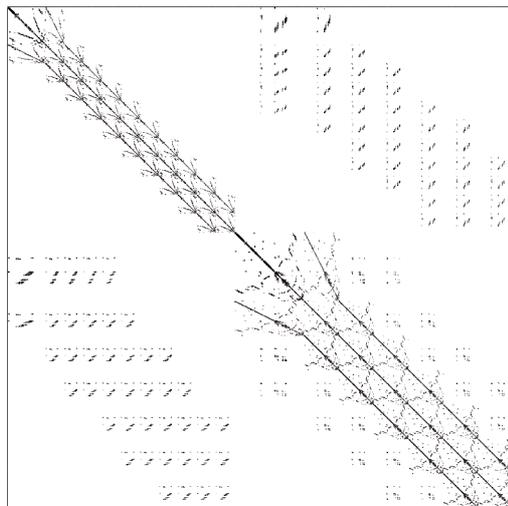


Figure 2.6: Exemplary sparsity pattern of system matrix \mathbf{A} resulting from mortar finite element discretization of a tied contact problem, where black dots represent non-zero entries.

of the tangential stiffness matrix $\mathbf{K}_T(\mathbf{d}_{n+1-\alpha_f})$ within *each* iteration step. In practice, this often leads to the application of quasi-Newton methods or modified Newton methods, which are based on a computationally cheaper approximation of the stiffness matrix (e.g. via secants), but sacrifice optimal convergence behavior. Apart from that, many extensions of the Newton–Raphson method aim at enlarging its local convergence radius. Popular examples of such globalization strategies are line search methods and the pseudo-transient continuation (PTC) technique, see e.g. Gee et al. [48] and references therein.

2.3.6 Solution techniques for linear equations

Another important aspect associated with finite element formulations for both linear and nonlinear solid mechanics is an efficient solution technique for the resulting linear systems of equations. Especially for large finite element models, the linear solver steps often account for the computationally most expensive part of the overall solution process within each time step. Whereas the specific structure and appearance of the system to be solved here has been presented in (2.92), it can also be recast in the following rather abstract notation instead:

$$\mathbf{Ax} = \mathbf{b}, \quad (2.94)$$

with \mathbf{A} representing the square system matrix, \mathbf{b} the so-called right-hand side vector and \mathbf{x} containing the discrete unknowns. As a consequence of finite element discretization with its locally defined shape functions, \mathbf{A} will typically be a sparse matrix with a pronounced band structure, see Figure 2.6 for an exemplary visualization of a resulting sparsity pattern. There exists an abundant amount of literature and countless different methods for solving sparse linear systems as given in (2.94). Since an in-depth investigation of such techniques is far beyond the scope of this thesis, only a brief overview and classification of methods is given in the following. For details, the reader is exemplarily referred to the comprehensive book by Quarteroni et al. [136].

Direct solution techniques, for instance based on an LU-decomposition of the underlying system matrix \mathbf{A} , represent the first class of methods. Unfortunately, the applicability of direct solution techniques is usually limited by the system size, as they become inefficient and too memory consuming for very large systems due to an inevitable “fill-in” effect generated during factorization and Gaussian elimination. In particular, an efficient parallelization of direct solution techniques is in general not straightforward.

The second large class of methods is given by preconditioned iterative solution techniques, in particular so-called Krylov subspace methods such as the conjugate gradient (CG) method or the generalized minimal residual (GMRES) method, cf. Saad [141]. As the efficiency of iterative linear solvers is largely governed by the condition number, i.e. the ratio of the maximum and minimum eigenvalue of the system matrix \mathbf{A} , preconditioning techniques play a decisive role. Simply speaking, the idea of preconditioning is to transform the linear system (2.94) into an equivalent system with reduced condition number by scaling it with a suitable linear operator. In many cases, preconditioning is a prerequisite for iterative linear solvers to be applicable at all. Among the many preconditioners available, only some important ones will be mentioned here, e.g. the Jacobi and Gauss–Seidel methods, incomplete factorization (ILU) methods and algebraic multigrid (AMG) methods. Moreover, domain decomposition (DD) methods, see e.g. Smith et al. [152] and Section 2.5, are commonly used as preconditioners for iterative Krylov methods. Yet, to avoid confusion with respect to naming, it should be pointed out that DD will exclusively be considered as non-conforming discretization technique in this thesis.

Preconditioned iterative solvers represent today’s state-of-the-art for an efficient solution of very large linear systems of equations. However, they are usually tailored for specific properties of the underlying system matrix \mathbf{A} and therefore show a non-optimal performance when applied to certain linear system types without any modification. Contact mechanics problems, for instance, typically require specific preconditioners taking into account the additional constraint conditions. While efficient preconditioners for contact systems have recently evolved into a quite active field of research (see e.g. Krause [98], Wohlmuth and Krause [178]), they are not in the focus of interest of this thesis. Instead, well-established direct and iterative solution techniques will primarily be employed here.

2.4 Methods of constraint enforcement

As has been explained in Section 2.2, both mesh tying and frictionless or frictional contact problems are characterized by additional constraint conditions that must be incorporated into the finite element formulation. Thus, in contrast to the unconstrained solid mechanics case, contact problems must rather be interpreted as problems of constrained optimization. With the associated constraint enforcement techniques playing an important role for the development of numerical algorithms in this thesis, the idea of this paragraph is to present the most widely used methods in a very general manner. A more extensive overview is to be found in Luenberger and Ye [112], and the reader is also referred to Laursen [107] for a very illustrative introduction in the context of contact problems.

For the sake of simplicity, a conservative problem of nonlinear solid mechanics is assumed for the following derivations so that the discrete displacement solution \mathbf{d} is found by minimization of an underlying functional $\mathcal{W}(\mathbf{d})$ based on the principle of minimum of total potential energy,

see also Section 2.3.1. Nevertheless, the presented methods are equally applicable for problems involving dissipation, e.g. due to elastoplastic material behavior or frictional sliding in contact. The abstract minimization problem statement then yields

$$\min \mathcal{W}(\mathbf{d}) \text{ subject to } \mathbf{G}(\mathbf{d}) = \mathbf{0}, \quad (2.95)$$

where $\mathbf{G}(\mathbf{d}) = \mathbf{0}$ exemplarily represents a vector-valued discrete equality condition, which contains n_{co} scalar constraints to the system. Some remarks on the treatment of inequality constraints will be given in Section 2.4.4.

2.4.1 Lagrange multiplier method

The Lagrange multiplier method introduces a vector of additional unknowns λ , the so-called discrete Lagrange multipliers, for constraint enforcement. The global size of this vector is n_{co} , i.e. it is identical to the number of constraints. Adding the Lagrange multiplier contributions to the underlying elastic potential $\mathcal{W}(\mathbf{d})$ results in the following extended overall potential:

$$\mathcal{W}_{LM}(\mathbf{d}, \lambda) = \mathcal{W}(\mathbf{d}) + \sum_{j=1}^{n_{co}} \lambda_j G_j(\mathbf{d}). \quad (2.96)$$

With the underlying potential $\mathcal{W}(\mathbf{d})$ being convex and thus representing a positive definite operator with unique minimum, the solution to (2.96) constitutes a saddle point of the extended potential $\mathcal{W}_{LM}(\mathbf{d}, \lambda)$. Specifically, this means that the solution is a maximum of \mathcal{W}_{LM} with respect to the Lagrange multipliers λ and a minimum with respect to the displacements \mathbf{d} , see e.g. Kikuchi and Oden [94] or Luenberger and Ye [112] for details. Total variation of (2.96) leads to a mixed variational formulation with the stationary point to be determined by

$$\delta \mathcal{W}_{LM}(\mathbf{d}, \lambda) = \delta \mathbf{d}^T \left(\frac{\partial \mathcal{W}(\mathbf{d})}{\partial \mathbf{d}} + \sum_{j=1}^{n_{co}} \lambda_j \frac{\partial G_j(\mathbf{d})}{\partial \mathbf{d}} \right) + \delta \lambda^T \mathbf{G}(\mathbf{d}) \stackrel{!}{=} 0. \quad (2.97)$$

The corresponding linearized problem statement is given by

$$\begin{bmatrix} \mathbf{K}_{LM}^d + \mathbf{K}_{LM}^\lambda & \mathbf{C}^T \\ \mathbf{C} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{d} \\ \Delta \lambda \end{bmatrix} = \begin{bmatrix} -\mathbf{r}_{LM}^d \\ -\mathbf{r}_{LM}^\lambda \end{bmatrix}. \quad (2.98)$$

Herein, the saddle point structure of the system with its typical zero block on the main diagonal can clearly be observed. The right-hand sides of the Lagrange multiplier system are defined as

$$\mathbf{r}_{LM}^d = \frac{\partial \mathcal{W}(\mathbf{d})}{\partial \mathbf{d}} + \sum_{j=1}^{n_{co}} \lambda_j \frac{\partial G_j(\mathbf{d})}{\partial \mathbf{d}}, \quad \mathbf{r}_{LM}^\lambda = \mathbf{G}(\mathbf{d}), \quad (2.99)$$

and the matrix blocks are given as

$$\mathbf{K}_{LM}^d = \frac{\partial^2 \mathcal{W}(\mathbf{d})}{\partial \mathbf{d}^2}, \quad \mathbf{K}_{LM}^\lambda = \sum_{j=1}^{n_{co}} \lambda_j \frac{\partial^2 G_j(\mathbf{d})}{\partial \mathbf{d}^2}, \quad \mathbf{C} = \frac{\partial \mathbf{G}(\mathbf{d})}{\partial \mathbf{d}}. \quad (2.100)$$

While the Lagrange multiplier method allows for an exact satisfaction of the given discrete constraints $\mathbf{G}(\mathbf{d}) = \mathbf{0}$, it entails an undesirable increase in global system size on the other hand. Concretely, the linearized system in (2.98) consists of $\text{ndof} + \text{nco}$ equations, where ndof again refers to the total number of primary degrees of freedom (i.e. the number of discrete nodal displacements in \mathbf{d}) and nco is the number of constraints. Even more severe is the fact that the saddle point structure of the system matrix in (2.98) usually poses difficulties to state-of-the-art iterative linear solvers. Nevertheless, the Lagrange multiplier approach will be the method of choice for constraint enforcement throughout this thesis. The mentioned deficiencies will be overcome by applying special concepts and numerical procedures, e.g. the so-called dual Lagrange multiplier interpolation to be introduced in Sections 3.3 and 4.3.

2.4.2 Penalty method

The penalty method is another very widely used approach for constraint enforcement. The basic idea behind this method is to remove the constraints explicitly from the system and introduce a penalization of any constraint violation instead. This is typically reflected in an additional penalty potential contribution, which together with the underlying elastic potential $\mathcal{W}(\mathbf{d})$ forms the following extended overall potential:

$$\mathcal{W}_P(\mathbf{d}) = \mathcal{W}(\mathbf{d}) + \frac{1}{2}\epsilon \sum_{j=1}^{\text{nco}} G_j(\mathbf{d})G_j(\mathbf{d}), \quad (2.101)$$

where ϵ is the user-defined and problem-specific penalty parameter. It can easily be observed that $\mathcal{W}_P(\mathbf{d})$ does not depend on any additional unknowns, as has been the case for the Lagrange multiplier method, but rather still represents a problem of unconstrained optimization. Simply speaking, the penalization in (2.101) works such that it associates large energies with displacement solutions \mathbf{d} violating the constraints $\mathbf{G}(\mathbf{d}) = \mathbf{0}$. Total variation of (2.101) leads to a purely displacement-based formulation here, given as

$$\delta\mathcal{W}_P(\mathbf{d}) = \delta\mathbf{d}^T \left(\frac{\partial\mathcal{W}(\mathbf{d})}{\partial\mathbf{d}} + \epsilon \sum_{j=1}^{\text{nco}} G_j(\mathbf{d}) \frac{\partial G_j(\mathbf{d})}{\partial\mathbf{d}} \right) \stackrel{!}{=} 0. \quad (2.102)$$

Linearization yields the following discrete matrix vector system:

$$\mathbf{K}_P^d \Delta\mathbf{d} = -\mathbf{r}_P^d. \quad (2.103)$$

It can easily be seen that the system size remains unchanged at ndof , despite taking into account the nco constraints. The right-hand side of the penalty system is defined as

$$\mathbf{r}_P^d = \frac{\partial\mathcal{W}(\mathbf{d})}{\partial\mathbf{d}} + \epsilon \sum_{j=1}^{\text{nco}} G_j(\mathbf{d}) \frac{\partial G_j(\mathbf{d})}{\partial\mathbf{d}}, \quad (2.104)$$

and the system matrix is given as

$$\mathbf{K}_P^d = \frac{\partial^2\mathcal{W}(\mathbf{d})}{\partial\mathbf{d}^2} + \epsilon \sum_{j=1}^{\text{nco}} G_j(\mathbf{d}) \frac{\partial^2 G_j(\mathbf{d})}{\partial\mathbf{d}^2} + \epsilon \sum_{j=1}^{\text{nco}} \frac{\partial G_j(\mathbf{d})}{\partial\mathbf{d}} \frac{\partial G_j(\mathbf{d})}{\partial\mathbf{d}}. \quad (2.105)$$

The extended potential $\mathcal{W}_P(\mathbf{d})$ retains its convexity, if the underlying elastic potential is convex. Thus, the penalty system to be solved is usually positive definite and bears large similarities with the original unpenalized system, which is a very desirable property with regard to the unmodified re-use of certain numerical techniques (e.g. linear solvers).

However, penalty methods suffer from one serious drawback, viz. their dependency on the penalty parameter ϵ . It can easily be shown that the exact solution obtained with a Lagrange multiplier method is only recovered for the limit case $\epsilon \rightarrow \infty$. This situation inevitably generates a dilemma: It is not possible to choose very large penalty parameters because the resulting systems become more and more ill-conditioned. On the other hand, finite values of the penalty parameter always allow for a certain violation of the given constraints, which might become unacceptably large. However, with ϵ being problem-specific, no universally valid rule can be established for its determination. Nevertheless, penalty methods are widely used in practice, mostly due to their easy and efficient implementation. Apart from contact and friction problems, applications in solid mechanics for instance also include the penalization of the volume conservation constraint in nearly incompressible elasticity (with Poisson's ratio $\nu \rightarrow 0.5$).

2.4.3 Augmented Lagrange method

Simply speaking, the idea of the Augmented Lagrange approach is to find an optimal compromise between Lagrange multiplier and penalty methods, allowing for an exact enforcement of constraints in combination with a penalty-like regularization for easier numerical treatment. Similar to Lagrange multiplier methods, a vector of n_{co} additional unknowns $\boldsymbol{\lambda}$ is introduced, and the overall Augmented Lagrange functional then reads as follows:

$$\mathcal{W}_{AL}(\mathbf{d}, \boldsymbol{\lambda}) = \mathcal{W}(\mathbf{d}) + \sum_{j=1}^{n_{co}} \frac{1}{2\epsilon} (\lambda_j + \epsilon G_j(\mathbf{d}))^2 - \frac{1}{2\epsilon} \lambda_j^2. \quad (2.106)$$

Comparing (2.106) with (2.96) and (2.101), it becomes quite clear that the Augmented Lagrange approach basically only combines the Lagrange multiplier and penalty methods. The extended functional contains both a Lagrange multiplier and a penalty contribution. Consequently, total variation of (2.106) gives

$$\delta \mathcal{W}_{AL}(\mathbf{d}, \boldsymbol{\lambda}) = \delta \mathbf{d}^\top \left(\frac{\partial \mathcal{W}(\mathbf{d})}{\partial \mathbf{d}} + \sum_{j=1}^{n_{co}} (\lambda_j + \epsilon G_j(\mathbf{d})) \frac{\partial G_j(\mathbf{d})}{\partial \mathbf{d}} \right) + \delta \boldsymbol{\lambda}^\top \mathbf{G}(\mathbf{d}) \stackrel{!}{=} 0. \quad (2.107)$$

This equation can be examined in more detail, revealing that the penalty term vanishes in the case of an exact satisfaction of the constraints $\mathbf{G}(\mathbf{d}) = \mathbf{0}$. Thus, the Augmented Lagrange formulation reduces to its Lagrange multiplier counterpart in the limit case, meaning that it has exactly the same solution as the Lagrange multiplier method. Nevertheless, the approach in (2.107) comes with the important advantage that it allows for an easier numerical treatment than a pure Lagrange multiplier method. Constraint violation is penalized like for penalty methods, but this is achieved without changing the solution, even for finite values of the penalty parameter ϵ .

In practice, two different implementations of the Augmented Lagrange method exist. As this aspect is a regular source of confusion, especially in the context of contact mechanics (see Section 4.4.2), a short explanation is given here. On the one hand, it is possible to keep the Lagrange

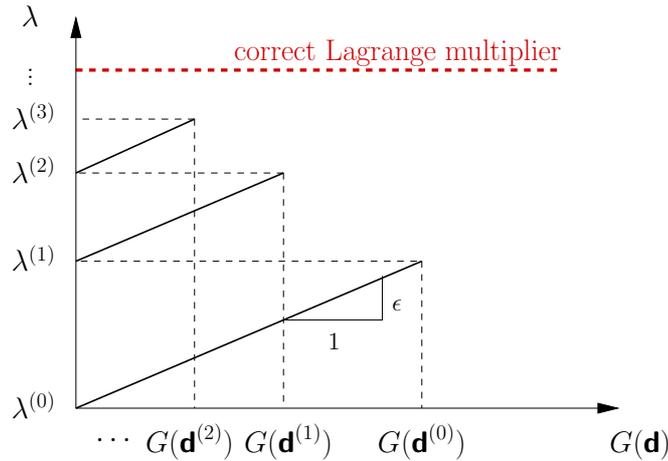


Figure 2.7: Schematic diagram of Uzawa’s algorithm for the Augmented Lagrange method with one single scalar constraint equation $G(\mathbf{d}) = 0$.

multipliers λ as additional unknowns and apply standard Newton–Raphson iteration as nonlinear solution method. As for the Lagrange multiplier method, this leads to an undesirable saddle point formulation, however a certain “convexification” (cf. Laursen [107]) of the underlying potential around the solution is achieved. The linearized problem statement is not explicitly given here, as it constitutes a mere combination of contributions already outlined in Sections 2.4.1 and 2.4.2.

On the other hand, the well-known Uzawa algorithm is commonly used as an alternative for solving (2.107). In that case, the Lagrange multipliers λ are assumed to be given quantities and thus they can be removed as additional unknowns. Instead, the Augmented Lagrange system is solved for the displacements \mathbf{d} only. This naturally defines an iterative procedure, i.e. an augmentation loop with iteration index k , for approaching the exact Lagrange multiplier solution. Therein, the penalty approach is used as kernel and the Lagrange multipliers $\lambda^{(k)}$ are fixed within each iteration step, see Figure 2.7. Usually, the procedure starts with an initial guess $\lambda^{(0)} = \mathbf{0}$, making the first augmentation step *identical* to the ordinary penalty method. An update of the Lagrange multipliers is then obtained via

$$\lambda^{(k+1)} = \lambda^{(k)} + \epsilon \mathbf{G}(\mathbf{d}^{(k)}). \quad (2.108)$$

Again, a linearized problem statement is not explicitly given due to its great similarity with the penalty case in (2.103). The outer iteration on the Lagrange multiplier $\lambda^{(k)}$ is repeated until a user-defined convergence criterion, usually monitoring the remaining violation of the constraints $\mathbf{G}(\mathbf{d}^{(k)}) = \mathbf{0}$ in a suitable norm, is met. As Uzawa’s algorithm only guarantees a linear convergence rate, heuristic acceleration techniques are often applied.

2.4.4 Treatment of inequality constraints

With the focus of this thesis being on unilateral contact problems, suitable techniques for the enforcement of inequality constraints need to be considered, too. First, the abstract optimization problem (2.95) is extended to the case of inequality constraints, yielding

$$\min \mathcal{W}(\mathbf{d}) \text{ subject to } \mathbf{G}(\mathbf{d}) \leq \mathbf{0}, \quad (2.109)$$

where $\mathcal{W}(\mathbf{d})$ is again the underlying elastic potential, and $\mathbf{G}(\mathbf{d}) \leq \mathbf{0}$ now represents a vector-valued condition with n_{co} scalar inequality constraints to the system. In principle, all constraint enforcement techniques described in the previous paragraphs for solving (2.95) can also be applied for solving (2.109). The following explanations will initially concentrate on the Lagrange multiplier method. It is well-known from optimization theory that, based on Lagrange multipliers λ , the problem (2.109) can be expressed as the following variational inequality:

$$\delta \mathbf{d}^\top \left(\frac{\partial \mathcal{W}(\mathbf{d})}{\partial \mathbf{d}} + \sum_{j=1}^{n_{co}} \lambda_j \frac{\partial G_j(\mathbf{d})}{\partial \mathbf{d}} \right) \stackrel{!}{=} 0, \quad (2.110)$$

$$(\delta \lambda_j - \lambda_j) G_j(\mathbf{d}) \stackrel{!}{\leq} 0, \quad j = 1, \dots, n_{co}, \quad (2.111)$$

where only variations $\delta \lambda_j \geq 0$ are admissible, see e.g. Laursen [107]. Moreover, (2.110) is a mere restatement of the stationarity requirement with respect to the displacements \mathbf{d} contained in (2.97). Replacing the variational inequality in (2.111) by the corresponding optimality conditions and taking into account the fact that the variations are arbitrary leads to

$$G_j(\mathbf{d}) \leq 0, \quad \lambda_j \geq 0, \quad \lambda_j G_j(\mathbf{d}) = 0, \quad j = 1, \dots, n_{co}, \quad (2.112)$$

where the typical set of Karush–Kuhn–Tucker conditions for inequality constraints arises, see e.g. Luenberger and Ye [112]. As compared with before, now so-called *active set strategies* are needed to deal with the additional complexity introduced by the inequality constraints. As a first step, the notions of an active set of constraints \mathcal{A} and of an inactive set of constraints \mathcal{I} can readily be defined as

$$\mathcal{A} = \{ j \mid G_j(\mathbf{d}) = 0, j = 1, \dots, n_{co} \}, \quad (2.113)$$

$$\mathcal{I} = \{ j \mid G_j(\mathbf{d}) < 0, j = 1, \dots, n_{co} \}, \quad (2.114)$$

with the obvious properties $\mathcal{A} \cap \mathcal{I} = \emptyset$ and $|\mathcal{A}| + |\mathcal{I}| = n_{co}$. In the simplest possible approach, this defines an iterative scheme (iteration index k) based on initial guesses $\mathcal{A}^{(0)}$ and $\mathcal{I}^{(0)}$ and a subsequent solution of (2.109) as a series of problems with $|\mathcal{A}^{(k)}|$ equality constraints, viz.

$$\min \mathcal{W}(\mathbf{d}) \text{ subject to } \mathbf{G}_{\mathcal{A}^{(k)}}(\mathbf{d}) = \mathbf{0}, \quad (2.115)$$

where $\mathbf{G}_{\mathcal{A}^{(k)}}(\mathbf{d})$ is the subset of all constraints contained in the current active set $\mathcal{A}^{(k)}$. At the end of each active set iteration, all constraints are checked and the sets are updated according to possible constraint violations. Concretely, this means that an inactive constraint j is included into the active set $\mathcal{A}^{(k+1)}$ if $G_j(\mathbf{d}^{(k)}) \geq 0$, and inversely an active constraint is moved to $\mathcal{I}^{(k+1)}$ if $G_j(\mathbf{d}^{(k)}) < 0$. This procedure of solving system (2.115) and updating the sets is repeated until convergence, i.e. until $\mathcal{A}^{(k+1)} = \mathcal{A}^{(k)}$. Although easy to implement, such a “naive” active set strategy is undesirable especially for nonlinear problems, as it leads to two nested iteration loops in that case, with the inner loop solving the nonlinear finite element formulation using a Newton–Raphson method and the outer loop solving for the correct active set.

However, there exists an attractive alternative specifically suited for KKT type inequality constraints, which are also typical in contact mechanics, see (2.55). The so-called primal–dual active

set strategies (PDASS) are based on a reformulation of the KKT conditions as *one* equivalent equality constraint, commonly referred to as nonlinear complementarity (NCP) function:

$$C_j(\lambda_j, \mathbf{d}) = \lambda_j - \max(0, \lambda_j + cG_j(\mathbf{d})) = 0, \quad (2.116)$$

where $c > 0$ is the so-called complementarity parameter. Detailed investigations regarding such active set strategies and their algorithmic realization as semi-smooth Newton methods will follow in Section 4.4. Exemplarily, the reader is also referred to Alart and Curnier [3], Christensen et al. [23], Hübner and Wohlmuth [80] and Hübner et al. [84]. Simply speaking, the PDASS allows for an integration of all possible nonlinearities, including finite deformations, nonlinear material behavior and, most notably, the search for the correct active set \mathcal{A} into one *single* nonlinear iteration scheme. It is well-known from the corresponding mathematical literature, e.g. Hintermüller et al. [77] and Qi and Sun [135] that the resulting semi-smooth (or generalized) Newton methods show a superlinear local rate of convergence.

For the sake of completeness, the treatment of inequality constraints in the context of penalty-based methods is briefly outlined here, too. Due to the explicit removal of constraints, which are replaced by an ad-hoc penalization of any occurring constraint violation, inequality constraints are more straightforward for penalty methods. Basically, the distinction between active and inactive constraints can easily be incorporated into the penalty potential (2.101) as

$$\mathcal{W}_P(\mathbf{d}) = \mathcal{W}(\mathbf{d}) + \frac{1}{2}\epsilon \sum_{j=1}^{\text{nco}} \langle G_j(\mathbf{d}) \rangle \langle G_j(\mathbf{d}) \rangle, \quad (2.117)$$

where $\langle \cdot \rangle$ denotes the Macaulay bracket defined as

$$\langle x \rangle := \begin{cases} x & \text{if } x > 0, \\ 0 & \text{if } x \leq 0. \end{cases} \quad (2.118)$$

Using this simple modification, only a violation of the inequality constraints $\mathbf{G}(\mathbf{d}) \leq \mathbf{0}$ is penalized, thus making the penalty method likewise applicable to problems with inequality constraints. Total variation of (2.117) leads to the final penalty formulation for inequality constraints:

$$\delta \mathcal{W}_P(\mathbf{d}) = \delta \mathbf{d}^\top \left(\frac{\partial \mathcal{W}(\mathbf{d})}{\partial \mathbf{d}} + \epsilon \sum_{j=1}^{\text{nco}} \langle G_j(\mathbf{d}) \rangle \frac{\partial G_j(\mathbf{d})}{\partial \mathbf{d}} \right) \stackrel{!}{=} 0. \quad (2.119)$$

An extension of the presented ideas to the enforcement of inequality constraints using the Augmented Lagrange method is also straightforward. Including the distinction between active and inactive constraints similarly as above yields

$$\mathcal{W}_{AL}(\mathbf{d}, \boldsymbol{\lambda}) = \mathcal{W}(\mathbf{d}) + \sum_{j=1}^{\text{nco}} \frac{1}{2\epsilon} \langle \lambda_j + \epsilon G_j(\mathbf{d}) \rangle^2 - \frac{1}{2\epsilon} \lambda_j^2. \quad (2.120)$$

While an implementation of the Augmented Lagrange method based on Uzawa's algorithm is again very simple and thus not considered in detail here, an interesting connection comes to light for the case of a Lagrange multiplier treatment. Total variation of (2.120) with respect to both

displacements and Lagrange multipliers leads to

$$\delta \mathbf{d}^T \left(\frac{\partial \mathcal{W}(\mathbf{d})}{\partial \mathbf{d}} + \sum_{j=1}^{\text{nco}} \langle \lambda_j + \epsilon G_j(\mathbf{d}) \rangle \frac{\partial G_j(\mathbf{d})}{\partial \mathbf{d}} \right) \stackrel{!}{=} 0, \quad (2.121)$$

$$\delta \lambda_j (\lambda_j - \langle \lambda_j + \epsilon G_j(\mathbf{d}) \rangle) = 0, \quad j = 1, \dots, \text{nco}. \quad (2.122)$$

Taking into account that variations are arbitrary, comparing (2.122) with the nonlinear complementarity function defined in (2.116) and bearing in mind that $\langle x \rangle = \max(0, x)$, it becomes obvious that the Augmented Lagrange formulation and the Lagrange multiplier formulation based on NCP functions and the PDASS are virtually *equivalent*, see Alart and Curnier [3]. The only slight difference that remains between the two problem formulations is that the term $\langle \lambda_j + \epsilon G_j(\mathbf{d}) \rangle$ also shows up in the variation with respect to the displacements in (2.121), while only the Lagrange multiplier λ_j does instead in (2.110).

Though providing a consistent method in terms of giving the exact solution to (2.109), both approaches incorporate a certain regularization of the constraints. Concretely, this means that as long as the solution $(\mathbf{d}, \boldsymbol{\lambda})$ has not yet fully converged, a slight violation of the KKT conditions is allowed. It is important to point out that ϵ again does not play the classical role of a penalty parameter in this context, but rather that of an algorithmic regularization parameter. The Augmented Lagrange method for inequality constraints reproduces the exact Lagrange multiplier solution, even for finite values of ϵ . In the equivalent Lagrange multiplier formulation with the NCP function, the corresponding parameter c is therefore termed *complementarity parameter* from the outset. Detailed investigations on the algorithmic realization and thus on the equivalence between Augmented Lagrange formulation and the PDASS or semi-smooth Newton methods will be presented in Section 4.4.

2.5 Discretization techniques based on domain decomposition

Both mesh tying and contact problems typically consist of several subdomains or elastic bodies, respectively. All of these need to be discretized in space using finite element methods, but additionally also coupling terms for the information transfer between subdomains must be established. This is where domain decomposition methods come into play. The focus throughout this thesis will exclusively be on mortar finite element methods, which represent one of the most prominent domain decomposition approaches. The following paragraphs introduce the general concept of mortar finite element methods as discretization technique and give an outlook on the role of mortar methods in computational contact mechanics. All details concerning the implementation of mortar methods for mesh tying in nonlinear solid mechanics then follow in Chapter 3 and for finite deformation contact in Chapter 4.

2.5.1 General concepts of domain decomposition

First of all, it has to be pointed out that there exist several different meanings or interpretations of the term domain decomposition methods depending on the context of their application. In

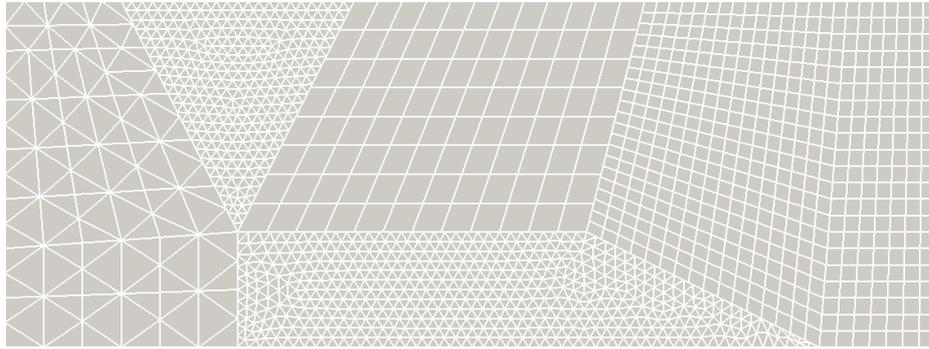


Figure 2.8: Exemplary application of domain decomposition methods for non-conforming finite element discretization, i.e. the coupling of non-matching meshes at subdomain interfaces.

principle, three different denotations can be distinguished. The first denotation, common in parallel computing, takes DD methods as a parallelization technique for distributing data among several processors. Secondly, DD methods are widely used within the scope of iterative solvers for large linear systems. Concretely, efficient preconditioning techniques for Krylov subspace methods such as CG or GMRES (see Section 2.3.6) can be designed based on a domain decomposition approach. For details on this DD application, the interested reader is exemplarily referred to Smith et al. [152] and Toselli and Widlund [162].

The third denotation of domain decomposition methods, which is solely considered in the following, is their application as optimal discretization techniques for interface conditions. This includes a multitude of coupling scenarios, ranging from the decomposition of a heterogeneous problem into homogeneous subproblems to the coupling of different physical models, discretization schemes and non-matching meshes. Such problems are usually characterized by continuity conditions or jump conditions for the primary variables (i.e. the displacements in solid mechanics) at the interface, which are reformulated in a weak, variationally consistent manner. Figure 2.8 exemplarily illustrates the coupling of several subdomains with dissimilar meshes as a typical application of DD methods.

From the mathematical point of view, one of the main interests is to assure optimality in the sense that the consistency error introduced by the DD method is small as compared with the discretization errors on the individual subdomains. In other words, an optimal information transfer and communication between the subdomains is sought. Mortar finite element methods (cf. the upcoming paragraph) can be shown to provide this optimality for a wide range of applications and will be employed in this thesis to establish interface coupling conditions for mesh tying and unilateral contact in nonlinear solid mechanics. For the sake of completeness, it should be mentioned that there also exists a common classification of DD methods into overlapping and non-overlapping methods, with mortar methods belonging to the latter group.

2.5.2 Mortar finite element methods

Originally introduced as a DD technique for spectral elements (see Bernardi et al. [16]), mortar methods are nowadays also widely used within finite element formulations for many different

problem classes. The first investigations on mortar finite element methods were typically performed for model problems of Laplace operator type, e.g. the Poisson equation, and formulated as non-conforming variational problem with the coupling constraints directly introduced into the global solution space. Yet, only shortly afterwards an alternative formulation became popular, which is not based on a constrained solution space, but rather introduces Lagrange multipliers in the sense of constrained minimization, thus leading to a typical saddle point formulation. Details of both approaches can be found in Ben Belgacem [11] and Wohlmuth [175]. However, only the saddle point formulation with Lagrange multipliers is considered here.

An in-depth presentation of the mathematical problem statements and mortar finite element discretization for mesh tying and unilateral contact will follow in Sections 3.1 and 4.1, respectively. At this point, only a very brief overview of the main theoretical results with regard to mortar methods is given, and the interested reader is referred to the comprehensive introduction in Wohlmuth [176] for the corresponding proofs and derivations. Most importantly, suitable inf-sup conditions and optimal a priori error estimates for the consistency error and the best approximation error have been established for the most widely used finite element discretizations and for different choices of the discrete Lagrange multiplier space, at least in the context of mortar mesh tying. Considering first-order finite elements as an example, this concretely means that their theoretically optimal spatial convergence rate $\mathcal{O}(h^2)$ measured in the L^2 -norm is *preserved* by mortar methods, despite the fact that now one or more non-conforming interfaces may be involved. Establishing optimal a priori error bounds for unilateral contact problems is more intricate due to the typically reduced regularity of the solution. The reader is referred to Section 4.3 as well as Wohlmuth et al. [179] and references therein.

Especially the choice of the discrete Lagrange multiplier space is an essential question with great implications on the actual algorithmic realization of mortar methods, see Section 3.3 and Section 4.3. Throughout this thesis, two possible families, namely *standard* and *dual* Lagrange multipliers, will be considered, with a strong focus on the latter. Furthermore, mortar coupling for second-order finite elements will also be discussed in the upcoming chapters.

2.5.3 Application to mesh tying in solid mechanics

Mortar finite element methods applied to mesh tying in nonlinear solid mechanics will be the model problem considered in Chapter 3 of this thesis, and thus serve as a basis for all further investigations and developments in the context of finite deformation contact. It will be seen that many aspects related to mortar finite element discretization can already be completely established within this simpler framework before being transferred and extended for the unilateral contact case. The associated problem formulation is based on the IBVP of nonlinear solid mechanics in combination with the tied contact constraints introduced in Section 2.2.2, and thus only demands the enforcement of equality constraints. Some related approaches for mesh tying in both linear and nonlinear solid mechanics can be found in Dohrmann et al. [34], Hesch and Betsch [72], Krause and Wohlmuth [99] and Puso [129].

2.5.4 Application to contact mechanics

The actual focus of this thesis is on mortar finite element methods for computational contact mechanics, and the corresponding methods will be presented in Chapter 4. As mentioned before,

some parts concerning mortar methods and finite element discretization can be directly adopted from the mesh tying case, while others require completely new approaches. Especially with respect to the contact typical inequality conditions at the interface, contact mechanics can be interpreted as a non-standard application of mortar methods, which were initially earmarked for equality constraints. In contrast to classical domain decomposition applications, mortar methods for finite deformation contact need to incorporate a consistent linearization of mortar coupling, a distinct treatment of normal and tangential directions at the interface and suitable active set strategies for inequality constraints. Again, several related approaches can be found in the literature, e.g. the contributions in Dickopf and Krause [32], Fischer and Wriggers [39], Hesch and Betsch [71], Hübner and Wohlmuth [80], Hübner et al. [84], McDevitt and Laursen [116], Puso and Laursen [132, 133], Tur et al. [163] and Yang et al. [189].

3 Mortar Methods for Finite Deformation Solid Mechanics

Mesh tying (also referred to as tied contact) serves as a model problem for the introduction to mortar finite element methods here. The basic motivation for such mortar mesh tying algorithms is to connect dissimilar meshes in nonlinear solid mechanics in a variationally consistent manner. Reasons for the occurrence of non-matching meshes can be manifold and range from different resolution requirements in the individual subdomains over the use of different types of finite element interpolation to the rather practical experience that the submodels to be connected are commonly meshed independently.

As for numerical contact algorithms, traditionally, nodal collocation schemes and so-called multi-point constraints have been employed to approximate the coupling of dissimilar meshes, see e.g. Hallquist et al. [64]. In the meanwhile, mortar methods and related approaches are quite well-established, and the reader is also referred to Dohrmann et al. [34], Park et al. [122], Puso [129] and the references therein for a comprehensive overview.

The following derivations start from the mesh tying problem statement and its weak formulation, before focusing on the evaluation and numerical integration of the typical discrete mortar finite element contributions. Moreover, a detailed overview of discrete Lagrange multiplier spaces to be used in conjunction with the mortar approach is given, and particularly new results for second-order interpolation in 3D are presented. Thereafter, the resulting solution algorithms with their respective algebraic representations are outlined, where the main focus of attention is on the dual Lagrange multiplier formulation. Finally, selected numerical examples are analyzed to validate the proposed methods in terms of numerical consistency, spatial convergence, efficiency and mechanical conservation properties among others.

3.1 Mesh tying problem statement

The problem statement for mesh tying presented in this paragraph is based on the IBVP of nonlinear solid mechanics (cf. Section 2.1.4) and the tied contact constraints (cf. Section 2.2.2). After recapitulating some basic notation and the strong formulation, a weak formulation of the mesh tying problem with two subdomains will be introduced. Here, only the interpretation as constrained minimization problem is considered, leading to an indefinite saddle point formulation based on Lagrange multipliers. Alternatively, the domain decomposition concept could also be captured by an equivalent non-conforming variational problem with a constrained solution space. While such an approach would yield a positive definite formulation, it is not possible to construct a solution basis with local support in that case. For further details and explanations concerning the two possible ways of formulating domain decomposition methods in the even simpler context of a Laplace operator model problem, the interested reader is referred to Wohlmuth [176].

3.1.1 Strong formulation

Without loss of generality, only the case of a body with one sole tied contact interface is considered. A generalization to the case of multiple interfaces is however possible without conceptual differences. Some remarks on the mostly technical, special treatment of so-called crosspoints, where more than two subdomains meet, will be given in Section 3.3.3. Figure 2.4 has already given an overview of the general problem setup and has introduced some basic notation. As usual, $\Omega_0^{(i)} \subset \mathbb{R}^3$ and $\Omega_t^{(i)} \subset \mathbb{R}^3$, $i = 1, 2$, represent the two subdomains of the contemplated body in the reference and current configuration, respectively. The subdomain boundaries $\partial\Omega_0^{(i)}$ are divided into three disjoint subsets $\Gamma_u^{(i)}$, $\Gamma_\sigma^{(i)}$ and $\Gamma_c^{(i)}$, where $\Gamma_u^{(i)}$ is the Dirichlet boundary, $\Gamma_\sigma^{(i)}$ is the Neumann boundary, and $\Gamma_c^{(i)}$ represents the mesh tying interface. Any gaps and overlaps between the subdomains are excluded, i.e. $\Gamma_c^{(1)} \equiv \Gamma_c^{(2)} \equiv \Gamma_c$. The counterparts of the described boundary sets in the current configuration are denoted as $\gamma_u^{(i)}$, $\gamma_\sigma^{(i)}$ and $\gamma_c^{(i)}$. As explained in Section 2.2.1, the superscript $\cdot^{(1)}$ is commonly referred to as slave side of the problem, whereas the superscript $\cdot^{(2)}$ denotes the master side.

On each subdomain $\Omega_0^{(i)}$, the initial boundary value problem of finite deformation elastodynamics needs to be satisfied, viz.

$$\text{Div} \mathbf{P}^{(i)} + \hat{\mathbf{b}}_0^{(i)} = \rho_0^{(i)} \ddot{\mathbf{u}}^{(i)} \quad \text{in } \Omega_0^{(i)} \times [0, T], \quad (3.1)$$

$$\mathbf{u}^{(i)} = \hat{\mathbf{u}}^{(i)} \quad \text{on } \Gamma_u^{(i)} \times [0, T], \quad (3.2)$$

$$\mathbf{P}^{(i)} \cdot \mathbf{N}^{(i)} = \hat{\mathbf{t}}_0^{(i)} \quad \text{on } \Gamma_\sigma^{(i)} \times [0, T], \quad (3.3)$$

$$\mathbf{u}^{(i)}(\mathbf{X}^{(i)}, 0) = \hat{\mathbf{u}}_0^{(i)}(\mathbf{X}^{(i)}) \quad \text{in } \Omega_0^{(i)}, \quad (3.4)$$

$$\dot{\mathbf{u}}^{(i)}(\mathbf{X}^{(i)}, 0) = \hat{\dot{\mathbf{u}}}_0^{(i)}(\mathbf{X}^{(i)}) \quad \text{in } \Omega_0^{(i)}. \quad (3.5)$$

The tied contact constraint, also formulated in the reference configuration, is given as

$$\mathbf{u}^{(1)} = \mathbf{u}^{(2)} \quad \text{on } \Gamma_c \times [0, T]. \quad (3.6)$$

Equations (3.1)–(3.6) represent the final strong form of a mesh tying problem in nonlinear solid mechanics. In the course of deriving a weak formulation (see next paragraph), the balance of linear momentum at the mesh tying interface Γ_c is typically exploited and a Lagrange multiplier vector field $\boldsymbol{\lambda}$ is introduced, thus setting the basis for a mixed variational approach.

3.1.2 Weak formulation

To start the derivation of a weak formulation of (3.1)–(3.6), appropriate solution spaces $\mathcal{U}^{(i)}$ and weighting spaces $\mathcal{V}^{(i)}$ need to be defined as

$$\mathcal{U}^{(i)} = \{ \mathbf{u}^{(i)} \in H^1(\Omega) \mid \mathbf{u}^{(i)} = \hat{\mathbf{u}}^{(i)} \text{ on } \Gamma_u \}, \quad (3.7)$$

$$\mathcal{V}^{(i)} = \{ \delta \mathbf{u}^{(i)} \in H^1(\Omega) \mid \delta \mathbf{u}^{(i)} = \mathbf{0} \text{ on } \Gamma_u \}. \quad (3.8)$$

Moreover, the Lagrange multiplier vector $\boldsymbol{\lambda} = -\mathbf{t}_c^{(1)}$, which represents the *negative* slave side contact traction $\mathbf{t}_c^{(1)}$ and is supposed to enforce the mesh tying constraint (3.6), is chosen from a corresponding solution space denoted as \mathcal{M} . In terms of its classification in functional analysis,

this space represents the dual space of the trace space $\mathcal{W}^{(1)}$ of $\mathcal{V}^{(1)}$. In the given context, this means that $\mathcal{M} = H^{-1/2}(\Gamma_c)$ and $\mathcal{W}^{(1)} = H^{1/2}(\Gamma_c)$, where \mathcal{M} and $\mathcal{W}^{(1)}$ denote single scalar components of the corresponding vector-valued spaces \mathcal{M} and \mathcal{W} .

Based on these considerations, a saddle point type weak formulation is derived next. Basically, this can be done by extending the standard weak formulation of nonlinear solid mechanics as defined in (2.61) to two subdomains and combining it with the Lagrange multiplier coupling terms introduced in generic form in (2.97). Find $\mathbf{u}^{(i)} \in \mathcal{U}^{(i)}$ and $\boldsymbol{\lambda} \in \mathcal{M}$ such that

$$-\delta\mathcal{W}_{\text{kin}}(\mathbf{u}^{(i)}, \delta\mathbf{u}^{(i)}) - \delta\mathcal{W}_{\text{int,ext}}(\mathbf{u}^{(i)}, \delta\mathbf{u}^{(i)}) - \delta\mathcal{W}_{\text{mt}}(\boldsymbol{\lambda}, \delta\mathbf{u}^{(i)}) = 0 \quad \forall \delta\mathbf{u}^{(i)} \in \mathcal{V}^{(i)}, \quad (3.9)$$

$$\delta\mathcal{W}_\lambda(\mathbf{u}^{(i)}, \delta\boldsymbol{\lambda}) = 0 \quad \forall \delta\boldsymbol{\lambda} \in \mathcal{M}. \quad (3.10)$$

Herein, the kinetic contribution $\delta\mathcal{W}_{\text{kin}}$, the internal and external contributions $\delta\mathcal{W}_{\text{int,ext}}$ and the mesh tying interface contribution $\delta\mathcal{W}_{\text{mt}}$ to the overall virtual work on the two subdomains, as well as the weak form of the mesh tying constraint $\delta\mathcal{W}_\lambda$, have been abbreviated as

$$-\delta\mathcal{W}_{\text{kin}} = \sum_{i=1}^2 \left[\int_{\Omega_0^{(i)}} \rho_0^{(i)} \ddot{\mathbf{u}}^{(i)} \cdot \delta\mathbf{u}^{(i)} \, dV_0 \right], \quad (3.11)$$

$$-\delta\mathcal{W}_{\text{int,ext}} = \sum_{i=1}^2 \left[\int_{\Omega_0^{(i)}} \left(\mathbf{S}^{(i)} : \delta\mathbf{E}^{(i)} - \hat{\mathbf{b}}_0^{(i)} \cdot \delta\mathbf{u}^{(i)} \right) \, dV_0 - \int_{\Gamma_\sigma^{(i)}} \hat{\mathbf{t}}_0^{(i)} \cdot \delta\mathbf{u}^{(i)} \, dA_0 \right], \quad (3.12)$$

$$-\delta\mathcal{W}_{\text{mt}} = \int_{\Gamma_c} \boldsymbol{\lambda} \cdot (\delta\mathbf{u}^{(1)} - \delta\mathbf{u}^{(2)}) \, dA_0, \quad (3.13)$$

$$\delta\mathcal{W}_\lambda = \int_{\Gamma_c} \delta\boldsymbol{\lambda} \cdot (\mathbf{u}^{(1)} - \mathbf{u}^{(2)}) \, dA_0. \quad (3.14)$$

It is important to point out that, strictly speaking, the coupling bilinear forms $\delta\mathcal{W}_{\text{mt}}$ and $\delta\mathcal{W}_\lambda$ cannot be represented by integrals, because the involved spaces $H^{1/2}(\Gamma_c)$ and $H^{-1/2}(\Gamma_c)$ do not satisfy the requirements for a proper integral definition. Instead, a mathematically correct notation would use so-called duality pairings $\langle \boldsymbol{\lambda}, (\delta\mathbf{u}^{(1)} - \delta\mathbf{u}^{(2)}) \rangle_{\Gamma_c}$ and $\langle \delta\boldsymbol{\lambda}, (\mathbf{u}^{(1)} - \mathbf{u}^{(2)}) \rangle_{\Gamma_c}$, see e.g. Wohlmuth [175]. However, during finite element discretization the solution spaces are restricted to discrete subsets of $L^2(\Gamma_c)$ functions, and by then at the latest the coupling terms may be formulated as surface integrals. Moreover, even in the mathematical literature the distinction between duality pairing and integral is not treated consistently, and thus the slightly inaccurate formulation in (3.13) and (3.14) is preferred here due to readability.

The coupling terms on Γ_c also allow for a direct interpretation in terms of variational formulations and the principle of virtual work. Whereas the contribution in (3.13) represents the virtual work of the unknown interface tractions $\boldsymbol{\lambda} = -\mathbf{t}_c^{(1)} = \mathbf{t}_c^{(2)}$, the contribution in (3.14) ensures a weak, variationally consistent enforcement of the tied contact constraint (3.6). Unlike for unilateral contact with inequality constraints, there exist no further restrictions on the Lagrange multiplier space \mathcal{M} here (such as e.g. positivity). Nevertheless, the concrete choice of the discrete Lagrange multiplier space \mathcal{M}_h in the context of mortar finite element discretizations is decisive for the stability of the method and for optimal a priori error bounds, cf. Section 3.3. Finally, it is pointed out that the weak formulation (3.9) and (3.10) possesses all characteristics of saddle point problems and Lagrange multiplier methods, as indicated in Section 2.4.1.

3.2 Discretization and numerical integration

In the following, the fundamental steps towards a mortar finite element discretization of the mesh tying problem will be outlined. Starting from an overview of the typically employed types of finite elements, the main focus of interest is on the evaluation of the resulting mortar coupling terms and Lagrange multiplier contributions. The corresponding numerical integration algorithms in both 2D and 3D will be derived in detail with step-by-step explanations. With the employed quadrature scheme being of great importance for the overall accuracy of the numerical simulation, a short overview of possible integration schemes is given and the most accurate version, based on a thorough segmentation of the mesh tying interface, is presented in detail. Finally, some remarks are given on implicit time integration for tied contact problems as well as on how to satisfy typical mechanical conservation laws.

3.2.1 Finite element discretization

For the spatial discretization of the tied contact problem (3.9) and (3.10), standard isoparametric finite elements are employed. This defines the usual finite dimensional subspaces $\mathcal{U}_h^{(i)}$ and $\mathcal{V}_h^{(i)}$ being approximations of $\mathcal{U}^{(i)}$ and $\mathcal{V}^{(i)}$, respectively. Throughout this thesis, both first-order and second-order interpolation is considered with finite element meshes typically consisting of 3-node triangular (*tri3*), 4-node quadrilateral (*quad4*), 6-node triangular (*tri6*), 8-node quadrilateral (*quad8*) and 9-node quadrilateral (*quad9*) elements in 2D, and of 4-node tetrahedral (*tet4*), 8-node hexahedral (*hex8*), 10-node tetrahedral (*tet10*), 20-node hexahedral (*hex20*) and 27-node hexahedral (*hex27*) elements in 3D.

With the focus being on the finite element discretization of the coupling terms here, only the geometry, displacement and Lagrange multiplier interpolations on $\Gamma_{c,h}^{(i)}$ will be considered in the following. Discretization of the remaining contributions to (3.9) is not discussed, but the reader is instead referred to the abundant literature. As explained in Section 2.3.2, the subscript \cdot_h refers to a spatially discretized quantity. Obviously, there exists a connection between the employed finite elements in the domains $\Omega_{0,h}^{(i)}$ and the resulting surface facets on the mesh tying interfaces $\Gamma_{c,h}^{(i)}$. For example, a mixed 3D finite element mesh composed of *tet4* and *hex8* elements yields *tri3* and *quad4* facets on the surface of tied contact. Consequently, the following general form of geometry and displacement interpolation on the discrete mesh tying surfaces holds:

$$\mathbf{x}_h^{(1)}|_{\Gamma_{c,h}^{(1)}} = \sum_{k=1}^{n^{(1)}} N_k^{(1)}(\xi^{(1)}, \eta^{(1)}) \mathbf{x}_k^{(1)}, \quad \mathbf{x}_h^{(2)}|_{\Gamma_{c,h}^{(2)}} = \sum_{l=1}^{n^{(2)}} N_l^{(2)}(\xi^{(2)}, \eta^{(2)}) \mathbf{x}_l^{(2)}, \quad (3.15)$$

$$\mathbf{u}_h^{(1)}|_{\Gamma_{c,h}^{(1)}} = \sum_{k=1}^{n^{(1)}} N_k^{(1)}(\xi^{(1)}, \eta^{(1)}) \mathbf{d}_k^{(1)}, \quad \mathbf{u}_h^{(2)}|_{\Gamma_{c,h}^{(2)}} = \sum_{l=1}^{n^{(2)}} N_l^{(2)}(\xi^{(2)}, \eta^{(2)}) \mathbf{d}_l^{(2)}. \quad (3.16)$$

The total number of slave nodes on $\Gamma_{c,h}^{(1)}$ is $n^{(1)}$, and the total number of master nodes on $\Gamma_{c,h}^{(2)}$ is $n^{(2)}$. Discrete nodal positions and discrete nodal displacements are given by $\mathbf{x}_k^{(1)}$, $\mathbf{x}_l^{(2)}$, $\mathbf{d}_k^{(1)}$ and $\mathbf{d}_l^{(2)}$. The shape functions $N_k^{(1)}$ and $N_l^{(2)}$ are defined with respect to the usual finite element parameter space, commonly denoted as $\xi^{(i)}$ for two-dimensional problems (i.e. 1D mesh tying interfaces) and as $\boldsymbol{\xi}^{(i)} = (\xi^{(i)}, \eta^{(i)})$ for three-dimensional problems (i.e. 2D mesh tying interfaces).

As mentioned above, the shape functions are derived from the underlying bulk discretization. Although not studied here, the proposed algorithms can in principle be transferred to higher-order interpolation and alternative shape functions, such as non-uniform rational B-splines (NURBS), see e.g. Cottrell et al. [26], De Lorenzis et al. [29] and Temizer et al. [158, 159].

In addition, an adequate discretization of the Lagrange multiplier vector $\boldsymbol{\lambda}$ is needed, too, and will be based on a discrete Lagrange multiplier space $\boldsymbol{\mathcal{M}}_h$ being an approximation of $\boldsymbol{\mathcal{M}}$. All details concerning the choice of $\boldsymbol{\mathcal{M}}_h$, and especially concerning the two possible families of standard and dual Lagrange multipliers, will follow in Section 3.3. Thus, only a very general notation is given at this point:

$$\boldsymbol{\lambda}_h = \sum_{j=1}^{m^{(1)}} \Phi_j(\xi^{(1)}, \eta^{(1)}) \boldsymbol{\lambda}_j, \quad (3.17)$$

with the (still to be defined) shape functions Φ_j and the discrete nodal Lagrange multipliers $\boldsymbol{\lambda}_j$. The total number of slave nodes carrying additional Lagrange multiplier degrees of freedom is $m^{(1)}$. Typically for mortar methods, every slave node also serves as coupling node, and thus in the majority of cases $m^{(1)} = n^{(1)}$ will hold. However, in the context of second-order finite element interpolation, it will be favorable to chose $m^{(1)} < n^{(1)}$ in certain cases, see Section 3.3. Substituting (3.15) and (3.17) into the interface virtual work $\delta\mathcal{W}_{\text{mt}}$ in (3.9) yields

$$\begin{aligned} -\delta\mathcal{W}_{\text{mt},h} &= \sum_{j=1}^{m^{(1)}} \sum_{k=1}^{n^{(1)}} \boldsymbol{\lambda}_j^\top \left(\int_{\Gamma_{c,h}^{(1)}} \Phi_j N_k^{(1)} dA_0 \right) \delta\mathbf{d}_k^{(1)} \\ &\quad - \sum_{j=1}^{m^{(1)}} \sum_{l=1}^{n^{(2)}} \boldsymbol{\lambda}_j^\top \left(\int_{\Gamma_{c,h}^{(1)}} \Phi_j (N_l^{(2)} \circ \chi_h) dA_0 \right) \delta\mathbf{d}_l^{(2)}, \end{aligned} \quad (3.18)$$

where $\chi_h : \Gamma_{c,h}^{(1)} \rightarrow \Gamma_{c,h}^{(2)}$ defines a suitable discrete mapping from the slave to the master side of the mesh tying interface. Such a mapping (or projection) becomes necessary due to the fact that the discretized coupling surfaces $\Gamma_{c,h}^{(1)}$ and $\Gamma_{c,h}^{(2)}$ are, in general, no longer geometrically coincident. This becomes very clear when thinking of a curved mesh tying interface with non-matching finite element meshes on the two different sides. As illustrated in Figure 3.1, tiny gaps and overlaps may be generated in the discretized setting, although the surfaces had still been coincident in the continuum framework. Throughout this thesis, numerical integration of the mortar coupling terms is exclusively performed on the slave side $\Gamma_{c,h}^{(1)}$ of the interface. In (3.18), nodal blocks of the two mortar integral matrices commonly denoted as \mathbf{D} and \mathbf{M} can be identified. This leads to the following definitions:

$$\mathbf{D}[j, k] = D_{jk} \mathbf{1}_{\text{ndim}} = \int_{\Gamma_{c,h}^{(1)}} \Phi_j N_k^{(1)} dA_0 \mathbf{1}_{\text{ndim}}, \quad j = 1, \dots, m^{(1)}, \quad k = 1, \dots, n^{(1)}, \quad (3.19)$$

$$\mathbf{M}[j, l] = M_{jl} \mathbf{1}_{\text{ndim}} = \int_{\Gamma_{c,h}^{(1)}} \Phi_j (N_l^{(2)} \circ \chi_h) dA_0 \mathbf{1}_{\text{ndim}}, \quad j = 1, \dots, m^{(1)}, \quad l = 1, \dots, n^{(2)}. \quad (3.20)$$

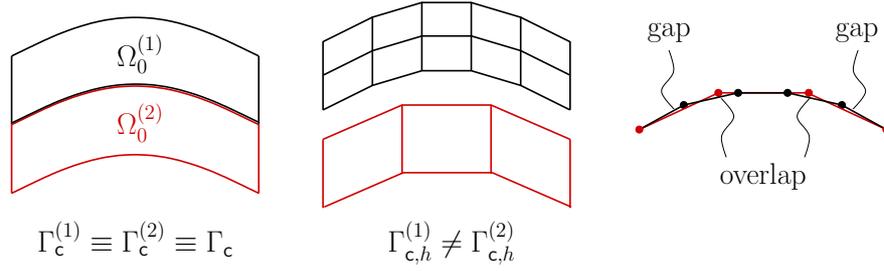


Figure 3.1: Gaps and overlaps in a curved mesh tying interface with non-matching FE meshes.

Note that $\mathbf{I}_{\text{ndim}} \in \mathbb{R}^{\text{ndim} \times \text{ndim}}$ is an identity matrix whose size is determined by the global problem dimension ndim , viz. either $\text{ndim} = 2$ or $\text{ndim} = 3$. In general, both mortar matrices \mathbf{D} and \mathbf{M} have a rectangular shape. However, \mathbf{D} becomes a square matrix for the common choice $m^{(1)} = n^{(1)}$. All details concerning the actual numerical integration of the mass matrix type of entries in \mathbf{D} and \mathbf{M} as well as the implementation of the interface mapping χ_h for both 2D and 3D will be given in the following Sections 3.2.2 and 3.2.3.

For the ease of notation, all nodes of the two subdomains $\Omega_0^{(1)}$ and $\Omega_0^{(2)}$, and correspondingly all degrees of freedom (DOFs) in the global discrete displacement vector \mathbf{d} , are sorted into three groups: a group \mathcal{S} containing all slave interface quantities, a group \mathcal{M} of all master quantities and a group denoted as \mathcal{N} , which comprises all remaining nodes or DOFs. The global discrete displacement vector can be sorted accordingly, yielding $\mathbf{d} = (\mathbf{d}_{\mathcal{N}}, \mathbf{d}_{\mathcal{M}}, \mathbf{d}_{\mathcal{S}})$. Going back to (3.18), this allows for the following definition:

$$-\delta \mathcal{W}_{\text{mt},h} = \delta \mathbf{d}_{\mathcal{S}}^{\text{T}} \mathbf{D}^{\text{T}} \boldsymbol{\lambda} - \delta \mathbf{d}_{\mathcal{M}}^{\text{T}} \mathbf{M}^{\text{T}} \boldsymbol{\lambda} = \delta \mathbf{d}^{\text{T}} \underbrace{\begin{bmatrix} \mathbf{0} \\ -\mathbf{M}^{\text{T}} \\ \mathbf{D}^{\text{T}} \end{bmatrix}}_{\mathbf{B}_{\text{mt}}^{\text{T}}} \boldsymbol{\lambda} = \delta \mathbf{d}^{\text{T}} \mathbf{f}_{\text{mt}}(\boldsymbol{\lambda}). \quad (3.21)$$

Herein, the discrete mortar mesh tying operator \mathbf{B}_{mt} and the resulting discrete vector of mesh tying forces $\mathbf{f}_{\text{mt}}(\boldsymbol{\lambda}) = \mathbf{B}_{\text{mt}}^{\text{T}} \boldsymbol{\lambda}$ acting on the slave and the master side of the interface are introduced. To finalize the discretization of the considered mesh tying problem, a closer look needs to be taken at the weak constraint contribution $\delta \mathcal{W}_{\lambda}$ in (3.10). Due to the saddle point characteristics and resulting symmetry of the mixed variational formulation in (3.9) and (3.10), all discrete components of $\delta \mathcal{W}_{\lambda}$ have already been introduced and the final formulation is given as

$$\delta \mathcal{W}_{\lambda,h} = \delta \boldsymbol{\lambda}^{\text{T}} \mathbf{D} \mathbf{d}_{\mathcal{S}} - \delta \boldsymbol{\lambda}^{\text{T}} \mathbf{M} \mathbf{d}_{\mathcal{M}} = \delta \boldsymbol{\lambda}^{\text{T}} \mathbf{B}_{\text{mt}} \mathbf{d} = \delta \boldsymbol{\lambda}^{\text{T}} \mathbf{g}_{\text{mt}}(\mathbf{d}), \quad (3.22)$$

with $\mathbf{g}_{\text{mt}}(\mathbf{d}) = \mathbf{B}_{\text{mt}} \mathbf{d}$ representing the discrete mesh tying constraint at the coupling interface. Taking into account the typical finite element discretization of all remaining contributions to the first part of the weak formulation (3.9), as previously outlined in Section 2.3.2, the semi-discrete equations of motion including tied contact forces and the constraint equations emerge as

$$\mathbf{M} \ddot{\mathbf{d}} + \mathbf{C} \dot{\mathbf{d}} + \mathbf{f}_{\text{int}}(\mathbf{d}) + \mathbf{f}_{\text{mt}}(\boldsymbol{\lambda}) - \mathbf{f}_{\text{ext}} = \mathbf{0}, \quad (3.23)$$

$$\mathbf{g}_{\text{mt}}(\mathbf{d}) = \mathbf{0}. \quad (3.24)$$

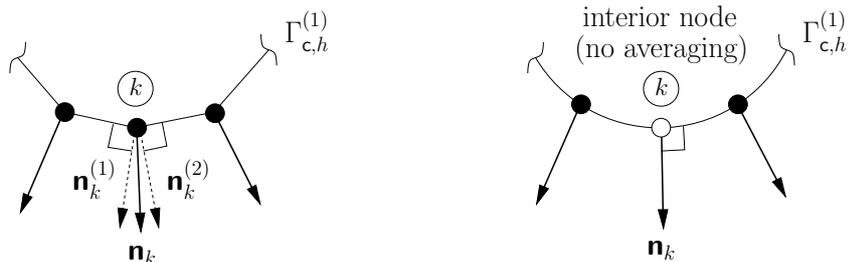


Figure 3.2: Continuous field of normals in 2D for first-order and second-order interpolation.

Mass matrix \mathbf{M} , damping matrix \mathbf{C} , internal forces $\mathbf{f}_{\text{int}}(\mathbf{d})$ and external forces \mathbf{f}_{ext} result from standard FE discretization. It is important to point out that the actual mortar-based interface coupling described here is completely independent of the concrete choice of the underlying finite element formulation. The same also holds true for the question which particular material model is applied. As both topics, i.e. nonlinear finite elements for continua and complex material models, are discussed at length in the literature specified in Section 2.3, details will not be repeated here but the focus will remain solely on the mesh tying terms $\mathbf{f}_{\text{mt}}(\boldsymbol{\lambda})$ and $\mathbf{g}_{\text{mt}}(\mathbf{d})$.

Examining the semi-discrete problem statement in (3.23) and (3.24) in more detail, the well-known nonlinearity of the internal forces $\mathbf{f}_{\text{int}}(\mathbf{d})$ due to the consideration of finite deformation kinematics and nonlinear material behavior becomes apparent. However, neither the discrete interface forces $\mathbf{f}_{\text{mt}}(\boldsymbol{\lambda})$ nor the mesh tying constraints $\mathbf{g}_{\text{mt}}(\mathbf{d})$ introduce an additional nonlinearity into the system. This is due to the fact that no relative movement of the subdomains is permitted in mesh tying problems. Therefore, the mortar integral matrices \mathbf{D} and \mathbf{M} and hence also the discrete mesh tying operator \mathbf{B}_{mt} only need to be evaluated *once* at problem initialization and thus do not depend on the actual displacements, even if finite deformations of the considered body are involved. With respect to numerical efficiency, this means that evaluating the mortar coupling terms for tied contact problems is a one-time cost, which can usually be neglected as compared with the remaining computational costs. Only for the unilateral contact case discussed in Chapter 4, this will no longer be the case. The question how to numerically evaluate the entries of \mathbf{B}_{mt} in 2D and 3D problems is discussed in the following two paragraphs.

3.2.2 Evaluation of mortar integrals in 2D

There exists one fundamental difference in the definitions of the two mortar matrices \mathbf{D} and \mathbf{M} in (3.19) and (3.20), namely with respect to the involvement of the master side of the mesh tying interface $\Gamma_{c,h}^{(2)}$. The integrand of a matrix entry D_{jk} contains only quantities that are defined on the slave side of the mesh tying interface, i.e. where numerical integration will actually be performed. On the contrary, evaluating the entries M_{jl} is much more intricate due to the fact that a product of master side shape functions $N_l^{(2)}$ and slave side shape functions Φ_j needs to be integrated over the slave surface $\Gamma_{c,h}^{(1)}$. Firstly, this inevitably involves the discrete mapping χ_h to project finite element nodes and integration points between slave and master sides. Secondly, the integration domain $\Gamma_{c,h}^{(1)}$ has to be split into so-called mortar segments, on which both Φ_j and $N_l^{(2)}$ are C^1 -continuous, as kinks in the function to be integrated would deteriorate the achievable accuracy of Gauss quadrature.

Although the entries of \mathbf{D} could theoretically be evaluated by elementwise Gauss quadrature on $\Gamma_{c,h}^{(1)}$ (without the need for defining mortar segments), the same integration scheme as for the entries of \mathbf{M} , i.e. involving mortar segments, is typically employed anyhow. The reason for this procedure is that it guarantees linear momentum conservation in the semi-discrete framework, as will be seen in Section 3.2.5. Yet, setting up the mortar segments is still quite straightforward for the 2D mesh tying problems considered in this paragraph. As a first step, discrete projection rules between slave and master boundary need to be established. Here, projection will be based on a continuous field of normal vectors defined on the slave side $\Gamma_{c,h}^{(1)}$, see also Yang et al. [189] and Popp et al. [123]. At each slave node $k \in \mathcal{S}$, an averaged nodal unit normal is defined as

$$\mathbf{n}_k = \frac{\sum_{e=1}^{n_k^{\text{adj}}} \mathbf{n}_k^{(e)}}{\left\| \sum_{e=1}^{n_k^{\text{adj}}} \mathbf{n}_k^{(e)} \right\|}, \quad (3.25)$$

where $\mathbf{n}_k^{(e)}$ is the outward pointing unit normal vector of the adjacent slave element e evaluated at slave node k , and n_k^{adj} represents the total number of slave facets adjacent to slave node k . Note that the mesh tying surfaces consist of two-noded line (*line2*) segments for first-order finite element interpolation and of three-noded line (*line3*) segments for the second-order case. Therefore, one typically has $n_k^{\text{adj}} = 2$, except for interior nodes of *line3* segments and towards ends of the mesh tying interface, where obviously $n_k^{\text{adj}} = 1$ and thus no averaging becomes necessary. Two exemplary situations are illustrated in Figure 3.2. For simplicity, no weighting whatsoever of the element normals $\mathbf{n}_k^{(e)}$ is applied in this thesis, although such a weighting procedure has initially been suggested in Yang et al. [189]. Accordant extensions are straightforward and have been tested on several numerical examples as well, however the actual influence of weighting on the numerical results is evanescent.

In two dimensions, the nodal unit tangent vector $\boldsymbol{\tau}_k$ follows directly from $\boldsymbol{\tau}_k = \mathbf{e}_3 \times \mathbf{n}_k$, where \mathbf{e}_3 is the Cartesian unit basis vector in z -direction. Based on the averaged nodal normals in (3.25), a C^0 -continuous field of normals is readily defined through FE interpolation, i.e.

$$\mathbf{n}_c = \sum_{k=1}^{n^{(1)}} N_k^{(1)}(\boldsymbol{\xi}^{(1)}) \mathbf{n}_k. \quad (3.26)$$

With this definition at hand, contact segments are set up and the mortar matrices \mathbf{D} and \mathbf{M} can be numerically integrated. Building projection rules based on (3.26), instead of the well-known closest point projection (CPP) procedures, offers several appealing advantages. Most importantly, the C^0 -continuous field of normals avoids typical problems due to degenerate cases such as crossing of the normals or non-uniqueness of the CPP, which are otherwise inevitable in the context of low-order finite element discretization, see e.g. McDevitt and Laursen [116] and Papadopoulos and Taylor [121]. Moreover, the definition in (3.26) can also be interpreted as introducing a kind of numerical “smoothing” into the mesh tying formulation. In many cases, jumps of the normal vector at surface nodes can be considered unphysical and rather the consequence of coarse low-order finite element meshes on curved interfaces. In contrast to real surface smoothing procedures (see e.g. Padmanabhan and Laursen [120], Puso and Laursen [130]), the approach in (3.26) does *not* modify the actual surface geometry interpolation, but still produces a field of normals that would be typical of a C^1 -continuous surface description.

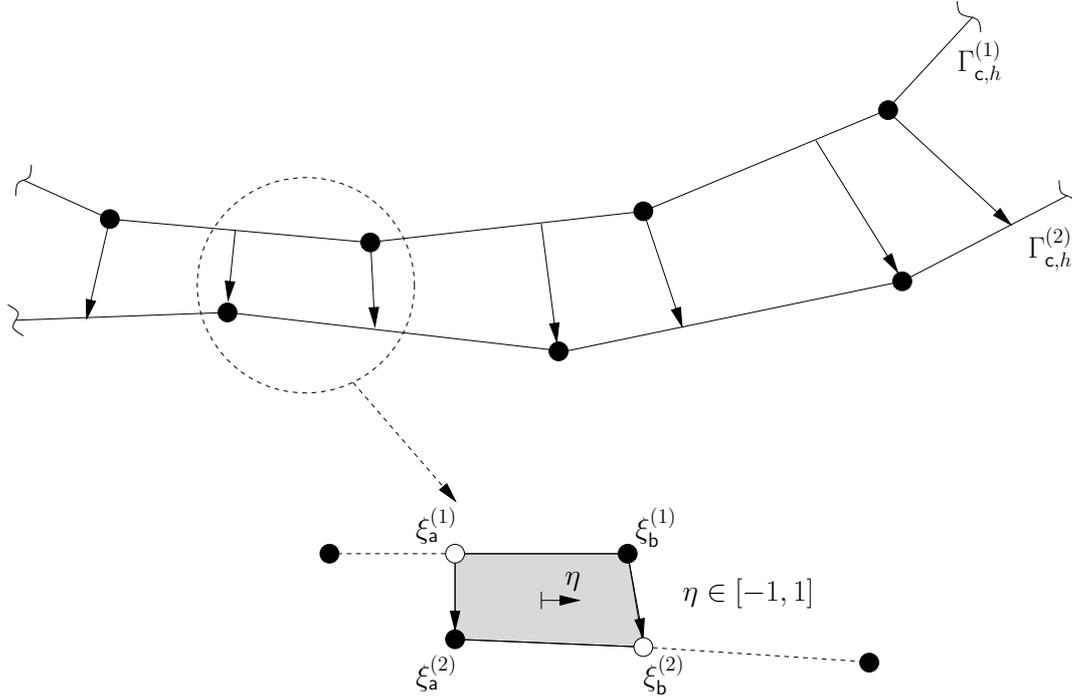


Figure 3.3: Mortar segmentation with continuous field of normals in 2D.

In the following, the 2D mortar projection and integration algorithm is only outlined for *one* pair of slave and master elements forming *one* mortar segment, as illustrated in Figure 3.3. Of course, this information about associated element pairs is to be gained from an efficient search strategy, cf. Section 4.6.2 and Yang and Laursen [186] for further details in the context of unilateral contact. The mortar segment end points are either a projection of slave nodes onto the master surface and vice versa or nodes themselves, see Figure 3.3. In the case of quadratic finite elements, it is sufficient to project the exterior nodes, because all shape functions provide C^1 -continuity at the interior nodes. The projection of a slave node $\mathbf{x}_k^{(1)}$ onto a master element e with nodes $\mathbf{x}_l^{(2)}$, $l = 1, \dots, n_m^e$, results from

$$\left[\left(\sum_{l=1}^{n_m^e} N_l^{(2)}(\xi^{(2)}) \mathbf{x}_l^{(2)} \right) - \mathbf{x}_k^{(1)} \right] \times \mathbf{n}_k = \mathbf{0}, \quad (3.27)$$

where $\xi^{(2)}$ is the sought-after projection coordinate of the master element. In a similar manner, the projection of a master node $\mathbf{x}_l^{(2)}$ onto a slave element e with nodes $\mathbf{x}_k^{(1)}$, $k = 1, \dots, n_s^e$ can be determined by solving

$$\left[\left(\sum_{k=1}^{n_s^e} N_k^{(1)}(\xi^{(1)}) \mathbf{x}_k^{(1)} \right) - \mathbf{x}_l^{(2)} \right] \times \left[\sum_{k=1}^{n_s^e} N_k^{(1)}(\xi^{(1)}) \mathbf{n}_k \right] = \mathbf{0}, \quad (3.28)$$

where now $\xi^{(1)}$ is the unknown projection coordinate of the slave element. In general, (3.27) and (3.28) are nonlinear conditions for the projection coordinates and thus are usually solved by a local Newton–Raphson iteration.

Gauss quadrature on the mortar segments requires the introduction of a segment parametrization defined as $\eta \in [-1, 1]$. The segment end points are given by local element coordinates on the slave side and master side of the mesh tying interface, respectively: $\xi_a^{(1)}$, $\xi_b^{(1)}$, $\xi_a^{(2)}$ and $\xi_b^{(2)}$. The mapping from slave element coordinates $\xi^{(1)}$ to segment coordinates η is

$$\xi^{(1)}(\eta) = \frac{1}{2}(1 - \eta)\xi_a^{(1)} + \frac{1}{2}(1 + \eta)\xi_b^{(1)}, \quad (3.29)$$

see again Figure 3.3. Note that $\xi_a^{(1)}$ and $\xi_b^{(1)}$ can either represent slave nodes or the projection of master nodes onto the slave side of the contact interface. For first-order interpolation, usually a similar definition for the mapping from master element coordinates $\xi^{(2)}$ to segment coordinates η is employed, see e.g. Hartmann [65] or Yang et al. [189]. Dealing with both first-order and second-order interpolation, a different procedure is proposed here. Instead of setting up an interpolation rule for $\xi^{(2)}(\eta)$, the individual Gauss points η_g (with global coordinates $\mathbf{x}_g^{(1)}$ and outward unit normal vector \mathbf{n}_g) are projected from slave surface to master surface, just as it has been done for the nodal points in (3.27), viz.

$$\left[\left(\sum_{l=1}^{n_m^e} N_l^{(2)}(\xi^{(2)}(\eta_g)) \mathbf{x}_l^{(2)} \right) - \mathbf{x}_g^{(1)} \right] \times \mathbf{n}_g = \mathbf{0}, \quad (3.30)$$

with $\xi^{(2)}(\eta_g)$ now representing the unknown projection of the Gauss point onto the master element. Thus, Gauss quadrature for the integral entries of the mortar matrices \mathbf{D} and \mathbf{M} on one single mortar segment can finally be expressed as

$$D_{jk(s,m)} = \sum_{g=1}^{n_{gp}} w_g \Phi_j(\xi^{(1)}(\eta_g)) N_k^{(1)}(\xi^{(1)}(\eta_g)) J_{\text{seg}}, \quad (3.31)$$

$$M_{jl(s,m)} = \sum_{g=1}^{n_{gp}} w_g \Phi_j(\xi^{(1)}(\eta_g)) N_k^{(2)}(\xi^{(2)}(\eta_g)) J_{\text{seg}}, \quad (3.32)$$

where J_{seg} comprises both the mapping from physical space to slave element parameter space and from slave element parameter space to segment coordinates:

$$J_{\text{seg}} = \left\| \frac{\partial \mathbf{x}_{e,h}^{(1)}}{\partial \xi^{(1)}} \right\| \cdot \left| \frac{\partial \xi^{(1)}}{\partial \eta} \right| = J(\xi^{(1)}(\eta_g)) \frac{\xi_b^{(1)} - \xi_a^{(1)}}{2}. \quad (3.33)$$

Expressions (3.31) and (3.32) represent the contribution to D_{jk} and M_{jl} formed by one specific slave and master element pair (s, m) . The total quantities are readily obtained by summing up all slave and master element pair contributions. It is pointed out that the given 2D mortar integration scheme also applies to quadratic finite elements without any conceptual differences. By default, five Gauss points are used to evaluate the integrals (3.31) and (3.32) in the presented implementation, thus allowing for an exact numerical integration of up to ninth-order polynomials.

3.2.3 Evaluation of mortar integrals in 3D

All general concepts of the evaluation of mortar integrals in 2D can be directly transferred to the 3D case. Again, the integral entries of both matrices \mathbf{D} and \mathbf{M} will be computed based on so-called mortar segments in order to achieve the maximum possible accuracy of Gauss quadrature

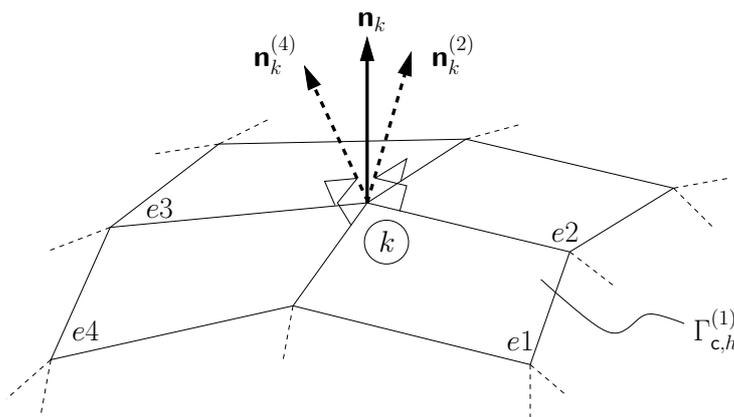


Figure 3.4: Nodally averaged normal vector \mathbf{n}_k at a slave node k with four adjacent slave facets $e1$ to $e4$. The element normal vectors $\mathbf{n}_k^{(e)}$ are exemplified for elements $e2$ and $e4$.

and to guarantee linear momentum conservation in the semi-discrete setting. Projection operations between slave surface $\Gamma_{c,h}^{(1)}$ and master surface $\Gamma_{c,h}^{(2)}$, which now consist of two-dimensional facets, are again based on nodal averaging and a C^0 -continuous field of normal vectors, cf. Figure 3.4. For 3D situations, the averaged nodal normal vector \mathbf{n}_k is again given as

$$\mathbf{n}_k = \frac{\sum_{e=1}^{n_k^{\text{adj}}} \mathbf{n}_k^{(e)}}{\left\| \sum_{e=1}^{n_k^{\text{adj}}} \mathbf{n}_k^{(e)} \right\|}, \quad (3.34)$$

where the total number of slave facets n_k^{adj} adjacent to slave node k may now vary within a much wider range than in 2D (for instance $n_k^{\text{adj}} = 4$ in Figure 3.4). In anticipation of unilateral contact formulations, (3.34) also defines a tangential plane at slave node k , from which the two unit tangent vectors $\boldsymbol{\tau}_k^\xi$ and $\boldsymbol{\tau}_k^\eta$ can be chosen to form an orthonormal basis together with \mathbf{n}_k as

$$\mathbf{n}_k \cdot \boldsymbol{\tau}_k^\xi = 0, \quad \boldsymbol{\tau}_k^\eta = \mathbf{n}_k \times \boldsymbol{\tau}_k^\xi. \quad (3.35)$$

As in 2D, mortar segments must be defined such that the shape function integrands in (3.19) and (3.20) are C^1 -continuous on these surface subsets. However, it is quite obvious that this task is much more complex in three dimensions than it has been in two dimensions, because mortar segments are now arbitrarily shaped polygons as compared with line segments in the 2D case. Beyond that, the choice of an adequate mortar integration surface itself is quite difficult. In the 2D mortar mesh tying formulation discussed above, integration is performed directly on the slave surface $\Gamma_{c,h}^{(1)}$. Unfortunately, it is not trivial to directly transfer this approach to three dimensions, because of the possible warping of surface facets. Thus, a slightly simplified coupling algorithm, performing integration not on the slave surface $\Gamma_{c,h}^{(1)}$ itself, but on its geometrical approximation with piecewise flat segments, has been proposed in Puso [129] and will also be employed here. For further details and an in-depth mathematical analysis of this algorithm, the reader is also referred to Puso and Laursen [132, 133] and Dickopf and Krause [32].

In Figure 3.5, the main steps of the 3D numerical integration algorithm for the mortar integrals in \mathbf{D} and \mathbf{M} are illustrated. In the following, the algorithm is outlined for one pair of slave and master elements (s, m) , which are close to each other and thus form an arbitrary overlap.

