# Adaptive Finite Element Methods for Nonlinear Hyperbolic Problems of Second Order

**Dissertation**

zur
Erlangung des Grades
eines
Doktors der Naturwissenschaften

Der Fakultät für Mathematik
der Technischen Universität Dortmund
vorgelegt von

Dipl.-Math. **Andreas Josef Rademacher**

aus Arnsberg

am 16. Juli 2009

Adresse des Autors:

**Andreas Rademacher**

Oelinghauser-Heide 43

59757 Arnsberg, Deutschland

E-mail: Andreas.Rademacher@tu-dortmund.de

Die vorliegende Arbeit wurde von der Fakultät für Mathematik der Technischen Universität Dortmund als Dissertation zur Erlangung des Grades eines Doktors der Naturwissenschaften genehmigt.

Promotionsausschuss:

| | |
|---|---|
| Vorsitzender: | Prof. Dr. R. Brück |
| Erster Gutachter: | Prof. Dr. H. Blum |
| Zweiter Gutachter: | Prof. Dr. A. Schröder |
| Zusätzlicher Prüfer: | Prof. Dr. W. Achtziger |
| Beisitzer: | Dr. G. Skoruppa |

| | |
|---|---|
| Tag der Einreichung: | 16. Juli 2009 |
| Tag der Disputation: | 11. September 2009 |

# Danksagung

Die vorliegende Dissertation ist während meiner Tätigkeit am Lehrstuhl X für Wissenschaftliches Rechnen der Fakultät für Mathematik der TU Dortmund entstanden. Dem Lehrstuhlinhaber und meinem Betreuer Prof. Dr. Heribert Blum gilt mein ganz besonderer Dank für die ausgezeichnete Betreuung der Arbeit, seine Unterstützung und seine stete Bereitschaft zur Diskussion der Resultate. Gleiches gilt auch für den Zweitgutachter, Prof. Dr. A. Schröder, der die Entstehung der Arbeit von Beginn an begleitet hat. Der Dank richtet sich selbstverständlich auch an alle weiteren Mitglieder der Prüfungskommission für ihre Bereitschaft zur Mitwirkung und den reibungslosen Ablauf. Bedanken möchte ich mich darüber hinaus bei allen Mitgliedern des Lehrstuhl X für Wissenschaftliches Rechnen für die gute Atmosphäre und die vielen interessanten Diskussionen. Ein besonderer Dank gebührt Dipl. Math. Heiko Kleemann für das Korrekturlesen der Arbeit und die vielen Dikussionen über die softwaretechnische Umsetzung.

Im Rahmen des Projektes “Ganzheitliche Betrachtung des Spindel- Werkstück- Systems beim NC-Formschleifen von Freiformflächen durch simulative Vorhersage des Prozessverhaltens auf Basis numerischer Verfahren” im Schwerpunktprogramm 1180 ”Prognose und Beeinflussung der Wechselwirkungen von Strukturen und Prozessen” der Deutschen Forschungsgemeinschaft (DFG) ist die vorliegende Dissertation entstanden. Das Projekt wird gemeinsam mit dem Institut für Spanende Fertigung der TU Dortmund durchgeführt. Für die gute Zusammenarbeit und die interessanten Einblicke in die Produktionstechnik möchte ich mich bei dem Institutsleiter Prof. Dr.-Ing. Dirk Biermann, bei dem ehemaligen Institutsleiter Prof. i.R. Dr.-Ing. Dr. h.c. Klaus Weinert sowie bei den Mitarbeitern Dr.-Ing. Tim Jansen und Dipl.-Ing. Anastasia Viviana Scheidler bedanken.

Nicht fehlen darf an dieser Stelle der Dank an meine Freundin, Daniela Schneider, meine Eltern, Rita und Heinz-Josef Rademacher, meine Familie sowie meine Freunde und Freundinnen. Die vorliegende Dissertation ist Euch gewidmet, denn ohne euer Verständnis und eure Unterstützung wäre ihre Erstellung niemals möglich gewesen. Besonders bedanken möchte ich mich bei Christoph Müller und Matthias Niemand für die sprachliche Korrektur der Arbeit.

Andreas Rademacher

Dortmund, im September 2009

# Zusammenfassung

Nichtlineare hyperbolische Probleme zweiter Ordnung stehen im Focus der vorliegenden Dissertation. Zwei verschiedene Arten von Nichtlinearitäten werden betrachtet: Quasilineare elliptische Differentialoperatoren und Probleme mit Kontakt. Dabei ist das Ziel, a posteriori Fehlerabschätzungen für den Diskretisierungsfehler in beliebigen nichtlinearen Zielfunktionalen herzuleiten. Darauf aufbauend werden angemessene adaptive Verfeinerungsalgorithmen entwickelt, um die Effizienz der Diskretisierungsverfahren zu steigern.

Im ersten Teil werden quasilineare hyperbolische Gleichungen zweiter Ordnung untersucht. Aufbauend auf einer geeigneten Problemformulierung erfolgt die Diskretisierung mit einem Ort-Zeit Petrov-Galerkin Verfahren. Der dann hergeleitete zielorientierte a posteriori Fehlerschätzer basiert auf der Lösung eines dualen Problems. Zu dessen numerischer Lösung wird das durch den Fehlerschätzer vorgegebene Diskretisierungsverfahren verwendet. Aus dem Fehlerschätzer werden anschließend geeignete Verfeinerungsindikatoren gewonnen, die die Basis für die adaptive Verfeinerung bilden. Die vorgestellten Verfeinerungsstrategien führen zu Finite-Elemente-Netzen, die nicht unbedingt den Anforderungen des Diskretisierungsverfahrens und des Fehlerschätzers genügen. Deshalb wird eine geeignete Netzstruktur mit Regularisierungsalgorithmen sichergestellt. Numerische Resultate belegen die Genauigkeit der a posteriori Fehlerabschätzung und die Effizienz des adaptiven Diskretisierungsverfahrens.

Probleme mit Kontakt bilden den Schwerpunkt des zweiten Teils. Zuerst werden a posteriori Fehlerschätzer, die den Diskretisierungsfehler in beliebigen nichtlinearen Zielfunktionalen messen, für statische Probleme hergeleitet. Dabei erfolgt die Diskretisierung mit einer gemischten Finite-Elemente-Methode. Bei der Behandlung dynamischer Probleme mit Kontakt ist neben den geometrischen Kontaktbedingungen auch eine „Impact“ Bedingung zu berücksichtigen. Auf Grundlage einer Ort-Zeit Petrov-Galerkin Diskretisierung wird dann aufbauend auf der Grundidee des statischen Falls ein a posteriori Fehlerschätzer entwickelt. Mehrere numerische Beispiele zeigen auch in diesem Fall die Genauigkeit der a posteriori Fehlerabschätzung und die Effizienz des adaptiven Diskretisierungsverfahrens.

# Summary

Nonlinear hyperbolic problems of second order are the focus of the thesis at hand. Two different kinds of nonlinearities are considered: Quasilinear elliptic differential operators and contact problems. The aim of this thesis is to derive a posteriori error estimates of the finite element discretisation error measured w.r.t. arbitrary nonlinear functionals of interest. Based on the a posteriori error estimate, adequate adaptive methods are developed to enhance the efficiency of the discretisation.

In the first part of this thesis, quasilinear hyperbolic equations of second order are examined. Based on an adequate problem formulation, a space-time Petrov-Galerkin method is used for discretisation. A goal-oriented a posteriori error estimator is derived. It is based on the solution of a dual problem, which is numerically solved by the discretisation scheme prescribed by the error estimator. Appropriate refinement indicators are deduced from the error estimator. They are the basis of the adaptive algorithm. The presented refinement strategies lead to finite element meshes, which do not necessarily fulfil the requirements of the discretisation scheme and the error estimator. Consequently, regularisation algorithms are developed, which ensure a suitable mesh structure. Numerical results substantiate the accuracy of the a posteriori error estimate and the efficiency of the adaptive discretisation.

Contact problems are the focal point of the second part. At first, a posteriori error estimators concerning static contact problems are derived. They measure the discretisation error w.r.t. an arbitrary nonlinear functional of interest. A mixed finite element method is used to discretise the static contact problems. Beside the geometric contact conditions, an impact condition has to be considered in dynamic contact problems. Based on a space-time Petrov-Galerkin method, an a posteriori error estimator is developed. The main idea in the derivation corresponds to the idea in the static case. Several numerical examples show the accuracy of the a posteriori error estimate and the efficiency of the adaptive discretisation scheme.