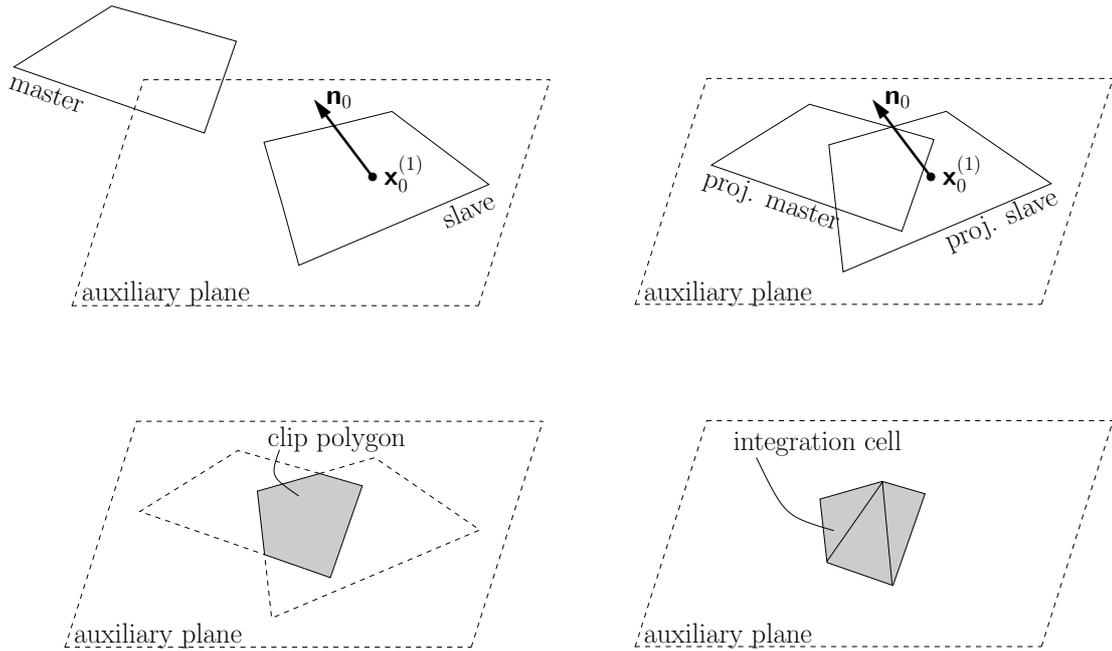


Figure 3.5: Main steps of 3D mortar coupling of one slave and master element pair. Construct an auxiliary plane (top left), project slave and master nodes into the auxiliary plane (top right), perform polygon clipping (bottom left), divide clip polygon into triangular integration cells and perform Gauss integration (bottom right).

Algorithm 1

1. Construct an auxiliary plane for numerical integration based on the slave element center $\mathbf{x}_0^{(1)}$ and the corresponding unit normal vector \mathbf{n}_0 .
2. Project all n_s^e slave element nodes $\mathbf{x}_k^{(1)}$, $k = 1, \dots, n_s^e$ onto the auxiliary plane along \mathbf{n}_0 to obtain the projected slave nodes $\tilde{\mathbf{x}}_k^{(1)}$. Steps 1 and 2 can also be interpreted as a geometrical approximation of the slave surface removing element warping.
3. Project all n_m^e master element nodes $\mathbf{x}_l^{(2)}$, $l = 1, \dots, n_m^e$ onto the auxiliary plane along \mathbf{n}_0 to obtain the projected master nodes $\tilde{\mathbf{x}}_l^{(2)}$.
4. Find the clip polygon of the projected slave and master elements in the auxiliary plane by applying a clipping algorithm, see e.g. Foley [42].
5. Establish n_{cell} triangular integration cells by applying Delaunay triangulation to the clip polygon. Each integration cell consists of three vertices $\tilde{\mathbf{x}}_v^{\text{cell}}$, $v = 1, 2, 3$ and is interpolated by standard triangular shape functions on the integration cell parameter space $\tilde{\boldsymbol{\eta}} = \{(\tilde{\xi}, \tilde{\eta}) | \tilde{\xi} \geq 0, \tilde{\eta} \geq 0, \tilde{\xi} + \tilde{\eta} \leq 1\}$.
6. Define n_{gp} Gauss integration points with coordinates $\tilde{\boldsymbol{\eta}}_g$, $g = 1, \dots, n_{\text{gp}}$ on each cell and project back along \mathbf{n}_0 to slave and master elements to obtain $\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)$ and $\boldsymbol{\xi}^{(2)}(\tilde{\boldsymbol{\eta}}_g)$.

7. Perform Gauss integration of $D_{jk(s,m)}$ and $M_{jl(s,m)}$, $j, k = 1, \dots, n_s^e$ and $l = 1, \dots, n_m^e$ on all integration cells

$$D_{jk(s,m)} = \sum_{c=1}^{n_{\text{cell}}} \left(\sum_{g=1}^{n_{\text{gp}}} w_g \Phi_j^{(1)}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) N_k^{(1)}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) J_c \right), \quad (3.36)$$

$$M_{jl(s,m)} = \sum_{c=1}^{n_{\text{cell}}} \left(\sum_{g=1}^{n_{\text{gp}}} w_g \Phi_j^{(1)}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) N_l^{(2)}(\boldsymbol{\xi}^{(2)}(\tilde{\boldsymbol{\eta}}_g)) J_c \right). \quad (3.37)$$

where J_c , $c = 1, \dots, n_{\text{cell}}$ is the integration cell Jacobian determinant.

Expressions (3.36) and (3.37) represent contributions to D_{jk} and M_{jl} given by one slave and master element pair (s, m) . Total quantities are obtained by summing up all slave and master element pair contributions. As pointed out in Puso [129], the above algorithm relies on the fact that the clip polygons of all slave and master element pairs are convex. For further explanations on prerequisites and properties of this numerical integration procedure, the reader is referred to the original paper by Puso [129].

In this work, seven point integration is used, which allows to exactly integrate polynomials of up to fifth-order. This order of accuracy is sufficient to exactly integrate (3.36) and (3.37) for *tri3* surface facets and unwarped *quad4* surface facets. Typical constant stress patch tests on flat interfaces could even be satisfied with much fewer quadrature points. However, it should be pointed out that in the case of surface facet warping, the mapping between slave and master sides introduces rational polynomial functions into the integrands in (3.36) and (3.37), and thus the numerical quadrature rule can never reproduce the exact integral value in such cases. However, numerical results including mesh refinement studies on curved mesh tying interfaces in Section 3.5 will demonstrate that the suggested choice of seven Gauss points per integration cell provides a sufficiently accurate quadrature rule. Figure 3.6 illustrates the generation of integration cells for 3D mortar coupling with a more complex example.

While Algorithm 1 undoubtedly provides the highest achievable accuracy for the numerical integration of D_{jk} and M_{jl} in 3D, some computationally more efficient alternatives have also been suggested in the literature. One prominent example is the simplified integration algorithm proposed in Fischer and Wriggers [39, 40] and later reused in De Lorenzis et al. [29] and Tur et al. [163]. Instead of thoroughly sub-dividing the mesh tying or contact interface into mortar segments, the numerical integration is simply performed elementwise in that approach, deliberately ignoring kinks of the functions to be integrated. Consequently, the devised integration schemes may indeed offer an appealing computational efficiency, but inevitably bring about difficulties with respect to accuracy of numerical integration. Even the exact satisfaction of a simple two-dimensional patch test, as investigated in Fischer and Wriggers [39], is strongly influenced by the total number of Gauss points chosen per slave element. An interesting improvement of this approach is suggested in Unger et al. [164], where adaptive refinement of the integration cells is performed based on a hierarchical quadtree structure. Simply speaking, refinement is only performed close to the kinks of the integrands in (3.36) and (3.37) and thus the associated error of numerical integration can be reduced.

In contrast to the 2D case, an extension of the segmentation and integration algorithm to second-order interpolation needs some additional considerations for three-dimensional mortar

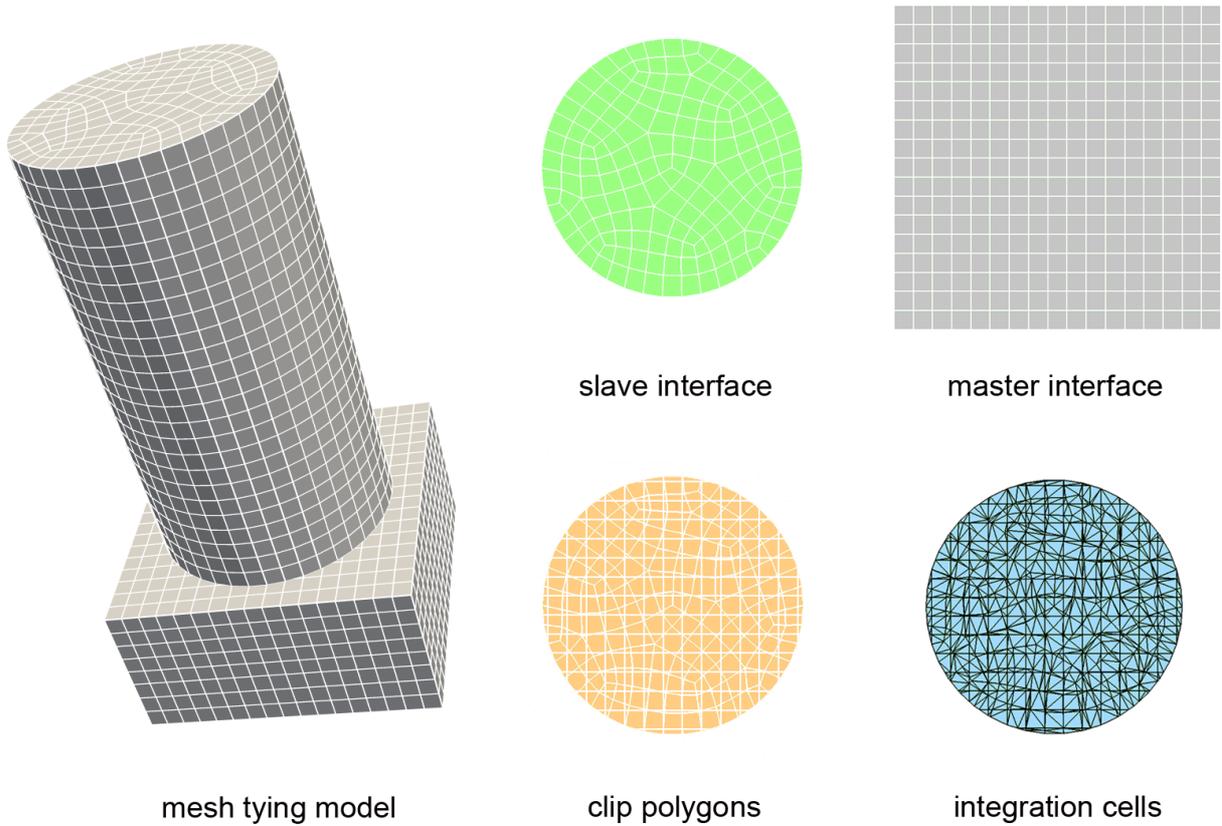


Figure 3.6: Main steps of 3D mortar coupling for a representative mesh tying example.

mesh tying problems. As explained above, the presented method for first-order interpolation is based on the projection of flattened surface elements. This approach has been directly extended to quadratic finite elements in Puso et al. [134], and is also employed here. The basic idea in Puso et al. [134] is to subdivide quadratic surface elements into linearly interpolated segments as exemplarily illustrated in Figure 3.7 for *quad9* facets. Numerical integration according to Algorithm 1 is then performed on the subsegments. As an example, consider the following mapping between parent element space and subsegment space of subsegment *sub3* for the *quad9* element in Figure 3.7, which is given by

$$\boldsymbol{\xi}^{\text{sub3}}(\boldsymbol{\xi}^{(1)}) = \begin{bmatrix} 2\xi^{(1)} - 1 \\ 2\eta^{(1)} - 1 \end{bmatrix}. \quad (3.38)$$

Similar mapping rules can also be readily established for *tri6* and *quad8* surface facets. It is important to point out that the approximation introduced by subdividing mortar elements only affects the integration domain itself, which no longer reflects the underlying quadratic finite element surfaces correctly. Yet, by making use of the aforementioned geometric mappings from parent element space to subsegment space and vice versa, one is still able to properly evaluate the higher-order shape function products in (3.36) and (3.37).

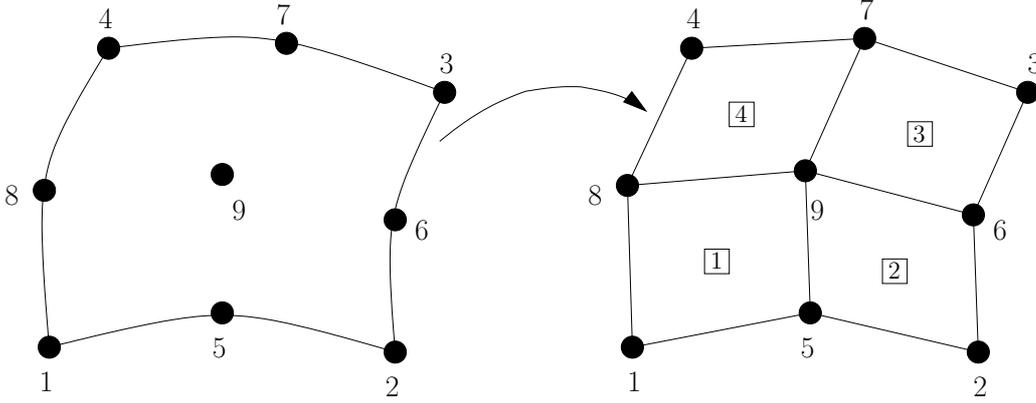


Figure 3.7: Subdivision of interface elements with second-order interpolation. Exemplarily, a *quad9* element is split into four *quad4* subsegments *sub1–sub4*, to which the 3D mortar integration algorithm is then applied nearly unchanged.

3.2.4 Time discretization

Time discretization of the semi-discrete mesh tying problem stated in (3.23) and (3.24) is based on the implicit time integration schemes discussed at length in Section 2.3.3, i.e first and foremost the generalized- α method. Basically, the fully discrete equilibrium of forces (2.81) only needs to be extended by the additional interface contribution $\mathbf{f}_{\text{mt}}(\boldsymbol{\lambda}) = \mathbf{B}_{\text{mt}}^{\text{T}} \boldsymbol{\lambda}$ and supplemented by the discrete mesh tying constraints $\mathbf{g}_{\text{mt}}(\mathbf{d}) = \mathbf{0}$. Thus, the final space and time discretized version of the mesh tying problem formulation is given as

$$\mathbf{M} \mathbf{a}_{n+1-\alpha_m} + \mathbf{C} \mathbf{v}_{n+1-\alpha_f} + \mathbf{f}_{\text{int}}(\mathbf{d}_{n+1-\alpha_f}) + \mathbf{f}_{\text{mt}}(\boldsymbol{\lambda}_{n+1-\alpha_f}) - \mathbf{f}_{\text{ext},n+1-\alpha_f} = \mathbf{0}, \quad (3.39)$$

$$\mathbf{g}_{\text{mt}}(\mathbf{d}_{n+1}) = \mathbf{0}. \quad (3.40)$$

Remember that the global vectors of discrete accelerations $\mathbf{a}_{n+1-\alpha_m}$ and velocities $\mathbf{v}_{n+1-\alpha_f}$ can be expressed exclusively in terms of the unknown displacements $\mathbf{d}_{n+1-\alpha_f}$ and known quantities at time t_n as defined in (2.75)–(2.80). An interesting point to discuss is the choice of adequate points in time for the evaluation of \mathbf{f}_{mt} and \mathbf{g}_{mt} in (3.39) and (3.40), where the generalized mid-point $t_{n+1-\alpha_f}$ has been selected for the interface forces, whereas the end-point t_{n+1} is the preferred choice for the constraints.

Evaluating the interface forces at $t_{n+1-\alpha_f}$ is self-evident due to the fact that internal and external force contributions are also associated with that point in time. As discussed for the internal force vector $\mathbf{f}_{\text{int}}(\mathbf{d}_{n+1-\alpha_f})$ in (2.83) and (2.84), there exist two possible ways of evaluating the mesh tying force vector $\mathbf{f}_{\text{mt}}(\boldsymbol{\lambda}_{n+1-\alpha_f})$, namely based on trapezoidal and midpoint rules. However, due to the fact that the interface forces are linear in $\boldsymbol{\lambda}$ and independent of the displacements, i.e. $\mathbf{f}_{\text{mt}} \neq \mathbf{f}_{\text{mt}}(\mathbf{d})$, and \mathbf{f}_{mt} , the two alternatives lead to identical results here. The mesh tying contribution can simply be written as

$$\mathbf{f}_{\text{mt}}(\boldsymbol{\lambda}_{n+1-\alpha_f}) = \mathbf{B}_{\text{mt}}^{\text{T}} [(1 - \alpha_f) \boldsymbol{\lambda}_{n+1} + \alpha_f \boldsymbol{\lambda}_n] := \mathbf{B}_{\text{mt}}^{\text{T}} \boldsymbol{\lambda}_{n+1-\alpha_f}, \quad (3.41)$$

thus introducing the mid-point Lagrange multipliers $\boldsymbol{\lambda}_{n+1-\alpha_f}$ as unknowns in addition to the unknown displacements $\mathbf{d}_{n+1-\alpha_f}$. For the constraints \mathbf{g}_{mt} , it is less obvious why they are evaluated

at the end-point t_{n+1} . As a matter of fact, this choice is arbitrary for mesh tying problems and could be shifted to any mid-point in the time interval $[t_n, t_{n+1}]$ due to the linearity of the considered tied contact constraints. However, evaluating the constraints at the end-point t_{n+1} of the time interval will become mandatory in the context of unilateral contact problems, cf. Section 4.2.5, and thus this approach is already followed here for the sake of consistency.

Finally, it is emphasized that the energy-momentum conserving time integration schemes presented in Section 2.3.3, such as the energy-momentum method (EMM) and its generalized version (GEMM), can be applied to mesh tying systems without any conceptual differences. As mentioned in Section 2.3.3, the only difference between the GEMM and the generalized- α method lies in the way the discrete internal force vector \mathbf{f}_{int} is evaluated, and this is completely unaffected by the additional mesh tying interface forces and constraints.

3.2.5 Conservation laws

The fundamental concepts of conservation laws concerning linear momentum, angular momentum and mechanical energy have already been introduced in Section 2.1.3. At this point, the proposed mesh tying formulation based on a mortar finite element discretization shall be analyzed in this regard. The focus is first set on conservation properties of the semi-discrete formulation in (3.23) and (3.24), i.e. independent of the employed time integration scheme. Time discretization typically poses additional restrictions on the satisfaction of the momentum balance equations. In the case of mechanical energy, a meaningful analysis of conservation laws is actually only possible when taking into account the time integration scheme.

In the semi-discrete setting, conservation of linear momentum as well as angular momentum can be interpreted as a kind of consistency check of the employed finite element discretization. For the nonlinear solid mechanics framework considered here, linear and angular momentum conservation is typically tested by analyzing translational and rotational invariance of all contributions to the discretized principle of virtual work. Here, it is sufficient to have a closer look at the mesh tying contribution in (3.18). As discussed in Puso [129], linear momentum conservation of the semi-discrete system is demonstrated by inserting a constant discrete vector $\mathbf{w} \neq \mathbf{0}$ into (3.21) for all (arbitrary) nodal displacement weighting functions, i.e. $\delta \mathbf{d}_k^{(1)} = \mathbf{w}$, $k = 1, \dots, n^{(1)}$ and $\delta \mathbf{d}_l^{(2)} = \mathbf{w}$, $l = 1, \dots, n^{(2)}$. This yields

$$\sum_{j=1}^{m^{(1)}} \left(\sum_{k=1}^{n^{(1)}} \mathbf{w}^T \mathbf{D}[j, k] - \sum_{l=1}^{n^{(2)}} \mathbf{w}^T \mathbf{M}[j, l] \right) \lambda_j = 0. \quad (3.42)$$

Since \mathbf{w} is non-zero, (3.42) is only satisfied if

$$\sum_{j=1}^{m^{(1)}} \left(\sum_{k=1}^{n^{(1)}} \mathbf{D}[j, k] \lambda_j - \sum_{l=1}^{n^{(2)}} \mathbf{M}[j, l] \lambda_j \right) := \mathbf{f}^{(1)} - \mathbf{f}^{(2)} = \mathbf{0}, \quad (3.43)$$

which means that the sums of all discrete mesh tying forces on the slave surface $\mathbf{f}^{(1)}$ and on the master surface $\mathbf{f}^{(2)}$ have to match. Thus, the proposed method *exactly* conserves linear momentum if master and slave side (i.e. the mortar integral matrices \mathbf{D} and \mathbf{M}) are both integrated based on the same numerical integration scheme, as already postulated in Section 3.2.2. This can be

proven easily, considering the fact that the standard finite element basis functions $N_k^{(1)}$ on the slave side and $N_l^{(2)}$ on the master side each fulfill a partition of unity, i.e. $\sum_{k=1}^{n^{(1)}} N_k^{(1)} = 1$ and $\sum_{l=1}^{n^{(2)}} N_l^{(2)} = 1$. This property yields

$$\sum_{k=1}^{n^{(1)}} \mathbf{D}[j, k] - \sum_{l=1}^{n^{(2)}} \mathbf{M}[j, l] = \left(\int_{\Gamma_{c,h}^{(1)}} \Phi_j dA_0 - \int_{\Gamma_{c,h}^{(2)}} \Phi_j dA_0 \right) \mathbf{I}_{\text{ndim}} = \mathbf{0}, \quad \forall j = 1, \dots, m^{(1)}, \quad (3.44)$$

which concludes the proof. If the matrix entries in \mathbf{D} were integrated independently of the matrix entries in \mathbf{M} , e.g. on an elementwise basis, then linear momentum conservation would *not* be guaranteed in general.

Quite similarly, angular momentum conservation of the semi-discrete system is typically investigated by inserting $\delta \mathbf{d}_k^{(1)} = \mathbf{w} \times \mathbf{x}_k^{(1)}$, $k = 1, \dots, n^{(1)}$ and $\delta \mathbf{d}_l^{(2)} = \mathbf{w} \times \mathbf{x}_l^{(2)}$, $l = 1, \dots, n^{(2)}$ into (3.21), viz.

$$\sum_{j=1}^{m^{(1)}} \left[\sum_{k=1}^{n^{(1)}} (\mathbf{w} \times \mathbf{x}_k^{(1)})^T \mathbf{D}[j, k] - \sum_{l=1}^{n^{(2)}} (\mathbf{w} \times \mathbf{x}_l^{(2)})^T \mathbf{M}[j, l] \right] \lambda_j = 0. \quad (3.45)$$

Since \mathbf{w} is again non-zero, (3.45) is only satisfied if

$$\sum_{j=1}^{m^{(1)}} \left[\sum_{k=1}^{n^{(1)}} \mathbf{x}_k^{(1)} \times (\mathbf{D}[j, k] \lambda_j) - \sum_{l=1}^{n^{(2)}} \mathbf{x}_l^{(2)} \times (\mathbf{M}[j, l] \lambda_j) \right] := \mathbf{m}^{(1)} - \mathbf{m}^{(2)} = \mathbf{0}. \quad (3.46)$$

Hence, the sum of all discrete interface moments on the slave surface $\mathbf{m}^{(1)}$ and on the master surface $\mathbf{m}^{(2)}$ have to match. Interestingly, the condition in (3.46) can also be reformulated as

$$\sum_{k=1}^{n^{(1)}} \mathbf{D}[j, k] \mathbf{x}_k^{(1)} - \sum_{l=1}^{n^{(2)}} \mathbf{M}[j, l] \mathbf{x}_l^{(2)} = \mathbf{0}, \quad \forall j = 1, \dots, m^{(1)}, \quad (3.47)$$

or even simpler as

$$\mathbf{D}\mathbf{x}_S - \mathbf{M}\mathbf{x}_M = \mathbf{0}, \quad (3.48)$$

where all discrete nodal positions of slave and master nodes in the current configuration are summarized in the vectors \mathbf{x}_S and \mathbf{x}_M , respectively. Expression (3.48) has the same structure as the mesh tying constraint $\mathbf{g}_{\text{mt}}(\mathbf{d}) = \mathbf{D}\mathbf{d}_S - \mathbf{M}\mathbf{d}_M = \mathbf{0}$ defined in (3.22), when written in terms of nodal positions \mathbf{x} instead of displacements \mathbf{d} . Taking into account the well-known connection between reference and current configuration, i.e. $\mathbf{x}_{S/M} = \mathbf{X}_{S/M} + \mathbf{d}_{S/M}$, angular momentum conservation requires that

$$\mathbf{D}\mathbf{X}_S - \mathbf{M}\mathbf{X}_M + \underbrace{\mathbf{D}\mathbf{d}_S - \mathbf{M}\mathbf{d}_M}_{=0} = \mathbf{0}. \quad (3.49)$$

Herein, the second part vanishes due to the mesh tying constraint $\mathbf{g}_{\text{mt}}(\mathbf{d}) = \mathbf{0}$. In other words, angular momentum conservation is guaranteed when the mesh tying constraints are also satisfied

in the reference configuration. While this is trivial for flat surfaces, an additional mesh initialization procedure based on the ideas in Puso [129] is applied for curved interfaces. Basically, this mesh initialization simply advocates a slight relocation of slave nodes $\mathbf{X}_k^{(1)}$ in the reference configuration such that (3.49) is satisfied. As pointed out in Puso [129], the method works well in most situations, but may encounter limitations due to possible element inversions in the case of coarse meshes and very large interface curvatures. Two quite similar extensions are proposed in Puso [129] and Hesch and Betsch [71], with the first being based on a rotation of the constraints using local element frames and the latter on a segment-wise reformulation.

While a detailed analysis of conservation laws in the time discretized setting is beyond the scope of this thesis, at least some brief remarks are given here. It will be seen in Section 4.2.5 that energy and momentum conservation are quite intricate for the unilateral contact case. However, at least the pure mesh tying case considered in the current chapter comes up with favorable results in this regard. In Section 2.3.3, the EMM originally proposed by Simo and Tarnow [150] has been introduced as a suitable framework for exact energy and momentum conservation in the context of nonlinear solid dynamics. Due to the fact that mesh tying only amends the problem formulation by simple equality constraints, all conservation properties of time integration schemes for nonlinear solid dynamics are also retained for tied contact systems. A corresponding numerical example using the EMM as time integration scheme will be presented in Section 3.5.3 in order to emphasize this statement.

3.3 Discrete Lagrange multiplier spaces

The discrete Lagrange multiplier space \mathcal{M}_h and associated shape functions Φ_j , $j = 1, \dots, m^{(1)}$, on the slave side of the mesh tying interface were already introduced in Section 3.2.1, although not specified in detail. Yet, this choice of the discrete Lagrange multiplier space is crucial for both the mathematical properties and the numerical efficiency of the resulting mortar approach. There exists a vast amount of literature discussing all relevant characteristics associated with the choice of \mathcal{M}_h , such as inf-sup stability of the underlying mixed formulation and optimal a priori error bounds, see e.g. Ben Belgacem [11], Bernardi et al. [16], Seshaiyer and Suri [147] and Wohlmuth [175]. With stability investigations and a priori error estimates not being in the focus of interest of this thesis, the following considerations rely on the fact that there exists a well-established framework of proofs and rigorous mathematical analyses, which guarantees the applicability of all discrete Lagrange multiplier spaces discussed here to mortar mesh tying problems. For a comprehensive overview, the reader is referred to Wohlmuth [176] and the references therein.

Throughout this thesis, two different families of discrete Lagrange multipliers, namely *standard* and so-called *dual* Lagrange multipliers, will be distinguished. Standard Lagrange multipliers represent the classical approach for mortar methods (cf. Ben Belgacem [11], Seshaiyer and Suri [147]) and are usually taken from the finite dimensional subset $\mathcal{W}_h^{(1)} \subset \mathcal{W}^{(1)}$ on the slave side of the interface, where $\mathcal{W}^{(1)}$ is the trace space of $\mathcal{V}^{(1)}$, as explained in Section 3.1.2. Thus, standard mortar methods typically lead to identical shape functions for Lagrange multiplier and slave displacement interpolation, i.e. $\Phi_j = N_j^{(1)}$.

In contrast, the dual approach is motivated by the observation that the Lagrange multipliers physically represent fluxes (tractions) on the mesh tying interface in the continuous setting. This duality argument is then reflected by constructing dual Lagrange multiplier shape

functions based on a so-called biorthogonality condition with the displacements in $\mathcal{W}_h^{(1)}$, see e.g. Wohlmuth [175]. While they are, in general, not continuous and cannot be interpreted as a trace of conforming finite elements, the biorthogonality condition assures that the Lagrange multiplier shape functions Φ_j are again well-defined and satisfy all required approximation properties. One crucial advantage of the dual approach lies in the fact that it heavily facilitates the treatment of typical mortar coupling conditions at the interface, while at the same time preserving the mathematical optimality of the method. Going back to (3.22), the discrete mesh tying condition can alternatively be expressed as

$$\mathbf{d}_S = \mathbf{D}^{-1} \mathbf{M} \mathbf{d}_{\mathcal{M}} := \mathbf{P} \mathbf{d}_{\mathcal{M}}, \quad (3.50)$$

where $\mathbf{P} = \mathbf{D}^{-1} \mathbf{M}$ represents the discrete interface coupling operator. As will be demonstrated later on for both mesh tying and unilateral contact problems, dual Lagrange multipliers avoid the necessity of solving a mass matrix type of system when evaluating (3.50), but *localize* the coupling conditions instead. Algebraically, this advantageous property of dual Lagrange multipliers can be observed by the mortar matrix \mathbf{D} in (3.19) reducing to a diagonal matrix. This allows for very efficient condensation procedures of the discrete Lagrange multiplier degrees of freedom, which completely remove the undesirable saddle point structure of the underlying mesh tying and later unilateral contact systems, see Sections 3.4.3 and 4.5.4.

In the following paragraphs, a detailed overview of how to define standard and dual Lagrange multipliers for both 2D and 3D mesh tying problems in the context of first-order and second-order finite elements is given. As explained in Section 2.5.2, mortar methods are optimal in the sense that the same convergence rates achieved in conforming situations can be expected to be retained for mesh tying problems, too. Typically, the discretization error $\mathbf{u} - \mathbf{u}_h$ is measured in the L^2 -norm and in the energy norm. Due to their fundamental importance, the optimal convergence rates already introduced in Section 2.3.4 are repeated here once again. For the L^2 -norm of the error $\|\mathbf{u} - \mathbf{u}_h\|$ one obtains $\mathcal{O}(h^{p+1})$, and for the energy norm of the error $\|\mathbf{u} - \mathbf{u}_h\|_{\text{energy}}$ the optimal convergence rate is $\mathcal{O}(h^p)$.

Apart from the choice of standard or dual Lagrange multipliers, further possible free parameters are the polynomial degree of the shape functions Φ_j and the total number of slave nodes carrying discrete Lagrange multipliers. Therefore, for all upcoming investigations the following four quantities are introduced or, in the case of p , brought to mind:

- p polynomial degree of the displacement shape functions $N_k^{(1)}$,
- p_λ polynomial degree of the Lagrange multiplier shape functions Φ_j ,
- r reproduction order of the discrete slave side displacement space $\mathcal{W}_h^{(1)}$,
- r_λ reproduction order of the discrete Lagrange multiplier space \mathcal{M}_h .

Especially for the discrete Lagrange multipliers, it is important to distinguish between polynomial degree and reproduction order. While p_λ denotes the highest polynomial order contained in the Lagrange multiplier interpolation, the reproduction order r_λ only refers to the order up to which all polynomial terms are *completely* contained in the Lagrange multiplier interpolation, and consequently $r_\lambda \leq p_\lambda$. For the slave displacements, no distinction between polynomial order

and reproduction properties is necessary. All first-order ($p = 1$) and second-order ($p = 2$) finite elements employed in this thesis yield $r \equiv p$ due to completeness requirements.

Coming back to the polynomial degree p_λ of the Lagrange multiplier shape functions as an interesting variable for mortar methods, the natural choice undoubtedly is to retain this polynomial degree identical to that of the slave displacement shape functions, i.e. $p_\lambda = p$, and to equip all slave nodes with discrete Lagrange multipliers. However, as discussed for instance in Puso et al. [134], Seshaiyer and Suri [147] and Popp et al. [128], there exist certain situations, e.g. for quadratic finite elements, where other choices are more preferable. To give an example: It has been demonstrated in Seshaiyer and Suri [147] in the context of a Laplace operator model problem and standard Lagrange multipliers, that for conforming finite elements of polynomial degree p , piecewise polynomials of degree $p_\lambda = r_\lambda = p - 1$ are sufficient for the Lagrange multiplier interpolation to preserve the optimal convergence of the discretization error. However, piecewise polynomials of degree $p_\lambda = r_\lambda = p - 2$, where $p \geq 2$, lead to a predicted deterioration of the discretization error by $\mathcal{O}(h^{1/2})$ measured in the energy norm. For second-order finite element interpolation, this means that only $\mathcal{O}(h^{3/2})$ instead of an optimal $\mathcal{O}(h^2)$ convergence in the energy norm could be expected from the theory for such a choice of Lagrange multipliers.

3.3.1 Standard Lagrange multipliers

As mentioned above, standard Lagrange multipliers based on the trace space $\mathbf{W}_h^{(1)}$ of the finite element discretization are the classical choice for the discrete Lagrange multiplier space in mortar methods. Apart from some special considerations necessary at so-called crosspoints where more than two subdomains meet, see Section 3.3.3, this approach leads to identical shape functions for slave displacement and Lagrange multiplier interpolation, i.e. $\Phi_j = N_j^{(1)}$. Thinking of its definition in (3.19), the mortar matrix \mathbf{D} thus remains a typical mass matrix with a densely populated inverse. Slave side and master side displacements are globally coupled, which is reflected by a densely populated projection operator \mathbf{P} in (3.50). Common choices for first-order and second-order finite element interpolation are presented in the upcoming paragraphs.

3.3.1.1 First-order interpolation in 2D

The first-order interpolation case in 2D is typically characterized by *tri3* and *quad4* finite elements in the individual subdomains, and consequently the mesh tying interface consists of *line2* segments. Standard Lagrange multiplier shape functions Φ_j then coincide with the well-known one-dimensional finite element hat functions, viz.

$$N_1^{(1)}(\xi) = \frac{1}{2}(1 - \xi), \quad N_2^{(1)}(\xi) = \frac{1}{2}(1 + \xi). \quad (3.51)$$

Thus, the Lagrange multiplier interpolation is non-negative in this case, and obviously one obtains $p_\lambda = r_\lambda = 1$. The superscript $\cdot^{(1)}$ for the finite element parameter space on the slave side $\Gamma_c^{(1)}$ of the interface has been omitted here for the ease of notation. Throughout the whole section on discrete Lagrange multiplier spaces, local coordinates ξ (for 2D and 3D) and η (only for 3D) always represent slave side quantities $\xi^{(1)}$ and $\eta^{(1)}$ without explicit labeling.

3.3.1.2 First-order interpolation in 3D

Similarly, the first-order interpolation case in 3D is typically characterized by *tet4* and *hex8* finite elements for the volume mesh. Hence, the resulting surface facets on $\Gamma_{c,h}^{(1)}$ are of *tri3* and *quad4* shape. The associated standard Lagrange multiplier shape functions Φ_j are identical to the usual linear two-dimensional hat functions, i.e.

$$N_1^{(1)}(\xi, \eta) = 1 - \xi - \eta, \quad N_2^{(1)}(\xi, \eta) = \xi, \quad N_3^{(1)}(\xi, \eta) = \eta, \quad (3.52)$$

for *tri3* facets in combination with the well-known parametrization of a triangle given as $\xi \geq 0$, $\eta \geq 0$ and $\xi + \eta \leq 1$, or by the bilinear two-dimensional hat functions, i.e.

$$\begin{aligned} N_1^{(1)}(\xi, \eta) &= \frac{1}{4}(1 - \xi)(1 - \eta), & N_2^{(1)}(\xi, \eta) &= \frac{1}{4}(1 + \xi)(1 - \eta), \\ N_3^{(1)}(\xi, \eta) &= \frac{1}{4}(1 + \xi)(1 + \eta), & N_4^{(1)}(\xi, \eta) &= \frac{1}{4}(1 - \xi)(1 + \eta), \end{aligned} \quad (3.53)$$

for *quad4* facets. In that case, the parameter space is defined as $-1 \leq \xi \leq 1$ and $-1 \leq \eta \leq 1$. Again, the resulting Lagrange multiplier shape functions are all non-negative and polynomial degree and reproduction order coincide, i.e. $p_\lambda = r_\lambda = 1$.

3.3.1.3 Second-order interpolation in 2D

Second-order finite element interpolation for 2D problems yields *tri6*, *quad8* and *quad9* elements in the subdomains, but in either case the mesh tying interface consists of *line3* segments. The corresponding Lagrange multiplier shape functions Φ_j coincide with the well-known one-dimensional quadratic polynomials given by

$$N_1^{(1)}(\xi) = \frac{1}{2}\xi(\xi - 1), \quad N_2^{(1)}(\xi) = \frac{1}{2}\xi(\xi + 1), \quad N_3^{(1)}(\xi) = (1 - \xi)(1 + \xi). \quad (3.54)$$

In contrast to the aforementioned first-order cases, the Lagrange multiplier shape functions are no longer positive everywhere now. However, it can be shown by simple analysis that at least integral positivity on each element e of the slave surface $\Gamma_{c,h}^{(1)}$, i.e.

$$\int_e \Phi_j de > 0, \quad (3.55)$$

is satisfied. As will be seen later in the context of dual Lagrange multipliers, and especially for unilateral contact, assuring integral positivity of the Lagrange multiplier shape functions is an effective way to guarantee that the discrete mortar coupling operator $\mathbf{P} = \mathbf{D}^{-1}\mathbf{M}$ is well-defined. For the sake of completeness, it is pointed out that the polynomial and reproduction orders are given as $p_\lambda = r_\lambda = 2$ here.

Yet, as indicated in the opening remarks of Section 3.3, a reduced polynomial degree and thus reduced reproduction order of the Lagrange multiplier interpolation is theoretically also sufficient for quadratic FE. Obviously, one could think of only equipping the slave side corner nodes with discrete Lagrange multiplier degrees of freedom, but not the (higher-order) middle nodes. This procedure yields the standard hat functions as Lagrange multiplier shape functions and, simply speaking, the middle nodes on the slave side of the interface can then be interpreted as being master nodes. In terms of its classification, this approach gives $p_\lambda = r_\lambda = 1$.

3.3.1.4 Second-order interpolation in 3D

The spectrum of finite element interpolations considered in this thesis is completed by the second-order case in 3D, with *tet10*, *hex20* and *hex27* being the most common element types for the volume mesh. These elements generate corresponding interface facets that are of *tri6*, *quad8* and *quad9* shape, respectively. For the first case (*tri6* facets), the standard quadratic Lagrange multiplier shape functions $\Phi_j = N_j^{(1)}$ are given as

$$\begin{aligned} N_1^{(1)}(\xi, \eta) &= (1 - \xi - \eta)(1 - 2\xi - 2\eta), & N_2^{(1)}(\xi, \eta) &= \xi(2\xi - 1), \\ N_3^{(1)}(\xi, \eta) &= \eta(2\eta - 1), & N_4^{(1)}(\xi, \eta) &= 4\xi(1 - \xi - \eta), \\ N_5^{(1)}(\xi, \eta) &= 4\xi\eta, & N_6^{(1)}(\xi, \eta) &= 4\eta(1 - \xi - \eta). \end{aligned} \quad (3.56)$$

The second case (*quad8* facets) is characterized by serendipity shape functions, viz.

$$\begin{aligned} N_1^{(1)}(\xi, \eta) &= -\frac{1}{4}(1 - \xi)(1 - \eta)(1 + \xi + \eta), & N_2^{(1)}(\xi, \eta) &= -\frac{1}{4}(1 + \xi)(1 - \eta)(1 - \xi + \eta), \\ N_3^{(1)}(\xi, \eta) &= -\frac{1}{4}(1 + \xi)(1 + \eta)(1 - \xi - \eta), & N_4^{(1)}(\xi, \eta) &= -\frac{1}{4}(1 - \xi)(1 + \eta)(1 + \xi - \eta), \\ N_5^{(1)}(\xi, \eta) &= \frac{1}{2}(1 - \xi^2)(1 - \eta), & N_6^{(1)}(\xi, \eta) &= \frac{1}{2}(1 + \xi)(1 - \eta^2), \\ N_7^{(1)}(\xi, \eta) &= \frac{1}{2}(1 - \xi^2)(1 + \eta), & N_8^{(1)}(\xi, \eta) &= \frac{1}{2}(1 - \xi)(1 - \eta^2). \end{aligned} \quad (3.57)$$

Finally, the third case (*quad9* facets) yields biquadratic Lagrange polynomials, i.e.

$$\begin{aligned} N_1^{(1)}(\xi, \eta) &= \frac{1}{4}(1 - \xi)(1 - \eta)\xi\eta, & N_2^{(1)}(\xi, \eta) &= -\frac{1}{4}(1 + \xi)(1 - \eta)\xi\eta, \\ N_3^{(1)}(\xi, \eta) &= \frac{1}{4}(1 + \xi)(1 + \eta)\xi\eta, & N_4^{(1)}(\xi, \eta) &= -\frac{1}{4}(1 - \xi)(1 + \eta)\xi\eta, \\ N_5^{(1)}(\xi, \eta) &= -\frac{1}{2}(1 - \xi^2)(1 - \eta)\eta, & N_6^{(1)}(\xi, \eta) &= \frac{1}{2}(1 + \xi)(1 - \eta^2)\xi, \\ N_7^{(1)}(\xi, \eta) &= \frac{1}{2}(1 - \xi^2)(1 + \eta)\eta, & N_8^{(1)}(\xi, \eta) &= -\frac{1}{2}(1 - \xi)(1 - \eta^2)\xi, \\ N_9^{(1)}(\xi, \eta) &= (1 - \xi^2)(1 - \eta^2). \end{aligned} \quad (3.58)$$

Further details and illustrations on how to derive such second-order shape functions can, for instance, be found in Hughes [86]. Polynomial degree and reproduction are given as $p_\lambda = r_\lambda = 2$. As for the second-order interpolation case in 2D, neither of the three sets of shape functions defined here is positive everywhere. For the biquadratic *quad9* facets, it can again easily be shown that at least integral positivity according to (3.55) is satisfied. However, certain finite element shape functions of *quad8* and *tri6* surfaces do not even fulfill this fundamental requirement. As an example, the shape functions $N_1^{(1)}$ associated with a corner node of a *quad8* facet and of a *tri6* facet both violate the integral positivity condition. Examining the reference quadrilateral element $(-1, 1)^2$ and the reference triangle with corners $(0, 0)$, $(1, 0)$ and $(0, 1)$ in local coordinates

ξ and η , it becomes obvious that

$$\int_e N_1^{(1),quad8} de = \int_e -\frac{1}{4}(1-\xi)(1-\eta)(1+\xi+\eta) de < 0, \quad (3.59)$$

$$\int_e N_1^{(1),tri6} de = \int_e (1-\xi-\eta)(1-2\xi-2\eta) de = 0. \quad (3.60)$$

This argument motivates a simple but effective modification of standard Lagrange multiplier shape functions for second-order interpolation in 3D. Instead of choosing $m^{(1)} = n^{(1)}$ and the Lagrange multiplier shape functions Φ_j identical to the displacement shape functions $N_j^{(1)}$, a Lagrange multiplier interpolation based on (bi-)linear polynomials has been suggested and successfully applied to unilateral contact analysis in Puso et al. [134]. As already explained in Section 3.3.1.3, only the corner nodes carry discrete Lagrange multipliers in that case, but not the edge nodes and the possible center node of *quad9* elements, i.e. $m^{(1)} < n^{(1)}$. Another alternative discussed in Puso et al. [134] is that all slave nodes still carry discrete Lagrange multipliers, i.e. $m^{(1)} = n^{(1)}$, but the Lagrange multiplier interpolation is rather based on *piecewise* linear polynomials on subsegments as defined in Section 3.2.3. Both choices yield $p_\lambda = r_\lambda = 1$ and, furthermore, do not only satisfy integral positivity but even pointwise positivity ($\Phi_j \geq 0$).

3.3.2 Dual Lagrange multipliers

As discussed in the opening remarks of Section 3.3, the motivation for dual Lagrange multipliers lies in the fact that an extension of the master side basis functions to the slave side of the interface has a global support for standard Lagrange multipliers. Based on duality arguments, it is however possible to construct dual shape functions such that the interface coupling subproblem in (3.50) reduces to a *localized* form, which is algebraically characterized by the mortar matrix \mathbf{D} becoming a diagonal matrix and the projection operator \mathbf{P} only being sparsely populated.

Details on how to define dual Lagrange multiplier shape functions Φ_j using the so-called biorthogonality relationship with the standard displacement shape functions $N_k^{(1)}$ have first been presented in Scott and Zhang [146] and Wohlmuth [175]. A common notation of the biorthogonality condition is

$$\int_{\Gamma_{c,h}^{(1)}} \Phi_j N_k^{(1)} dA_0 = \delta_{jk} \int_{\Gamma_{c,h}^{(1)}} N_k^{(1)} dA_0, \quad j, k = 1, \dots, m^{(1)}. \quad (3.61)$$

Herein, δ_{jk} is the Kronecker delta, and the most common choice $m^{(1)} = n^{(1)}$ is assumed. For practical reasons, the biorthogonality condition is typically applied locally on each slave element e , yielding

$$\int_e \Phi_j N_k^{(1)} de = \delta_{jk} \int_e N_k^{(1)} de, \quad j, k = 1, \dots, m_e^{(1)}, \quad (3.62)$$

where $m_e^{(1)}$ represents the number of Lagrange multiplier nodes of the considered slave element. Taking into account the assumption that all nodes also carry discrete Lagrange multiplier degrees of freedom, $m_e^{(1)}$ is simply the number of nodes of the current slave facet. Comparing (3.61)

and (3.19) also clearly reveals why dual shape functions reduce the mortar matrix \mathbf{D} to a diagonal matrix. The dual shape functions resulting from (3.61), or rather from the elementwise version in (3.62), have the same polynomial order as the employed standard shape functions, i.e. $p_\lambda = p$. Moreover, it can easily be shown that the biorthogonality condition guarantees a partition of unity property, i.e. $\sum_j \Phi_j = 1$, $j = 1, \dots, m_e^{(1)}$, see Flemisch and Wohlmuth [41] for a proof. On the other hand, the shape functions Φ_j are discontinuous, and thus the reproduction order of a discrete *dual* Lagrange multiplier interpolation is reduced to $r_\lambda = 0$, which means that only the constant polynomials are completely included in the multiplier space. When keeping in mind the aforementioned theoretical results in Seshaiyer and Suri [147] for instance, this choice can at least for first-order FE ($p = 1$) still be expected to be sufficient in order not to impinge upon the optimal convergence of the discretization error.

For the sake of completeness, another commonly used property of dual shape functions is briefly introduced here. Combining the biorthogonality condition in (3.62) and the partition of unity property of the dual shape functions, it follows that

$$\int_e \Phi_j \, de = \int_e N_j^{(1)} \, de, \quad j = 1, \dots, m_e^{(1)}. \quad (3.63)$$

This feature will be crucial in Section 3.3.2.4 to assure integral positivity of the Lagrange multiplier shape functions. It is important to point out that the elementwise biorthogonality condition in (3.62) must be satisfied in the physical space, i.e. in the reference configuration for tied contact problems, and not simply in the finite element parameter space. Consequently, dual shape functions depend on the actual distortion of the individual underlying finite element, and cannot be defined a priori for non-constant slave element Jacobian determinants. Instead, a local mass matrix system of size $m_e^{(1)} \times m_e^{(1)}$ must be solved on each slave element. The first step for doing this is to introduce unknown linear coefficients a_{jk} such that

$$\Phi_j(\xi, \eta) = a_{jk} N_k^{(1)}(\xi, \eta), \quad \mathbf{A}_e = [a_{jk}] \in \mathbb{R}^{m_e^{(1)} \times m_e^{(1)}}, \quad (3.64)$$

It can easily be verified that, as second step, insertion of (3.64) into (3.62) yields the unknown coefficient matrix \mathbf{A}_e as:

$$\begin{aligned} \mathbf{A}_e &= \mathbf{D}_e \mathbf{M}_e^{-1}, \\ \mathbf{D}_e &= [d_{jk}] \in \mathbb{R}^{m_e^{(1)} \times m_e^{(1)}}, \quad d_{jk} = \delta_{jk} \int_e N_k^{(1)}(\xi, \eta) J(\xi, \eta) \, de, \\ \mathbf{M}_e &= [m_{jk}] \in \mathbb{R}^{m_e^{(1)} \times m_e^{(1)}}, \quad m_{jk} = \int_e N_j^{(1)}(\xi, \eta) N_k^{(1)}(\xi, \eta) J(\xi, \eta) \, de, \end{aligned} \quad (3.65)$$

where $J(\xi, \eta)$ is the slave element Jacobian determinant. The global basis functions are then obtained by a simple gluing process as is standard in finite element methods. In the following paragraphs, the resulting dual shape functions for first-order and second-order finite element interpolation will be presented. Details on these quite intricate constructions can also be found in Flemisch and Wohlmuth [41], Lamichhane and Wohlmuth [104], Lamichhane et al. [105], Wohlmuth [176], Wohlmuth et al. [179] and Popp et al. [128].

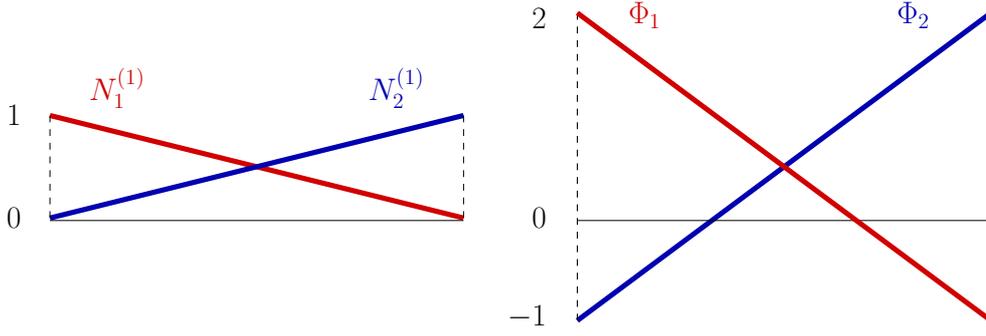


Figure 3.8: Slave side displacement shape functions $N_j^{(1)}$ (left) and dual Lagrange multiplier shape functions Φ_j (right) for a *line2* element.

3.3.2.1 First-order interpolation in 2D

As explained in Section 3.3.1.1, the first-order interpolation case in 2D leads to *line2* shaped mortar interface segments. With the Jacobian of *line2* segments being constant, the dual Lagrange multiplier shape functions determined by (3.64) and (3.65) are independent of element distortion, and can be defined a priori instead:

$$\Phi_1(\xi) = \frac{1}{2}(1 - 3\xi), \quad \Phi_2(\xi) = \frac{1}{2}(1 + 3\xi). \quad (3.66)$$

Figure 3.8 illustrates these dual shape functions along with their standard counterparts, i.e. the first-order slave displacement shape functions $N_j^{(1)}$. In contrast to the corresponding standard Lagrange multiplier case in Section 3.3.1.1, dual Lagrange multiplier shape functions can no longer be positive everywhere in order to fulfill the biorthogonality condition. However, due to the equality of the integral values established in (3.63), integral positivity according to (3.55) is guaranteed. Moreover, the above defined Φ_j are indeed locally linear polynomials and satisfy a partition of unity property, but nonetheless they represent discontinuous functions. Therefore, the following classification of the discrete Lagrange multiplier holds: $p_\lambda = 1$ and $r_\lambda = 0$, i.e. the reproduction order is reduced to constant polynomials.

3.3.2.2 First-order interpolation in 3D

As has been seen in Section 3.3.1.2, the first-order interpolation case in 3D generates *tri3* and *quad4* surface facets on the interface. The first type of mortar surface elements has a constant Jacobian and the associated dual shape functions again do not depend on a possible element distortion. Instead, (3.64) and (3.65) yield the following a priori definition:

$$\Phi_1(\xi, \eta) = 3 - 4\xi - 4\eta, \quad \Phi_2(\xi, \eta) = 4\xi - 1, \quad \Phi_3(\xi, \eta) = 4\eta - 1. \quad (3.67)$$

For *quad4* surface elements, however, the finite element approximation may yield non-constant Jacobians and thus an a priori definition of the corresponding dual shape functions is not possible in general. In other words, the dual shape functions depend on the element distortion in this case and must be individually computed for each slave element according to (3.64) and (3.65). Here,

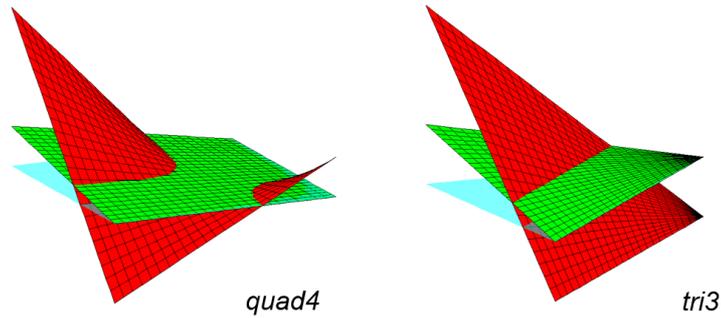


Figure 3.9: Displacement and dual Lagrange multiplier shape functions for an undistorted *quad4* element (left) and for a *tri3* element (right). In each case, the displacement shape function $N_j^{(1)}$ is illustrated in green and the dual shape function Φ_j in red.

only the results for the most straightforward scenario, namely for an undistorted quadrilateral, are given as an example. Dual Lagrange multiplier shape functions for this case are

$$\begin{aligned}
 \Phi_1(\xi, \eta) &= 4N_1^{(1)}(\xi, \eta) - 2N_2^{(1)}(\xi, \eta) + N_3^{(1)}(\xi, \eta) - 2N_4^{(1)}(\xi, \eta), \\
 \Phi_2(\xi, \eta) &= -2N_1^{(1)}(\xi, \eta) + 4N_2^{(1)}(\xi, \eta) - 2N_3^{(1)}(\xi, \eta) + N_4^{(1)}(\xi, \eta), \\
 \Phi_3(\xi, \eta) &= N_1^{(1)}(\xi, \eta) - 2N_2^{(1)}(\xi, \eta) + 4N_3^{(1)}(\xi, \eta) - 2N_4^{(1)}(\xi, \eta), \\
 \Phi_4(\xi, \eta) &= -2N_1^{(1)}(\xi, \eta) + N_2^{(1)}(\xi, \eta) - 2N_3^{(1)}(\xi, \eta) + 4N_4^{(1)}(\xi, \eta),
 \end{aligned} \tag{3.68}$$

where $N_j^{(1)}$, $j = 1, 2, 3, 4$, represent the standard bilinear shape functions. For exemplary calculations of element-specific dual shape functions in the case of distorted *quad4* surface elements, the interested reader is referred to Flemisch and Wohlmuth [41] and Hartmann et al. [66]. Figure 3.9 illustrates dual shape functions for both *tri3* and *quad4* elements along with their respective standard counterparts given in Section 3.3.1.2. Again, the resulting shape functions Φ_j are locally (bi-)linear polynomials. However, due to the fact that they are discontinuous and only satisfy partition of unity, one obtains $p_\lambda = 1$ and $r_\lambda = 0$.

3.3.2.3 Second-order interpolation in 2D

The second-order interpolation case in 2D is characterized by *line3* segments on the coupling interface. Thus, in general, the associated element Jacobian determinants are non-constant and the definition of dual shape functions Φ_j becomes element-specific. Again, only the exemplary case of an undistorted straight *line3* segment is presented here, for which the result of the Lagrange multiplier interpolation in (3.64) and (3.65) is given as

$$\begin{aligned}
 \Phi_1(\xi) &= N_1^{(1)}(\xi) - \frac{3}{4}N_3^{(1)}(\xi) + \frac{1}{2}, & \Phi_2(\xi) &= N_2^{(1)}(\xi) - \frac{3}{4}N_3^{(1)}(\xi) + \frac{1}{2}, \\
 \Phi_3(\xi) &= \frac{5}{2}N_3^{(1)}(\xi) - 1,
 \end{aligned} \tag{3.69}$$

where $N_j^{(1)}$, $j = 1, 2, 3$, represent the standard quadratic shape functions in one dimension, see Section 3.3.1.3. An illustration of the dual shape functions Φ_j and the associated standard shape

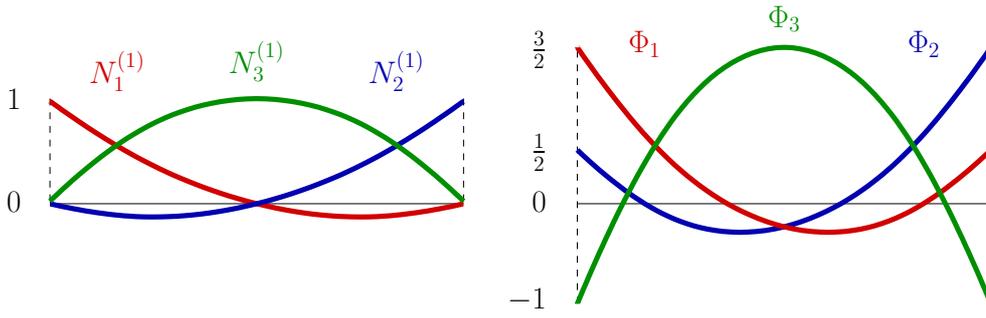


Figure 3.10: Slave side displacement shape functions $N_j^{(1)}$ (left) and dual Lagrange multiplier shape functions Φ_j (right) for an undistorted *line3* element.

functions is given in Figure 3.10. Thus, the resulting dual shape functions are locally quadratic polynomials, satisfying the integral positivity condition (3.55) as well as a partition of unity property. However, due to the involved discontinuities, the polynomial and reproduction orders are given as $p_\lambda = 2$ and $r_\lambda = 0$ here, i.e. still only constant polynomials can be completely reproduced. As discussed in Seshaiyer and Suri [147], this scenario leads to a predicted deterioration of the spatial convergence by half an order measured in the energy norm. Yet, numerical investigations performed in Seshaiyer and Suri [147] and later in this thesis for the 3D case do not confirm these theoretical predictions, but rather yield optimal convergence rates.

As discussed for standard Lagrange multiplier interpolation in Section 3.3.1.3, one could also think about not assigning discrete Lagrange multipliers to the middle nodes of *line3* segments and using linear dual shape functions for the remaining (corner) nodes instead. Due to the fact that the biorthogonality condition (3.62) only guarantees $r_\lambda = 0$ anyway, regardless of the actual polynomial degree p_λ , such a procedure seems especially favorable in the context of dual Lagrange multiplier interpolation. However, some additional numerical manipulations are required and will be exemplarily outlined in the following paragraph for the second-order interpolation case in 3D. For an alternative approach of generating higher-order dual shape functions in 2D based on Gauß–Lobatto nodes, the reader is also referred to Lamichhane and Wohlmuth [104].

3.3.2.4 Second-order interpolation in 3D

Interface facets of *tri6*, *quad8* and *quad9* shape need to be considered for second-order Lagrange multiplier interpolation in 3D problems. The corresponding element Jacobian determinants are non-constant in a general setting, and thus dual shape functions Φ_j need to be constructed in an element-specific way according to (3.64) and (3.65). However, this becomes much more difficult in 3D than it was in 2D, because the issues discussed for standard shape functions in Section 3.3.1.4 carry over and are aggravated by the fact that now the biorthogonality condition explicitly relies on non-zero shape functions integrals, i.e. $\int_e \Phi_j \, de = \int_e N_j^{(1)} \, de \neq 0$.

However, as shown in Section 3.3.1.4, this requirement might not be met for certain nodes of *quad8* and *tri6* facets, thus making the definition of feasible dual shape functions impossible. The *quad9* case satisfies integral positivity of all shape functions, and therefore is again straightforward. The standard biorthogonality condition (3.62) can be applied. To circumvent the problems occurring in the context of *quad8* and *tri6* facets, two novel strategies have been pro-

posed in Popp et al. [128], one based on (bi-)quadratic dual Lagrange multipliers ($m^{(1)} = n^{(1)}$) and the other one based on (bi-)linear dual Lagrange multipliers ($m^{(1)} < n^{(1)}$). The most important aspects of the derivation of both cases will be summarized in the following.

Locally quadratic dual Lagrange multipliers

As a first step towards defining suitable quadratic dual shape functions, a simple basis transformation for the displacement shape functions $N_j^{(1)}$ and nodal degrees of freedom on the slave surface is introduced, which guarantees that the resulting modified shape functions $\tilde{N}_j^{(1)}$ satisfy the integral positivity condition (3.55). As a consequence, dual Lagrange multiplier shape functions Φ_j constructed from these modified displacement shape functions and the standard biorthogonality condition (3.62) will also satisfy (3.55). The requirement of integral positivity still leaves a lot of freedom in arranging the actual basis transformation. Based on the usual node numbering in second-order finite elements, where first all corner nodes are listed and afterwards all edge nodes, one possible approach for *tri6* elements is the following:

$$\begin{bmatrix} \tilde{N}_1 \\ \tilde{N}_2 \\ \tilde{N}_3 \\ \tilde{N}_4 \\ \tilde{N}_5 \\ \tilde{N}_6 \end{bmatrix}_e^T = \begin{bmatrix} N_1 \\ N_2 \\ N_3 \\ N_4 \\ N_5 \\ N_6 \end{bmatrix}_e^T \cdot \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ \alpha & \alpha & 0 & 1-2\alpha & 0 & 0 \\ 0 & \alpha & \alpha & 0 & 1-2\alpha & 0 \\ \alpha & 0 & \alpha & 0 & 0 & 1-2\alpha \end{bmatrix}}_{:=\mathbf{T}_e}, \quad (3.70)$$

with the transformation matrix $\mathbf{T}_e \in \mathbb{R}^{6 \times 6}$ and the index e indicating a slave element. Computation of the inverse transformation matrix is straightforward, yielding

$$\mathbf{T}_e^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ -\frac{\alpha}{1-2\alpha} & -\frac{\alpha}{1-2\alpha} & 0 & \frac{1}{1-2\alpha} & 0 & 0 \\ 0 & -\frac{\alpha}{1-2\alpha} & -\frac{\alpha}{1-2\alpha} & 0 & \frac{1}{1-2\alpha} & 0 \\ -\frac{\alpha}{1-2\alpha} & 0 & -\frac{\alpha}{1-2\alpha} & 0 & 0 & \frac{1}{1-2\alpha} \end{bmatrix}. \quad (3.71)$$

Investigating (3.70) and (3.71) in more detail, it becomes obvious that the structure of the transformation matrix has been chosen as simple as possible. As the shape functions associated with the edge nodes of *tri6* elements are non-negative anyway, they can be beneficially used to guarantee $\int_e \tilde{N}_j^{(1)} de > 0$ also for the corner nodes. To simultaneously preserve $\int_e \tilde{N}_j^{(1)} de > 0$ for the edge nodes, it must be guaranteed that $\alpha < 1/2$. The proposed transformation is symmetric in the sense that edge nodes yield equal contributions to their two adjacent corner nodes and the modified shape functions satisfy partition of unity, i.e. $\sum_{j=1}^6 \tilde{N}_j^{(1)} = 1$. This choice is by no means based on restrictions, but it rather simplifies the biorthogonality construction later on. The *quad8* case can be derived in analogy, with the numbers of edge and corner nodes simply changing from three to four. The effect of the proposed basis transformation for both *tri6* and *quad8* elements is illustrated in Figure 3.11. For the sake of completeness, it is mentioned that no basis transformation is needed in the case of *quad9* surfaces, because the corresponding shape functions $N_j^{(1)}$ already satisfy (3.55), see e.g. Popp et al. [124].

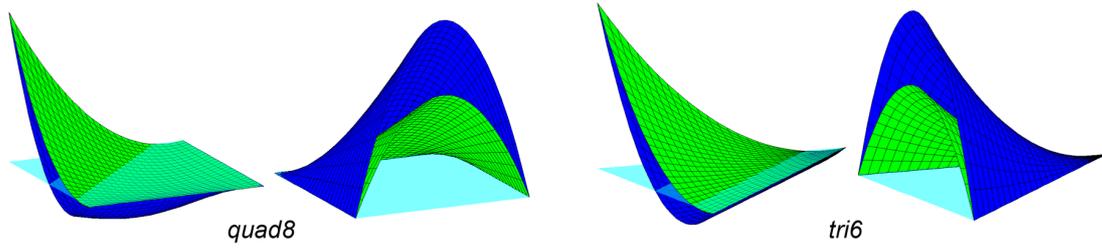


Figure 3.11: Displacement shape functions for a *quad8* element (left) and for a *tri6* element (right). In each case, the unmodified shape functions $N_j^{(1)}$ are illustrated in blue and the modified shape functions $\tilde{N}_j^{(1)}$ in green for both an exemplary corner node and an exemplary edge node.

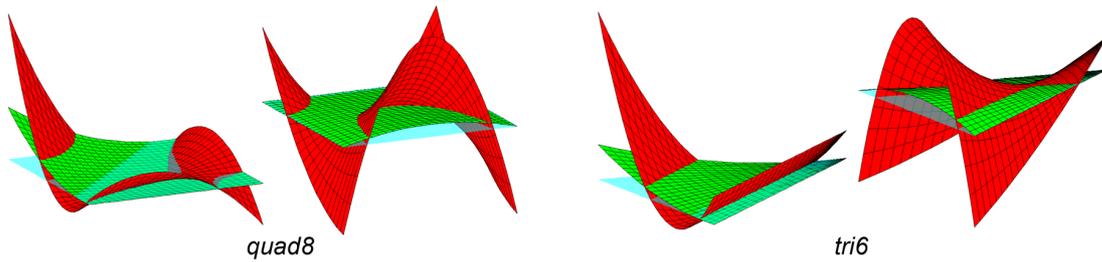


Figure 3.12: Displacement and Lagrange multiplier shape functions for a *quad8* element (left) and for a *tri6* element (right). In each case, the modified shape functions $\tilde{N}_j^{(1)}$ are illustrated in green and the dual shape functions Φ_j in red for both an exemplary corner node and an exemplary edge node.

Finally, the scalar α needs to be determined. Of course, α has to be chosen large enough to guarantee integral positivity of the modified displacement shape functions. It can easily be verified that in the case of undistorted finite elements, this requires $\alpha > 0$ for *tri6* elements and $\alpha > 1/8$ for *quad8* elements. A second criterion is elaborated in Lamichhane et al. [105], where the authors show that for undistorted finite elements with $\alpha = 1/12$ (*tri6* elements) and $\alpha = 1/5$ (*quad8* elements), respectively, it is even possible to have the piecewise linear polynomials included in the resulting dual Lagrange multiplier space \mathcal{M}_h . As mentioned before, the biorthogonality condition only guarantees that the constant polynomials are included in the dual Lagrange multiplier space ($r_\lambda = 0$). However, with the focus of this thesis later being extended to finite deformation contact, significant element distortions need to be taken into account and not only the special case of undistorted elements treated in Lamichhane et al. [105]. This makes it impossible to choose α such that the piecewise linear polynomials are always included in the resulting dual Lagrange multiplier space. Moreover, mixed meshes with both *tri6* and *quad8* surface elements should also be considered. Such mixed meshes require a common parameter α anyway, in order to assure continuity of the global displacement interpolation. Based on these considerations and on numerical experience, it is suggested to choose the scalar $\alpha = 1/5$, which guarantees integral positivity of the modified displacement shape functions for all element distortions occurring in practice (and $r_\lambda = 1$ for the special case of undistorted *quad8* facets).

Feasible dual Lagrange multiplier shape functions Φ_j are now readily obtained for all slave nodes ($m^{(1)} = n^{(1)}$) from the common biorthogonality condition applied locally on each slave

element to the basis transformed slave side displacement shape functions:

$$\int_e \Phi_j \tilde{N}_k^{(1)} de = \delta_{jk} \int_e \tilde{N}_k^{(1)} de, \quad j, k = 1, \dots, m_e^{(1)}, \quad (3.72)$$

where the number of Lagrange multiplier nodes is $m_e^{(1)} = 6$ for *tri6* elements and $m_e^{(1)} = 8$ for *quad8* elements. Figure 3.12 illustrates the new dual shape functions for both *tri6* and *quad8* elements. Furthermore, an alternative formulation of the slave displacement interpolation can be expressed as

$$\mathbf{u}_h^{(1)}|_{\Gamma_{c,h}^{(1)}} = \sum_{k=1}^{n^{(1)}} N_k^{(1)} \mathbf{d}_k^{(1)} = \sum_{k=1}^{n^{(1)}} \tilde{N}_k^{(1)} \tilde{\mathbf{d}}_k^{(1)}, \quad (3.73)$$

where the transformed nodal displacements $\tilde{\mathbf{d}}_k^{(1)}$ are simply given based on nodal connectivity information and the transformation matrix entries from (3.71) as

$$\begin{aligned} \tilde{\mathbf{d}}_k^{(1)} &= \mathbf{1} \mathbf{d}_k^{(1)} && \text{for corner nodes,} \\ \tilde{\mathbf{d}}_k^{(1)} &= \frac{1}{1-2\alpha} \mathbf{d}_k^{(1)} - \sum_{p=1}^{\text{adj}} \frac{\alpha}{1-2\alpha} \mathbf{d}_p^{(1)} && \text{for edge nodes,} \end{aligned} \quad (3.74)$$

with *adj* representing the number of adjacent slave nodes to slave node k . Node-wise assembly of (3.74) finally yields

$$\tilde{\mathbf{d}}_S = \mathbf{T}^{-1} \mathbf{d}_S, \quad (3.75)$$

where \mathbf{d}_S is the sub-vector of all nodal displacements on the slave surface as introduced in Section 3.2.1 and $\tilde{\mathbf{d}}_S$ is the corresponding vector with respect to the transformed basis. Moreover, $\mathbf{T} \in \mathbb{R}^{3n^{(1)} \times 3n^{(1)}}$ and $\mathbf{T}^{-1} \in \mathbb{R}^{3n^{(1)} \times 3n^{(1)}}$ are the global transformation matrix and its inverse, respectively. In terms of numerical efficiency, it is important to point out that the transformation matrices are sparse matrices that can be defined once at $t = 0$ and that stay unaltered during the entire finite deformation process afterwards. Besides, the matrix entries of \mathbf{T} and \mathbf{T}^{-1} can easily be computed from (3.74) and no actual matrix inversion needs to be carried out.

Owing to the basis transformation and taking into account that $m^{(1)} = n^{(1)}$, the mortar matrix \mathbf{D} is a square matrix and can be expressed as

$$\mathbf{D} = \tilde{\mathbf{D}} \mathbf{T}^{-1} \quad \text{with} \quad \tilde{D}_{jk} = \int_{\Gamma_{c,h}^{(1)}} \Phi_j \tilde{N}_k^{(1)} dA_0, \quad j, k = 1, \dots, m^{(1)}, \quad (3.76)$$

where $\tilde{\mathbf{D}}$ reduces to a diagonal matrix due to the biorthogonality relationship defined in (3.72). While the actual mortar matrix \mathbf{D} is no longer a diagonal matrix, it is important to stress that the proposed basis transformation nevertheless preserves all advantages of dual Lagrange multiplier interpolation. Due to the trivial inversion of both $\tilde{\mathbf{D}}$ and \mathbf{T} , the inverse of the mortar matrix \mathbf{D} is readily obtained as $\mathbf{D}^{-1} = \mathbf{T} \tilde{\mathbf{D}}^{-1}$. Thus, the major benefit of dual Lagrange multipliers, namely the possibility to condense all discrete Lagrange multiplier degrees of freedom from the global system of equations, is retained also for second-order interpolation in 3D. Algebraically, this can be observed by \mathbf{D}^{-1} still being a sparse matrix. Examining the locally quadratic, discontinuous

dual shape functions Φ_j defined in (3.72) in more detail, it can easily be verified that $p_\lambda = 2$ and $r_\lambda = 0$, i.e. only constant polynomials are fully contained.

Locally linear dual Lagrange multipliers

In the last paragraph, suitable dual Lagrange multipliers for second-order finite element interpolation have been derived based on a simple basis transformation procedure. Although the resulting dual shape functions are locally (bi-)quadratic, it is obvious that, in the general case of distorted elements, it is only possible to assure the constant polynomials to be contained in the global Lagrange multiplier interpolation. i.e. $r_\lambda = 0$. Thus, one may argue that reducing the total number of Lagrange multiplier degrees of freedom by using locally linear dual shape functions would make mortar coupling more efficient for quadratic finite elements, while at the same time the reproduction order r_λ and the convergence of the discretization error would be unaffected. Moreover, for the standard Lagrange multiplier case, the corresponding procedure has already been outlined in Section 3.3.1.4, see also Puso et al. [134].

This paragraph hence aims at deriving alternative dual Lagrange multipliers for second-order finite element interpolation in 3D, which are only based on locally *linear* shape functions. Only the corner nodes of quadratic finite elements, i.e. only three nodes for *tri6* elements and four nodes for *quad8* and *quad9* elements, carry discrete Lagrange multiplier degrees of freedom in this case ($m^{(1)} < n^{(1)}$). Using the same framework as in the last paragraph, the element-wise basis transformation for *tri6* elements is now defined as

$$\begin{bmatrix} \tilde{N}_1 \\ \tilde{N}_2 \\ \tilde{N}_3 \\ \tilde{N}_4 \\ \tilde{N}_5 \\ \tilde{N}_6 \end{bmatrix}_e^T = \begin{bmatrix} N_1 \\ N_2 \\ N_3 \\ N_4 \\ N_5 \\ N_6 \end{bmatrix}_e^T \cdot \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ \alpha & \alpha & 0 & 1 & 0 & 0 \\ 0 & \alpha & \alpha & 0 & 1 & 0 \\ \alpha & 0 & \alpha & 0 & 0 & 1 \end{bmatrix}}_{:=\mathbf{T}_e}, \quad (3.77)$$

with the transformation matrix $\mathbf{T}_e \in \mathbb{R}^{6 \times 6}$ and the index e indicating a slave element. Computation of the inverse is again straightforward, yielding

$$\mathbf{T}_e^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ -\alpha & -\alpha & 0 & 1 & 0 & 0 \\ 0 & -\alpha & -\alpha & 0 & 1 & 0 \\ -\alpha & 0 & -\alpha & 0 & 0 & 1 \end{bmatrix}. \quad (3.78)$$

The definition of the scalar α is easy this time. The simplest way of obtaining locally linear polynomials for the modified shape functions of the corner nodes is to set $\alpha = 1/2$, as this results in the well-known linear hat functions for $\tilde{N}_1^{(1)}$, $\tilde{N}_2^{(1)}$ and $\tilde{N}_3^{(1)}$. Unlike before, only these corner nodes will be used in the biorthogonality construction later on and carry dual Lagrange multipliers. The only fundamental difference of the transformation matrix as compared with the one introduced in (3.70) is the fact that partition of unity is now enforced for the corner nodes

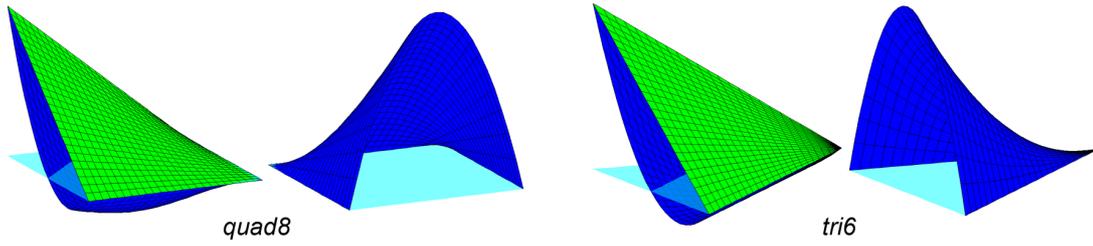


Figure 3.13: Displacement shape functions for a *quad8* element (left) and for a *tri6* element (right). In each case, the unmodified shape functions $N_j^{(1)}$ are illustrated in blue for both an exemplary corner node and an exemplary edge node. Modified shape functions $\tilde{N}_j^{(1)}$ only differ from the unmodified ones for corner nodes and are illustrated in green.

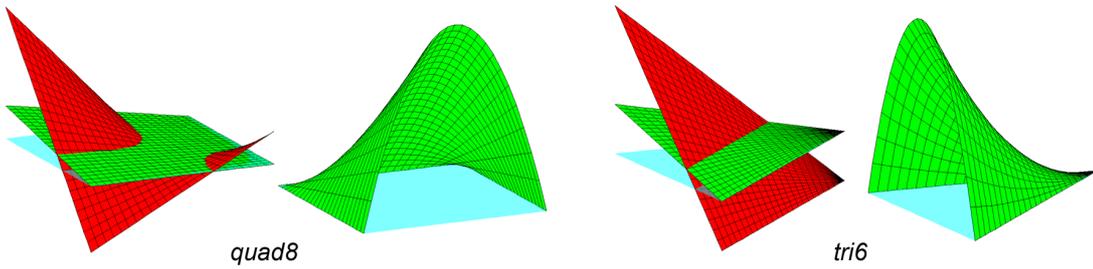


Figure 3.14: Displacement and Lagrange multiplier shape functions for a *quad8* element (left) and for a *tri6* element (right). In each case, the modified shape functions $\tilde{N}_j^{(1)}$ are illustrated in green for both an exemplary corner node and an exemplary edge node. Dual shape functions Φ_j only exist for corner nodes and are illustrated in red.

only, i.e. $\sum_{j=1}^3 \tilde{N}_j^{(1)} = 1$, but $\sum_{j=1}^6 \tilde{N}_j^{(1)} > 1$. Displacement shape functions of the edge nodes, on the other hand, remain unchanged in order to assure the invertibility of \mathbf{T}_e . Finally, it can be observed that the modified shape functions $\tilde{N}_j^{(1)}$ are simply the hierarchical basis functions. Again, the case of *quad8* elements can be derived in full analogy, with the numbers of edge and corner nodes simply changing from three to four. The effect of the second basis transformation for both *tri6* and *quad8* elements is illustrated in Figure 3.13. For the sake of completeness, it is pointed out that the approach is readily extendable to *quad9* elements, too.

Dual Lagrange multiplier shape functions Φ_j are now obtained from the modified slave displacement shape functions $\tilde{N}_j^{(1)}$ via the biorthogonality relationship (3.62). However, dual shape functions are *only* defined for the corner nodes, whereas edge nodes do not carry Lagrange multipliers. Local biorthogonality on each slave element then reads:

$$\int_e \Phi_j \tilde{N}_k^{(1)} de = \delta_{jk} \int_e \tilde{N}_k^{(1)} de, \quad j, k = 1, \dots, m_e^{(1)}, \quad (3.79)$$

where the number of Lagrange multiplier nodes is now only $m_e^{(1)} = 3$ for *tri6* elements and $m_e^{(1)} = 4$ for *quad8* elements. Exemplary dual shape functions resulting from (3.79) are illustrated in Figure 3.14 for both *tri6* and *quad8* elements. It can be seen that the dual shape functions are identical to the first-order case discussed in Section 3.3.2.2, which is not surprising in con-

sideration of the fact that both cases are based on simple (bi-)linear hat functions employed in the biorthogonality condition.

Similar as before, the transformed nodal displacements $\tilde{\mathbf{d}}_k^{(1)}$ are given based on nodal connectivity information and the transformation matrix entries from (3.78) as

$$\begin{aligned}\tilde{\mathbf{d}}_k^{(1)} &= \mathbf{1} \mathbf{d}_k^{(1)} && \text{for corner nodes,} \\ \tilde{\mathbf{d}}_k^{(1)} &= \mathbf{1} \mathbf{d}_k^{(1)} - \sum_{p=1}^{\text{adj}} \alpha \mathbf{d}_p^{(1)} && \text{for edge nodes,}\end{aligned}\quad (3.80)$$

with adj representing the number of adjacent slave nodes to slave node k . At the global level, the basis transformation of all slave displacements (including both corner nodes and edge nodes) can again be expressed as $\tilde{\mathbf{d}}_S = \mathbf{T}^{-1} \mathbf{d}_S$ and retains all the properties described before. Splitting all slave side displacement degrees of freedom into corner node displacements $\tilde{\mathbf{d}}_S^* \in \mathbb{R}^{3m^{(1)}}$ and edge node displacements $\tilde{\mathbf{d}}_S^\circ \in \mathbb{R}^{3n^{(1)}-3m^{(1)}}$ reveals the following substructure of the transformation matrix:

$$\tilde{\mathbf{d}}_S = \begin{bmatrix} \tilde{\mathbf{d}}_S^* \\ \tilde{\mathbf{d}}_S^\circ \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{A} & \mathbf{I} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{d}_S^* \\ \mathbf{d}_S^\circ \end{bmatrix} = \mathbf{T}^{-1} \mathbf{d}_S. \quad (3.81)$$

Therefore, the square matrix block \mathbf{D}^* associated with the slave corner nodes (i.e. the nodes actually carrying discrete Lagrange multipliers) can be expressed as

$$\begin{aligned}\mathbf{D}^* &= \tilde{\mathbf{D}}^* + \tilde{\mathbf{D}}^\circ \mathbf{A} && \text{with } \tilde{D}_{jk}^* = \int_{\Gamma_{c,h}^{(1)}} \Phi_j \tilde{N}_k^{(1)} dA_0, \quad j, k = 1, \dots, m^{(1)}, \\ \text{and } \tilde{D}_{jk}^\circ &= \int_{\Gamma_{c,h}^{(1)}} \Phi_j \tilde{N}_k^{(1)} dA_0, \quad j = 1, \dots, m^{(1)}, \quad k = m^{(1)} + 1, \dots, n^{(1)}.\end{aligned}\quad (3.82)$$

Whereas $\tilde{\mathbf{D}}^*$ reduces to a diagonal matrix owing to the biorthogonality relationship defined in (3.79), it can be observed that \mathbf{D}^* does not. Moreover, its inversion is also no longer trivial due to the coupling of corner nodes and edge node degrees of freedom represented by the matrix block \mathbf{A} of \mathbf{T}^{-1} . In order to nonetheless benefit from the major advantage of dual Lagrange multiplier interpolation, a different strategy is needed here. Remembering that $\tilde{\mathbf{D}}^*$ is diagonal, it becomes obvious that a trivial condensation of the discrete Lagrange multipliers is anyhow possible, namely when solving with respect to the basis transformed quantities. Thus, the opposite approach as before is followed by not shifting the mortar coupling terms to the unmodified basis \mathbf{d}_S , but rather shifting the internal and external force and stiffness terms to the modified basis $\tilde{\mathbf{d}}_S$. Again, this step is performed at negligible computational cost, because the basis transformation $\mathbf{d}_S = \mathbf{T} \tilde{\mathbf{d}}_S$ can be applied locally, i.e. at the element level before the global assembly procedure.