# Contents

# Introduction

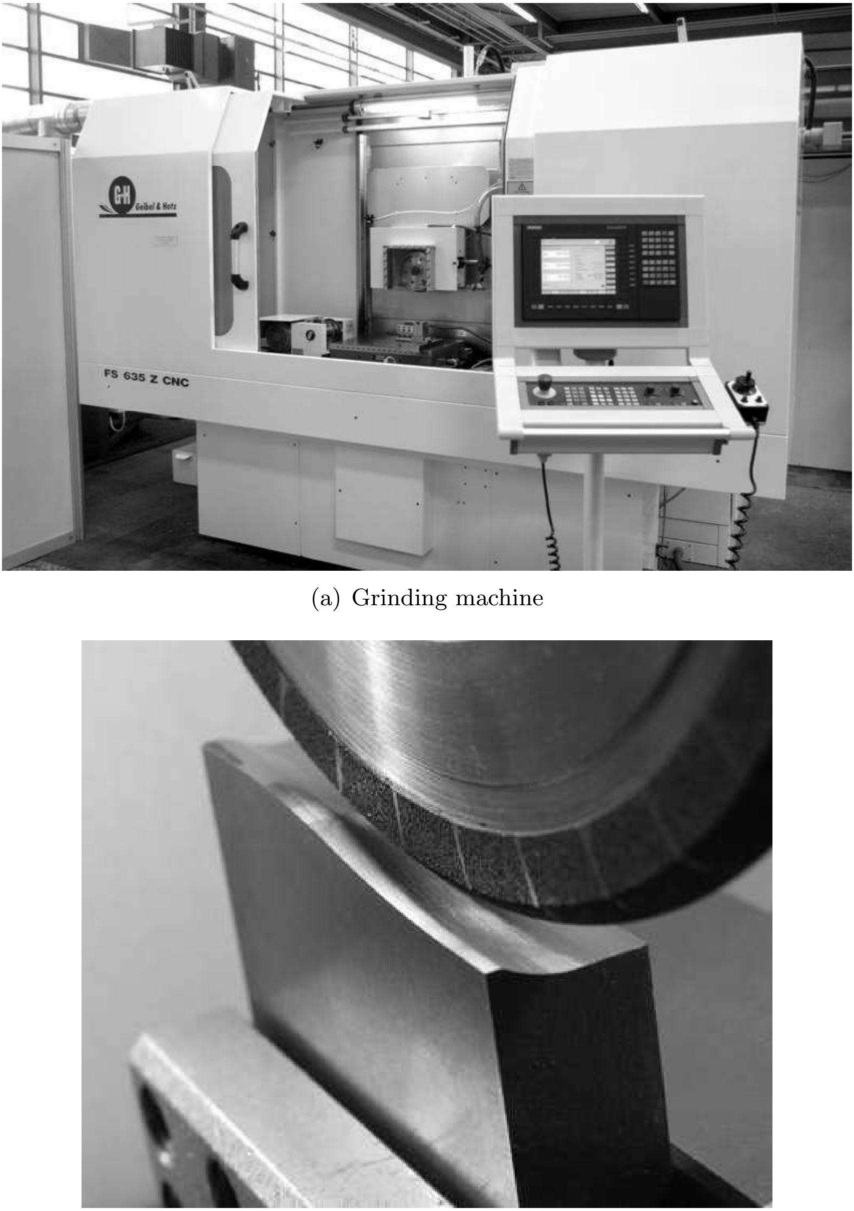
Adaptive finite element methods for nonlinear hyperbolic problems of second order are developed in this thesis. We work in the context of structural mechanics. The solution of the underlying problem, which depends on a spatial and a temporal coordinate, usually is the displacement. The first time derivative of the displacement is the velocity. The acceleration is given by the second time derivative. Hyperbolic problems of second order are composed of the acceleration and of a spatial differential operator applied to the displacement. We address two different nonlinear problem classes: In the first one, quasilinear elliptic differential operators in space are considered, which lead to quasilinear hyperbolic equations of second order. The second problem class consists of dynamic contact problems, for which the differential operator is assumed to be linear. However, the displacement is restricted by a rigid foundation. This leads to a hyperbolic variational inequality of second order. A space-time Galerkin approach is used to discretise the problems. A continuous Galerkin method based on linear trial functions is applied to discretise the temporal part, for the discretisation of the spatial part low order finite elements are used. Adopting the dual weighted residual (DWR) technique, a posteriori error estimates in arbitrary functionals of interest are derived. They are the basis of the adaptive mesh refinement methods, which are applied to enhance the discretisation error at minimum numerical effort.

Nonlinear second order hyperbolic problems arise in many engineering processes, e.g., in milling and grinding processes, vehicle design, and ballistics. Let us take a closer look at a specific engineering process, namely NC-shape grinding with toroid grinding wheels.

## NC-Shape Grinding with Toroid Grinding Wheels

In Figure 1.1.1(a), a photo of a typical grinding machine is depicted. The picture illustrates the different length scales of the system. A photo of the contact area between the grinding wheel and the workpiece is shown in Figure 1.1.1(b). There, the toroid grinding wheel and the free formed workpiece surface are clear to see. Shape grinding with toroid grinding wheels is favoured for achieving high material removal rates while grinding free formed surfaces. Based on the semicircular profile of the grinding wheel and a line by line movement, it is possible to machine geometrically complex surfaces. The contact area between grinding wheel and workpiece is therefore complex and varying. In Figure 1.1.2, the engagement of the grinding wheel is illustrated. See, e.g., [**65**] for a detailed description of the process.

Grinding processes, which are mainly used to finish the surface, are often the last step in the production chain for a workpiece. Accordingly the financial value of the component is already high. On account of this, workpiece errors due to the grinding process are expensive and have to be avoided. An accurate simulation provides a basis to improve this flexible production process and to ensure a suitable process strategy. In our simulation approach, which is described in detail in [**114**], a geometric-kinematical process simulation and a finite element simulation of the decisive parts of the grinding machine are coupled. In the geometric-kinematical simulation, a process force model, see [**20**], is provided, which is based on the contact surface between the grinding wheel and the workpiece. Using a dexel model of the workpiece, see [**113**], the simulation calculates the material-removal and reproduces the movement of the grinding wheel and the workpiece according to the NC data. In the finite element simulation, the grinding wheel and the spindle are explicitly included. The spindle-grinding wheel-system is depicted in Figure 1.1.3. The remaining parts of the grinding machine are modelled by elastic bearings. The workpiece interacts with the grinding wheel only



(a) Grinding machine

(b) Zoom on grinding wheel and workpiece

FIGURE 1.1.1. Picture of the grinding machine

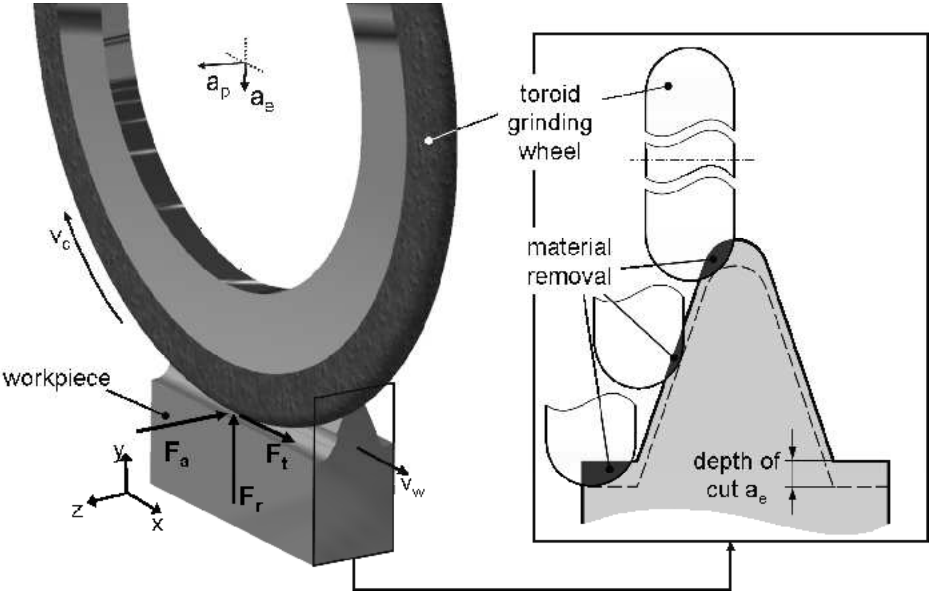


FIGURE 1.1.2. Schematic overview of the engagement of the grinding wheel

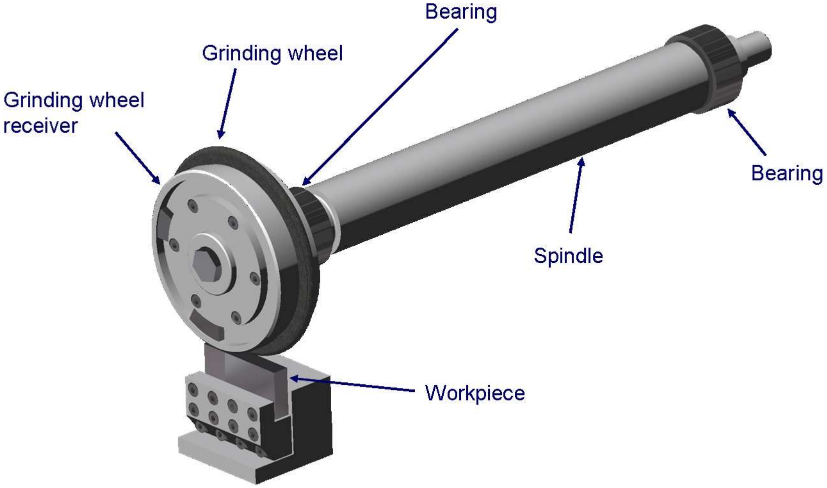


FIGURE 1.1.3. Spindle-grinding wheel-system

in a small contact zone. However, the behaviour of the grinding machine is strongly affected by the resulting contact force. This contact is modelled by a dynamic Signorini problem. The two parts of the simulation are coupled in such a way that the displacements of the grinding wheel determined by the finite element simulation are transferred to the geometric-kinematical simulation, which calculates the removal on the basis of these displacements. The coupling is discussed in detail in [**112**]. Grinding errors result to some extent from the dynamic compliance of the spindle. Consequently, inertial and damping effects have to be considered, see [**65**] for a discussion from the engineering point of view. Another reason for grinding errors are thermal effects. This topic is extensively discussed in [**20**]. The consideration of thermal effects leads to thermomechanic Signorini problems. They include the heat equation for the description of the heat transfer. Furthermore, frictional effects have to be considered because the energy dissipated due to friction is mostly transferred into heat. A finite element method for the discretisation of the mentioned thermomechanic Signorini problem is presented in [**23**].

For the reliable simulation of such a complex process, a precise prediction is required for the contact forces, the contact zone, and their effects onto the whole body. Furthermore, the contact zone and the contact forces are strongly depending on time. Hence, the precise consideration of these dependencies is essential in the numerical simulation. An adequate technique, which gives rise to a flexible and efficient finite element discretisation, is based on a posteriori error control and resulting adaptive mesh refinement. This leads to the aim of the thesis at hand, which is to derive a goal-oriented a posteriori error estimate for dynamic contact problems and to develop an adequate adaptive refinement method based on it. If the restriction is not active in dynamic contact problems, they reduce to linear hyperbolic equations of second order. Consequently, it makes sense to begin the investigations with linear hyperbolic problems of second order. However, we treat the more general quasilinear case, which includes linear equations as a special case.

## Adaptive Finite Elements for Quasilinear Hyperbolic Equations of Second Order

Quasilinear hyperbolic equations of second order are the topic of Chapter 1. They are continuously formulated as a system of first order. We shortly discuss their well-posedness. A space-time Petrov-Galerkin method is used to discretise the quasilinear hyperbolic equations of second order. The trial space consists of piecewise linear and globally continuous basis functions in time. The temporal test functions are piecewise constant and discontinuous. In space, bilinear finite elements are used. The discontinuity of the test functions allows for the decoupling of the time steps. This decoupling leads to the corresponding time stepping scheme. In each time step, the solution of a nonlinear elliptic system and an $L^2$-projection have to be calculated. We present a solution algorithm based on a stabilised Newton method. Furthermore, we discuss the treatment of spatial and temporal hanging nodes. The existence and uniqueness of a discrete solution as well as the convergence to the continuous solution are ensured. The a priori error estimates show that the presented method is convergent of second order in time and that the convergence properties of bilinear finite elements known from elliptic problems carry over to hyperbolic equations of second order. But we also observe that the convergence can be disturbed, if the spatial mesh is changed during the calculation. Consequently, we have to take into account the effects of mesh changes in the adaptive refinement method.