Finally, the resulting locally linear dual shape functions Φ_j , $j = 1, \dots, m^{(1)}$ have the same global properties as the locally quadratic ones, i.e. they are discontinuous and in general only the constant polynomials are included in the global Lagrange multiplier interpolation ($p_\lambda = 1$ and $r_\lambda = 0$). However, whereas for the first alternative all slave nodes also carry discrete Lagrange multipliers ($m^{(1)} = n^{(1)}$), the second alternative only allocates additional degrees of freedom to the slave corner nodes and thus works with a reduced number of constraints ($m^{(1)} < n^{(1)}$). Assuming a regular interface mesh, one readily obtains the estimates $m^{(1)} \approx n^{(1)}/4$ for *tri6* and *quad9* elements and $m^{(1)} \approx n^{(1)}/3$ for *quad8* elements.

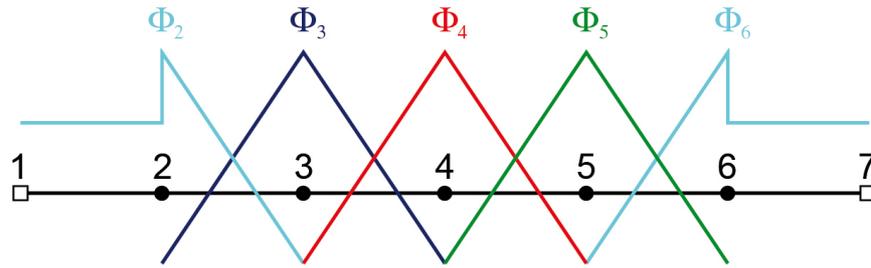


Figure 3.15: Dual shape functions Φ_j for 2D mortar coupling with modifications that allow for crosspoints or Dirichlet boundary conditions. The two end nodes 1 and 7 do not carry discrete Lagrange multipliers in this case.

3.3.3 Crosspoints and Dirichlet boundary conditions

Wherever a mortar mesh tying interface $\Gamma_{c,h}^{(1)}$ and a Dirichlet boundary $\Gamma_u^{(1)}$ meet on the slave side, a conflict of constraint enforcement will emerge. The reason is that the discrete Lagrange multipliers are supposed to enforce the weak displacement continuity constraint at such points, while at the same time the Dirichlet boundary condition already fixes the solution. From an engineering point of view, this situation can best be described as a problem of over-constraint. While strictly speaking already excluded in (2.45), it is nevertheless worthwhile having a closer look at such scenarios, especially when considering that an almost identical situation also arises at so-called crosspoints, where more than two subdomains meet. In that case, there are several sets of discrete Lagrange multipliers, which again lead to an over-constraint problem of the available degrees of freedom, and it should be mentioned that standard and dual Lagrange multipliers are affected in equal measure.

Special treatment of crosspoints and Dirichlet boundary conditions is well-established in mortar finite element methods, see e.g. Wohlmuth [176] and Puso and Laursen [131] for a comprehensive overview in both 2D and 3D. The basic idea is a slight modification of the Lagrange multiplier shape functions in the vicinity of such critical points (in two dimensions) or lines (in three dimensions). Exemplarily, only the simple 2D case with first-order dual shape functions is presented here. As Figure 3.15 illustrates, the Lagrange multiplier interpolation on elements adjacent to a crosspoint is modified such that the crosspoint itself does no longer carry a discrete Lagrange multiplier. Instead, the remaining shape functions must be adapted such that the interpolation remains consistent, e.g. still satisfying a partition of unity property. For first-order interpolation this simply yields a constant Lagrange multiplier shape function for the next-to-last node. Extensions to second-order interpolation and to 3D are conceptually straightforward, however especially the 3D case is more cumbersome due to different possible mesh topological configurations, see Wohlmuth [176] for all details.

In the context of Dirichlet boundary conditions, a rather simple and very common alternative in engineering practice is to apply the boundary condition *solely* to the master side nodes of the interface. The Dirichlet condition is then basically enforced weakly on the slave side, when considering the weak mesh tying constraint in (3.22). Exploiting model symmetries with respect to certain coordinate axes is another application of rather high practical relevance. In that case, it is possible to selectively remove mortar constraints and the associated Lagrange multiplier

components in certain axis directions and replace them with the Dirichlet (symmetry) conditions, see Puso and Laursen [131]. A representative numerical example involving crosspoints will be analyzed in Section 3.5.1.

3.4 Solution methods

After this overview of discrete Lagrange multiplier spaces, the attention is now turned back to the actual mortar finite element approach for tied contact derived in Section 3.2, and in particular to the final fully discretized expression given in (3.39) and (3.40). All solution methods for this system of $\text{ndof} + \text{nco}$ nonlinear discrete algebraic equations, where the global number of constraints is given by $\text{nco} = \text{ndim} \cdot m^{(1)}$, are based on a standard Newton–Raphson iteration as introduced in Section 2.3.5. With only equality constraints being present, no active set strategies are needed for mesh tying systems, but the iterative solution techniques can be applied directly, thus yielding standard (or smooth) Newton methods. Primal-dual active set strategies and the associated notion of semi-smooth Newton methods only become important in the context of unilateral contact considered in Chapter 4.

First, some basic notation and algebraic representations of the linearized quantities within each nonlinear solution step of the Newton–Raphson algorithm will be given. The main purpose then is to give an overview of the different ways of formulating the final linearized system of equations depending on the employed constraint enforcement approach. While the methods proposed in this thesis put a strong emphasis on dual Lagrange multipliers, the standard Lagrange multiplier approach will also be presented for comparison purposes and to be able to better classify different solution algorithms. For the same reason, a penalty treatment of the constraints and its Augmented Lagrange version based on Uzawa’s algorithm are briefly outlined, too.

3.4.1 Linearization and standard Newton algorithm

As explained in Section 2.3.5, the Newton–Raphson method (or Newton’s method) is based on a subsequent linearization of the residual, here defined by the discrete balance of linear momentum and the discrete mesh tying constraints in (3.39) and (3.40). Each nonlinear solution step (iteration index i) then consists of solving the resulting linearized system of equations and an incremental update of the unknown displacements \mathbf{d}_{n+1} and Lagrange multipliers $\lambda_{n+1-\alpha_f}$ until a user-defined convergence criterion is met. Taking into account that the discrete mesh tying operator \mathbf{B}_{mt} defined in (3.21) does not depend on the displacements, consistent linearization in iteration step i yields:

$$\mathbf{K}_{\text{effdyn}}(\mathbf{d}_{n+1}^i) \Delta \mathbf{d}_{n+1}^{i+1} + \mathbf{B}_{\text{mt}} \lambda_{n+1-\alpha_f}^i = -\mathbf{r}_{\text{effdyn}}(\mathbf{d}_{n+1}^i), \quad (3.83)$$

$$\left. \frac{\partial \mathbf{g}_{\text{mt}}(\mathbf{d}_{n+1})}{\partial \mathbf{d}_{n+1}} \right|_{\mathbf{d}_{n+1}^i} \Delta \mathbf{d}_{n+1}^{i+1} = -\mathbf{g}_{\text{mt}}(\mathbf{d}_{n+1}^i). \quad (3.84)$$

Herein, the fact that the Lagrange multipliers only enter the discrete mesh tying in a linear fashion has been made use of. Due to this linearity, it is possible to solve directly for the unknown Lagrange multipliers $\lambda_{n+1-\alpha_f}^i$ in each iteration step instead of an incremental formulation. Moreover, as mentioned in Section 2.3.5, all discrete force terms (inertia, damping, internal

and external forces) except for the additional mesh tying forces $\mathbf{f}_{\text{mt}}(\boldsymbol{\lambda}_{n+1-\alpha_f}^i)$ are summarized in the residual $\mathbf{r}_{\text{effdyn}}(\mathbf{d}_{n+1}^i)$ and the partial derivative of $\mathbf{r}_{\text{effdyn}}(\mathbf{d}_{n+1}^i)$ with respect to the displacements \mathbf{d} is commonly referred to as dynamic effective tangential stiffness matrix $\mathbf{K}_{\text{effdyn}}(\mathbf{d}_{n+1}^i)$, as introduced in (2.88). Finally, it is pointed out that the constraints $\mathbf{g}_{\text{mt}}(\mathbf{d}_{n+1}) = \mathbf{0}$ are already enforced at time $t = 0$ to assure angular momentum conservation (see also the nodal relocation procedure in Section 3.2.5). Thus, the right-hand side of the linearized constraint equation in (3.84) simply reduces to zero.

The linearized statement in (3.83) and (3.84) already gives a hint as to the typical saddle point structure of the resulting Lagrange multiplier system discussed in Section 2.4.1. This structure will be revealed even more clearly in the following paragraphs on algebraic notation. There, the final linearized system to be solved within each Newton step is presented in detailed matrix-vector format for the most common constraint enforcement strategies.

3.4.2 Standard Lagrange multiplier approach

Analyzing the linearized mesh tying system (3.83) in more detail and splitting the global displacement vector $\mathbf{d} = (\mathbf{d}_{\mathcal{N}}, \mathbf{d}_{\mathcal{M}}, \mathbf{d}_{\mathcal{S}})$ as well as all other involved quantities into three subsets as defined in Section 3.2.1 leads to the following notation in matrix-vector notation:

$$\begin{bmatrix} \mathbf{K}_{\mathcal{N}\mathcal{N}} & \mathbf{K}_{\mathcal{N}\mathcal{M}} & \mathbf{K}_{\mathcal{N}\mathcal{S}} & \mathbf{0} \\ \mathbf{K}_{\mathcal{M}\mathcal{N}} & \mathbf{K}_{\mathcal{M}\mathcal{M}} & \mathbf{0} & -\mathbf{M}^T \\ \mathbf{K}_{\mathcal{S}\mathcal{N}} & \mathbf{0} & \mathbf{K}_{\mathcal{S}\mathcal{S}} & \mathbf{D}^T \\ \mathbf{0} & -\mathbf{M} & \mathbf{D} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{d}_{n+1, \mathcal{N}} \\ \Delta \mathbf{d}_{n+1, \mathcal{M}} \\ \Delta \mathbf{d}_{n+1, \mathcal{S}} \\ \boldsymbol{\lambda}_{n+1-\alpha_f} \end{bmatrix} = - \begin{bmatrix} \mathbf{r}_{\mathcal{N}} \\ \mathbf{r}_{\mathcal{M}} \\ \mathbf{r}_{\mathcal{S}} \\ \mathbf{0} \end{bmatrix}. \quad (3.85)$$

Herein, the nonlinear iteration index i and the subscript \cdot_{effdyn} of the residual vector $\mathbf{r}_{\text{effdyn}}$ and the tangential stiffness matrix $\mathbf{K}_{\text{effdyn}}$ have been omitted for the ease of notation. Note that no matrix blocks $\mathbf{K}_{\mathcal{M}\mathcal{S}}$ and $\mathbf{K}_{\mathcal{S}\mathcal{M}}$ exist, because slave and master side degrees of freedom are only coupled via the mortar approach. Due to the inherent symmetry of $\mathbf{K}_{\text{effdyn}}$, the global linearized mesh tying system (3.85) is also *symmetric* and has the typical saddle point structure with a zero matrix block associated with the Lagrange multipliers $\boldsymbol{\lambda}_{n+1-\alpha_f}$ on the main diagonal. Thus, while a conforming discretization would yield a positive definite system, the coupled mesh tying system considered here becomes indefinite with both positive and negative eigenvalues due to the saddle point characteristics of the Lagrange multiplier method, see Section 2.4.1.

The linear system (3.85) needs to be solved within each nonlinear iteration step. Unfortunately, efficient iterative solution techniques and especially the associated preconditioners discussed in Section 2.3.6 usually perform very poorly for such indefinite systems or are not applicable at all. The main reason for this lies in the fact that common preconditioning techniques, e.g. the Jacobi and Gauss–Seidel methods, fail for zero diagonal matrix entries as occurring in (3.85). Nevertheless, there exist some specific solution methods for this type of saddle point matrix block system, which are both well-established and quite efficient. One popular representative, also employed as preconditioner in this thesis whenever large mesh tying and contact systems are considered with a standard Lagrange multiplier approach, is given by the so-called semi-implicit method for pressure-linked equations (SIMPLE) and its many descendants, see e.g. Elman et al. [37] for a very comprehensive overview in the context of the incompressible Navier–Stokes equations for fluid dynamics.

3.4.3 Dual Lagrange multiplier approach

As explained in Section 3.3, the dual Lagrange multiplier approach is characterized by its localization of the coupling constraints at the mesh tying interface, and thus algebraically by mortar matrix \mathbf{D} reducing to a diagonal matrix. This makes \mathbf{D} trivial to invert and allows for efficient condensation operations of the slave side degrees of freedom, i.e. both Lagrange multipliers and the discrete slave side displacements. The basis for this condensation is given by the saddle point system in (3.85), which is of course equally valid for dual Lagrange multiplier interpolation. In preparation of a first condensation step, the third row of (3.85) is used to express the unknown Lagrange multipliers $\lambda_{n+1-\alpha_f}$ as

$$\lambda_{n+1-\alpha_f} = \mathbf{D}^{-\top} (-\mathbf{r}_S - \mathbf{K}_{SN}\Delta\mathbf{d}_{n+1,N} - \mathbf{K}_{SS}\Delta\mathbf{d}_{n+1,S}). \quad (3.86)$$

Insertion into the second row of (3.85) yields the following intermediate system:

$$\begin{bmatrix} \mathbf{K}_{NN} & \mathbf{K}_{NM} & \mathbf{K}_{NS} \\ \mathbf{K}_{MN} + \mathbf{P}^\top \mathbf{K}_{SN} & \mathbf{K}_{MM} & \mathbf{P}^\top \mathbf{K}_{SS} \\ \mathbf{0} & -\mathbf{M} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{d}_{n+1,N} \\ \Delta\mathbf{d}_{n+1,M} \\ \Delta\mathbf{d}_{n+1,S} \end{bmatrix} = - \begin{bmatrix} \mathbf{r}_N \\ \mathbf{r}_M + \mathbf{P}^\top \mathbf{r}_S \\ \mathbf{0} \end{bmatrix}, \quad (3.87)$$

where the mortar projection operator $\mathbf{P} = \mathbf{D}^{-1}\mathbf{M}$ introduced in (3.50) is used to abbreviate the notation. As a second step, the constraint equation in the last row of (3.87) can alternatively be expressed as

$$\Delta\mathbf{d}_{n+1,S} = \mathbf{D}^{-1}\mathbf{M}\Delta\mathbf{d}_{n+1,M} = \mathbf{P}\Delta\mathbf{d}_{n+1,M}. \quad (3.88)$$

The final condensed system for the dual Lagrange multiplier approach is then obtained by reinserting this result into the first row and second row of the intermediate system, viz.

$$\begin{bmatrix} \mathbf{K}_{NN} & \mathbf{K}_{NM} + \mathbf{K}_{NS}\mathbf{P} \\ \mathbf{K}_{MN} + \mathbf{P}^\top \mathbf{K}_{SN} & \mathbf{K}_{MM} + \mathbf{P}^\top \mathbf{K}_{SS}\mathbf{P} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{d}_{n+1,N} \\ \Delta\mathbf{d}_{n+1,M} \end{bmatrix} = - \begin{bmatrix} \mathbf{r}_N \\ \mathbf{r}_M + \mathbf{P}^\top \mathbf{r}_S \end{bmatrix}. \quad (3.89)$$

This final linearized system unifies several beneficial properties as compared with the equivalent saddle point formulation given in (3.85). Firstly, the discrete Lagrange multiplier degrees of freedom $\lambda_{n+1-\alpha_f}$ have been removed from the global system and thus the commonly cited disadvantage of an increased system size for Lagrange multiplier methods is resolved. In fact, owing to the second condensation step, which removes the slave side displacement degrees of freedom $\Delta\mathbf{d}_{n+1,S}$, the final system size is even reduced as compared with a conforming discretization. Secondly, and more importantly, the typical saddle point structure with a zero diagonal matrix block has been completely removed on the way towards the final system (3.89), which is instead symmetric and positive definite again.

With regard to linear solvers, the dual Lagrange multiplier approach virtually allows for an “out-of-the-box” application of state-of-the-art iterative solution and preconditioning techniques, such as the CG or GMRES approach in combination with algebraic multigrid (AMG) methods. Simply speaking, all solvers that were optimized for conforming discretizations in nonlinear solid mechanics are equally applicable to the non-conforming mortar formulation with dual Lagrange multipliers in (3.89) due to similar system properties. The additional computational effort associated with the condensation operations can be considered very low. In a first, naive implementation, setting up the condensed system would simply require some additional matrix-matrix

products of interface-sized matrix blocks such as the discrete projection operator \mathbf{P} . However, a more elaborate implementation could even do without explicit matrix-matrix products, but would rather introduce modified local assembly procedures for the individual finite element contributions to the tangential stiffness matrix blocks \mathbf{K}_{NS} , \mathbf{K}_{SN} and \mathbf{K}_{SS} , taking into account the associated local entries of the mortar projection operator \mathbf{P} . In any case, the improved properties and the more efficient solvability of (3.89) as compared with (3.85) by far outweigh additional computational costs for the condensation, which makes the dual Lagrange multiplier approach the preferred choice throughout this thesis.

For the sake of completeness, two details should be pointed out. Firstly, the described condensation operations are of course also applicable for standard Lagrange multiplier interpolation with a non-diagonal mortar matrix \mathbf{D} , at least theoretically. In practice, however, the inverse matrix \mathbf{D}^{-1} would be densely populated in such a case, which forbids the actual computation and storage of \mathbf{D}^{-1} or likewise \mathbf{P} for moderate or even large system sizes. For dual Lagrange multiplier interpolation, on the contrary, inversion of \mathbf{D} and storage of the sparsely populated matrix \mathbf{P} remain easily manageable even for large-scale mortar mesh tying simulations. Secondly, node-matching interface meshes are contained as a special case in the given mortar formulation. Without going into all the details, this situation basically leads to \mathbf{P} becoming an identity operator, establishing a one-to-one mapping between slave side and master side displacements. Expression (3.89) then reduces to exactly the same linearized system that is obtained for a conforming finite element discretization.

3.4.4 Penalty approach

The penalty approach represents a very popular class of methods for constrained optimization problems, such as the mortar mesh tying situation considered here. In contrast to Lagrange multiplier methods, the penalty approach does not generate a mixed formulation with additional degrees of freedom, but the formulation instead remains purely displacement-based. However, as explained in Section 2.4.2, penalty methods also suffer from severe drawbacks, including an inevitable violation of the occurring interface conditions and a well-known sensitivity with respect to the penalty parameter ϵ . Thus, the penalty approach is often combined with an Augmented Lagrange strategy based on Uzawa's algorithm in practice, which leads to a nested iterative solution strategy with the inner loop solving the nonlinear solid mechanics problem and the outer loop updating the Lagrange multipliers.

For comparison purposes, an algebraic representation of the final linearized mesh tying system using such penalty-based methods is presented in the following. Starting point for all derivations is a relaxation of the constraint $\mathbf{g}_{\text{mt}}(\mathbf{d}_{n+1}) = \mathbf{0}$, which is typically replaced by a linear relationship between interface Lagrange multipliers $\boldsymbol{\lambda}$ and constraint violation \mathbf{g}_{mt} in the form

$$\boldsymbol{\lambda} = \epsilon \mathbf{g}_{\text{mt}}(\mathbf{d}_{n+1}). \quad (3.90)$$

Consequently, the Lagrange multipliers are no longer independent variables, but the discrete mesh tying forces \mathbf{f}_{mt} in (3.39) can rather be formulated as a function of the unknown displacements \mathbf{d}_{n+1} , viz.

$$\mathbf{f}_{\text{mt}}(\mathbf{d}_{n+1}) = \epsilon \mathbf{B}_{\text{mt}}^{\text{T}} \mathbf{B}_{\text{mt}} \mathbf{d}_{n+1} = \epsilon \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{M}^{\text{T}} \mathbf{M} & -\mathbf{M}^{\text{T}} \mathbf{D} \\ \mathbf{0} & -\mathbf{D}^{\text{T}} \mathbf{M} & \mathbf{D}^{\text{T}} \mathbf{D} \end{bmatrix} \mathbf{d}_{n+1}. \quad (3.91)$$

This yields the following linearized penalty system to be solved within each Newton step:

$$\begin{bmatrix} \mathbf{K}_{\mathcal{NN}} & \mathbf{K}_{\mathcal{NM}} & \mathbf{K}_{\mathcal{NS}} \\ \mathbf{K}_{\mathcal{MN}} & \mathbf{K}_{\mathcal{MM}} + \mathbf{M}^\top \mathbf{M} & -\mathbf{M}^\top \mathbf{D} \\ \mathbf{K}_{\mathcal{SN}} & -\mathbf{D}^\top \mathbf{M} & \mathbf{K}_{\mathcal{SS}} + \mathbf{D}^\top \mathbf{D} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{d}_{n+1, \mathcal{N}} \\ \Delta \mathbf{d}_{n+1, \mathcal{M}} \\ \Delta \mathbf{d}_{n+1, \mathcal{S}} \end{bmatrix} = - \begin{bmatrix} \mathbf{r}_{\mathcal{N}} \\ \mathbf{r}_{\mathcal{M}} \\ \mathbf{r}_{\mathcal{S}} \end{bmatrix}, \quad (3.92)$$

which obviously has a very simple structure without saddle point characteristics or the need for a condensation procedure. On the other hand, the performance of state-of-the-art iterative linear solution and preconditioning techniques applied to (3.92) will strongly depend on the penalty parameter ϵ . Large penalty parameters are desirable to reduce the inevitable violation of the mesh tying constraint $\mathbf{g}_{\text{mt}}(\mathbf{d}_{n+1}) = \mathbf{0}$, but drastically increase the condition number of the system matrix in (3.92), and thus hamper the applicability of iterative linear solvers. At the same time, the solution accuracy obtained with small or moderate penalty parameters might not be sufficient for many contact scenarios.

For this reason, the pure penalty approach is usually only used as a kernel for an Augmented Lagrange strategy based on Uzawa's algorithm. As outlined in Section 2.4.3, an additional outer loop (iteration index k) is introduced based on the following update of the Lagrange multipliers:

$$\boldsymbol{\lambda}^{(k+1)} = \boldsymbol{\lambda}^{(k)} + \epsilon \mathbf{g}_{\text{mt}}(\mathbf{d}_{n+1}^{(k)}). \quad (3.93)$$

Usually, the procedure starts with an initial guess $\boldsymbol{\lambda}^{(0)} = \mathbf{0}$, making the first Uzawa step identical to the penalty method. The outer iteration on the Lagrange multiplier $\boldsymbol{\lambda}^{(k)}$ is then repeated until a user-defined convergence criterion is met. The Uzawa algorithm as described above is algorithmically very simple and commonly allows for a beneficial reduction of the penalty parameter as compared with pure penalty methods, see Laursen [107]. However, convergence is often quite slow so that heuristic acceleration strategies are needed, e.g. based on an update of the penalty parameter during augmentation or linear regression, see Zavarise and De Lorenzis [191].

For the sake of completeness, it should be mentioned that an alternative interpretation of the Augmented Lagrange approach is also possible. As explained in Section 2.4.3, the Lagrange multipliers can be kept as unknowns and the resulting Augmented Lagrange approach must then be understood as a Lagrange multiplier method to which penalty contributions are added in order to improve the numerical properties of the formulation. Basically, the resulting linearized mesh tying system for this case is a mere combination of (3.85) and (3.92), and thus will not be considered in detail here.

3.5 Numerical examples

Three numerical examples are presented and analyzed in order to evaluate the numerical properties of the proposed mortar finite element methods for mesh tying applications in nonlinear solid mechanics. If not stated otherwise, the dual Lagrange multiplier version based on the condensed linearized system (3.89) is employed in combination with first-order finite element interpolation. Moreover, the focus is almost exclusively on 3D examples, due to the much higher complexity of mortar interface coupling and numerical integration in that case.

The given examples are basically supposed to validate some fundamental characteristics of the mortar approach, such as consistency for non-matching meshes (see patch tests in Section 3.5.1),

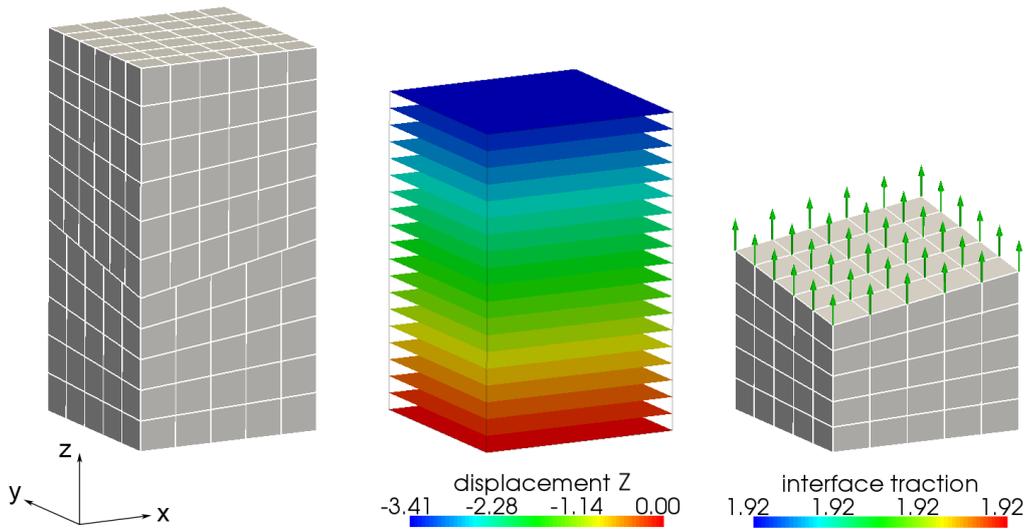


Figure 3.16: 3D patch test with inclined interface – finite element mesh (left), displacement u_z (middle) and interface tractions represented by the discrete Lagrange multipliers λ (right).

optimal convergence rates of the discretization error (see mesh refinement tests in Section 3.5.2) as well as suitability for finite deformation scenarios and the satisfaction of mechanical conservation laws (see L-shaped block example in Section 3.5.3). A special emphasis is also set on the analysis of the novel discrete dual Lagrange multiplier spaces for second-order finite element interpolation in 3D as introduced in Section 3.3.2.4.

3.5.1 Consistency – Patch tests

Patch tests are arguably one of the most common validation tools in finite element analysis, typically used as a first important step towards an assessment of the consistency of new element formulations, see e.g. Irons [89] and Taylor et al. [157]. In the present context of mesh tying and contact mechanics, patch tests are investigated in order to analyze the ability of mortar methods to exactly represent the simplest possible (i.e. constant) stress states across arbitrary non-conforming interfaces. However, it is well-known that collocation-based methods such as the classical node-to-segment (NTS) approach for mesh tying and unilateral contact typically fail the patch test, as will be discussed in more detail in Section 4.7.1. Mortar finite element methods, with their variationally consistent interpolation of the interface traction via discrete Lagrange multipliers λ , guarantee the exact satisfaction of typical patch tests by design.

As a first test setup, two stacked cubes with an inclined but flat mesh tying interface, as illustrated in Figure 3.16, are investigated. This geometric model is obtained by first considering two identical cubes of side length 10 and then moving two opposite corners of the interface by a distance of ± 2 in z -direction. The compressible Neo–Hookean material law introduced in Section 2.1.2 is employed with Young’s modulus $E = 10$ and Poisson’s ratio $\nu = 0.4$. A constant pressure load $p = -0.2$ is applied to the top surface of the upper block, and the bottom surface of the lower block is supported such that any rigid body movement is precluded, but the bodies are free to expand laterally. The lower block is defined as slave side for mortar coupling and the

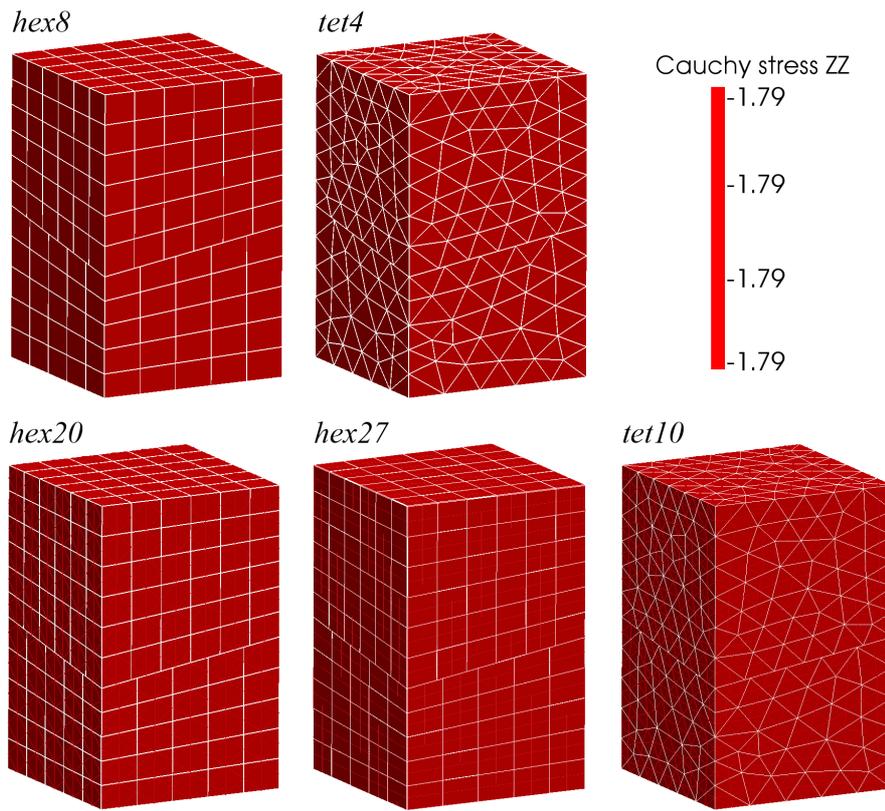


Figure 3.17: 3D patch test with inclined interface – Cauchy stress σ_{zz} for several different types of first-order and second-order mortar finite element interpolation.

chosen mesh size ratio of $h^{(1)}/h^{(2)} = 5/6$ generates a non-matching situation at the interface. Figure 3.16 exemplarily illustrates the displacement solution as well as the Lagrange multiplier (i.e. interface traction) solution in z -direction for a *hex8* discretization. As expected, a linear displacement field and constant interface tractions are obtained. The fact that the patch test is actually passed to machine precision for any first-order or second-order finite element type is emphasized in Figure 3.17, where the normal stress component in z -direction of the Cauchy stress tensor σ is visualized. While all presented results have been obtained with dual Lagrange multiplier interpolation according to Section 3.3.2, standard Lagrange multipliers according to Section 3.3.1 yield identical results.

The second patch test investigated here is a 2D rectangular strip (length $l = 8$, width $w = 3$) with five subdomains, each discretized with different first-order and second-order finite elements (i.e. *tri3*, *quad4*, *tri6*, *quad8* and *quad9* elements), see Figure 3.18. While this admittedly constitutes a rather academic example, it strikingly demonstrates the mesh generation flexibility offered by mortar methods, and especially also the possibility of a consistent treatment of cross-points as discussed in Section 3.3.3. Again, a compressible Neo–Hookean constitutive model is employed ($E = 10$, $\nu = 0.3$) and the strip is subject to unilateral loading in y -direction. Both displacement and stress solution confirm that this 2D patch test is passed to machine precision. As mentioned in Section 3.3.3, the treatment of crosspoints is readily extended to three dimensions, see e.g. Wohlmuth [176].

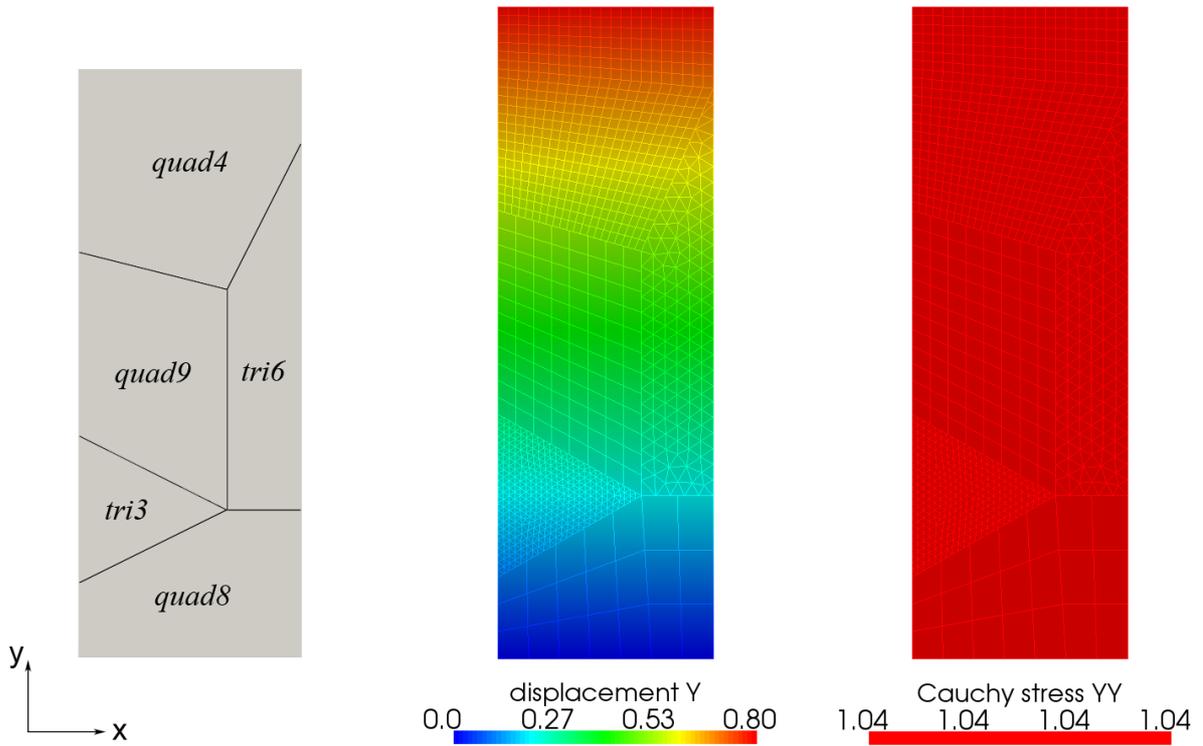


Figure 3.18: 2D patch test with crosspoints – types of finite element interpolation in the individual subdomains (left), displacement u_y (middle) and Cauchy stress σ_{yy} (right).

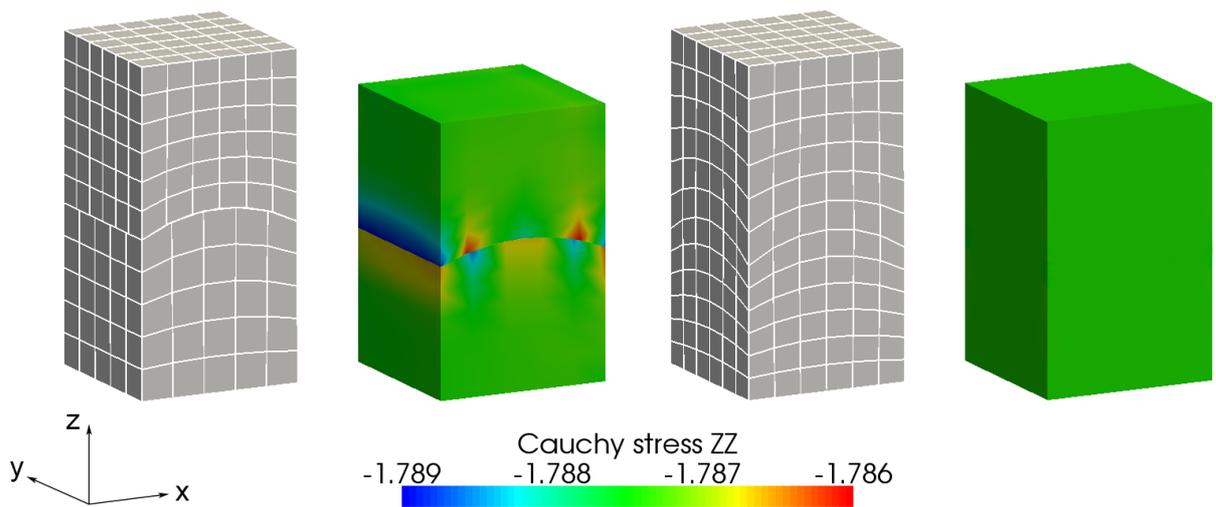


Figure 3.19: 3D patch test with curved interface – finite element mesh and Cauchy stress σ_{zz} for non-conforming interfaces (left) and for node-matching interfaces (right).

Finally, the first patch test model is reconsidered, but now with a curved mesh tying interface. The exemplary results for a *hex8* mesh in the left part of Figure 3.19 illustrate the limits of mortar finite element methods with regard to exact patch test satisfaction. It can be seen quite clearly that the patch test is not satisfied to machine precision in that case, but instead a small error is introduced in the vicinity of the interface. The reason for this result has already been explained in Section 3.2.1 and lies in the fact that the discrete surfaces $\Gamma_{c,h}^{(1)}$ and $\Gamma_{c,h}^{(2)}$ are no longer geometrically coincident for non-matching meshes on curved interfaces, but tiny gaps and overlapping regions appear. Thus, a discrete projection step is needed, which inevitably precludes the constant stress solution to be recovered exactly. This becomes even clearer when analyzing a curved mesh tying interface with node-matching meshes, as visualized in the right part of Figure 3.19. In that case, the discrete mesh tying surfaces $\Gamma_{c,h}^{(1)}$ and $\Gamma_{c,h}^{(2)}$ are again coincident, the mortar projection operator \mathbf{P} reduces to an identity mapping and the patch test is satisfied exactly. Nevertheless, it should be pointed out that the error of mortar methods in curved patch tests is only marginal and can factually be neglected from an engineering point of view. Besides, the curved patch test behavior of mortar methods is still significantly better than that of classical NTS schemes, see also Hesch and Betsch [72].

3.5.2 Spatial convergence – Bending structure

The following numerical example is supposed to verify the optimality of the proposed mortar finite element methods. Most importantly, no deterioration of the convergence rate of the discretization error measured in the appropriate norms is permitted as compared with a standard, conforming discretization. As outlined in Sections 2.3.4 and 3.3, this means that for finite elements of polynomial order p and a characteristic mesh size h , convergence rates $\mathcal{O}(h^p)$ can be expected in the energy norm, which will be the exemplarily analyzed norm here. The fully nonlinear problem formulation given in Section 3.1 is simplified for this example by assuming small deformations. Kinematics are then defined via the linearized strain tensor $\epsilon = 1/2 (\nabla \mathbf{u} + (\nabla \mathbf{u})^\top)$. Material behavior is assumed to be linear-elastic based on Hooke's law $\sigma = \mathbb{C} : \epsilon$, which relates the Cauchy stress tensor σ and the linearized strain tensor ϵ via a constant fourth-order constitutive tensor \mathbb{C} . The components of \mathbb{C} are given as

$$C_{ijkl} = \frac{E\nu}{(1+\nu)(1-2\nu)} (\delta_{ij}\delta_{kl}) + \frac{E}{2(1+\nu)} (\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}), \quad (3.94)$$

with Young's modulus E and Poisson's ratio ν .

In the following, the simple bending problem of a cuboid structure (see left part of Figure 3.20) with dimensions $l_x \times l_y \times l_z$ is analyzed, which is supported such that all rigid body modes are removed. The whole setup is symmetric with respect to the xy - and yz -planes and Dirichlet boundary conditions are given as

$$\begin{aligned} u_x(0, 0, 0) &= u_y(0, 0, 0) = u_z(0, 0, 0) = 0, \\ u_x(0, \frac{l_y}{2}, 0) &= u_z(0, \frac{l_y}{2}, 0) = u_z(\frac{l_x}{2}, 0, 0) = 0. \end{aligned} \quad (3.95)$$

Pure bending around the z -axis is readily obtained by applying distributed loads $f_x = \pm 2q y/l_y$ to the two surfaces $x = \pm l_x/2$. The analytical solution for this 3D bending problem of linear

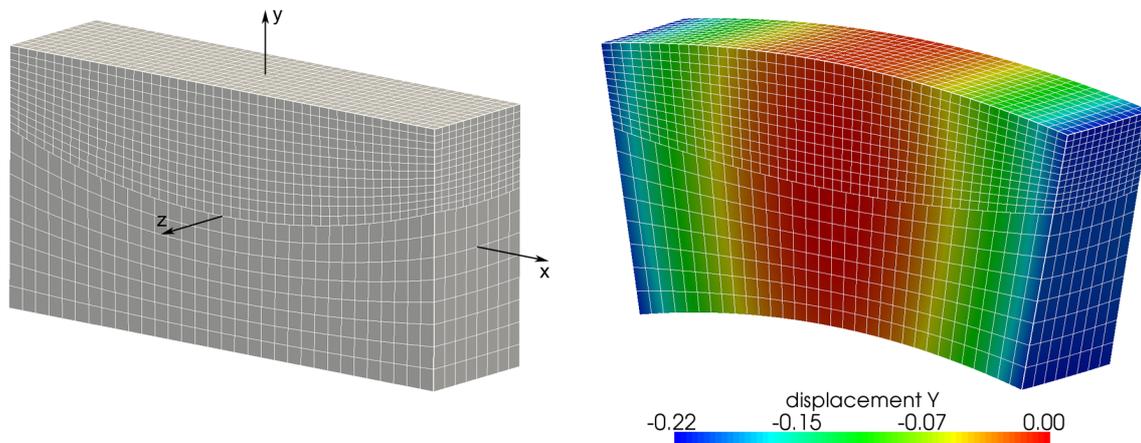


Figure 3.20: 3D bending structure – model setup with curved mortar interface and exemplary finite element mesh (left) and numerical solution for the displacement u_y (right).

elasticity is well-known, see Timoshenko and Goodier [161], and can be expressed as

$$u_x = \frac{2q}{El_y} xy, \quad u_y = \frac{q}{El_y} (-x^2 - \nu y^2 + \nu z^2), \quad u_z = -\frac{2q\nu}{El_y} yz. \quad (3.96)$$

For the numerical simulations, geometry, material and loading parameters have been chosen as $l_x = 4$, $l_y = 2$, $l_z = 1$, $E = 1000$, $\nu = 0.3$ and $q = 100$. The deformed configuration for an exemplary finite element mesh is also illustrated in Figure 3.20. The mortar interface, which has intentionally been given a curved shape to make the problem more general, cuts the structure into two non-matching parts. Both subdomains are discretized with hexahedral and tetrahedral finite elements of polynomial degree p , where either $p = 1$ (first-order elements) or $p = 2$ (second-order elements). Both standard and dual shape functions are used for the discrete Lagrange multipliers and a focus is set on the different types of dual Lagrange multipliers for second-order interpolation introduced in Section 3.3.2.4. Uniform mesh refinement is applied with the element size ratio of slave and master side being fixed at $h^{(1)}/h^{(2)} = 2/3$.

The results for first-order finite elements ($p = 1$) in Figure 3.21 demonstrate that all considered cases converge asymptotically with the optimal order that can be expected in the given example, i.e. $\mathcal{O}(h)$. With regard to dual Lagrange multipliers, it should be emphasized that this result is obtained despite a reduced reproduction order $r_\lambda = p - 1 = 0$ due to biorthogonality in that case. While not shown here, spatial convergence is not affected by the choice of slave and master side, but is preserved over a wide range of element size ratios $h^{(1)}/h^{(2)}$.

Analyzing the second-order cases ($p = 2$) in more detail, it can be seen that both linear (denoted by the ending “lin” in Figure 3.21) and quadratic (denoted by the ending “quad” in Figure 3.21) *standard* Lagrange multiplier interpolation according to Section 3.3.1.4 asymptotically yield an optimal $\mathcal{O}(h^2)$ convergence. Again, this result emphasizes that a reduced reproduction order, here $r_\lambda = p - 1 = 1$, is sufficient to achieve the expected convergence rates. Finally, the two corresponding versions of linear and quadratic *dual* Lagrange multiplier interpolation for *hex20*, *hex27* and *tet10* meshes according to Section 3.3.2.4 are investigated, too, with both cases only providing a further reduced reproduction order $r_\lambda = p - 2 = 0$. While the proposed

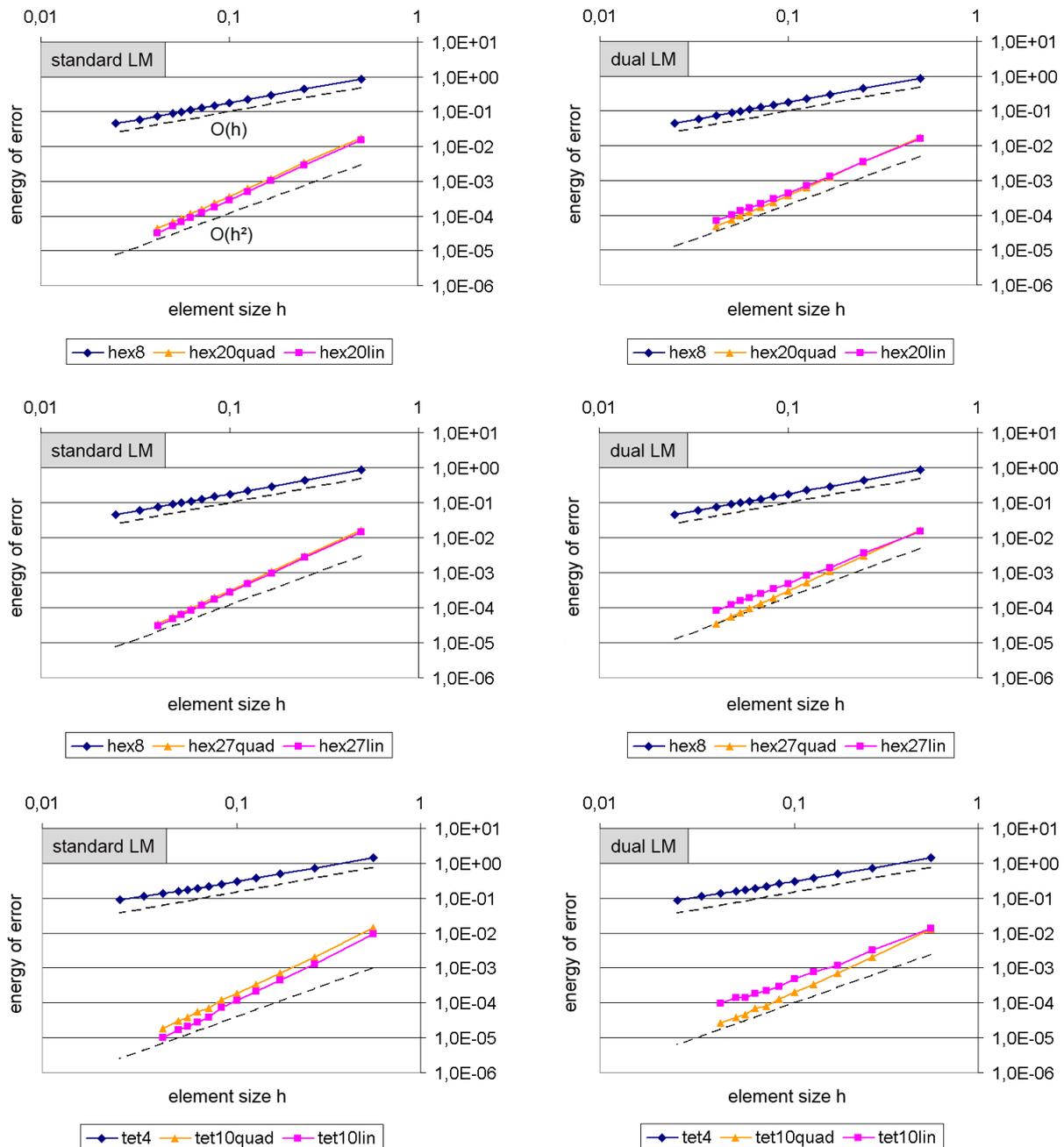


Figure 3.21: 3D bending structure – convergence of the discretization error measured in the energy norm with uniform mesh refinement. First-order and second-order hexahedral meshes (top and middle) as well as tetrahedral meshes (bottom) are considered using both standard Lagrange multiplier interpolation (left) and dual Lagrange multiplier interpolation (right).

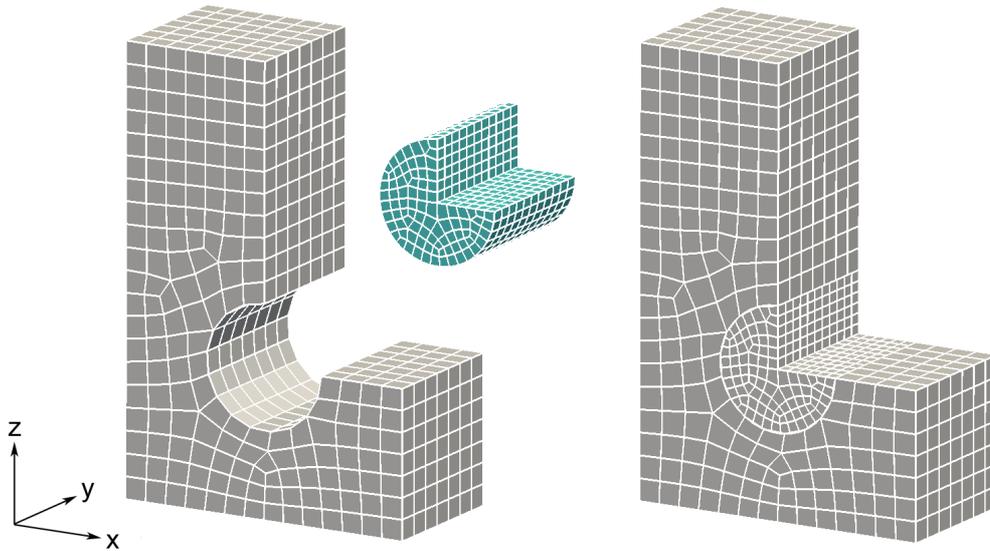


Figure 3.22: L-shaped block – model generation (left) and final setup with curved mortar mesh tying interface and exemplary finite element mesh (right).

locally quadratic dual Lagrange multiplier interpolation still yields optimal results, the behavior for the alternative locally linear dual Lagrange multipliers shows some marginal deterioration, especially in combination with the *tet10* elements. The reason for this slightly suboptimal behavior is not directly apparent and may be a topic of further investigation.

All in all, however, a deterioration by $\mathcal{O}(h^{1/2})$ for second-order mortar finite elements, as predicted by convergence theory if the global Lagrange multiplier interpolation is only of reproduction order $r_\lambda = p - 2$, cannot be observed in the numerical results. This is in accordance with the results in Seshaiyer and Suri [147], where different mortar methods have been analyzed for a Laplace operator model problem. The interested reader is also referred to Puso et al. [134] for a very similar numerical example in the context of standard Lagrange multipliers only. Other error measures, e.g. the L^2 -norm of the discretization error, give equally conclusive results. Of course, the convergence results obtained in this example are not directly transferable to unilateral contact problems and friction, because the regularity requirements for establishing similar a priori error estimates do not necessarily hold for such cases, see e.g. Wohlmuth et al. [179]. Nevertheless, the given validation of standard and dual Lagrange multipliers in the mesh tying setting at least represents a strong indicator for their likewise applicability to contact analysis.

3.5.3 Conservation properties – L-shaped block

The final numerical example investigates the proposed dual mortar mesh tying algorithms in the most general context of transient solid dynamics with finite deformations and nonlinear material behavior. As illustrated in Figure 3.22, the model consists of an L-shaped block, whose larger part has the dimensions $1.2 \times 1.2 \times 3.6$, while the smaller part is simply a cube with side length 1.2. Constitutive behavior is modeled according to a compressible Neo–Hookean law ($E = 10,000$, $\nu = 0.4$), and the density is set to $\rho_0 = 100$. The left and right surfaces of the L-shaped block are both subject to a pressure load $p(t) = 2000 \cdot \sin(2\pi t)$ in negative surface

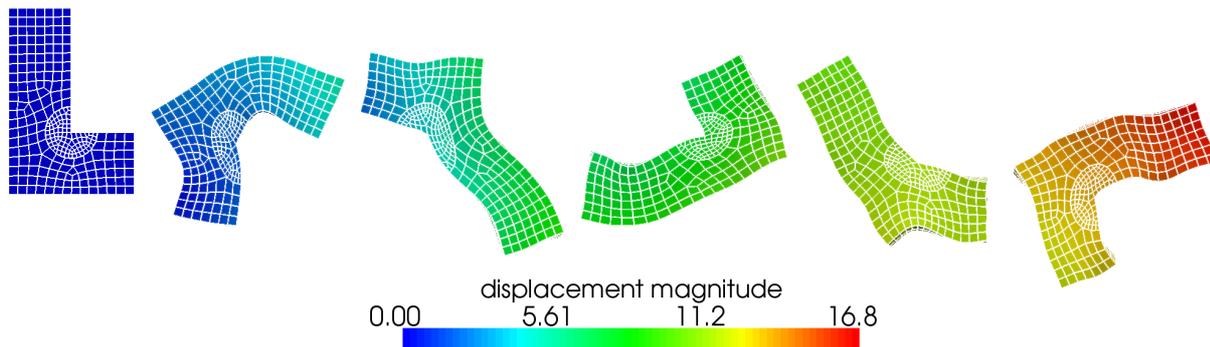


Figure 3.23: L-shaped block – characteristic stages of deformation at $t = 0, 1, 2, 3, 4, 5$ and numerical solution for the displacement magnitude $\|\mathbf{u}\|$.

normal direction in the time interval $t \in [0, 0.5]$, while no external forces are acting on the body in the time interval $t \in [0.5, 5]$. This adds up to a total of 500 time steps with the constant time step size $\Delta t = 0.01$. A curved non-matching mortar interface is introduced to make the mortar setting as general as possible, see Figure 3.20, with the outer surface of the cylindrical inclusion being chosen as slave side. The interested reader is also referred to Hesch and Betsch [72] for a similar investigation of mechanical conservation properties with a planar interface.

In order to assure exact algorithmic conservation of linear and angular momentum as well as mechanical energy, the EMM initially proposed by Simo and Tarnow [150] is employed here. A consistent treatment of arbitrary hyperelastic material models other than the simple St.-Venant–Kirchhoff model within the EMM framework has been derived in Gonzalez [57]. Some characteristic stages of deformation are visualized in Figure 3.21 and emphasize the strong non-linearities involved in this numerical simulation.

However, the main focus of interest for the presented example lies in the mechanical conservation properties. As can be seen from Figures 3.24, 3.25 and 3.26, linear and angular momentum as well as mechanical energies are *exactly* conserved when combining the proposed dual mortar finite element discretization and time integration based on the EMM. As explained in Section 3.2.5, linear momentum conservation is assured by using the same numerical integration procedure for both mortar matrices \mathbf{D} and \mathbf{M} , while the mesh initialization procedure suggested in Puso [129] and the EMM together guarantee angular momentum conservation. Finally, energy conservation is a direct consequence of the employed time integration scheme and could not be achieved when using a standard generalized- α method instead.

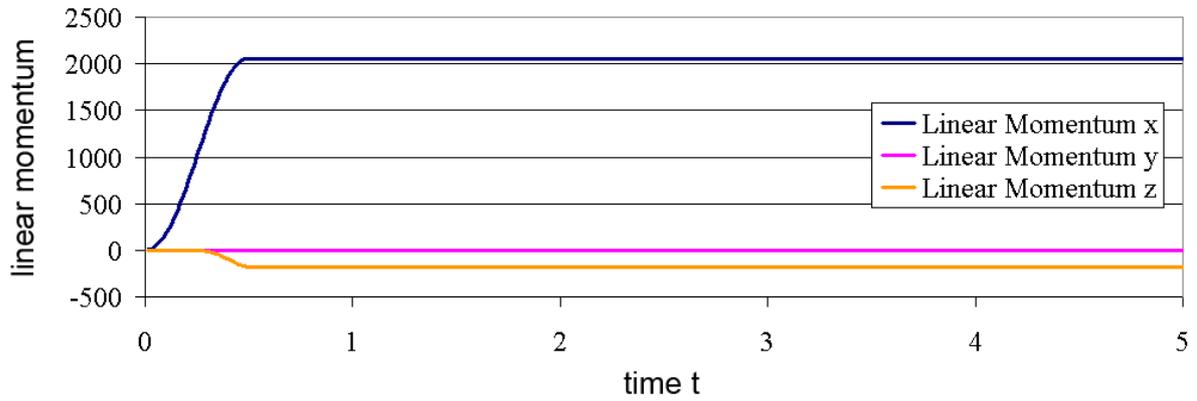


Figure 3.24: L-shaped block – conservation of linear momentum.

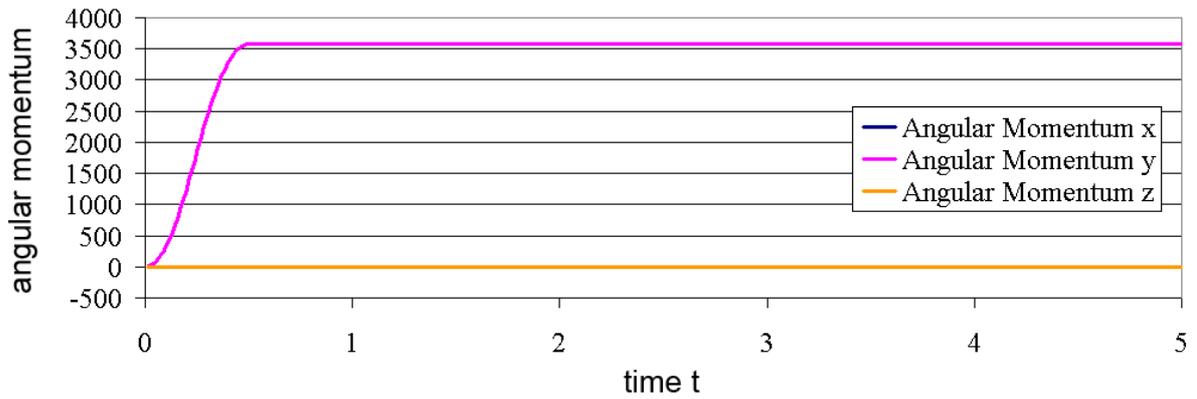


Figure 3.25: L-shaped block – conservation of angular momentum.

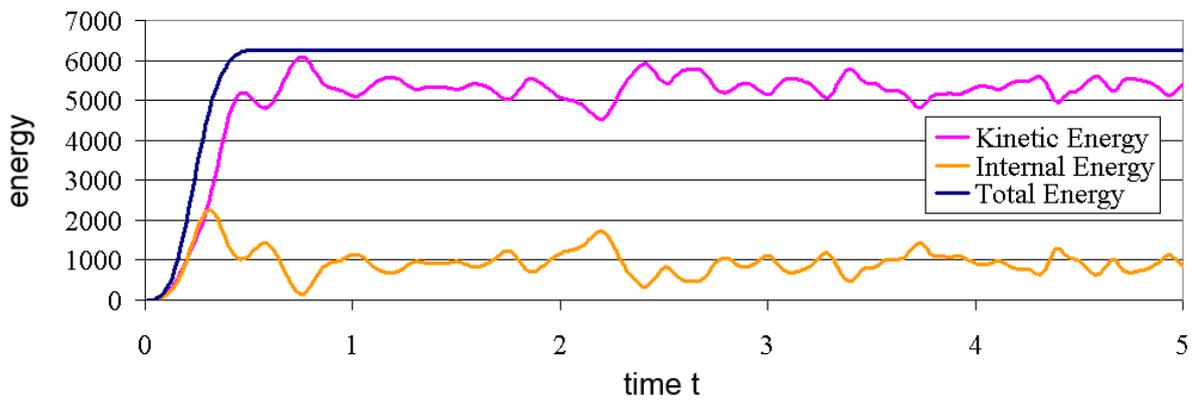


Figure 3.26: L-shaped block – conservation of mechanical energies.

4 Mortar Methods for Finite Deformation Contact Mechanics

Contact interaction in nonlinear solid mechanics and the use of mortar finite element methods in this context are the main focus of interest of this thesis. The goal of all developments presented is to be able to analyze and accurately predict the mechanical response in highly nonlinear unilateral contact scenarios, i.e. including very large deformations and sliding, continuous changes of the active contact area and possibly nonlinear material behavior. From a method development point of view, many aspects of mortar methods already introduced for mesh tying in Chapter 3 can either be re-used directly or in a slightly modified way in order to meet contact-specific demands. This involves the general mortar finite element concept, the 2D and 3D integration algorithms for mortar matrices \mathbf{D} and \mathbf{M} as well as the proposed discrete Lagrange multiplier spaces. For further reference concerning the explanations in this chapter, the reader may also want to consult the contributions in Gitterle et al. [55], Hartmann et al. [66], Hübner and Wohlmuth [80] and Popp et al. [123, 124, 128].

Again, the starting point is a thorough problem description of finite deformation contact mechanics in strong and weak formulation. After that, all necessary extensions of the mortar finite element discretization and numerical integration as compared with the simple mesh tying case will be presented, with a focus on consistent linearization. Similarly, special requirements with regard to the discrete Lagrange multiplier spaces (especially their dual versions) will be outlined, and new contributions, such as the consistent definition and linearization of dual shape functions in the context of finite deformation contact, will be derived. Another key aspect and additional complexity lies in the definition of efficient active set strategies for identifying the active contact constraints. Semi-smooth Newton schemes are the method of choice in this thesis, and will be applied to fully nonlinear mortar-based contact algorithms for the first time. As for tied contact, an overview of solution methods is given, and then some important aspects of efficient implementation and parallel computing are sketched. Finally, selected numerical examples demonstrate fundamental properties of the resulting contact algorithms, such as consistency and optimal convergence, as well as their applicability to challenging large-scale simulations.

4.1 Unilateral contact problem statement

The finite deformation contact problem statement presented in the following is based on the IBVP of nonlinear solid mechanics (see Section 2.1.4) in combination with the frictionless and frictional contact constraints (see Sections 2.2.3 and 2.2.4). After reviewing some basic notation and the strong problem formulation, a weak formulation of the two body contact problem with frictionless and frictional sliding is derived in full detail. In contrast to the mesh tying case considered so far, unilateral contact leads to a constrained minimization problem with *inequality*

constraints, or more generally to so-called variational inequalities. It should be mentioned that both frictionless and frictional contact can either be formulated as variational inequalities with a constrained solution or as saddle point problems based on Lagrange multipliers, where the focus will be on the latter approach here. For further theoretical considerations and an in-depth analysis of the mathematical foundations of contact mechanics, the comprehensive textbook by Kikuchi and Oden [94] and the recent review article by Wohlmuth [177] should be consulted.

4.1.1 Strong formulation

For the sake of simplicity, only the case of two contacting bodies with one sole contact interface is considered here. However, a generalization to multiple bodies and self contact is rather straightforward and mostly a matter of efficient search algorithms, see Section 4.6.2 and Section 4.6.3. All necessary notations for the finite deformation unilateral contact problem have already been introduced in Figure 2.3, to which the reader is once again referred at this point. The domains $\Omega_0^{(i)} \subset \mathbb{R}^3$ and $\Omega_t^{(i)} \subset \mathbb{R}^3$, $i = 1, 2$, represent two separate bodies in the reference and current configuration, respectively. To allow for the usual Dirichlet and Neumann boundary conditions as well as contact interaction, the surfaces $\partial\Omega_0^{(i)}$ are divided into three disjoint subsets $\Gamma_u^{(i)}$, $\Gamma_\sigma^{(i)}$ and $\Gamma_c^{(i)}$, where $\Gamma_c^{(i)}$ represents the *potential* contact surface. Similarly, the spatial surface descriptions $\partial\Omega_t^{(i)}$ are split into $\gamma_u^{(i)}$, $\gamma_\sigma^{(i)}$ and $\gamma_c^{(i)}$. Retaining a customary nomenclature in contact mechanics, $\Gamma_c^{(1)}$ is again referred to as slave surface and $\Gamma_c^{(2)}$ as master surface.

On each subdomain $\Omega_0^{(i)}$ the initial boundary value problem of finite deformation elastodynamics needs to be satisfied, viz.

$$\text{Div} \mathbf{P}^{(i)} + \hat{\mathbf{b}}_0^{(i)} = \rho_0^{(i)} \ddot{\mathbf{u}}^{(i)} \quad \text{in } \Omega_0^{(i)} \times [0, T], \quad (4.1)$$

$$\mathbf{u}^{(i)} = \hat{\mathbf{u}}^{(i)} \quad \text{on } \Gamma_u^{(i)} \times [0, T], \quad (4.2)$$

$$\mathbf{P}^{(i)} \mathbf{N}^{(i)} = \hat{\mathbf{t}}_0^{(i)} \quad \text{on } \Gamma_\sigma^{(i)} \times [0, T], \quad (4.3)$$

$$\mathbf{u}^{(i)}(\mathbf{X}^{(i)}, 0) = \hat{\mathbf{u}}_0^{(i)}(\mathbf{X}^{(i)}) \quad \text{in } \Omega_0^{(i)}, \quad (4.4)$$

$$\dot{\mathbf{u}}^{(i)}(\mathbf{X}^{(i)}, 0) = \hat{\dot{\mathbf{u}}}_0^{(i)}(\mathbf{X}^{(i)}) \quad \text{in } \Omega_0^{(i)}. \quad (4.5)$$

The contact constraints in normal direction are typically given in form of KKT conditions as defined in (2.55), while frictional sliding according to Coulomb's law has been introduced in (2.57). For the sake of completeness of the strong formulation, both sets of conditions are repeated:

$$g_n \geq 0, \quad p_n \leq 0, \quad p_n g_n = 0 \quad \text{on } \gamma_c^{(1)} \times [0, T], \quad (4.6)$$

$$\Phi := \|\mathbf{t}_\tau\| - \mathfrak{F}|p_n| \leq 0, \quad \mathbf{v}_{\tau, \text{rel}} + \beta \mathbf{t}_\tau = \mathbf{0}, \quad \beta \geq 0, \quad \Phi \beta = 0 \quad \text{on } \gamma_c^{(1)} \times [0, T]. \quad (4.7)$$

Equations (4.1)–(4.7) represent the final strong form of a unilateral contact problem in nonlinear solid mechanics. In the course of deriving a weak formulation (see next paragraph), the balance of linear momentum at the contact interface is typically exploited and a Lagrange multiplier vector $\boldsymbol{\lambda}$ is introduced, thus setting the basis for a mixed variational approach. In contrast to the mesh tying case in Chapter 3, it is striking that the unilateral contact constraints are typically formulated (and later also numerically evaluated) in the current configuration.

4.1.2 Weak formulation

In the first instance, the most general weak formulation including Coulomb friction is considered. Similar to the pure solid mechanics case in Section 2.3.1 and the mesh tying case in Section 3.1.2, the well-known solution spaces $\mathbf{U}^{(i)}$ and weighting spaces $\mathbf{V}^{(i)}$ are defined as

$$\mathbf{U}^{(i)} = \{ \mathbf{u}^{(i)} \in H^1(\Omega) \mid \mathbf{u}^{(i)} = \hat{\mathbf{u}}^{(i)} \text{ on } \Gamma_u \} , \quad (4.8)$$

$$\mathbf{V}^{(i)} = \{ \delta \mathbf{u}^{(i)} \in H^1(\Omega) \mid \delta \mathbf{u}^{(i)} = \mathbf{0} \text{ on } \Gamma_u \} . \quad (4.9)$$

Moreover, the Lagrange multiplier vector $\boldsymbol{\lambda} = -\mathbf{t}_c^{(1)}$, which represents the *negative* slave side contact traction $\mathbf{t}_c^{(1)}$ and is supposed to enforce the contact constraints (4.6) and (4.7), is chosen from the convex cone $\mathcal{M}(\boldsymbol{\lambda}) \subset \mathcal{M}$ given by

$$\mathcal{M}(\boldsymbol{\lambda}) = \left\{ \boldsymbol{\mu} \in \mathcal{M} \mid \langle \boldsymbol{\mu}, \mathbf{v} \rangle_{\gamma_c^{(1)}} \leq \langle \mathfrak{F} \boldsymbol{\lambda}_n, \|\mathbf{v}_\tau\| \rangle_{\gamma_c^{(1)}}, \mathbf{v} \in \mathcal{W} \text{ with } v_n \leq 0 \right\} . \quad (4.10)$$

Herein, $\langle \cdot, \cdot \rangle_{\gamma_c^{(1)}}$ again stands for the scalar or vector-valued duality pairing between $H^{-1/2}$ and $H^{1/2}$ on $\gamma_c^{(1)}$, see also Section 3.1.2. Moreover, \mathcal{M} is the dual space of the trace space $\mathcal{W}^{(1)}$ of $\mathbf{V}^{(1)}$ restricted to $\gamma_c^{(1)}$, i.e. $\mathcal{M} = H^{-1/2}(\gamma_c^{(1)})$ and $\mathcal{W}^{(1)} = H^{1/2}(\gamma_c^{(1)})$, where \mathcal{M} and $\mathcal{W}^{(1)}$ denote single scalar components of the corresponding vector-valued spaces \mathcal{M} and \mathcal{W} . Thus, the definition of the solution cone for the Lagrange multipliers in (4.10) satisfies the conditions on $\boldsymbol{\lambda}$ of the Coulomb friction law in a weak sense.

Based on these considerations, the weak saddle point formulation is derived next. Basically, this can be done by extending the standard weak formulation of nonlinear solid mechanics as defined in (2.61) to two bodies and combining it with contact-specific Lagrange multiplier contributions. Find $\mathbf{u}^{(i)} \in \mathbf{U}^{(i)}$ and $\boldsymbol{\lambda} \in \mathcal{M}(\boldsymbol{\lambda})$ such that

$$-\delta \mathcal{W}_{\text{kin}}(\mathbf{u}^{(i)}, \delta \mathbf{u}^{(i)}) - \delta \mathcal{W}_{\text{int,ext}}(\mathbf{u}^{(i)}, \delta \mathbf{u}^{(i)}) - \delta \mathcal{W}_{\text{co}}(\boldsymbol{\lambda}, \delta \mathbf{u}^{(i)}) = 0 \quad \forall \delta \mathbf{u}^{(i)} \in \mathbf{V}^{(i)} , \quad (4.11)$$

$$\delta \mathcal{W}_\lambda(\mathbf{u}^{(i)}, \delta \boldsymbol{\lambda}) \geq 0 \quad \forall \delta \boldsymbol{\lambda} \in \mathcal{M}(\boldsymbol{\lambda}) . \quad (4.12)$$

Herein, the kinetic contribution $\delta \mathcal{W}_{\text{kin}}$ as well as the internal and external contributions $\delta \mathcal{W}_{\text{int,ext}}$ to the overall virtual work of the two bodies do not change as compared with the mesh tying case in (3.11) and (3.12). However, the contact contribution $\delta \mathcal{W}_{\text{co}}$ and the weak constraints $\delta \mathcal{W}_\lambda$, including non-penetration and frictional sliding conditions, are given in full length as

$$-\delta \mathcal{W}_{\text{co}} = \int_{\gamma_c^{(1)}} \boldsymbol{\lambda} (\delta \mathbf{u}^{(1)} - \delta \mathbf{u}^{(2)} \circ \chi) \, dA , \quad (4.13)$$

$$\delta \mathcal{W}_\lambda = \int_{\gamma_c^{(1)}} (\delta \lambda_n - \lambda_n) g_n \, dA - \int_{\gamma_c^{(1)}} (\delta \boldsymbol{\lambda}_\tau - \boldsymbol{\lambda}_\tau) \mathbf{v}_{\tau,\text{rel}} \, dA , \quad (4.14)$$

where $\chi : \gamma_c^{(1)} \rightarrow \gamma_c^{(2)}$ defines a suitable mapping from slave to master side of the contact surface, see also Section 2.2.1. In contrast to the mesh tying case, where this mapping only came into play in the discrete setting, $\gamma_c^{(1)}$ and $\gamma_c^{(2)}$ cannot even be guaranteed to be identical in the continuum framework for unilateral contact, because they not only comprise the actual contact surfaces but the potential contact surfaces. As explained in detail in Section 3.1.2, the integral expressions in the coupling bilinear forms $\delta \mathcal{W}_{\text{co}}$ and $\delta \mathcal{W}_\lambda$ would need to be replaced by duality

pairings $\langle \cdot, \cdot \rangle_{\gamma_c^{(1)}}$ in order to be mathematically concise. However, the integral diction in (3.11) and (3.12) is preferred here due to readability. The coupling terms on $\gamma_c^{(1)}$ also allow for a direct interpretation in terms of variational formulations and the principle of virtual work. Whereas the contribution in (4.13) represents the virtual work of the unknown contact tractions $\boldsymbol{\lambda} = -\boldsymbol{t}_c^{(1)}$, the contribution in (4.14) ensures a weak, variationally consistent enforcement of the unilateral contact constraints in normal direction as well as the Coulomb friction law. The equivalence of the strong pointwise conditions given in (4.6) and (4.7) and the corresponding variational inequalities in (4.14) can readily be proven, see e.g. Wohlmuth [177].

The main focus of this thesis is on mortar finite element methods for contact mechanics in general, and on discrete dual Lagrange multiplier spaces in particular, rather than on the physical foundations of frictional sliding or other interface effects. Many scientific questions investigated and answered in the following are completely independent of the precise tangential contact model. Thus, for the sake of simplicity, the weak formulation is restricted to the *frictionless* case from now on, as well as the upcoming derivations concerning finite element discretization. Nevertheless, Coulomb friction is included in the actual implementation originating from this work, and special remarks on frictional sliding will be given where important, e.g. when considering semi-smooth Newton type active set strategies in Section 4.4 or in the numerical examples section. Without claiming that this list is exhaustive, details on the mortar finite element discretization of *frictional* contact can be found in Gitterle [54], Gitterle et al. [55], Hübner et al. [84], Puso and Laursen [133], Tur et al. [163], Wohlmuth [177] and Yang et al. [189].

For frictionless sliding, the tangential part \boldsymbol{t}_τ of the slave side contact traction $\boldsymbol{t}_c^{(1)}$ is supposed to vanish, and thus the set of frictional sliding conditions in (4.7) is simply replaced by

$$\boldsymbol{t}_\tau = \mathbf{0}. \quad (4.15)$$

Considering appropriate solution spaces, it becomes obvious that frictionless contact allows for a significant simplification of the convex cone of Lagrange multipliers, which is now given as

$$\mathcal{M}^+ = \left\{ \boldsymbol{\mu} \in \mathcal{M} \mid \boldsymbol{\mu}_\tau = \mathbf{0}, \langle \mu_n, w \rangle_{\gamma_c^{(1)}} \geq 0, w \in \mathcal{W}^+ \right\}. \quad (4.16)$$

Herein, \mathcal{W}^+ is a closed non-empty convex cone being defined by $\mathcal{W}^+ = \{w \in \mathcal{W}, w \geq 0\}$. The weak solution of the frictionless contact problem is then obtained from the following saddle point formulation: Find $\boldsymbol{u}^{(i)} \in \boldsymbol{\mathcal{U}}^{(i)}$ and $\boldsymbol{\lambda} \in \mathcal{M}^+$ such that

$$-\delta\mathcal{W}_{\text{kin}}(\boldsymbol{u}^{(i)}, \delta\boldsymbol{u}^{(i)}) - \delta\mathcal{W}_{\text{int,ext}}(\boldsymbol{u}^{(i)}, \delta\boldsymbol{u}^{(i)}) - \delta\mathcal{W}_{\text{co}}(\boldsymbol{\lambda}, \delta\boldsymbol{u}^{(i)}) = 0 \quad \forall \delta\boldsymbol{u}^{(i)} \in \boldsymbol{\mathcal{V}}^{(i)}, \quad (4.17)$$

$$\delta\mathcal{W}_\lambda(\boldsymbol{u}^{(i)}, \delta\boldsymbol{\lambda}) \geq 0 \quad \forall \delta\boldsymbol{\lambda} \in \mathcal{M}^+. \quad (4.18)$$

The contributions $\delta\mathcal{W}_{\text{kin}}$, $\delta\mathcal{W}_{\text{int,ext}}$ and $\delta\mathcal{W}_{\text{co}}$ remain unchanged as previously defined in (3.11), (3.12) and (4.13). However, the weak contact constraints $\delta\mathcal{W}_\lambda$ now reduce to

$$\delta\mathcal{W}_\lambda = \int_{\gamma_c^{(1)}} (\delta\lambda_n - \lambda_n) g_n \, dA. \quad (4.19)$$

Strictly speaking, a scalar Lagrange multiplier λ_n would be completely sufficient to enforce the non-penetration condition here. Yet, in view of the more general case of frictional contact, a

vector-valued Lagrange multiplier will also be employed for the frictionless case in this thesis, which allows for the nice interpretation of frictionless sliding as a special case of Coulomb's law with $\mathfrak{F} = 0$ and the convex cone of Lagrange multipliers $\mathcal{M}(\boldsymbol{\lambda})$ reducing to \mathcal{M}^+ . As compared with the mesh tying case in Section 3.1.2, it is noticeable that the weak formulation contains *inequality* conditions for unilateral contact. These require a particular numerical treatment based on active set strategies, as will be explained in Section 4.4. As mentioned before, all standard terms (representing kinetic, internal and external virtual work) are formulated in the reference configuration, while the contact virtual work term $\delta\mathcal{W}_{\text{co}}$ and the constraints $\delta\mathcal{W}_\lambda$ are typically formulated in the current configuration for the considered finite deformation contact problems. This is convenient due to the fact that the contact mapping $\chi : \gamma_c^{(1)} \rightarrow \gamma_c^{(2)}$ needs to be evaluated with respect to the deformed geometry, anyway.

4.2 Discretization and numerical integration

This paragraph first gives an overview of the most important steps associated with mortar finite element discretization of the frictionless unilateral contact problem described above. There exist certainly some similarities to the mesh tying case presented in Section 3.2, such as the definition of mortar matrices \mathbf{D} and \mathbf{M} and the corresponding numerical integration algorithms in both 2D and 3D. However, unilateral contact also introduces several additional complexities, which will be especially emphasized in the following. As an example, the combination of finite deformations and large relative tangential movements at the contact interface require a consistent linearization of all mortar coupling terms. Moreover, the non-penetration and frictionless sliding constraints need to be discretized, too, which leads to the important notion of weighted nodal gaps. It is shown, how the numerical integration of these weighted gaps as well as their consistent linearization is readily incorporated into the usual mortar coupling algorithms based on segmentation of the contact interface. Finally, some remarks are made on implicit time integration and energy and momentum conservation for unilateral contact problems, which proves to be a considerably more intricate task than for pure solid mechanics or mesh tying.

4.2.1 Finite element discretization

Similar to the tied contact case, all common types of first-order and second-order finite element interpolations in 2D and 3D are considered here, which again define finite dimensional subspaces $\mathcal{U}_h^{(i)}$ and $\mathcal{V}_h^{(i)}$ being approximations of $\mathcal{U}^{(i)}$ and $\mathcal{V}^{(i)}$, respectively. The general notations of slave and master side displacement interpolation given in (3.15), as well as the Lagrange multiplier interpolation defined in (3.17) are still valid. Substituting everything into the contact virtual work expression $\delta\mathcal{W}_{\text{co}}$ in (4.13) yields

$$\begin{aligned} -\delta\mathcal{W}_{\text{co},h} &= \sum_{j=1}^{m^{(1)}} \sum_{k=1}^{n^{(1)}} \boldsymbol{\lambda}_j^\top \left(\int_{\gamma_{c,h}^{(1)}} \Phi_j N_k^{(1)} \, dA \right) \delta\mathbf{d}_k^{(1)} \\ &\quad - \sum_{j=1}^{m^{(1)}} \sum_{l=1}^{n^{(2)}} \boldsymbol{\lambda}_j^\top \left(\int_{\gamma_{c,h}^{(1)}} \Phi_j (N_l^{(2)} \circ \chi_h) \, dA \right) \delta\mathbf{d}_l^{(2)}. \end{aligned} \quad (4.20)$$

Herein, the only two differences to the mesh tying case lie in the integration domain (spatial description $\gamma_{c,h}^{(1)}$ instead of material description $\Gamma_{c,h}^{(1)}$) and in the fact that the discrete contact mapping $\chi_h : \gamma_{c,h}^{(1)} \rightarrow \gamma_{c,h}^{(2)}$ now continuously changes due to a relative movement of slave and master surfaces. Thus, as will be seen later on, it is not sufficient to evaluate the mapping only once as for mesh tying, but the mortar matrices \mathbf{D} and \mathbf{M} become deformation-dependent instead. Due to the fundamental importance of the discrete mortar matrices, their blockwise definition is repeated here, although only slightly modified as compared with (3.19) and (3.20), i.e.

$$\mathbf{D}[j, k] = D_{jk} \mathbf{I}_{\text{ndim}} = \int_{\gamma_{c,h}^{(1)}} \Phi_j N_k^{(1)} dA \mathbf{I}_{\text{ndim}}, \quad j = 1, \dots, m^{(1)}, \quad k = 1, \dots, n^{(1)}, \quad (4.21)$$

$$\mathbf{M}[j, l] = M_{jl} \mathbf{I}_{\text{ndim}} = \int_{\gamma_{c,h}^{(1)}} \Phi_j (N_l^{(2)} \circ \chi_h) dA \mathbf{I}_{\text{ndim}}, \quad j = 1, \dots, m^{(1)}, \quad l = 1, \dots, n^{(2)}. \quad (4.22)$$

In analogy to (3.21), the discrete contact virtual work contribution can be expressed as

$$-\delta \mathcal{W}_{\text{co},h} = \delta \mathbf{d}_S^T \mathbf{D}^T \boldsymbol{\lambda} - \delta \mathbf{d}_M^T \mathbf{M}^T \boldsymbol{\lambda} = \delta \mathbf{d}^T \underbrace{\begin{bmatrix} \mathbf{0} \\ -\mathbf{M}^T \\ \mathbf{D}^T \end{bmatrix}}_{\mathbf{B}_{\text{co}}(\mathbf{d})^T} \boldsymbol{\lambda} = \delta \mathbf{d}^T \mathbf{f}_{\text{co}}(\mathbf{d}, \boldsymbol{\lambda}), \quad (4.23)$$

where the discrete mortar contact operator $\mathbf{B}_{\text{co}}(\mathbf{d})$ and the resulting discrete vector of contact forces $\mathbf{f}_{\text{co}}(\mathbf{d}, \boldsymbol{\lambda}) = \mathbf{B}_{\text{co}}(\mathbf{d})^T \boldsymbol{\lambda}$ acting on slave and master sides of the interface now depend *non-linearly* on the current deformation state \mathbf{d} .