The derivation of a goal-oriented a posteriori error estimate is the focal point of Chapter 1. The aim is to estimate the discretisation error in an arbitrary nonlinear functional of interest. The first step is to represent the error in terms of the discrete primal and dual residual weighted by the approximation error of the dual and primal solution, respectively. The primal problem is the quasilinear hyperbolic equation of second order. The dual problem is specified by the assumptions of the error representation. It is a linear hyperbolic problem of second order, which runs backward in time. The error representation involves a discrete dual problem, which is solved numerically. The solution scheme of the discrete dual problem is prescribed by the assumptions of the error representation, too. We derive the dual time stepping scheme and discuss its properties. It significantly differs from the primal discretisation scheme.

The error representation includes the continuous primal and dual solution. Thus it cannot be evaluated numerically. We introduce an approximative evaluation of the error identity, which is based on a higher order interpolation in space and time of the discrete primal and dual solution. Due to the tensor product structure of the discrete basis functions, the higher order interpolation allows to split the error estimate into two parts, one of them measuring the temporal error and one representing the spatial error. Numerical experiments show the accuracy of the a posteriori error estimate and substantiate the definition of the temporal and spatial error estimator.

### Adaptive Mesh Refinement

The error estimator presented in Chapter 1 does not only provide an accurate estimation of the discretisation error measured in the functional of interest. It also reflects the spatial and temporal distribution of the error. This second property is the basis for the adaptive mesh refinement techniques presented in Chapter 2. First of all, we define appropriate refinement indicators, which represent the error on a single mesh cell. For this purpose, the error estimate is localised to the single mesh cells. In the temporal direction, it is sufficient to consider the integrals over the single subintervals. The localisation in the spatial direction is more involved. We discuss two alternatives: Cellwise integration by parts and filtering. Refinement indicators have to be sorted according to their effective size in the refinement strategy. The evaluation of this measure has a substantial influence on the behaviour of the adaptive refinement method. We discuss several alternative methods. The final step in the preparation of the refinement indicators consists in a smoothing procedure, which is needed because of the oscillatoric behaviour of the underlying problem.

The adaptation of the refinement indicators involves several parameters. Numerical experiments are presented, which provides the basis for the choise of those parameters.

The refinement strategy consists in three parts: The so called space-time refinement strategy decides whether the spatial meshes, the temporal mesh, or both are adaptively refined. We use the optimal mesh strategy to select those spatial mesh cells or those temporal subintervals, respectively, which should be refined. For the approximative evaluation of the error identity, all spatial meshes need to have patch structure. In order to avoid large additional errors due to mesh changes, we only permit temporal hanging nodes of maximum degree one. This property closely connects the meshes of two consecutive time steps. Spatial hanging nodes of higher degrees can lead to oscillations in the discrete solution. Thus, spatial hanging nodes of maximum degree one are only allowed. During the adaptive refinement, the mentioned properties of the mesh sequence may be and usually are destroyed. Consequently, we develop regularisation algorithms, which ensure this mesh structure. After the adaptive refinement of a spatial mesh, the spatial regularisation algorithm restores the patch structure of the mesh and gets rid of all spatial hanging nodes of degree greater than one. The temporal regularisation algorithm removes all backward hanging nodes of degree greater than one in a forward loop over all time steps. Vice versa, all forward hanging nodes of degree greater than one are regularised in a backward loop. The spatial mesh structure is conserved during the temporal regularisation. We analyse the regularisation algorithms and show that they really ensure the mentioned mesh structure. In particular, we obtain that only one temporal regularisation loop is needed. The use of hierarchic meshes is a main ingredient of the proofs. Furthermore, they rely on the fact that a spatial mesh cell is refined only once during the adaptive refinement and the regularisation.

Putting all parts together, we present the adaptive solution algorithm for the quasilinear hyperbolic equations of second order and discuss some implementation issues, such as the mesh management. For comparison, we introduce a heuristic error indicator, which is cheap to be evaluated but controls the error in a global norm. The error indicator is based on a residual estimator in space and on an estimator known from ordinary differential equations in time. Two numerical examples illustrate the adaptive finite element discretisation. We compare the adaptive discretisation based on the presented error estimator to uniform discretisations, to adaptive methods based on the heuristic error indicator, and to explicit methods. The presented adaptive method is more efficient with respect to the number of unknowns as well as in terms of computing time in these examples.

## Adaptive Finite Elements for Static Contact Problems

Chapter 3 is devoted to static contact problems. We discuss the static problems for two reasons: Much more analytical results are known in the static than in the dynamic case. Furthermore, many concepts developed for static problems, such as the solution approach for the discrete problems or the basic idea of the a posteriori error estimate, carry over to the dynamic case. The initial point of our discretisation is the mixed formulation of the static contact problem, where the Lagrange multiplier may be interpreted as contact force. We present a stable finite element discretisation. Bilinear finite elements are applied to discretise the displacement. A piecewise constant approximation of the Lagrange multiplier on a possibly coarser mesh is used. The discretisation leads to a quadratic program in the Lagrange multiplier with sign constraints only, which can be solved by adequate nonlinear optimisation algorithms. The convergence properties of the bilinear finite elements known from the linear elliptic case carry over to the contact problems, if the contact situation is sufficiently smooth. Furthermore, the Lagrange multiplier provides a more accurate approximation of the contact stress than the post-processed values based on the displacement do.

We derive a goal-oriented a posteriori error estimate, where the discretisation error is controlled in an arbitrary, even nonlinear, functional of interest. The basic idea of the derivation is to define a linear elliptic auxiliary problem, which includes the discrete Lagrange multiplier as Neumann boundary condition. The discrete solution of the auxiliary problem is equal to the discrete solution of the contact problem. We additively split the error into two parts. One represents the error between the discrete solution of the contact problem and the continuous solution of the auxiliary problem. This error is estimated by the standard dual weighted residual method, which involves the discrete solution of a linear elliptic dual problem. Using the same dual problem, the second part, namely the error between the continuous solution of the auxiliary problem and the continuous solution of the contact problem, is estimated. This estimate is mainly governed by the discretisation error of the Lagrange multiplier. Using a higher order interpolation of the Lagrange multiplier, we obtain a computational evaluable error estimate. Numerical results show the accuracy of the proposed approach. Defining approriate refinement indicators, we also obtain an efficient adaptive finite element method based on the presented a posteriori error estimate.

## Adaptive Finite Elements for Dynamic Contact Problems

Dynamic contact problems are the topic of Chapter 4. As in the static case, we introduce a mixed problem formulation. Beside the geometrical contact conditions known from static contact problems, an impact law has to be considered. We work with a purely elastic impact model, i.e. energy conservation is enforced during the contact. A mixed space-time Petrov-Galerkin method on the basis of the mixed problem formulation is used to discretise the dynamic contact problems. Especially, the numerical solution of the discrete problems in the single time steps is discussed. A linear optimisation problem with nonlinear constraints has to be solved in each time step. A benchmark or model problem provides first insight into the presented method.

The derivation of the goal-oriented a posteriori error estimate proceeds in many steps analogously to the static case. The auxiliary problem is a linear hyperbolic equation of second order just like the dual problem. We use the error estimator presented in Chapter 1 to estimate the discretisation error with respect to the functional of interest in the auxiliary problem. Using a space-time higher order interpolation of the Lagrange multiplier, we obtain a computationally evaluable estimate for the error between the continuous solution of the auxiliary problem and the continuous solution of the contact problem in the functional of interest. The same technique as in Chapter 1 allows for splitting this estimated error into a spatial and a temporal part. We define adequate refinement indicators and use the techniques presented in Chapter 2 to realise the adaptive mesh refinement. The accuracy of the error estimator and the efficiency of the resulting adaptive method are substantiated by two numerical examples.

Following the description of the numerical examples in Chapter 4, we discuss further open question in the context of adaptive methods for hyperbolic problems of second order and more general for time dependent problems. To this end, we present first results concerning dynamic contact problems, which are beyond the scope of the work at hand. For instance, we discuss a thermomechanic example. The thesis concludes with a discussion of the results in Chapter 5.

CHAPTER 1

# Adaptive Finite Elements for Quasilinear Hyperbolic Equations of Second Order

In this chapter, we discuss quasilinear hyperbolic equations of second order. Starting from the strong formulation, the weak one is presented. After a short discussion of the analytical properties, a Petrov-Galerkin space-time discretisation scheme is introduced and analysed. Then an a posteriori error estimator, which estimates the error with respect to (w.r.t.) an arbitrary functional, is derived and tested by a numerical example.

The continuous Galerkin method, which we use for discretisation, is presented in [**8, 49**]. The velocity is coupled with the time derivative of the displacement not by the $L^2$-scalar product but by the underlying bilinear form in [**47**]. Another possible approach is based on discontinuous Galerkin methods, which are presented, e.g., in [**61, 62, 64, 67, 77**]. Besides the space-time Galerkin approach, a lot of finite difference methods for the temporal discretisation of this problem class exist. An overview is given for instance in the monograph [**60**].

For elliptic problems, many different a posteriori error estimators are presented in literature, see, e.g., the monographs [**4, 111**] for an overview. For hyperbolic equations of second order, only few contributions exist. One approach, which is used to estimate the error in global norms, is based on finite difference discretisations in time. Here, separate error estimators are used for the space and the time direction [**27, 79, 119**] or error estimates for the whole problem are derived [**2, 18**]. The other approach, which is used here, is based on a space-time Galerkin method.

Discontinuous Galerkin schemes are the basis for the error estimators presented in [**7, 67, 78**]. There, the norm of the error in the last time step is controlled, where the dual solution is estimated by analytical arguments and not solved numerically. The same approach for a continuous space-time Galerkin method was presented in [**47**]. The DWR method was used in [**9, 10, 11, 12**] to derive an a posteriori error estimate for the linear wave equation. Only the primal residual was considered and the adaptive algorithm was restricted to the spatial discretisation. Several results concerning goal-oriented adaptive finite element methods for structural dynamics have been published by the group of Schweizerhoff [**71, 72, 83, 87**]. The results from Johnson and Bangerth are used and somewhat extended. An important topic of their work is the reduction of the numerical effort of the error estimation.

Our approach to goal-oriented adaptivity for quasilinear hyperbolic equations of second order is based on optimisation arguments as the approach presented in [**99**] for nonlinear parabolic problems. We will point out the differences from and similarites to the derivation in [**99**] throughout the text.

## 1.1. Quasilinear Hyperbolic Equations of Second Order

After some introductory remarks concerning Sobolev spaces, the strong formulation of quasilinear hyperbolic equations of second order is presented. Using integration by parts, the weak formulation is derived. Analytical properties like existence and uniqueness of weak solutions are discussed.

### 1.1.1. Problem Formulation.

First, we introduce some notations. Let us begin with the basic domain $\Omega \subset \mathbb{R}^2$. Throughout this thesis, we will restrict ourselves to the two dimensional case to simplify the notation. Furthermore, the presentation of adaptive results is more convenient in two dimensions. The coordinates are given by $x = (x_1, x_2)$. The boundary $\Gamma := \partial\Omega$ of $\Omega$ is divided into two parts $\Gamma = \Gamma_D \cup \Gamma_N$, which are mutually disjoint. Each part $\Gamma_D$ or $\Gamma_N$ can be empty. In contrast to elliptic problems, a unique solution of the pure Neumann problem ($\Gamma_N = \partial\Omega$) can be obtained due to the initial values. The time interval is given by $I = [0, T]$ with $T \in \mathbb{R}$, $T > 0$, where 0 is chosen as the initial point without loss of generality.

We study quasilinear hyperbolic equations of second order, whose solution are given by functions $u : \bar{\Omega} \to \mathbb{R}$. To ease the notation, we do not consider systems of equations, although the theory can easily be transferred. However, we discuss some examples of this type. The solution $u$ is called displacement, too. The gradient of the displacement $u$ in space direction is denoted by $\nabla u$. We study differential operators of the form

$$\mathcal{A}(u) := \frac{\partial a_1\left(x, t, u, \nabla u\right)}{\partial x_1} + \frac{\partial a_2\left(x, t, u, \nabla u\right)}{\partial x_2} + b\left(x, t, u, \nabla u\right)$$

with $a_1, a_2, b : \Omega \times I \times \mathbb{R} \times \mathbb{R}^2 \to \mathbb{R}$, where $a_1$ and $a_2$ have to be continuously differentiable and $b$ continuous. Some examples of operators of this type are

$$\Delta u, \quad \Delta u - u^3, \quad \operatorname{div}\left(\mathbf{a}\left(\nabla u\right)\right),$$

where $\Delta u$ is the usual Laplace operator applied to $u$. The operator $\mathcal{A}$ is called linear, if the functions $a_1$, $a_2$, and $b$ are linear. For linear functions $a_1$ and $a_2$ as well as a nonlinear function $b$, it is semilinear. In the case of $a_1$, $a_2$, and $b$ nonlinear, we refer to $\mathcal{A}$ as quasilinear operator.

We specify initial and boundary values, $u_s$ for the initial displacement and $v_s$ for the initial velocity. For notational simplicity, we assume homogeneous Dirichlet boundary conditions on $\Gamma_D$. The nonhomogeneous Neumann boundary conditions on $\Gamma_N$ are given by $q$. The first and second time derivatives are denoted by $\dot{u}$ and $\ddot{u}$, respectively. The gradient operator in the outward normal direction $\nu(x)$ on the boundary connected to $\mathcal{A}$ is called $\mathcal{B}$. If the boundary and $u$ are sufficiently smooth, we obtain the explicit form

$$\mathcal{B}(u) = a_1\left(x, t, u, \nabla u\right) \cdot \nu_1 + a_2\left(x, t, u, \nabla u\right) \cdot \nu_2.$$

Definition 1.1.1. A function $u \in C^2(\Omega \times I)$ is a strong solution of the quasilinear second order hyperbolic initial/boundary-value problem, if and only if

$$\begin{aligned} \rho\ddot{u} - \mathcal{A}(u) &= f && \text{in } \Omega \times I && (1.1.1)\\ u &= 0 && \text{on } \Gamma_D \times I && \\ \mathcal{B}(u) &= q && \text{on } \Gamma_N \times I && (1.1.2)\\ u(0) &= u_s && \text{in } \Omega && \\ \dot{u}(0) &= v_s && \text{in } \Omega && \end{aligned}$$

holds.

The density $\rho$ is set to 1 for ease of notation.

### 1.1.2. Sobolev Spaces.

In this section, we briefly present the underlying function spaces. A detailed description of Sobolev spaces can be found, e.g., in [**1**]. An overview of the spaces, which are mainly used in the context of contact problems, is given in [**70**]. For the time dependent Sobolev spaces, see, e.g., [**42, 48**].

The basic function space is $L^2(\Omega) := \mathcal{L}^2(\Omega)/\mathcal{N}(\Omega)$, where

$$\begin{aligned} \mathcal{L}^2(\Omega) &:= \left\{ f : \Omega \to \mathbb{R} \,\middle|\, f \text{ measurable}, \int_\Omega f^2 \, dx < \infty \right\} \\ \mathcal{N}(\Omega) &:= \left\{ f \in \mathcal{L}^2(\Omega) \,\middle|\, \int_\Omega f^2 \, dx = 0 \right\}. \end{aligned}$$

To ease the notation, we do not distinguish between $[f]$ and $f$ any longer. The $L^2$-scalar product is defined by

$$(\omega, \varphi) := (\omega, \varphi)_\Omega := \int_\Omega \omega\varphi \, dx$$

for $\omega, \varphi \in L^2(\Omega)$. The space $L^2(\Omega)$ provided with the scalar product $(\cdot, \cdot)$ is a Hilbert space and the corresponding norm is $\|\omega\|_0^2 := \|\omega\|_{0,\Omega}^2 := (\omega, \omega)$.

Let $\alpha = (\alpha_1, \alpha_2) \in \mathbb{N}_0^2$ be a multiindex of order $|\alpha| = \alpha_1 + \alpha_2$. The partial derivative w.r.t. $\alpha$ in the classic sense is given by

$$\partial^\alpha = \frac{\partial^{|\alpha|}}{\partial^{\alpha_1} x_1 \partial^{\alpha_2} x_2}.$$

The weak partial derivative $\partial^\alpha \omega \in L^2(\Omega)$ of a function $\omega \in L^2(\Omega)$ is defined by

$$\forall \varphi \in C_0^\infty(\Omega) : \quad (\partial^\alpha \omega, \varphi) = (-1)^{|\alpha|} (\omega, \partial^\alpha \varphi),$$

where $C_0^\infty(\Omega)$ is the space of infinitely times differentiable functions with compact support. The weak derivative is well defined and it corresponds to the classic derivative, if $\omega \in C^{|\alpha|}(\Omega)$. The Sobolev space $H^k(\Omega)$ consists of all functions $u \in L^2(\Omega)$ such that for each multiindex $\alpha$ with $|\alpha| \leq k$, $\partial^\alpha u$ exists in the weak sense and belongs to $L^2(\Omega)$. The space $H^k(\Omega)$ provided with the scalar product

$$(\omega, \varphi)_k := \sum_{|\alpha| \leq k} (\partial^\alpha \omega, \partial^\alpha \varphi)$$

is a Hilbert space and the corresponding norm is

$$\|\omega\|_k^2 := \|\omega\|_{k,\Omega}^2 := (\omega, \omega)_k .$$

We also use the $H^1$-semi-norm $|u|_1$ with $|u|_1^2 := (\nabla u, \nabla u)$.

If $\Omega$ is a bounded domain with a piecewise smooth boundary and if $\Omega$ satisfies a cone property, then there exists a bounded, linear operator $\gamma : H^1(\Omega) \to L^2(\partial\Omega)$ such that $\gamma(v) = v_{|\partial\Omega}$ if $u \in H^1(\Omega) \cap C(\bar{\Omega})$ and $\|\gamma(v)\|_{0,\partial\Omega} \leq C \|v\|_1$ for each $v \in H^1(\Omega)$. The operator $\gamma$ is called trace operator and $\gamma(u)$ trace of $u$ on the boundary. The Hilbert space

$$H^{1/2}(\partial\Omega) := \gamma\left(H^1(\Omega)\right)$$

with the norm

$$\|w\|_{1/2} := \inf_{\substack{v \in H^1(\Omega) \\ \gamma(v) = w}} \|v\|_1$$

is dense in $L^2(\partial\Omega)$. The dual space of $H^{1/2}(\partial\Omega)$ is $H^{-1/2}(\partial\Omega)$ with the norm $\|\cdot\|_{-1/2}$.

Using the trace operator, we define

$$H^1(\Omega,\Gamma_D) := \left\{\varphi \in H^1(\Omega) \,\middle|\, \gamma_{|\Gamma_D}(\varphi) = 0\right\}.$$

Furthermore, we set $H_0^1(\Omega) := H^1(\Omega,\partial\Omega)$. The space $H_0^1(\Omega)$ can equivalently be defined as the closure of $C_0^\infty(\Omega)$ in $H^1(\Omega)$. The dual space $\left(H_0^1(\Omega)\right)^\star$ is called $H^{-1}(\Omega)$, too. The dual pairing is denoted by $\langle\cdot,\cdot\rangle$.

We define the space $H^{1/2}(\Gamma_N) := \gamma\left(H^1(\Omega,\Gamma_D)\right)$. It is a closed subspace of $H^{1/2}(\partial\Omega)$ and is a Hilbert space provided with the associated norm $\|\cdot\|_{1/2,\Gamma_N}$. The space $H^{-1/2}(\Gamma_N)$ is the dual space of $H^{1/2}(\Gamma_N)$. The norm connected to $H^{-1/2}(\Gamma_N)$ is called $\|\cdot\|_{-1/2,\Gamma_N}$.