Next, the focus is shifted towards the weak constraint contribution for frictionless contact defined in (4.19), where more profound differences to the mesh tying case can be expected. As shown in great detail in Hübner [79], the discretized version of the weak formulation in (4.18) and (4.19) is equivalent to the following set of pointwise conditions:

$$(\tilde{g}_n)_j \geq 0, \quad (\lambda_n)_j \geq 0, \quad (\tilde{g}_n)_j (\lambda_n)_j = 0, \quad j = 1, \dots, m^{(1)}, \quad (4.24)$$

where the discrete weighted gap $(\tilde{g}_n)_j$ at slave node j is given by

$$(\tilde{g}_n)_j = \int_{\gamma_c^{(1)}} \Phi_j g_{n,h} dA. \quad (4.25)$$

Herein, $g_{n,h}$ is the discretized version of the gap function g_n introduced in (2.46). Examining the last two equations in more detail, an interesting analogy becomes apparent. Basically, (4.24) represents nothing less than a discrete formulation of the original KKT conditions in (4.6) with an additional weighting based on the Lagrange multiplier shape functions Φ_j . It is worth noting that although a segment-based (mortar) approach has been followed, decoupled constraints at the discrete nodal points are eventually enforced independently, just as it is well-known from traditional NTS schemes. However, the nodal constraints (4.24) in the mortar formulation convey a substantially increased level of information as compared with the truly nodal constraints in a

NTS formulation, owing to the underlying variational approach which is algebraically reflected in the weighted (integral) gap formulation in (4.25).

For the sake of completeness, it should be pointed out that the nodal decoupling of constraints and thus the final formulation given in (4.24) is strictly speaking only valid for *dual* Lagrange multiplier interpolation, see Hübner [79] for the corresponding mathematical proof, which relies on biorthogonality as defined in (3.61). In the case of *standard* Lagrange multiplier interpolation, the conversion of (4.18) and (4.19) into (4.24) involves an additional, yet only slight, approximation, see Hübner [79]. Finally, the frictionless sliding constraint contained in the definition of the convex cone \mathcal{M}^+ is readily enforced on a discrete nodal basis, i.e. $(\boldsymbol{\lambda}_\tau)_j = \mathbf{0}$. To sum up, the final space discretized but still time continuous problem formulation, consisting of the semi-discrete equations of motion and the frictionless contact constraints for all slave nodes also carrying discrete Lagrange multiplier degrees of freedom, can be expressed as

$$\mathbf{M}\ddot{\mathbf{d}} + \mathbf{C}\dot{\mathbf{d}} + \mathbf{f}_{\text{int}}(\mathbf{d}) + \mathbf{f}_{\text{co}}(\mathbf{d}, \boldsymbol{\lambda}) - \mathbf{f}_{\text{ext}} = \mathbf{0}, \quad (4.26)$$

$$(\tilde{g}_n)_j \geq 0, \quad (\lambda_n)_j \geq 0, \quad (\tilde{g}_n)_j(\lambda_n)_j = 0, \quad j = 1, \dots, m^{(1)}, \quad (4.27)$$

$$(\boldsymbol{\lambda}_\tau)_j = \mathbf{0}, \quad j = 1, \dots, m^{(1)}. \quad (4.28)$$

While this finite element formulation has some strong similarities with the mesh tying case in (3.23) and (3.24), it also contains three striking additional complexities. Firstly, unilateral contact involves inequality constraints, which require a suitable active set strategy as part of the global solution algorithm (cf. Section 4.4). Secondly, normal and tangential contact directions need to be treated separately in order to enforce the different underlying physical principles (non-penetration, frictionless or frictional sliding). Thirdly, and most importantly from the viewpoint of implementation, the contact forces in (4.26) as well as the contact constraints in (4.27) and (4.28) are deformation-dependent. This introduces an additional nonlinearity into the global system and thus demands for an incessant re-evaluation of mortar coupling terms including a consistent linearization for implicit time integration. Corresponding extensions of the numerical integration scheme for the discrete contact operator $\mathbf{B}_{\text{co}}(\mathbf{d})$ and the discrete weighted gaps $(\tilde{g}_n)_j$ in both 2D and 3D will be presented in the next three paragraphs.

Finally, a short outlook is also given on the weak constraint contribution for frictional contact according to Coulomb's law as defined in (4.14), although the frictional part is not in the focus of interest here. Again, it has been shown in great detail in Hübner [79] and can be readily understood that the discretized version of the tangential part of the weak formulation in (4.12) and (4.14) is equivalent to the following set of pointwise conditions:

$$\begin{aligned} \Phi_j &:= \|(\boldsymbol{\lambda}_\tau)_j\| - \mathfrak{F}|(\lambda_n)_j| \leq 0, \\ (\tilde{\mathbf{v}}_{\tau,\text{rel}})_j + \beta_j(\boldsymbol{\lambda}_\tau)_j &= \mathbf{0}, \quad \beta_j \geq 0, \quad \Phi_j\beta_j = 0, \quad j = 1, \dots, m^{(1)}. \end{aligned} \quad (4.29)$$

where the discrete relative tangential velocity $(\tilde{\mathbf{v}}_{\tau,\text{rel}})_j$ at slave node j is determined such that it satisfies the requirement of frame indifference, see e.g. Yang et al. [189] and Gitterle et al. [55] for further explanations. Similar to the non-penetration condition, it can be observed that (4.29) basically represents a weak formulation of the original Coulomb friction conditions in (4.7) with an additional weighting based on the Lagrange multiplier shape functions Φ_j . In the semi-discrete formulation for Coulomb friction, the set of conditions in (4.29) would simply replace (4.28), while (4.26) and (4.27) would remain unchanged. While by no means exhaustive,

the given outlook demonstrates that an extension of the proposed mortar finite element framework towards any tangential constitutive law (e.g. Tresca friction, Coulomb friction) is pretty straightforward. Most importantly, the discrete frictional expressions such as the discrete relative tangential velocity $(\tilde{\mathbf{v}}_{\tau, \text{rel}})_j$ do not require any additional numerical integration efforts, but can rather be constructed from the well-known mortar matrices \mathbf{D} and \mathbf{M} (including history values due to path dependency) and the nodal tangent vectors $\boldsymbol{\tau}_j^\xi$ and $\boldsymbol{\tau}_j^\eta$ defined in (3.35).

4.2.2 Evaluation and linearization of mortar integrals in 2D

The main steps for evaluating the entries of the mortar integral matrices \mathbf{D} and \mathbf{M} in 2D have already been presented in Section 3.2.2 in the context of tied contact and apply in almost the same manner to the unilateral case considered now. Concretely, this regards the definition of averaged nodal normal vectors in (3.25) and the mortar segmentation algorithm (cf. Figure 3.3) with its associated projection procedures. Thus, as a starting point for the following derivations, the contribution of one mortar segment to \mathbf{D} and \mathbf{M} is simply repeated:

$$D_{jk} = \sum_{g=1}^{n_{\text{gp}}} w_g \Phi_j(\xi^{(1)}(\eta_g)) N_k^{(1)}(\xi^{(1)}(\eta_g)) J(\xi^{(1)}(\eta_g)) \frac{\xi_b^{(1)} - \xi_a^{(1)}}{2}, \quad (4.30)$$

$$M_{jl} = \sum_{g=1}^{n_{\text{gp}}} w_g \Phi_j(\xi^{(1)}(\eta_g)) N_k^{(2)}(\xi^{(2)}(\eta_g)) J(\xi^{(1)}(\eta_g)) \frac{\xi_b^{(1)} - \xi_a^{(1)}}{2}, \quad (4.31)$$

where n_{gp} is the number of Gauss integration points (with local segment coordinates η_g and weights w_g). Moreover, $\xi_a^{(1)}$ and $\xi_b^{(1)}$ represent the segment end points in local slave element coordinates, see Figure 3.3, and $J(\xi^{(1)}(\eta_g))$ denotes the slave element Jacobian determinant. While the underlying surface for the numerical integration of \mathbf{D} and \mathbf{M} has been fixed for mesh tying, namely $\Gamma_{c,h}^{(1)}$ in the reference configuration, the integration surface $\gamma_{c,h}^{(1)}$ for unilateral contact constantly changes. Consequently, projections between slave and master surface and thus the definition of integration segments in (4.30) and (4.31) become deformation-dependent, too. With regard to implicit time stepping and nonlinear solution schemes of Newton–Raphson type, this requires the consistent linearization of (4.30) and (4.31) with respect to the discrete nodal displacements \mathbf{d} . The linearization of one integration segment contribution (i.e. one slave and master element pair) to the entries D_{jk} of the first mortar matrix in 2D yields

$$\begin{aligned} \Delta D_{jk} = & \sum_{g=1}^{n_{\text{gp}}} w_g \Delta \Phi_j(\xi^{(1)}(\eta_g)) N_k^{(1)}(\xi^{(1)}(\eta_g)) J(\xi^{(1)}(\eta_g)) \frac{\xi_b^{(1)} - \xi_a^{(1)}}{2} \\ & + \sum_{g=1}^{n_{\text{gp}}} w_g \Phi_j(\xi^{(1)}(\eta_g)) \Delta N_k^{(1)}(\xi^{(1)}(\eta_g)) J(\xi^{(1)}(\eta_g)) \frac{\xi_b^{(1)} - \xi_a^{(1)}}{2} \\ & + \sum_{g=1}^{n_{\text{gp}}} w_g \Phi_j(\xi^{(1)}(\eta_g)) N_k^{(1)}(\xi^{(1)}(\eta_g)) \Delta J(\xi^{(1)}(\eta_g)) \frac{\xi_b^{(1)} - \xi_a^{(1)}}{2} \\ & + \sum_{g=1}^{n_{\text{gp}}} w_g \Phi_j(\xi^{(1)}(\eta_g)) N_k^{(1)}(\xi^{(1)}(\eta_g)) J(\xi^{(1)}(\eta_g)) \frac{\Delta \xi_b^{(1)} - \Delta \xi_a^{(1)}}{2} \end{aligned} \quad (4.32)$$

whereas linearization of the entries M_{jl} of the second mortar matrix is given by

$$\begin{aligned}
 \Delta M_{jl} = & \sum_{g=1}^{n_{gp}} w_g \Delta \Phi_j(\xi^{(1)}(\eta_g)) N_k^{(2)}(\xi^{(2)}(\eta_g)) J(\xi^{(1)}(\eta_g)) \frac{\xi_b^{(1)} - \xi_a^{(1)}}{2} \\
 & + \sum_{g=1}^{n_{gp}} w_g \Phi_j(\xi^{(1)}(\eta_g)) \Delta N_k^{(2)}(\xi^{(2)}(\eta_g)) J(\xi^{(1)}(\eta_g)) \frac{\xi_b^{(1)} - \xi_a^{(1)}}{2} \\
 & + \sum_{g=1}^{n_{gp}} w_g \Phi_j(\xi^{(1)}(\eta_g)) N_k^{(2)}(\xi^{(2)}(\eta_g)) \Delta J(\xi^{(1)}(\eta_g)) \frac{\xi_b^{(1)} - \xi_a^{(1)}}{2} \\
 & + \sum_{g=1}^{n_{gp}} w_g \Phi_j(\xi^{(1)}(\eta_g)) N_k^{(2)}(\xi^{(2)}(\eta_g)) J(\xi^{(1)}(\eta_g)) \frac{\Delta \xi_b^{(1)} - \Delta \xi_a^{(1)}}{2}. \quad (4.33)
 \end{aligned}$$

Herein, and for all upcoming derivations, the directional derivative of an arbitrary quantity with respect to the discrete nodal displacements \mathbf{d} is defined as

$$\Delta(\cdot) = \frac{\partial(\cdot)}{\partial \mathbf{d}} \Delta \mathbf{d}, \quad (4.34)$$

which allows for a very compact notation of complex linearizations. Examining the directional derivatives of D_{jk} and M_{jl} in more detail, one can identify several individual elementary contributions that require further explanations. Concretely, this encompasses the dual shape functions Φ_j , the slave and master displacement shape functions $N_k^{(1)}$ and $N_l^{(2)}$, the slave side segment end points $\xi_a^{(1)}$ and $\xi_b^{(1)}$ and the slave element Jacobian determinant $J(\xi^{(1)}(\eta_g))$. Applying the well-known chain rule of differentiation, further elementary linearizations, e.g. that of the Gauss point coordinates $\xi^{(1)}(\eta_g)$ and $\xi^{(2)}(\eta_g)$ and of the averaged nodal normal vector \mathbf{n}_k at slave node k , come to light. As the correct implementation of these linearizations accounts for the better part of the numerical effort associated with mortar discretization of finite deformation contact, detailed derivations of all individual terms in (4.32) and (4.33) are given in Appendix A.1.

4.2.3 Evaluation and linearization of mortar integrals in 3D

Again, the main steps for evaluating the entries of the mortar integral matrices \mathbf{D} and \mathbf{M} in 3D have already been presented in Section 3.2.3 in the context of tied contact and can be directly transferred to unilateral contact. Concretely, this encompasses the definition of averaged nodal normal vectors in (3.25) and the 3D mortar segmentation algorithm (cf. Figure 3.5) with its associated projection, clipping and triangulation procedures. Thus, as a starting point, the contribution of one mortar integration cell to \mathbf{D} and \mathbf{M} is recapitulated:

$$D_{jk} = \sum_{g=1}^{n_{gp}} w_g \Phi_j^{(1)}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) N_k^{(1)}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) J_{\text{cell}}, \quad (4.35)$$

$$M_{jl} = \sum_{g=1}^{n_{gp}} w_g \Phi_j^{(1)}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) N_l^{(2)}(\boldsymbol{\xi}^{(2)}(\tilde{\boldsymbol{\eta}}_g)) J_{\text{cell}}. \quad (4.36)$$

Herein, J_{cell} denotes the Jacobian determinant of the considered triangular integration cell, see Figure 3.5. Again, due to the fact that mortar segmentation and numerical integration are performed in the current configuration, i.e. on the current slave surface $\gamma_{c,h}^{(1)}$, all involved operations, such as the mapping between slave and master surfaces and the determination of integration cells for (4.35) and (4.36), become deformation-dependent. Thus, consistent linearizations of the mortar matrices \mathbf{D} and \mathbf{M} with respect to the discrete nodal displacements \mathbf{d} are needed. To begin with, the directional derivative of one integration cell contribution (i.e. one slave and master element pair) to the entries D_{jk} of the first mortar matrix in 3D yields

$$\begin{aligned} \Delta D_{jk} &= \sum_{g=1}^{n_{\text{gp}}} w_g \Delta \Phi_j^{(1)}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) N_k^{(1)}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) J_{\text{cell}} \\ &+ \sum_{g=1}^{n_{\text{gp}}} w_g \Phi_j(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) \Delta N_k^{(1)}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) J_{\text{cell}} \\ &+ \sum_{g=1}^{n_{\text{gp}}} w_g \Phi_j(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) N_k^{(1)}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) \Delta J_{\text{cell}}, \end{aligned} \quad (4.37)$$

whereas the directional derivative of the entries M_{jl} of the second mortar matrix is given by

$$\begin{aligned} \Delta M_{jl} &= \sum_{g=1}^{n_{\text{gp}}} w_g \Delta \Phi_j^{(1)}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) N_l^{(2)}(\boldsymbol{\xi}^{(2)}(\tilde{\boldsymbol{\eta}}_g)) J_{\text{cell}} \\ &+ \sum_{g=1}^{n_{\text{gp}}} w_g \Phi_j^{(1)}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) \Delta N_l^{(2)}(\boldsymbol{\xi}^{(2)}(\tilde{\boldsymbol{\eta}}_g)) J_{\text{cell}} \\ &+ \sum_{g=1}^{n_{\text{gp}}} w_g \Phi_j^{(1)}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) N_l^{(2)}(\boldsymbol{\xi}^{(2)}(\tilde{\boldsymbol{\eta}}_g)) \Delta J_{\text{cell}}. \end{aligned} \quad (4.38)$$

Examining these expressions in more detail, one can identify several individual elementary contributions that require further investigation. Concretely, the linearizations of D_{jk} and M_{jl} contain derivatives of the dual shape functions Φ_j , the slave and master displacement shape functions $N_k^{(1)}$ and $N_l^{(2)}$ and the integration cell Jacobian J_{cell} . Taking into account the chain rule of differentiation, it becomes obvious that linearizations of the Gauss point coordinates $\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)$ and $\boldsymbol{\xi}^{(2)}(\tilde{\boldsymbol{\eta}}_g)$, the integration cell vertices $\tilde{\mathbf{x}}_{1,2,3}^{\text{cell}}$ and the averaged nodal normal vector \mathbf{n}_k at slave node k need to be determined, too. Similar as for the 2D case, all details concerning these elementary directional derivatives in (4.37) and (4.33) are given in Appendix A.2.

4.2.4 Evaluation and linearization of weighted gaps

Numerical integration of the weighted gaps $(\tilde{g}_n)_j$ defined in (4.25) is based on the same algorithms as numerical integration of the mortar matrices, i.e. using mortar segments in 2D and mortar integration cells in 3D. In two dimensions, the contribution of one segment, defined by one slave and master element pair, is given as

$$(\tilde{g}_n)_j = \sum_{g=1}^{n_{\text{gp}}} w_g \Phi_j(\boldsymbol{\xi}^{(1)}(\eta_g)) g_{n,h}(\boldsymbol{\xi}^{(1)}(\eta_g), \boldsymbol{\xi}^{(2)}(\eta_g)) J(\boldsymbol{\xi}^{(1)}(\eta_g)) \frac{\xi_b^{(1)} - \xi_a^{(1)}}{2}, \quad (4.39)$$

whereas the contribution of one integration cell in 3D yields the expression

$$(\tilde{g}_n)_j = \sum_{g=1}^{n_{gp}} w_g \Phi_j(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) g_{n,h}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g), \boldsymbol{\xi}^{(2)}(\tilde{\boldsymbol{\eta}}_g)) J_{\text{cell}}. \quad (4.40)$$

Note, that both definitions contain the discrete gap function $g_{n,h}$, which is based on both slave and master side displacement interpolations according to (2.46). This fact is illustrated by the notations $g_{n,h}(\xi^{(1)}(\eta_g), \xi^{(2)}(\eta_g))$ and $g_{n,h}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g), \boldsymbol{\xi}^{(2)}(\tilde{\boldsymbol{\eta}}_g))$, respectively. Consistent linearization of the 2D version of the weighted gaps yields

$$\begin{aligned} \Delta(\tilde{g}_n)_j &= \sum_{g=1}^{n_{gp}} w_g \Delta \Phi_j(\xi^{(1)}(\eta_g)) g_{n,h}(\xi^{(1)}(\eta_g), \xi^{(2)}(\eta_g)) J(\xi^{(1)}(\eta_g)) \frac{\xi_b^{(1)} - \xi_a^{(1)}}{2} \\ &+ \sum_{g=1}^{n_{gp}} w_g \Phi_j(\xi^{(1)}(\eta_g)) \Delta g_{n,h}(\xi^{(1)}(\eta_g), \xi^{(2)}(\eta_g)) J(\xi^{(1)}(\eta_g)) \frac{\xi_b^{(1)} - \xi_a^{(1)}}{2} \\ &+ \sum_{g=1}^{n_{gp}} w_g \Phi_j(\xi^{(1)}(\eta_g)) g_{n,h}(\xi^{(1)}(\eta_g), \xi^{(2)}(\eta_g)) \Delta J(\xi^{(1)}(\eta_g)) \frac{\xi_b^{(1)} - \xi_a^{(1)}}{2} \\ &+ \sum_{g=1}^{n_{gp}} w_g \Phi_j(\xi^{(1)}(\eta_g)) g_{n,h}(\xi^{(1)}(\eta_g), \xi^{(2)}(\eta_g)) J(\xi^{(1)}(\eta_g)) \frac{\Delta \xi_b^{(1)} - \Delta \xi_a^{(1)}}{2}, \end{aligned} \quad (4.41)$$

whereas the directional derivative of the 3D version is given as

$$\begin{aligned} \Delta(\tilde{g}_n)_j &= \sum_{g=1}^{n_{gp}} w_g \Delta \Phi_j(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) g_{n,h}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g), \boldsymbol{\xi}^{(2)}(\tilde{\boldsymbol{\eta}}_g)) J_{\text{cell}} \\ &+ \sum_{g=1}^{n_{gp}} w_g \Phi_j(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) \Delta g_{n,h}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g), \boldsymbol{\xi}^{(2)}(\tilde{\boldsymbol{\eta}}_g)) J_{\text{cell}} \\ &+ \sum_{g=1}^{n_{gp}} w_g \Phi_j(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) g_{n,h}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g), \boldsymbol{\xi}^{(2)}(\tilde{\boldsymbol{\eta}}_g)) \Delta J_{\text{cell}}. \end{aligned} \quad (4.42)$$

Similar to the last two paragraphs, one can identify several individual elementary linearizations in (4.41) and (4.42) that require further investigation. Apart from well-known directional derivatives also occurring in ΔD_{jk} and ΔM_{jl} , $\Delta(\tilde{g}_n)_j$ now also contains the directional derivatives of the discrete gap function $g_{n,h}$. Further details concerning the derivation of $\Delta g_{n,h}$ for both 2D and 3D can be found in the Appendices A.1 and A.2 along with all other elementary linearizations.

4.2.5 Time discretization

Implicit time integration and especially energy and momentum conservation are an intricate topic for unilateral contact problems, and thus are still today subject to intensive research. Basically, one of the main reasons for this is the non-smoothness introduced by contact interaction, which is for instance reflected in discontinuities (jumps) of interface velocities in the event of an impact, see e.g. Laursen and Love [109]. Obviously, such discontinuities cannot be accurately captured with the common implicit time integration schemes, such as the generalized- α method or

the generalized energy-momentum method (GEMM) discussed in Section 2.3.3, which assume the time derivatives of the unknown displacements \mathbf{d} to be continuous. Thus, all well-known (smooth) time integrators based on midpoint rule, trapezoidal rule, Newmark's method and derived methods may exhibit some deficiencies, especially with respect to mechanical conservation laws, when applied to contact scenarios without specific adaptations. Some important remarks on this topic are given towards the end of this paragraph and in Section 4.2.6.

Nevertheless, time discretization of the semi-discrete contact problem in (4.26)–(4.28) is primarily done with the generalized- α method throughout this thesis, and provides a sufficient level of robustness and solution accuracy in all considered test cases. The final space and time discretized version of the frictionless contact problem formulation is obtained by adding the contact forces $\mathbf{f}_{\text{co}}(\mathbf{d}, \boldsymbol{\lambda}) = \mathbf{B}_{\text{co}}(\mathbf{d})^T \boldsymbol{\lambda}$ and the discrete contact constraints to the final problem formulation of nonlinear solid mechanics:

$$\mathbf{M}\mathbf{a}_{n+1-\alpha_m} + \mathbf{C}\mathbf{v}_{n+1-\alpha_f} + \mathbf{f}_{\text{int}}(\mathbf{d}_{n+1-\alpha_f}) + \mathbf{f}_{\text{co}}(\mathbf{d}_{n+1-\alpha_f}, \boldsymbol{\lambda}_{n+1-\alpha_f}) - \mathbf{f}_{\text{ext},n+1-\alpha_f} = \mathbf{0}, \quad (4.43)$$

$$(\tilde{g}_n)_{j,n+1} \geq 0, \quad (\lambda_n)_{j,n+1} \geq 0, \quad (\tilde{g}_n)_{j,n+1}(\lambda_n)_{j,n+1} = 0, \quad j = 1, \dots, m^{(1)}, \quad (4.44)$$

$$(\boldsymbol{\lambda}_\tau)_{j,n+1} = \mathbf{0}, \quad j = 1, \dots, m^{(1)}. \quad (4.45)$$

Again, in the context of generalized- α time integration, the most interesting aspect to discuss here is the choice of adequate points in time for the evaluation of the contact forces and the constraints in (4.43)–(4.45). As for mesh tying, the generalized mid-point $t_{n+1-\alpha_f}$ has been selected for the interface forces, whereas the end-point t_{n+1} is the preferred choice for constraint enforcement. In contrast to the tied contact case (see (3.41) and the related explanations), the unilateral contact constraints now depend *nonlinearly* on the discrete displacements \mathbf{d} , which suggests constraint enforcement at the end-point t_{n+1} of the considered time interval. Evaluating the contact forces at $t_{n+1-\alpha_f}$ is again perspicuous, because of the fact that the internal and external forces are also associated with that point in time in a generalized- α scheme. In full analogy to the internal force vector definitions in (2.83) and (2.84), the contact forces can either be determined based on the well-known trapezoidal rule, i.e.

$$\begin{aligned} \mathbf{f}_{\text{co}}(\mathbf{d}_{n+1-\alpha_f}, \boldsymbol{\lambda}_{n+1-\alpha_f}) &= (1 - \alpha_f)\mathbf{f}_{\text{co}}(\mathbf{d}_{n+1}, \boldsymbol{\lambda}_{n+1}) + \alpha_f\mathbf{f}_{\text{co}}(\mathbf{d}_n, \boldsymbol{\lambda}_n) \\ &= (1 - \alpha_f)\mathbf{B}_{\text{co}}(\mathbf{d}_{n+1})^T \boldsymbol{\lambda}_{n+1} + \alpha_f\mathbf{B}_{\text{co}}(\mathbf{d}_n)^T \boldsymbol{\lambda}_n, \end{aligned} \quad (4.46)$$

or alternatively based on the midpoint rule, i.e.

$$\mathbf{f}_{\text{co}}(\mathbf{d}_{n+1-\alpha_f}, \boldsymbol{\lambda}_{n+1-\alpha_f}) = \mathbf{B}_{\text{co}}(\mathbf{d}_{n+1-\alpha_f})^T \boldsymbol{\lambda}_{n+1-\alpha_f}. \quad (4.47)$$

As for the internal forces in (2.83) and (2.84), but unlike for the mesh tying forces in (3.41), these two possible definitions are *not* identical here. The trapezoidal rule version proposed in (4.46) seems more advantageous due to the fact that the resulting mortar contact formulation only requires an evaluation of all mortar coupling terms at the end-point t_{n+1} within each time step. If the midpoint rule version in (4.47) was chosen instead, mortar interface-related numerical integration would have to be performed partly at t_{n+1} (i.e. the weighted gaps $(\tilde{g}_n)_j$ for the constraints), but partly also at $t_{n+1-\alpha_f}$ (i.e. the mortar matrices \mathbf{D} and \mathbf{M} for the contact forces). To avoid this almost doubling of the computational effort, the contact forces based on the trapezoidal rule as defined in (4.46) are exclusively employed throughout this thesis.

As mentioned at the beginning of this paragraph, the issue of non-smoothness is an important aspect of time discretization in the context of unilateral contact and impact. Typically, the problem of non-smoothness (i.e. velocity jumps at the contact interface) is not addressed with special non-smooth time integrators, but rather through suitable modifications of standard time integrators, such as the generalized- α method or the GEMM. There exists a manifold of suggestions for improved implicit time integration schemes for contact and impact situations in the literature, however it still seems to be difficult to unify energy consistency, exact fulfillment of the non-penetration condition and second-order accuracy in time. For example, Laursen and Chawla [108] have achieved algorithmic energy conservation for frictionless contact within a NTS formulation, though at the price of slightly violating geometrical admissibility (i.e. non-penetration) of the solution. An enhanced approach, the so-called velocity update method (VUM), is presented in Laursen and Love [109] and indeed assures both energy conservation and exact constraint enforcement in the context of NTS contact, however the VUM sacrifices second-order accuracy in time. Only quite recently, all three requirements have successfully been unified in Hesch and Betsch [73, 74] by applying special coordinate augmentation techniques for both NTS and mortar-based discretizations.

A second well-known problem associated with classical time integration schemes for unilateral contact is induced by spurious oscillations, which may occur at the contact interface in the numerical solution due to an improper time discretization of the constraints. Some stabilization techniques, mostly based on modified Newmark methods, have been suggested to overcome this issue, see e.g. Kane et al. [92] and Deuffhard et al. [31]. Although being dissipative by design, the so-called contact modified Newmark algorithm proposed in Deuffhard et al. [31] has gained much popularity due to its quite easy implementation and efficient removal of spurious oscillations. A completely different approach to avoid numerical oscillations at the contact interface has been taken in Hager and Wohlmuth [62] based on modified quadrature formulas, which basically assign no mass to the contact interface nodes.

However, neither the topic of algorithmic energy conservation nor that of spurious oscillations are at the focus of interest in this thesis. Instead, mainly the well-known generalized- α and GEMM time integration schemes will be employed here, and their advantages as well as their limitations for unilateral contact problems will be analyzed with a suitable numerical example in Section 4.7.6. Exemplarily, one state-of-the-art approach, viz. the velocity update method as proposed in Laursen and Love [109], has been implemented and will also be investigated in Section 4.7.6.

4.2.6 Conservation laws

The fundamental conservation laws of linear momentum, angular momentum and mechanical energy have already been analyzed in detail for mortar finite element formulations in the context of tied contact, see Section 3.2.5. In the following, it is investigated which results can be directly transferred to the unilateral contact case, but also where additional difficulties arise. Again, results for momentum conservation in the semi-discrete setting, i.e. after finite element space discretization but before time discretization, are presented first. As elaborated in Section 3.2.5, the

requirements for linear and angular momentum conservation can be expressed as

$$\sum_{j=1}^{m^{(1)}} \left(\sum_{k=1}^{n^{(1)}} \mathbf{D}[j, k] \boldsymbol{\lambda}_j - \sum_{l=1}^{n^{(2)}} \mathbf{M}[j, l] \boldsymbol{\lambda}_j \right) := \mathbf{f}^{(1)} - \mathbf{f}^{(2)} = \mathbf{0}, \quad (4.48)$$

$$\sum_{j=1}^{m^{(1)}} \left[\sum_{k=1}^{n^{(1)}} \mathbf{x}_k^{(1)} \times (\mathbf{D}[j, k] \boldsymbol{\lambda}_j) - \sum_{l=1}^{n^{(2)}} \mathbf{x}_l^{(2)} \times (\mathbf{M}[j, l] \boldsymbol{\lambda}_j) \right] := \mathbf{m}^{(1)} - \mathbf{m}^{(2)} = \mathbf{0}. \quad (4.49)$$

These conditions for momentum conservation remain unchanged for unilateral contact due to the fact that mesh tying interface forces \mathbf{f}_{mt} in (3.21) and contact forces \mathbf{f}_{co} in (4.23) basically share a common definition, except for the deformation dependency of the mortar matrices \mathbf{D} and \mathbf{M} in the latter case. Thus, as explained for mesh tying, the proposed mortar methods for unilateral contact exactly conserve linear momentum if the two mortar integral matrices \mathbf{D} and \mathbf{M} are both integrated based on the same numerical integration scheme (see (3.44) for a proof).

Unfortunately, angular momentum conservation in the semi-discrete setting becomes more intricate for unilateral contact than it has been for mesh tying. According to (4.49), it requires that either the displacement jump vector \mathbf{g}_j becomes zero for each active slave node j , i.e.

$$\mathbf{g}_j = \sum_{k=1}^{n^{(1)}} \mathbf{D}[j, k] \mathbf{x}_k^{(1)} - \sum_{l=1}^{n^{(2)}} \mathbf{M}[j, l] \mathbf{x}_l^{(2)} = \mathbf{0}, \quad (4.50)$$

or alternatively that the discrete nodal Lagrange multiplier vector $\boldsymbol{\lambda}_j$ and \mathbf{g}_j are always collinear. However, as investigated by several authors, e.g. in Yang et al. [189], both conditions will usually be slightly violated for mortar-based contact formulations. This difference as compared with mesh tying can easily be understood when considering that only the nodal weighted gap $(\tilde{g}_n)_j$, which can be interpreted as the normal part of (4.50), is forced to be zero for unilateral contact. Therefore, conservation of angular momentum and rotational invariance of the semi-discrete system are a challenging issue in the context of mortar methods.

Commonly, and also throughout this thesis, the variation of the mortar integrals is neglected when deriving the discrete contact virtual work expression (4.23), see also Puso and Laursen [132, 133]. Yet, it has been demonstrated recently in Hesch and Betsch [71] that these additional terms must be considered in order to assure *exact* angular momentum conservation. If exact angular momentum conservation is important, it is possible to modify the presented approach such that the variation of the mortar integrals \mathbf{D} and \mathbf{M} is taken into account in the discrete contact virtual work. Consistent linearization of this extended formulation would include second derivatives of all deformation-dependent quantities in \mathbf{D} and \mathbf{M} . Yet, whenever contact surfaces become planar or conforming, exact angular momentum conservation is recovered for the proposed method. Moreover, numerical examples in Section 4.7 demonstrate that the violations of angular momentum conservation are typically not severe in practice. This is due to the fact that even for most relevant cases including curved interfaces, one can expect the resulting interface moment in (4.49) to be very small, as the weighted nodal displacement jump vectors \mathbf{g}_j of active nodes are typically very small.

Finally, some brief remarks are given on the fully, i.e. space and time discretized setting, where time discretization is based on the generalized- α method or the GEMM, see Section 2.3.3 and

the preceding paragraph. While not the focus in this thesis, numerical investigations show that conservation properties concerning angular momentum and mechanical energies are not necessarily retained for these methods, when transferred from pure nonlinear solid mechanics or mesh tying to unilateral contact situations. Only linear momentum conservation remains untouched and can always be guaranteed for mortar contact formulations, if numerical integration of the coupling terms at the interface is performed as described above. Further details on this issue can be found in Section 4.7.6 and the corresponding literature, e.g. in Hesch and Betsch [71, 74].

4.3 Discrete Lagrange multiplier spaces

The most important aspects of both standard and dual Lagrange multiplier interpolation, along with suitable discrete Lagrange multiplier spaces for first-order and second-order finite elements in 2D and 3D, have already been introduced in the context of tied contact in Section 3.3. Apart from certain additional requirements that need to be taken into account, these discrete Lagrange multiplier spaces derived in Sections 3.3.1.1–3.3.1.4 as well as Sections 3.3.2.1–3.3.2.4 can completely be re-used for mortar formulations of unilateral contact. The following paragraph first emphasizes special demands on the Lagrange multipliers in that context, which basically have to do with the inequality nature of contact constraints and thus with positivity requirements for the Lagrange multiplier shape functions. Moreover, it will be demonstrated that similar to the discrete mortar matrices \mathbf{D} and \mathbf{M} , a consistent linearization of the dual shape functions themselves becomes necessary for unilateral contact due to biorthogonality and the resulting deformation-dependency of the shape functions Φ_j . The last paragraph is devoted to the issue of consistency of dual Lagrange multipliers in partly integrated surface elements, which is highly relevant for actual real-world applications, e.g. when considering two bodies sliding off each other at a sharp edge (often referred to as so-called dropping edge problems).

In principle, all explanations concerning the reproduction order r_λ of the Lagrange multiplier interpolation as well as the expected spatial convergence rates of the discretization error, i.e. $\mathcal{O}(h^{p+1})$ in the L^2 -norm and $\mathcal{O}(h^p)$ in the energy norm, are still valid. However, due to the a priori unknown transition between contact and non-contact regions and the associated change of type of boundary conditions from Dirichlet to Neumann type, the solutions of contact problems typically exhibit a reduced regularity as compared with smooth solid mechanics. Exemplarily, the reader is referred to Hübner and Wohlmuth [81] and the references therein for a comprehensive overview of a priori error estimates for first-order mortar finite elements and unilateral contact. Most importantly, optimal $\mathcal{O}(h)$ convergence of first-order elements measured in the energy norm can be demonstrated under quite weak assumptions on regularity, i.e. if the solution is H^2 -regular. The case of second-order interpolation is more intricate, and error analysis has mostly focused on some special cases and the two-dimensional setting, see e.g. Belhachmi and Ben Belgacem [8], Hild and Laborde [76] and Hübner et al. [83]. Recently, a very general framework for a priori estimates for contact problems in 3D with quadratic finite elements using standard or dual Lagrange multipliers has been developed in Wohlmuth et al. [179]. Therein, the most important result is that $\mathcal{O}(h^{t-1})$, $2 < t < 5/2$, a priori results in the energy norm for the discretization error can be established provided that the solution is regular enough. With the solution of a unilateral contact problem typically being only $H^{5/2-\epsilon}$ -regular with $\epsilon > 0$, this means that it is not possible to obtain better a priori estimates than $\mathcal{O}(h^{3/2})$ in the energy

norm. Consequently, for $t < 2$ no qualitative gain from the use of quadratic finite elements can be expected, but only for $2 < t < 5/2$. A suitable numerical example in Section 4.7.2 will demonstrate the validity of these theoretical considerations on optimal convergence of mortar finite element methods for frictionless unilateral contact. However, mathematical analysis and the actual derivation of a priori error estimates are not at the focus of interest in this thesis, and the interested reader is instead referred to Wohlmuth et al. [179] and the references therein.

4.3.1 Special requirements for contact mechanics

First of all, the biorthogonality condition for dual Lagrange multipliers as introduced in (3.61) needs to be enforced in the *current* configuration for unilateral contact, i.e.

$$\int_{\gamma_{c,h}^{(1)}} \Phi_j N_k^{(1)} dA = \delta_{jk} \int_{\gamma_{c,h}^{(1)}} N_k^{(1)} dA, \quad j, k = 1, \dots, m^{(1)}, \quad (4.51)$$

where again the most common choice $m^{(1)} = n^{(1)}$ is assumed, i.e. all slave nodes also carry discrete Lagrange multiplier degrees of freedom. The elementwise biorthogonality condition typically employed in practice yields

$$\int_e \Phi_j N_k^{(1)} de = \delta_{jk} \int_e N_k^{(1)} de, \quad j, k = 1, \dots, m_e^{(1)}. \quad (4.52)$$

where, in contrast to (3.62), the integration domain e now represents a slave element in *spatial* description. Thus, similar to the mortar matrices \mathbf{D} and \mathbf{M} , the definition of dual shape functions themselves becomes deformation-dependent in the unilateral contact case.

Before considering the unilateral contact case, some fundamental requirements for discrete Lagrange multiplier spaces in the context of mortar mesh tying are reviewed here, see Section 3.3. First of all, both standard and dual Lagrange multiplier interpolation are guaranteed to satisfy partition of unity, i.e. $\sum_j \Phi_j = 1$, $j = 1, \dots, m_e^{(1)}$, on each slave element. While obvious for standard Lagrange multipliers, this property is assured by the biorthogonality condition in the dual interpolation case. Another requirement already introduced for tied contact is that the Lagrange multiplier shape functions must satisfy integral positivity, i.e. $\int_e \Phi_j de > 0$. However, integral positivity has not yet been discussed in detail, but was introduced without supplying a reasonable explanation for its imposition. Only the unilateral contact case considered now is able to provide the reason why integral positivity is a fundamental requirement for the Lagrange multiplier shape functions. In the continuum setting, the non-penetration condition can be expressed in terms of non-negativity of the gap function, i.e. $g_n \geq 0$. As discussed in Section 4.2.1, this is reflected in a mortar FE discretization by weighted nodal gaps $(\tilde{g}_n)_j$ defined as

$$(\tilde{g}_n)_j = \int_{\gamma_c^{(1)}} \Phi_j g_{n,h} dA \geq 0. \quad (4.53)$$

Assuming the most simple case of a constant positive gap function g_n , i.e. two planar opposite contact surfaces $\gamma_c^{(1)}$ and $\gamma_c^{(2)}$, allows for an obvious physical motivation of integral positivity of the Lagrange multiplier shape functions. It is reasonable to demand that all weighted nodal gaps $(\tilde{g}_n)_j$ must be positive in that case. This, in turn, requires that

$$\int_e \Phi_j de > 0, \quad (4.54)$$

which is nothing less than a re-statement of the already well-known integral positivity requirement. For the sake of completeness, it should be pointed out that this property would not necessarily have been required in the context of mortar mesh tying, but non-zero integrals $\int_e \Phi_j \, de \neq 0$ would have been sufficient there. However, the condition in (4.54) is mandatory for unilateral contact, and thus for reasons of simplicity of presentation has already been enforced for the mesh tying case, too. Therefore, all definitions of standard and dual Lagrange multiplier spaces given in Section 3.3 can be applied to mortar formulations of unilateral contact without any modifications, except from the aforementioned fact that dual shape functions must be determined with respect to the current slave surface $\gamma_c^{(1)}$, see (4.51) and (4.52).

Finally, it should be pointed out that, while integral positivity of the Lagrange multiplier shape functions is a *minimum* requirement for unilateral contact, a further restriction to non-negative shape functions, i.e. $\Phi_j \geq 0$, is even more desirable. This proposition is easy to understand when considering a positive, but non-constant gap function $g_n > 0$ and the definition of the nodal weighted gaps $(\tilde{g}_n)_j$. Such situations may spuriously result in a negative weighted gap value $(\tilde{g}_n)_j < 0$ at certain slave nodes, which not only contradicts the continuum setting, but may lead to unacceptable errors or even to a non-converging active set strategy, see Section 4.4. Unfortunately, the stricter requirement of non-negativity is only met by standard Lagrange multiplier shape functions for first-order finite elements, but neither by the dual version for first-order elements nor for any second-order interpolation case discussed in this thesis. Nevertheless, a possible remedy, at least for first-order dual Lagrange multipliers, has recently been presented in Popp et al. [127]. Instead of keeping the usual strict separation of standard and dual Lagrange multiplier interpolation, the strengths of both approaches are unified in a kind of Petrov–Galerkin interpolation for the Lagrange multipliers, using different interpolations for λ_h and $\delta\lambda_h$, i.e.

$$\lambda_h = \sum_{j=1}^{m^{(1)}} \Phi_j \lambda_j, \quad \delta\lambda_h = \sum_{j=1}^{m^{(1)}} N_j^{(1)} \delta\lambda_j. \quad (4.55)$$

Dual shape functions are employed for the Lagrange multiplier field, thus resulting in the desired diagonal structure of mortar matrix \mathbf{D} , which allows for the condensation of the discrete Lagrange multiplier degrees of freedom, see Sections 3.4.3 and 4.5.4. The interpolation of $\delta\lambda_h$ in the constraint equations, on the other hand, is done by standard shape functions. In the case of first-order finite elements, these shape functions are strictly positive on the entire contact surface $\gamma_{c,h}^{(1)}$. Hence, arbitrary positive gap functions also yield positive weighted gap values $(\tilde{g}_n)_j$. Therefore, as mentioned above, this Petrov–Galerkin type of approach for the Lagrange multipliers unifies the advantages of dual and standard mortar methods. The discrete Lagrange multiplier degrees of freedom can still be condensed from the global system of equations, as typical and beneficial of the dual approach, but at the same time it provides the robustness typical of standard Lagrange multiplier methods in the active set search. The interested reader is referred to Popp et al. [127] for further details and numerical validation.

4.3.2 Linearization of dual Lagrange multipliers

As discussed in Section 3.3.2 and in the last paragraph, dual shape functions can in general not be defined a priori, but rather depend on the actual slave element distortion. In the context of unilateral contact, where the mortar matrices \mathbf{D} and \mathbf{M} and thus also the biorthogonality

condition (4.51) must be evaluated in the current configuration, the dual shape functions Φ_j themselves become nonlinear functions of the current displacement state \mathbf{d} and require consistent linearization for the application of Newton–Raphson methods as nonlinear solution scheme. The corresponding numerical procedure, basically based on linearizing the biorthogonality condition, has first been presented in Popp et al. [123, 124] and will be outlined in the following.

In (3.64) and (3.65), it has been shown how element-specific dual shape functions are constructed in the reference configuration for mesh tying problems. In the case of unilateral contact, the general approach stays the same, i.e.

$$\Phi_j(\xi, \eta) = a_{jk} N_k^{(1)}(\xi, \eta), \quad \mathbf{A}_e = [a_{jk}] \in \mathbb{R}^{m_e^{(1)} \times m_e^{(1)}}. \quad (4.56)$$

However, (3.65) must be evaluated in the current configuration instead, viz.

$$\begin{aligned} \mathbf{A}_e &= \mathbf{D}_e \mathbf{M}_e^{-1}, \\ \mathbf{D}_e &= [d_{jk}] \in \mathbb{R}^{m_e^{(1)} \times m_e^{(1)}}, \quad d_{jk} = \delta_{jk} \int_e N_k^{(1)}(\xi, \eta) J(\xi, \eta) de, \\ \mathbf{M}_e &= [m_{jk}] \in \mathbb{R}^{m_e^{(1)} \times m_e^{(1)}}, \quad m_{jk} = \int_e N_j^{(1)}(\xi, \eta) N_k^{(1)}(\xi, \eta) J(\xi, \eta) de, \end{aligned} \quad (4.57)$$

where $J(\xi, \eta)$ is now the Jacobian determinant of slave element e in the current configuration, and depends nonlinearly on the discrete nodal displacements \mathbf{d} . Linearization of the coefficient matrix \mathbf{A}_e in the first line of (4.57) yields the directional derivative

$$\Delta \mathbf{A}_e = \Delta \mathbf{D}_e \mathbf{M}_e^{-1} - \mathbf{A}_e \Delta \mathbf{M}_e \mathbf{M}_e^{-1}, \quad (4.58)$$

where corresponding expressions for the entries d_{jk} and m_{jk} , $j, k = 1, \dots, m_e^{(1)}$ of the coefficient matrices \mathbf{D}_e and \mathbf{M}_e based on Gauss quadrature are given as

$$\begin{aligned} \Delta d_{jk} &= \delta_{jk} \sum_{g=1}^{n_{gp}} w_g N_k(\xi_g, \eta_g) \Delta J(\xi_g, \eta_g), \\ \Delta m_{jk} &= \sum_{g=1}^{n_{gp}} w_g N_j(\xi_g, \eta_g) N_k(\xi_g, \eta_g) \Delta J(\xi_g, \eta_g). \end{aligned} \quad (4.59)$$

It can easily be observed that a linearization of the slave element Jacobian determinant $J(\xi_g, \eta_g)$ at each Gauss point is necessary for the evaluation of (4.59). Details concerning the derivation of ΔJ for both 2D and 3D can be found in the Appendices A.1 and A.2 along with all other elementary directional derivatives needed for mortar coupling at contact interfaces.

4.3.3 Consistency of dual Lagrange multipliers

Another important algorithmic issue for mortar-based finite deformation contact is the consistent definition of biorthogonality and dual Lagrange multiplier shape functions in partially integrated slave elements. The dual shape function definition employed so far requires that all slave elements on $\gamma_{c,h}^{(1)}$ are *completely* integrated when determining the mortar matrices \mathbf{D} and \mathbf{M} . However, in the course of certain contact simulations, slave elements may not always fully participate

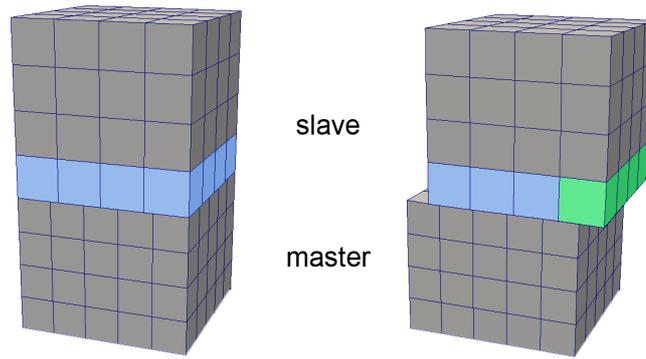


Figure 4.1: Consistency of dual Lagrange multipliers in 3D – standard biorthogonality is sufficient for fully integrated slave elements (associated volume elements in blue). A modification is required for partially integrated slave elements (associated volume elements in green).

in the contact integration zone, for example in so-called dropping edge problems, when one body slides off another at a sharp edge (see Figure 4.1). In this case the use of dual shape functions calculated according to (4.56) and (4.57) does not yield a diagonal mortar matrix \mathbf{D} . This is due to the fact that different integration domains are employed; while the biorthogonality in (4.52) is still defined on each entire slave element e , the entries of \mathbf{D} and \mathbf{M} can only be calculated on the part of $\gamma_{c,h}^{(1)}$ where a feasible projection onto the master surface is possible. As first analyzed for the 2D case in Cichosz and Bischoff [25], the resulting mortar contact formulation could not assure biorthogonality and thus the crucial advantage of dual shape functions would be lost.

The basic idea behind the remedy presented in Cichosz and Bischoff [25] is to restrict the biorthogonality condition on only that fraction of each slave element actually being part of the numerical integration region. Recently, a generalization of this approach to the 3D mortar setting has been proposed in Popp et al. [127], and the most important steps are summarized in the following. Simply speaking, the only necessary modification in order to make the biorthogonality condition consistent in any event, is to replace the integration domain in (4.57) by the fraction $e_{\text{int}} \subseteq e$ of the slave element, that actually participates in the contact integration area, i.e.

$$\begin{aligned} \mathbf{A}_e &= \mathbf{D}_e \mathbf{M}_e^{-1}, & (4.60) \\ \mathbf{D}_e &= [d_{jk}] \in \mathbb{R}^{m_e^{(1)} \times m_e^{(1)}}, \quad d_{jk} = \delta_{jk} \int_{e_{\text{int}}} N_k^{(1)}(\xi, \eta) J(\xi, \eta) de_{\text{int}}, \\ \mathbf{M}_e &= [m_{jk}] \in \mathbb{R}^{m_e^{(1)} \times m_e^{(1)}}, \quad m_{jk} = \int_{e_{\text{int}}} N_j^{(1)}(\xi, \eta) N_k^{(1)}(\xi, \eta) J(\xi, \eta) de_{\text{int}}. \end{aligned}$$

Within a two-dimensional problem setting, the contact surface consists of 1D slave elements with a local coordinate $\xi \in [-1, 1]$. The beginning and the end of the contact integration area can then be easily identified by two parameters ξ_{\min} and ξ_{\max} , and the integration in (4.60) can be restricted to the interval $[\xi_{\min}, \xi_{\max}]$, as has been shown in Cichosz and Bischoff [25]. Figure 4.2 exemplarily illustrates the resulting difference between the well-known dual shape functions Φ_j and their modified version.

In 3D however, the detection of this overlapping region is not as simple. To evaluate the integrals in (4.60), the slave surface is projected onto the master surface and the overlap between

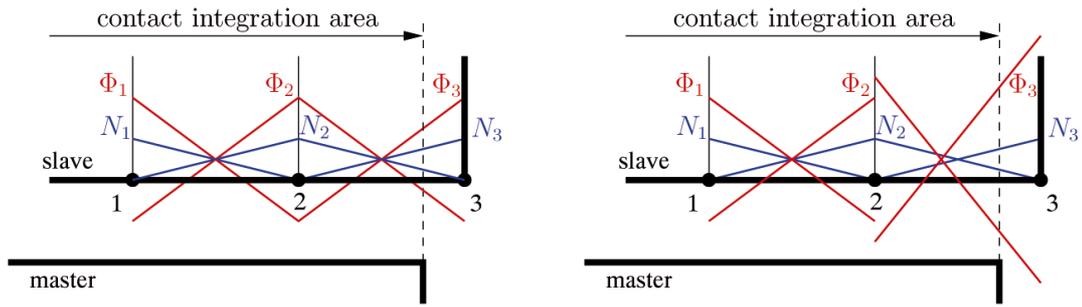


Figure 4.2: Detection of contact integration area and consistent dual shape functions in fully integrated and partially integrated slave elements in 2D.

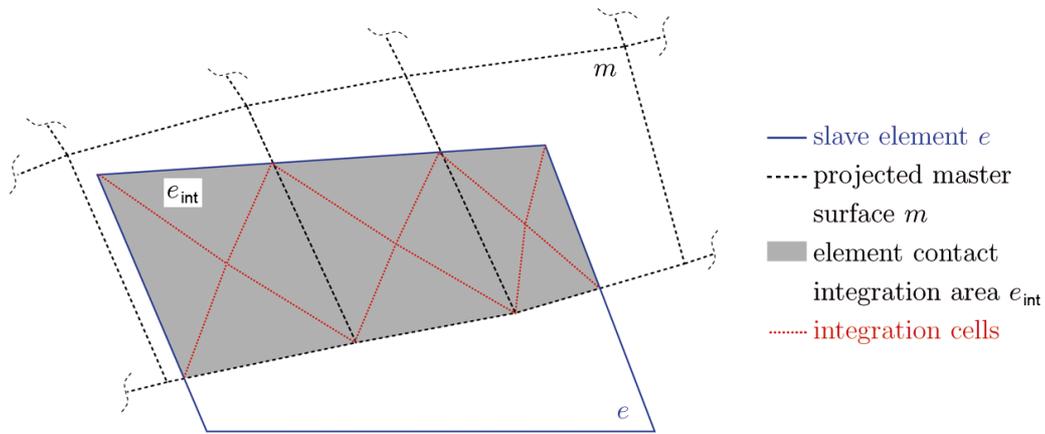


Figure 4.3: Detection of contact integration area in a partially integrated element in 3D.

each element pair is triangulated to obtain integration cells. The coefficient matrix \mathbf{A}_e for one particular slave element is then calculated over all integration cells associated with this element. Interestingly, the projection, overlap detection and triangulation procedure is exactly the same as it is typically used for the mortar matrices in 3D anyway, see Section 3.2.3, and can be used again here. Thus, the proposed modification fits perfectly into the usual mortar coupling framework, as it merely requires a re-arrangement of existing algorithm components. Figure 4.3 illustrates one such slave element, which only partially contributes to the contact integration area.

To ensure well-defined dual shape functions it is now necessary to not only have integrally positive, but non-negative displacement shape functions $N_k^{(1)}$, because arbitrary fractions of the slave element domain could be the integration domain in (4.60) and no integral value may become zero. Again, in the case of first-order finite elements, this is always guaranteed, while second-order interpolation would require an additional basis transformation procedure similar to the one discussed in Section 3.3.2.4. The resulting modified dual shape functions Φ_j satisfy the minimum requirement for unilateral contact, i.e. $\int_{\gamma_{c,h}^{(1)}} \Phi_j dA > 0$ not only for fully integrated but also for partially integrated slave elements. Further details and remarks, including consistent linearization of the modified dual shape functions, a nodal scaling procedure to avoid ill-conditioning of the global system matrix due to very small slave element fractions and extensive numerical validations are to be found in Popp et al. [127].

4.4 Active set strategy and semi-smooth Newton methods

As mentioned before, the fully discrete problem statement of unilateral contact in (4.43)–(4.45) causes one major additional complexity with regard to global solution schemes as compared with the mesh tying case, namely the contact specific *inequality* constraints, which divide the set of all discrete constraints (i.e. the equivalent of all slave nodes) into two a priori unknown sets of active and inactive constraints. Mathematically speaking, this introduces an additional source of nonlinearity apart from the well-known geometrical and material nonlinearities of nonlinear solid mechanics. To resolve this contact nonlinearity, so-called primal-dual active set strategies (PDASS) will be employed in the solution algorithms developed throughout this thesis.

In the following, a short overview of the chosen PDASS approach and its reformulation based on nonlinear complementarity functions will be given. Then, the concrete application to frictionless contact in the fully nonlinear realm (i.e. including finite deformations) will be proposed and presented in full detail. Moreover, the close relation of the resulting solution algorithm based on semi-smooth Newtons to the Augmented Lagrange method, or to be more precise, to the variant of the Augmented Lagrange method which actually keeps the Lagrange multipliers as additional unknowns, is emphasized once again (see also the explanations in Section 2.4.4). Finally, the section is concluded with some remarks on a possible further extension of the PDASS and semi-smooth Newton methods towards the Coulomb friction case in finite sliding contact situations.

4.4.1 General concepts of the primal-dual active set strategy

The idea of any active set strategy in the context of unilateral contact is to find the correct subset of all slave nodes which are in contact with the master surface at the end of the currently considered time interval $[t_n, t_{n+1}]$. As discussed in Section 4.2.1, the contact constraints can be enforced nodally at each slave node $j \in \mathcal{S}$, with $j = 1, \dots, m^{(1)}$, despite the fact that a segment-based mortar approach is employed here. Consequently, the so-called active set $\mathcal{A} \subseteq \mathcal{S}$ defines a subset of the set of all slave nodes \mathcal{S} , and the definition of the inactive set $\mathcal{I} = \mathcal{S} \setminus \mathcal{A}$ is straightforward. Before considering possible formulations of active set strategies, the final KKT conditions defined in (4.44) are repeated here, with the time index $n + 1$ being omitted in the following for the sake of notational simplicity, i.e.

$$\begin{aligned} (\tilde{g}_n)_j &\geq 0, & \forall j \in \mathcal{S} \\ (\lambda_n)_j &\geq 0, & \forall j \in \mathcal{S} \\ (\tilde{g}_n)_j (\lambda_n)_j &= 0, & \forall j \in \mathcal{S}. \end{aligned} \tag{4.61}$$

The aforementioned definitions of the active set and the inactive set in combination with the complementarity condition $(\tilde{g}_n)_j (\lambda_n)_j = 0$ motivate a first, naive reformulation of the KKT conditions using only equality constraints:

$$\begin{aligned} (\tilde{g}_n)_j &= 0, & \forall j \in \mathcal{A} \\ (\lambda_n)_j &= 0, & \forall j \in \mathcal{I} \\ (\tilde{g}_n)_j (\lambda_n)_j &= 0, & \forall j \in \mathcal{S}. \end{aligned} \tag{4.62}$$

Obviously, the PDASS in (4.62) suffers from a serious drawback: the contact nonlinearity, i.e. finding the correct active set \mathcal{A} can not be resolved by a Newton–Raphson type approach. This is due to the fact that no directional derivative of the sets themselves with respect to the nodal displacements \mathbf{d} can be extracted from (4.62). Instead, the given formulation inevitably leads to two nested iterative solution schemes, with the outer (fixed-point type) loop solving for the correct active set and the inner (Newton–Raphson type) loop solving a constrained nonlinear finite element problem while the active set is *fixed*. Consequently, this approach does not provide the desired efficiency and will not be followed any further in this thesis. Further information on such a fixed-point type treatment of the active set in the context of finite deformation mortar contact can for instance be found in Hartmann et al. [66] and Hesch and Betsch [71].

Based on the above considerations, the basic idea of an alternative PDASS formulation is to rearrange the KKT conditions such that a Newton–Raphson type algorithm can be applied not only for geometrical and material nonlinearities, but also for the nonlinearity stemming from contact itself, i.e. the active set search. The resulting primal-dual active set approach is well-known from the general mathematical literature on constrained optimization, see e.g. in Hintermüller et al. [77] and Qi and Sun [135], and can equivalently be interpreted as a semi-smooth Newton method. Applications to classical NTS contact formulations can be found in Alart and Curnier [3], Christensen et al. [23] and Strömberg et al. [153], and small deformation mortar contact has been investigated in Hübner and Wohlmuth [80]. Here, the first successful consistent extension to a finite deformation mortar contact formulation is presented, cf. also Popp et al. [123, 124]. The main idea is to reformulate the discrete KKT conditions within a so-called nonlinear complementarity (NCP) function, where all details for frictionless and frictional contact are given in the upcoming paragraphs. For the sake of completeness, it should be mentioned that the concept of NCP functions is also applicable to other well-known solid mechanics problems involving inequality constraints such as computational plasticity. For a comprehensive and more general overview, the reader is exemplarily referred to Hager [61].

4.4.2 Application to frictionless contact

The first step for frictionless contact is to reformulate the discrete KKT-conditions in (4.61) within a complementarity function C_j for each slave node $j \in \mathcal{S}$ as

$$C_j(\mathbf{d}, \boldsymbol{\lambda}) = (\lambda_n)_j - \max(0, (\lambda_n)_j - c_n(\tilde{g}_n)_j) = 0, \quad c_n > 0. \quad (4.63)$$

This is a nonlinear function of the discrete displacements as both the nodal normal vector \mathbf{n}_j in $(\lambda_n)_j = \mathbf{n}_j \cdot \boldsymbol{\lambda}_j$ and the nodal weighted gap $(\tilde{g}_n)_j$ defined in (4.25) depend nonlinearly on \mathbf{d} . It can be easily shown that the resulting equality constraint $C_j = 0$ is *equivalent* to the complete set of KKT inequality conditions in (4.61), and that this equivalence holds for arbitrary positive values of the so-called complementarity parameter c_n . The concrete role of c_n will be explained later in this paragraph. Figure 4.4 exemplarily illustrates the nodal complementarity function and emphasizes the equivalence with the KKT conditions.

It is important to see that a distinction between the active set \mathcal{A} and the inactive set \mathcal{I} is implicitly contained in the complementarity function C_j : the \max -function is non-smooth and thus consists of two different solution branches. In other words, C_j provides a certain regularization of the non-smooth decision between each slave node being currently active or inactive, yet without

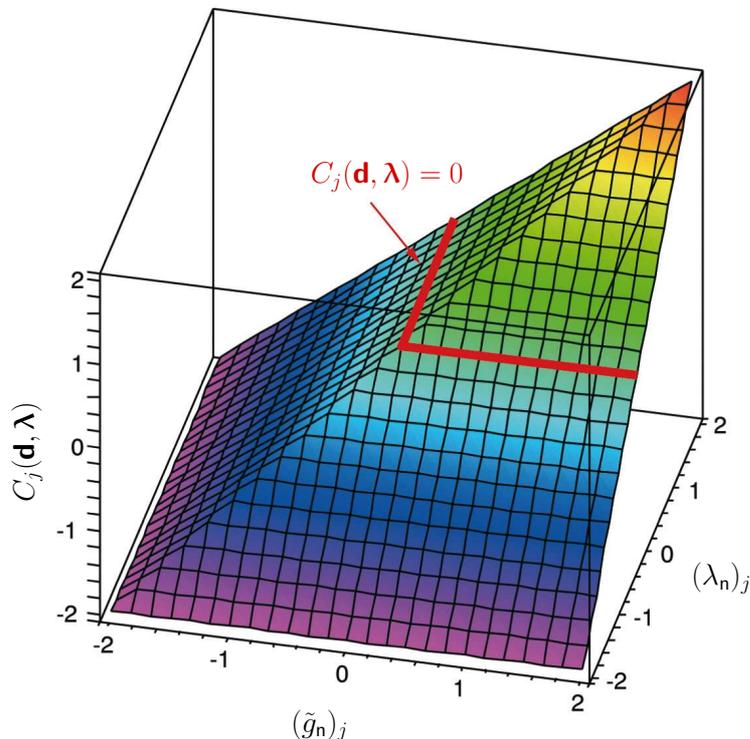


Figure 4.4: Exemplary nodal NCP function $C_j(\mathbf{d}, \boldsymbol{\lambda})$ as a function of the nodal weighted gap $(\tilde{g}_n)_j$ and the normal part of the nodal Lagrange multiplier $(\lambda_n)_j$ for a complementarity parameter $c_n = 1$. The equivalence with the KKT conditions is indicated in red color.

introducing any additional approximation. Thus, the resulting PDASS contains derivative information on the sets themselves and allows for the application of a Newton–Raphson type solution scheme also for the nonlinearity stemming from contact. Consequently, all sources of nonlinearities, i.e. finite deformations, nonlinear material behavior and contact itself, can be treated within one single iterative scheme. While C_j is a continuous function, it is non-smooth and has no uniquely defined derivative at the positions $(\lambda_n)_j - c_n(\tilde{g}_n)_j = 0$. Yet, it is well-known from mathematical literature on constrained optimization that the *max*-function can be classified as so-called semi-smooth function, and therefore a semi-smooth (or generalized) Newton method can still be applied. The interested reader is referred to Hintermüller et al. [77] and Qi and Sun [135] for more detailed information on semi-smooth Newton methods, for example including a concise proof of their superlinear local convergence behavior. The actual linearization of the NCP function in (4.63) is based on the concept of generalized derivatives (e.g. the generalized derivative of the *max*-function) and will be presented in Section 4.5.1 along with the remaining parts of the global solution algorithm.

It should be pointed out that the complementarity parameter c_n represents a purely algorithmic parameter. Although quite some similarities appear at first sight, c_n is in stark contrast to a penalty parameter, because it does not influence the accuracy of results. Instead, the weak non-penetration condition in (4.61) will be satisfied exactly, as can be expected from a Lagrange

multiplier method. The choice of c_n only improves or deteriorates convergence of the resulting semi-smooth Newton method. In Hübner and Wohlmuth [80], c_n has been suggested to be chosen at the order of Young's modulus E of the contacting bodies to obtain optimal convergence. Numerical investigations for 2D and 3D mortar contact in Popp et al. [123, 124], though, have shown very little influence on semi-smooth Newton convergence along a very broad spectrum of values for c_n . Numerical tests in Section 4.7 will reveal that even for relatively large step sizes and fine contacting meshes, the correct active set is usually found after only a few Newton steps. Once the sets remain constant, of course, quadratic convergence is obtained due to the underlying consistent linearization presented in Section 4.2 and Appendices A.1 and A.2.

Examining the NCP function for frictionless contact in (4.63) in more detail, and bearing in mind the general remarks on Lagrange multiplier treatment of inequality constraints in Section 2.4.4, allows for an interesting and important observation: there exists an extremely close relationship between the proposed PDASS with its algorithmic realization as semi-smooth Newton method and the classical Augmented Lagrange method, see also the seminal paper by Alart and Curnier [3] in this context. Due to the fact that most references in contact literature usually already imply a numerical implementation based on Uzawa's algorithm when referring to the Augmented Lagrange approach, this is a regular source of confusion in computational contact mechanics. However, as explained at length in Section 2.4.4, the Augmented Lagrange method can also be formulated such that the Lagrange multipliers remain as additional unknowns, see (2.121) and (2.122). If this procedure is pursued, the resulting constraints are identical with the ones obtained from the NCP function in (4.63) and must be solved with a semi-smooth Newton method based on the concept of generalized derivatives. Thus, the PDASS with its NCP functions and the Augmented Lagrange method can be interpreted as two different interpretations of the same circumstance, namely that the non-smoothness of the KKT conditions requires some regularization for an efficient numerical treatment. Simply speaking, the Augmented Lagrange approach as discussed in Alart and Curnier [3] aims at a regularized variational formulation, the PDASS and NCP function concept applies at a later stage with a regularized constraint enforcement.

4.4.3 Extension to Coulomb friction

Again, no detailed derivation of the Coulomb friction case is given here, but the interested reader is instead referred to Gitterle [54], Gitterle et al. [55], Hübner et al. [84] and Wohlmuth [177] for all details on the semi-smooth Newton approach for frictional contact problems. However, at least the fundamental concepts of such an extended PDASS and its implementation based on generalized derivatives shall be outlined in the following. As a starting point, the discrete frictional sliding conditions according to Coulomb's law as introduced in (4.29) are repeated here. At each slave node $j \in \mathcal{S}$, with $j = 1, \dots, m^{(1)}$, this yields

$$\begin{aligned} \Phi_j &:= \|(\boldsymbol{\lambda}_\tau)_j\| - \mathfrak{F}|(\lambda_n)_j| \leq 0, \\ (\tilde{\mathbf{v}}_{\tau, \text{rel}})_j + \beta_j(\boldsymbol{\lambda}_\tau)_j &= \mathbf{0}, \quad \beta_j \geq 0, \quad \Phi_j \beta_j = 0, \quad \forall j \in \mathcal{S}. \end{aligned} \quad (4.64)$$

Thus, as compared with the frictionless scenario, the frictionless sliding condition $(\boldsymbol{\lambda}_\tau)_j = \mathbf{0}$ is replaced by another set of inequality constraints describing the frictional effects taking place in the tangential plane of each contact point. Again, an efficient PDASS based on the reformulation of (4.64) within a NCP function can be derived, which then builds the basis for a global solution

algorithm of semi-smooth Newton type. In addition to the active set decision, which is handled by the NCP function for the normal direction introduced above, now a further distinction of active nodes into the two sets of stick and slip nodes becomes necessary. Consequently, the so-called slip set $\mathcal{G} \subseteq \mathcal{A}$ defines a subset of the set of all active nodes \mathcal{A} , and the definition of the stick set $\mathcal{H} = \mathcal{A} \setminus \mathcal{G}$ is straightforward. For Coulomb friction, this distinction between stick or slip status is represented by the following NCP function:

$$\begin{aligned} \mathbf{D}_j(\mathbf{d}, \boldsymbol{\lambda}) = & \max [\mathfrak{F}((\lambda_n)_j - c_n(\tilde{g}_n)_j), \|(\boldsymbol{\lambda}_\tau)_j + c_\tau(\tilde{\mathbf{v}}_{\tau, \text{rel}})_j\|] (\boldsymbol{\lambda}_\tau)_j \\ & - \mathfrak{F} \max [0, (\lambda_n)_j - c_n(\tilde{g}_n)_j] ((\boldsymbol{\lambda}_\tau)_j + c_\tau(\tilde{\mathbf{v}}_{\tau, \text{rel}})_j) = \mathbf{0}, \quad c_n > 0, c_\tau > 0. \end{aligned} \quad (4.65)$$

Note that this function is formulated for the 3D case and thus constitutes a vector \mathbf{D}_j with two components representing the two tangential directions at a contact point. For two-dimensional frictional contact problems, expression (4.65) naturally reduces to a scalar NCP function. Again, it can be easily shown that the resulting equality constraint $\mathbf{D}_j = \mathbf{0}$ is *equivalent* to the complete set of Coulomb friction inequality conditions in (4.64), and that this equivalence holds for arbitrary positive values of the tangential and normal complementarity parameters c_τ and c_n , respectively. Without going into details, it is pointed out that c_τ plays an analogous role for the stick and slip decision as c_n plays for the active or inactive decision, i.e. the role of an algorithmic regularization parameter, see Gitterle et al. [55] for further details.

The distinction between the slip set \mathcal{G} and the stick set \mathcal{H} is again implicitly contained in the NCP function \mathbf{D}_j owing to the non-smoothness of the max-function. The resulting PDASS for frictional contact combines (4.63) and (4.65), and thus contains derivative information on the three subsets of all slave nodes, i.e. $\mathcal{I} \cup \mathcal{H} \cup \mathcal{G} = \mathcal{S}$. Therefore, a solution scheme of Newton–Raphson type can be applied also for the additional nonlinearity stemming from frictional sliding. Consequently, all sources of nonlinearities, i.e. finite deformations, material behavior as well as contact and friction themselves, can be treated within one single iterative scheme. The actual linearization of the NCP function for Coulomb friction in (4.65) is omitted here, but can be found in Gitterle [54] and Gitterle et al. [55]. For the sake of completeness, it should be mentioned that several alternative formulations of the frictional NCP function $\mathbf{D}_j = \mathbf{0}$ exist in the literature. These modified versions are all mathematically equivalent in the sense that they yield the same numerical solution, however their properties and thus the resulting semi-smooth Newton convergence behavior slightly differs from each other. A comprehensive overview in the context of small deformation mortar contact is given in Hübner [79] and Hübner et al. [84].

4.5 Solution methods

Now that the treatment of the contact typical inequality constraints has been explained, the focus of attention is returned to the actual nonlinear solution schemes for the fully (i.e. space and time) discretized problem formulation in (4.43)–(4.45). As for mortar meshtying, this system consists of $\text{ndof} + \text{nco}$ nonlinear discrete algebraic equations, where the number of constraints is $\text{nco} = \text{ndim} \cdot m^{(1)}$. While standard (smooth) Newton–Raphson methods were the method of choice for mesh tying problems in Section 3.4, the active set strategies now require a semi-smooth Newton approach as discussed in the last paragraph. Nevertheless, for frictionless contact this non-smoothness solely affects the contact constraints in normal direction in (4.44) or to be

more precise their reformulation as NCP function in (4.63). All remaining parts of the nonlinear system, i.e. both the discrete equilibrium of forces in (4.43) and the frictionless sliding conditions in (4.45) still show a smooth behavior.

The following derivations will first focus on the consistent linearization of (4.43), (4.45) and (4.63) and especially on the resulting semi-smooth Newton solution algorithm within each time step. In further preparation of the expressions for the final linear systems of equations, algebraic representations of all linearized quantities will be introduced. The remaining paragraphs are then on different formulations of the global matrix vector system depending on the employed constraint enforcement approach, where the dual Lagrange multiplier approach proposed in this thesis is obviously emphasized most. However, for reasons of comparison the standard Lagrange multiplier approach is presented, too. Finally, a short overview of the penalty-based treatment of contact constraints within a mortar finite element formulations as well as its Augmented Lagrange version formulation based on Uzawa's algorithm is given, too.

4.5.1 Linearization and semi-smooth Newton algorithm

As explained in Section 2.3.5, the Newton–Raphson method is based on a subsequent linearization of the residual, here defined by the discrete balance of linear momentum in (4.43) and the discrete contact constraints in (4.45) and (4.63). Each nonlinear solution step (iteration index i) then consists of solving the resulting linearized system of equations and applying an incremental update of the unknown displacements \mathbf{d}_{n+1} and Lagrange multipliers $\boldsymbol{\lambda}_{n+1}$ until a user-defined convergence criterion is met. Examining the residual in (4.43) in more detail, an important difference to the mesh tying case becomes apparent: the discrete contact operator $\mathbf{B}_{\text{co}}(\mathbf{d})$ defined in (4.23), and thus the contact forces $\mathbf{f}_{\text{co}}(\mathbf{d}, \boldsymbol{\lambda})$, depend nonlinearly on the displacements and yield additional contact stiffness blocks when being linearized, i.e.

$$\begin{aligned} [\mathbf{K}_{\text{effdyn}}(\mathbf{d}_{n+1}^i) + (1 - \alpha_f)\mathbf{K}_{\text{co}}(\mathbf{d}_{n+1}^i, \boldsymbol{\lambda}_{n+1}^i)] \Delta \mathbf{d}_{n+1}^{i+1} + (1 - \alpha_f)\mathbf{B}_{\text{co}}(\mathbf{d}_{n+1}^i)\boldsymbol{\lambda}_{n+1}^{i+1} = \\ = -\mathbf{r}_{\text{effdyn}}(\mathbf{d}_{n+1}^i) - \alpha_f \mathbf{B}_{\text{co}}(\mathbf{d}_n)\boldsymbol{\lambda}_n. \end{aligned} \quad (4.66)$$

Herein, the contact stiffness \mathbf{K}_{co} is defined as

$$\mathbf{K}_{\text{co}}(\mathbf{d}_{n+1}^i, \boldsymbol{\lambda}_{n+1}^i) = \left. \frac{\partial(\mathbf{B}_{\text{co}}(\mathbf{d}_{n+1})\boldsymbol{\lambda}_{n+1})}{\partial \mathbf{d}_{n+1}} \right| ^i. \quad (4.67)$$

Moreover, it should be pointed out that contact-related quantities from the last converged time step n appear on the right-hand side of (4.66) due to the employed generalized- α time integration in combination with a trapezoidal rule interpolation of the contact forces as defined in (4.46). Similar to the mesh tying case, the interface forces are still linear with respect to the discrete Lagrange multipliers. Consequently, it is possible to solve directly for the unknowns $\boldsymbol{\lambda}_{n+1}^{i+1}$ in each iteration step and no incremental formulation is needed for this variable.

As a preliminary step for the consistent linearization of the NCP function in (4.63), the generalized derivative of the semi-smooth max-function can be expressed as

$$f(x) = \max(a, x) \longrightarrow \Delta f(x) = \begin{cases} 0 & \text{if } x \leq a \\ 1 & \text{if } x > a \end{cases}. \quad (4.68)$$

Similar to (4.34), directional derivatives of time-discretized quantities within a nonlinear solution scheme, i.e. for time step index $n + 1$ and nonlinear iteration index i , are defined as

$$\Delta(\cdot)_{n+1}^i = \left. \frac{\partial(\cdot)_{n+1}}{\partial \mathbf{d}_{n+1}} \right| \Delta \mathbf{d}_{n+1}^{i+1}. \quad (4.69)$$

Re-introducing the time step index $n + 1$ into (4.63) and applying consistent linearization with respect to the discrete nodal displacements \mathbf{d}_{n+1} yields the following expression:

$$\begin{aligned} & \Delta(\mathbf{n}_j)_{n+1}^i (\boldsymbol{\lambda}_j)_{n+1}^i + (\mathbf{n}_j)_{n+1}^i (\boldsymbol{\lambda}_j)_{n+1}^{i+1} \\ & - \chi_A \left[\Delta(\mathbf{n}_j)_{n+1}^i (\boldsymbol{\lambda}_j)_{n+1}^i + (\mathbf{n}_j)_{n+1}^i \Delta(\boldsymbol{\lambda}_j)_{n+1}^i - c_n \Delta((\tilde{g}_n)_j)_{n+1}^i \right] = \\ & = -\max(0, ((\lambda_n)_j)_{n+1}^i - c_n ((\tilde{g}_n)_j)_{n+1}^i) \quad \forall j \in \mathcal{S}. \end{aligned} \quad (4.70)$$

Herein, $\Delta(\boldsymbol{\lambda}_j)_{n+1}^i$ is simply an increment of the discrete Lagrange multiplier and must not be confused with the directional derivative notation introduced in (4.69). Moreover, χ_A can be interpreted as a characteristic function for the active set \mathcal{A}_{n+1}^i in the current iteration step i . Thus, it separates (4.70) into an inactive branch, i.e. $\chi_A = 0$ if $((\lambda_n)_j)_{n+1}^i - c_n ((\tilde{g}_n)_j)_{n+1}^i \leq 0$ and an active branch, i.e. $\chi_A = 1$ if $((\lambda_n)_j)_{n+1}^i - c_n ((\tilde{g}_n)_j)_{n+1}^i > 0$. Consequently, one can define the following separation of the slave node set \mathcal{S} into inactive nodes \mathcal{I}_{n+1}^i and active nodes \mathcal{A}_{n+1}^i for the current iteration step:

$$\begin{aligned} \mathcal{I}_{n+1}^i & := \{j \in \mathcal{S} \mid ((\lambda_n)_j)_{n+1}^i - c_n ((\tilde{g}_n)_j)_{n+1}^i \leq 0\}, \\ \mathcal{A}_{n+1}^i & := \{j \in \mathcal{S} \mid ((\lambda_n)_j)_{n+1}^i - c_n ((\tilde{g}_n)_j)_{n+1}^i > 0\}, \end{aligned} \quad (4.71)$$

which is tantamount to an update formula to be applied after *each* semi-smooth Newton step in order to obtain \mathcal{I}_{n+1}^{i+1} and \mathcal{A}_{n+1}^{k+1} . Consideration of (4.70) and (4.71) results in the following, very compact notation of the semi-smooth Newton step for the nonlinear complementarity (NCP) function C_j representing the KKT conditions:

$$\Delta(\mathbf{n}_j)_{n+1}^i (\boldsymbol{\lambda}_j)_{n+1}^i + (\mathbf{n}_j)_{n+1}^i (\boldsymbol{\lambda}_j)_{n+1}^{i+1} = 0 \quad \forall j \in \mathcal{I}_{n+1}^i, \quad (4.72)$$

$$\Delta((\tilde{g}_n)_j)_{n+1}^i = -((\tilde{g}_n)_j)_{n+1}^i \quad \forall j \in \mathcal{A}_{n+1}^i. \quad (4.73)$$

Consistent linearization of the frictionless tangential contact constraint in (4.45) yields two scalar conditions in a 3D setting, i.e.

$$\Delta(\boldsymbol{\tau}_j^\xi)_{n+1}^i (\boldsymbol{\lambda}_j)_{n+1}^i + (\boldsymbol{\tau}_j^\xi)_{n+1}^i (\boldsymbol{\lambda}_j)_{n+1}^{i+1} = 0 \quad \forall j \in \mathcal{S}, \quad (4.74)$$

$$\Delta(\boldsymbol{\tau}_j^\eta)_{n+1}^i (\boldsymbol{\lambda}_j)_{n+1}^i + (\boldsymbol{\tau}_j^\eta)_{n+1}^i (\boldsymbol{\lambda}_j)_{n+1}^{i+1} = 0 \quad \forall j \in \mathcal{S}. \quad (4.75)$$

For nodes of the inactive set \mathcal{I}_{n+1}^i , the tangential contact constraints in (4.74) and the normal contact constraints in (4.72) can be merged, thus reducing to the following obvious condition:

$$(\boldsymbol{\lambda}_j)_{n+1}^{i+1} = \mathbf{0} \quad \forall j \in \mathcal{I}_{n+1}^i. \quad (4.76)$$

Finally, the nonlinear solution scheme for solving the fully discrete mortar contact formulation given above can be summarized. Repeatedly performing semi-smooth Newton steps (iteration index i), each to be solved for the primal-dual pair of discrete variables $(\Delta \mathbf{d}_{n+1}^{i+1}, \boldsymbol{\lambda}_{n+1}^{i+1})$, yields the following solution algorithm within the time step $[t_n, t_{n+1}]$:

Algorithm 2

1. Set $i = 0$ and initialize the solution $(\mathbf{d}_{n+1}^0, \boldsymbol{\lambda}_{n+1}^0)$
2. Initialize \mathcal{A}_{n+1}^0 and \mathcal{I}_{n+1}^0 such that $\mathcal{A}_{n+1}^0 \cup \mathcal{I}_{n+1}^0 = \mathcal{S}$ and $\mathcal{A}_{n+1}^0 \cap \mathcal{I}_{n+1}^0 = \emptyset$
3. Find the primal-dual pair $(\Delta \mathbf{d}_{n+1}^{i+1}, \boldsymbol{\lambda}_{n+1}^{i+1})$ by solving

$$\tilde{\mathbf{K}}_{\text{effdyn,co}} \Delta \mathbf{d}_{n+1}^{i+1} + (1 - \alpha_f) \mathbf{B}_{\text{co}}(\mathbf{d}_{n+1}^i) \boldsymbol{\lambda}_{n+1}^{i+1} = -\tilde{\mathbf{r}}_{\text{effdyn,co}}, \quad (4.77)$$

$$(\boldsymbol{\lambda}_j)_{n+1}^{i+1} = \mathbf{0} \quad \forall j \in \mathcal{I}_{n+1}^i, \quad (4.78)$$

$$\Delta((\tilde{g}_n)_j)_{n+1}^i = -((\tilde{g}_n)_j)_{n+1}^i \quad \forall j \in \mathcal{A}_{n+1}^i, \quad (4.79)$$

$$\Delta(\boldsymbol{\tau}_j^\xi)_{n+1}^i (\boldsymbol{\lambda}_j)_{n+1}^i + (\boldsymbol{\tau}_j^\xi)_{n+1}^i (\boldsymbol{\lambda}_j)_{n+1}^{i+1} = 0 \quad \forall j \in \mathcal{S}, \quad (4.80)$$

$$\Delta(\boldsymbol{\tau}_j^\eta)_{n+1}^i (\boldsymbol{\lambda}_j)_{n+1}^i + (\boldsymbol{\tau}_j^\eta)_{n+1}^i (\boldsymbol{\lambda}_j)_{n+1}^{i+1} = 0 \quad \forall j \in \mathcal{S}. \quad (4.81)$$

4. Update $\mathbf{d}_{n+1}^{i+1} = \mathbf{d}_{n+1}^i + \Delta \mathbf{d}_{n+1}^{i+1}$
5. Set \mathcal{A}_{n+1}^{i+1} and \mathcal{I}_{n+1}^{i+1} to

$$\begin{aligned} \mathcal{I}_{n+1}^{i+1} &:= \{j \in \mathcal{S} \mid ((\lambda_n)_j)_{n+1}^{i+1} - c_n((\tilde{g}_n)_j)_{n+1}^{i+1} \geq 0\}, \\ \mathcal{A}_{n+1}^{i+1} &:= \{j \in \mathcal{S} \mid ((\lambda_n)_j)_{n+1}^{i+1} - c_n((\tilde{g}_n)_j)_{n+1}^{i+1} < 0\}. \end{aligned} \quad (4.82)$$

6. If $\mathcal{A}_{n+1}^{i+1} = \mathcal{A}_{n+1}^i$, $\mathcal{I}_{n+1}^{i+1} = \mathcal{I}_{n+1}^i$ and $\|\mathbf{r}_{\text{tot}}\| \leq \epsilon_r$, then stop, else set $i := i + 1$ and go to step (3).