Using the Bochner integral theory, we can study Sobolev spaces involving time. Here, $X$ is a real Banach space with norm $\|\cdot\|$. The space $L^p(I;X)$ consists of all strongly measureable functions $\varphi : I \to X$ with

$$\|\varphi\|_{L^p(I;X)} := \left(\int_I \|\varphi\|^p \, dt\right)^{1/p} < \infty, \qquad (1 \le p < \infty),$$

and

$$\|\varphi\|_{L^\infty(I;X)} := \operatorname*{ess\,sup}_{t\in I} \|\varphi(t)\| < \infty, \qquad (p = \infty).$$

All continuous functions $\varphi : I \to X$ with

$$\|\varphi\|_{C(I;X)} := \max_{t\in I} \|\varphi(t)\| < \infty$$

form the space $C(I;X)$. The first weak time derivative of a function $u \in L^2(I;X)$ is denoted by $\dot{u} \in L^2(I;X)$, where

$$\int_I \dot{u}(t)\phi(t)\, dt = -\int_I u(t)\dot{\phi}(t)\, dt$$

holds for all scalar test functions $\phi \in C_0^\infty(I)$. The second weak time derivative $\ddot{u}$ is analogously defined. As in the spatial setting, the weak time derivatives correspond to the usual time derivative, if $u$ is smooth. If $X$ is a Hilbert space with scalar product $(\cdot,\cdot)$, then the space-time scalar product is denoted by

$$((u,v)) := \int_I (u(t), v(t))\ dt.$$

In general, an outer parenthesis denotes the integration over $I$. The following result is the basis for the definition of initial values:

PROPOSITION 1.1.2. *Suppose $\varphi \in W$ with*

$$W := \left\{ u \in L^2\left(I; H^1\left(\Omega, \Gamma_D\right)\right) \middle| \begin{array}{l} \dot{u} \in L^2\left(I; L^2\left(\Omega\right)\right), \\ \ddot{u} \in L^2\left(I; \left(H^1\left(\Omega, \Gamma_D\right)^\star\right)\right) \end{array} \right\}.$$

*Then $\varphi \in C\left(I; H^1\left(\Omega, \Gamma_D\right)\right)$ and $\dot{\varphi} \in C\left(I; L^2\left(\Omega\right)\right)$.*

PROOF OF PROPOSITION 1.1.2. Remark 1, in [**42**], Chapter XVIII, Section 5. □

**1.1.3. Weak Solutions.** After the discussion of the appropriate function spaces, we come to the definition of a weak solution. We need the semi-linear form

$$\begin{aligned} a\left(u\right)\left(\varphi\right): &= \left(a_1\left(x,t,u,\nabla u\right), \frac{\partial \varphi}{\partial x_1}\right) + \left(a_2\left(x,t,u\nabla u\right), \frac{\partial \varphi}{\partial x_2}\right) \\ &\quad + \left(b\left(x,t,u,\nabla u\right), \varphi\right) \end{aligned}$$

with $u, \varphi \in H^1\left(\Omega, \Gamma_D\right)$ and $a_1, a_2, b \in L^2\left(\Omega \times I \times \mathbb{R} \times \mathbb{R}^2\right)$. If it holds $u \in H^1\left(\Omega, \Gamma_D\right) \cap H^2\left(\Omega\right)$, then

$$a(u)(\varphi) = (-\mathcal{A}(u), \varphi) + (\mathcal{B}(u), \varphi)_{\Gamma_N}$$

holds by integration by parts for all $\varphi \in H^1\left(\Omega, \Gamma_D\right)$.

The right hand side $f$ belongs to $L^2\left(I; L^2\left(\Omega\right)\right)$ and $q$ to $L^2\left(I; L^2\left(\Gamma_N\right)\right)$. We define a weak solution of the quasilinear hyperbolic problem of second order as follows:

DEFINITION 1.1.3. A function $u \in W$ is a weak solution of the quasilinear hyperbolic equation of second order provided $u(0) = u_s \in H^1\left(\Omega, \Gamma_D\right)$, $\dot{u}(0) = v_s \in L^2\left(\Omega\right)$, and

$$\langle \ddot{u}, \varphi \rangle + a\left(u\right)\left(\varphi\right) = (f, \varphi) + (q, \varphi)_{\Gamma_N}$$

for each $\varphi \in H^1(\Omega, \Gamma_D)$ and a.e. $t \in I$.

Remark 1.1.4. By Proposition 1.1.2, we know

$$\begin{array}{rcl} u & \in & C\left(I; H^1(\Omega, \Gamma_D)\right), \\ \dot{u} & \in & C\left(I; L^2(\Omega)\right). \end{array}$$

Consequently, the statement of the initial conditions makes sense.

The basis for the discretisation scheme presented in Section 1.2 is the following definition of a weak solution:

Definition 1.1.5. A weak solution of the quasilinear second order hyperbolic problem is given by a function $w = (u, v) \in U \times V$, if

$$\forall \varphi = (\psi, \chi) \in V \times U : \quad A(w)(\varphi) = 0 \tag{1.1.3}$$

holds. The function $u$ represents the displacement and $v$ the velocity.

We have rewritten the hyperbolic problem of second order as a first order system. The space-time semilinear form is given by

$$\begin{array}{rcl} A(w)(\varphi) & := & ((\dot{u} - v, \psi)) + ((\dot{v}, \chi)) \\ & & + \left(a(u)(\chi)\right) - ((f, \chi)) - \left((q, \chi)_{\Gamma_N}\right) \\ & & + (u(0) - u_s, \chi(0)) + (v(0) - v_s, \psi(0)), \end{array}$$

where the initial conditions are enforced weakly through the terms

$$(u(0) - u_s, \chi(0)) \text{ and } (v(0) - v_s, \psi(0)).$$

The weak form of the partial differential equation and of the Neumann boundary conditions is contained in

$$((\dot{v}, \chi)) + \left(a(u)(\chi)\right) - ((f, \chi)) - \left((q, \chi)_{\Gamma_N}\right).$$

The term

$$((\dot{u} - v, \psi))$$

weakly ensures the equality of $v$ and $\dot{u}$. The function spaces $U$ and $V$ are defined as

$$\begin{aligned} U &:= \left\{\psi \in L^2\left(I; H^1\left(\Omega, \Gamma_D\right)\right) \middle| \dot{\psi} \in L^2\left(I; H^1\left(\Omega, \Gamma_D\right)\right)\right\}, \\ V &:= \left\{\chi \in L^2\left(I; H^1\left(\Omega, \Gamma_D\right)\right) \middle| \dot{\chi} \in L^2\left(I; L^2\left(\Omega\right)\right)\right\}. \end{aligned}$$

REMARK 1.1.6. It holds $U \subset C\left(I; H^1\left(\Omega, \Gamma_D\right)\right)$ and $V \subset C\left(I; H^1\left(\Omega, \Gamma_D\right)\right)$. The assumptions on the trial spaces $U$ and $V$ are somewhat stronger than those on $W$ in Definition 1.1.3. They are only made to ease the notation.

**1.1.4. Properties of the Continuous Solution.** In this section, we study the analytical properties of the continuous solution of the quasilinear hyperbolic equation of second order. Let us begin with the existence and uniqueness of a weak solution. This question cannot be answered in the general setting. To be complete, we mention the results for some important special cases: We start with an existence and uniqueness result for the linear case:

PROPOSITION 1.1.7. *Let $\mathcal{A}$ be a linear and uniformly elliptic operator. Then a unique weak solution exists. It depends continuously on the data.*

PROOF OF PROPOSITION 1.1.7. [**42**], Chapter XVIII, Section 5. □

We have the following regularity result:

PROPOSITION 1.1.8. *Assume $\mathcal{A}$ to be linear, $\Gamma_D = \partial\Omega$, $u_s \in H^{m+1}(\Omega) \cap H_0^1(\Omega)$, $v_s \in H^m(\Omega)$, and $\frac{\partial^k f}{\partial t^k} \in L^2\left(I; H^{m-k}(\Omega)\right), \quad k = 0, \ldots, m$. Also suppose that the $m^{th}$-order compatibility conditions hold. Then*

$$\frac{\partial^k u}{\partial t^k} \in L^\infty\left(I; H^{m+1-k}(\Omega)\right), \quad k = 0, \ldots, m+1.$$

PROOF OF PROPOSITION 1.1.8. [**48**], Section 7.2. □

In the semilinear case, there exists a unique local solution:

PROPOSITION 1.1.9. *If $\mathcal{A}$ is a semilinear operator, which fulfils the growth- and smoothness-assumption specified in* [**36**], *then a unique solution exists on* $[0, T(u_s, v_s)]$, *where* $u_s$, $v_s$, *and* $f$ *are sufficiently smooth.*

PROOF OF PROPOSITION 1.1.9. [**36**], Chapter 6. □

Finally, we consider the quasilinear case:

PROPOSITION 1.1.10. *Suppose $\mathcal{A}$ is a quasilinear operator satisfying some strong growth- and smoothness-assumption given, e.g., in* [**63**]. *Furthermore, let* $u_s$, $v_s$, *and* $f$ *be sufficiently smooth. Then there exists a unique solution on* $[0, T(u_s, v_s)]$.

PROOF OF PROPOSITION 1.1.10. The proof of this assertion has been managed only under non-optimal smoothness assumptions on $u_s$ and $v_s$, yet. A first result is contained in [**63**]. An improved result is given, e.g., in [**50**]. See the references therein for further results. □

We denote the Fréchet derivative of an operator $\mathcal{D}$ by $\mathcal{D}'$, see, e.g., [**115**] Section III.5. We make the following assumption, which ensures that the numerical solution algorithm and the a posteriori error estimate work:

ASSUMPTION 1.1.11. *Let the semilinear form $a(\cdot)(\cdot)$ be three times continuously Fréchet differentiable w.r.t. the first argument. We assume that an isolated weak solution $w = (u, v)$ according to Definition 1.1.5 exists. Furthermore, the Fréchet derivative of $A$ leads to a well-posed linear problem in a neighbourhood of $w = (u, v)$.*

The solution of a hyperbolic problem of second order fulfils several conservation properties. Of great importance is the conservation of energy. We want to discuss the conservation of energy in the linear case in detail, because this topic becomes important in the treatment of dynamic contact problems. The energy can only be conserved, if no outer forces act on the body. Consequently, $f$ and $q$ are zero. The kinetic energy is given by

$$E_{kin} := \frac{1}{2}\left(\dot{u}, \dot{u}\right) = \frac{1}{2}\left(v, v\right).$$

The potential or inner energy is specified by $E_{in} := \frac{1}{2}a\left(u, u\right)$. The total energy is $E_{tot} := E_{in} + E_{kin}$.

PROPOSITION 1.1.12. *If $\varphi = (-\dot{v}, \dot{u})$ is a suitable test function in equation* (1.1.3)*, then the total energy $E_{tot}$ is conserved.*

PROOF OF PROPOSITION 1.1.12. Since $\varphi = (-\dot{v}, \dot{u})$ is a suitable test function in equation (1.1.3), we obtain

$$\begin{aligned} 0 &= \left(-\left(\dot{u} - v, \dot{v}\right)\right) + \left(\left(\dot{v}, \dot{u}\right)\right) + \left(a\left(u, \dot{u}\right)\right) \\ &= \left(\frac{\partial}{\partial t}\frac{1}{2}\left(v, v\right)\right) + \left(\frac{\partial}{\partial t}\frac{1}{2}a\left(u, u\right)\right) \\ &= E_{kin}(T) - E_{kin}(0) + E_{in}(T) - E_{in}(0) \\ &= E_{tot}(T) - E_{tot}(0). \end{aligned}$$

The final point in time $T$ can be chosen arbitrarily. Consequently, the energy is constant for all $t \in I$. □

## 1.2. Finite Element Discretisation in Space and Time

In this section, the discretisation scheme for the quasilinear hyperbolic equations of second order is introduced. It is based on a space-time finite element ansatz. After the introduction of the finite element approach, the solution algorithm of the discrete problem is discussed. Finally, the discretisation approach is analysed.