Herein, the following abbreviations have been introduced for notational simplicity:

$$\tilde{\mathbf{K}}_{\text{effdyn,co}} = \mathbf{K}_{\text{effdyn}}(\mathbf{d}_{n+1}^i) + (1 - \alpha_f) \mathbf{K}_{\text{co}}(\mathbf{d}_{n+1}^i, \boldsymbol{\lambda}_{n+1}^i), \quad (4.83)$$

$$\tilde{\mathbf{r}}_{\text{effdyn,co}} = \mathbf{r}_{\text{effdyn}}(\mathbf{d}_{n+1}^i) + \alpha_f \mathbf{B}_{\text{co}}(\mathbf{d}_n) \boldsymbol{\lambda}_n. \quad (4.84)$$

Moreover, the variable ϵ_r denotes an absolute Newton convergence tolerance for the L^2 -norm of the total residual vector \mathbf{r}_{tot} , which comprises the force residual and the residual of the contact constraints (4.78)–(4.81). All types of nonlinearities including the search for the correct active set are resolved within one single nonlinear solution scheme, with the sets \mathcal{I}_{n+1}^i and \mathcal{A}_{n+1}^i being updated after each semi-smooth Newton step.

The convergence behavior of the resulting solution scheme is very good, which will be demonstrated by various numerical examples in Section 4.7. As long as the correct active set is not found, and thus the contact typical non-smoothness is not yet resolved, locally superlinear convergence rates are obtained, see e.g. Hintermüller et al. [77]. Once the sets are fixed, the nonlinear iteration scheme reduces to a standard (smooth) Newton–Raphson method, and thus even locally quadratic convergence rates are achieved in the limit owing to the underlying consistent linearization. While not discussed here, similar observations can also be made for frictional contact according to Coulomb’s law and the associated search for the correct stick and slip sets, see e.g. Gitterle [54], Gitterle et al. [55] and Hübner et al. [84].

4.5.2 Algebraic representation

In this section, an algebraic representation of the linearized system to be solved within each semi-smooth Newton step is derived and globally assembled matrix notations for the directional derivatives in (4.77)–(4.81) are provided. With the assembly procedure itself being rather straightforward in finite element methods, only abstract definitions of the individual matrix blocks are given here. Details on the linearization of the contained quantities can be found in Sections 4.2.3 and 4.2.3 and also in Appendix A.

First, the contact stiffness matrix \mathbf{K}_{co} has already been introduced in (4.67) as linearization of the discrete contact force vector \mathbf{f}_{co} , and can further be split into sub-blocks as follows:

$$\Delta \mathbf{f}_{\text{co}}(\mathbf{d}_{n+1}^i, \boldsymbol{\lambda}_{n+1}^i) = \underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_{\text{co},\mathcal{M}\mathcal{M}} & \mathbf{K}_{\text{co},\mathcal{M}\mathcal{S}} \\ \mathbf{0} & \mathbf{K}_{\text{co},\mathcal{S}\mathcal{M}} & \mathbf{K}_{\text{co},\mathcal{S}\mathcal{S}} \end{bmatrix}}_{\mathbf{K}_{\text{co}}(\mathbf{d}_{n+1}^i, \boldsymbol{\lambda}_{n+1}^i)} \begin{bmatrix} \Delta \mathbf{d}_{n+1,\mathcal{N}}^{i+1} \\ \Delta \mathbf{d}_{n+1,\mathcal{M}}^{i+1} \\ \Delta \mathbf{d}_{n+1,\mathcal{S}}^{i+1} \end{bmatrix}. \quad (4.85)$$

Herein, the contact stiffness matrix contains the directional derivatives of both mortar matrices \mathbf{D} and \mathbf{M} together with the current Lagrange multiplier values $\boldsymbol{\lambda}_{n+1}^i$, see (4.67). The global vector of displacement increments $\Delta \mathbf{d}$ has been split into three subgroups \mathcal{N} , \mathcal{M} and \mathcal{S} as usual. Considering the non-penetration constraints for all active nodes in (4.79) yields

$$\mathbf{A} \Delta((\tilde{g}_n)_j)_{n+1}^i = \mathbf{N}_{\mathcal{S}} \Delta \mathbf{d}_{n+1,\mathcal{S}}^{i+1} + \mathbf{N}_{\mathcal{M}} \Delta \mathbf{d}_{n+1,\mathcal{M}}^{i+1}, \quad (4.86)$$

where matrices $\mathbf{N}_{\mathcal{S}} \in \mathbb{R}^{n^{(a)} \times \text{ndim} \cdot n^{(1)}}$ and $\mathbf{N}_{\mathcal{M}} \in \mathbb{R}^{n^{(a)} \times \text{ndim} \cdot n^{(2)}}$ represent the assembly of all directional derivatives covered in $\Delta(\tilde{g}_n)_j$ (see Section 4.2.4) and $n^{(a)}$ is the current number of active slave nodes. Finally, \mathbf{A} denotes the standard finite element assembly operator. An algebraic formulation for the frictionless sliding conditions in the tangential directions defined by $\boldsymbol{\tau}_j^\xi$ (for 2D and 3D problems) and $\boldsymbol{\tau}_j^\eta$ (only for 3D problems) is given as

$$\mathbf{A} \Delta(\boldsymbol{\tau}_j^\xi)_{n+1}^i (\boldsymbol{\lambda}_j)_{n+1}^i + (\boldsymbol{\tau}_j^\xi)_{n+1}^i (\boldsymbol{\lambda}_j)_{n+1}^{i+1} = \mathbf{F}_{\mathcal{S}}^\xi \Delta \mathbf{d}_{n+1,\mathcal{S}}^{i+1} + \mathbf{T}_{\mathcal{A}}^\xi \boldsymbol{\lambda}_{n+1,\mathcal{A}}^{i+1}, \quad (4.87)$$

$$\mathbf{A} \Delta(\boldsymbol{\tau}_j^\eta)_{n+1}^i (\boldsymbol{\lambda}_j)_{n+1}^i + (\boldsymbol{\tau}_j^\eta)_{n+1}^i (\boldsymbol{\lambda}_j)_{n+1}^{i+1} = \mathbf{F}_{\mathcal{S}}^\eta \Delta \mathbf{d}_{n+1,\mathcal{S}}^{i+1} + \mathbf{T}_{\mathcal{A}}^\eta \boldsymbol{\lambda}_{n+1,\mathcal{A}}^{i+1}, \quad (4.88)$$

where matrices $\mathbf{F}_{\mathcal{S}}^\xi, \mathbf{F}_{\mathcal{S}}^\eta \in \mathbb{R}^{n^{(a)} \times \text{ndim} \cdot n^{(1)}}$ are assembled forms of the tangent vector directional derivatives (see Appendix A) together with the current Lagrange multiplier values. The two matrices $\mathbf{T}_{\mathcal{A}}^\xi, \mathbf{T}_{\mathcal{A}}^\eta \in \mathbb{R}^{n^{(a)} \times \text{ndim} \cdot n^{(a)}}$ are defined as

$$\mathbf{T}_{\mathcal{A}}^{\xi/\eta} = \begin{bmatrix} \ddots & & 0 & & 0 \\ 0 & \{(\boldsymbol{\tau}_j^{\xi/\eta})_{n+1}^i\}^\top & & & 0 \\ 0 & & 0 & & \ddots \end{bmatrix} \quad \text{with } j \in \mathcal{A}_{n+1}^i. \quad (4.89)$$

Of course, (4.87) and (4.88) are easily replaced by one single algebraic formulation, when concatenating the blocks $\mathbf{F}_{\mathcal{S}}^\xi, \mathbf{F}_{\mathcal{S}}^\eta$ and $\mathbf{T}_{\mathcal{A}}^\xi, \mathbf{T}_{\mathcal{A}}^\eta$ to eventually form $\mathbf{F}_{\mathcal{S}} \in \mathbb{R}^{(\text{ndim}-1) \cdot n^{(a)} \times \text{ndim} \cdot n^{(1)}}$ and

$\mathbf{T}_{\mathcal{A}} \in \mathbb{R}^{(\text{ndim}-1) \cdot n^{(a)} \times \text{ndim} \cdot n^{(a)}}$, respectively. Note, that only the first tangent direction ξ and thus only the corresponding matrix blocks $\mathbf{F}_{\mathcal{S}}^{\xi}$ and $\mathbf{T}_{\mathcal{A}}^{\xi}$ exist in a two-dimensional setting. Finally, the first and second mortar matrices \mathbf{D} and \mathbf{M} are decomposed into active and inactive blocks, i.e.

$$\mathbf{D} = \begin{bmatrix} \mathbf{D}_{II} & \mathbf{D}_{IA} \\ \mathbf{D}_{AI} & \mathbf{D}_{AA} \end{bmatrix}, \quad \mathbf{M} = \begin{bmatrix} \mathbf{M}_{\mathcal{I}} \\ \mathbf{M}_{\mathcal{A}} \end{bmatrix}. \quad (4.90)$$

4.5.3 Standard Lagrange multiplier approach

Reformulating (4.77)–(4.81) in matrix-vector notation for a standard Lagrange multiplier approach leads to a typical saddle point type of system. However, in the case of frictionless unilateral contact considered here, the resulting system is no longer symmetric as has been the case for mesh tying in Section 3.4.2. This is due to the use of a vector-valued Lagrange multiplier throughout this thesis. It is important to point out that a scalar Lagrange multiplier is theoretically sufficient for constraint enforcement in the frictionless case, and that a symmetric formulation could indeed still be obtained in that case. However, taking into account that frictional constraints (and other interface effects) eventually lead to non-symmetric linear systems anyway, and considering the fact that relatively efficient non-symmetric solvers exist nowadays, symmetry of linear systems is not considered most important here.

Based on the algebraic representations (4.85)–(4.90) established in the previous paragraph, the final system to be solved within each semi-smooth Newton step can be expressed as follows:

$$\begin{bmatrix} \tilde{\mathbf{K}}_{\mathcal{N}\mathcal{N}} & \tilde{\mathbf{K}}_{\mathcal{N}\mathcal{M}} & \tilde{\mathbf{K}}_{\mathcal{N}\mathcal{I}} & \tilde{\mathbf{K}}_{\mathcal{N}\mathcal{A}} & \mathbf{0} & \mathbf{0} \\ \tilde{\mathbf{K}}_{\mathcal{M}\mathcal{N}} & \tilde{\mathbf{K}}_{\mathcal{M}\mathcal{M}} & \tilde{\mathbf{K}}_{\mathcal{M}\mathcal{I}} & \tilde{\mathbf{K}}_{\mathcal{M}\mathcal{A}} & -a\mathbf{M}_{\mathcal{I}}^{\top} & -a\mathbf{M}_{\mathcal{A}}^{\top} \\ \tilde{\mathbf{K}}_{\mathcal{I}\mathcal{N}} & \tilde{\mathbf{K}}_{\mathcal{I}\mathcal{M}} & \tilde{\mathbf{K}}_{\mathcal{I}\mathcal{I}} & \tilde{\mathbf{K}}_{\mathcal{I}\mathcal{A}} & a\mathbf{D}_{\mathcal{I}\mathcal{I}}^{\top} & a\mathbf{D}_{\mathcal{I}\mathcal{A}}^{\top} \\ \tilde{\mathbf{K}}_{\mathcal{A}\mathcal{N}} & \tilde{\mathbf{K}}_{\mathcal{A}\mathcal{M}} & \tilde{\mathbf{K}}_{\mathcal{A}\mathcal{I}} & \tilde{\mathbf{K}}_{\mathcal{A}\mathcal{A}} & a\mathbf{D}_{\mathcal{A}\mathcal{I}}^{\top} & a\mathbf{D}_{\mathcal{A}\mathcal{A}}^{\top} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_{\mathcal{I}} & \mathbf{0} \\ \mathbf{0} & \mathbf{N}_{\mathcal{M}} & \mathbf{N}_{\mathcal{I}} & \mathbf{N}_{\mathcal{A}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{F}_{\mathcal{I}} & \mathbf{F}_{\mathcal{A}} & \mathbf{0} & \mathbf{T}_{\mathcal{A}} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{d}_{n+1, \mathcal{N}} \\ \Delta \mathbf{d}_{n+1, \mathcal{M}} \\ \Delta \mathbf{d}_{n+1, \mathcal{I}} \\ \Delta \mathbf{d}_{n+1, \mathcal{A}} \\ \lambda_{n+1, \mathcal{I}} \\ \lambda_{n+1, \mathcal{A}} \end{bmatrix} = - \begin{bmatrix} \tilde{\mathbf{r}}_{\mathcal{N}} \\ \tilde{\mathbf{r}}_{\mathcal{M}} \\ \tilde{\mathbf{r}}_{\mathcal{I}} \\ \tilde{\mathbf{r}}_{\mathcal{A}} \\ \mathbf{0} \\ \tilde{\mathbf{g}}_{\mathcal{A}} \\ \mathbf{0} \end{bmatrix}. \quad (4.91)$$

Herein, the scalar $a := 1 - \alpha_f$ abbreviates the weighting factor introduced by generalized- α time integration. Moreover, the nonlinear iteration index i as well as the subscript $\cdot_{\text{effdyn,co}}$ of the residual vector $\tilde{\mathbf{r}}_{\text{effdyn,co}}$ given in (4.83) and the effective stiffness matrix $\tilde{\mathbf{K}}_{\text{effdyn,co}}$ defined in (4.84) have been omitted for the ease of notation. The discrete vector $\tilde{\mathbf{g}}_{\mathcal{A}} \in \mathbb{R}^{n^{(a)}}$ contains all weighted gap values associated with active nodes, i.e. it represents the assembled right-hand side of (4.79). Examining the linear system (4.91) in more detail, one can identify the first four rows as linearized algebraic form of the discrete force equilibrium in (4.43). The fifth row represents the contact constraint condition for nodes of the inactive set \mathcal{I}_{n+1}^i , where $\mathbf{I}_{\mathcal{I}} \in \mathbb{R}^{\text{ndim} \cdot (n^{(1)} - n^{(a)})}$ is an identity matrix. For nodes of the active set \mathcal{A}_{n+1}^i , however, one has to distinguish between normal contact constraints (non-penetration) in the sixth row and tangential constraints (frictionless sliding) in the seventh row of (4.91). The given formulation could be extended to the case of frictional contact based on Coulomb's law when replacing the seventh row by suitable frictional constraints, see Gitterle [54], Gitterle et al. [55] and Hübner et al. [84].

Note, that the system block matrix in (4.91) is of course a square matrix, although the given algebraic representation consists of seven rows and only six columns. This is due to the fact that the normal and tangential contributions of the active node constraints in the last two rows

are written down separately. Furthermore, the typical saddle point structure with a zero diagonal block in the sixth row can easily be recognized. As explained in detail for mesh tying in Section 3.4.2, such saddle point systems are problematic for efficient iterative linear solvers and associated preconditioning techniques. However, it is still possible to solve these systems using either direct solvers or quite well-established iterative methods tailored for coupled systems of equations including zero diagonal blocks, see Section 3.4.2.

4.5.4 Dual Lagrange multiplier approach

Again, as has been the case for mesh tying, the *dual* Lagrange multiplier approach can be beneficially exploited to simplify the final linear system of equations. Based on the biorthogonality condition (4.52), the mortar matrix \mathbf{D} reduces to a diagonal matrix, i.e.

$$\mathbf{D} = \begin{bmatrix} \mathbf{D}_{\mathcal{I}\mathcal{I}} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_{\mathcal{A}\mathcal{A}} \end{bmatrix}, \quad (4.92)$$

and thus its inversion as well as the condensation of the discrete Lagrange multiplier degrees of freedom become trivial. Consequently, the undesirable saddle point structure of (4.91) can be removed for the dual Lagrange multiplier approach, while the saddle point formulation nonetheless remains equally valid.

In a first step, the Lagrange multipliers $\lambda_{n+1,\mathcal{I}}$ associated with inactive slave nodes are easily condensed by simply extracting the identity $\lambda_{n+1,\mathcal{I}} = \mathbf{0}$ from the fifth row of (4.91). This basically removes the fifth row and the fifth column of the original saddle point system. More importantly, based on the fourth row of (4.91), the Lagrange multipliers $\lambda_{n+1,\mathcal{A}}$ associated with active slave nodes can be expressed as

$$\lambda_{n+1,\mathcal{A}} = \frac{1}{a} \mathbf{D}_{\mathcal{A}\mathcal{A}}^{-\top} \left(-\tilde{\mathbf{r}}_{\mathcal{A}} - \tilde{\mathbf{K}}_{\mathcal{A}\mathcal{N}} \Delta \mathbf{d}_{n+1,\mathcal{N}} - \tilde{\mathbf{K}}_{\mathcal{A}\mathcal{M}} \Delta \mathbf{d}_{n+1,\mathcal{M}} - \tilde{\mathbf{K}}_{\mathcal{A}\mathcal{I}} \Delta \mathbf{d}_{n+1,\mathcal{I}} - \tilde{\mathbf{K}}_{\mathcal{A}\mathcal{A}} \Delta \mathbf{d}_{n+1,\mathcal{A}} \right). \quad (4.93)$$

Again, the weighting factor of generalized- α time integration is abbreviated as $a := 1 - \alpha_f$. As compared with the mesh tying case in Section 3.4.3, it becomes apparent that only the discrete Lagrange multipliers λ_{n+1} , but not the slave side displacement increments $\Delta \mathbf{d}_{n+1,\mathcal{S}}$ can be condensed for unilateral contact. Thus, the final condensed system is in a kind of “hybrid” state, meaning that the discrete Lagrange multiplier degrees of freedom are condensed, but the (active) constraint equations themselves remain unchanged within the system. Nevertheless, and most importantly for numerical efficiency, the typical saddle point structure with its zero diagonal entries is completely removed. Similar to Section 3.4.3, the active part of the mortar projection operator $\mathbf{P} = \mathbf{D}^{-1} \mathbf{M}$ can be defined as

$$\mathbf{P}_{\mathcal{A}} = \mathbf{D}_{\mathcal{A}\mathcal{A}}^{-1} \mathbf{M}_{\mathcal{A}}. \quad (4.94)$$

Inserting (4.93) into the second and seventh row of (4.91) yields

$$\begin{bmatrix} \tilde{\mathbf{K}}_{\mathcal{N}\mathcal{N}} & \tilde{\mathbf{K}}_{\mathcal{N}\mathcal{M}} & \tilde{\mathbf{K}}_{\mathcal{N}\mathcal{I}} & \tilde{\mathbf{K}}_{\mathcal{N}\mathcal{A}} \\ \tilde{\mathbf{K}}_{\mathcal{M}\mathcal{N}} + \mathbf{P}_{\mathcal{A}}^{\mathbf{T}}\tilde{\mathbf{K}}_{\mathcal{A}\mathcal{N}} & \tilde{\mathbf{K}}_{\mathcal{M}\mathcal{M}} + \mathbf{P}_{\mathcal{A}}^{\mathbf{T}}\tilde{\mathbf{K}}_{\mathcal{A}\mathcal{M}} & \tilde{\mathbf{K}}_{\mathcal{M}\mathcal{I}} + \mathbf{P}_{\mathcal{A}}^{\mathbf{T}}\tilde{\mathbf{K}}_{\mathcal{A}\mathcal{I}} & \tilde{\mathbf{K}}_{\mathcal{M}\mathcal{A}} + \mathbf{P}_{\mathcal{A}}^{\mathbf{T}}\tilde{\mathbf{K}}_{\mathcal{A}\mathcal{A}} \\ \tilde{\mathbf{K}}_{\mathcal{I}\mathcal{N}} & \tilde{\mathbf{K}}_{\mathcal{I}\mathcal{M}} & \tilde{\mathbf{K}}_{\mathcal{I}\mathcal{I}} & \tilde{\mathbf{K}}_{\mathcal{I}\mathcal{A}} \\ \mathbf{0} & \mathbf{N}_{\mathcal{M}} & \mathbf{N}_{\mathcal{I}} & \mathbf{N}_{\mathcal{A}} \\ a\mathbf{T}_{\mathcal{A}}\mathbf{D}_{\mathcal{A}\mathcal{A}}^{-1}\tilde{\mathbf{K}}_{\mathcal{A}\mathcal{N}} & a\mathbf{T}_{\mathcal{A}}\mathbf{D}_{\mathcal{A}\mathcal{A}}^{-1}\tilde{\mathbf{K}}_{\mathcal{A}\mathcal{M}} & a\mathbf{T}_{\mathcal{A}}\mathbf{D}_{\mathcal{A}\mathcal{A}}^{-1}\tilde{\mathbf{K}}_{\mathcal{A}\mathcal{I}} - \mathbf{F}_{\mathcal{I}} & a\mathbf{T}_{\mathcal{A}}\mathbf{D}_{\mathcal{A}\mathcal{A}}^{-1}\tilde{\mathbf{K}}_{\mathcal{A}\mathcal{A}} - \mathbf{F}_{\mathcal{A}} \end{bmatrix} \cdot \begin{bmatrix} \Delta \mathbf{d}_{n+1,\mathcal{N}} \\ \Delta \mathbf{d}_{n+1,\mathcal{M}} \\ \Delta \mathbf{d}_{n+1,\mathcal{I}} \\ \Delta \mathbf{d}_{n+1,\mathcal{A}} \end{bmatrix} = - \begin{bmatrix} \tilde{\mathbf{r}}_{\mathcal{N}} \\ \tilde{\mathbf{r}}_{\mathcal{M}} + \mathbf{P}_{\mathcal{A}}^{\mathbf{T}}\tilde{\mathbf{r}}_{\mathcal{A}} \\ \tilde{\mathbf{r}}_{\mathcal{I}} \\ \tilde{\mathbf{g}}_{\mathcal{A}} \\ a\mathbf{T}_{\mathcal{A}}\mathbf{D}_{\mathcal{A}\mathcal{A}}^{-1}\tilde{\mathbf{r}}_{\mathcal{A}} \end{bmatrix}. \quad (4.95)$$

In the following, some important numerical properties of the condensed linear system based on dual Lagrange multiplier interpolation are summarized. Firstly, the formulation given in (4.95) contains only displacement degrees of freedom and does not suffer from an increased or possibly varying system size due to discrete Lagrange multipliers. While inevitable for standard Lagrange multiplier interpolation, the undesirable saddle point structure of (4.91) with its typical zero diagonal block has successfully been removed. This allows for a more straightforward application of state-of-the-art iterative solution techniques, however this topic is beyond the scope of the given thesis. Some promising investigations in this direction based on geometric and algebraic multigrid methods can exemplarily be found in Brunssen et al. [20], Krause [98] and Wohlmuth and Krause [178]. Again, the interested reader is referred to Gitterle [54], Gitterle et al. [55] and Hieber et al. [84] for an extension of the given algorithms to Coulomb friction.

Finally, it should be mentioned that the discrete Lagrange multipliers, and thus their physical interpretation as contact tractions, are recovered from the displacement solution in a variationally consistent way. This recovery can be performed as a pure postprocessing step at the end of each time interval based on the relation given in (4.93). This section is concluded with a brief remark on standard Lagrange multiplier interpolation, which, at least theoretically and for small problem sizes, also allows for the condensation procedure outlined above. In practice however, the fact that \mathbf{D}^{-1} is densely populated for standard Lagrange multipliers makes such an approach unaffordable from both the numerical efficiency and the memory consumption perspective. Special methods based on discrete null spaces have been suggested (see e.g. Hesch and Betsch [71] for the 2D case with first-order interpolation) to circumvent saddle point systems without resorting to dual Lagrange multiplier interpolation. However, in the author's opinion, such methods are neither computationally efficient nor applicable to general contact problems, and thus cannot be considered a genuine alternative to the dual mortar approach proposed in this thesis.

4.5.5 Penalty approach

Constraint enforcement using the penalty approach or an Augmented Lagrange version based on Uzawa's algorithm is very popular for problems of constrained optimization. For computational contact mechanics (i.e. unilateral contact) this holds even more than for the mesh tying case considered in Section 3.4.4. As a characteristic feature of the penalty approach, the occurrent constraint conditions are usually relaxed via regularization of the underlying physical models,

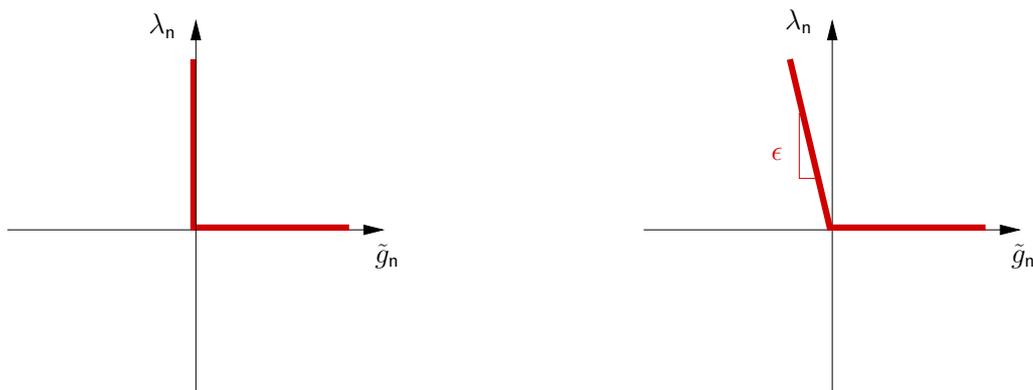


Figure 4.5: Original discrete non-penetration condition (left) and its regularized version based on a penalty approach with the penalty parameter ϵ (right).

though at the price of a reduced solution accuracy. Consequently, the additional degrees of freedom needed in a Lagrange multiplier approach disappear, and the problem can be dealt within a purely displacement-based formulation instead. While the focus is on (dual) Lagrange multiplier methods throughout this thesis, the most important aspects of penalty-based contact treatment will at least be outlined in the following for comparison purposes. Again, only the frictionless case is considered for the sake of simplicity.

As a starting point, the KKT conditions resulting from mortar finite element discretization in space are reconsidered, see (4.27). The originally non-uniquely defined relationship between contact pressure λ_n and weighted gap \tilde{g}_n at each slave node is regularized by introducing a penalty parameter $0 < \epsilon < \infty$ as

$$(\lambda_n)_j = \epsilon \langle -(\tilde{g}_n)_j \rangle. \quad (4.96)$$

The effect of penalty regularization on the non-penetration condition is visualized in Figure 4.5. It can clearly be seen that the finite value of the penalty parameter allows for a certain (unphysical) violation of the non-penetration condition, but on the other hand generates a uniquely defined force-penetration relationship in the first place. The inequality nature of the KKT conditions is taken care of by the so-called Macauley bracket already introduced in (2.118). Thus, a standard penalty formulation of unilateral contact strictly speaking remains non-smooth with respect to the active set decision, with the generalized derivative of the Macauley bracket bearing some similarities with semi-smooth Newton methods as described in Section 4.4.

This eventually leads to the following definition of the semi-discrete (i.e. spatially discretized but time continuous) contact forces $\mathbf{f}_{\text{co}}(\mathbf{d})$, where the Lagrange multipliers λ are now no longer independent unknowns, but can rather be derived from the discrete nodal displacements according to (4.96), viz.

$$\mathbf{f}_{\text{co}}(\mathbf{d}) = \mathbf{B}_{\text{co}}(\mathbf{d})^T \lambda(\mathbf{d}), \quad (4.97)$$

where

$$\lambda(\mathbf{d}) = \mathbf{A} \lambda_j = \mathbf{A} \epsilon \langle -(\tilde{g}_n)_j \rangle \mathbf{n}_j. \quad (4.98)$$

Herein, \mathbf{n}_j represents the discrete nodal normal vector at slave node j , possibly determined based on nodal averaging (cf. Section 3.2). All aspects of time discretization, e.g. the choice whether

the discrete contact forces are interpolated based on a trapezoidal rule or a midpoint rule, remain unaffected by penalty regularization. Thus, the fully discretized problem formulation, given for a Lagrange multiplier formulation in (4.43)–(4.45), simply reduces to

$$\mathbf{M}\mathbf{a}_{n+1-\alpha_m} + \mathbf{C}\mathbf{v}_{n+1-\alpha_f} + \mathbf{f}_{\text{int}}(\mathbf{d}_{n+1-\alpha_f}) + \mathbf{f}_{\text{co}}(\mathbf{d}_{n+1-\alpha_f}) - \mathbf{f}_{\text{ext},n+1-\alpha_f} = \mathbf{0}. \quad (4.99)$$

Throughout this thesis, a trapezoidal rule type of interpolation is applied for the contact forces within the generalized- α method. For the penalty approach considered here, this yields

$$\mathbf{f}_{\text{co}}(\mathbf{d}_{n+1-\alpha_f}) = (1 - \alpha_f)\mathbf{f}_{\text{co}}(\mathbf{d}_{n+1}) + \alpha_f\mathbf{f}_{\text{co}}(\mathbf{d}_n). \quad (4.100)$$

Consequently, consistent linearization of (4.99) yields the following linearized system to be solved within each Newton–Raphson step (iteration index i):

$$\begin{aligned} & [\mathbf{K}_{\text{effdyn}}(\mathbf{d}_{n+1}^i) + (1 - \alpha_f)\mathbf{K}_{\text{co,P}}(\mathbf{d}_{n+1}^i)] \Delta \mathbf{d}_{n+1}^{i+1} = \\ & = -\mathbf{r}_{\text{effdyn}}(\mathbf{d}_{n+1}^i) - (1 - \alpha_f)\mathbf{f}_{\text{co}}(\mathbf{d}_{n+1}^i) - \alpha_f\mathbf{f}_{\text{co}}(\mathbf{d}_n). \end{aligned} \quad (4.101)$$

where the contact stiffness matrix is given as

$$\mathbf{K}_{\text{co,P}}(\mathbf{d}_{n+1}^i) = \left. \frac{\partial \mathbf{f}_{\text{co}}(\mathbf{d}_{n+1})}{\partial \mathbf{d}_{n+1}} \right|_i. \quad (4.102)$$

While details are omitted here, it is quite obvious that the individual contributions to (4.102) bear strong similarities to the contact stiffness contributions in a Lagrange multiplier approach. Basically, all ingredients for the implementation of (4.102), e.g. linearizations of the mortar matrices \mathbf{D} and \mathbf{M} , the weighted gaps $(\tilde{g}_n)_j$ or the averaged nodal normal vectors \mathbf{n}_j can be found in Section 4.2 and in Appendix A. For a comprehensive treatment of penalty regularized mortar finite element methods for computational contact mechanics, the interested reader is exemplarily referred to Laursen [107], Puso and Laursen [132, 133], Puso et al. [134] and Yang et al. [189].

As mentioned before, the main advantage of the penalty approach for a mortar finite element discretization lies in its purely displacement-based formulation without any additional Lagrange multiplier degrees of freedom. Above all, this makes state-of-the-art iterative solution techniques applicable in a relatively easy way. Nevertheless, the well-known deficiencies of penalty methods must not be dismissed: first, the dilemma of solution accuracy versus acceptable conditioning of the resulting linear systems cannot be resolved. Moreover, although adaptive penalty methods have been proposed, the choice of a user-defined penalty parameter remains largely heuristic and problem-dependent. An Augmented Lagrange version based on Uzawa’s algorithm and using the classical penalty approach as a kernel can - at least to some extent - remedy the mentioned shortcomings. However, this comes at the price of increased computational costs due to an additional nonlinear iteration loop (augmentation), typically only showing a linear convergence rate. While there exist many acceleration procedures for the Uzawa algorithm in the literature, which also guarantee superlinear convergence rates, see e.g. Zavarise and Wriggers [192], generalized or semi-smooth Newton methods as proposed in Alart and Curnier [3] and throughout this thesis arguably provide contact solution algorithms with a superior numerical efficiency. No nested iteration loops are necessary here, but all nonlinearities - including the exact enforcement of contact constraints - are efficiently dealt with within one Newton–Raphson scheme, featuring a quadratic convergence rate in the limit. Further applying the presented concept of dual Lagrange multiplier interpolation allows for eliminating the commonly cited drawbacks of Lagrange multiplier methods, such as increased system size and saddle point system characteristics.

4.6 Aspects of implementation and parallel computing

The discussion of implementation and software design issues associated with the proposed mortar finite element methods is not in the focus of this thesis, but rather the underlying mathematical and mechanical foundations as well as their transfer into numerical algorithms. Nevertheless, the following section at least aims at addressing and outlining some of the most important aspects of the devised implementation. Thus, this interlude is supposed to give the reader a more complete picture of the developed methods and to provide references in the fields of parallel programming and software design for computational contact mechanics, see also Popp et al. [126] for details.

The upcoming paragraph first puts an emphasis on inter-processor redistribution and dynamic load balancing strategies for mortar methods, which in turn requires a brief introduction to the employed paradigm of parallel programming. Furthermore, the basic concepts of efficient parallel search algorithms for two body contact, self contact and multiple body systems (e.g. agglomerations of elastic particles) will be presented. All explanations exclusively refer to implementations devised in the context of the present thesis, and subsequently integrated into the in-house finite element software package BACI (cf. Wall and Gee [171]), developed jointly at the Institute for Computational Mechanics at Technische Universität München.

4.6.1 Parallel redistribution and dynamic load balancing

The mortar-based mesh tying and contact algorithms developed throughout this thesis are designed for the use on large interconnected computer systems (clusters) with many central processing units (CPUs) and a distributed main memory. Being able to efficiently run large simulations in parallel requires strategies for the partitioning and parallel distribution of the problem data, i.e. finite element meshes (consisting of nodes and elements) as well as global vectors and matrices, into several independent processes, each assigned to a corresponding *processor*. For the sake of simplicity, the term processor refers to an independent processing unit throughout this thesis without implying any specific hardware configuration (such as a single-core or multi-core architecture). Within BACI, this so-called domain (or data) decomposition functionality is provided by the third-party library ParMETIS, see e.g. Karypis and Kumar [93].

An example of such decompositions is visualized in Figure 4.6 for a simple partitioning including only two processors, see also Gee [47]. It can be seen that each node in the mesh is uniquely assigned to one specific processor, and the same holds true for the elements. In addition, some nodes and elements at the transition between different processors must be stored redundantly within all adjacent processors. Therefore, this type of partitioning is commonly denoted as *overlapping* decomposition. For the methods developed in this thesis, it is sufficient to consider only the most straightforward case of minimal overlap between the individual partitions, i.e. an overlap of one layer of elements or nodes, respectively. Obviously, this concept of overlapping decomposition fits quite naturally to the typical tasks within a finite element program: first, each processor performs an elementwise integration of its own partition of the computational domain including the (relatively few) elements at the inter-processor boundaries. Then, the resulting quantities (e.g. local element load vectors and stiffness matrices) are assembled into the respective FE nodes of each processor. Thus, overlapping domain decomposition as described above provides a very elegant way of processing finite element integration and assembly, which is completely free of communication due to the distributed storage of the resulting global

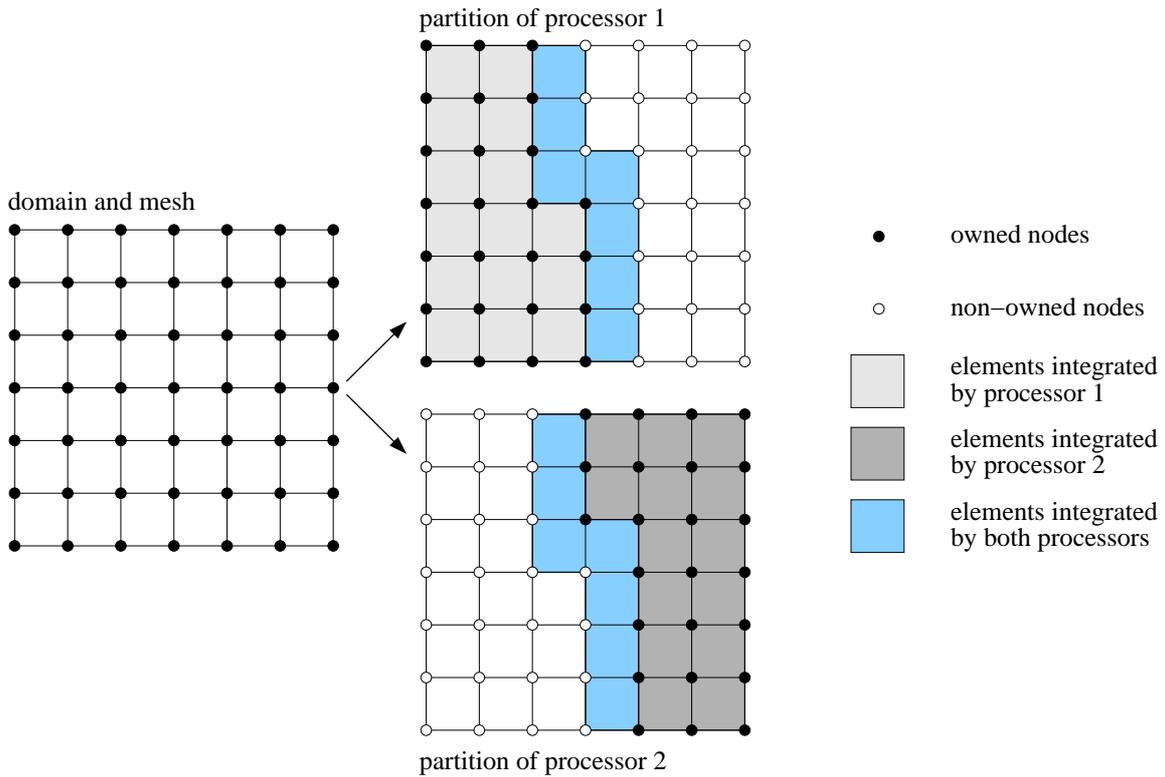


Figure 4.6: An example of overlapping domain decomposition and parallel assembly involving two independent processors.

vector and matrix objects. While this rough introduction is by far not complete or rigorous from the viewpoint of parallel software design, it is sufficient for the following ideas on redistribution and load balancing to be comprehensible. For further details on the C++ based implementation of parallel (i.e. distributed) matrix and vector objects as well as the associated linear algebra, the interested reader is exemplarily referred to the documentation of open-source libraries of the Trilinos Project conducted by Sandia National Laboratories (Heroux et al. [69]).

Returning to the efficient parallel treatment of mortar methods and the derived mesh tying and contact algorithms, an exemplary mesh tying problem setup consisting of two cubic bodies as depicted in Figure 4.7 is considered now. In total, the FE model contains 681,476 volume elements (with 2,136,177 displacement degrees of freedom) and 15,041 contact interface elements, which are distributed in parallel among several processors. As explained in the last paragraph, this partitioning generated via the ParMETIS library is in a sense optimal for the integration and assembly of the individual volume finite elements of the two bodies, i.e. the corresponding workload is equally distributed among all processors. For both tied and unilateral contact interaction, however, additional (but conceptually similar) tasks have to be performed locally at the interface: as explained in detail in Section 4.2, computing the interface contributions to the overall discrete problem formulation involves the mortar segmentation process, integration and assembly of the mortar matrices \mathbf{D} and \mathbf{M} , to name only the most important tasks. Especially in three dimensions and for large interfaces, these computations may become quite time-consuming, so that they actually carry considerable weight as compared to the remaining time needed for FE

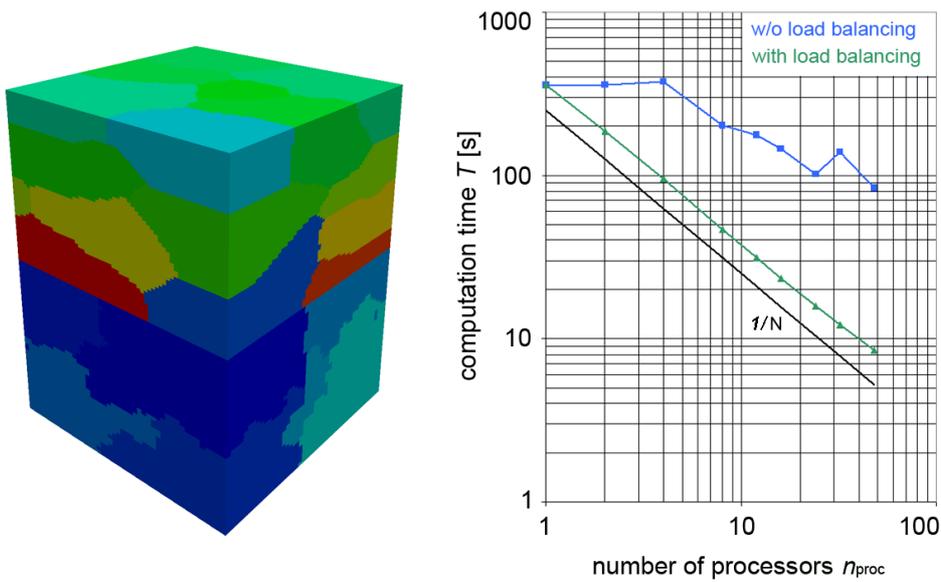


Figure 4.7: Parallel redistribution and load balancing – initial partitioning for exemplary mesh tying problem setup using 32 processors (*left*) and strong scaling diagram (*right*)

evaluation and linear solvers. In contrast to NTS formulations, the high approximation quality of mortar methods comes at a price here. Unfortunately, the parallel distribution of the mortar interface itself is not optimal at all, which can easily be seen in Figure 4.7. In this context, it is important to commemorate the slave-master concept typically used for implementing contact algorithms, where the interface-related workload is completely assigned to the slave side (or non-mortar side) whereas the master side (or mortar side) is passive. Thus, in the given example, the slave side of the interface (and thus the entire workload related to mesh tying) is associated with only 4 out of 16 processors.

The right hand side of Figure 4.7 illustrates typical results for the parallel efficiency of the presented mortar algorithms in a so-called *strong scaling* diagram. Therein, the computation time for numerical integration and assembly of all interface-related quantities T is plotted against the total number of processors n_{proc} with logarithmic scales applied to both axes. Perfect scalability of the examined numerical algorithm is represented by a straight line with a negative slope of -1 , thus representing the evident relation

$$T = \frac{c}{n_{\text{proc}}} \quad \text{with } c > 0. \quad (4.103)$$

It can clearly be seen that no perfect scalability is achieved with the presented algorithms without load balancing (blue curve in Figure 4.7). This is due to the non-optimal distribution of the slave surface among the participating processors as already described above. The results clearly motivate the development of an efficient parallel redistribution and load balancing strategy for mortar finite element methods. The approach proposed in the following is based on three steps, where the first one is of fundamental importance and is therefore needed for both mesh tying and contact applications. In contrast, the second and third step are purely contact-specific.

The rather simple basic idea of the first step is an *independent* parallel distribution of the finite elements in the domain and the mortar elements at the mesh tying or contact interface in order to

achieve optimal parallel scalability of the computational tasks associated with both, i.e. integration and assembly in $\Omega^{(1)}$ and $\Omega^{(2)}$ as well as integration and assembly on $\gamma_c^{(1)}$ and $\gamma_c^{(2)}$. Again using ParMETIS, this redistribution of the interface elements can readily be performed during problem initialization at $t = 0$. Results for the test model introduced above are also visualized (green curve in Figure 4.7), thus demonstrating that this simple modification already allows for excellent parallel scalability within a wide range concerning the number of processors n_{proc} . However, dependent on the considered problem size, parallel redistribution only makes sense up to a certain n_{proc} . It is quite natural that such a limit exists, because there are of course some computational costs associated with the proposed redistribution procedure itself. If too many processors are used in relation to the problem size, these costs (mainly due to communication) become dominant and redistribution is no longer profitable beyond this point.

As already mentioned, this strategy can be further refined for unilateral contact applications. In contrast to mesh tying, contact interfaces are characterized by two additional complexities: the actual contact zone is not known a priori and it may constantly and significantly vary over time. Thus, in a second and third step, the proposed redistribution strategy is adapted such that it accommodates these additional complexities. Concretely, it can be seen from the Hertzian contact example in Figure 4.8 that parallel redistribution must be limited to the actual contact area instead of the potential contact area, because the entire computational effort of numerical integration and assembly is connected with the former. Moreover, whenever finite deformations and large sliding motions occur, the described redistribution needs to be performed dynamically, i.e. over and over again. Such a dynamic load balancing strategy is then typically triggered by a suitable measure for the workload of each individual processor. The parallel balance of the workload among all processors is monitored and a simple criterion whether to apply dynamic load balancing within the current time step or not can be formulated as

$$IF \left(\frac{T^{\max}}{T^{\min}} > r \right) \rightsquigarrow \text{redistribute} . \quad (4.104)$$

Herein, the minimum and maximum computation times of one individual processor in the last time step are denoted as T^{\min} and T^{\max} , respectively. The parameter $r > 1$ represents a user-defined tolerance. For example, choosing $r = 1.2$ implies that at most 20% unbalance of the parallel workload distribution are tolerated. Of course, the rather simple condition in (4.104) can easily be extended to incorporate more sophisticated criteria for dynamic load balancing. However, already the short overview given here shows that redistribution and load balancing provide an efficient tool for increased parallel efficiency of mortar algorithms for mesh tying and contact simulations. Corresponding numerical examples (see e.g. Section 4.7.9) demonstrate that the proposed approach is actually indispensable when considering large-scale applications.

4.6.2 Search algorithm for two body contact

The search for bodies or individual finite elements that might possibly come into contact is an important algorithmic aspect of any FEM contact formulation. In particular, this is true in the context of finite deformations and large sliding motions as primarily considered throughout this thesis, because the contact situation (active contact set) continuously changes in such scenarios. Contact search algorithms have been the subject of intensive research since the very beginnings

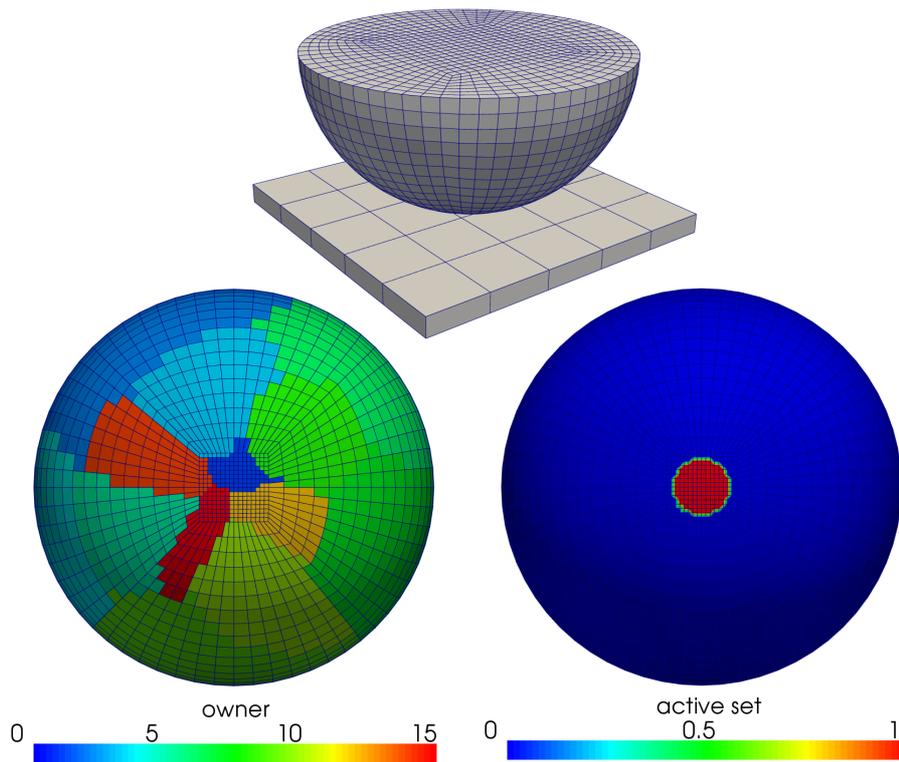


Figure 4.8: Motivation for parallel redistribution exemplified with a Hertzian contact example – the active contact region (bottom right) is relatively small as compared with the potential contact surface (i.e. the whole hemisphere). Without redistribution only 6 out of 16 processors would carry the entire workload associated with contact evaluation (bottom left).

of computational treatment of contact mechanics problems. Some well-known approaches can be found in Benson and Hallquist [15], Williams and O’Connor [173, 174] and Zhong and Nilsson [194, 195]. Furthermore, the interested reader is referred to the textbook by Wriggers [180] for a very comprehensive overview on the topic.

The basic motivation for efficient contact search algorithms can be easily understood. Neglecting possible self contact for a while (see the following Section 4.6.3), a naive search approach for two body contact would require to check *all* finite elements on the slave side against *all* finite elements on the master side for proximity. Thus, the associated number of operations would be $N \cdot M$, where N and M are the total numbers of slave and master elements, respectively. Assuming $M \approx N$ for the sake of simplicity, the resulting computational complexity of such so-called brute force search algorithms is $\mathcal{O}(N^2)$. Clearly, this makes contact search unacceptably slow already for rather moderate problem sizes. Thus, brute force search algorithms are only suitable for very basic method development purposes, but not for challenging contact applications based on large-scale simulation models.

In the following, a short overview of the parallel search algorithm for two body contact used in this thesis will be given, which is closely related to the work by Yang and Laursen [187]. Basically, most contact search algorithms consist of two components, i.e. a hierarchical global search structure (so-called search tree) and an efficient local geometry representation (so-called bound-

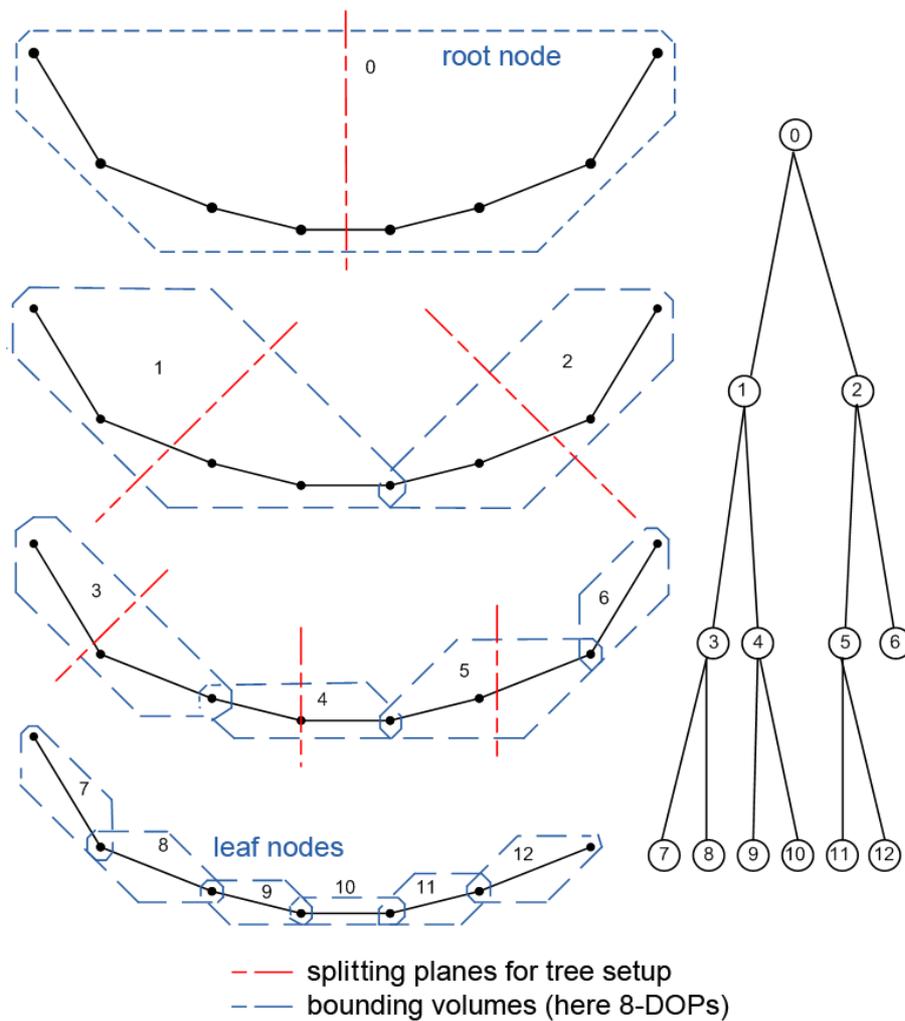


Figure 4.9: Search algorithm for two body contact – 2D example based on 8-DOPs and a hierarchical binary tree structure.

ing volumes). Here, discretized orientation polytopes with k edges (k -DOPs) serve as bounding volumes. Compared to the also commonly employed axis-aligned bounding boxes (AABBs), the k -DOPs allow for a much tighter and thus more efficient geometrical representation of the contact surfaces. For 2D simulations, the bounding volumes are typically 8-DOPs, while 18-DOPs are employed in the 3D case. Figure 4.9 provides a schematic illustration of these ideas for a two-dimensional setting. Further details and comprehensive illustrations can be found in Yang and Laursen [187]. As can also be seen from Figure 4.9, both slave and master surface are then organized and stored within hierarchical binary tree structures, which allow for very fast search and update procedures. The search tree is typically only built once during problem initialization in a top-down way. This process starts from a so-called root node, which contains the entire slave or master contact surface, and then the considered surfaces are continuously divided into halves until arriving at the individual finite elements (so-called leaf nodes of the search tree). An update of the tree, i.e. of the contact geometry, must be performed after each nonlinear iteration step due to the fact that finite deformations are considered here.

The search procedure itself basically consists of a recursive algorithm starting with an intersection test of slave and master root nodes. Wherever necessary, i.e. wherever an overlap of the corresponding bounding volumes is detected, the search algorithm proceeds into the lower tree levels until the leaf level is reached. A theoretical analysis in Yang and Laursen [187] predicts the resulting algorithm complexity to be $\mathcal{O}(N \cdot \log(N^2))$, which has also been confirmed in various numerical investigations. Going beyond the implementation in Yang and Laursen [187], which is limited to the single-processor case, the presented search algorithm has been extended to fit into the parallel FE simulation framework developed in this thesis. As explained in Section 4.6.1, slave and master surface are then distributed among several independent processors and the tree update as well as search procedures are only performed on the fraction of the slave contact surface that is actually part of the problem partition of the respective processor. This generates a distributed search algorithm with optimal parallel scalability, where only the geometry of the (likewise distributed) master contact surface needs to be communicated among all processors in order to detect all possible contact pairs. For all two body contact examples considered in Section 4.7 of this thesis, the described approach reduces the contribution of contact search to the overall computational costs to a maximum of only a few percent. Thus, the computational costs associated with contact search become negligible from an engineering perspective, for which reason no detailed time measurements for contact search will be presented here.

4.6.3 Search algorithm for self contact and multiple bodies

This paragraph gives an overview of two special problem classes in computational contact mechanics, namely self contact and contact of multiple bodies, which both are of great relevance in engineering practice. Typical self contact applications include certain forming processes or the folding of energy absorbing crash structures in vehicles, while multiple bodies (or particles) typically need to be considered for flow of granular media and certain biomechanical scenarios, e.g. for capillary flow of red blood cells. Two characteristic examples for the mentioned problem classes are illustrated in Figure 4.10.

Of course, the underlying physical principles of contact interaction remain unchanged, whether self contact, classical two body contact or multiple bodies are investigated. Therefore, it is obvious that the mortar finite element formulations developed in this thesis can also be applied to self contact and contact of multiple bodies without conceptual changes. The only algorithmic difference to the two body case exclusively considered so far is the way in which the contact search is realized. As can be seen in Figure 4.10, self contact is characterized by only one *single* potential self contact surface, so that no a priori definition of slave and master surfaces is possible but this assignment rather needs to be done in a dynamic manner. Thus, the search procedure presented in Section 4.6.2 needs to be adapted and extended in order to accommodate possible self contact. Moreover, Figure 4.10 indicates that the case of multiple bodies lies somewhere in between two body contact and self contact with regard to its numerical treatment. Again, it is not possible to find a unique a priori definition of slave and master surfaces. However, once slave and master pairs are (dynamically) assigned, this scenario can basically be interpreted and treated as a great many of simple two body contacts.

The search algorithm for self contact and multiple bodies employed here is again closely related to the work given in Yang and Laursen [186, 187]. Many fundamental concepts already introduced in the previous paragraph, such as the binary tree hierarchy and the k-DOP bounding

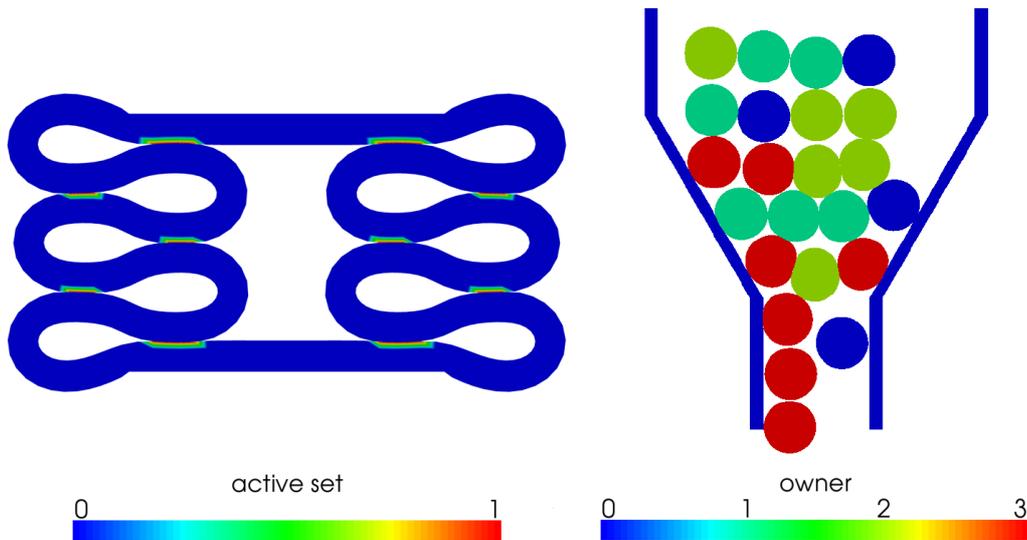


Figure 4.10: Examples for self contact and contact of multiple bodies. The active set is visualized for self contact (*left*) and the parallel distribution is shown for the multiple body case (*right*).

volumes, stay exactly the same. In addition, some algorithmic extensions are needed and will be outlined in the following. Most importantly, the binary tree hierarchy is now typically built in a bottom-up way based on mesh connectivity, e.g. using a dual graph as illustrated in Figure 4.11. The update procedure of the self contact geometry remains conceptually unchanged, and the only difference for the contact search itself lies in the fact that it is now performed within one single tree structure instead of between two independent binary trees. As described in Yang and Laursen [186], self contact search may be further accelerated by sorting out infeasible pairs, for example by taking into account adjacency information and curvature criteria. An important new aspect of self contact as compared to the classical two body case is the dynamic assignment of slave and master regions. For this purpose, all self contact sub-surfaces (so-called self contact patches) are identified and opposite patches are then given slave and master status, respectively. Finally, it should be pointed out that an efficient parallelization is especially challenging in the case of self contact due to the fact that parts of the single contact surface must be communicated between the different processors based on a ring token strategy. The realization of such a communication pattern goes beyond the scope of this thesis.

Contact of multiple bodies (or particles) as depicted in Figure 4.10 allows for a straightforward algorithmic treatment within the same framework as self contact. Therefore, all surfaces of the involved bodies together are interpreted as one disconnected self contact surface and the search algorithm outlined above is applied. As this procedure leads to several independent root nodes in the binary tree structure, it even allows for a parallel distribution to some extent. The independent root nodes, i.e. the individual bodies, are distributed equally among all processors and search is then performed in parallel. In doing so, both two body contact between different root nodes of the tree *and* possible self contact within one root node need to be considered, thus making contact of multiple bodies kind of an in-between of the two classical scenarios described above. Exemplary applications are presented in Section 4.7.7 for self contact and in Section 4.7.8 for multiple bodies in order to validate the proposed search methods.

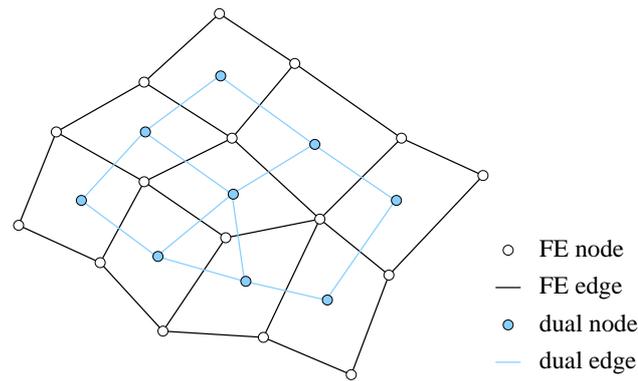


Figure 4.11: Search algorithm for self contact – exemplary finite element surface mesh and the corresponding dual graph used for the bottom-up setup of a binary tree.

4.7 Numerical examples

Several numerical examples are presented and analyzed in the following to evaluate the proposed mortar approach for finite deformation contact. To some extent, these investigations build upon the previous results for the mesh tying case given in Section 3.5, but the focus is now especially on the additional complexities associated with unilateral contact scenarios, such as the active set strategy and the consistent linearization of mortar coupling terms. If not stated otherwise, first-order finite elements in 2D and 3D combined with dual Lagrange multiplier interpolation are used for the numerical examples, and thus the final linear systems of equations can be condensed according to the procedure outlined in Section 4.5.4.

First, some advantageous characteristic properties of the devised numerical algorithms are illustrated, which can all be ascribed to the proposed mortar finite element approach. These properties include the consistent treatment of non-matching meshes (see patch tests in Section 4.7.1), optimal spatial convergence rates (see mesh refinement studies in Section 4.7.2) and the excellent performance of semi-smooth Newton methods as nonlinear solution scheme in finite deformation scenarios (see sliding contact example in Section 4.7.3). The consistent extension of mortar discretization and the semi-smooth Newton approach towards Coulomb friction is presented in Section 4.7.4. Furthermore, a special focus is set on dual Lagrange multiplier interpolation for quadratic finite elements in 3D as introduced in Section 3.3.2, with a corresponding numerical example being investigated in Section 4.7.5. Mechanical conservation properties for the unilateral contact case are discussed in Section 4.7.6, whereas special contact situations such as self contact and impact of multiple bodies follow in Sections 4.7.7 and 4.7.8. Finally, it is demonstrated that all proposed methods are readily applicable to challenging real world contact scenarios, due to the fact that they allow for a very efficient algorithmic treatment of the underlying large-scale simulation models containing several million degrees of freedom. For the validation of this parallel efficiency, a torus impact simulation is investigated in Section 4.7.9.

4.7.1 Consistency – Patch tests

Patch tests are a simple means applied here for testing the ability of mortar finite element methods to represent constant stress states across non-matching contact interfaces, i.e. for testing

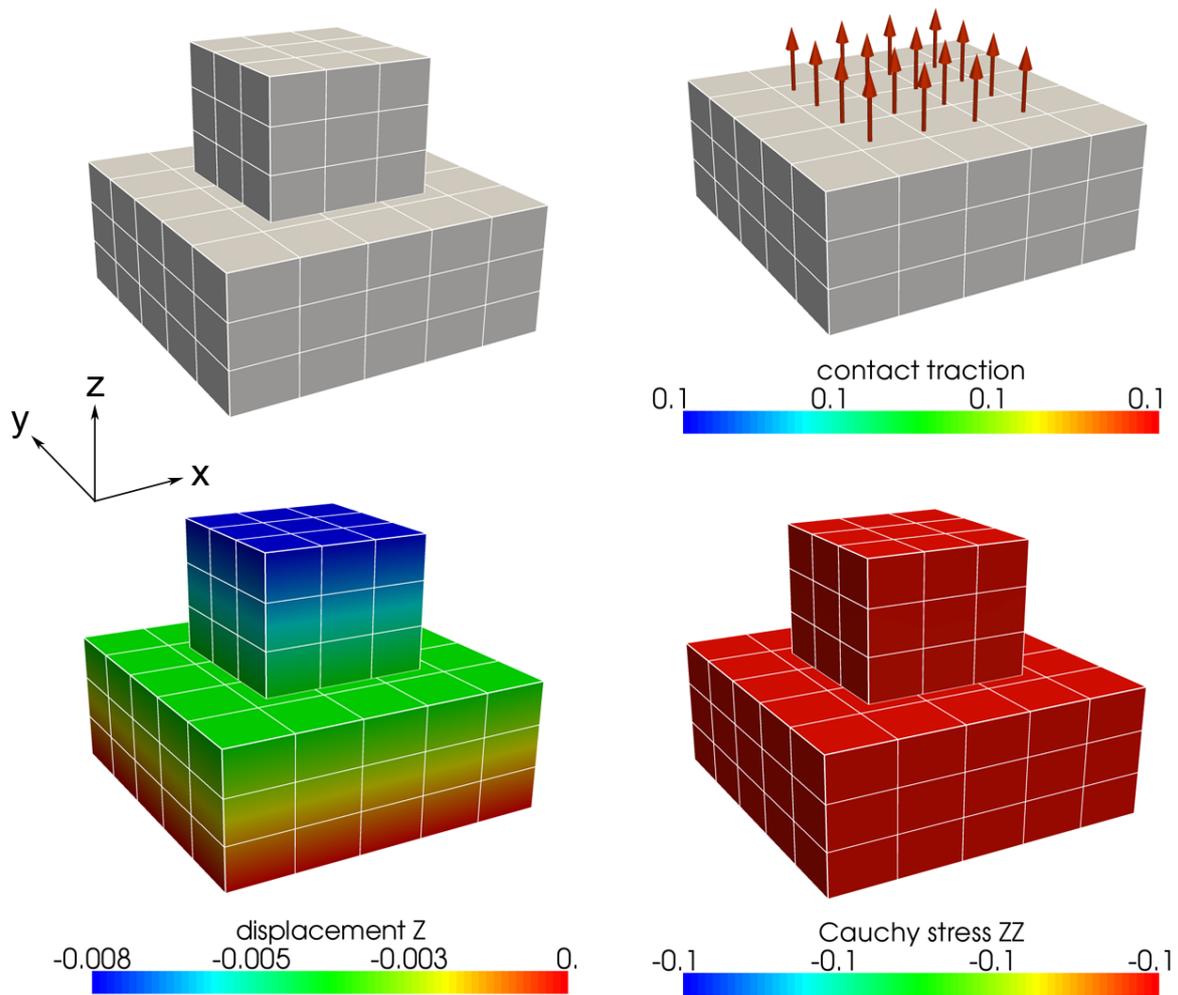


Figure 4.12: Contact patch test – finite element mesh with *hex8* elements (top left), interface tractions represented by the discrete Lagrange multipliers λ (top right), displacement u_z (bottom left) and Cauchy stress σ_{zz} (bottom right).

consistency of the proposed numerical methods. While the mesh tying case has already been investigated in detail in Section 3.5.1, a similar example is now presented in the context of unilateral contact constraints and frictionless sliding. It is well-known that the classical NTS method fails to satisfy typical contact patch tests exactly, see e.g. Crisfield [27], El-Abbasi and Bathe [36] and especially the early work of Taylor and Papadopoulos [156]. Nevertheless, it should be mentioned that modifications to the NTS approach are possible which guarantee patch test satisfaction. One successful implementation of such modified NTS algorithms can be found in Zavarise and De Lorenzis [190]. In contrast, mortar methods as presented in this thesis are guaranteed to satisfy typical contact patch tests exactly by construction due to the underlying variational framework and the consistent interpolation of the contact tractions.

In the following, a test setup is considered which consists of two blocks with non-matching meshes and a plane contact interface, see Figure 4.12. The lower block is supported at its bottom surface and both the upper surface of the upper block and the non-contact upper surface part of

the lower block are loaded with a constant pressure $p = -0.1$ in z -direction. A compressible Neo–Hookean material law with Young’s modulus $E = 100$ and Poisson’s ratio $\nu = 0.3$ is employed for both bodies. Results are exemplarily shown in Figure 4.12 for first-order interpolation with *hex8* elements and dual Lagrange multipliers, however all remaining first-order and second-order interpolation types as well as standard Lagrange multipliers yield equally perfect results. As expected, the proposed mortar contact approach is able to exactly transmit a constant contact pressure across the non-matching interface, thus resulting in a linear displacement field and a constant stress field up to machine precision.

It is important to point out that exact patch test satisfaction is not influenced by the choice of slave and master sides. If the lower block is chosen as slave side, some slave surface elements are not fully contained in the contact integration area and thus the modification of dual shape functions for partially integrated elements as presented in Section 4.3.3 becomes necessary. Finally, a brief remark on penalty methods in combination with mortar contact discretization should be made, which again emphasizes why Lagrange multiplier methods are more favorable for computational contact mechanics. In general, penalty regularization hampers exact patch test satisfaction because penalty forces are applied proportionately to the weighted nodal gaps and the weighted gaps in turn are proportional to the respective support of their nodal basis functions. In order to achieve exact patch test satisfaction nonetheless, the ad hoc definition of an additional nodal scaling factor is required, see e.g. Yang et al. [189].

4.7.2 Spatial convergence – Hertzian contact

In this paragraph, the 3D Hertzian contact problem consisting of an elastic hemisphere ($R = 8$, St.-Venant–Kirchhoff material model with $E = 200$, $\nu = 0.3$) and a rigid planar surface is analyzed. A constant pressure $p = -0.2$ is applied to the top surface of the hemisphere and contact interaction is assumed to be frictionless. Analytical solutions for the contact traction distribution are well-known for Hertzian contact problems and can usually be characterized via the maximum normal contact traction λ_n^{\max} and the radius ρ of the circular contact zone, i.e.

$$\lambda_n^{\max} = 0.388 \sqrt[3]{4p\pi E^2}, \quad \rho = 1.109 \sqrt[3]{\frac{pR^3\pi}{2E}}, \quad (4.105)$$

see e.g. Timoshenko and Goodier [161] for further details of the derivation. For the given set of parameters, one obtains $\lambda_n^{\max} = 18.042$ and $\rho = 1.031$ as analytical solution. The mortar finite element discretization is based on 20-node hexahedral elements and dual Lagrange multiplier interpolation as proposed in Section 3.3.2. The problem setup, an exemplary mesh and an exemplary numerical solution for the contact tractions are illustrated in Figure 4.13.

Uniform mesh refinement is applied to analyze spatial convergence of the numerical solution for the maximum contact traction λ_n^{\max} , see Figure 4.14, and of the discretization error $\mathbf{u} - \mathbf{u}_h$ measured in the energy norm, see Figure 4.15. The finest mesh analyzed consists of approximately 3 million degrees of freedom and the relative error of λ_n^{\max} with respect to the analytical solution is 2.8%. This deviation is solely due to the fact that while the analytical solution is based on the small deformation assumption, the implementation fully considers all nonlinearities and thus the mortar matrices \mathbf{D} and \mathbf{M} as well as the weighted gaps $(\tilde{g}_n)_j$ are deformation-dependent. However, the implementation is still consistent (or actually even more

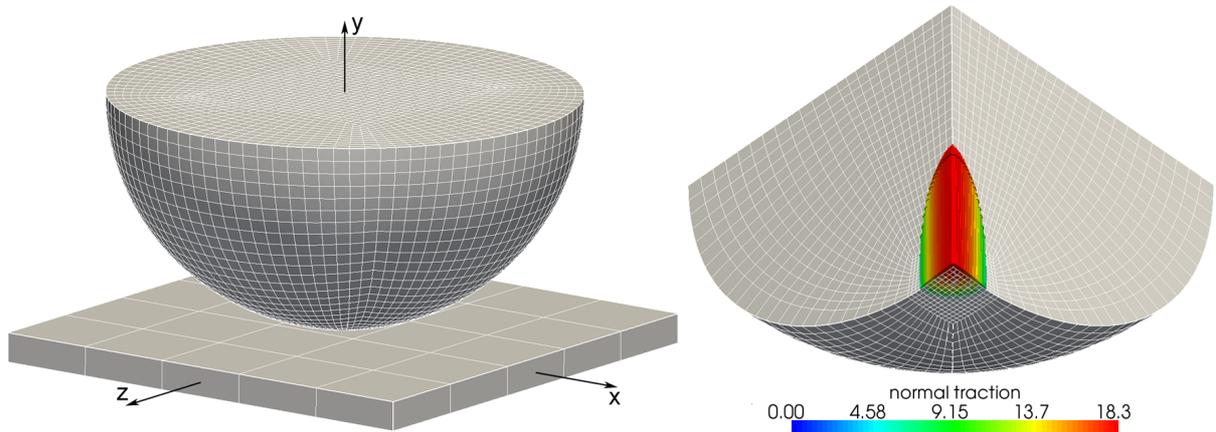


Figure 4.13: Hertzian contact – problem setup and exemplary finite element mesh (left), one quarter of the deformed geometry and schematic normal contact traction solution (right).

consistent than the analytical solution, which apparently contains an approximative assumption for the given load magnitude). This becomes clear when the external load is exemplarily reduced to $p = 0.05$. Then, the difference between small deformation and finite deformation formulation and thus also the relative deviation of λ_n^{\max} further diminishes. Precisely, the analytical solution is given as $\lambda_n^{\max} = 11.366$ and $\rho = 0.650$ in that case and the relative error for the finest mesh reduces to 1.7%, which can also be seen in Figure 4.14.

The results of the convergence study of the discretization error are summarized in Figure 4.15, where only the case $p = 0.2$ is considered. As there indeed exists an analytical solution for the maximum contact traction λ_n^{\max} but not for the displacement field \mathbf{u} , a numerical reference solution is computed based on a sufficiently fine *hex20* mesh. Asymptotically, one can observe $\mathcal{O}(h^{3/2})$ convergence in the energy norm for second-order interpolation, which is in accordance with theoretical results for unilateral contact, see also Popp et al. [128] and Wohlmuth et al. [179]. As discussed in Section 4.3, no better a priori estimates than $\mathcal{O}(h^{3/2})$ can be expected due to the reduced regularity of contact solutions. First-order interpolation based on *hex8* elements is investigated for comparison purposes and yields optimal $\mathcal{O}(h)$ results in the energy norm.

Numerical results for the active contact zone and for the normal contact traction distribution are illustrated in Figure 4.16 for different mesh sizes h . It can be seen very clearly how the circular shape of the contact zone and the traction distribution are resolved more and more with mesh refinement. Furthermore, an excellent agreement of the numerical solution with the analytical reference value for the contact zone radius ρ is visually confirmed. Yet, Figure 4.16 also shows another well-known fact, namely that the contact zone radius cannot be reproduced exactly for very coarse meshes. This is due to the fact that it is practically impossible that nodes in a coarse contact surface mesh actually coincide with the analytically computed location of the transition between contact zone and non-contact zone. However, the described small discrepancies further diminish with mesh refinement, see Figure 4.16, and second-order interpolation. Alternatively, special node relocation strategies can be applied as proposed in Franke et al. [45]. All in all, the results demonstrate the unrestricted applicability of the proposed mortar methods using dual Lagrange multiplier interpolation to unilateral contact analysis.

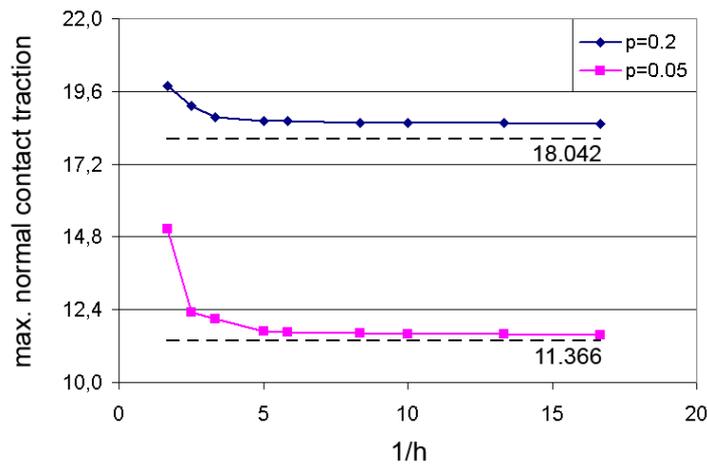


Figure 4.14: Hertzian contact – convergence of the maximum normal contact traction λ_n^{\max} with mesh refinement for an external load $p = 0.2$ and for a reduced external load $p = 0.05$.

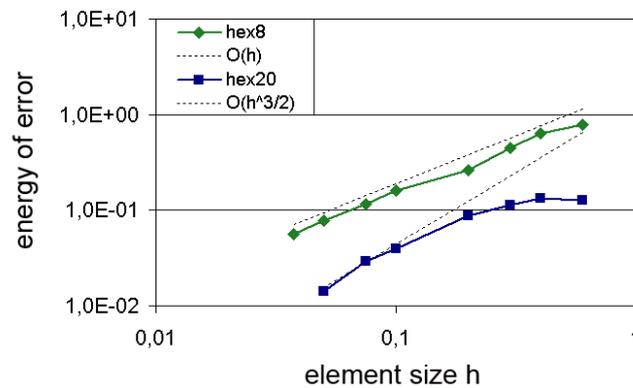


Figure 4.15: Hertzian contact – convergence of the discretization error $\mathbf{u} - \mathbf{u}_h$ measured in the energy norm with uniform mesh refinement for first-order and second-order hexahedral meshes.

Characteristic results concerning the numerical efficiency of the employed semi-smooth Newton type active set strategy as proposed in Section 4.4 are given in Figure 4.17. For exemplary *hex8* and *hex20* based meshes, the relative L^2 -norm of the total residual and the number of active nodes are monitored over all nonlinear iteration steps of the single load step needed to solve this quasistatic Hertzian contact example. Regardless of the interpolation order, the semi-smooth Newton approach locates the correct active set (which consists of up to 1669 nodes here) within only a few iteration steps. With the active set being fixed, the nonlinear iteration scheme reduces to a standard (smooth) Newton–Raphson method, and one obtains quadratic convergence in the limit owing to the underlying consistent linearization. It is worth noting that the semi-smooth Newton scheme cannot be started with an empty active set $\mathcal{A} = \emptyset$ in this quasistatic example, since the hemisphere would then initially have no constraints in y -direction. There exist different possibilities to initialize the active set in such a case (e.g. using only the lowest node of the hemisphere), but the convergence behavior of the active set strategy remains almost unaffected by this choice, see Hüeber and Wohlmuth [80] for detailed investigations.

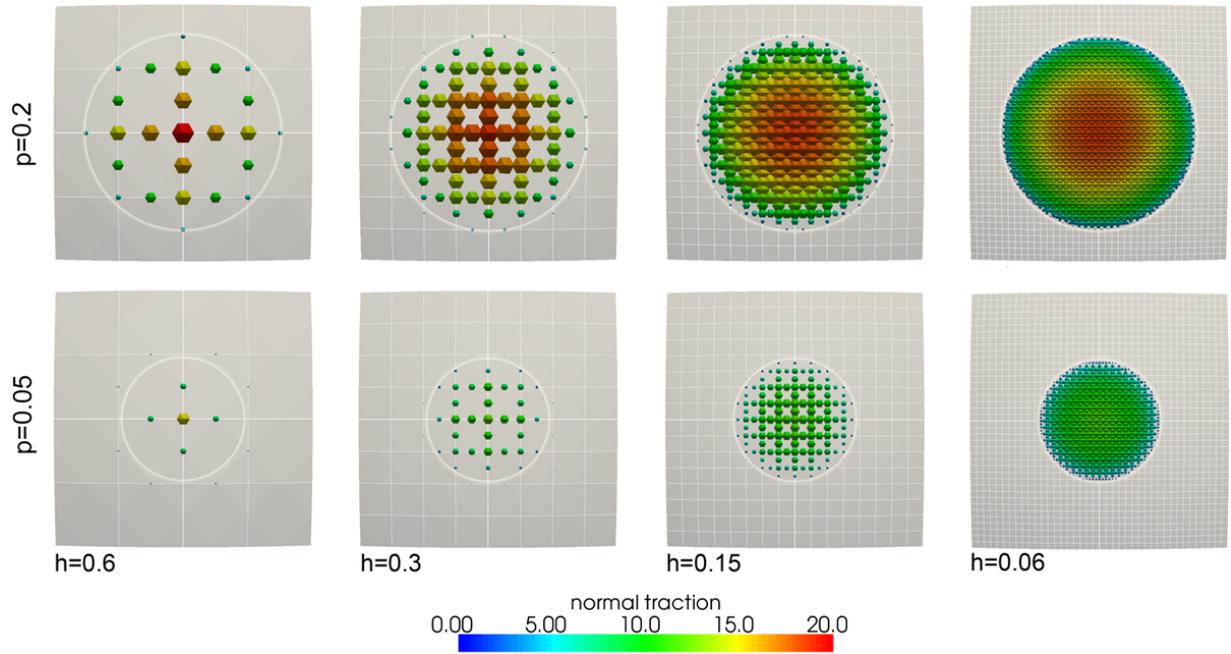


Figure 4.16: Hertzian contact – vertical closeup view of the active contact zone and visualization of the normal contact traction solution for different mesh sizes h when applying an external load $p = 0.2$ (top) or a reduced external load $p = 0.05$ (bottom). The respective analytical solution for the contact zone radius ρ is indicated with a light circle.