**1.2.1. A Continuous Petrov Galerkin Method.** We present the space-time finite element method. The temporal discretisation is based on a decomposition of the time interval $I = [0, T]$ into $M \in \mathbb{N}$ subintervals $I_m = (t_{m-1}, t_m]$ with

$$0 = t_0 < t_1 < \ldots < t_M = T \quad \text{and} \quad I = \{0\} \cup I_1 \cup \ldots \cup I_M.$$

The length of a subinterval $I_m$ is $k_m := t_m - t_{m-1}$. The time instances $t_i$, $0 \leq i \leq M$ correspond to the time steps in a finite difference approach. We also call this decomposition the temporal mesh $\mathbb{T}_k$. By the time step $m$, we denote the step from $t_{m-1}$ to $t_m$.

The basic domain $\Omega$ is subdivided by meshes $\mathbb{T}_h^m = \{\mathcal{T}_1^m, \ldots, \mathcal{T}_{N^m}^m\}$, $0 \leq m \leq M$. Here, $\mathcal{T}_i^m$ are the single mesh cells with

$$\bigcup_{\mathcal{T} \in \mathbb{T}_h^m} \bar{\mathcal{T}} = \bar{\Omega},$$

where $\overline{\omega}$ denotes the closure of a set $\omega$. We work with quadrilaterals in this thesis. All mesh cells are open and $\mathcal{T}_i^m \cap \mathcal{T}_j^m = \emptyset$ for $i \neq j$. Furthermore, the meshes $\mathbb{T}_h^m$ have to be regular, i.e. $\bar{\mathcal{T}}_i^m \cap \bar{\mathcal{T}}_j^m$ is empty, a vertex of $\mathcal{T}_i$, or an edge of $\mathcal{T}_i$. The mesh width of $\mathbb{T}_h^m$ is defined as

$$h^m := \max_{\mathcal{T} \in \mathbb{T}_h^m} h_{\mathcal{T}},$$

where

$$h_{\mathcal{T}} := \sup_{x,y \in \mathcal{T}} \|x - y\|_2$$

is the maximum diameter of an element $\mathcal{T}$. A sequence of meshes $\{\mathbb{T}_h^m\}$ is called shape regular or quasiuniform, if and only if a number $\kappa > 0$ exists such that

$$\forall h > 0 : \quad \min_{\mathcal{T} \in \mathbb{T}_h^m} r_{\mathcal{T}} \geq \kappa h,$$

with

$$r_{\mathcal{T}} := \sup \left\{ r \in \mathbb{R}_{>0} \, | B_r(x) \subset \mathcal{T}, \, x \in \mathcal{T} \right\},$$

where $B_r(x) := \left\{ y \in \mathbb{R}^2 \middle| \, \|x - y\|_2 \leq r \right\}$ is the closed ball with radius $r$. Starting from now, we assume shape regularity. The sequence of the meshes in the single time steps is called $\mathbb{M}_h := (\mathbb{T}_h^m)_{0 \leq m \leq M}$.

The next step is the definition of the basis functions on the mesh elements. We begin with the spatial ones. Bilinear basis functions are used on the quadrilaterals. Together with the spatial mesh, they form the finite element space

$$V_h^m := \left\{ \varphi \in C\left(\bar{\Omega}, \Gamma_D\right) \middle| \forall \mathcal{T} \in \mathbb{T}_h^m : \, \varphi_{|\mathcal{T}} \in Q_1\left(\mathcal{T}; \mathbb{R}\right) \right\} \subset H^1\left(\Omega, \Gamma_D\right).$$

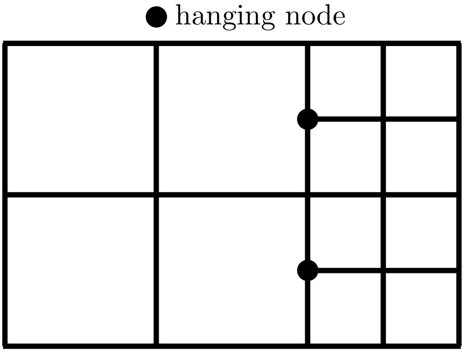
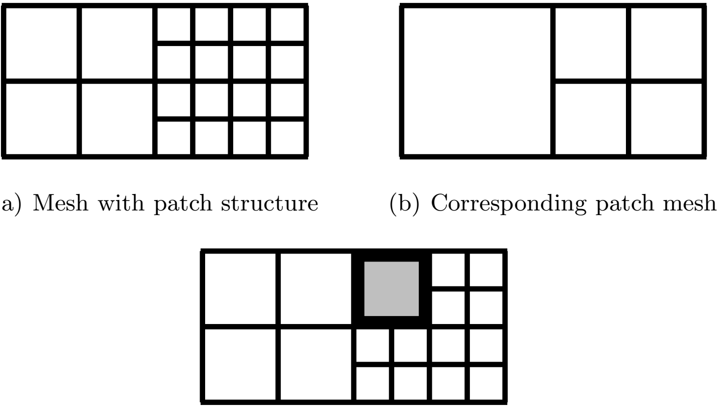


Figure 1.2.1. Mesh with hanging nodes



(a) Mesh with patch structure

(b) Corresponding patch mesh

(c) Mesh without patch structure

Figure 1.2.2. Graphical description of meshes with and without patch structure

Here, $Q_1(\mathcal{T};\mathbb{R})$ is the set of bilinear basis functions on a mesh cell $\mathcal{T}$. A detailed presentation of the spatial finite element ansatz may be found, e.g., in [**30**].

To realise adaptive mesh refinement, we have to allow for so called hanging nodes in the discretisation, see Figure 1.2.1 for an illustration. Since functions from $V_h^m$ are continuous, the basis value connected to this node is not free, but determined by the adjacent nodes. An extensive discussion of this topic is found, e.g., in [**34**]. The mathematical description and the used FE-software SOFAR (c.f. [**22**]) allow the use of more than one hanging node per edge. However, we restrict ourselves to one hanging node per edge for stability reasons. An additional property, which has to be satisfied by the mesh, is the patch structure. It is illustrated in Figure 1.2.2. We need the patch structure for the evaluation of the presented a posteriori error estimate, see Section 1.3.3. We say, a mesh has patch structure, if one can always merge four adjacent mesh cells to one patch element. In Chapter 2, we extensively discuss hanging nodes, the patch structure and the regularisation algorithms, which ensure that all adaptively refined meshes have these properties.

After the presentation of the spatial finite element spaces $V_h^m$, we will now discuss the trial and the test space for the space-time finite element method. Let us begin with the test space: It is defined as

$$W_{kh} := \left\{ \varphi_{kh} \in L^2\left(I; H^1\left(\Omega, \Gamma_D\right)\right) \,\middle|\, \begin{array}{c} \varphi_{kh|I_m} \in \mathcal{P}_0\left(I_m; V_h^m\right), \\ m = 1, 2, \dots, M, \\ \varphi_{kh}(0) \in V_h^0 \end{array} \right\}.$$

Here, $\mathcal{P}_q\left(\omega; X\right)$ is the linear space of polynomials on $\omega \subset \mathbb{R}^d$ with values in $X$, which have the maximum degree $q$. Functions from $W_{kh}$ are piecewise constant and are possibly discontinuous at $t_i$, $i = 0, 1, \dots, M$. The definition of the trial space $V_{kh}$ is more involved, since it is difficult to ensure the global continuity, if the spaces $V_h^m$ vary. Then hanging nodes in time arise and have to be treated in an appropriate way. A temporal hanging node is a degree of freedom, which is contained in $V_h^m$ but not in $V_h^{m-1}$ or vice versa. See Section 2.3 for the precise definition. We work with the approach presented in [**16, 85, 99**] for parabolic problems. A discussion of hanging nodes in time in the context of the wave equation is given in [**9**]. We use linear temporal basis functions and choose the usual Lagrange basis of $\mathcal{P}_1\left(I_m; \mathbb{R}\right)$

$$\tau_0(t) = \frac{t_m - t}{k_m} \quad \text{and} \quad \tau_1(t) = \frac{t - t_{m-1}}{k_m}.$$

We define the set of the local basis functions by

$$\tilde{\mathcal{P}}_1^m := \left\{ \tau_i \varphi_i \,\middle|\, \varphi_i \in V_h^{m-1+i},\, i = 0, 1 \right\}.$$

The space $\tilde{\mathcal{P}}_1^m$ coincides with $\mathcal{P}_1(I_m; V_h)$, if $V_h^{m-1} = V_h^m = V_h$ holds. The trial space is given by

$$V_{kh} := \left\{ \varphi_{kh} \in C\left(I; H^1(\Omega, \Gamma_D)\right) \,\middle|\, \varphi_{kh|I_m} \in \tilde{\mathcal{P}}_1^m,\, m = 1, 2, \dots, M \right\}.$$

Spatial basis functions from $V_h^{m-1}$, which vanish when stepping from $t_{m-1}$ to $t_m$, are only connected to the temporal basis funtion $\tau_0$, which is zero at $t_m$. Spatial basis functions from $V_h^m$, which arise when stepping from $t_{m-1}$ to $t_m$, are only coupled with the temporal basis function $\tau_1$ and $\tau_1$ vanishes at $t_{m-1}$. Thus, all functions in $V_{kh}$ are globally continuous.

Eventually, we can define the discrete solution:

Definition 1.2.1. We say $w_{kh} = (u_{kh}, v_{kh}) \in V_{kh} \times V_{kh}$ is a discrete solution of the quasilinear hyperbolic equation of second order, if and only if

$$\forall \varphi_{kh} \in W_{kh} \times W_{kh} : \quad A_{kh}(w_{kh})(\varphi_{kh}) = 0. \tag{1.2.1}$$

holds with

$$\begin{aligned} A_{kh}(w_{kh})(\varphi_{kh}) \quad := \quad & \sum_{m=1}^{M} \{((\dot{u}_{kh} - v_{kh}, \psi_{kh}))_m + ((\dot{v}_{kh} - f, \chi_{kh}))_m\} \\ & + \sum_{m=1}^{M} \left\{ (a(u_{kh})(\chi_{kh}))_m - \left( (q, \chi_{kh})_{\Gamma_N} \right)_m \right\} \\ & + (u_{kh}(0) - u_s, \chi_{kh}(0)) + (v_{kh}(0) - v_s, \psi_{kh}(0)). \end{aligned}$$

Here, $(\cdot)_m$ denotes the time integral $\int_{I_m} \cdot \, dt$.