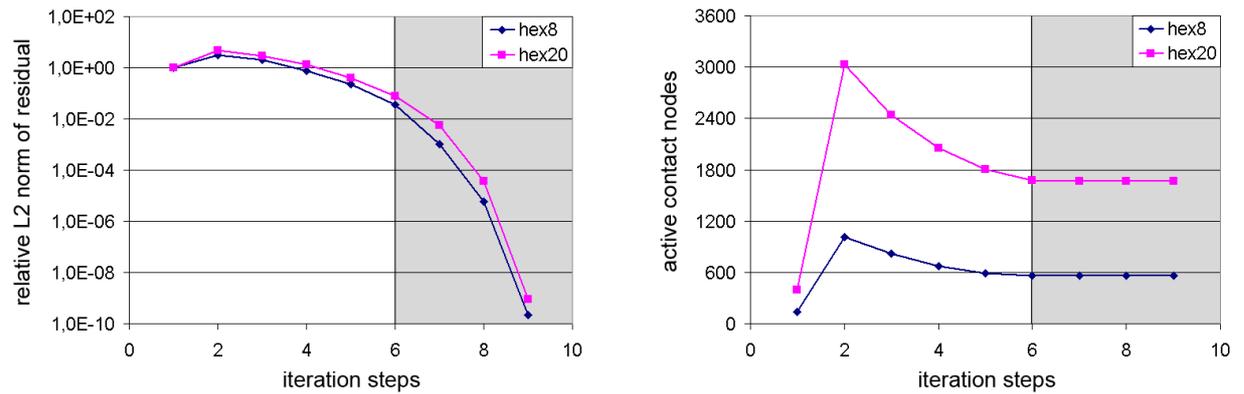


Figure 4.17: Hertzian contact – exemplary convergence behavior of the semi-smooth Newton method in terms of the relative L^2 -norm of the residual (left) and in terms of the active contact set (right) for a mesh size of $h = 0.075$. The shaded regions indicate that the active contact set is already fully converged within these iteration steps.

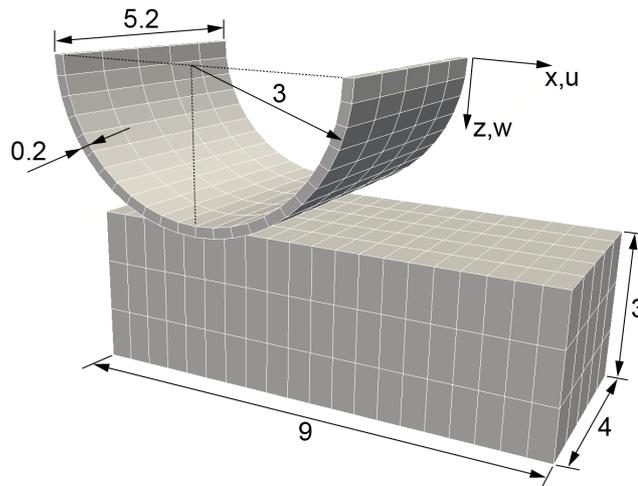


Figure 4.18: Ironing example – geometry, finite element mesh and prescribed displacements.

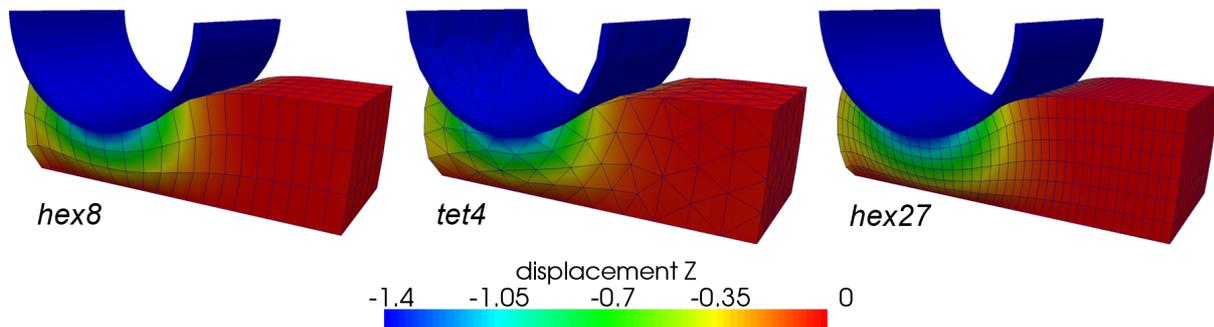


Figure 4.19: Ironing example – exemplary deformation state for different interpolation types. The 27-node hexahedral elements are represented as eight 8-node hexahedral elements.

4.7.3 Finite deformations – Ironing example

In this paragraph, finite deformation contact of a half-cylindrical elastic die (Neo–Hookean model, $E = 1000$, $\nu = 0.3$) intruding into an elastic block (Neo–Hookean model, $E = 1.0$, $\nu = 0.3$) is analyzed, which is also commonly referred to as ironing example. Similar investigations have been made in Puso and Laursen [132, 133], where also further details can be found. The die is first pressed into the block by prescribing a vertical displacement $w = 1.4$ in z -direction within 10 quasistatic time steps. Then it slides along the block during 65 further time steps until a prescribed horizontal displacement $u = 4.0$ in x -direction is reached. Geometry and an exemplary finite element mesh are shown in Figure 4.18. For discretization, 8-node hexahedral, 4-node tetrahedral and 27-node hexahedral elements are considered, yielding bilinear, linear and biquadratic interpolation on the contact surfaces, respectively.

An exemplary deformation state for all considered finite element types is illustrated in Figure 4.19. The numerical efficiency of the proposed 3D mortar contact formulation in finite deformation situations is evaluated by monitoring the total residual norm during the nonlinear iterations of one representative quasistatic time step. Different types of nonlinear solution schemes

Table 4.1: Ironing example – convergence behavior in terms of the total residual norm for the time step starting from $u = 0$, $w = 0.98$. Comparison of different solution algorithms.

Step	Semi-smooth Newton full linearization	Fixed-point approach full linearization	Fixed-point approach incomplete linearization
1	6.536e+03 (*)	6.536e+03	6.535e+03
2	2.494e+03 (*)	2.495e+03	2.493e+03
3	8.267e+02	8.269e+02	8.261e+02
4	1.795e+02	1.796e+02	1.793e+02
5	1.403e+01	1.404e+01	1.400e+01
6	1.133e−01	1.136e−01	1.174e−01
7	8.703e−06	8.731e−06	2.171e−03
8	3.195e−12	3.369e−12 (+)	4.462e−04
9		↓	1.656e−04
⋮		1 further	⋮
25		Newton cycle	1.395e−09
26			6.457e−10 (+)
			↓
			1 further
			Newton cycle
Σ	8	16	52

(*) = change in active contact set

(+)= check of active contact set, Newton cycle has to be repeated

and active set strategies are compared in Table 4.1, and the chosen time step involves both a significant increment in displacement ($\Delta w = 0.14$) and considerable changes of the active contact set. First, only the linear interpolation case with 8-node hexahedral elements is analyzed.

The fully linearized semi-smooth Newton scheme proposed in Section 4.4 (column 1 of Table 4.1) is compared with a fixed-point approach for the active set search (column 2 of Table 4.1) where Newton-cycles have to be repeated whenever changes in the active contact set occur (see also Section 4.4 for details). To illustrate the necessity of a consistent linearization, an algorithm partly omitting this step is also considered (see Hartmann [65], column 3 of Table 4.1). This case, without linearization of the two mortar matrices \mathbf{D} and \mathbf{M} and the weighted gaps $(\tilde{g}_n)_j$, is labeled “incomplete linearization” in Table 4.1. Note that the absolute values of the total residual norm within the first steps are very similar in all columns, which is merely a consequence of the problem setup with its high stiffness ratio and prescribed displacements. The total residual norm $\|\mathbf{r}_{\text{tot}}\|$ is dominated by the force residual norm $\|\mathbf{r}\|$, whereas the absolute value of the constraint residual norm is comparatively small.

The convergence results demonstrate that the proposed semi-smooth Newton method features excellent convergence in this example, while the approach with two nested loops is far less efficient because Newton cycles have to be repeated whenever changes in the active contact set occur. In this specific example, one repetition is necessary, which doubles the total number of iterations in comparison with the proposed method in column 1 of Table 4.1. Thus, the integration of all types of nonlinearities into a semi-smooth Newton iteration can avoid tremendous com-

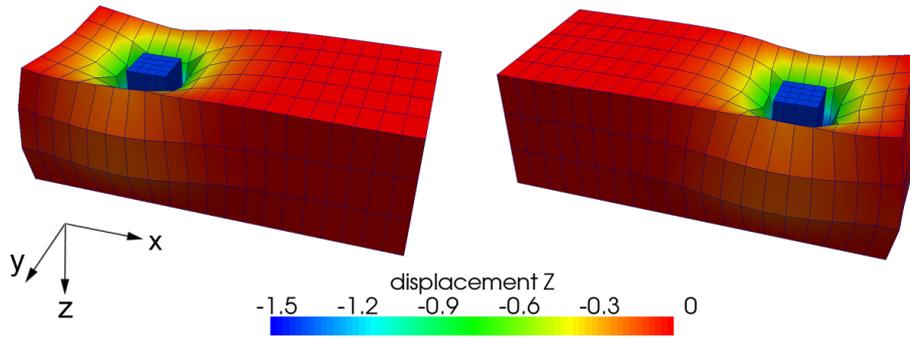


Figure 4.20: Ironing example – two characteristic deformation states for the cubic indenter.

putational cost, especially in finite deformation situations where the active set continuously and considerably changes. It is remarkable that merging the treatment of active set search and finite deformations results in no or very few extra iteration steps within one Newton cycle. Examining column 3 in Table 4.1, it is obvious that consistent linearization of all quantities, including contact forces as well as normal and tangential vectors, is crucial in order to avoid deterioration of convergence. Full linearization and the proposed semi-smooth Newton approach are shown to be indispensable features for an efficient primal-dual active set strategy in the context of finite deformation contact. An equally good convergence behavior is achieved when using 4-node tetrahedral elements or second-order interpolation based on 27-node hexahedral elements.

Finally, a modified ironing problem is considered, with the half-cylindrical die being replaced by a cubic indenter (Neo–Hookean model, $E = 1000$, $\nu = 0.3$) with an edge length of 1.0. As has been explained in Puso and Laursen [132], this example is extremely challenging for contact formulations owing to the stress near-singularities occurring at the corners of the indenter. Exemplary deformation results in Figure 4.20 confirm that the proposed dual mortar approach is nonetheless capable of handling this situation, which further underlines its robustness.

4.7.4 Coulomb friction – Sliding contact

This numerical example demonstrates the performance of the proposed algorithms in the most general contact case, including finite sliding and Coulomb friction as well as significant active set changes. Mortar finite element discretization of the Coulomb friction case and a corresponding extension of the semi-smooth Newton approach presented in Section 4.4 are described in great detail in Gitterle [54], Gitterle et al. [55] and Hübner et al. [84].

The considered test setup, first suggested in Hübner [79], consists of two half-cylindrical structures as depicted in Figure 4.21. The inner and outer radii of the upper body (slave) and the lower body (master) are $r_i^{(1)} = 0.09$, $r_o^{(1)} = 0.13$, $r_i^{(2)} = 0.13$, $r_o^{(2)} = 0.17$ and the lengths of the two bodies are defined as $l^{(1)} = 0.11$ and $l^{(2)} = 0.17$. Initially, the two bodies are separated by a gap $g = 0.01$. The setup is made fully unsymmetric by not placing the upper body centrally on the lower body, but instead moving it a distance $d = 0.02$ in negative y -direction.

Dirichlet conditions are applied as follows: The lower right surface (A) is completely fixed, whereas the upper right surface (B) is only fixed in the xy -plane and given a prescribed displacement $u_z = -0.1$ in z -direction. The lower left surface (C) is fixed in the yz -plane, while the

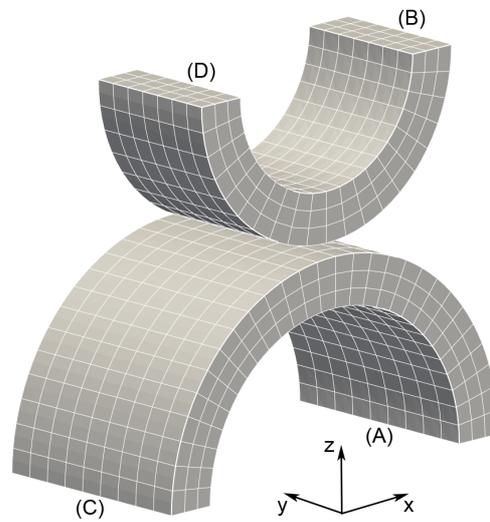


Figure 4.21: Sliding contact – problem setup and finite element mesh.

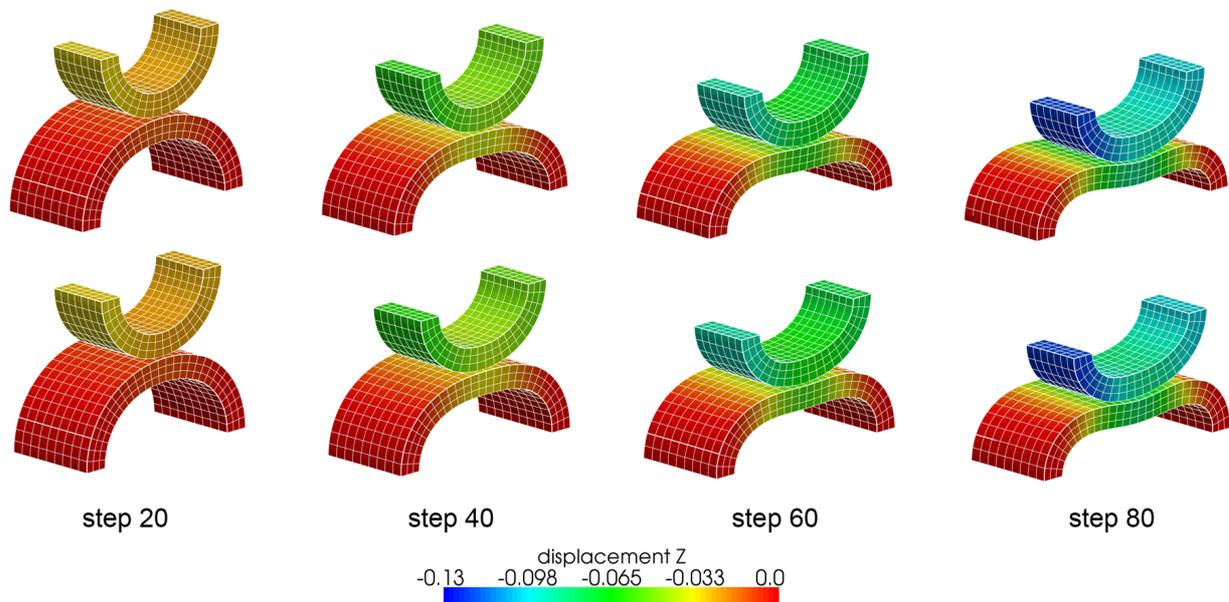


Figure 4.22: Sliding contact – deformation for the frictionless case (top) and for the frictional case with $\mathfrak{F} = 0.3$ (bottom), where the numerical solution for the displacement u_z is shown.

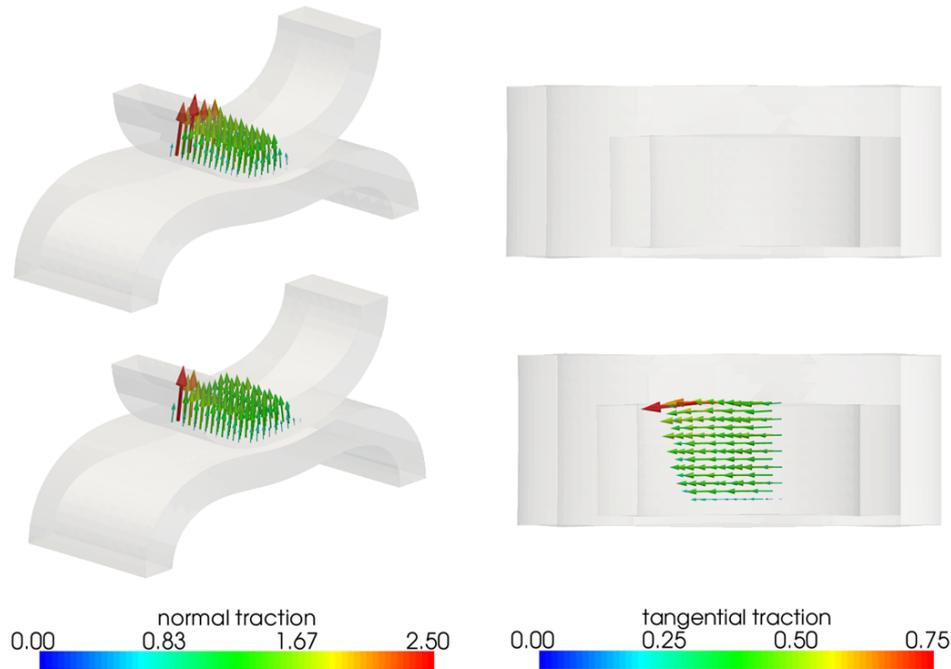


Figure 4.23: Sliding contact – normal and tangential contact tractions for the frictionless case (top) and for the frictional case with $\mathfrak{F} = 0.3$ (bottom).

upper left surface (D) is fixed in y -direction and given a prescribed displacement $u_z = -0.12$ in z -direction. Both surfaces (C) and (D) can move freely in x -direction. The prescribed displacements are applied in 80 quasistatic time steps and a compressible Neo–Hookean constitutive model is used for both bodies, with the material parameters $E^{(1)} = 120$, $\nu^{(1)} = 0.3$, $E^{(2)} = 60$ and $\nu^{(2)} = 0.25$. Figure 4.21 also illustrates the employed finite element mesh consisting of 20-node hexahedral elements. Moreover, the discrete dual Lagrange multipliers introduced in Section 3.3.2.4 are used.

The Coulomb friction case with a coefficient of friction $\mathfrak{F} = 0.3$ and the frictionless case are compared. Numerical results and some characteristic stages of deformation for both scenarios are shown in Figure 4.22. Due to the tangential forces in the contact zone, there is considerably less relative sliding in the contact zone for the frictional case. Unfortunately, apart from slight differences in the final deformed configurations, this effect can hardly be seen in Figure 4.22. Thus, an additional visualization of the normal and tangential contact tractions is provided in Figure 4.23, which illustrates the frictional effects in the contact interface more clearly.

The numerical efficiency of the semi-smooth Newton type PDASS is evaluated by monitoring the relative L^2 -norm of the total residual $\|\mathbf{r}\|^i / \|\mathbf{r}\|^0$ in iteration step i of a representative time step in Figure 4.24. Newton convergence results demonstrate that the PDASS allows for a very efficient solution of the considered fully nonlinear contact problem. It also becomes clear from Figure 4.24 that contact with Coulomb friction is apparently slightly harder to solve than the frictionless case and thus requires some more active set iterations as well as semi-smooth Newton steps. However, in all cases the correct active set and now also the correct stick and slip regions are found within only a few iteration steps. With the sets being fixed, the nonlinear iteration scheme again reduces to a standard (smooth) Newton method, and thus quadratic convergence

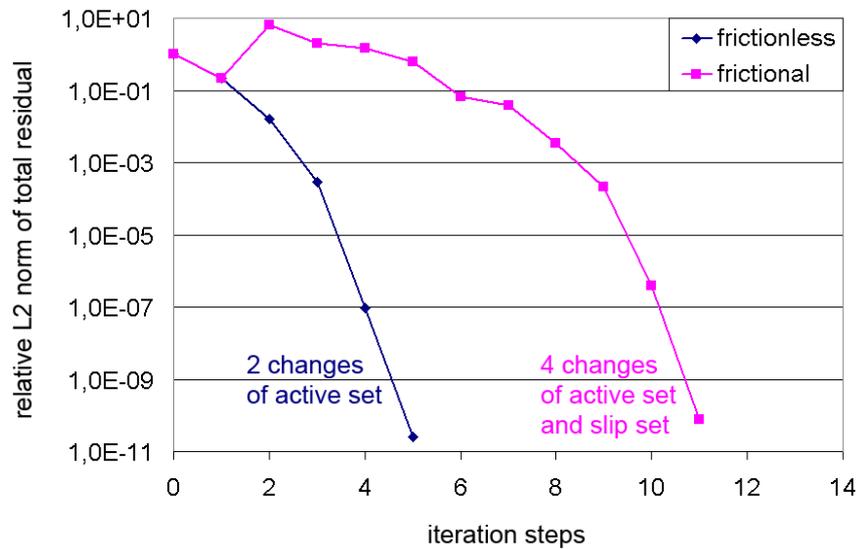


Figure 4.24: Sliding contact – convergence behavior of the semi-smooth Newton method in terms of the relative L^2 -norm of the total residual for a representative time step.

is obtained in the limit owing to the underlying consistent linearization. For further numerical examples focusing on frictional sliding within the mortar contact scheme proposed in this thesis, the interested reader is referred to Gitterle [54].

4.7.5 Second-order interpolation – Torus impact

This numerical example illustrates the robustness of second-order mortar finite element methods in 3D and associated dual Lagrange multipliers as proposed in Section 3.3.2.4 in the framework of finite deformation contact with significant active set changes. The considered test setup consists of a hollow half-torus (Neo–Hookean material model with Young’s modulus $E = 100$, Poisson’s ratio $\nu = 0.3$) and a rigid planar surface. The major and minor radii of the half-torus are 76 and 24, respectively, and the wall thickness is 4.5. The bottom surfaces of the half-torus are completely fixed, and an impact situation is generated by moving the rigid wall towards the elastic body with a prescribed displacement $u = 50$ in y -direction accumulated over 50 quasistatic time steps. Figure 4.25 shows the employed FE mesh consisting of 20-node hexahedral elements (with 50,720 nodes in total) as well as some characteristic stages of deformation. Shortly after the final step shown here, self contact occurs on the inside of the hollow torus.

The evolution of the active contact zone and the contact traction distribution can be tracked in Figure 4.26. Particularly interesting here is the fact that, while the actual load transfer is mainly restricted to a narrow region close to the boundary of the contact zone, the inner part of the contact zone nevertheless remains almost entirely active. However, the contact traction magnitudes in the interior of the active contact zone are very small as compared with the boundary regions. Considering the typical structure of the discrete KKT type contact conditions in (4.44), where *either* the weighted gap *or* the normal contact traction is forced to zero, the given situation can be considered extremely challenging for the employed active set strategy. However, as Table 4.2 exemplarily confirms for one representative time step, the semi-smooth Newton type

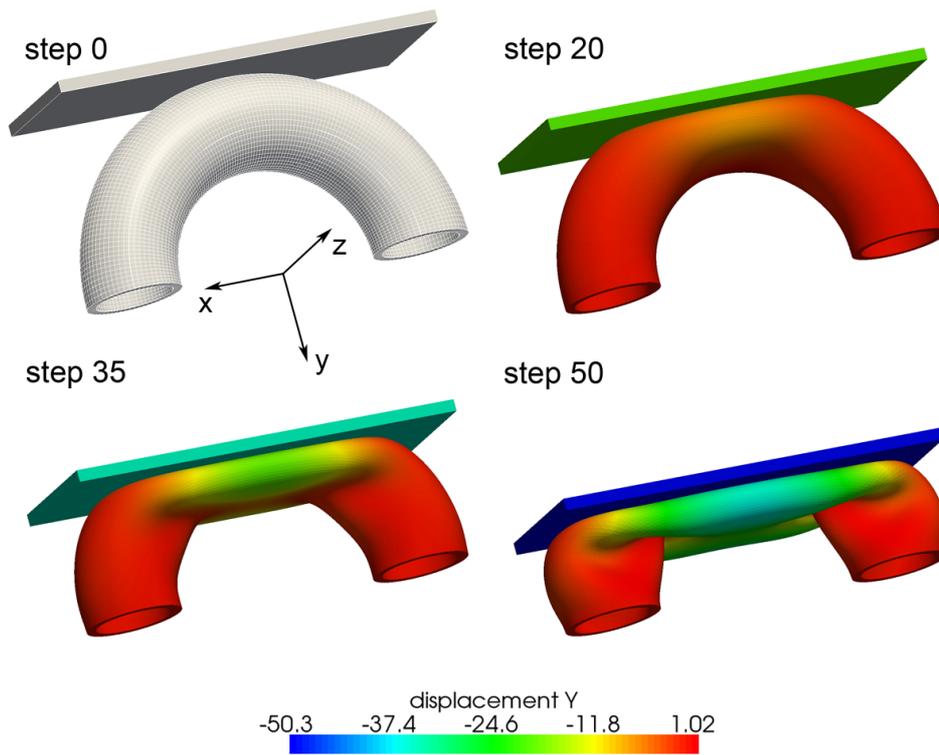


Figure 4.25: Torus impact – finite element mesh and characteristic stages of deformation.

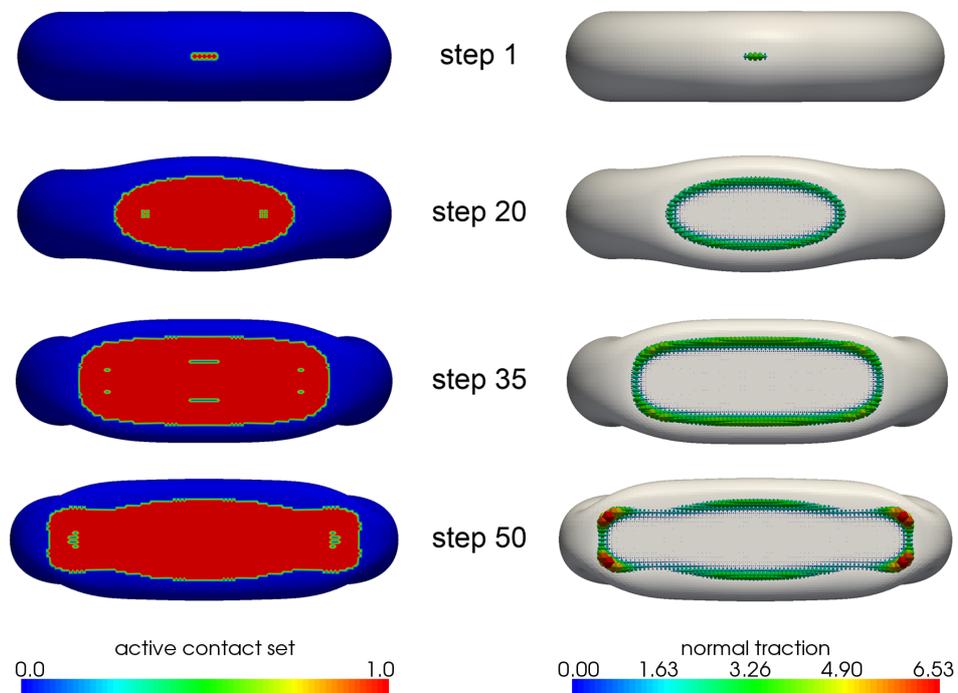


Figure 4.26: Torus impact – top view of the torus structure with visualization of the active contact set (left, 1=active) and of the normal contact traction solution (right).

Table 4.2: Torus impact – convergence behavior of the semi-smooth Newton method in terms of the relative L^2 -norm of the total residual for a representative time step.

Step	relative L^2 -norm of residual
1	7.31e+01 (*)
2	6.67e+01 (*)
3	3.54e+01 (*)
4	8.16e+00 (*)
5	4.76e−01
6	8.34e−05
7	9.66e−09

(*) = change in active contact set

active set strategy presented in this thesis does not have any problems with the described situation, but resolves all nonlinearities (including the search for the correct active set) within only a few Newton–Raphson steps. Again, owing to the underlying consistent linearization, quadratic convergence is obtained in the limit.

4.7.6 Conservation properties – Two rings

As explained in Sections 4.2.5 and 4.2.6, exact algorithmic energy and momentum conservation is an intricate topic for unilateral contact simulations, when using common implicit time integration methods. This is due to the inherent non-smoothness of the underlying physical effects. While not in the focus of this thesis, it has been pointed out that there nevertheless exist several appealing ideas in the literature how to resolve these issues. One widely used approach, putting an emphasis on exact *energy* conservation, is the so-called velocity update method (VUM) introduced by Laursen [107] in the context of node-to-segment contact algorithms. A straightforward extension to mortar-based contact formulations has been proposed in Hartmann [65] and Hartmann et al. [66]. The VUM is based in the observation that there appears a discontinuity of velocities in an impact event. This discontinuity cannot be captured accurately when a continuous behavior of the time derivatives (i.e. velocities and accelerations) is assumed, which is typically done within standard (smooth) time integrators such as the generalized- α method or the GEMM used in this thesis. Therefore, the velocity update method introduces a post-processing step at the end of each time step, which adds a contact specific correction term to the computed discrete velocities \mathbf{v}_{n+1} such that the total energy of the considered system is conserved. While the fundamental concept of the VUM is rather simple and very efficient, it nevertheless interferes with the underlying time discretization and thus the resulting numerical algorithm loses the property of second-order accuracy in time, see Hesch and Betsch [73, 74]. All details concerning the derivation of the VUM can readily be found in the original contribution in Laursen and Love [109]. Within this thesis, the VUM is only implemented for comparison purposes as presented in the following.

Before considering the actual example of interest in this paragraph, i.e. the impact of two rings, an academic one-element example is discussed. The setup has been inspired by a similar, but only two-dimensional example presented in Hartmann et al. [66]. A 3D cube with edge length 1.0 (St.-Venant–Kirchhoff material with $E = 10^7$, $\nu = 0.4$ and $\rho_0 = 950$) is accelerated

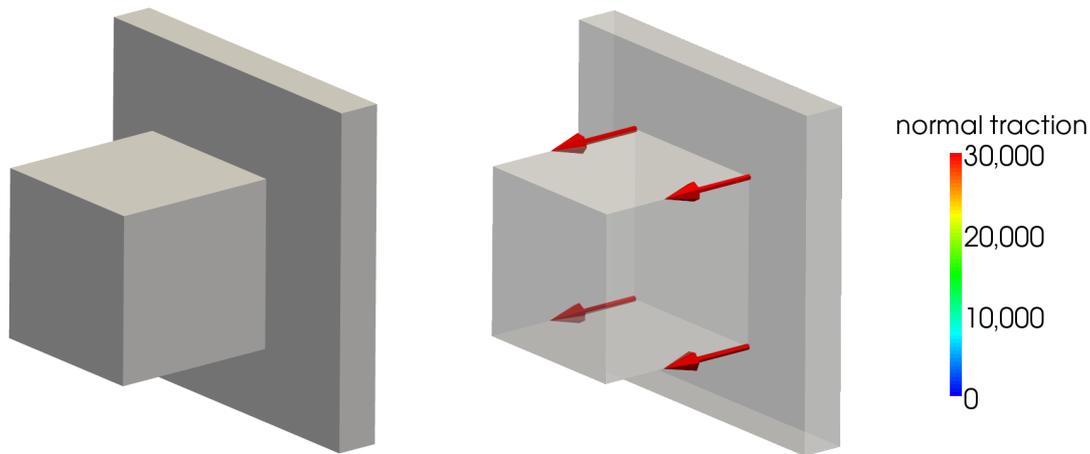


Figure 4.27: One-element example for velocity update method – problem setup (left) and exemplary visualization of normal contact tractions during impact (right).

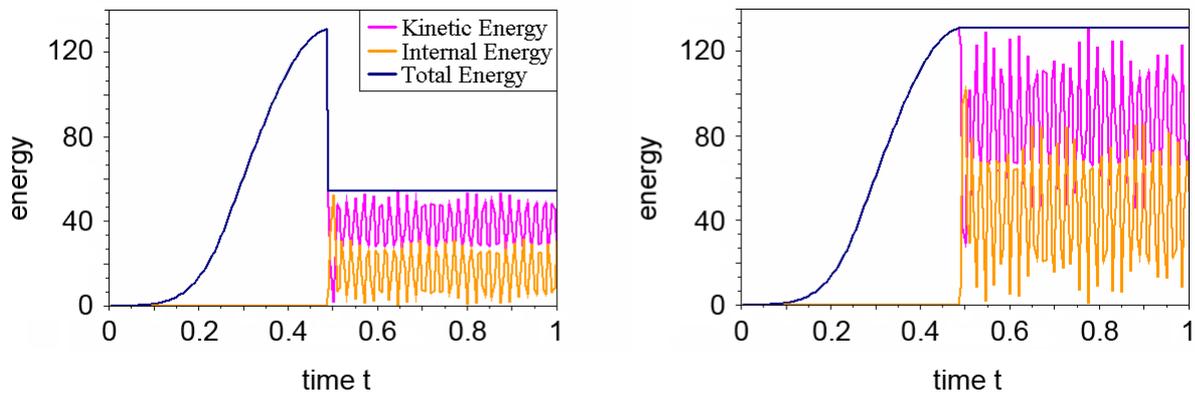


Figure 4.28: One-element example for velocity update method – loss in total energy without the VUM (left) and exact conservation of total energy when using the VUM (right).

towards a rigid wall as illustrated in Figure 4.27 due to a body force acting between $t = 0$ and $t = 0.5$ (see also Hartmann [65] for more details). Time discretization is based on the energy-momentum method (EMM) with a constant time step size $\Delta t = 0.005$ and no numerical dissipation. The cube hits the wall with a kinetic energy $E_{\text{kin}} = 131.6$. Since no external forces or moments are applied after the initial acceleration phase, the total energy must be conserved during the following impact event. However, as can be seen from Figure 4.28, the EMM time integrator fails to achieve algorithmic energy conservation but instead suffers from a severe loss of energy exactly in the instance of contact. Only when adding the velocity update procedure at the end of each time step, the spurious energy loss vanishes and exact algorithmic energy conservation is guaranteed instead (see again Figure 4.28).

After this short interlude and very basic validation of the concept of the VUM, the impact of two thin-walled rings as described in Laursen and Love [109] for a penalty-based NTS approach is investigated. The initial problem setup is illustrated in Figure 4.29, with both rings having an inner diameter $d_i = 19.4$, an outer diameter $d_o = 20$ and a thickness $t = 1$. The distance between

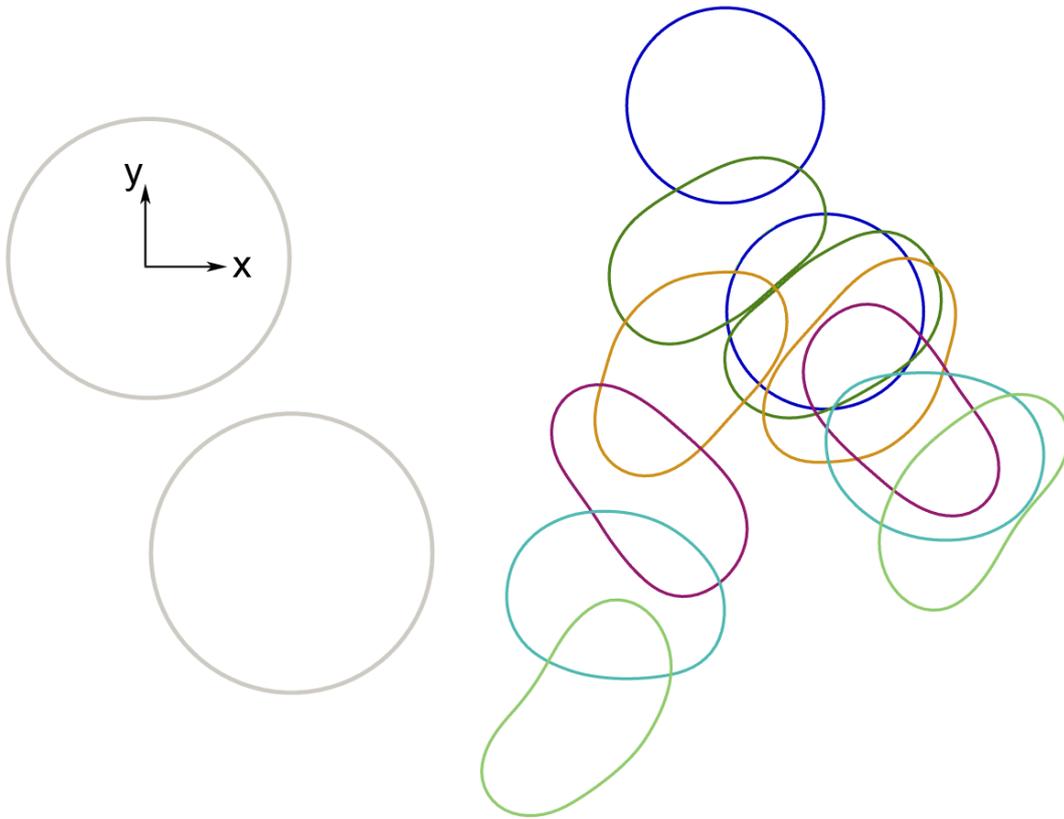


Figure 4.29: Two rings – problem setup (left) and characteristic stages of deformation (right). Visualizations are given for the following points in time: $t = 0$ (blue), $t = 4$ (dark green), $t = 8$ (orange), $t = 12$ (purple), $t = 16$ (turquoise) and $t = 20$ (light green).

the centers of the two rings is 10 in x -direction and 20 in y -direction and the upper ring is given an initial velocity $v_0 = -4.0$ in y -direction. For constitutive modeling, a St.-Venant–Kirchhoff law with Young’s modulus $E = 1000$, Poisson’s ratio $\nu = 1/6$, density $\rho_0 = 0.1$ and a plane strain assumption is employed. Each ring is discretized with 234 four-node quadrilateral (*quad4*) elements and time integration is based on the EMM with a constant time step size $\Delta t = 0.1$. During the numerical simulation, no external forces or moments are acting on the two rings and no sources of dissipation (e.g. frictional sliding or material damping) are present. Thus, linear and angular momentum as well as the total mechanical energy should be conserved by the space and time discretized finite element model.

Some characteristic stages of deformation are also depicted in Figure 4.29, while Figure 4.30 shows the obtained history of momentum and energy over time. All three fundamental conservation properties are satisfactorily retained by the proposed mortar contact approach using dual Lagrange multiplier interpolation in combination with the energy-momentum method as implicit time integrator. As discussed at length in Section 4.2.6, linear momentum conservation is exact to machine precision owing to the fact that the same numerical integration procedure is used for both discrete mortar matrices \mathbf{D} and \mathbf{M} . In contrast to the mesh tying example discussed in Section 3.5.3, angular momentum conservation of the proposed mortar formulation cannot be guaranteed for the *unilateral* contact case considered here. However, as already explained in

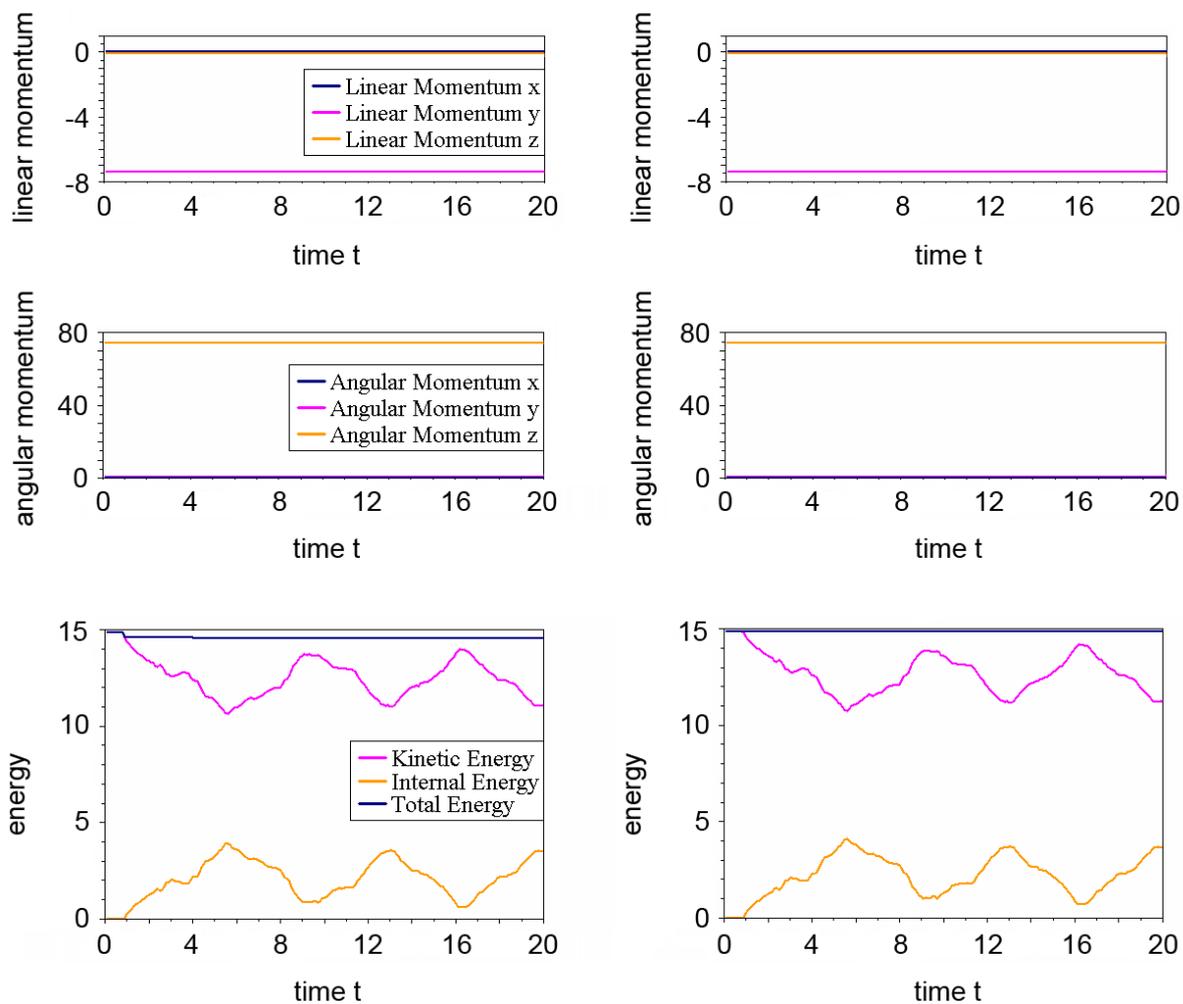


Figure 4.30: Two rings – conservation of linear momentum and angular momentum but loss in total energy without the VUM (left) and conservation of linear momentum, angular momentum and total energy when using the VUM (right).

Section 4.2.6, the occurring deviation from exact angular momentum conservation is often negligible from an engineering point of view (or, as here, not even noticeable). Again, Figure 4.30 illustrates that exact algorithmic energy conservation is only assured when applying the velocity update method. Without the VUM, around 2% of the total energy are lost during the first impact event in this numerical example.

As mentioned at the beginning of this paragraph, the issues of energy and momentum conservation are only discussed for the sake of completeness within this thesis, but no novel approach to solving these problems is proposed. For a very comprehensive overview of the state-of-the-art in this field, the interested reader is referred to Hesch and Betsch [71, 73, 74]. While energy-momentum conserving time integration schemes for unilateral contact are definitely an important and worthwhile research direction, the numerical results obtained here also demonstrate that there exist many practical applications for which such modifications are not absolutely necessary in order to obtain a reasonable engineering accuracy of numerical solutions.

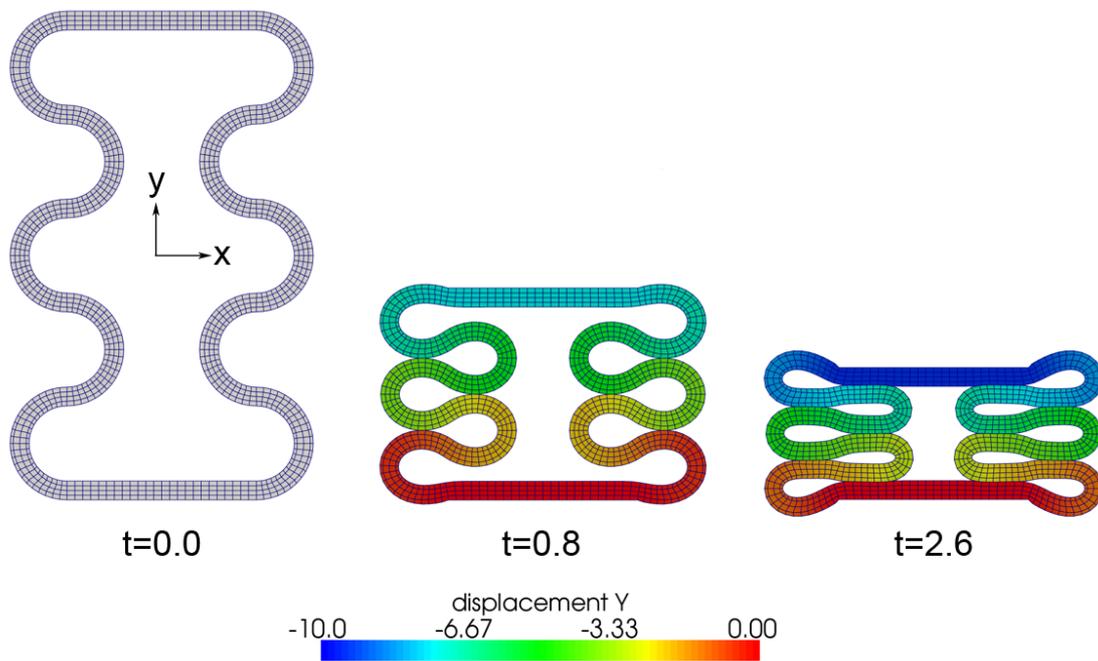


Figure 4.31: Folding structure – finite element mesh and characteristic stages of deformation.

4.7.7 Self contact – Folding structure

The simple 2D example of a folding structure coming into self contact is analyzed in the following. While no detailed quantitative investigations are performed, the obtained results nevertheless provide a qualitative proof of concept for the self contact formulation and associated search algorithms presented in Section 4.6.3. The considered 2D structure and an exemplary finite element mesh are illustrated in Figure 4.31. A two-dimensional version of the Neo–Hookean model based on a plane strain assumption is used, with the material parameters set to $E = 15,000$ and $\nu = 0.125$ and the density $\rho_0 = 0.125$. Time discretization is based on the generalized- α method with a fixed time step size $\Delta t = 0.005$ and no numerical dissipation, i.e. $\rho_\infty = 1.0$. While the bottom surface is completely constrained, a prescribed displacement is applied to the top surface, thus gradually compressing the structure.

Characteristic snapshots of the resulting deformation are also shown in Figure 4.31. With advancing simulation time, local self contact zones develop on both the inner and outer surface of the folding structure. As discussed in Section 4.6.3, it is not possible to define slave and master sides of these self contact zones a priori, but such an assignment must rather be made dynamically as part of the self contact search algorithm. Figure 4.32 visualizes the resulting normal contact tractions for the final deformation state, thus again highlighting the multitude of self contact patches. While not shown here, detailed time measurements and mesh refinement studies have been carried out for several representative two body contact as well as self contact examples. In all cases, the algorithm complexity has been found to be in good agreement with theoretical predictions, see Yang and Laursen [186, 187], and thus the computational costs associated with (self) contact search become negligible from an engineering perspective.

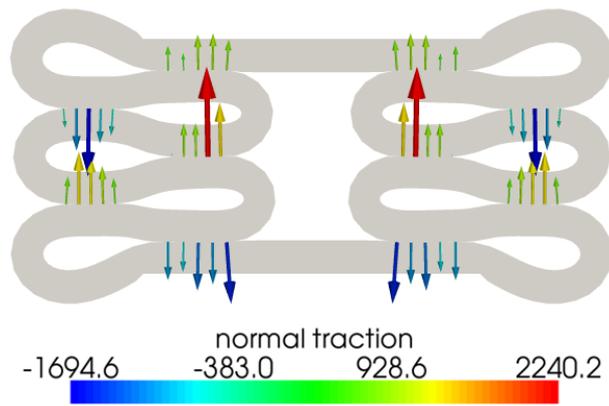


Figure 4.32: Folding structure – normal contact tractions for the final deformation state.

4.7.8 Multiple bodies – Elastic particle system

The special scenario of multiple bodies coming into contact is exemplified with an elastic particle system consisting of 200 two-dimensional rings in the following. Similar to the last paragraph on self contact, all previously described properties of the employed mortar-based contact formulation also apply here. Again, the obtained numerical results rather serve as a proof of concept for the parallel search algorithms and the dynamic assignment of slave and master sub-surfaces presented in Section 4.6.3. The considered setup consists of 200 rings based on the plane strain assumption and a Neo–Hookean constitutive law with the parameters $E = 10.5$, $\nu = 0.3$ and $\rho_0 = 7.83e-06$. The rings are arranged in a rectangle and are subject to gravitation, inducing an acceleration in negative y -direction towards a rigid wall. Time discretization is based on the generalized- α method with a time step size $\Delta t = 0.0005$ and a spectral radius $\rho_\infty = 0.9$.

The initial configuration and some characteristic stages of deformation are shown in Figure 4.33. The multitude of contact interactions between the individual bodies in combination with significant deformations introduce extremely strong nonlinearities into the simulation. As a consequence, the active contact set continuously and in parts dramatically changes from time step to time step. Nevertheless, the semi-smooth Newton method proposed as nonlinear solution scheme in Section 4.4 remains stable and yields an excellent convergence behavior in all considered time steps. To further illustrate the complexity of the occurring contact interactions, an exemplary visualization of the normal contact tractions is given in Figure 4.34.

Altogether, the results in this paragraph imposingly demonstrate that mortar methods and semi-smooth Newton solution schemes for finite deformations are not only well-suited for classical two body contact, but just as well for contact of multiple bodies. While particle systems are today most commonly analyzed based on the assumption of rigid or linear elastic bodies, e.g. using the discrete element method (DEM), the fully nonlinear finite element approach presented here opens up an entirely new field of applications, namely that of highly deformable particle systems. One particularly interesting example of such systems is the flow of red blood cells through narrow capillaries or microfluidic devices, see e.g. Bow et al. [18].

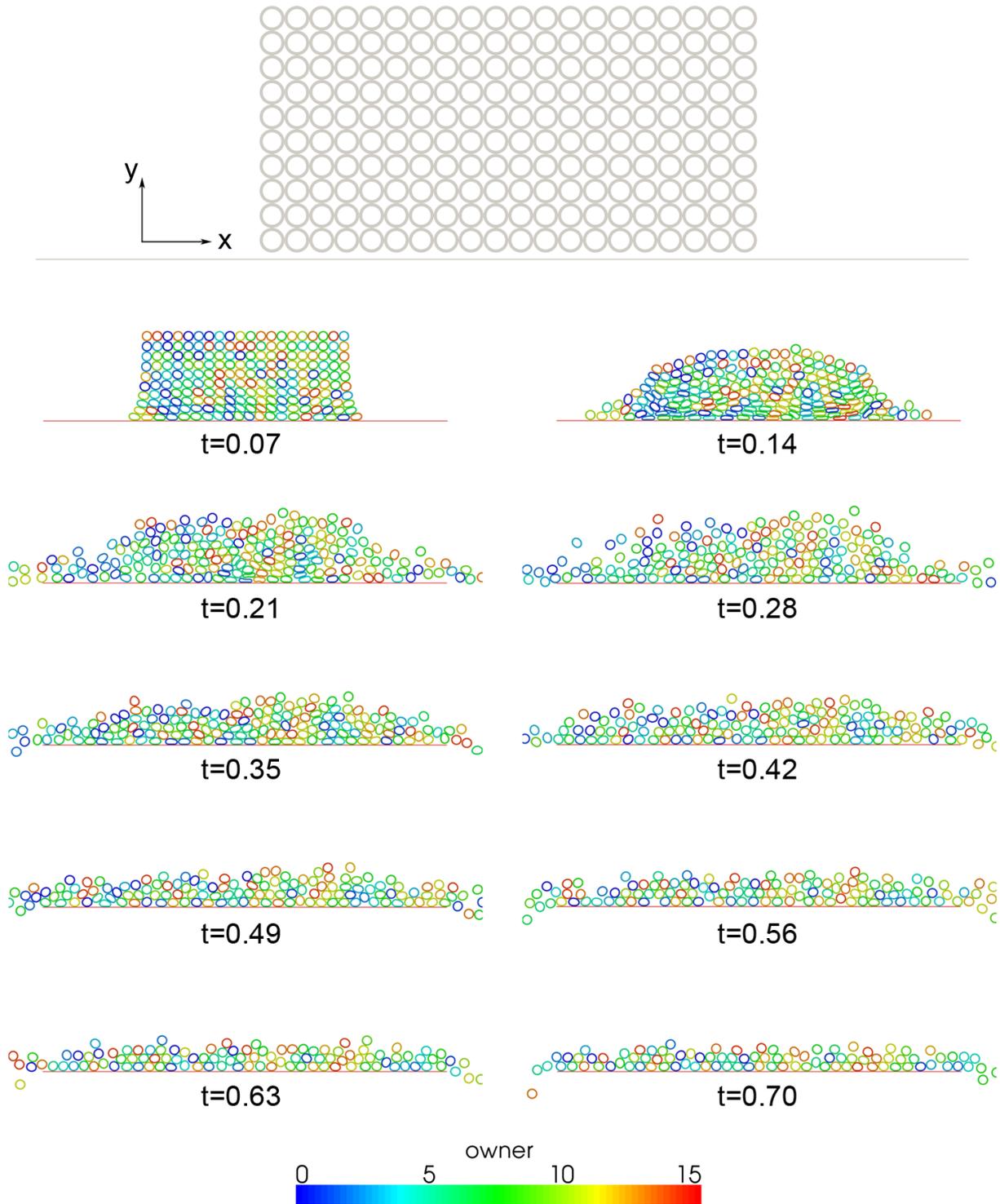


Figure 4.33: Elastic particle system – initial configuration, characteristic stages of deformation and exemplary parallel distribution among 16 independent processors.

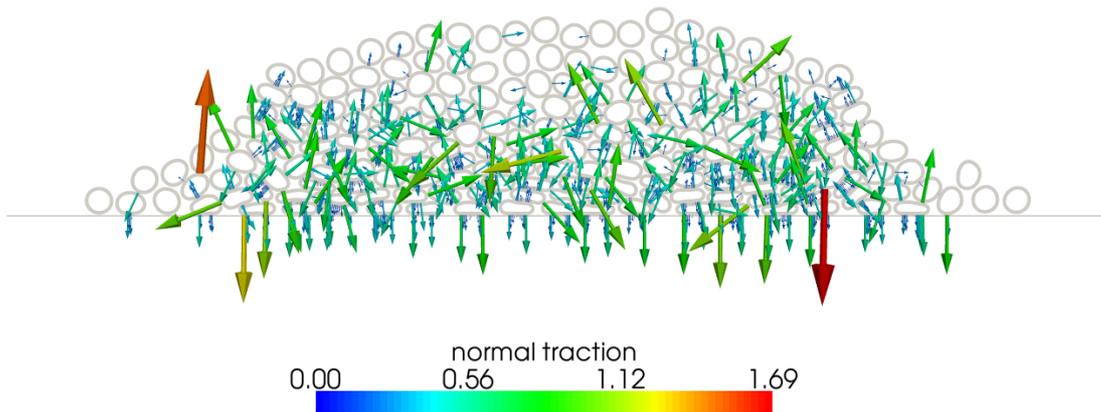


Figure 4.34: Elastic particle system – normal contact tractions at $t = 0.14$.

4.7.9 Large-scale simulations – Two torus impact

The final numerical example presented in this chapter demonstrates the applicability of the proposed mortar contact formulations, including the parallel search algorithms and dynamic load balancing strategies described in Section 4.6, for large-scale simulations on parallel high-performance computing (HPC) systems. The investigated setup, illustrated in Figure 4.35, consists of two thin-walled tori with a Neo–Hookean material model ($E = 3000$, $\nu = 0.3$, $\rho_0 = 0.1$). The major and minor radius of the two hollow tori is 76 and 24, respectively, and the wall thickness is 4.5. The lower torus lies in the xy -plane and the upper torus is rotated around the y -axis by 45 degrees. Both the chosen geometry and loading conditions are inspired by a very similar analysis presented in Yang and Laursen [187] to evaluate contact search strategies. Transient structural dynamics using a generalized- α time integration scheme are considered for the solution within 500 time steps and a constant time step size $\Delta t = 0.02$. As can be seen from the exemplary snapshots of deformation in Figure 4.35, the lower torus is first accelerated towards the upper torus by a body force and then a very general oblique impact situation with large structural deformations occurs.

The finite element mesh for this 3D impact model involves 4,255,360 first-order hexahedral (*hex8*) elements and 13,994,880 degrees of freedom in total, with both slave and master surfaces consisting of 204,800 contact elements each. The numerical solution is performed in parallel on 120 processors within an overall simulation time of approximately 48 hours. For the simulation results presented here, the penalty regularized version of the mortar approach as described in Section 4.5.5 along with standard Lagrange multiplier interpolation has been used. The penalty parameter is chosen as $\epsilon = 1000$. However, it is important to point out that this choice of the solution method is by no means motivated by restrictions of the dual mortar schemes mainly advocated in this thesis, but rather due to linear solver issues. The GMRES solver with an ILU preconditioner employed here could unfortunately not be applied to the dual Lagrange multiplier version with its condensed system matrix (4.95). Nevertheless, the parallel efficiency of the remaining aspects of the proposed mortar formulations, e.g. the numerical integration of the mortar matrices and weighted gaps in 3D as well as the contact search and parallel redistribution methods, can successfully be proven with this numerical example.

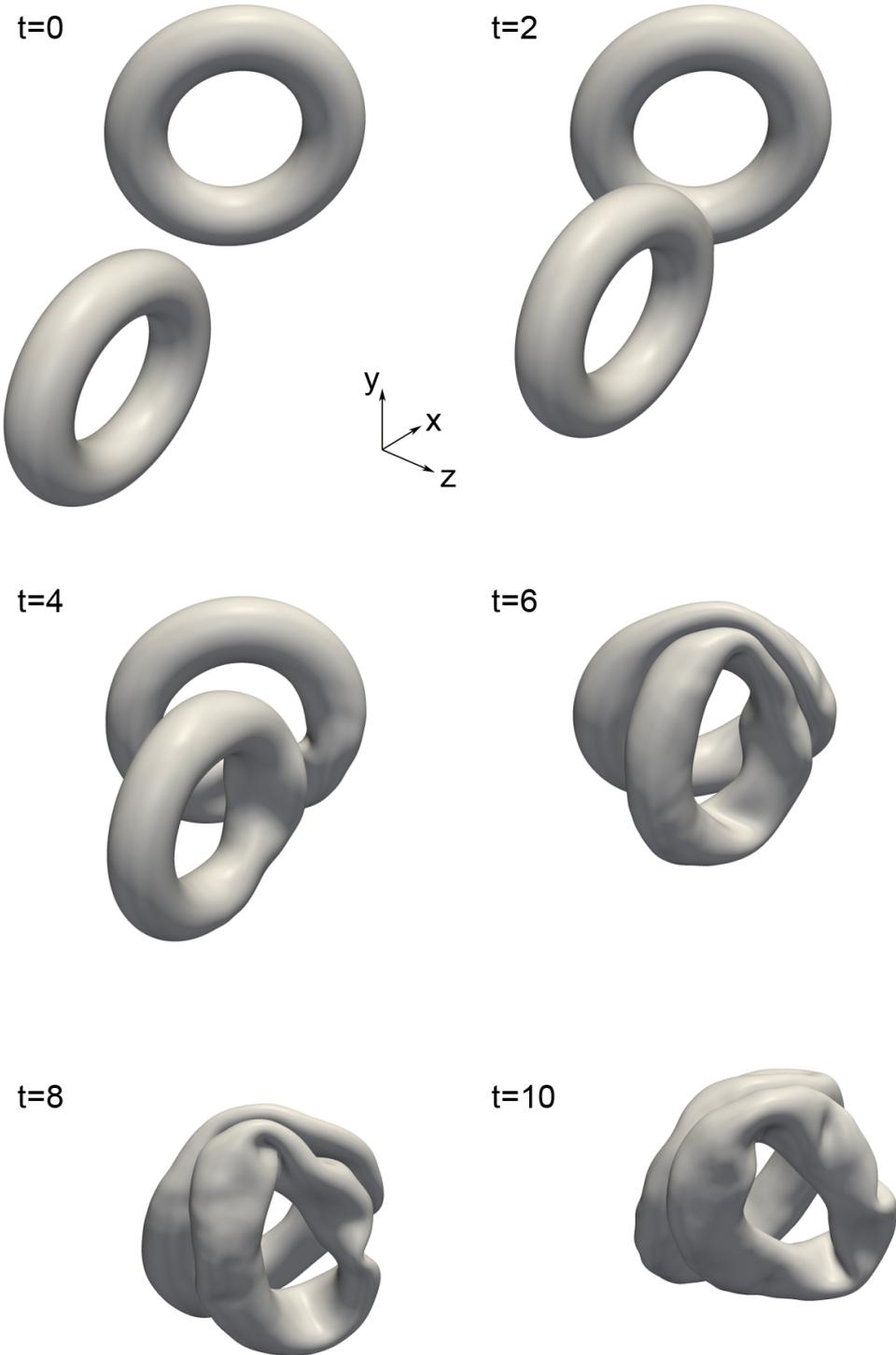


Figure 4.35: Two torus impact – initial configuration and characteristic stages of deformation.

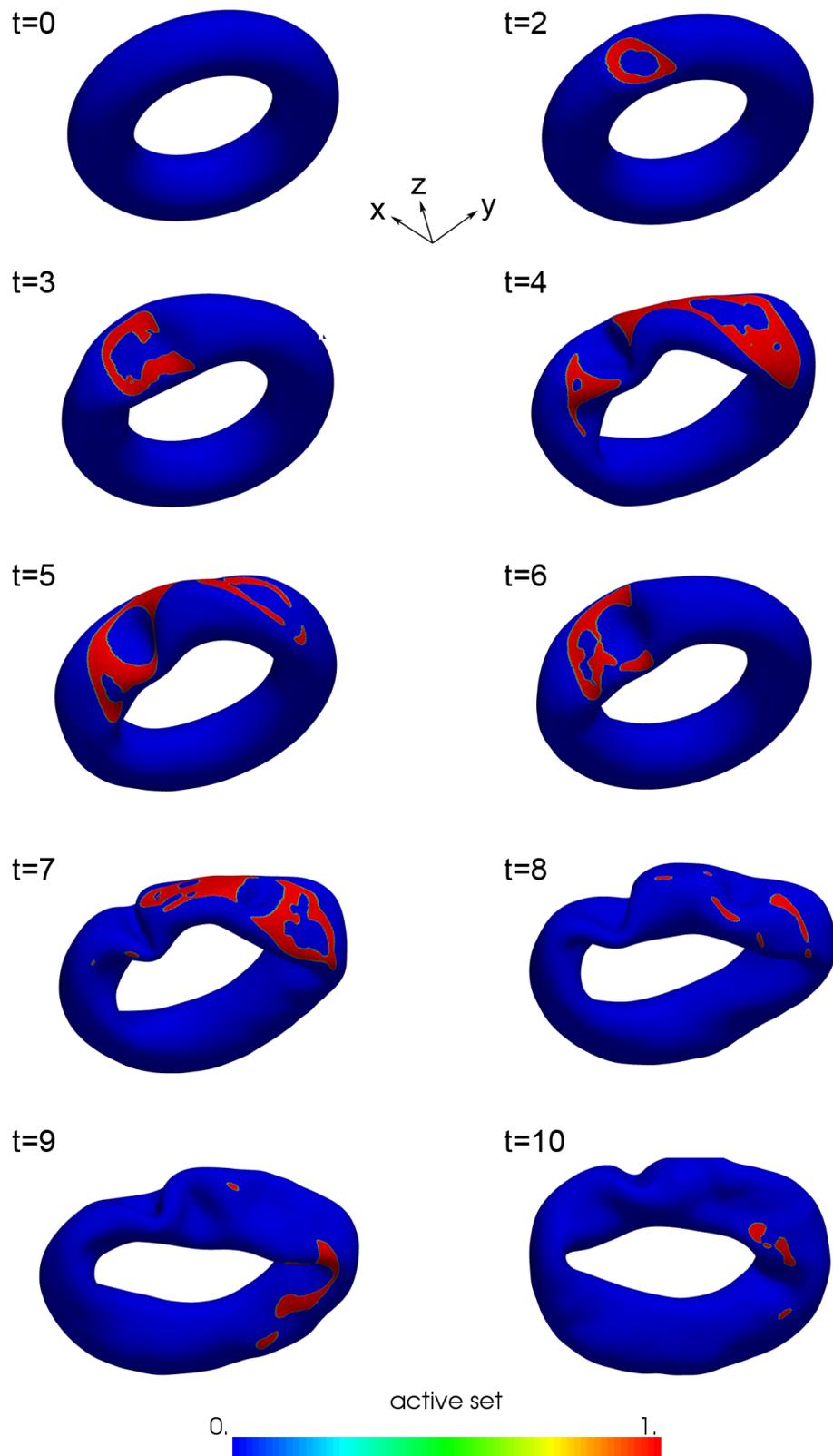


Figure 4.36: Two torus impact – lower torus and visualization of the active contact set (1=active).

Figures 4.36 and 4.37 further illustrate the complexity of the considered simulation model with severe changes of the active contact set and an extremely fine mesh resolution. While there always remains room for improvements of the parallel efficiency (e.g. with respect to efficient linear solvers, see Chapter 6), the results nevertheless strikingly emphasize that the implementation devised within this thesis is already very mature in this regard.

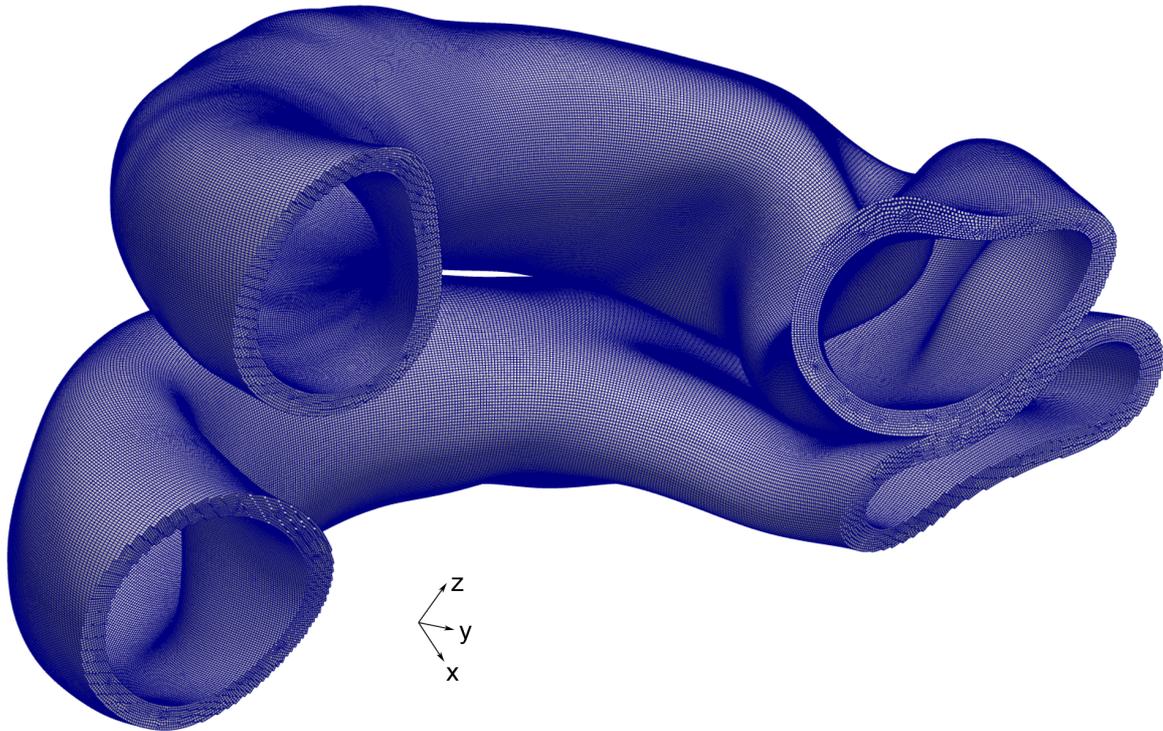


Figure 4.37: Two torus impact – exemplary cut through the contact zone at time $t = 4$ and visualization of the finite element mesh.

5 Mortar Methods for Fluid Mechanics and Fluid-Structure Interaction

The focus of this thesis lies on the development of mortar finite element methods for finite deformation solid mechanics (i.e. mesh tying) and contact mechanics in particular. Nevertheless, the developed methods and numerical algorithms can also be beneficially used for the simulation of other physical fields in computational science and engineering. The following chapter gives an overview of several such applications, including both single-field and coupled problems. Each of the three given examples provides a promising basis for the efficient use of the methods presented in Chapters 3 and 4 for challenging applications in fluid mechanics or fluid-structure interaction.

First, the coupling of subdomains with non-matching grids in fluid mechanics is investigated in Section 5.1. The presented approach is based on a variational multiscale finite element framework for incompressible flow in combination with mortar methods and dual Lagrange multipliers for the treatment of interface constraints, see also Ehrl et al. [35].

However, mortar methods are also well-suited as coupling scheme within numerical algorithms for multiphysics applications. Exemplarily, a mortar-based treatment of non-matching interfaces in fluid-structure interaction (FSI), as proposed in Klöppel et al. [96], will be analyzed in Section 5.2. The resulting FSI formulation employs an Arbitrary Lagrangian Eulerian (ALE) description of the flow field to accommodate moving fluid grids.

With regard to computational contact mechanics, the previous chapter put a strong emphasis on method development and robust numerical algorithms, but possible extensions towards more complex physical phenomena at contact interfaces, such as adhesion, wear or lubrication, have hardly been addressed so far. Thus, in Section 5.3, a novel computational approach for the simulation of contact interaction within the context of fluid-structure interaction will be presented, see also Mayer et al. [115]. From a methodological point of view, this approach is based on a fixed-grid FSI framework using the extended finite element method (XFEM), into which a mortar contact formulation is algorithmically integrated. Possible applications of the resulting fluid-structure-contact interaction (FSCI) scheme cover the whole range of wet contact phenomena, most importantly elastohydrodynamic lubrication.

5.1 Subdomain coupling in fluid mechanics

For many applications in different fields of engineering and applied sciences, fluid systems in relatively large computational domains have to be investigated. While most of the domain can often be discretized with a rather coarse discretization length without jeopardizing the overall solution quality, a rather small characteristic discretization length is required locally, for instance due to boundary layers that need to be resolved. An adequate resolution of such boundary layers is usually linked with high computational cost. Therefore, it is desirable to develop efficient methods

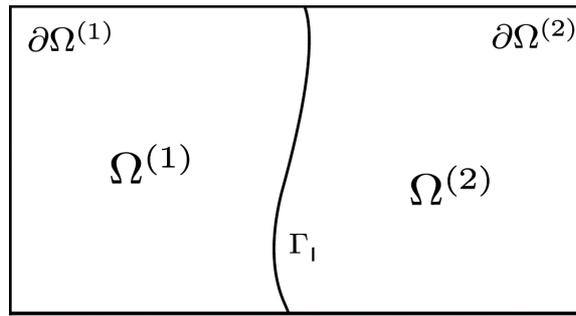


Figure 5.1: Decomposition into two subdomains $\Omega^{(1)}$ and $\Omega^{(2)}$ connected by the interface Γ_1 .

enabling the use of different discretizations for boundary layer regions and the bulk of the flow domain. Splitting the fluid domain into subdomains coupled via internal fluid-fluid interfaces offers several advantages. On the one hand, the generation of meshes for complex configurations is simplified, since independently discretized subdomains may be assembled. The application of this concept enables the adaption of individual partial meshes without the need for modifying the other ones. On the other hand, the resolution of the boundary layers for flow problems can be improved, for example by using finer discretizations or higher-order interpolation functions locally. Such locally restricted mesh adjustments are expected to improve the results without disproportionately raising the computational cost.

Mortar methods as discussed in the previous chapters provide a very convenient framework for subdomain coupling in fluid mechanics, which will be outlined in the following paragraphs. After briefly describing the underlying problem statement, an overview of variational multiscale and stabilized finite element methods for incompressible flow problems is given, see also Hughes et al. [88]. The variational multiscale formulation used here was basically proposed in Gravenmeier and Wall [58] in the more general context of variable-density flow. Mortar coupling itself at the non-matching fluid-fluid interfaces is not discussed in detail, as all relevant aspects have already been introduced in Chapter 3 and directly carry over from solid mechanics to fluid mechanics. Again, dual Lagrange multipliers (see Section 3.3) play a key role in the derivation of efficient solution algorithms owing to the fact that they allow for the localization of interface constraints. Finally, numerical results for a 3D benchmark example are provided.

5.1.1 Problem statement

The polygonally shaped and bounded domain $\Omega \subset \mathbb{R}^{\text{ndim}}$, where $\text{ndim} \leq 3$ is the number of spatial dimensions, is decomposed into subdomains $\Omega^{(k)}$ as shown in Figure 5.1 for $k \in [1, 2]$. Only decompositions based on two subdomains are considered here for the sake of simplicity. However, an extension to the treatment of more than two subdomains is straightforward and mostly technical. The boundary of the subdomain $\Omega^{(k)}$ is denoted by $\partial\Omega^{(k)}$ and assumed to be sufficiently smooth. As typical for fluid mechanics, a so-called Eulerian approach is followed here for the mathematical problem formulation. The Eulerian approach can maybe best be imagined by placing an observer at each fixed spatial point, thus describing the properties of the fluid particles passing by this point. In contrast to the Lagrangian approach introduced for solid me-

chanics in Section 2.1.1, this yields a fixed fluid grid in the current (spatial) configuration and the notion of reference configuration becomes obsolete.

The fluid is modeled by the incompressible Navier-Stokes equations in the spatial configuration $\Omega^{(k)} \times [0, T]$, where T is the total simulation time, in the following form:

$$\nabla \cdot \mathbf{u}^{(k)} = 0, \quad (5.1)$$

$$\rho \frac{\partial \mathbf{u}^{(k)}}{\partial t} + \rho (\mathbf{u}^{(k)} \cdot \nabla) \mathbf{u}^{(k)} + \nabla p^{(k)} - 2\mu \nabla \cdot \boldsymbol{\varepsilon}(\mathbf{u}^{(k)}) = \hat{\mathbf{f}}^{(k)}, \quad (5.2)$$

where $\hat{\mathbf{f}}^{(k)}$ is the applied body force per unit volume on subdomain $\Omega^{(k)}$. In addition, $\mathbf{u}^{(k)}$ denotes the fluid velocity, $p^{(k)}$ the pressure, ρ the constant fluid density, μ the constant dynamic viscosity, t the time, and $\boldsymbol{\varepsilon}(\mathbf{u}^{(k)})$ the symmetric strain rate tensor defined as

$$\boldsymbol{\varepsilon}(\mathbf{u}^{(k)}) = \frac{1}{2} \left(\nabla \mathbf{u}^{(k)} + (\nabla \mathbf{u}^{(k)})^\top \right). \quad (5.3)$$

Based on the partition $\partial\Omega^{(k)} = \Gamma_D^{(k)} \cup \Gamma_N^{(k)} \cup \Gamma_I$, with the three boundary regions assumed to be pairwise disjoint, Dirichlet and Neumann boundary conditions can be formulated as

$$\mathbf{u}^{(k)} = \hat{\mathbf{u}}^{(k)} \quad \text{on } \Gamma_D^{(k)} \times [0, T], \quad (5.4)$$

$$\boldsymbol{\sigma}^{(k)} \cdot \mathbf{n}^{(k)} = \hat{\mathbf{h}}^{(k)} \quad \text{on } \Gamma_N^{(k)} \times [0, T], \quad (5.5)$$

respectively. Herein, the Cauchy stress tensor $\boldsymbol{\sigma}^{(k)}$ is defined as

$$\boldsymbol{\sigma}^{(k)} = -p^{(k)} \mathbf{I} + 2\mu \boldsymbol{\varepsilon}(\mathbf{u}^{(k)}). \quad (5.6)$$

Moreover, $\hat{\mathbf{u}}^{(k)}$ is the velocity prescribed on $\Gamma_D^{(k)}$, $\mathbf{n}^{(k)}$ the unit outer normal vector to the Neumann boundary and $\hat{\mathbf{h}}^{(k)}$ the given boundary traction per unit surface area on $\Gamma_N^{(k)}$. The initial conditions are given as

$$\mathbf{u}^{(k)} = \hat{\mathbf{u}}_0^{(k)} \quad \text{in } \Omega^{(k)}, \quad (5.7)$$

where $\hat{\mathbf{u}}_0^{(k)}$ denotes a solenoidal initial velocity field. Finally, the velocity coupling condition on the interface Γ_I can be written in the following form:

$$\mathbf{u}^{(1)} = \mathbf{u}^{(2)} \quad \text{on } \Gamma_I \times [0, T], \quad (5.8)$$

Comparing (5.8) with (3.6), the similarity of the given interface constraints for subdomain coupling in fluid mechanics with the problem statement for mesh tying in solid mechanics becomes obvious. In full analogy to Section 3.1, the unknown interface tractions can be interpreted as Lagrange multiplier vector, i.e.

$$\boldsymbol{\lambda} = \boldsymbol{\sigma}^{(1)} \cdot \mathbf{n}^{(1)} = -\boldsymbol{\sigma}^{(2)} \cdot \mathbf{n}^{(2)}, \quad (5.9)$$

where also the balance of linear momentum on Γ_I has been exploited. The Lagrange multiplier vector can then be used to derive a weak formulation of (5.1)–(5.7) and the coupling condition (5.8). To do so, appropriate solution function spaces $\mathcal{S}_u^{(k)}$, $\mathcal{S}_p^{(k)}$ and \mathcal{S}_λ for $\mathbf{u}^{(k)}$, $p^{(k)}$ and $\boldsymbol{\lambda}$

as well as corresponding weighting function spaces $\mathcal{V}_u^{(k)}$, $\mathcal{V}_p^{(k)}$ and \mathcal{V}_λ for the respective weighting functions $\mathbf{v}^{(k)}$, $q^{(k)}$ and $\boldsymbol{\mu}$ are defined. This yields the following weak form: find $\mathbf{u}^{(k)} \in \mathcal{S}_u^{(k)}$, $p^{(k)} \in \mathcal{S}_p^{(k)}$, and $\boldsymbol{\lambda} \in \mathcal{S}_\lambda$ such that

$$\sum_{k=1}^2 (q^{(k)}, \nabla \cdot \mathbf{u}^{(k)}) = 0 \quad \forall q^{(k)} \in \mathcal{V}_p^{(k)}, \quad (5.10)$$

$$\begin{aligned} & \sum_{k=1}^2 \left[\left(\mathbf{v}^{(k)}, \rho \frac{\partial \mathbf{u}^{(k)}}{\partial t} \right) + \left(\mathbf{v}^{(k)}, \rho (\mathbf{u}^{(k)} \cdot \nabla) \mathbf{u}^{(k)} \right) - \left(\nabla \cdot \mathbf{v}^{(k)}, p^{(k)} \right) + \left(\boldsymbol{\varepsilon}(\mathbf{v}^{(k)}), 2\mu \boldsymbol{\varepsilon}(\mathbf{u}^{(k)}) \right) \right] \\ &= \sum_{k=1}^2 \left[\left(\mathbf{v}^{(k)}, \hat{\mathbf{f}}^{(k)} \right) + \left(\mathbf{v}^{(k)}, \hat{\mathbf{h}}^{(k)} \right)_{\Gamma_N^{(k)}} \right] + \left(\mathbf{v}^{(1)} - \mathbf{v}^{(2)}, \boldsymbol{\lambda} \right)_{\Gamma_1} \quad \forall \mathbf{v}^{(k)} \in \mathcal{V}_u^{(k)}, \quad (5.11) \end{aligned}$$

$$\left(\boldsymbol{\mu}, \mathbf{u}^{(1)} - \mathbf{u}^{(2)} \right)_{\Gamma_1} = 0 \quad \forall \boldsymbol{\mu} \in \mathcal{V}_\lambda. \quad (5.12)$$

Here, the L^2 -inner products in $\Omega^{(k)}$, on $\Gamma_N^{(k)}$ and on Γ_1 are denoted by (\cdot, \cdot) , $(\cdot, \cdot)_{\Gamma_N^{(k)}}$ and $(\cdot, \cdot)_{\Gamma_1}$, respectively. In the mathematical literature for incompressible fluid flow, some articles deal with mortar finite element methods for the weak problem formulation given in (5.10)–(5.12). For example, in Achdou et al. [1], mortar methods were proposed for constructing efficient Navier-Stokes solvers, while in Ben Belgacem [12] and Ben Belgacem et al. [14], fundamental theoretical results such as a rigorous convergence analysis were presented. However, only standard Lagrange multipliers and so-called inf-sup stable or Ladyzhenskaya–Babuška–Brezzi (LBB) stable finite elements were considered using a standard Galerkin finite element formulation. In simple terms, the LBB condition requires that the polynomial order of the pressure interpolation has to be lower than the one of the velocity interpolation, see e.g. Gresho et al. [59] and Brezzi and Fortin [19] for more details. However, as briefly described in the following paragraph, the LBB condition can be circumvented by introducing specic stabilization terms.