Remark 1.2.2. The discretisation scheme specified in Definition 1.2.1 is called cG(1)cG(1) method. This name expresses that a continuous Galerkin (cG) method with first order basis functions in the displacement and in the velocity is used. The cG(1)cG(1) method is a nonconforming Petrov-Galerkin scheme, since trial and test space are different. The trial space $V_{kh}$ is a subset of $V$ but the test space $W_{kh}$ is not, since it contains discontinuous functions.

**1.2.2. Solution of the Discrete Problem.** In this section, we derive the time stepping scheme, which results from (1.2.1). In the first step, we use the test functions $(\psi_{kh}, 0)$ and $(0, \chi_{kh})$ in (1.2.1) and end up with the following system of equations

$$\begin{aligned}((\dot{u}_{kh} - v_{kh}, \psi_{kh})) &= -(u_{kh}(0) - u_s, \psi_{kh}(0)) &(1.2.2)\\ ((\dot{v}_{kh}, \chi_{kh})) &= ((f, \chi_{kh})) + \Big((q, \chi_{kh})_{\Gamma_N}\Big) &(1.2.3)\\ &\quad - (a(u_{kh})(\chi_{kh}))\\ &\quad - (v_{kh}(0) - v_s, \chi_{kh}(0)),\end{aligned}$$

which has to hold for all $\psi_{kh}, \chi_{kh} \in W_{kh}$. We set $\varphi_{kh}^m := \varphi_{kh}(t_m)$. One possibility to choose the piecewise constant temporal basis functions of $W_{kh}$ is

$$\phi_0(t) := \begin{cases} 1 & \text{for } t = 0 \\ 0 & \text{else} \end{cases}$$

$$\phi_m(t) := \begin{cases} 1 & \text{for } t \in I_m \\ 0 & \text{else} \end{cases}$$

with $1 \leq m \leq M$. We use $\phi_m \psi_h^m$ and $\phi_m \chi_h^m$ with $\psi_h^m$, $\chi_h^m \in V_h^m$ for $m = 0, 1, \ldots M$ successively as test functions in the equations (1.2.2) and (1.2.3) and obtain the time stepping scheme:

Time Stepping Scheme 1.2.3. Find $w_{kh} = (u_{kh}, v_{kh}) \in V_{kh} \times V_{kh}$, where $w_{kh}^0 \in V_h^0 \times V_h^0$ is given by

$$\begin{aligned}\forall \psi_h \in V_h^0: \quad \left(u_{kh}^0 - u_s, \psi_h\right) &= 0,\\ \forall \chi_h \in V_h^0: \quad \left(v_{kh}^0 - v_s, \chi_h\right) &= 0.\end{aligned}$$

For $m = 1, 2, \ldots, M$, $w_{kh}^m \in V_h^m \times V_h^m$ is the solution of the system

$$\left(u_{kh}^m - u_{kh}^{m-1}, \psi_h\right) = \frac{1}{2} k_m \left(v_{kh}^m + v_{kh}^{m-1}, \psi_h\right) \qquad (1.2.4)$$

$$\begin{aligned}\left(v_{kh}^m - v_{kh}^{m-1}, \chi_h\right) &= -\frac{1}{2}k_m\left[a\left(u_{kh}^m\right)(\chi_h) + a\left(u_{kh}^{m-1}\right)(\chi_h)\right] \quad (1.2.5)\\ &+\frac{1}{2}k_m\left(f^m + f^{m-1}, \chi_h\right)\\ &+\frac{1}{2}k_m\left(q^m + q^{m-1}, \chi_h\right)_{\Gamma_N},\end{aligned}$$

which has to hold for all $\psi_h, \chi_h \in V_h^m$.

We evaluate the time integrals over terms only including the test and trial functions exactly. The time integrals $((f, \chi_{kh}))_m$ and $\Big((q, \chi_{kh})_{\Gamma_N}\Big)_m$ are approximated by the trapezoidal rule, see Section A.1. This ensures the optimal order of convergence for the presented scheme. However, one may obtain a more accurate solution by choosing a quadrature rule of higher order like the Simpson or the two point Gauß rule. The additional error introduced by the numerical quadrature is discussed in [**80, 81**]. There, continuous Galerkin methods in the context of ODEs are considered.

The Time Stepping Scheme 1.2.3 corresponds to the Crank-Nicolson method (see, e.g., [**60**]). By rewriting this scheme for the second order problem, we recover the Newmark method (c.f. [**60**]) with $\gamma = 2\beta = \frac{1}{2}$.

The system (1.2.4-1.2.5) can be decoupled by inserting (1.2.4) into (1.2.5). We obtain the system

$$\begin{aligned}\left(v_{kh}^m, \psi_h\right) &= \frac{2}{k_m}\left(u_{kh}^m - u_{kh}^{m-1}, \psi_h\right) - \left(v_{kh}^{m-1}, \psi_h\right) \quad (1.2.6)\\ \left(u_{kh}^m, \chi_h\right) &= \frac{1}{4}k_m^2\left(f^m + f^{m-1}, \chi_h\right) + \left(u_{kh}^{m-1}, \chi_h\right) \quad (1.2.7)\\ &-\frac{1}{4}k_m^2 a\left(u_{kh}^m\right)(\chi_h) + k_m\left(v_{kh}^{m-1}, \chi_h\right)\\ &+\frac{1}{4}k_m^2\left(q^m + q^{m-1}, \chi_h\right)_{\Gamma_N}\\ &-\frac{1}{4}k_m^2 a\left(u_{kh}^{m-1}\right)(\chi_h)\end{aligned}$$

for all $\psi_h, \chi_h \in V_h^m$.

We first solve (1.2.7) and next calculate the solution of (1.2.6). The numerical effort for solving equation (1.2.6) is small, since it is a simple

$L^2$-projection. If the finite element spaces $V_h^m$ do not vary, equation (1.2.6) reduces to a simple linear combination of vectors in matrix-vector notation. The solution of (1.2.7) is more involved, since it is a nonlinear equation. We use a damped Newton scheme (see, e.g., [**57**]) to solve (1.2.7). We drop here the index $kh$ to simplify the notation. As initial value $u_0^m$ for Newton's method, we use the $L^2$-projection on the mesh $\mathbb{T}_h^m$ of the solution of the previous time step $u^{m-1}$. The update $d_j \in V_h^m$ in the $j^{\text{th}}$ Newton iteration is given by

$$\left(d_j, \chi_h\right) + \frac{1}{4} k_m^2 a' \left(u_j^m\right) \left(d_j, \chi_h\right) \quad = \quad R\left(u_j^m\right)\left(\chi_h\right) \tag{1.2.8}$$

for all $\chi_h \in V_h^m$ with the residual

$$\begin{aligned} R\left(u_j^m\right)\left(\chi_h\right) \quad := \quad & \left(u_j^m, \chi_h\right) + \frac{1}{4} k_m^2 a\left(u_j^m\right)\left(\chi_h\right) \\ & - \frac{1}{4} k_m^2 \left(f^m + f^{m-1}, \chi_h\right) - \left(u_{kh}^{m-1}, \chi_h\right) \\ & - \frac{1}{4} k_m^2 \left(q^m + q^{m-1}, \chi_h\right)_{\Gamma_N} \\ & + \frac{1}{4} k_m^2 a\left(u_{kh}^{m-1}\right)\left(\chi_h\right) - k_m \left(v_{kh}^{m-1}, \chi_h\right). \end{aligned}$$

The existence and uniqueness of the update $d_j$ is ensured by Assumption 1.1.11. In the classic Newton scheme, $u_{j+1}^m$ is obtained by $u_{j+1}^m = u_j^m - d_j$. To improve the stability of Newton's method, we use the following simple damping scheme, which may lead to a convergent iteration, even if the classic scheme fails. We determine the first integer $i \geq 0$ for which strict monotonicity holds, i.e.

$$\forall \chi_h \in V_h^m : \quad R\left(u_j^m - 2^{-i} d_j\right)\left(\chi_h\right) < R\left(u_j^m\right)\left(\chi_h\right).$$

The update is then given by $u_{j+1}^m = u_j^m - 2^{-i} d_j$. More sophisticated globalisations of Newton's method are described in [**44**]. If $\|R\| \leq \text{tol}_1$ or $\|d_j\| \leq \text{tol}_2$ with appropriate constants $\text{tol}_1 > 0$ and $\text{tol}_2 > 0$, $u_{j+1}^m$ is accepted as a solution of (1.2.7). We typically need only one or two classic Newton steps to solve the nonlinear equation, because the solution of the previous time step is a good initial value in the Newton iteration.

The update equation of Newton's method (1.2.8) is solved by a direct solver, namely UMFPACK, see [**43**]. We use a CG method to determine the solution of the $L^2$-projection in equation (1.2.6), where the linear system is preconditioned by an SSOR scheme.

The technical details of the solution procedure are described in Section 2.5. In particular, the evaluation of the terms $\left(u_{kh}^{m-1}, \psi\right)$, $\left(v_{kh}^{m-1}, \psi\right)$ and $a\left(u_{kh}^{m-1}\right)(\chi_h)$ is discussed there, which has to be done very carefully, if the meshes $\mathbb{T}_h^m$ and $\mathbb{T}_h^{m-1}$ are different.

**1.2.3. A Priori Analysis.** The analysis of the continuous Galerkin method in the context of nonlinear hyperbolic equations of second order is in the early stages. We summarise the results in this section. In the linear case, more results are available. In [**47**], a continuous Galerkin method similar to the one presented here is analysed in the linear case. We discuss the same results, namely: The existence of a discrete solution, a priori error estimation and energy conservation.