5.1.2 Residual-based variational multiscale approach

In the given approach, a residual-based variational multiscale method is employed instead of a standard Galerkin formulation of (5.10)–(5.12), which resolves the issue of LBB stability and allows for equal-order interpolation of velocity and pressure field. Moreover, the present formulation aims at fully exploiting the advantages of dual Lagrange multiplier interpolation. Restricting solution and weighting function spaces to finite dimensional subspaces may basically be interpreted as a projection. Using the concept of the variational multiscale method as originally proposed in Hughes [85], the unknown physical fields, i.e. the pressure $p^{(k)}$ and the velocity $\mathbf{u}^{(k)}$, are decomposed into their resolved and unresolved scales by variational projection:

$$p^{(k)} = p_h^{(k)} + \tilde{p}, \quad \mathbf{u}^{(k)} = \mathbf{u}_h^{(k)} + \tilde{\mathbf{u}}. \quad (5.13)$$

Neglecting the unresolved scales leads to the standard Galerkin FEM. However, within the variational multiscale approach employed here the unresolved scales are modeled by

$$\tilde{p} = -\tau_C \mathcal{R}_{C,h}^{(k)}, \quad \tilde{\mathbf{u}} = -\tau_M \mathcal{R}_{M,h}^{(k)}, \quad (5.14)$$

where τ_C , τ_M and $\mathcal{R}_{C,h}^{(k)}$, $\mathcal{R}_{M,h}^{(k)}$ denote the respective stabilization parameters and residuals of continuity and momentum equation. The discrete residuals in strong form are defined as

$$\begin{aligned}\mathcal{R}_{C,h}^{(k)} &= \nabla \cdot \mathbf{u}_h^{(k)} \\ \mathcal{R}_{M,h}^{(k)} &= \rho \frac{\partial \mathbf{u}_h^{(k)}}{\partial t} + \rho(\mathbf{u}_h^{(k)} \cdot \nabla) \mathbf{u}_h^{(k)} + \nabla p_h^{(k)} - 2\mu \nabla \cdot \boldsymbol{\varepsilon}(\mathbf{u}_h^{(k)}) - \hat{\mathbf{f}}_h^{(k)}.\end{aligned}\quad (5.15)$$

Many different definitions for the stabilization parameters τ_M and τ_C exist in the literature, and detailed explanations concerning their choice are beyond the scope of this thesis. The interested reader is referred to Ehrl et al. [35], where a definition inspired by Taylor et al. [155] and Whiting and Jansen [172] is employed. Instead, the focus is set on conceptual changes regarding the mortar coupling approach, which are induced by the variational multiscale method as compared with the standard Galerkin FEM. As can be reasoned from (5.11) and (5.12), the latter only couples the velocity degrees of freedom via Lagrange multipliers $\boldsymbol{\lambda}$, whereas the pressure along the internal interface is determined by the traction vector defined in (5.9). However, in the context of the variational multiscale finite element formulation, coupling of velocity *and* pressure field via individual Lagrange multipliers $\boldsymbol{\lambda}_u$ and λ_p improves the accuracy of the proposed method. This is due to parasitic contributions at the internal interface resulting from the variational multiscale finite element formulation, which are then stabilized by the additional Lagrange multiplier λ_p , see Ehrl et al. [35] for an extensive numerical validation.

Consequently, additional appropriate solution function spaces $\mathcal{S}_{\boldsymbol{\lambda}_u}$ and \mathcal{S}_{λ_p} for $\boldsymbol{\lambda}_u$ and λ_p as well as corresponding weighting function spaces $\mathcal{V}_{\boldsymbol{\lambda}_u}$ and \mathcal{V}_{λ_p} for the respective weighting functions $\boldsymbol{\mu}_u$ and μ_p are defined. Regarding the restriction of the Lagrange multiplier function spaces to finite-dimensional subspaces, e.g. $\mathcal{V}_{\boldsymbol{\lambda}_u,h} \subset \mathcal{V}_{\boldsymbol{\lambda}_u}$, all considerations on standard and dual Lagrange multipliers made in Section 3.3 directly apply. The variational multiscale finite element formulation including mortar-based treatment of non-conforming interfaces is then given as follows: find $\mathbf{u}_h^{(k)} \in \mathcal{S}_{\mathbf{u},h}^{(k)}$, $p_h^{(k)} \in \mathcal{S}_{p,h}^{(k)}$, $\boldsymbol{\lambda}_{u,h} \in \mathcal{S}_{\boldsymbol{\lambda}_u,h}$, $\lambda_{p,h} \in \mathcal{S}_{\lambda_p,h}$ such that

$$\begin{aligned}& \sum_{k=1}^2 \left[\left(q_h^{(k)}, \nabla \cdot \mathbf{u}_h^{(k)} \right) + \sum_{e=1}^{\text{nele}^{(k)}} \left(\nabla q_h^{(k)}, \tau_M \mathcal{R}_{M,h}^{(k)} \right)_{\Omega_e^{(k)}} \right] \\ &= \left(q_h^{(1)} - q_h^{(2)}, \lambda_{p,h} \right)_{\Gamma_{I,h}^{(1)}} \quad \forall q_h^{(k)} \in \mathcal{V}_{p,h}^{(k)},\end{aligned}\quad (5.16)$$

$$\begin{aligned}& \sum_{k=1}^2 \left[\left(\mathbf{v}_h^{(k)}, \rho \frac{\partial \mathbf{u}_h^{(k)}}{\partial t} \right) + \left(\mathbf{v}_h^{(k)}, \rho(\mathbf{u}_h^{(k)} \cdot \nabla) \mathbf{u}_h^{(k)} \right) - \left(\nabla \cdot \mathbf{v}_h^{(k)}, p_h^{(k)} \right) + \left(\boldsymbol{\varepsilon}(\mathbf{v}_h^{(k)}), 2\mu \boldsymbol{\varepsilon}(\mathbf{u}_h^{(k)}) \right) \right. \\ &+ \left. \sum_{e=1}^{\text{nele}^{(k)}} \left(\rho(\mathbf{u}_h^{(k)} \cdot \nabla) \mathbf{v}_h^{(k)}, \tau_M \mathcal{R}_{M,h}^{(k)} \right)_{\Omega_e^{(k)}} + \sum_{e=1}^{\text{nele}^{(k)}} \left(\nabla \cdot \mathbf{v}_h^{(k)}, \tau_C \mathcal{R}_{C,h}^{(k)} \right)_{\Omega_e^{(k)}} \right] \\ &= \sum_{k=1}^2 \left[\left(\mathbf{v}_h^{(k)}, \hat{\mathbf{f}}_h^{(k)} \right) + \left(\mathbf{v}_h^{(k)}, \hat{\mathbf{h}}_h^{(k)} \right)_{\Gamma_{N,h}^{(k)}} \right] + \left(\mathbf{v}_h^{(1)} - \mathbf{v}_h^{(2)}, \boldsymbol{\lambda}_{u,h} \right)_{\Gamma_{I,h}^{(1)}} \quad \forall \mathbf{v}_h^{(k)} \in \mathcal{V}_{\mathbf{u},h}^{(k)},\end{aligned}\quad (5.17)$$

$$\left(\boldsymbol{\mu}_{u,h}, \mathbf{u}_h^{(1)} - \mathbf{u}_h^{(2)} \right)_{\Gamma_{I,h}^{(1)}} = \mathbf{0} \quad \forall \boldsymbol{\mu}_{u,h} \in \mathcal{V}_{\boldsymbol{\lambda}_u,h}, \quad (5.18)$$

$$\left(\mu_{p,h}, p_h^{(1)} - p_h^{(2)} \right)_{\Gamma_{I,h}^{(1)}} = 0 \quad \forall \mu_{p,h} \in \mathcal{V}_{\lambda_p,h}. \quad (5.19)$$

For the sake of completeness, different stabilization terms typically introduced in the context of the residual-based variational multiscale method are briefly mentioned here without going into details. Firstly, the pressure stabilizing Petrov–Galerkin (PSPG) term $(\nabla q_h^{(k)}, \tau_M \mathcal{R}_{M,h}^{(k)})$ relaxes the incompressibility condition and thus allows for equal-order interpolation of velocity and pressure field. Secondly, a stabilization of convection dominated problems is achieved with the streamline upwind Petrov–Galerkin (SUPG) term $(\rho(\mathbf{u}_h^{(k)} \cdot \nabla) \mathbf{v}_h^{(k)}, \tau_M \mathcal{R}_{M,h}^{(k)})$. Finally, a so-called grad-div term $(\nabla \cdot \mathbf{v}_h^{(k)}, \tau_C \mathcal{R}_{C,h}^{(k)})$ is introduced to enhance discrete mass conservation. All additional terms are defined as sums over all element domains $\Omega_e^{(k)}$, where $n_e^{(k)}$ denotes the number of elements in the discretization of the subdomain k . Detailed investigations of stabilization techniques, including an adequate choice of the associated stabilization parameters, can be found e.g. in Förster [43] and Wall [169] and the references therein.

Details on mortar interface discretization, time integration, consistent linearization and the resulting algebraic representation of (5.16)–(5.19) are omitted here, and the reader is referred to Ehrl et al. [35] instead. However, it is important to point out that the final linear system of equations to be solved within each nonlinear iteration step of each time step takes the same saddle point structure as the mesh tying system for solid mechanics discussed in Section 3.4.2. Owing to the application of dual Lagrange multiplier interpolation, it is then possible to perform the well-known condensation of all degrees of freedom associated with the slave side of the coupling interface (i.e. Lagrange multipliers as well as slave interface velocity and pressure degrees of freedom in this case). The final condensed system does not feature any zero diagonal blocks and is therefore accessible for the same state-of-the-art iterative solution techniques that are also commonly applied to variational multiscale formulations of incompressible flow *without* additional mortar interfaces.

5.1.3 Numerical example

Only one representative numerical example is investigated here to demonstrate the accuracy and efficiency of the proposed mortar approach for subdomain coupling in fluid mechanics. Due to its popularity for the validation of FE formulations for incompressible flow, the well-known circular cylinder benchmark (i.e. the 3D laminar flow past a circular cylinder in a cuboid channel) has been chosen for this purpose, see e.g. Schäfer and Turek [144] for a geometry description and further details. No-slip boundary conditions are prescribed at all channel walls and the surface of the cylinder. At the inflow plane, the time-dependent inflow velocities are prescribed as

$$u_x = \frac{16 U_{\max} (H - y)(H - z)yz}{H^4} \sin\left(\frac{\pi t}{8}\right),$$

where $U_{\max} = 2.25$ m/s defines the maximal velocity in the center of the inflow plane. The parameter $H = 0.41$ m denotes the width and height of the channel and y, z the position on the inflow plane. The kinematic viscosity of the fluid is $\nu = \mu/\rho = 0.001$ m²/s, resulting in a maximal Reynolds number $\text{Re}_{\max} = 100$. At the outflow plane, a zero Neumann boundary condition is assumed. To assess the treatment of non-matching interfaces with the proposed dual mortar method, a boundary layer mesh of 0.03 m thickness is defined around the cylinder. A mortar-based discretization is generated by rotating the boundary layer mesh about two degrees with

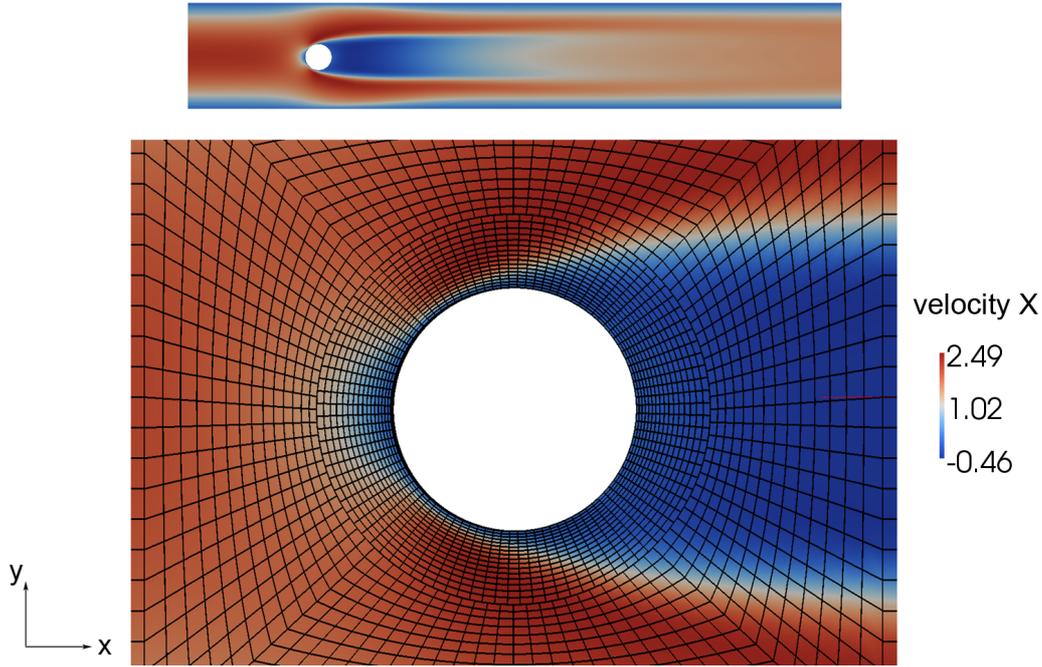


Figure 5.2: 3D cylinder benchmark – exemplary cut through the non-matching mortar-based discretization. The velocity field u_x is visualized in the entire cutting plane and in the boundary layer around the cylinder (detailed view).

respect to the bulk mesh. Numerical results are compared with the case of a conforming discretization. A generalized- α time integration scheme as proposed in Jansen et al. [90] is applied, and the free parameter is chosen to be $\rho_\infty = 1/2$.

In Figure 5.2, the velocity component in x -direction is depicted for a longitudinal cut through the center of the channel. The results feature all flow characteristics expected for the given setup, including a stagnation point at the front of the cylinder and a re-circulation zone behind the cylinder. Note that there is not any visible deviation at the non-matching mortar interface. Figure 5.3 exemplarily illustrates the temporal evolution of the drag and lift coefficients, defined as

$$c_D = \frac{2F_D}{\rho \bar{U}^2 D H}, \quad c_L = \frac{2F_L}{\rho \bar{U}^2 D H}, \quad (5.20)$$

where $D = 0.1$ m denotes the diameter of the cylinder, \bar{U} the characteristic velocity, $\rho = 1.0$ kg/m³ the density and F_D and F_L drag and lift forces, respectively. The characteristic velocity is calculated as $\bar{U} = (4U_{\max}/9) = 1.0$ m/s. As can be observed from Figure 5.3, the drag and lift coefficients for the configurations with and without mortar interface are in excellent agreement. Additionally, the maximal values of the drag and lift coefficients fit in the reference result spectra provided in Schäfer and Turek [144] ($c_{D_{\max}} = 3.29$, reference result spectrum: $c_{D_{\max}} = 3.20 - 3.30$, $c_{L_{\max}} = 0.0028$, reference result spectrum: $c_{L_{\max}} = 0.0020 - 0.0040$). To further demonstrate the flexibility of the presented mortar framework, the first-order (*hex8*) elements in the boundary layer are replaced by second-order (*hex27*) elements. The obtained results are included in Figure 5.3 and no visible difference can be observed for either using only first-order elements or using a combination of first-order and second-order elements.

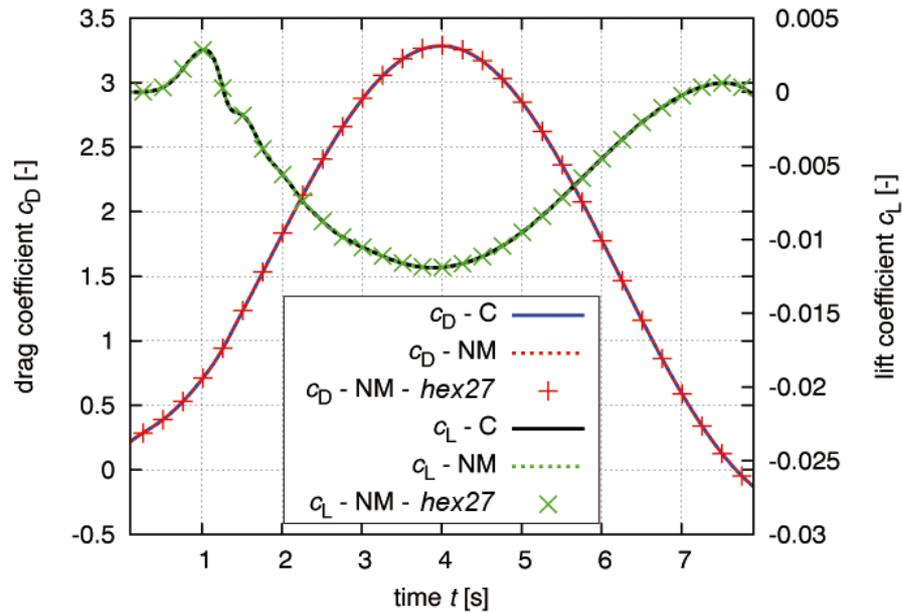


Figure 5.3: 3D cylinder benchmark – drag and lift coefficients over time.

Mesh	Setup	ndof	ndof $_{\Gamma_1}$	n_{proc}	Total time	Condensation	Linear iterations
A	C	248,640	-	8	20.64 s	-	20.70
	NM	253,680	5040	8	30.34 s	4.66 s	23.34
B	C	959,904	-	16	46.75 s	-	22.34
	NM	972,576	12,672	16	59.49 s	5.94 s	26.78
C	C	2,040,192	-	32	69.44 s	-	32.48
	NM	2,061,024	20,832	32	84.60 s	5.88 s	34.27

Table 5.1: Comparison of solver performance averaged over the time steps for three levels of mesh refinement – conforming (C) and non-matching (NM) mortar-based discretization.

Finally, the parallel performance (using up to 32 processors) is evaluated for three levels of mesh refinement denoted as mesh A, B and C. In all cases, an iterative Krylov solver with an algebraic multigrid (AMG) preconditioner is used for solving the linear system of equations. The results are listed in Table 5.1, where ndof denotes the number of degrees of freedom in the entire domain, ndof_{Γ_1} the number of degrees of freedom on the mortar interface Γ_1 , and n_{proc} the number of processors used to solve the system. The number of Newton iterations per time step is not affected by the presence of an internal interface and thus not listed. First of all, it can be observed that the configuration with non-matching meshes increases the average number of linear solver iterations per Newton step by about 5 – 20%. Apparently, the condensed system of equations resulting from mortar discretization using dual Lagrange multipliers is more difficult to solve than a system resulting from conforming discretization due to an increased fill-in.

The second slight increase in the overall computational costs per time step is caused by the condensation operation, as described for mesh tying in solid mechanics in Section 3.4.3. The condensation operation is an additional step in the solution algorithm, which is not necessary for

conforming discretizations. Additionally, in the current implementation, the nodes on the internal interface are not equally distributed over all processors. Hence, the condensation operation may be performed by only a fraction of all available processors, whereas the remaining ones are inactive. Improved load balancing between the different processors as suggested in Section 4.6.1 is a potential remedy for further improving the performance of the proposed numerical method. Besides these two factors, the mortar approach does not require additional resources, since the evaluation of the discrete mortar matrices \mathbf{D} and \mathbf{M} (see Section 3.2.1) only needs to be performed once at the outset of the simulation. For further details on the resulting algorithms and for further results, e.g. numerical convergence studies with both linear and quadratic mortar finite elements in 3D, the interested reader is referred to Ehrl et al. [35].

5.2 Interface coupling in fluid-structure interaction

The numerical simulation of FSI phenomena has long been a field of intensive research owing to its many applications in civil, mechanical, aerospace and biomechanical engineering. Of particular interest is the interaction of incompressible flow with flexible structures undergoing finite deformations. Possible solution strategies range from weakly coupled partitioned over strongly coupled partitioned to monolithic schemes. Partitioned schemes allow for the use of established field solvers and are hence relatively simple to implement. However, monolithic schemes provide an attractive alternative due to their superior numerical robustness and efficiency, see e.g. Gee et al. [49], Heil [68] and Küttler [102]. In general, monolithic schemes are derived based on the assumption of a conforming interface discretization, i.e. fluid and structure share a common interface mesh. In these cases, enforcement of coupling conditions is straightforward. But only in very rare cases, this assumption will hold. Due to a manifold of reasons one generally has to deal with non-matching grids at the fluid-structure interface. Most often, different resolution requirements in the different physical domains or quite simply the presence of complex interface geometries (e.g. in patient-specific biomechanics modeling) make the creation of matching fluid and structure meshes cumbersome or even impossible.

A possible remedy is again provided by mortar methods as presented in the previous chapters in the context of solid and contact mechanics. Different other coupling methods for non-conforming FSI interfaces have been discussed for example in de Boer et al. [28], Dettmer and Perić [30], Farhat et al. [38], Kim [95], Ross et al. [140] and Unger et al. [164]. The mortar method is referred to as a method with desirable mathematical and numerical properties in de Boer et al. [28] and Farhat et al. [38], but it has not yet been used for monolithic FSI computations in a competitive manner. Thus, the consistent integration of mortar methods using dual Lagrange multiplier interpolation into a state-of-the-art monolithic FSI framework will be outlined in the following paragraphs.

After briefly describing the underlying problem statement, an overview of the Arbitrary Lagrangian Eulerian FSI approach on moving grids including mortar-based coupling of the non-matching interface meshes is given. Mortar coupling itself is again not discussed in detail, as all relevant aspects have already been introduced in Chapter 3 and directly carry over from solid mechanics to fluid-structure interaction. Similar to the fluid mechanics application in Section 5.1, dual Lagrange multiplier interpolation is the decisive ingredient for deriving efficient solution algorithms. This is due to the fact that the concept of biorthogonality (cf. Section 3.3) allows

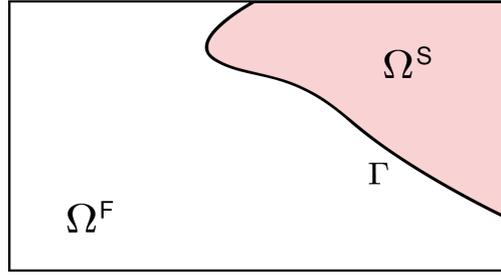


Figure 5.4: Fluid domain Ω^F and structure domain Ω^S separated by the FSI interface Γ .

for the trivial condensation of the additional Lagrange multiplier degrees of freedom from the global linear system of equations. A representative numerical example demonstrates the applicability of mortar methods for challenging 3D FSI scenarios with non-matching interfaces and the numerical efficiency of the devised algorithms, see Klöppel et al. [96] for all details.

5.2.1 Problem statement

The relevant governing equations of solid mechanics and fluid mechanics have already been introduced in previous chapters, see Section 2.1.4 for the IBVP of nonlinear elastodynamics and Section 5.1.1 for the incompressible Navier–Stokes equations. Nevertheless, in order to make the upcoming presentation of fluid-structure coupling rather self-contained and to introduce some FSI specific notation, the participating physical fields with their respective governing equations are reviewed once again. The general problem setup is illustrated in Figure 5.4.

The present FSI approach assumes a fluid field governed by the instationary, incompressible Navier-Stokes equations for a Newtonian fluid on a deformable fluid domain Ω^F . The unknown fluid domain deformation \mathbf{d}^G is defined by a unique mapping φ , which is governed by the mesh interface displacement \mathbf{d}_Γ^G , i.e.

$$\mathbf{d}^G(\mathbf{x}, t) = \varphi(\mathbf{d}_\Gamma^G, \mathbf{x}, t) \quad \text{in } \Omega^F \times [0, T], \quad (5.21)$$

where \mathbf{d}_Γ^G is later connected with the structure interface displacement \mathbf{d}_Γ^S . In principle, the mapping φ is arbitrary, but in order to minimize discretization errors it is desirable to keep element distortions as small and uniformly distributed as possible. One widely used possibility is the so-called pseudo-structure approach, see e.g. Wall [169]. The domain velocity \mathbf{u}^G is given by

$$\mathbf{u}^G = \frac{\partial \mathbf{d}^G}{\partial t} \quad \text{in } \Omega^F \times [0, T], \quad (5.22)$$

and has to match the fluid velocity \mathbf{u}_Γ^F at the interface Γ , i.e.

$$\mathbf{u}_\Gamma^F = \mathbf{u}_\Gamma^G \quad \text{on } \Gamma \times [0, T]. \quad (5.23)$$

Equation (5.22) allows for the definition of the ALE convective velocity $\mathbf{c} = \mathbf{u}^F - \mathbf{u}^G$, representing the fluid velocity relative to the arbitrarily moving fluid domain. The Navier-Stokes equations of the fluid field in ALE formulation hence read

$$\nabla \cdot \mathbf{u}^F = 0, \quad (5.24)$$

$$\rho^F \frac{\partial \mathbf{u}^F}{\partial t} + \rho^F (\mathbf{c} \cdot \nabla) \mathbf{u}^F + \nabla p^F - 2\mu \nabla \cdot \varepsilon(\mathbf{u}^F) = \hat{\mathbf{f}}^F, \quad (5.25)$$

both valid in $\Omega^F \times [0, T]$, where fluid velocity \mathbf{u}^F and fluid pressure p^F are unknown. In the momentum equation (5.25), $\hat{\mathbf{f}}^F$ denotes a body force per unit volume, $\boldsymbol{\varepsilon}(\mathbf{u}^F) = 1/2 (\nabla \mathbf{u}^F + (\nabla \mathbf{u}^F)^\top)$ the strain rate tensor of the Newtonian fluid and μ its dynamic viscosity. Equation (5.24) states the fluid's incompressibility deduced from the conservation of mass and a constant density ρ^F . Boundary conditions at the Dirichlet and Neumann boundaries Γ_D^F and Γ_N^F can be stated as

$$\mathbf{u}^F = \hat{\mathbf{u}}^F \quad \text{on } \Gamma_D^F \times [0, T], \quad (5.26)$$

$$\boldsymbol{\sigma}^F \cdot \mathbf{n}^F = \hat{\mathbf{h}}^F \quad \text{on } \Gamma_N^F \times [0, T], \quad (5.27)$$

where the Cauchy stress tensor $\boldsymbol{\sigma}^F$ is introduced via

$$\boldsymbol{\sigma}^F = -p^F \mathbf{I} + 2\mu \boldsymbol{\varepsilon}(\mathbf{u}^F), \quad (5.28)$$

with the second-order identity tensor \mathbf{I} . Moreover, $\hat{\mathbf{u}}^F$ is the velocity prescribed on the Dirichlet boundary, \mathbf{n}^F the unit outer normal vector to the Neumann boundary and $\hat{\mathbf{h}}^F$ the given traction per unit surface area on Γ_N^F . As in Section 5.1.1, a divergence-free initial velocity field $\mathbf{u}^F(\mathbf{x}, 0) = \mathbf{u}_0^F(\mathbf{x})$ for $\mathbf{x} \in \Omega^F$ is additionally required to round off the strong formulation.

On the structural side, the governing equations of nonlinear elastodynamics yield

$$\nabla \cdot \mathbf{P} + \hat{\mathbf{f}}^S = \rho^S \ddot{\mathbf{d}}^S \quad \text{in } \Omega^S \times [0, T], \quad (5.29)$$

that states an equilibrium between the forces of inertia, internal forces and an external body force $\hat{\mathbf{f}}^S$ per unit volume in the reference configuration Ω^S . Given the structural density ρ^S defined per unit undeformed volume, (5.29) has to be solved for the unknown displacements \mathbf{d}^S . The internal forces are expressed in terms of the first Piola–Kirchhoff stress tensor \mathbf{P} . Different constitutive relations can be employed in this context, see Section 2.1.2 for details. The boundary conditions defined on the Dirichlet boundary Γ_D^S and on the Neumann boundary Γ_N^S read

$$\mathbf{d}^S = \hat{\mathbf{d}}^S \quad \text{on } \Gamma_D^S \times [0, T], \quad (5.30)$$

$$\mathbf{P} \cdot \mathbf{N} = \hat{\mathbf{h}}^S \quad \text{on } \Gamma_N^S \times [0, T]. \quad (5.31)$$

where $\hat{\mathbf{d}}^S$ is the displacement prescribed on the Dirichlet boundary, \mathbf{N} the unit outer normal vector to the Neumann boundary and $\hat{\mathbf{h}}^S$ the associated traction per unit surface area. Given initial displacements and velocities $\hat{\mathbf{d}}_0^S(\mathbf{x})$ and $\hat{\dot{\mathbf{d}}}_0^S(\mathbf{x})$, respectively, the following initial conditions have to be additionally satisfied:

$$\mathbf{d}^S(\mathbf{x}, 0) = \hat{\mathbf{d}}_0^S(\mathbf{x}) \quad \text{in } \Omega^S, \quad (5.32)$$

$$\dot{\mathbf{d}}^S(\mathbf{x}, 0) = \hat{\dot{\mathbf{d}}}_0^S(\mathbf{x}) \quad \text{in } \Omega^S. \quad (5.33)$$

Coupling of the different fields is realized by enforcing kinematic and dynamic constraints at the fluid-structure interface Γ . Usually, the no-slip boundary condition

$$\frac{\partial \mathbf{d}_\Gamma^S}{\partial t} = \mathbf{u}_\Gamma^F \quad \text{on } \Gamma \times [0, T] \quad (5.34)$$

is applied, which prohibits both a mass flow across and a relative tangential movement of fluid and structure at the fluid-structure interface. In combination with (5.23), this condition (5.34) is equivalent to

$$\mathbf{d}_\Gamma^S = \mathbf{d}_\Gamma^G \quad \text{on } \Gamma \times [0, T], \quad (5.35)$$

stating that structural deformation and ALE-based fluid domain deformation (represented by \mathbf{d}_Γ^G) must match on Γ . Note that the equivalence of (5.34) and (5.35) only holds in the continuous setting. In a discrete setting, one of these two equations may not be satisfied exactly if different time integration schemes are used for the different fields. This effect, which is well-known in fluid-structure interaction, can usually be considered negligible. In addition, equilibrium of forces requires the surface tractions of fluid and structure to be equal, yielding

$$\mathbf{h}_\Gamma^S = -\mathbf{h}_\Gamma^F \quad \text{on } \Gamma \times [0, T]. \quad (5.36)$$

With the governing equations for fluid field, structure field and interface coupling at hand, a weak formulation for the resulting fluid-structure interaction system can be formulated by applying the method of weighted residuals. However, before doing so, the Lagrange multiplier field $\boldsymbol{\lambda}$ and corresponding test functions $\boldsymbol{\mu}$ on the fluid-structure interface Γ need to be introduced as additional coupling variables to be solved for. As usual, the Lagrange multipliers $\boldsymbol{\lambda}$ are identified with the unknown surface tractions as $\boldsymbol{\lambda} = \mathbf{h}_\Gamma^S = -\mathbf{h}_\Gamma^F$. The definition of suitable solution spaces \mathcal{S} and weighting spaces \mathcal{V} for the fluid field unknowns \mathbf{u}^F and p^F , the structure field unknowns \mathbf{d}^S and the Lagrange multipliers $\boldsymbol{\lambda}$ is similar to Section 5.1.1 and thus not repeated here. The weak form of the incompressible Navier–Stokes equations (5.24) and (5.25) including a coupling term due to the weak imposition of the unknown surface tractions $\boldsymbol{\lambda}$ is obtained as

$$(\delta p^F, \nabla \cdot \mathbf{u}^F)_{\Omega^F} = 0, \quad (5.37)$$

$$\begin{aligned} & \left(\delta \mathbf{u}^F, \rho^F \frac{\partial \mathbf{u}^F}{\partial t} \right)_{\Omega^F} + (\delta \mathbf{u}^F, \rho^F (\mathbf{c} \cdot \nabla) \mathbf{u}^S)_{\Omega^F} + (\nabla \delta \mathbf{u}^F, 2\mu \boldsymbol{\varepsilon}(\mathbf{u}^F))_{\Omega^F} \\ & - (\nabla \cdot \delta \mathbf{u}^F, p^F)_{\Omega^F} - (\delta \mathbf{u}^F, \hat{\mathbf{f}}^F)_{\Omega^F} - (\delta \mathbf{u}^F, \hat{\mathbf{h}}^F)_{\Gamma_N^F} + (\delta \mathbf{u}_\Gamma^F, \boldsymbol{\lambda})_\Gamma = 0. \end{aligned} \quad (5.38)$$

Similarly, testing (5.29) with the virtual displacements $\delta \mathbf{d}^S$ and applying integration by parts yields the following weak form of the structure field:

$$\left(\delta \mathbf{d}^S, \rho^S \ddot{\mathbf{d}}^S \right)_{\Omega^S} + (\nabla \delta \mathbf{d}^S, \mathbf{P})_{\Omega^S} - (\delta \mathbf{d}^S, \hat{\mathbf{f}}^S)_{\Omega^S} - (\delta \mathbf{d}^S, \hat{\mathbf{h}}^S)_{\Gamma_N^S} - (\delta \mathbf{d}_\Gamma^S, \boldsymbol{\lambda})_\Gamma = 0. \quad (5.39)$$

Again, the influence of the FSI interface on the structure field is accounted for by a coupling term including the Lagrange multipliers $\boldsymbol{\lambda}$. Finally, applying the method of weighted residuals also to the interface constraint (5.35) leads to

$$(\delta \boldsymbol{\lambda}, \mathbf{d}_\Gamma^S - \mathbf{d}_\Gamma^G)_\Gamma = 0. \quad (5.40)$$

This adds an integral version of the continuity constraint (5.35) to the general problem definition. Thus, the entire fluid-structure coupling has been established in a weak sense, which formally leads to a four field FSI system (including the discrete ALE mesh displacements \mathbf{d}^G) and builds the basis for the following mortar finite element discretization using dual Lagrange multipliers.

5.2.2 ALE-based moving-grid approach

The field equations are discretized in space and time, but the actual discretization is of little importance for outlining the most important aspects of the monolithic FSI approach presented here. Implicit time integration schemes are used for fluid, ALE and structure fields, thus leading to a set of nonlinear algebraic equations. The interface conditions are enforced using a mortar method with dual Lagrange multipliers. The complete nonlinear FSI problem is then solved using a Newton–Raphson type method, where a set of corresponding linearized equations has to be solved in every iteration step of the algorithm.

Details on mortar interface discretization, time integration, consistent linearization and the resulting algebraic representation of (5.37)–(5.40) are omitted here, and the reader is referred to Klöppel et al. [96] instead. However, it is important to point out that the final linear system of equations again takes the same saddle point structure as the mesh tying system for solid mechanics discussed in Section 3.4.2. Owing to the application of dual Lagrange multiplier interpolation, it is then possible to perform the well-known condensation of all degrees of freedom associated with the slave side of the coupling interface (i.e. Lagrange multipliers as well as slave side primary degrees of freedom). The final condensed system does not feature any zero diagonal blocks and is therefore accessible for the same state-of-the-art iterative solution techniques and tailored preconditioners that are also commonly applied to moving-grid FSI formulations with *node-matching* interface meshes, see Gee et al. [49].

For the sake of completeness, it should be mentioned that either the fluid or the structure interface can be chosen as slave side for mortar coupling in the proposed monolithic FSI scheme. Furthermore, it could also be shown in Klöppel et al. [96] that Dirichlet-Neumann partitioned FSI schemes equally benefit from the dual mortar approach. The coupling of slave and master degrees of freedom at the FSI interface does not require the solution of a linear system of equations in that case, which significantly reduces computational costs.

5.2.3 Numerical example

In the following, a representative numerical example is analyzed to validate the proposed mortar approach for coupling non-matching interface meshes in moving-grid FSI. For this purpose, the widely used pressure wave benchmark (i.e. the 3D fluid-structure interaction in a thin-walled cylindrical tube) has been chosen, see e.g. Gee et al. [49] and Gerbeau and Vidrascu [50] for details. The simple problem setup, mimicking hemodynamic conditions in arteries, is given in Figure 5.5. The tube is $l = 0.1$ m long, has an inner radius of $d_i = 0.01$ m and an outer radius of $d_o = 0.011$ m. The structure is described with a Neo–Hookean material law with Young’s modulus $E = 10^5$ Pa, Poisson’s ratio $\nu = 0.3$ and density $\rho^S = 1200$ kg/m³. The Newtonian fluid inside the tube has a dynamic viscosity of $\mu = 0.003$ Pa s and a density of $\rho^F = 1000$ kg/m³. The inflow surface is loaded with a surface traction of 1000 Pa for 0.003 s. For the computation a time step size of $\Delta t = 1.0 \times 10^{-4}$ s is used and 250 timesteps are performed. A characteristic deformed configuration for $t = 0.01$ s is shown in Figure 5.6.

The fluid is discretized with stabilized first-order (*hex8*) finite elements, the structure with eight-node solid shell elements as proposed in Vu-Quoc and Tan [167, 168]. In this example, results and computational times for conforming and non-conforming (mortar) meshes of different mesh sizes are compared. The coarsest non-conforming mesh used in this study is shown in

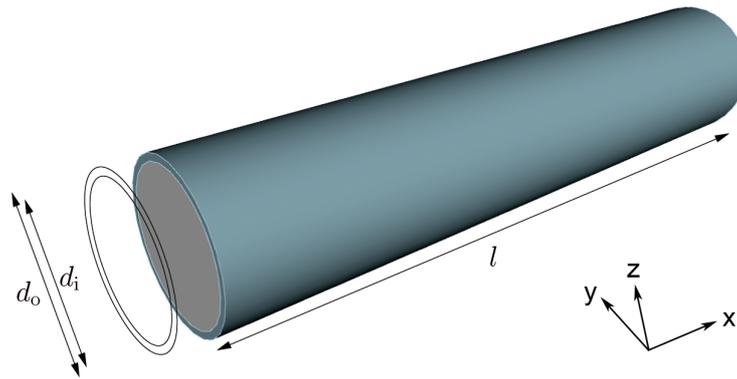


Figure 5.5: 3D pressure wave – problem setup and initial geometry.

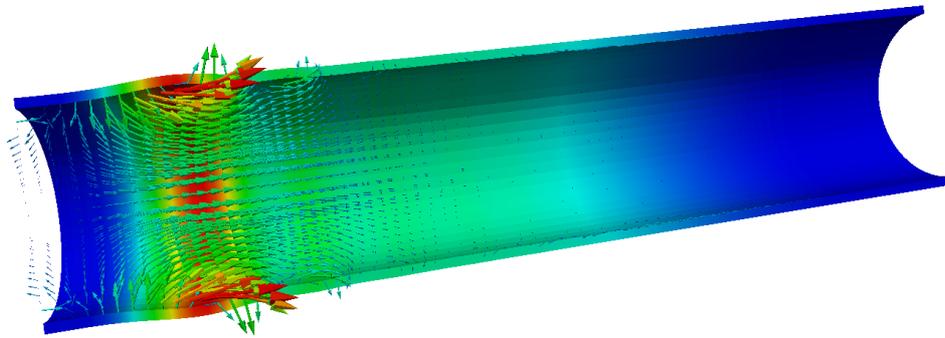


Figure 5.6: 3D pressure wave – deformed configuration for $t = 0.01$ s. Structural displacements d^S and fluid velocities u^F are visualized as contour and vector plot, respectively.

Figure 5.7, but the ratio of fluid and structure mesh sizes holds for all considered cases. Along the circumferential direction both discretizations have the same number of elements but the fluid mesh is rotated such that the meshes overlap for approximately a third of an element length as can be seen in Figure 5.7. This number of elements is denoted by n_{ele} and serves as characteristic quantity for the convergence studies. Throughout the structural wall thickness the number of elements depends on n_{ele} to obtain similar shaped structure elements for all meshes, where one element is used for the coarsest mesh considered. The element length in lateral direction is chosen such that the elements at the outer surface of the fluid domain have approximately square facets. Across the length of the tube the structure discretization contains two elements less than the fluid discretization to realize non-matching meshes. Four different mesh sizes characterized by $n_{\text{ele}} = 16, 32, 48, 64$ are considered in the following. The resulting numbers of degrees of freedom are listed in Table 5.2 for the non-conforming case. Note that these numbers for the conforming case differ only slightly for the structure and match for fluid and ALE and are hence not explicitly given here.

Computations are performed in parallel on up to 12 processors. In order to study efficiency and scalability of the proposed approach, three cases are distinguished for any given mesh size: non-matching (NM), conforming with mortar coupling (CM) and conforming (C). In all cases the GMRES method is used as linear solver in combination with an FSI-specific algebraic multigrid

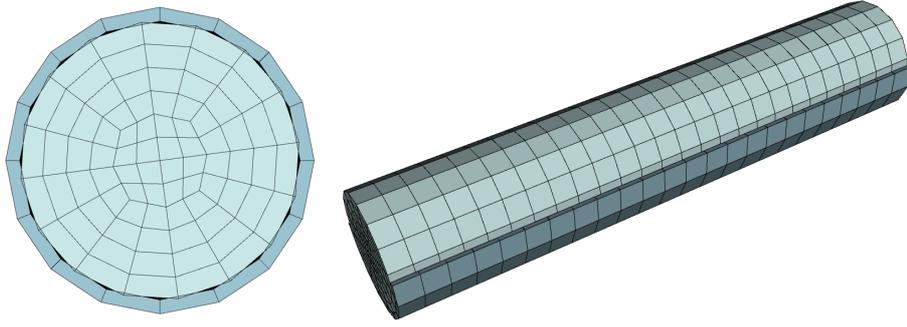
Figure 5.7: 3D pressure wave – exemplary non-conforming mesh ($n_{\text{ele}} = 16$).

Table 5.2: 3D pressure wave – numbers of degrees of freedom for different discretizations with non-matching (NM) meshes.

n_{ele}	Structure	Fluid	ALE	Total
16	2400	9612	7209	19,221
32	14,688	71,444	53,583	139,715
48	44,352	189,916	142,437	376,705
64	96,960	540,956	405,717	1,043,633

preconditioner as introduced in Gee et al. [49] to solve the resulting monolithic linear FSI system. The numerical behavior is assessed by the average number of Newton iterations, the average number of GMRES iterations per Newton step and the average computation time per timestep. These quantities are listed in Table 5.3.

As could be expected, the number of Newton iterations and linear iterations of the GMRES algorithm for the two conforming mesh cases C and CM coincide. The mortar-based approach and the corresponding matrix operations, which are needed to eliminate the discrete Lagrange multipliers and to set up the condensed system matrices, affect the numerical costs only very slightly. The average time spent for a timestep increases by less than 2% when going from case C to case CM on all mesh refinement levels considered, which indicates the numerical efficiency of the proposed approach.

Having to deal with non-conforming meshes (rows marked with “NM” in Table 5.3) and mortar discretization leads to an increase in computation time of approximately 25% for this example. This is completely due to linear systems of equations that are apparently harder to solve, which becomes obvious by comparing the average number of GMRES steps and Newton steps. The relative increase of iteration steps needed in the GMRES corresponds exactly to the relative increase in computation time, whereas the number of Newton iterations is unchanged. While this increase cannot be neglected, it is however important to point out that the preconditioners have not been optimized for the non-matching case but are simply carried over from the conforming case. The computational cost associated with the evaluation of dual mortar coupling itself is virtually zero. This is above all due to the fact that the mortar integrals (i.e. matrices \mathbf{D} and \mathbf{M}) only need to be evaluated once during problem initialization and remain unchanged afterwards.

To evaluate the convergence behavior for non-matching meshes in relation to the employed element sizes, the results are compared with a reference solution computed on an even finer

Table 5.3: 3D pressure wave – linear solver and nonlinear solver performance averaged over timestep and number of processors used.

n_{ele}	Mesh	n_{proc}	Newton	GMRES	time
16	C	4	2.97	29.2	4.44
16	CM	4	2.97	29.2	4.56
16	NM	4	2.97	36.1	6.09
32	C	8	2.98	30.0	22.99
32	CM	8	2.98	30.0	23.59
32	NM	8	2.98	32.8	27.1
48	C	12	2.97	30.9	41.09
48	CM	12	2.97	30.9	41.43
48	NM	12	2.97	37.1	50.68
64	C	12	2.78	33.9	99.44
64	CM	12	2.78	33.9	101.12
64	NM	12	2.78	42.7	121.68

discretization ($n_{\text{ele}} = 80$) with conforming meshes. The reference model has approximately 2 million degrees of freedom and was run on 20 processors. For the sake of simplicity, only the discrete structure displacement magnitude at an arbitrary (due to the rotational symmetry) node of the structure side of the interface Γ and on a cut through at the middle of the tube is monitored. The relative error err_q of a quantity q with respect to the reference solution q_{ref} , given at discrete timesteps $t_i, i \in \{1, 2, \dots, 250\}$ is then defined as

$$\text{err}_q = \frac{1}{250} \sqrt{\sum_{i=1}^{250} \left(\frac{q(t_i) - q_{\text{ref}}(t_i)}{q_{\text{ref}}(t_i)} \right)^2}. \quad (5.41)$$

This quantity does not define an error norm and the results may to some extent depend on the node considered. However, (5.41) serves as a good indicator of how the proposed mortar method influences the accuracy of the FSI simulation. The resulting convergence plot is shown in Figure 5.8. Only for the very coarse mesh the different mesh options yield slightly different results. For reasonably fine meshes, the results of the conforming and the non-matching case agree very well and converge equally fast to the reference solution. For further details on the resulting mortar FSI algorithms and for a thorough numerical validation the interested reader is referred to Klöppel et al. [96].

5.3 Fluid-structure-contact interaction

A variety of problems in engineering and applied sciences require the simulation of unilateral contact of solids surrounded by an incompressible fluid. Important fields of application include machine parts, such as gaskets or sliding-contact bearings, and biomechanical systems, such as heart valves or capillary flow of red blood cells. From the method development point of view, fluid-structure interaction problems coupled with structural contact require powerful simulation

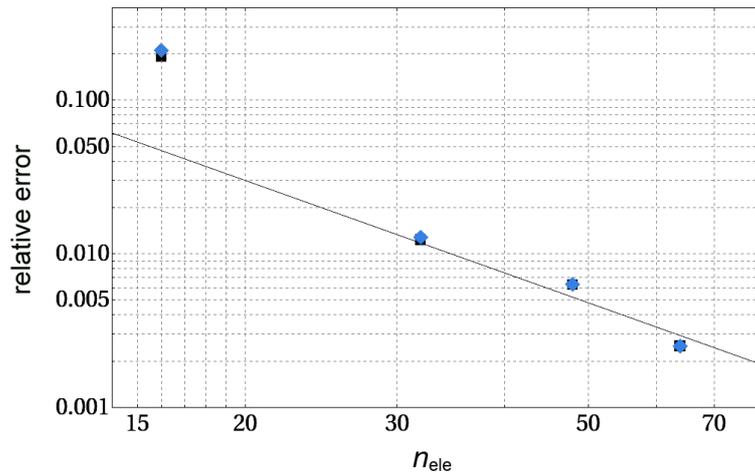


Figure 5.8: 3D pressure wave – convergence study for structure displacement with respect to element numbers, characterized by n_{ele} . Diamonds are used for non-matching (NM), rectangles for conforming (C) interface meshes. The straight line indicates quadratic decrease.

approaches for both algorithmic ‘building blocks’: FSI and contact. Even though many different FSI approaches and contact formulations exist, most of them are hard or even impossible to combine algorithmically with a reasonable programming effort. To give an example, ALE methods are a very popular category of FSI methods, see Section 5.2. For this so-called moving grid FSI approach, the fluid mesh is connected to the solid mesh and deforms according to the structural deformations. However, serious problems occur in the case of approaching bodies, since fluid elements between contact interfaces are completely squeezed together. While problems due to excessive mesh distortion could still be resolved by remeshing, the fluid domain eventually even undergoes a topological change in the limit case of contact. This feature precludes a straightforward application of ALE methods for simulations of contact in a surrounding fluid. Nevertheless, a successful but rather heuristic workaround has been proposed in Sathe and Tezduyar [142] and Tezduyar and Sathe [160], where the authors introduce a so-called surface-edge-node contact tracking algorithm used in combination with a moving grid FSI formulation. When two bodies approach each other, penalty forces are applied in such a way that the contacting surfaces are always kept slightly apart in order to protect the fluid mesh in between. However, as mentioned above, the limit case of actual contact cannot be reproduced exactly.

Other efforts in developing numerical approaches for FSI including contact concentrated on fixed grid FSI formulations in rather specific problem scenarios, such as heart valve dynamics in Astorino et al. [5], Diniz dos Santos et al. [33] and van Loon et al. [165]. The main idea in this chapter now is to bring together recent developments in the fields of fluid-structure interaction and computational contact mechanics in order to create a novel and very general numerical approach. A finite element formulation is proposed, which combines a fixed grid FSI method based on extended finite element methods and the dual mortar contact formulation discussed in Chapter 4. Owing to its generality, the resulting fluid-structure-contact interaction method is applicable to a broad range of problems involving elastic contact of solids embedded in an interacting flow field. The first building block is a two-field XFEM FSI method as proposed in Gerstenberger and Wall [52] and Gerstenberger [51]. The fluid field and the structural field

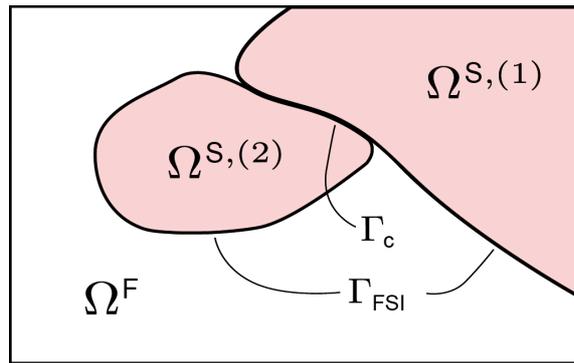


Figure 5.9: Fluid domain Ω^F and two independent structure domains $\Omega^{S,(1)}$ and $\Omega^{S,(2)}$ separated by the FSI interface Γ_{FSI} and the contact interface Γ_c .

live on two distinct grids. An interface handling algorithm, see Mayer et al. [114], allows to localize the fluid-structure interface of arbitrarily moving as well as deforming structures and provides an exact representation of the discretized interface within the fixed fluid background mesh. Thus, no fluid computation is performed between the structures in contact. Continuity between non-matching fluid and interface grids is weakly imposed by a stress-based Lagrange multiplier technique introduced in Gerstenberger and Wall [53]. This coupling approach is numerically stable without heuristic stabilization parameters and does not generate a saddle point structure for the resulting FSI system. These features are mainly responsible for the convenient integration of the mortar-based contact formulation. Flow patterns around surfaces very close to contact can be resolved with high accuracy. Moreover, an exactly represented interface in combination with a fixed fluid grid allows to simulate approaching bodies as well as the limit case of actual contact. No fluid mesh distortion problems occur and the topological changes of the fluid domain due to contact can be handled properly.

The following sections give an overview of the FSCI formulation presented in full detail in Mayer et al. [114]. After first reviewing the problem statement of fluid-structure interaction, some explanations on the XFEM-based fixed-grid approach and an algorithmic integration of the mortar contact formulation with dual Lagrange multipliers are given. Finally, the effectiveness of the resulting algorithm is demonstrated with a representative numerical example.

5.3.1 Problem statement

All relevant governing equations of solid and contact mechanics as well as fluid mechanics have already been introduced in previous chapters, see Section 2.1.4 for the IBVP of nonlinear elastodynamics, Section 4.1 for contact interaction and Section 5.1.1 for the incompressible Navier–Stokes equations. Thus, these equations are not repeated in detail here. Nevertheless, as several aspects of notation slightly change in the present context of fluid-structure-contact interaction, some additional remarks should at least be made here.

The general FSCI problem setup is illustrated in Figure 5.9. Due to the fact that a fixed-grid fluid formulation is considered, the Navier–Stokes equations are identical to the version given in (5.1) and (5.2) when dropping the subdomain superscript $\cdot^{(k)}$. Alternatively, when starting from the Navier–Stokes equations as formulated in (5.24) and (5.25), the ALE-convective ve-

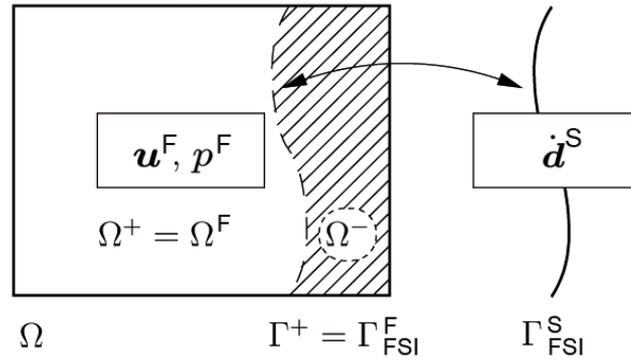


Figure 5.10: Fluid domain with embedded Dirichlet boundary conditions on an intersecting FSI interface representing the structural surface.

locity c (see Section 5.2.1) can simply be replaced by the actual fluid velocity \mathbf{u}^F , since the grid velocity \mathbf{u}^G vanishes in the fixed-grid case. The coupling conditions at the FSI interface given in (5.34) and (5.36) remain unchanged. As typical for contact interaction, the structure part of the problem statement is again derived assuming two independent subdomains, which are denoted as $\Omega^{S,(1)}$ and $\Omega^{S,(2)}$ here. Thus, the strong form for the structure part of the considered FSCI problem, including frictionless contact conditions at the contact interface Γ_c , completely carries over from Section 4.1.1. Corresponding weak formulations of fluid, structure, FSI coupling and contact conditions can be expressed in full analogy to the previous paragraphs.

Finally, it is important to point out that *both* an FSI boundary Γ_{FSI} and a potential contact boundary $\Gamma_c \subset \Gamma_{FSI}$ exist for the given problem class, which requires some additional practical considerations. However, this is a purely algorithmic issue and can readily be dealt with in the context of contact search and enforcement of boundary conditions. As long as the two bodies are separated by a finite gap, FSI coupling conditions are evaluated on the contact boundary. Only as soon as parts of Γ_c actually become active, this region is removed from the FSI boundary Γ_{FSI} and thus dry contact interaction is achieved.

5.3.2 XFEM-based fixed-grid approach

A characteristic feature of the presented fixed-grid approach for fluid-structure-contact interaction is that the fluid field and the structure field live on two distinct meshes: a fixed mesh for the fluid and moving Lagrangian meshes for the structures. The structures may move or deform arbitrarily in the background fluid mesh. An exact fluid-structure interface representation allows to capture flow patterns around the structures with high accuracy and it also enables a straightforward combination of the FSI approach with the contact formulation. Obviously, the fluid-structure interface does not generally match with the fluid mesh here, as it has been the case in the ALE-based moving grid approach in Section 5.2. Thus, an embedded Dirichlet formulation first proposed in Gerstenberger and Wall [53] to impose Dirichlet boundary conditions weakly on non-fitting grids is applied. A sketch of the fluid subproblem is depicted in Figure 5.10.

The entire problem domain Ω is divided into a physical domain Ω^+ , which is identical with the fluid domain Ω^F , and a fictitious domain Ω^- , which coincides with the structural domain Ω^S in the present context. Two identical fluid-structure interface representations are obtained: the

interface $\Gamma_{\text{FSI}}^{\text{S}}$ belongs to the structural mesh and the interface $\Gamma_{\text{FSI}}^{\text{F}} = \Gamma^+$ to the fluid mesh. The structural interface velocity $\dot{\mathbf{d}}^{\text{S}}$ has to be imposed as an embedded Dirichlet condition on the velocity field \mathbf{u}^+ at the fluid interface. Formulating inf-sup stable Lagrange multipliers as traction or vector field living on the interface appears to be a very hard task for embedded interfaces, especially in 3D. Instead, it has been proposed in Gerstenberger and Wall [53] to introduce an additional stress field $\bar{\boldsymbol{\sigma}}$ directly in the fluid domain Ω^{F} , from which a surface traction is easily recovered as $\bar{\boldsymbol{\sigma}} \cdot \mathbf{n}^{\text{F}}$. The corresponding test function to $\bar{\boldsymbol{\sigma}}$ is denoted as $\bar{\boldsymbol{\gamma}}$. The weak kinematic coupling along the fluid-structure interface is therefore given as

$$(\bar{\boldsymbol{\gamma}} \cdot \mathbf{n}^{\text{F}}, \mathbf{u}^+ - \dot{\mathbf{d}}^{\text{S}})_{\Gamma^+} = 0. \quad (5.42)$$

However, substituting the traction vector by an additional unknown stress field $\bar{\boldsymbol{\sigma}}$ leads to only three equations for six unknowns (assuming a 3D problem). Thus, an additional strain rate balance needs to be formulated to close the set of equations, i.e.

$$(\bar{\boldsymbol{\gamma}}, \bar{\boldsymbol{\varepsilon}} - \boldsymbol{\varepsilon})_{\Omega^{\text{F}}} = 0. \quad (5.43)$$

The strain rate balance only needs to be enforced weakly on elements which are actually intersected by the fluid-structure interface. Both strain rates are computed from the primary fluid unknowns \mathbf{u}^{F} , p^{F} and $\bar{\boldsymbol{\sigma}}$, viz.

$$\boldsymbol{\varepsilon} = \frac{1}{2}(\nabla \mathbf{u}^{\text{F}} + (\nabla \mathbf{u}^{\text{F}})^{\text{T}}), \quad (5.44)$$

$$\bar{\boldsymbol{\varepsilon}} = \frac{1}{2\mu}(\bar{\boldsymbol{\sigma}} + p^{\text{F}} \mathbf{I}). \quad (5.45)$$

Details on the XFEM discretization and geometric interface handling are not given here. However, at least the basic idea how to handle the sharp interface description within an element intersected by Ω^+ in this context shall be highlighted. The XFEM allows to model arbitrary discontinuities or interfaces within a single element by enriching the ansatz function space. It was initially developed for discontinuous solid mechanics problems such as crack growth in Belytschko and Black [9] and Moës et al. [117]. Any physical field is approximated by the sum of a continuous term (as usual) and a discontinuous term, where the latter introduces additional degrees of freedom. The interpolation function in the discontinuous term consists of a product of shape functions and a discontinuous enrichment function. According to the underlying physics that have to be modeled, i.e. FSI in this context, the well-known Heaviside function is chosen as enrichment function here, such that it equals one in the fluid domain Ω^+ and zero in the fictitious part Ω^- occupied by the structure. The enrichment function can be expressed as

$$\Psi(\mathbf{x}) = \begin{cases} +1 & \text{in } \Omega^+ \\ 0 & \text{in } \Omega^- \end{cases}. \quad (5.46)$$

The following equation represents schematically the solution and weighting functions for velocities, pressure as well as stresses. As explained above, each quantity is approximated by a continuous term and a discontinuous term based on the product of common FE shape functions $N(\mathbf{X})$ and enrichment function $\Psi(\mathbf{X})$. The continuous term is tagged by a superscript \cdot^c and the discontinuous term by \cdot^{dc} , viz.

$$(\cdot)^h(\mathbf{X}, t) = \sum_I N_I(\mathbf{X})(\cdot)_I^c(t) + \sum_J N_J(\mathbf{X})\Psi(\mathbf{X})(\cdot)_J^{dc}(t). \quad (5.47)$$

In the presented FSI formulation, first-order or second-order shape functions $N(\mathbf{X})$ are applied, which are continuous at element boundaries for the velocity and pressure discretization, but discontinuous at element boundaries for the stress discretization.

The remaining algorithmic steps, including mortar contact discretization, time integration, consistent linearization and the resulting algebraic representation are omitted here, and the reader is referred to Mayer et al. [115] instead. However, it is important to point out that the final linear system of equations does not contain any zero diagonal blocks, and thus does not suffer from the common drawbacks of such saddle point type systems. This is due to the fact that the additional degrees of freedom associated with *both* the traction-based contact Lagrange multiplier and the stress-based Lagrange multiplier for the embedded FSI interface can easily be condensed from the global system. While dual Lagrange multiplier interpolation as discussed in Chapters 3 and 4 is again the key feature for global condensation of the contact variables, the discontinuous stress field $\bar{\sigma}$ can even be condensed at the element level before global assembly, see Gerstenberger and Wall [53] and Gerstenberger [51]. With the contact formulation thus being “hidden” inside the structure block of the resulting FSI system, available partitioned and monolithic coupling schemes can be applied without any conceptual changes.

5.3.3 Numerical example

A representative numerical example is presented here to demonstrate the effectiveness of the proposed numerical framework for fluid-structure-contact interaction, i.e. its ability to deal with finite structural deformations and especially contact interaction in combination with classical FSI. A beam-like structure (Young’s modulus $E = 2000$, Poisson’s ratio $\nu = 0.4$) is positioned in a two-dimensional channel flow, see Figure 5.11. It should be pointed out that the devised implementation in BACI (cf. Wall and Gee [171]) is inherently three-dimensional, so that this 2D example is actually modeled as a 3D problem with just one layer of elements in the third direction. A parabolic inflow profile is applied as Dirichlet boundary condition at the left and a zero traction Neumann boundary condition is assumed at the outflow. All remaining channel boundaries are rigid walls with contact occurring between the beam-like structure and a circular obstacle. Exemplarily, *hex8* elements are used for both the fluid mesh and the structural discretization. Plane strain conditions are enforced by constraining any movement orthogonal to the paper-plane.

The resulting flow field and structural deformation including contact are illustrated in Figure 5.11, giving an impression of this highly dynamic FSCI process. The beam-like structure exhibits large deformations: At first, they are primarily induced by fluid stresses resulting from the increasing fluid pressure, i.e. a typical fluid-structure interaction process is initiated. At later stages the gap between beam and obstacle closes and the structural deformation is then dominated by contact interaction. It should be mentioned that the given example represents more of a qualitative proof of concept for the successful integration of the mortar contact formulation into a fixed-grid FSI framework. Further investigations can also be found in Mayer et al. [115]. While the obtained preliminary results are definitely promising towards the simulation of more challenging FSCI applications, the complex physical phenomena occurring during the approach of the two bodies and the associated transition of boundary conditions from FSI type to contact type are admittedly not yet fully captured due to an insufficient fluid mesh resolution. However, the following paragraph will give a short outlook on the detailed analysis of such wet contact

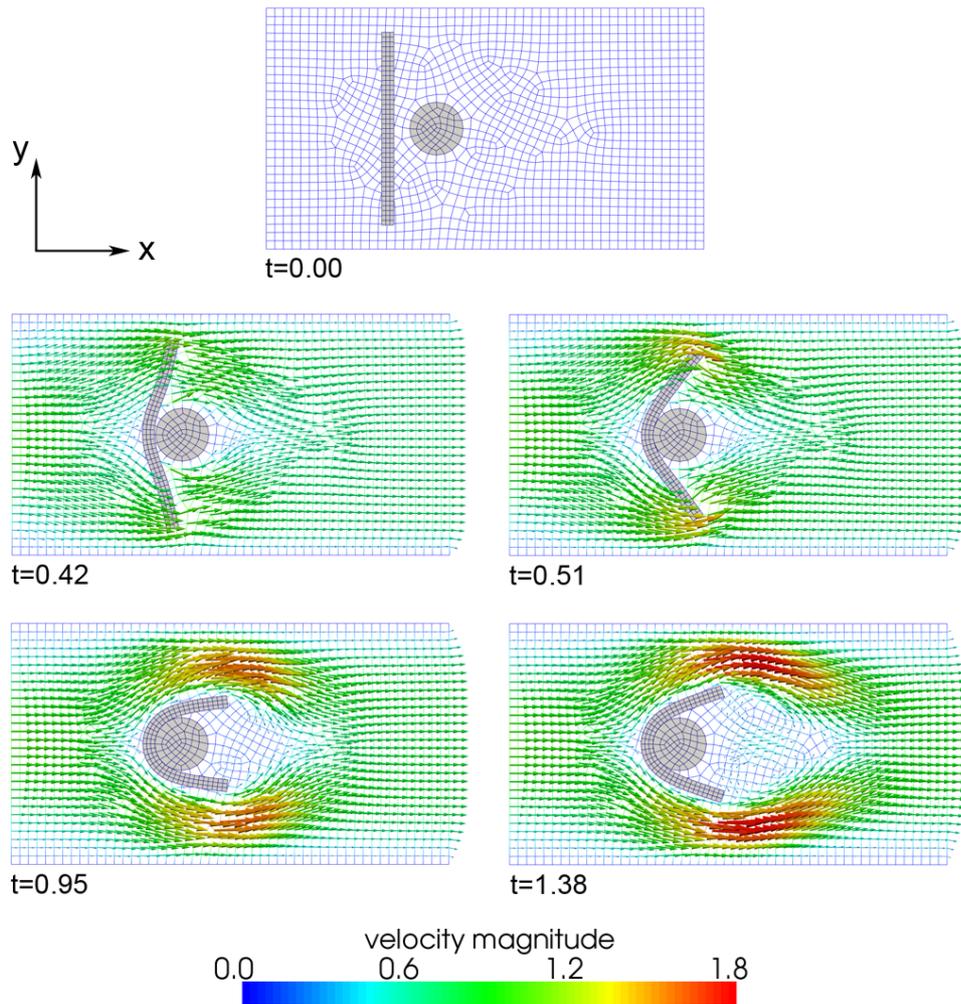


Figure 5.11: Beam-like structure in channel flow – finite element mesh (top), fluid velocity and structural deformation in several characteristic time steps.

and elastohydrodynamic lubrication effects using the proposed fluid-structure-contact interaction scheme.

5.3.4 Extension towards elastohydrodynamic lubrication

This section highlights the numerical simulation of elastohydrodynamic lubrication (EHL) processes as a promising future research direction for which the methods developed in this thesis can be beneficially used. For details about the physical background of lubrication, the reader is referred to Section 2.2.5 and the references mentioned therein. In simple terms, lubrication processes are characterized by a thin fluid film layer, the so-called lubricant, which is established and kept up between the contacting bodies. Classical applications range from so-called hard EHL processes, as for example in machine parts such as fluid bearings, to EHL processes involving rather soft components, including rubber seals, tire-road contact or contact lenses.

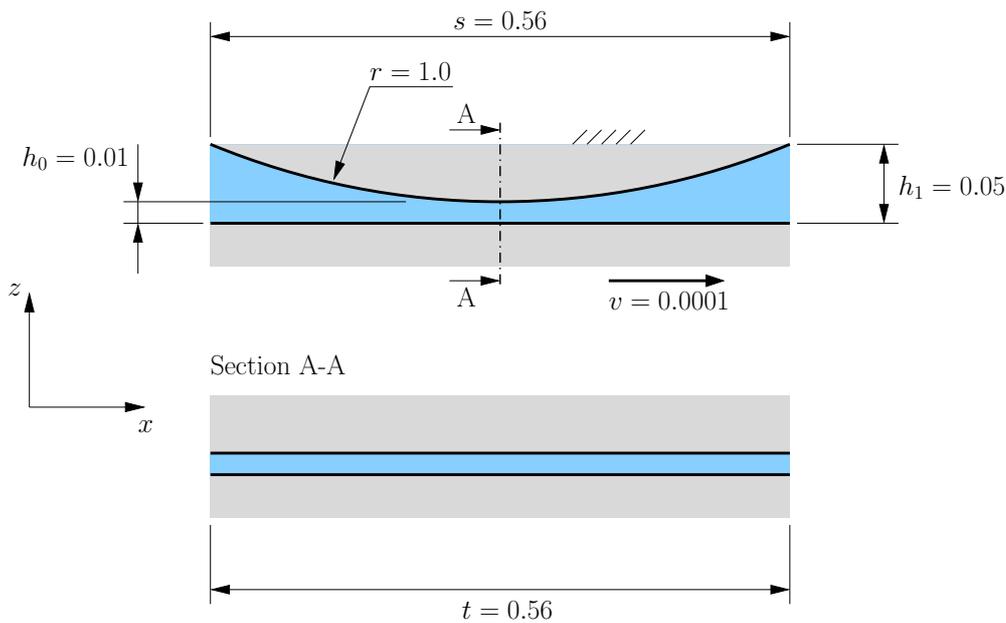


Figure 5.12: Exemplary elastohydrodynamic lubrication model – geometry data.

Before outlining how the fixed-grid FSCI scheme proposed above also covers EHL processes, some remarks are given on the current state-of-the-art treatment of lubrication in computational mechanics. During the last years, the numerical solution of the so-called Reynolds equation for thin film lubrication in combination with penalty methods has become particularly popular, see e.g. Stupkiewicz [154] and Yang and Laursen [188]. The Reynolds equation basically represents a dimensional reduction of the full Navier–Stokes equations by one, i.e. a thickness averaging is introduced, and physically describes the pressure distribution in the lubricant layer. Thus, for a three-dimensional EHL problem in the context of finite element methods, the 2D Reynolds equation is solved based on the surface discretization of the contacting bodies. This explains why the development of such methods is often closely related to the development of robust contact discretizations, even if only fully established lubricant layers are considered without actual structural contact of the involved bodies.

While the mentioned dimensional reduction allows for the derivation of very efficient coupled solution algorithms, some limitations can nevertheless be foreseen for approaches as proposed in Stupkiewicz [154] and Yang and Laursen [188]. This becomes evident when thinking of rather complex scenarios including partial lubrication in combination with dry contact in some regions, or when also considering cavitation regions where the fluid film is ruptured. For such physical effects to be captured, the numerical approach must arguably be capable of representing the physical behavior of the fluid phase in between the two bodies more accurately than is possible with the Reynolds equation. Thus, the presented fluid-structure-contact interaction method based on a fixed-grid FSI scheme and a mortar contact discretization may provide a promising alternative for the simulation of EHL processes, as it allows for a full resolution of the underlying physical effects in the sense of a direct numerical simulation (DNS). Admittedly, the computational effort associated with such a procedure may be quite high, but on the other hand it may offer unrivaled accuracy in the analysis of local lubrication, contact or cavitation phenomena.

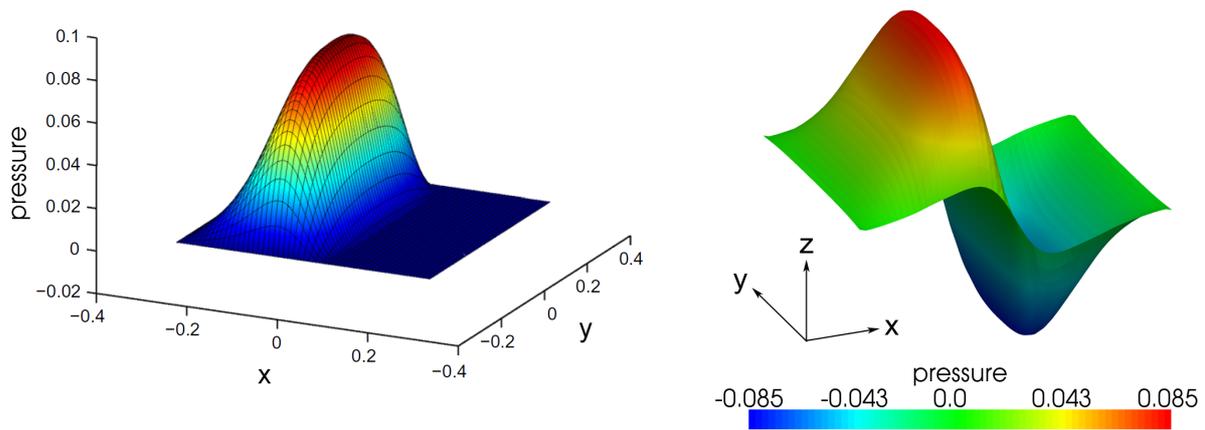


Figure 5.13: Exemplary elastohydrodynamic lubrication model – numerical solutions for the pressure in the lubricant based on the Reynolds equation (left, taken and modified from Yang and Laursen [188]) and based on the proposed FSCI scheme (right).

As a preliminary example, lubrication between an elastic (but rather stiff) cylinder and a rigid surface is investigated as inspired by Yang and Laursen [188]. The problem setup with all relevant dimensions is depicted in Figure 5.12. The rigid surface is moving at a constant velocity of $v = 0.0001$ and thus a pressure increase in the fluid film is induced, where only the stationary solution is of interest here for the sake of simplicity. The kinematic viscosity of the lubricant is $\nu = 1.0$ and potential side leakage of the fluid is accounted for with zero Neumann boundary conditions. Figure 5.13 illustrates the obtained numerical results for the lubricant pressure in comparison with the results given in Yang and Laursen [188]. Note that no thickness averaging or dimensional reduction of any kind is applied in the proposed FSCI scheme, and the pressure solution is evaluated on the rigid surface side of the lubricant gap. Although completely different modeling and solution methods have been used, the results for the pressure profile on the “inflow” side are nearly identical. The so-called Reynolds boundary, marking the transition between lubrication and cavitation regions, is also determined equally well with both methods. Only on the “outflow” side, a methodological difference becomes apparent in the numerical results: while the penalty-based approach given in Yang and Laursen [188] cannot capture the pressure decrease in this region, it is consistently represented by the XFEM-based FSCI scheme. As already mentioned above, this gives rise to the hope that the presented FSCI approach will allow to investigate more complex lubrication scenarios (e.g. including partial lubrication and cavitation phenomena) than possible to date.

6 Summary and Outlook

In this thesis, mortar finite element methods have been investigated in the context of solid mechanics, fluid mechanics and fluid-structure interaction (FSI), with a special emphasis on a novel mortar approach for unilateral contact interaction in the fully nonlinear realm. As a first step, some well-established basic principles of mortar methods have exemplarily been recapitulated for mesh tying (tied contact) in solid mechanics. The concepts of both standard and dual Lagrange multiplier interpolation were addressed with a focus on the latter throughout the entire thesis. The most important favorable feature of dual Lagrange multiplier techniques is the resulting localization of the occurring interface constraints based on a biorthogonalization procedure. Algebraically, this is reflected in the possibility to easily condense the discrete Lagrange multiplier degrees of freedom (DOFs) associated with the non-matching mortar interfaces from the final linear systems of equations. The new scientific contribution of the present work in that regard, i.e. concerning the fundamentals of mortar methods, is the consistent definition of dual Lagrange multipliers for second-order finite element (FE) interpolation in 3D. Two procedures, both based on the classical biorthogonality condition in combination with a special basis transformation, have been suggested and generate locally quadratic and linear dual Lagrange multiplier shape functions, respectively. As for the well-established cases of first-order interpolation or standard Lagrange multipliers, optimal spatial convergence rates could be proven numerically. With second-order finite elements being very popular in engineering practice, the presented approach fills the last missing gap towards a comprehensive dual mortar approach.

The main focus of the present work, however, was an extension of mortar methods and the dual Lagrange multiplier concept to unilateral contact scenarios including finite deformations and nonlinear material behavior. Concerning contact interaction, three major contributions have been presented. Firstly, a consistent linearization of the dual mortar approach for finite deformation contact within implicit time stepping schemes has been derived. While this linearization procedure is already available for penalty-based mortar contact formulations, it was for the first time carried over to the dual Lagrange multiplier case, e.g. including a consistent linearization of the deformation-dependent dual shape functions themselves. All characteristic features of dual Lagrange multipliers mentioned above, such as condensation of the additional DOFs, have the same advantageous effects for unilateral contact as for mesh tying.

Secondly, nonlinear solution techniques with superior numerical efficiency have been proposed for fully nonlinear mortar-based contact formulations. Unlike for penalty methods, the discrete contact constraints are enforced exactly and no problem-specific and user-defined penalty parameter becomes necessary. Owing to an interpretation of primal-dual active set strategies as semi-smooth Newton methods, all occurring nonlinearities (including the search for the active contact set) can be solved within one single iterative scheme, which is not possible for the widely used Augmented Lagrange methods based on Uzawa's algorithm. At the same time, an undesirable increase in global system size due to the additional Lagrange multiplier DOFs is circumvented with the dual Lagrange multiplier approach.

Thirdly, the proposed methods and algorithms have been explicitly designed to meet the requirements of state-of-the-art high-performance computing systems and to provide a readily extendable simulation framework. In terms of parallel computing, the applicability of the proposed mortar methods to large-scale FE models occurring in engineering practice has been proven with suitable numerical examples. Due to parallel redistribution and load balancing algorithms as well as search procedures for two body contact and self contact, the resulting implementation has already reached an advanced degree of maturity with regard to parallel efficiency and scalability. While only discussed briefly in this thesis, the devised numerical algorithms include the classical Coulomb model for frictional sliding.

In the last part of the present work, the proposed mortar methods with dual Lagrange multipliers have successfully been extended to other interface problems beyond pure solid and contact mechanics, including both single-field and multi-field applications. The result of these efforts are novel algorithms for the coupling of non-matching interface meshes in computational fluid mechanics as well as FSI. Concerning fluid mechanics, it has been demonstrated that the dual mortar approach allows for a very efficient coupling of arbitrarily discretized subdomains. While only slightly increasing computational costs and retaining the optimal spatial convergence rates of conforming FE discretizations, the mortar approach significantly increases the flexibility of mesh generation, for example when coupling very fine boundary layer meshes with rather coarsely resolved bulk regions. Similarly, the dual mortar approach has been applied as interface coupling scheme in Arbitrary Lagrangian Eulerian FSI simulations. As a result of mortar-based discretization, fluid and solid meshes can then be generated independently, while established monolithic or partitioned FSI algorithms as well as state-of-the-art iterative solvers remain unchanged. Finally, a novel simulation approach for contacting bodies surrounded by fluid has been developed and termed fluid-structure-contact interaction (FSCI) approach. The resulting FSCI algorithm combines a fixed-grid FSI scheme with independent fluid and solid meshes and the proposed dual mortar contact formulation. In contrast to available tools for such scenarios, the treatment of the limit case of actual contact is straightforward, with applications ranging from flow of contacting deformable particles to elasto-hydrodynamic lubrication (EHL).

In summary, it can be stated that mortar finite element methods with dual Lagrange multipliers have been extended to second-order FE interpolation in 3D and successfully applied to contact mechanics in the fully nonlinear realm, fluid mechanics, FSI as well as coupled FSCI simulations for the first time. However, although substantial progress towards a truly general purpose simulation tool for computational contact mechanics or, more general, for non-conforming discretization and interface coupling in complex multiphysics simulations has been made, there is still room for improvements with regard to several aspects, which were only marginally covered by this thesis or not addressed at all. In the following, an outlook on selected concepts for further improving the proposed methods and numerical algorithms, as well as on promising future research directions associated with mortar methods and computational contact mechanics, will be briefly addressed.

The first issue is related to the smooth interpolation of contact surfaces with large curvatures. Although second-order finite elements as discussed in this thesis already provide improved smoothness as compared with typical first-order elements, the resulting surface description is still only C^0 -continuous due to possible kinks at the nodes. However, there may exist certain applications, such as frictional sliding on highly curved surfaces, where even the slight geometric approximation introduced by second-order FE interpolation prevents an accurate prediction

of the traction distribution, the stick-slip behavior or other relevant contact-related quantities. Moreover, spurious oscillations and even convergence problems of the employed nonlinear solution scheme might appear as a consequence of the lack of a smooth surface description. So-called contact smoothing procedures (e.g. based on Bezier or Hermite patches) are a well-known remedy and have been applied to classical node-to-segment formulations for a long time, see e.g. Wriggers et al. [185] and Puso and Laursen [130]. Thus, in principle, the transfer of such methods to mortar based contact formulations should be a rather straightforward task. Another currently very active research direction is the combination of mortar contact formulations and smooth discretizations based on isogeometric analysis, e.g. using non-uniform rational B-splines as ansatz functions, see De Lorenzis et al. [29] and Temizer et al. [158, 159]. In this regard, the question of how to introduce the concepts of biorthogonality and dual Lagrange multipliers seems particularly interesting and challenging.

On the basis of the work presented in this thesis, further extensions towards complex interface effects should be considered. In many engineering applications, an accurate treatment of non-penetration and Coulomb friction conditions at the contact interfaces is not sufficient to draw all technically relevant conclusions. Stress analysis and lifetime prediction of blade-to-disc joints in aircraft engines is an illustrative example for this statement. Such analyses require a detailed modeling and simulation of the manifold physical phenomena occurring at the contact interfaces. This possibly includes anisotropic friction, the dependency of friction coefficients on state variables (e.g. temperature), heat transmission, dissipation due to frictional sliding and surface degradation due to wear, see e.g. Gitterle [54] and references therein. Other applications may require the ability to analyze contact on both the macroscopic level as well as on a micromechanical scale. Multiscale approaches for contact mechanics can serve as a valuable tool to devise macroscopic models for phenomena such as lubrication and micro-friction or to take into account surface roughness effects, see e.g. Wriggers and Reinelt [181].

To further enhance the numerical accuracy for impact phenomena in computational contact dynamics, energy-momentum conserving time integration schemes tailored for contact applications should be explored, see e.g. Hesch and Betsch [73, 74]. The fundamental drawback of most existing approaches covering this topic is that the fundamental underlying problem, i.e. the discrepancy between smooth velocities and accelerations assumed by standard time integrators and quasi-instantaneous velocity jumps occurring in reality during impact, is not really tackled. Moreover, the combination of mortar contact formulations as developed in this thesis and adaptive finite element methods is a worthwhile future research direction. While adaptive time stepping would possibly help to overcome the issue of impact discussed above, adaptive mesh refinement based on contact-specific error estimators is indispensable in order to achieve a sufficiently fine resolution in highly localized contact zones with reasonable computational effort, see e.g. Hübner and Wohlmuth [82]. Moreover, adaptive meshes provide a means to restore optimal spatial convergence rates, which cannot be obtained based on uniform mesh refinement in certain situations due to the low regularity of unilateral contact problems.

Apart from extending the existing mortar contact formulations by implementing additional functionalities, future work will also be concerned with improvements of the underlying numerical algorithms with respect to efficiency and parallel scalability. For instance, the parallel redistribution and load balancing strategy as well as the contact search procedures presented in this thesis should be further refined. For example, a full parallelization of self contact search and a minimization of the contact-related memory requirements can be achieved by improved com-

munication patterns between the individual processors (e.g. using a ring token strategy). More importantly, the issue of tailored iterative solvers, or rather preconditioning techniques, for the resulting linear systems of equations within each solution step still has a lot of room for improvement. While it has been demonstrated in the present work that dual Lagrange multiplier interpolation allows for removing the undesirable saddle point structure of such linear systems and thus facilitates the application of state-of-the-art iterative solvers, no contact-specific preconditioning techniques have been devised yet. An out-of-the-box application of available preconditioners developed for general solid mechanics problems (e.g. algebraic multigrid methods with smoothed aggregation, see Vaněk et al. [166]) yields satisfactory results in many cases, but the non-conforming structure of contact discretizations may preclude an efficient solution in other cases. In order to tap the full potential of such solution techniques for the dual mortar approach, knowledge about the different involved groups of DOFs (i.e. slave, master and non-contact DOFs) should be included in the design of the preconditioner.

Finally, the applicability of the developed computational approach to realistic scenarios in solid mechanics, contact mechanics, fluid mechanics, FSI and other single-field and multi-field problems has to be proven with further large-scale simulations. The results obtained in the last chapters of this thesis also suggest that an extension of the dual mortar approach towards an even more general framework for non-conforming discretization and interface coupling is a promising future research direction. Exemplary applications beyond those already presented above include multiphysics systems such as in thermomechanics and electrochemistry, the flow of red blood cells through microfluidic devices, complex EHL scenarios or contact of deformable particles. For all mentioned applications, the dual mortar methods developed in this thesis provide an important building block in order to obtain more accurate numerical solutions than possible to date, or even to gain insight into phenomena that have hardly been accessible for computational analysis until now.

A Details Concerning Consistent Linearization

As the computation of directional derivatives accounts for a significant portion of the implementation effort associated with the proposed mortar contact methods, linearization details are provided in this appendix. While Appendix A.1 focuses on the 2D case presented in Section 4.2.2, the linearizations for 3D mortar contact as discussed in Section 4.2.3 are given in Appendix A.2. A similar overview can also be found in Yang et al. [189] within the framework of a penalty-regularized mortar contact formulation for first-order interpolation in 2D. In contrast, the following derivations are not restricted to linear shape functions but remain valid independent of the polynomial degree of the finite element interpolation. For further details, the interested reader is also referred to Popp et al. [123, 124].

A.1 Linearization of 2D mortar contact

In the case of two-dimensional finite deformation contact problems in combination with the solution approach based on semi-smooth Newton methods presented in Section 4.4, the starting point for consistent linearization is given by the directional derivatives of the first mortar matrix \mathbf{D} in (4.32), the second mortar matrix \mathbf{M} in (4.33) and the discrete weighted gap $(\tilde{g}_n)_j$ at slave node j defined in (4.41). Evaluating these expressions requires knowledge of the following elementary linearizations, which are contained in ΔD_{jk} , ΔM_{jl} and $\Delta(\tilde{g}_n)_j$, respectively:

- the nodal normal and tangential vectors \mathbf{n}_j and $\boldsymbol{\tau}_j$ defined in (3.25),
- the slave element Jacobian determinant $J(\xi^{(1)})$ defined in (3.33),
- the integration segment end points $\xi_{a,b}^{(1,2)}$ defined in (3.27) and (3.28),
- the integration segment Gauss points $\xi_g^{(1,2)}$ defined in (3.30),
- the dual shape functions Φ_j defined in (3.64) and (3.65),
- the discretized version of the gap function $g_{n,h}$ defined in (2.46).

Eventually, algebraic representations of all directional derivatives are required for the actual numerical implementation of the presented linearizations. Yet, as the process of assembling directional derivatives to an equivalent matrix-vector formulation is quite straightforward and well-known from standard finite element formulations, details are omitted here.

A.1.1 Linearization of normal and tangential vectors

Linearization of the averaged nodal normal vector \mathbf{n}_j and of the associated nodal tangent vectors $\boldsymbol{\tau}_j^\xi$ and $\boldsymbol{\tau}_j^\eta$ at slave node j is described in the following. This process can be traced back to elementary linearizations of nodal coordinates (i.e. nodal displacement increments) within three steps, the first of which starts with the directional derivative of (3.25) as

$$\Delta \mathbf{n}_j = \Delta \left(\frac{\hat{\mathbf{n}}_j}{\|\hat{\mathbf{n}}_j\|} \right) = \frac{\Delta \hat{\mathbf{n}}_j}{\|\hat{\mathbf{n}}_j\|} - \frac{(\hat{\mathbf{n}}_j \cdot \Delta \hat{\mathbf{n}}_j) \hat{\mathbf{n}}_j}{\|\hat{\mathbf{n}}_j\|^3}, \quad (\text{A.1})$$

where $\hat{\mathbf{n}}_j$ has been introduced as abbreviation for the non-unit nodal normal vector defined by adding all adjacent element normals $\mathbf{n}_j^{(e)}$, i.e. $\hat{\mathbf{n}}_j = \sum_{e=1}^{n_j^{\text{adj}}} \mathbf{n}_j^{(e)}$ (see also Figure 3.2 for a 2D example and Figure 3.4 for a 3D example). In a second step, this intermediate quantity is then linearized itself, yielding

$$\Delta \hat{\mathbf{n}}_j = \Delta \left(\sum_{e=1}^{n_j^{\text{adj}}} \mathbf{n}_j^{(e)} \right) = \left(\sum_{e=1}^{n_j^{\text{adj}}} \frac{\hat{\mathbf{n}}_j^{(e)}}{\|\hat{\mathbf{n}}_j^{(e)}\|} \right) = \sum_{e=1}^{n_j^{\text{adj}}} \left(\frac{\Delta \hat{\mathbf{n}}_j^{(e)}}{\|\hat{\mathbf{n}}_j^{(e)}\|} - \frac{(\hat{\mathbf{n}}_j^{(e)} \cdot \Delta \hat{\mathbf{n}}_j^{(e)}) \hat{\mathbf{n}}_j^{(e)}}{\|\hat{\mathbf{n}}_j^{(e)}\|^3} \right). \quad (\text{A.2})$$

Here, all n_j^{adj} elements adjacent to slave node j have to be considered with their respective non-unit element normals $\hat{\mathbf{n}}_j^{(e)}$, $e = 1, \dots, n_j^{\text{adj}}$. Eventually, directional derivatives of these element normals can be computed as

$$\begin{aligned} \Delta \hat{\mathbf{n}}_j^{(e)} &= \Delta (\mathbf{x}_{,\xi}(\xi_j) \times \mathbf{x}_{,\eta}(\xi_j)) = \left(\sum_{k=1}^{n_\xi^e} N_{k,\xi}(\xi_j) \Delta \mathbf{x}_k \right) \times \left(\sum_{k=1}^{n_\eta^e} N_{k,\eta}(\xi_j) \mathbf{x}_k \right) + \\ &+ \left(\sum_{k=1}^{n_\xi^e} N_{k,\xi}(\xi_j) \mathbf{x}_k \right) \times \left(\sum_{k=1}^{n_\eta^e} N_{k,\eta}(\xi_j) \Delta \mathbf{x}_k \right), \end{aligned} \quad (\text{A.3})$$

where n_ξ^e is the number of nodes associated with slave element e . Note that $\mathbf{x}_{,\xi}(\xi_j)$ and $\mathbf{x}_{,\eta}(\xi_j)$ denote the element tangent vectors evaluated at the local element coordinate ξ_j of slave node j . By backwards inserting (A.3) and (A.2) into (A.1), the directional derivative of the averaged nodal normal is expressed exclusively in terms of nodal coordinate linearizations, i.e. nodal displacement increments, and is thus fully defined. With the directional derivative of the nodal normal vector at hand, and considering the definition of the nodal tangent vector as $\boldsymbol{\tau}_j = \mathbf{e}_3 \times \mathbf{n}_j$, the corresponding tangent linearization is straightforward:

$$\Delta \boldsymbol{\tau}_j = \mathbf{e}_3 \times \Delta \mathbf{n}_j. \quad (\text{A.4})$$

A.1.2 Linearization of slave element Jacobian determinant

Considering a slave element with n_ξ^e nodes in a two-dimensional contact situation, the associated Jacobian determinant is defined as

$$J(\xi^{(1)}) = \left\| \sum_{k=1}^{n_\xi^e} N_{k,\xi}(\xi^{(1)}) \mathbf{x}_k^{(1)} \right\|. \quad (\text{A.5})$$

The directional derivative of this slave element Jacobian determinant evaluated at the Gauss point $\xi^{(1)}(\eta_g)$ of a 2D mortar integration segment can then be expressed as

$$\Delta J(\xi^{(1)}(\eta_g)) = \frac{1}{\left\| \sum_{k=1}^{n_s^e} N_{k,\xi}(\xi^{(1)}(\eta_g)) \mathbf{x}_k^{(1)} \right\|} \left(\sum_{k=1}^{n_s^e} N_{k,\xi}(\xi^{(1)}(\eta_g)) \mathbf{x}_k^{(1)} \right) \cdot \left(\sum_{k=1}^{n_s^e} N_{k,\xi\xi}(\xi^{(1)}(\eta_g)) \Delta(\xi^{(1)}(\eta_g)) \mathbf{x}_k^{(1)} + \sum_{k=1}^{n_s^e} N_{k,\xi}(\xi^{(1)}(\eta_g)) \Delta \mathbf{x}_k^{(1)} \right). \quad (\text{A.6})$$

Herein, the term containing the second derivative $N_{k,\xi\xi}(\xi^{(1)}(\eta_g))$ only becomes relevant for second-order or higher-order finite element interpolation. Due to the fact that individual integration segment contributions are evaluated during 2D mortar coupling, see Figure 3.3, the linearization of the Gauss point coordinate $\Delta \xi^{(1)}(\eta_g)$ is typically non-zero. The reason for this is that one or both segment end coordinates on the slave side (i.e. $\xi_a^{(1)}$ and $\xi_b^{(1)}$) are possibly defined via deformation-dependent projections of master nodes as per (3.28).

A.1.3 Linearization of integration segment end points

Figure 3.3 illustrates that two possibilities exist for the origin of the integration segment end coordinates $\xi_a^{(1)}$ and $\xi_b^{(1)}$ on the slave side and $\xi_a^{(2)}$ and $\xi_b^{(2)}$ on the master side, respectively. They can either coincide with a finite element node of the corresponding surface or with the projection of a finite element node from the respective other surface. Of course, in the first case, the linearization of this coordinate vanishes. In the latter case, the directional derivative has to be computed by linearizing the associated projection condition in (3.27) or (3.28).

Derivations similar to the ones following here can also be found in Yang et al. [189]. First, let $\xi_a^{(1)}$ be a slave node, i.e.

$$\Delta \xi_a^{(1)} = 0, \quad (\text{A.7})$$

and consequently $\Delta \xi_a^{(2)}$ can be found by linearization of (3.27), as $\xi_a^{(2)}$ represents the projection of a slave node $\mathbf{x}_a^{(1)}$ onto a master element e with the nodes $\mathbf{x}_l^{(2)}$, $l = 1, \dots, n_m^e$. This yields

$$\Delta \xi_a^{(2)} = - \frac{1}{\left(\sum_{l=1}^{n_m^e} N_{l,\xi}^{(2)}(\xi_a^{(2)}) x_l^{(2)} \right) n_a^y - \left(\sum_{l=1}^{n_m^e} N_{l,\xi}^{(2)}(\xi_a^{(2)}) y_l^{(2)} \right) n_a^x} \cdot \left[\left(\sum_{l=1}^{n_m^e} \left(N_l^{(2)}(\xi_a^{(2)}) \Delta x_l^{(2)} \right) - \Delta x_a^{(1)} \right) n_a^y - \left(\sum_{l=1}^{n_m^e} \left(N_l^{(2)}(\xi_a^{(2)}) \Delta y_l^{(2)} \right) - \Delta y_a^{(1)} \right) n_a^x \right. \\ \left. + \left(\sum_{l=1}^{n_m^e} \left(N_l^{(2)}(\xi_a^{(2)}) x_l^{(2)} \right) - x_a^{(1)} \right) \Delta n_a^y - \left(\sum_{l=1}^{n_m^e} \left(N_l^{(2)}(\xi_a^{(2)}) y_l^{(2)} \right) - y_a^{(1)} \right) \Delta n_a^x \right]. \quad (\text{A.8})$$

Herein, $\mathbf{n}_a = (n_a^x, n_a^y)^\top$ is the unit normal vector at slave node $\mathbf{x}_a^{(1)}$ and $\Delta \mathbf{n}_a = (\Delta n_a^x, \Delta n_a^y)^\top$ denotes the corresponding directional derivative, which can readily be computed according to the procedure defined in (A.1)–(A.3). Now, the second possible case for a segment end coordinate is examined in more detail. Thus, let $\xi_a^{(2)}$ be a master node, i.e.

$$\Delta \xi_a^{(2)} = 0, \quad (\text{A.9})$$

and consequently $\Delta\xi_a^{(1)}$ can be found by linearization of (3.28), as $\xi_a^{(1)}$ represents the projection of a master node $\mathbf{x}_a^{(2)}$ onto a slave element e with the nodes $\mathbf{x}_k^{(1)}$, $k = 1, \dots, n_s^e$. This yields

$$\Delta\xi_a^{(1)} = \frac{num}{denom}, \quad (\text{A.10})$$

with

$$\begin{aligned} denom = & - \left[\sum_{k=1}^{n_s^e} N_{k,\xi}^{(1)}(\xi_a^{(1)}) x_k^{(1)} \right] \left[\sum_{k=1}^{n_s^e} N_k^{(1)}(\xi_a^{(1)}) n_k^y \right] \\ & + \left[\sum_{k=1}^{n_s^e} N_{k,\xi}^{(1)}(\xi_a^{(1)}) y_k^{(1)} \right] \left[\sum_{k=1}^{n_s^e} N_k^{(1)}(\xi_a^{(1)}) n_k^x \right] \\ & - \left[\sum_{k=1}^{n_s^e} \left(N_k^{(1)}(\xi_a^{(1)}) x_k^{(1)} \right) - x_a^{(2)} \right] \left[\sum_{k=1}^{n_s^e} N_{k,\xi}^{(1)}(\xi_a^{(1)}) n_k^y \right] \\ & + \left[\sum_{k=1}^{n_s^e} \left(N_k^{(1)}(\xi_a^{(1)}) y_k^{(1)} \right) - y_a^{(2)} \right] \left[\sum_{k=1}^{n_s^e} N_{k,\xi}^{(1)}(\xi_a^{(1)}) n_k^x \right] \end{aligned} \quad (\text{A.11})$$

and

$$\begin{aligned} num = & \left[\sum_{k=1}^{n_s^e} \left(N_k^{(1)}(\xi_a^{(1)}) \Delta x_k^{(1)} \right) - \Delta x_a^{(2)} \right] \left[\sum_{k=1}^{n_s^e} N_k^{(1)}(\xi_a^{(1)}) n_k^y \right] \\ & - \left[\sum_{k=1}^{n_s^e} \left(N_k^{(1)}(\xi_a^{(1)}) \Delta y_k^{(1)} \right) - \Delta y_a^{(2)} \right] \left[\sum_{k=1}^{n_s^e} N_k^{(1)}(\xi_a^{(1)}) n_k^x \right] \\ & + \left[\sum_{k=1}^{n_s^e} \left(N_k^{(1)}(\xi_a^{(1)}) x_k^{(1)} \right) - x_a^{(2)} \right] \left[\sum_{k=1}^{n_s^e} N_{k,\xi}^{(1)}(\xi_a^{(1)}) \Delta n_k^y \right] \\ & - \left[\sum_{k=1}^{n_s^e} \left(N_k^{(1)}(\xi_a^{(1)}) y_k^{(1)} \right) - y_a^{(2)} \right] \left[\sum_{k=1}^{n_s^e} N_{k,\xi}^{(1)}(\xi_a^{(1)}) \Delta n_k^x \right]. \end{aligned} \quad (\text{A.12})$$

Again, $\mathbf{n}_k = (n_k^x, n_k^y)^\top$ is the unit normal vector at slave node $\mathbf{x}_k^{(1)}$ and $\Delta\mathbf{n}_k = (\Delta n_k^x, \Delta n_k^y)^\top$ denotes the corresponding directional derivative. It is obvious that all considerations above can be applied to the remaining integration segment end coordinates $\xi_b^{(1)}$ and $\xi_b^{(2)}$, too.

A.1.4 Linearization of integration segment Gauss points

With directional derivatives for the segment end coordinates at hand, linearizations of the Gauss point coordinates $\xi^{(1)}(\eta_g)$ and $\xi^{(2)}(\eta_g)$ can be computed next. For the slave side, this is rather straightforward when considering the simple mapping from segment Gauss point coordinate η_g to slave element Gauss point coordinate $\xi^{(1)}(\eta_g)$ given in (3.29). One directly obtains

$$\Delta\xi^{(1)}(\eta_g) = \frac{1}{2}(1 - \eta_g)\Delta\xi_a^{(1)} + \frac{1}{2}(1 + \eta_g)\Delta\xi_b^{(1)}. \quad (\text{A.13})$$

The directional derivative of the master side Gauss point coordinate $\xi^{(2)}(\eta_g)$ is more intricate due to the fact that it has been defined as a projection of the segment Gauss point η_g from slave element to master element as expressed in (3.30). Consequently, $\Delta\xi^{(2)}(\eta_g)$ is determined by linearization of (3.30), viz.

$$\begin{aligned} \Delta\xi_g^{(2)} = & -\frac{1}{\left(\sum_{k=1}^{n_m^e} N_{k,\xi}^{(2)}(\xi^{(2)}(\eta_g))x_k^{(2)}\right)n_g^y - \left(\sum_{k=1}^{n_m^e} N_{k,\xi}^{(2)}(\xi^{(2)}(\eta_g))y_k^{(2)}\right)n_g^x} \\ & \cdot \left[\left(\sum_{k=1}^{n_m^e} \left(N_k^{(2)}(\xi^{(2)}(\eta_g))\Delta x_k^{(2)}\right) - \Delta x_g^{(1)}\right)n_g^y - \left(\sum_{k=1}^{n_m^e} \left(N_k^{(2)}(\xi^{(2)}(\eta_g))\Delta y_k^{(2)}\right) - \Delta y_g^{(1)}\right)n_g^x \right. \\ & \left. + \left(\sum_{k=1}^{n_m^e} \left(N_k^{(2)}(\xi^{(2)}(\eta_g))x_k^{(2)}\right) - x_g^{(1)}\right)\Delta n_g^y - \left(\sum_{k=1}^{n_m^e} \left(N_k^{(2)}(\xi^{(2)}(\eta_g))y_k^{(2)}\right) - y_g^{(1)}\right)\Delta n_g^x \right]. \end{aligned} \quad (\text{A.14})$$

Herein, $\mathbf{n}_g = (n_g^x, n_g^y)^\top$ is the unit normal vector at the Gauss point $\mathbf{x}_g^{(1)}$ and $\Delta\mathbf{n}_g = (\Delta n_g^x, \Delta n_g^y)^\top$ denotes the corresponding directional derivative. Obviously, (A.14) is identical to (A.8) when replacing the segment Gauss point index g by the segment end point index a . In order to conclude this directional derivative, the global slave side Gauss point coordinates $\mathbf{x}_g^{(1)}$ and the Gauss point unit normal vector \mathbf{n}_g remain to be defined. The former can easily be expressed as

$$\Delta\mathbf{x}_g^{(1)} = \sum_{k=1}^{n_s^e} \left(N_k^{(1)}(\xi^{(1)}(\eta_g))\Delta\mathbf{x}_k^{(1)}\right) + \sum_{k=1}^{n_s^e} \left(N_{k,\xi}^{(1)}(\xi^{(1)}(\eta_g))\Delta\xi^{(1)}(\eta_g)\mathbf{x}_k^{(1)}\right). \quad (\text{A.15})$$

Taking into account (3.26), linearization of the unit normal vector \mathbf{n}_g is defined as

$$\Delta\mathbf{n}_g = \sum_{k=1}^{n_s^e} N_{k,\xi}^{(1)}(\xi^{(1)}(\eta_g))\Delta\xi^{(1)}(\eta_g)\mathbf{n}_k + \sum_{k=1}^{n_s^e} N_k^{(1)}(\xi^{(1)}(\eta_g))\Delta\mathbf{n}_k, \quad (\text{A.16})$$

where $\Delta\mathbf{n}_k$ is the directional derivative of the nodal normal vector discussed in Appendix A.1.1.

A.1.5 Linearization of dual shape functions

A general notation for both standard and dual shape functions occurring in the directional derivatives ΔD_{jk} , ΔM_{jl} and $\Delta(\tilde{g}_n)_j$ in a 2D setting reads

$$\Delta\Phi_j(\xi^{(1)}(\eta_g)) = \Phi_{j,d}\Delta\mathbf{d} + \Phi_{j,\xi}^{(1)}\Delta\xi^{(1)}(\eta_g), \quad (\text{A.17})$$

$$\Delta N_k^{(1)}(\xi^{(1)}(\eta_g)) = N_{k,\xi}^{(1)}\Delta\xi^{(1)}(\eta_g), \quad (\text{A.18})$$

$$\Delta N_l^{(2)}(\xi^{(2)}(\eta_g)) = N_{l,\xi}^{(2)}\Delta\xi^{(2)}(\eta_g). \quad (\text{A.19})$$

In contrast to standard finite element shape functions defined on a reference element, dual shape functions are in general deformation-dependent. To be more precise, this deformation-dependency is a specific feature of dual shape functions for contact elements with non-constant Jacobian determinant. This fact explains the first term in (A.17), which contains directional

derivatives with respect to the nodal displacements. Linearization of the slave Gauss point coordinate $\Delta\xi^{(1)}(\eta_g)$ in (A.18) has already been addressed in Section A.2.3 together with the master Gauss point coordinate $\Delta\xi^{(2)}(\eta_g)$ in (A.19).

Let e be a slave element with $m_e^{(1)}$ Lagrange multiplier nodes. Then the directional derivative of the dual shape functions defined in (3.65) is given as

$$\begin{bmatrix} \Delta_{\mathbf{d}}\Phi_1(\xi^{(1)}) \\ \vdots \\ \Delta_{\mathbf{d}}\Phi_{m_e^{(1)}}(\xi^{(1)}) \end{bmatrix} = \Delta\mathbf{A}_e \begin{bmatrix} N_1(\xi^{(1)}) \\ \vdots \\ N_{m_e^{(1)}}(\xi^{(1)}) \end{bmatrix}, \quad (\text{A.20})$$

where $\Delta_{\mathbf{d}}$ denotes only the ‘‘direct’’ directional derivative with respect to the displacements in order to distinguish it from the total linearization given in (A.17). The directional derivative of the coefficient matrix $\Delta\mathbf{A}_e \in \mathbb{R}^{m_e^{(1)} \times m_e^{(1)}}$ has already been discussed in detail in Section 4.3.2.

A.1.6 Linearization of discretized gap function

In the course of determining the directional derivative of the weighted gap $(\tilde{g}_n)_j$ at slave node j , the discretized version of the gap function $g_{n,h}(\xi^{(1)}(\eta_g), \xi^{(2)}(\eta_g))$ at each Gauss point appears, see (4.41). Taking into account the definitions of the gap function, of the contact surface interpolation and of the averaged normal interpolation, one obtains the following expression:

$$\begin{aligned} \Delta g_{n,h} &= \Delta \left[-\mathbf{n}_g \cdot \left(\mathbf{x}_g^{(1)} - \hat{\mathbf{x}}_g^{(2)} \right) \right] \\ &= \Delta \left[- \left(\sum_{k=1}^{n_s^e} N_k^{(1)}(\xi^{(1)}(\tilde{\eta}_g)) \mathbf{n}_k \right) \cdot \left(\sum_{k=1}^{n_s^e} N_k^{(1)}(\xi^{(1)}(\tilde{\eta}_g)) \mathbf{x}_k^{(1)} - \sum_{l=1}^{n_m^e} N_l^{(2)}(\xi^{(2)}(\tilde{\eta}_g)) \mathbf{x}_l^{(2)} \right) \right], \end{aligned} \quad (\text{A.21})$$

Herein, the slave element nodes are denoted as $\mathbf{x}_k^{(1)}$, $k = 1, \dots, n_s^e$ and the master element nodes as $\mathbf{x}_l^{(2)}$, $l = 1, \dots, n_m^e$. The expression given in (A.21) can be further specified as

$$\begin{aligned} \Delta g_{n,h} &= - \left(\mathbf{x}_g^{(1)} - \hat{\mathbf{x}}_g^{(2)} \right) \cdot \left(\sum_{k=1}^{n_s^e} \left(N_{k,\xi}^{(1)}(\xi^{(1)}(\tilde{\eta}_g)) \Delta\xi^{(1)}(\tilde{\eta}_g) \mathbf{n}_k + N_k^{(1)}(\xi^{(1)}(\tilde{\eta}_g)) \Delta\mathbf{n}_k \right) \right) \\ &\quad - \mathbf{n}_g \cdot \left(\sum_{k=1}^{n_s^e} \left(N_{k,\xi}^{(1)}(\xi^{(1)}(\tilde{\eta}_g)) \Delta\xi^{(1)}(\tilde{\eta}_g) \mathbf{x}_k^{(1)} + N_k^{(1)}(\xi^{(1)}(\tilde{\eta}_g)) \Delta\mathbf{x}_k^{(1)} \right) \right) \\ &\quad + \mathbf{n}_g \cdot \left(\sum_{l=1}^{n_m^e} \left(N_{l,\xi}^{(2)}(\xi^{(2)}(\tilde{\eta}_g)) \Delta\xi^{(2)}(\tilde{\eta}_g) \mathbf{x}_l^{(2)} + N_l^{(2)}(\xi^{(2)}(\tilde{\eta}_g)) \Delta\mathbf{x}_l^{(2)} \right) \right), \end{aligned} \quad (\text{A.22})$$

with the global Gauss point coordinates $\mathbf{x}_g^{(1)}$ and $\hat{\mathbf{x}}_g^{(2)}$ and the Gauss point normal

$$\mathbf{n}_g = \frac{\sum_{k=1}^{n_s^e} N_k^{(1)}(\xi^{(1)}(\tilde{\eta}_g)) \mathbf{n}_k}{\left\| \sum_{k=1}^{n_s^e} N_k^{(1)}(\xi^{(1)}(\tilde{\eta}_g)) \mathbf{n}_k \right\|}. \quad (\text{A.23})$$

The linearization of the Gauss point normal is identical to the first step of the linearization of the nodal normal \mathbf{n}_k in (A.1), when $\hat{\mathbf{n}}_k$ is replaced by $\sum_{k=1}^{n_s^e} N_k^{(1)}(\xi^{(1)}(\tilde{\eta}_g)) \mathbf{n}_k$.

A.2 Linearization of 3D mortar contact

Similar to the 2D case, the starting point for consistent linearization of three-dimensional mortar contact formulations is given by the directional derivatives of the first mortar matrix \mathbf{D} in (4.37), the second mortar matrix \mathbf{M} in (4.38) and the discrete weighted gap $(\tilde{g}_n)_j$ at slave node j defined in (4.42). Evaluating these expressions requires knowledge of the following elementary directional derivatives, which are contained in ΔD_{jk} , ΔM_{jl} and $\Delta(\tilde{g}_n)_j$, respectively:

- the normal and tangential vectors \mathbf{n}_j , $\boldsymbol{\tau}_j^\xi$ and $\boldsymbol{\tau}_j^\eta$ defined in (3.34) and (3.35),
- the integration cell Jacobian determinant J_{cell} introduced in Algorithm 1 in Section 3.2.3,
- the integration cell Gauss points $\boldsymbol{\xi}^{(1,2)}(\tilde{\eta}_g)$ introduced in Algorithm 1 in Section 3.2.3,
- the integration cell vertices $\tilde{\mathbf{x}}_v^{\text{cell}}$ introduced in Algorithm 1 in Section 3.2.3,
- the dual shape functions Φ_j defined in (3.64) and (3.65),
- the discretized version of the gap function $g_{n,h}$ defined in (2.46).

Eventually, algebraic representations of all directional derivatives are required for the actual numerical implementation of the presented linearizations. Yet, as the process of assembling directional derivatives to an equivalent matrix-vector formulation is quite straightforward and well-known from standard finite element formulations, details are again omitted here.

A.2.1 Linearization of normal and tangential vectors

Linearization of the averaged nodal normal vector \mathbf{n}_j and of the associated nodal tangent vectors $\boldsymbol{\tau}_j^\xi$ and $\boldsymbol{\tau}_j^\eta$ at slave node j is addressed in the following. Due to identical definitions in (3.25) and (3.34), the 3D version of the normal vector directional derivative does not differ from the 2D version presented in Appendix A.1.1.

However, linearization of the associated nodal tangent vectors $\boldsymbol{\tau}_j^\xi$ and $\boldsymbol{\tau}_j^\eta$ adds some complexity in 3D situations. The three vectors together form an orthonormal basis at node j , see (3.35), but the definition of the tangent directions is not unique within the tangent plane to node j . One of the two vectors can first be chosen freely within the tangent plane, whereas the other then follows from evaluating the cross product in (3.35). This has to be kept in mind when deducing the directional derivatives $\Delta\boldsymbol{\tau}_j^\xi$ and $\Delta\boldsymbol{\tau}_j^\eta$ from the nodal normal counterpart $\Delta\mathbf{n}_j$.

A.2.2 Linearization of integration cell Jacobian determinant

As a result of the 3D mortar segmentation procedure described in Section 3.2.3, all integration cells are 3-node triangles (*tri3*) with vertices $\mathbf{x}_v^{\text{cell}}$, $v = 1, 2, 3$, in global coordinates. Thus,

consistent linearization of the corresponding Jacobian determinant J_{cell} yields

$$\begin{aligned} \Delta J_{\text{cell}} &= \Delta \left(\|(\mathbf{x}_2^{\text{cell}} - \mathbf{x}_1^{\text{cell}}) \times (\mathbf{x}_3^{\text{cell}} - \mathbf{x}_1^{\text{cell}})\| \right) \\ &= \frac{(\mathbf{x}_2^{\text{cell}} - \mathbf{x}_1^{\text{cell}}) \times (\mathbf{x}_3^{\text{cell}} - \mathbf{x}_1^{\text{cell}})}{\|(\mathbf{x}_2^{\text{cell}} - \mathbf{x}_1^{\text{cell}}) \times (\mathbf{x}_3^{\text{cell}} - \mathbf{x}_1^{\text{cell}})\|} \cdot [(\Delta \mathbf{x}_2^{\text{cell}} - \Delta \mathbf{x}_1^{\text{cell}}) \times (\mathbf{x}_3^{\text{cell}} - \mathbf{x}_1^{\text{cell}})] \\ &\quad + \frac{(\mathbf{x}_2^{\text{cell}} - \mathbf{x}_1^{\text{cell}}) \times (\mathbf{x}_3^{\text{cell}} - \mathbf{x}_1^{\text{cell}})}{\|(\mathbf{x}_2^{\text{cell}} - \mathbf{x}_1^{\text{cell}}) \times (\mathbf{x}_3^{\text{cell}} - \mathbf{x}_1^{\text{cell}})\|} \cdot [(\mathbf{x}_2^{\text{cell}} - \mathbf{x}_1^{\text{cell}}) \times (\Delta \mathbf{x}_3^{\text{cell}} - \Delta \mathbf{x}_1^{\text{cell}})] , \end{aligned} \quad (\text{A.24})$$

which reduces to the task of determining directional derivatives of the integration cell vertex coordinates $\mathbf{x}_v^{\text{cell}}$. These elementary contributions are presented in Appendix A.2.4.

A.2.3 Linearization of integration cell Gauss points

It is explained in Section 3.2.3 that slave and master Gauss point coordinates $\boldsymbol{\xi}^{(1,2)}(\tilde{\boldsymbol{\eta}}_g)$ are obtained from integration cell Gauss point coordinates $\tilde{\boldsymbol{\eta}}_g$ by applying an inverse mapping operation. According to Algorithm 1 in Section 3.2.3, the setup of the integration cells is done by performing polygon clipping in the auxiliary plane after first having projected both slave and master nodes into this plane along the slave element center normal \mathbf{n}_0 . Consequently, Gauss point coordinates $\boldsymbol{\xi}^{(1,2)}(\tilde{\boldsymbol{\eta}}_g)$ are computed by projecting the integration cell Gauss points $\tilde{\boldsymbol{\eta}}_g$ back onto slave and master element, respectively. In general, this requires the solution of a system of three nonlinear equations, which is achieved by local Newton iteration, see also Section 3.2.2 as well as Popp et al. [123] and Yang et al. [189] for the same procedure in the 2D case. Hence, the relevant projection conditions are

$$\sum_{k=1}^{n_s^e} \left(N_k^{(1)}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) \mathbf{x}_k^{(1)} \right) - \alpha \mathbf{n}_0 - \mathbf{x}_g = 0 , \quad (\text{A.25})$$

$$\sum_{l=1}^{n_m^e} \left(N_l^{(2)}(\boldsymbol{\xi}^{(2)}(\tilde{\boldsymbol{\eta}}_g)) \mathbf{x}_l^{(2)} \right) - \alpha \mathbf{n}_0 - \mathbf{x}_g = 0 , \quad (\text{A.26})$$

with the slave nodes $\mathbf{x}_k^{(1)}$, $k = 1, \dots, n_s^e$ and the master nodes $\mathbf{x}_l^{(2)}$, $l = 1, \dots, n_m^e$. Equations (A.25) and (A.26) are to be solved for the unknowns α and $\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)$ or $\boldsymbol{\xi}^{(2)}(\tilde{\boldsymbol{\eta}}_g)$, respectively. The integration cell Gauss point \mathbf{x}_g is given in global coordinates. Linearization of (A.25), i.e. the projection condition for the slave side, yields

$$\begin{pmatrix} \Delta \boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g) \\ \Delta \boldsymbol{\eta}^{(1)}(\tilde{\boldsymbol{\eta}}_g) \\ \Delta \alpha \end{pmatrix} = \mathbf{L}^{-1} \cdot \left(\alpha \Delta \mathbf{n}_0 + \Delta \mathbf{x}_g - \sum_{k=1}^{n_s^e} N_k^{(1)}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) \Delta \mathbf{x}_k^{(1)} \right) , \quad (\text{A.27})$$

with the matrix $\mathbf{L} \in \mathbb{R}^{3 \times 3}$ being defined as

$$\mathbf{L} = \left[\sum_{k=1}^{n_s^e} N_{k,\boldsymbol{\xi}}^{(1)}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) \mathbf{x}_k^{(1)} \mid \sum_{k=1}^{n_s^e} N_{k,\boldsymbol{\eta}}^{(1)}(\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)) \mathbf{x}_k^{(1)} \mid -\mathbf{n}_0 \right] . \quad (\text{A.28})$$

Linearization of the master Gauss point coordinates in the projection condition (A.26) is absolutely analogous and thus will be omitted here. The individual contributions to (A.27) are

investigated more closely and must either be traced back to elementary nodal displacement increments or to known directional derivatives from Sections A.2.1 and A.2.2. By definition, the auxiliary plane normal vector has the following directional derivative

$$\Delta \mathbf{n}_0 = \Delta \left(\frac{\sum_{k=1}^{n_s^e} N_k^{(1)}(\boldsymbol{\xi}_0^{(1)}) \mathbf{n}_k}{\left\| \sum_{k=1}^{n_s^e} N_k^{(1)}(\boldsymbol{\xi}_0^{(1)}) \mathbf{n}_k \right\|} \right), \quad (\text{A.29})$$

with the local slave element center coordinates $\boldsymbol{\xi}_0^{(1)}$. Expression (A.1) can be identified here when replacing $\hat{\mathbf{n}}_k$ by $\sum_{k=1}^{n_s^e} N_k^{(1)}(\boldsymbol{\xi}_0^{(1)}) \mathbf{n}_k$. Therefore, \mathbf{n}_0 is simply a combination of nodal normal directional derivatives as derived in Section A.2.1.

Considering an integration cell interpolation based on standard linear shape functions N_v , $v = 1, 2, 3$, in the cell parameter space $\tilde{\boldsymbol{\eta}}$, i.e. the reference triangle as defined in Section 3.2.3, the linearization of an integration cell Gauss point in global coordinates is obtained as

$$\Delta \mathbf{x}_g = \sum_{v=1}^3 N_v(\tilde{\boldsymbol{\eta}}_g) \Delta \mathbf{x}_v^{\text{cell}}, \quad (\text{A.30})$$

with the integration cell vertex coordinates $\mathbf{x}_v^{\text{cell}}$. Thus, the last term in the directional derivative of the slave and master Gauss point coordinates depends on the directional derivative of the integration cell vertices, which in turn is presented in the next paragraph.

A.2.4 Linearization of integration cell vertices

Whereas all derivations presented in Sections A.2.1–A.2.3 have many aspects in common with the two-dimensional mortar coupling algorithms in Appendix A.1, the linearizations described in this paragraph are characteristic of the 3D mortar coupling scheme. As explained in Section 3.2.3, the triangular integration cells are the result of a thorough segmentation of slave and master surfaces based on suitable projection, polygon clipping and Delaunay triangulation algorithms, see Figures 3.5 and 3.6. Before being able to derive a consistent linearization of the vertex coordinates $\mathbf{x}_v^{\text{cell}}$, one must be aware of three different cases from which an integration cell vertex may have originated. Vertex $\mathbf{x}_v^{\text{cell}}$ may be the projection of a slave node or a master node into the auxiliary plane, denoted as cases (1) and (2) in Figure A.1. However, vertex $\mathbf{x}_v^{\text{cell}}$ could also be the intersection of two projected master and slave edges, and thus the result of a line clipping operation as illustrated in case (3) in Figure A.1.

Considering the simple projection formulas for a slave or master node into the auxiliary plane, cases (1) and (2) lead to the following directional derivative:

$$\begin{aligned} \Delta \mathbf{x}_v^{\text{cell}} &= \Delta \left(\mathbf{x}^{(1,2)} - \left[\left(\mathbf{x}^{(1,2)} - \mathbf{x}_0^{(1)} \right) \cdot \mathbf{n}_0 \right] \mathbf{n}_0 \right) \\ &= \Delta \mathbf{x}^{(1,2)} - \left[\left(\Delta \mathbf{x}^{(1,2)} - \Delta \mathbf{x}_0^{(1)} \right) \cdot \mathbf{n}_0 + \left(\mathbf{x}^{(1,2)} - \mathbf{x}_0^{(1)} \right) \cdot \Delta \mathbf{n}_0 \right] \mathbf{n}_0 \\ &\quad - \left[\left(\mathbf{x}^{(1,2)} - \mathbf{x}_0^{(1)} \right) \cdot \mathbf{n}_0 \right] \Delta \mathbf{n}_0. \end{aligned} \quad (\text{A.31})$$

Herein, $\mathbf{x}^{(1,2)}$ represents the slave or master node of which the integration cell vertex $\mathbf{x}_v^{\text{cell}}$ is the projection. Moreover, the slave element center $\mathbf{x}_0^{(1)}$ and the auxiliary plane normal \mathbf{n}_0 are

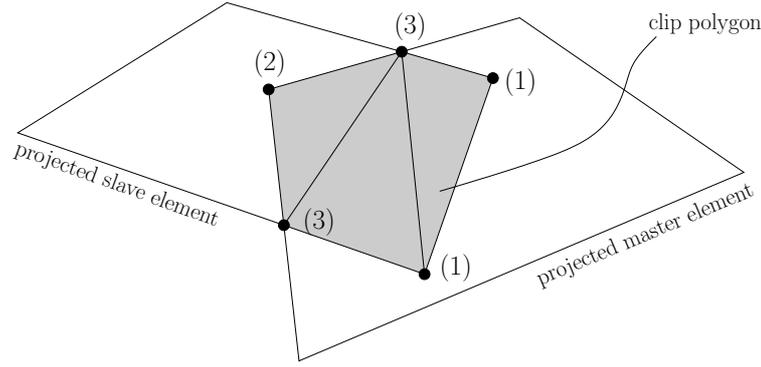


Figure A.1: Different cases for the origin of an integration cell vertex: (1) projected slave node, (2) projected master node, (3) line clipping operation.

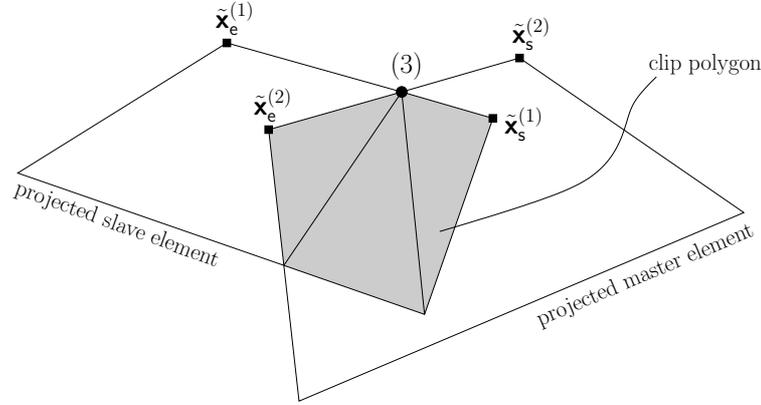


Figure A.2: Integration cell vertex originating from line clipping. The vertex lies at the intersection of the projected slave edge $\tilde{\mathbf{x}}_s^{(1)} \rightarrow \tilde{\mathbf{x}}_e^{(1)}$ and the projected master edge $\tilde{\mathbf{x}}_s^{(2)} \rightarrow \tilde{\mathbf{x}}_e^{(2)}$.

involved. All linearizations in (A.31) can easily be represented by nodal coordinate linearizations $\Delta \mathbf{x}_k^{(1)}$ and $\Delta \mathbf{x}_l^{(2)}$ of the corresponding slave or master element (i.e. nodal displacement increments), respectively. The directional derivative of \mathbf{n}_0 has already been given in (A.29) and the slave element center $\mathbf{x}_0^{(1)}$ can be treated similarly as

$$\Delta \mathbf{x}_0^{(1)} = \sum_{k=1}^{n_s^e} N_k^{(1)}(\boldsymbol{\xi}_0^{(1)}) \Delta \mathbf{x}_k^{(1)}. \quad (\text{A.32})$$

Linearization of case (3) necessitates a linearization of the line clipping algorithm used to compute $\mathbf{x}_v^{\text{cell}}$. In Figure A.2, such a case is exemplarily considered with the projected slave edge being defined by $\tilde{\mathbf{x}}_e^{(1)}$ and $\tilde{\mathbf{x}}_s^{(1)}$ and the projected master edge being defined by $\tilde{\mathbf{x}}_e^{(2)}$ and $\tilde{\mathbf{x}}_s^{(2)}$ as end and starting points. Applying the line clipping algorithm adapted from Foley [42] yields

$$\mathbf{x}_v^{\text{cell}} = \tilde{\mathbf{x}}_s^{(1)} - \frac{(\tilde{\mathbf{x}}_s^{(1)} - \tilde{\mathbf{x}}_s^{(2)}) \times (\tilde{\mathbf{x}}_e^{(2)} - \tilde{\mathbf{x}}_s^{(2)}) \cdot \mathbf{n}_0}{(\tilde{\mathbf{x}}_e^{(1)} - \tilde{\mathbf{x}}_s^{(1)}) \times (\tilde{\mathbf{x}}_e^{(2)} - \tilde{\mathbf{x}}_s^{(2)}) \cdot \mathbf{n}_0} (\tilde{\mathbf{x}}_e^{(1)} - \tilde{\mathbf{x}}_s^{(1)}). \quad (\text{A.33})$$

As discussed in Puso and Laursen [132, 133], consistent linearization of (A.33) is quite tedious but straightforward. One can see very clearly that the directional derivative of $\mathbf{x}_v^{\text{cell}}$ is represented

by the already known directional derivatives of the projected slave and master nodes, see (A.31), and of the auxiliary plane normal \mathbf{n}_0 . Note that (A.33) is no longer valid if the two edges become parallel. Such pathological cases are automatically detected by the line clipping algorithm and treated properly, i.e. reduced to the cases (1) and (2) described above.

A.2.5 Linearization of dual shape functions

The procedure for linearization of the dual shape functions has been introduced in Appendix A.1.5 for the 2D case and is applicable to 3D mortar contact analysis without conceptual differences. With regard to notation, the Gauss point coordinates $\xi^{(1)}(\eta_g)$ and $\xi^{(2)}(\eta_g)$ simply need to be replaced by their vector-valued counterparts $\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)$ and $\boldsymbol{\xi}^{(2)}(\tilde{\boldsymbol{\eta}}_g)$ when considering the three-dimensional setting.

A.2.6 Linearization of discretized gap function

Again, all necessary steps for linearization of the discretized gap function have already been introduced in Appendix A.1.6 for the 2D case and are equally applicable to the 3D version of the mortar contact algorithms. The Gauss point coordinates are vector-valued quantities in the three-dimensional setting, i.e. for example $\boldsymbol{\xi}^{(1)}(\tilde{\boldsymbol{\eta}}_g)$ instead of $\xi^{(1)}(\eta_g)$.

Bibliography

- [1] Y. Achdou, G. Abdoulaev, J.-C. Hontand, Y. A. Kuznetsov, O. Pironneau, and C. Prud'homme, Nonmatching grids for fluids, In *Domain Decomposition Methods 10, Contemporary Mathematics 218*, AMS, Providence, 1998.
- [2] C. Agelet de Saracibar and M. Chiumenti, On the numerical modeling of frictional wear phenomena, *Computer Methods in Applied Mechanics and Engineering* **177**, 401–426, 1999.
- [3] P. Alart and A. Curnier, A mixed formulation for frictional contact problems prone to Newton like solution methods, *Computer Methods in Applied Mechanics and Engineering* **92**, 353–375, 1991.
- [4] J. F. Archard, Contact and rubbing of flat surfaces, *Journal of Applied Physics* **24**, 981–988, 1953.
- [5] M. Astorino, J.-F. Gerbeau, O. Pantz, and K.-F. Traor, Fluid-structure interaction and multi-body contact: Application to aortic valves, *Computer Methods in Applied Mechanics and Engineering* **198**, 3603–3612, 2009.
- [6] K.-J. Bathe, *Finite element procedures*, Prentice Hall, 1996.
- [7] K.-J. Bathe and A. Chaudhary, A solution method for planar and axisymmetric contact problems, *International Journal for Numerical Methods in Engineering* **21**, 65–88, 1985.
- [8] Z. Belhachmi and F. Ben Belgacem, Quadratic finite element approximation of the Signorini problem, *Mathematics of Computation* **72**, 83–104, 2003.
- [9] T. Belytschko and T. Black, Elastic crack growth in finite elements with minimal remeshing, *International Journal for Numerical Methods in Engineering* **45**, 601–620, 1999.
- [10] T. Belytschko, W. K. Liu, and B. Moran, *Nonlinear finite elements for continua and structures*, Wiley, 2000.
- [11] F. Ben Belgacem, The mortar finite element method with Lagrange multipliers, *Numerische Mathematik* **84**, 173–197, 1999.
- [12] F. Ben Belgacem, The mixed mortar finite element method for the incompressible Stokes problem: Convergence analysis, *SIAM Journal on Numerical Analysis* **37**, 1085–1100, 2000.
- [13] F. Ben Belgacem, P. Hild, and P. Laborde, The mortar finite element method for contact problems, *Mathematical and Computer Modelling* **28**, 263–271, 1998.

- [14] F. Ben Belgacem, L. K. Chilton, and P. Seshaiyer, The hp-mortar finite-element method for the mixed elasticity and Stokes problems, *Computers & Mathematics with Applications* **46**, 35–55, 2003.
- [15] D. J. Benson and J. O. Hallquist, A single surface contact algorithm for the post-buckling analysis of shell structures, *Computer Methods in Applied Mechanics and Engineering* **78**, 141–163, 1990.
- [16] C. Bernardi, Y. Maday, and A. T. Patera, A new nonconforming approach to domain decomposition: the mortar element method, In H. Brezis and J. Lions (eds.), *Nonlinear partial differential equations and their applications*, pages 13–51, Pitman/Wiley: London/New York, 1994.
- [17] J. Bonet and R. D. Wood, *Nonlinear continuum mechanics for finite element analysis*, Cambridge University Press, 1997.
- [18] H. Bow, I. V. Pivkin, M. Diez-Silva, S. J. Goldfless, M. Dao, J. C. Niles, S. Suresh, and J. Han, A microfabricated deformability-based flow cytometer with application to malaria, *Lab on a Chip* **11**, 1065–1073, 2011.
- [19] F. Brezzi and M. Fortin, *Mixed and hybrid finite element methods*, Springer-Verlag, 1991.
- [20] S. Brunssen, F. Schmid, M. Schäfer, and B. I. Wohlmuth, A fast and robust iterative solver for nonlinear contact problems using a primal-dual active set strategy and algebraic multigrid, *International Journal for Numerical Methods in Engineering* **69**, 524–543, 2007.
- [21] C. Carstensen, O. Scherf, and P. Wriggers, Adaptive finite elements for elastic bodies in contact, *SIAM Journal on Scientific Computing* **20**, 1605–1626, 1999.
- [22] P. W. Christensen, A semi-smooth Newton method for elasto-plastic contact problems, *International Journal of Solids and Structures* **39**, 2323 – 2341, 2002.
- [23] P. W. Christensen, A. Klarbring, J. S. Pang, and N. Strömberg, Formulation and comparison of algorithms for frictional contact problems, *International Journal for Numerical Methods in Engineering* **42**, 145–173, 1998.
- [24] J. Chung and G. M. Hulbert, A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized- α method, *Journal of Applied Mechanics* **60**, 371–375, 1993.
- [25] T. Cichosz and M. Bischoff, Consistent treatment of boundaries with mortar contact formulations using dual Lagrange multipliers, *Computer Methods in Applied Mechanics and Engineering* **200**, 1317–1332, 2011.
- [26] J. A. Cottrell, T. J. R. Hughes, and Y. Bazilevs, *Isogeometric analysis: toward integration of CAD and FEA*, Wiley, 2009.
- [27] M. A. Crisfield, Re-visiting the contact patch test, *International Journal for Numerical Methods in Engineering* **48**, 435–449, 2000.

-
- [28] 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.
- [29] L. De Lorenzis, I. Temizer, P. Wriggers, and G. Zavarise, A large deformation frictional contact formulation using NURBS-based isogeometric analysis, *International Journal for Numerical Methods in Engineering* **87**, 1278–1300, 2011.
- [30] W. Dettmer and D. Perić, A computational framework for fluid-structure interaction: Finite element formulation and applications, *Computer Methods in Applied Mechanics and Engineering* **195**, 5754–5779, 2006.
- [31] P. Deuffhard, R. Krause, and S. Ertel, A contact-stabilized Newmark method for dynamical contact problems, *International Journal for Numerical Methods in Engineering* **73**, 1274–1290, 2008.
- [32] T. Dickopf and R. Krause, Efficient simulation of multi-body contact problems on complex geometries: A flexible decomposition approach using constrained minimization, *International Journal for Numerical Methods in Engineering* **77**, 1834–1862, 2009.
- [33] N. Diniz dos Santos, J.-F. Gerbeau, and J.-F. Bourgat, A partitioned fluid-structure algorithm for elastic thin valves with contact, *Computer Methods in Applied Mechanics and Engineering* **197**, 1750–1761, 2008.
- [34] C. R. Dohrmann, S. W. Key, and M. W. Heinstein, Methods for connecting dissimilar three-dimensional finite element meshes, *International Journal for Numerical Methods in Engineering* **47**, 1057–1080, 2000.
- [35] A. Ehrl, A. Popp, V. Gravemeier, and W. A. Wall, A mortar approach with dual Lagrange multipliers for mesh tying within a variational multiscale method for incompressible flow, *Computer Methods in Applied Mechanics and Engineering*, submitted, 2012.
- [36] N. El-Abbasi and K.-J. Bathe, Stability and patch test performance of contact discretizations and a new solution algorithm, *Computers & Structures* **79**, 1473–1486, 2001.
- [37] H. Elman, V. E. Howle, J. Shadid, R. Shuttleworth, and R. Tuminaro, A taxonomy and comparison of parallel block multi-level preconditioners for the incompressible Navier–Stokes equations, *Journal of Computational Physics* **227**, 1790–1808, 2008.
- [38] 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.
- [39] K. A. Fischer and P. Wriggers, Frictionless 2D contact formulations for finite deformations based on the mortar method, *Computational Mechanics* **36**, 226–244, 2005.

- [40] K. A. Fischer and P. Wriggers, Mortar based frictional contact formulation for higher order interpolations using the moving friction cone, *Computer Methods in Applied Mechanics and Engineering* **195**, 5020–5036, 2006.
- [41] B. Flemisch and B. I. Wohlmuth, Stable Lagrange multipliers for quadrilateral meshes of curved interfaces in 3D, *Computer Methods in Applied Mechanics and Engineering* **196**, 1589–1602, 2007.
- [42] J. Foley, *Computer graphics: Principles and practice*, Addison-Wesley, 1997.
- [43] C. Förster, *Robust methods for fluid-structure interaction with stabilised finite elements*, PhD thesis, Universität Stuttgart, 2007.
- [44] A. Francavilla and O. C. Zienkiewicz, A note on numerical computation of elastic contact problems, *International Journal for Numerical Methods in Engineering* **9**, 913–924, 1975.
- [45] D. Franke, A. Düster, V. Nübel, and E. Rank, A comparison of the h-, p-, hp-, and rp-version of the FEM for the solution of the 2D Hertzian contact problem, *Computational Mechanics* **45**, 513–522, 2010.
- [46] M. Frémond, Adhérence des solides, *Journal de Mécanique Théorique et Appliquée* **6**, 383–407, 1987.
- [47] M. W. Gee, *Effiziente Lösungsstrategien in der nichtlinearen Schalenmechanik*, PhD thesis, Universität Stuttgart, 2004.
- [48] M. W. Gee, C. T. Kelley, and R. B. Lehoucq, Pseudo-transient continuation for nonlinear transient elasticity, *International Journal for Numerical Methods in Engineering* **78**, 1209–1219, 2009.
- [49] M. W. Gee, U. Küttler, and W. A. Wall, Truly monolithic algebraic multigrid for fluid-structure interaction, *International Journal for Numerical Methods in Engineering* **85**, 987–1016, 2011.
- [50] J.-F. Gerbeau and M. Vidrascu, A quasi-Newton algorithm based on a reduced model for fluid-structure interaction problems in blood flow, *Mathematical Modelling and Numerical Analysis* **37**, 631–647, 2003.
- [51] A. Gerstenberger, *An XFEM based fixed-grid approach to fluid-structure interaction*, PhD thesis, Technische Universität München, 2011.
- [52] A. Gerstenberger and W. A. Wall, Enhancement of fixed-grid methods towards complex fluid-structure interaction applications, *International Journal for Numerical Methods in Fluids* **57**, 1227–1248, 2008.
- [53] A. Gerstenberger and W. A. Wall, An embedded Dirichlet formulation for 3D continua, *International Journal for Numerical Methods in Engineering* **82**, 537–563, 2010.
- [54] M. Gitterle, *A dual mortar formulation for finite deformation frictional contact problems including wear and thermal coupling*, PhD thesis, Technische Universität München, 2012.

-
- [55] M. Gitterle, A. Popp, M. W. Gee, and W. A. Wall, Finite deformation frictional mortar contact using a semi-smooth Newton method with consistent linearization, *International Journal for Numerical Methods in Engineering* **84**, 543–571, 2010.
- [56] R. Gohar, *Elastohydrodynamics*, Imperial College Press, 2001.
- [57] O. Gonzalez, Exact energy and momentum conserving algorithms for general models in nonlinear elasticity, *Computer Methods in Applied Mechanics and Engineering* **190**, 1763–1783, 2000.
- [58] V. Gravemeier and W. A. Wall, Residual-based variational multiscale methods for laminar, transitional and turbulent variable-density flow at low Mach number, *International Journal for Numerical Methods in Fluids* **65**, 1260–1278, 2011.
- [59] P. M. Gresho, R. L. Sani, and M. S. Engelman, *Incompressible flow and the finite element method: Isothermal laminar flow*, Wiley, 2000.
- [60] M. E. Gurtin, *An introduction to continuum mechanics*, Academic Press, 1981.
- [61] C. Hager, *Robust numerical algorithms for dynamic frictional contact problems with different time and space scales*, PhD thesis, Universität Stuttgart, 2010.
- [62] C. Hager and B. I. Wohlmuth, Analysis of a space-time discretization for dynamic elasticity problems based on mass-free surface elements, *SIAM Journal on Numerical Analysis* **47**, 1863–1885, 2009.
- [63] C. Hager, S. Hübner, and B. I. Wohlmuth, A stable energy-conserving approach for frictional contact problems based on quadrature formulas, *International Journal for Numerical Methods in Engineering* **73**, 205–225, 2008.
- [64] J. O. Hallquist, G. L. Goudreau, and D. J. Benson, Sliding interfaces with contact-impact in large-scale Lagrangian computations, *Computer Methods in Applied Mechanics and Engineering* **51**, 107–137, 1985.
- [65] S. Hartmann, *Kontaktanalyse dünnwandiger Strukturen bei großen Deformationen*, PhD thesis, Universität Stuttgart, 2007.
- [66] S. Hartmann, S. Brunssen, E. Ramm, and B. I. Wohlmuth, Unilateral non-linear dynamic contact of thin-walled structures using a primal-dual active set strategy, *International Journal for Numerical Methods in Engineering* **70**, 883–912, 2007.
- [67] S. Hartmann, J. Oliver, R. Weyler, J. Cante, and J. Hernandez, A contact domain method for large deformation frictional contact problems. Part 2: Numerical aspects, *Computer Methods in Applied Mechanics and Engineering* **198**, 2607–2631, 2009.
- [68] 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**, 1–23, 2004.

- [69] M. A. Heroux, R. A. Bartlett, V. E. Howle, R. J. Hoekstra, J. J. Hu, T. G. Kolda, R. B. Lehoucq, K. R. Long, R. P. Pawlowski, E. T. Phipps, A. G. Salinger, H. K. Thornquist, R. S. Tuminaro, J. M. Willenbring, A. Williams, and K. S. Stanley, An overview of the Trilinos project, *ACM Transactions on Mathematical Software* **31**, 397–423, 2005.
- [70] H. Hertz, Über die Berührung fester elastischer Körper, *Journal für die reine und angewandte Mathematik* **92**, 156–171, 1882.
- [71] C. Hesch and P. Betsch, A mortar method for energy-momentum conserving schemes in frictionless dynamic contact problems, *International Journal for Numerical Methods in Engineering* **77**, 1468–1500, 2009.
- [72] C. Hesch and P. Betsch, Transient three-dimensional domain decomposition problems: Frame-indifferent mortar constraints and conserving integration, *International Journal for Numerical Methods in Engineering* **82**, 329–358, 2010.
- [73] C. Hesch and P. Betsch, Transient 3d contact problems - NTS method: mixed methods and conserving integration, *Computational Mechanics* **48**, 437–449, 2011.
- [74] C. Hesch and P. Betsch, Transient three-dimensional contact problems: mortar method. Mixed methods and conserving integration, *Computational Mechanics* **48**, 461–475, 2011.
- [75] P. Hild, Numerical implementation of two nonconforming finite element methods for unilateral contact, *Computer Methods in Applied Mechanics and Engineering* **184**, 99 – 123, 2000.
- [76] P. Hild and P. Laborde, Quadratic finite element methods for unilateral contact problems, *Applied Numerical Mathematics* **41**, 401–421, 2002.
- [77] M. Hintermüller, K. Ito, and K. Kunisch, The primal-dual active set strategy as a semi-smooth Newton method, *SIAM Journal on Optimization* **13**, 865–888, 2002.
- [78] G. A. Holzapfel, *Nonlinear solid mechanics: a continuum approach for engineering*, Wiley, 2000.
- [79] S. Hübner, *Discretization techniques and efficient algorithms for contact problems*, PhD thesis, Universität Stuttgart, 2008.
- [80] S. Hübner and B. I. Wohlmuth, A primal-dual active set strategy for non-linear multibody contact problems, *Computer Methods in Applied Mechanics and Engineering* **194**, 3147–3166, 2005.
- [81] S. Hübner and B. I. Wohlmuth, Thermo-mechanical contact problems on non-matching meshes, *Computer Methods in Applied Mechanics and Engineering* **198**, 1338–1350, 2009.
- [82] S. Hübner and B. I. Wohlmuth, Equilibration techniques for solving contact problems with Coulomb friction, *Computer Methods in Applied Mechanics and Engineering* **205–208**, 29–45, 2012.

-
- [83] S. Hübner, M. Mair, and B. I. Wohlmuth, A priori error estimates and an inexact primal-dual active set strategy for linear and quadratic finite elements applied to multibody contact problems, *Applied Numerical Mathematics* **54**, 555–576, 2005.
- [84] S. Hübner, G. Stadler, and B. I. Wohlmuth, A primal-dual active set algorithm for three-dimensional contact problems with Coulomb friction, *SIAM Journal on Scientific Computing* **30**, 572–596, 2008.
- [85] T. J. R. Hughes, Multiscale phenomena: Green’s functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods, *Computer Methods in Applied Mechanics and Engineering* **127**, 387–401, 1995.
- [86] T. J. R. Hughes, *The finite element method: linear static and dynamic finite element analysis*, Dover Publications, 2000.
- [87] T. J. R. Hughes, R. L. Taylor, J. L. Sackman, A. Curnier, and W. Kanoknukulchai, A finite element method for a class of contact-impact problems, *Computer Methods in Applied Mechanics and Engineering* **8**, 249–276, 1976.
- [88] T. J. R. Hughes, G. Scovazzi, and L. P. Franca, Multiscale and stabilized methods, In E. Stein, R. de Borst, and T. J. R. Hughes (eds.), *Encyclopedia of Computational Mechanics*, pages 5–59, John Wiley & Sons, Chichester, 2004.
- [89] B. M. Irons, Numerical integration applied to finite element methods, In *Proceedings Conference on the Use of Digital Computers in Structural Engineering*, University of Newcastle, 1966.
- [90] K. E. Jansen, C. H. Whiting, and G. M. Hulbert, A generalized- α method for integrating the filtered Navier–Stokes equations with a stabilized finite element method, *Computer Methods in Applied Mechanics and Engineering* **190**, 305–319, 2000.
- [91] K. L. Johnson, *Contact mechanics*, Cambridge University Press, 1985.
- [92] C. Kane, E. A. Repetto, M. Ortiz, and J. E. Marsden, Finite element analysis of nonsmooth contact, *Computer Methods in Applied Mechanics and Engineering* **180**, 1–26, 1999.
- [93] G. Karypis and V. Kumar, A parallel algorithm for multilevel graph partitioning and sparse matrix ordering, *Journal of Parallel and Distributed Computing* **48**, 71–95, 1998.
- [94] N. Kikuchi and J. T. Oden, *Contact problems in elasticity: A study of variational inequalities and finite element methods*, SIAM, Philadelphia, 1988.
- [95] H.-G. Kim, A new coupling strategy for fluid-solid interaction problems by using the interface element method, *International Journal for Numerical Methods in Engineering* **81**, 403–428, 2010.
- [96] T. Klöppel, A. Popp, U. Küttler, and W. A. Wall, Fluid-structure interaction for non-conforming interfaces based on a dual mortar formulation, *Computer Methods in Applied Mechanics and Engineering* **200**, 3111–3126, 2011.

- [97] A. Konyukhov and K. Schweizerhof, On the solvability of closest point projection procedures in contact analysis: Analysis and solution strategy for surfaces of arbitrary geometry, *Computer Methods in Applied Mechanics and Engineering* **197**, 3045–3056, 2008.
- [98] R. Krause, A nonsmooth multiscale method for solving frictional two-body contact problems in 2D and 3D with multigrid efficiency, *SIAM Journal on Scientific Computing* **31**, 1399–1423, 2009.
- [99] R. Krause and B. I. Wohlmuth, Nonconforming domain decomposition techniques for linear elasticity, *East-West Journal of Numerical Mathematics* **8**, 177–206, 2000.
- [100] D. Kuhl and M. A. Crisfield, Energy-conserving and decaying algorithms in non-linear structural dynamics, *International Journal for Numerical Methods in Engineering* **45**, 569–599, 1999.
- [101] D. Kuhl and E. Ramm, Generalized energy-momentum method for non-linear adaptive shell dynamics, *Computer Methods in Applied Mechanics and Engineering* **178**, 343–366, 1999.
- [102] U. Küttler, *Effiziente Lösungsverfahren für Fluid-Struktur-Interaktions-Probleme*, PhD thesis, Technische Universität München, 2009.
- [103] B. P. Lamichhane, *Higher order mortar finite elements with dual Lagrange multiplier spaces and applications*, PhD thesis, Universität Stuttgart, 2006.
- [104] B. P. Lamichhane and B. I. Wohlmuth, Biorthogonal bases with local support and approximation properties, *Mathematics of Computation* **76**, 233–249, 2007.
- [105] B. P. Lamichhane, R. P. Stevenson, and B. I. Wohlmuth, Higher order mortar finite element methods in 3D with dual Lagrange multiplier bases, *Numerische Mathematik* **102**, 93–121, 2005.
- [106] T. A. Laursen, *Formulation and treatment of frictional contact problems using finite elements*, PhD thesis, Stanford University, 1992.
- [107] T. A. Laursen, *Computational contact and impact mechanics*, Springer-Verlag Berlin Heidelberg, 2002.
- [108] T. A. Laursen and V. Chawla, Design of energy conserving algorithms for frictionless dynamic contact problems, *International Journal for Numerical Methods in Engineering* **40**, 863–886, 1997.
- [109] T. A. Laursen and G. R. Love, Improved implicit integrators for transient impact problems - geometric admissibility within the conserving framework, *International Journal for Numerical Methods in Engineering* **53**, 245–274, 2002.
- [110] T. A. Laursen and J. C. Simo, A continuum-based finite element formulation for the implicit solution of multibody, large deformation-frictional contact problems, *International Journal for Numerical Methods in Engineering* **36**, 3451–3485, 1993.

-
- [111] T. A. Laursen, M. A. Puso, and J. Sanders, Mortar contact formulations for deformable-deformable contact: past contributions and new extensions for enriched and embedded interface formulations, *Computer Methods in Applied Mechanics and Engineering* **205–208**, 3–15, 2012.
- [112] D. G. Luenberger and Y. Ye, *Linear and nonlinear programming*, Springer, 2008.
- [113] J. E. Marsden and T. J. R. Hughes, *Mathematical foundations of elasticity*, Dover, 1994.
- [114] U. M. Mayer, A. Gerstenberger, and W. A. Wall, Interface handling for three-dimensional higher-order XFEM-computations in fluid-structure interaction, *International Journal for Numerical Methods in Engineering* **79**, 846–869, 2009.
- [115] U. M. Mayer, A. Popp, A. Gerstenberger, and W. A. Wall, 3D fluid-structure-contact interaction based on a combined XFEM FSI and dual mortar contact approach, *Computational Mechanics* **46**, 53–67, 2010.
- [116] T. W. McDevitt and T. A. Laursen, A mortar-finite element formulation for frictional contact problems, *International Journal for Numerical Methods in Engineering* **48**, 1525–1547, 2000.
- [117] N. Moës, J. Dolbow, and T. Belytschko, A finite element method for crack growth without remeshing, *International Journal for Numerical Methods in Engineering* **46**, 131–150, 1999.
- [118] R. W. Ogden, *Non-linear elastic deformations*, Dover Publications, 1997.
- [119] J. Oliver, S. Hartmann, J. C. Cante, R. Weyler, and J. A. Hernandez, A contact domain method for large deformation frictional contact problems. Part 1: Theoretical basis, *Computer Methods in Applied Mechanics and Engineering* **198**, 2591–2606, 2009.
- [120] V. Padmanabhan and T. A. Laursen, A framework for development of surface smoothing procedures in large deformation frictional contact analysis, *Finite Elements in Analysis and Design* **37**, 173–198, 2001.
- [121] P. Papadopoulos and R. L. Taylor, A mixed formulation for the finite element solution of contact problems, *Computer Methods in Applied Mechanics and Engineering* **94**, 373–389, 1992.
- [122] K. C. Park, C. A. Felippa, and G. Rebel, A simple algorithm for localized construction of non-matching structural interfaces, *International Journal for Numerical Methods in Engineering* **53**, 2117–2142, 2002.
- [123] A. Popp, M. W. Gee, and W. A. Wall, A finite deformation mortar contact formulation using a primal-dual active set strategy, *International Journal for Numerical Methods in Engineering* **79**, 1354–1391, 2009.
- [124] A. Popp, M. Gitterle, M. W. Gee, and W. A. Wall, A dual mortar approach for 3D finite deformation contact with consistent linearization, *International Journal for Numerical Methods in Engineering* **83**, 1428–1465, 2010.

- [125] A. Popp, M. W. Gee, and W. A. Wall, Finite deformation contact based on a 3D dual mortar and semi-smooth Newton approach, In G. Zavarise and P. Wriggers (eds.), *Lecture Notes in Applied and Computational Mechanics*, Volume 58, pages 57–77, Springer-Verlag Berlin Heidelberg, 2011.
- [126] A. Popp, M. W. Gee, and W. A. Wall, Mortar methods for single- and multi-field applications in computational mechanics, In M. Resch, X. Wang, E. Focht, W. Bez, H. Kobayashi, and S. Roller (eds.), *High Performance Computing on Vector Systems 2012*, Springer-Verlag Berlin Heidelberg, 2012.
- [127] A. Popp, A. Seitz, M. W. Gee, and W. A. Wall, A dual mortar approach for improved robustness and consistency of 3D contact algorithms, *Computer Methods in Applied Mechanics and Engineering*, submitted, 2012.
- [128] A. Popp, B. I. Wohlmuth, M. W. Gee, and W. A. Wall, Dual quadratic mortar finite element methods for 3D finite deformation contact, *SIAM Journal on Scientific Computing* **34**, B421–B446, 2012.
- [129] M. A. Puso, A 3D mortar method for solid mechanics, *International Journal for Numerical Methods in Engineering* **59**, 315–336, 2004.
- [130] M. A. Puso and T. A. Laursen, A 3D contact smoothing method using Gregory patches, *International Journal for Numerical Methods in Engineering* **54**, 1161–1194, 2002.
- [131] M. A. Puso and T. A. Laursen, Mesh tying on curved interfaces in 3D, *Engineering Computations* **20**, 305–319, 2003.
- [132] M. A. Puso and T. A. Laursen, A mortar segment-to-segment contact method for large deformation solid mechanics, *Computer Methods in Applied Mechanics and Engineering* **193**, 601–629, 2004.
- [133] M. A. Puso and T. A. Laursen, A mortar segment-to-segment frictional contact method for large deformations, *Computer Methods in Applied Mechanics and Engineering* **193**, 4891–4913, 2004.
- [134] M. A. Puso, T. A. Laursen, and J. Solberg, A segment-to-segment mortar contact method for quadratic elements and large deformations, *Computer Methods in Applied Mechanics and Engineering* **197**, 555–566, 2008.
- [135] L. Qi and J. Sun, A nonsmooth version of Newton’s method, *Mathematical Programming* **58**, 353–367, 1993.
- [136] A. Quarteroni, R. Sacco, and F. Saleri, *Numerical mathematics*, Springer, 2007.
- [137] E. Rabinowicz, *Friction and wear of materials*, Wiley, 1995.
- [138] M. Raous, L. Cangemi, and M. Cocu, A consistent model coupling adhesion, friction, and unilateral contact, *Computer Methods in Applied Mechanics and Engineering* **177**, 383–399, 1999.

-
- [139] J. N. Reddy, *An introduction to nonlinear finite element analysis*, Oxford University Press, 2004.
- [140] M. R. Ross, M. A. Sprague, C. A. Felippa, and K. C. Park, Treatment of acoustic fluid-structure interaction by localized Lagrange multipliers and comparison to alternative interface-coupling methods, *Computer Methods in Applied Mechanics and Engineering* **198**, 986–1005, 2009.
- [141] Y. Saad, *Iterative methods for sparse linear systems*, SIAM, 2003.
- [142] S. Sathe and T. E. Tezduyar, Modeling of fluid-structure interactions with the space-time finite elements: contact problems, *Computational Mechanics* **43**, 51–60, 2008.
- [143] R. A. Sauer, Enriched contact finite elements for stable peeling computations, *International Journal for Numerical Methods in Engineering* **87**, 593–616, 2011.
- [144] M. Schäfer and J. Turek, Benchmark computations of laminar flow around a cylinder, In *Flow simulation with high-performance computers II, Notes on numerical fluid mechanics, volume 52*, pages 547–566, Vieweg, 1996.
- [145] K. Schweizerhof and A. Konyukhov, Covariant description for frictional contact problems, *Computational Mechanics* **35**, 190–213, 2005.
- [146] L. R. Scott and S. Zhang, Finite element interpolation of nonsmooth functions satisfying boundary conditions, *Mathematics of Computation* **54**, 483–492, 1990.
- [147] P. Seshaiyer and M. Suri, hp submeshing via non-conforming finite element methods, *Computer Methods in Applied Mechanics and Engineering* **189**, 1011–1030, 2000.
- [148] J. C. Simo and T. J. R. Hughes, *Computational inelasticity*, Springer, 1998.
- [149] J. C. Simo and T. A. Laursen, An augmented Lagrangian treatment of contact problems involving friction, *Computers & Structures* **42**, 97–116, 1992.
- [150] J. C. Simo and N. Tarnow, The discrete energy-momentum method. Conserving algorithms for nonlinear elastodynamics, *Zeitschrift für Angewandte Mathematik und Physik (ZAMP)* **43**, 757–792, 1992.
- [151] J. C. Simo, P. Wriggers, and R. L. Taylor, A perturbed Lagrangian formulation for the finite element solution of contact problems, *Computer Methods in Applied Mechanics and Engineering* **50**, 163–180, 1985.
- [152] B. F. Smith, P. E. Bjorstad, and W. Gropp, *Domain decomposition: Parallel multilevel methods for elliptic partial differential equations*, Cambridge University Press, 2004.
- [153] N. Strömberg, L. Johansson, and A. Klarbring, Derivation and analysis of a generalized standard model for contact, friction and wear, *International Journal of Solids and Structures* **33**, 1817–1836, 1996.

- [154] S. Stupkiewicz, Finite element treatment of soft elastohydrodynamic lubrication problems in the finite deformation regime, *Computational Mechanics* **44**, 605–619, 2009.
- [155] C. A. Taylor, T. J. R. Hughes, and C. K. Zarins, Finite element modeling of blood flow in arteries, *Computer Methods in Applied Mechanics and Engineering* **158**, 155–196, 1998.
- [156] R. L. Taylor and P. Papadopoulos, On a patch test for contact problems in two dimensions, In P. Wriggers and W. Wagner (eds.), *Computational Methods in Nonlinear Mechanics*, Springer-Verlag Berlin Heidelberg, 1991.
- [157] R. L. Taylor, J. C. Simo, O. C. Zienkiewicz, and A. C. H. Chan, The patch test: A condition for assessing FEM convergence, *International Journal for Numerical Methods in Engineering* **22**, 39–62, 1986.
- [158] I. Temizer, P. Wriggers, and T. J. R. Hughes, Contact treatment in isogeometric analysis with NURBS, *Computer Methods in Applied Mechanics and Engineering* **200**, 1100–1112, 2011.
- [159] I. Temizer, P. Wriggers, and T. J. R. Hughes, Three-dimensional mortar-based frictional contact treatment in isogeometric analysis with NURBS, *Computer Methods in Applied Mechanics and Engineering* **209–212**, 115–128, 2012.
- [160] T. E. Tezduyar and S. Sathe, Modelling of fluid-structure interactions with the space-time finite elements: Solution techniques, *International Journal for Numerical Methods in Fluids* **54**, 855–900, 2007.
- [161] S. P. Timoshenko and J. N. Goodier, *Theory of elasticity*, McGraw-Hill, 1970.
- [162] A. Toselli and O. B. Widlund, *Domain decomposition methods – algorithms and theory*, Springer, 2005.
- [163] M. Tur, F. J. Fuenmayor, and P. Wriggers, A mortar-based frictional contact formulation for large deformations using Lagrange multipliers, *Computer Methods in Applied Mechanics and Engineering* **198**, 2860–2873, 2009.
- [164] R. Unger, M. C. Haupt, and P. Horst, Application of Lagrange multipliers for coupled problems in fluid and structural interactions, *Computers & Structures* **85**, 796–809, 2007.
- [165] R. van Loon, P. D. Anderson, and F. N. van de Vosse, A fluid-structure interaction method with solid-rigid contact for heart valve dynamics, *Journal of Computational Physics* **217**, 806–823, 2006.
- [166] P. Vaněk, J. Mandel, and M. Brezina, Algebraic multigrid by smoothed aggregation for second and fourth order elliptic problems, *Computing* **56**, 179–196, 1996.
- [167] L. Vu-Quoc and X. G. Tan, Optimal solid shells for non-linear analyses of multilayer composites. I. Statics, *Computer Methods in Applied Mechanics and Engineering* **192**, 975–1016, 2003.

-
- [168] L. Vu-Quoc and X. G. Tan, Optimal solid shells for non-linear analyses of multilayer composites. II. Dynamics, *Computer Methods in Applied Mechanics and Engineering* **192**, 1017–1059, 2003.
- [169] W. A. Wall, *Fluid-Struktur-Interaktion mit stabilisierten Finiten Elementen*, PhD thesis, Universität Stuttgart, 1999.
- [170] W. A. Wall and C. J. Cyron, *Nichtlineare Kontinuumsmechanik*, Institute for Computational Mechanics, Technische Universität München, 2011.
- [171] W. A. Wall and M. W. Gee, *BACI: A multiphysics simulation environment*, Technical report, Technische Universität München, 2012.
- [172] C. H. Whiting and K. E. Jansen, A stabilized finite element method for the incompressible Navier–Stokes equations using a hierarchical basis, *International Journal for Numerical Methods in Fluids* **35**, 93–116, 2001.
- [173] J. R. Williams and R. O’Connor, A linear complexity intersection algorithm for discrete element simulation of arbitrary geometries, *Engineering Computations* **12**, 185–201, 1995.
- [174] J. R. Williams and R. O’Connor, Discrete element simulation and the contact problem, *Archives of Computational Methods in Engineering* **6**, 279–304, 1999.
- [175] B. I. Wohlmuth, A mortar finite element method using dual spaces for the Lagrange multiplier, *SIAM Journal on Numerical Analysis* **38**, 989–1012, 2000.
- [176] B. I. Wohlmuth, *Discretization methods and iterative solvers based on domain decomposition*, Springer-Verlag Berlin Heidelberg, 2001.
- [177] B. I. Wohlmuth, Variationally consistent discretization schemes and numerical algorithms for contact problems, *Acta Numerica* **20**, 569–734, 2011.
- [178] B. I. Wohlmuth and R. H. Krause, Monotone multigrid methods on nonmatching grids for nonlinear multibody contact problems, *SIAM Journal on Scientific Computing* **25**, 324–347, 2003.
- [179] B. I. Wohlmuth, A. Popp, M. W. Gee, and W. A. Wall, An abstract framework for a priori estimates for contact problems in 3D with quadratic finite elements, *Computational Mechanics* **49**, 735–747, 2012.
- [180] P. Wriggers, *Computational contact mechanics*, Springer-Verlag Berlin Heidelberg, 2006.
- [181] P. Wriggers and J. Reinelt, Multi-scale approach for frictional contact of elastomers on rough rigid surfaces, *Computer Methods in Applied Mechanics and Engineering* **198**, 1996–2008, 2009.
- [182] P. Wriggers and O. Scherf, An adaptive finite element algorithm for contact problems in plasticity, *Computational Mechanics* **17**, 88–97, 1995.

- [183] P. Wriggers and G. Zavarise, On contact between three-dimensional beams undergoing large deflections, *Communications in Numerical Methods in Engineering* **13**, 429–438, 1997.
- [184] P. Wriggers, T. Vu Van, and E. Stein, Finite element formulation of large deformation impact-contact problems with friction, *Computers & Structures* **37**, 319–331, 1990.
- [185] P. Wriggers, L. Krstulovic-Opara, and J. Korelc, Smooth C1-interpolations for two-dimensional frictional contact problems, *International Journal for Numerical Methods in Engineering* **51**, 1469–1495, 2001.
- [186] B. Yang and T. A. Laursen, A large deformation mortar formulation of self contact with finite sliding, *Computer Methods in Applied Mechanics and Engineering* **197**, 756–772, 2008.
- [187] B. Yang and T. A. Laursen, A contact searching algorithm including bounding volume trees applied to finite sliding mortar formulations, *Computational Mechanics* **41**, 189–205, 2008.
- [188] B. Yang and T. A. Laursen, A mortar-finite element approach to lubricated contact problems, *Computer Methods in Applied Mechanics and Engineering* **198**, 3656–3669, 2009.
- [189] B. Yang, T. A. Laursen, and X. Meng, Two dimensional mortar contact methods for large deformation frictional sliding, *International Journal for Numerical Methods in Engineering* **62**, 1183–1225, 2005.
- [190] G. Zavarise and L. De Lorenzis, A modified node-to-segment algorithm passing the contact patch test, *International Journal for Numerical Methods in Engineering* **79**, 379–416, 2009.
- [191] G. Zavarise and L. De Lorenzis, An augmented lagrangian algorithm for contact mechanics based on linear regression, *International Journal for Numerical Methods in Engineering* **91**, 825–842, 2012.
- [192] G. Zavarise and P. Wriggers, A superlinear convergent augmented Lagrangian procedure for contact problems, *Engineering Computations* **16**, 88–119, 1999.
- [193] G. Zavarise and P. Wriggers, Contact with friction between beams in 3-D space, *International Journal for Numerical Methods in Engineering* **49**, 977–1006, 2000.
- [194] Z.-H. Zhong and L. Nilsson, A contact searching algorithm for general contact problems, *Computers & Structures* **33**, 197–209, 1989.
- [195] Z.-H. Zhong and L. Nilsson, A contact searching algorithm for general 3-D contact-impact problems, *Computers & Structures* **34**, 327–335, 1990.
- [196] O. C. Zienkiewicz and R. L. Taylor, *The finite element method for solid and structural mechanics*, Elsevier Butterworth-Heinemann, 2005.
- [197] O. C. Zienkiewicz, R. L. Taylor, and J. Z. Zhu, *The finite element method: Its basis & fundamentals*, Elsevier Butterworth-Heinemann, 2005.