First, we present results concerning the existence of a discrete solution. For the quasilinear case, no results are available at the best of the author's knowledge. If $\mathcal{A}$ is a semilinear operator, where the nonlinear part is Lipschitz continuous in $u$, we have the following result:

THEOREM 1.2.4. *For a sufficiently small $k_m$, there exists a unique discrete solution $w_{kh}$.*

PROOF OF THEOREM 1.2.4. Theorem 2.1 in [**69**]. □

The operator $\pi_h^m$ denotes the projection of the function $w_{kh}^{m-1}$ onto the space $V_h^m \times V_h^m$. The a priori error estimate in the semilinear case is given in the following theorem:

THEOREM 1.2.5. *We have the a priori error estimate*

$$\max_{t\in I}\left\{\|u(t)-u_{kh}(t)\|_0+\|v(t)-v_{kh}(t)\|_0\right\}$$
$$\leq \quad C\left\{\max_{1\leq m\leq M} k_m^2 C_t(u)+\max_{1\leq m\leq M} h_m^2 C_x(u)+\sqrt{\mathcal{N}_C}\max_{1\leq m\leq M}\|J^m\|_1\right\}$$

*and*

$$\begin{aligned} & \max_{t\in I} \|u(t) - u_{kh}(t)\|_\infty \\ \leq \; & CL_h \left\{ \max_{1\leq m\leq M} k_m^2 C_t(u) + \max_{1\leq m\leq M} h_m^2 C_x(u) + \sqrt{\mathcal{N}_C} \max_{1\leq m\leq M} \|J^m\|_1 \right\}. \end{aligned}$$

*Here, $C > 0$ is a constant independent of $h$, $k$ and $w$, $L_h$ is a factor, which grows logarithmically in $h$. The number of mesh changes during the calculation is given by $\mathcal{N}_C$ and $J^m := w_{kh}^{m-1} - \pi_h^m w_{kh}^{m-1}$ expresses the differences between the solution due to the mesh change. The constants $C_t$ and $C_x$ only depend on $u$.*

Proof of Theorem 1.2.5. Theorem 3.1 in [**69**]. □

Remark 1.2.6. In [**49**], superconvergence results in the temporal grid points $t_m$ are derived, which hold for linear $\mathcal{A}$. In the time points $t_m$, the convergence rate is of order $k^{2q}$, where $q$ is the polynomial degree of the temporal basis functions. Since linear polynomials are used, we obtain a convergence rate of $k^2$ in $t_m$, which corresponds to the convergence rate on $I$. Consequently, no superconvergence properties are observed for the cG(1)cG(1) scheme.

In Proposition 1.1.12, we see that the continuous solution conserves the total energy. We expect from the discrete solution the conservation of energy, too. The cG(1)cG(1) method is mainly used, since the total energy is constant for the discrete solution. See Proposition 1.2.7 below. A detailed discussion of energy conservation is found, for instance, in [**47**]. The operator $\mathcal{A}$ is assumed to be linear. The potential or inner energy in time step $m$ is defined as $E_{in}^m := \frac{1}{2} a\left(u_{kh}^m, u_{kh}^m\right)$, the kinetic energy as $E_{kin}^m := \frac{1}{2}\left(v_{kh}^m, v_{kh}^m\right)$ and the total energy as $E_{tot}^m := E_{kin}^m + E_{in}^m$. As in the continuous setting, we have the following result concerning energy conservation:

Proposition 1.2.7. *If $f$ and $q$ are zero and if $V_h^{m-1} \subseteq V_h^m$ holds for all $m \in \{0, 1, \dots, M\}$, then the total energy is constant.*

Proof of Proposition 1.2.7. Let us assume for a moment that $V_h^{m-1} = V_h^m$ holds. Then, we are allowed to test equation (1.2.1) on a subinterval $I_m$ with $(-\dot{v}_{kh}, \dot{u}_{kh})$ and obtain

$$\begin{aligned} 0 &= ((v_{kh}, \dot{v}_{kh}))_{I_m} + (a(u_{kh}, \dot{u}_{kh}))_{I_m} \\ &= \frac{1}{2}(v_{kh}^m, v_{kh}^m) - \frac{1}{2}\left(v_{kh}^{m-1}, v_{kh}^{m-1}\right) + \frac{1}{2}a(u_{kh}^m, u_{kh}^m) - \frac{1}{2}a\left(u_{kh}^{m-1}, u_{kh}^{m-1}\right) \\ &= E_{kin}^m - E_{kin}^{m-1} + E_{in}^m - E_{in}^{m-1}. \end{aligned}$$

Consequently, it holds

$$E_{tot}^m = E_{tot}^{m-1}.$$

If $V_h^{m-1} \neq V_h^m$ holds, we have to project $u_{kh}^{m-1}$ and $v_{kh}^{m-1}$ by the operator $\pi_h^m$ onto the space $V_h^m$. Since $\pi_h^m \varphi_{kh}$ is equal to $\varphi_{kh}$ for functions $\varphi_{kh} \in V_{kh}^{m-1}$ under the assumption $V_h^{m-1} \subseteq V_h^m$, the calculation above stays valid. □

Remark 1.2.8. By a careful analysis of the arguments above one sees that the difference in the energy is closely connected to the projection error $\left\|u_{kh}^{m-1} - \pi_h^m u_{kh}^{m-1}\right\|$ in the $m^{\text{th}}$ time step.

## 1.3. A Posteriori Error Estimation

This section is devoted to the derivation of the basic error idendity. It involves a so called dual problem, whose analytical properties are discussed. Furthermore, an adequate time stepping scheme for its numerical solution is derived and analysed. Finally, we discuss the approximate numerical evaluation of the error identity. The derivation is based on the DWR technique, see, for instance, [**13, 17**].

**1.3.1. General Result.** The aim is to derive an a posteriori error estimate for the discretisation error in a more or less arbitrary functional, which represents the quantity of interest. We consider functionals of the type

$$J(w) := \int_0^T J_1(w)\,dt + J_2(w(T)), \tag{1.3.1}$$

where $J_1,\ J_2 \in \left(H^1\left(\Omega,\Gamma_D\right)\times H^1\left(\Omega,\Gamma_D\right)\right)^\star$ are three times continuously Fréchet differentiable. The form of $J$ specified in (1.3.1) considers two typical situations: One is interested in the mean value of a quantity over $I$ or in the value at the end point.

The derivation of the error estimate is based on optimisation arguments. To embed the error estimation in the optimisation context, we define the Lagrangian

$$\mathcal{L}(\omega,\vartheta) := J(\omega) - A(\omega)(\vartheta)$$

for $\omega \in U \times V$ and $\vartheta \in V \times U$. We call $\mathcal{L}$ the continuous Lagrangian. The discrete analogon is given by

$$\mathcal{L}_{kh}\left(\omega,\vartheta\right) := J\left(\omega\right) - A_{kh}\left(\omega\right)\left(\vartheta\right)$$

for $\omega \in U \times V$ and $\vartheta \in (W_{kh}\times W_{kh}) \cup (V \times U)$. The connection between the Lagrangians and the quasilinear hyperbolic problem of second order becomes apparent as soon as we consider the Fréchet derivative of $\mathcal{L}$ w.r.t. $z$:

$$\mathcal{L}'_z\left(w,z\right)\left(\delta w,\delta z\right) = -A(w)(\delta z). \tag{1.3.2}$$

It corresponds to the weak formulation of the quasilinear hyperbolic equation of second order. In the discrete case, we recover the space-time Galerkin approximation. The Fréchet derivative w.r.t. $w$ is

$$\mathcal{L}'_w(w,z)(\delta w,\delta z) = J'(w)(\delta w) - A'(w)(\delta w, z). \tag{1.3.3}$$

The right hand side of (1.3.3) and its discrete analogon are discussed in detail in Section 1.3.2.

The central assumption for the a posteriori error estimation is

ASSUMPTION 1.3.1. *The point $(w,z) \in (U\times V)\times(V\times U)$ is a stationary point of $\mathcal{L}$, i.e.*

$$\mathcal{L}'(w,z)(\delta w,\delta z) = 0 \tag{1.3.4}$$

*for all $(\delta w,\delta z) \in (U\times V)\times(V\times U)$. Analogously, $(w_{kh},z_{kh}) \in (V_{kh}\times V_{kh})\times(W_{kh}\times W_{kh})$ is a stationary point of $\mathcal{L}_{kh}$, i.e.*

$$\mathcal{L}'_{kh}\left(w_{kh},z_{kh}\right)\left(\delta w_{kh},\delta z_{kh}\right) = 0 \tag{1.3.5}$$

*for all* $(\delta w_{kh}, \delta z_{kh}) \in (V_{kh} \times V_{kh}) \times (W_{kh} \times W_{kh})$.

Assumption 1.3.1 together with (1.3.2) says that $w$ is a weak solution of the quasilinear hyperbolic equation of second order. Equation (1.3.5) ensures that $w_{kh}$ is a discrete solution. Furthermore, the equation (1.3.4) claims that $z$ is a solution of the equation

$$\forall \delta w \in U \times V : \quad J'(w)(\delta w) - A'(w)(\delta w, z) = 0. \tag{1.3.6}$$

According to equation (1.3.3), $z_{kh}$ fulfils

$$\forall \delta w_{kh} \in U_{kh} \times U_{kh} : \quad J'\left(w_{kh}\right)\left(\delta w_{kh}\right) - A'\left(w_{kh}\right)\left(\delta w_{kh}, z_{kh}\right) = 0. \tag{1.3.7}$$

We will discuss the problems (1.3.6) and (1.3.7) after the presentation of the basic result for the a posteriori error estimate:

Proposition 1.3.2. *Under the Assumption 1.3.1, the error identity*

$$\begin{aligned} & J\left(w\right) - J\left(w_{kh}\right) \\ =& \frac{1}{2} \mathcal{L}'_{kh}\left(w_{kh}, z_{kh}\right)\left(w - \tilde{w}_{kh}, z - \tilde{z}_{kh}\right) + \mathcal{R}_{kh} \\ =& \frac{1}{2}\left[\rho\left(w_{kh}\right)\left(z - \tilde{z}_{kh}\right) + \rho^{\star}\left(w_{kh}, z_{kh}\right)\left(w - \tilde{w}_{kh}\right)\right] + \mathcal{R}_{kh} \end{aligned} \tag{1.3.8}$$

*holds, where the functions* $\tilde{z}_{kh} \in W_{kh} \times W_{kh}$ *and* $\tilde{w}_{kh} \in V_{kh} \times V_{kh}$ *are arbitrary. The remainder term*

$$\mathcal{R}_{kh} := \frac{1}{2} \int_0^1 \mathcal{L}'''\left(w_{kh} + se\right)\left(e, e, e\right) s\left(s - 1\right)\, ds$$

*is of third order in the error* $e = w - w_{kh}$. *The primal residual* $\rho$ *is given by*

$$\rho\left(w\right)\left(\varphi\right) := \mathcal{L}'_{kh,z}\left(w\right)\left(\varphi\right) = -A_{kh}\left(w\right)\left(\varphi\right)$$

*and the dual residual* $\rho^{\star}$ *by*

$$\rho^{\star}\left(w, z\right)\left(\varphi\right) := \mathcal{L}'_{kh,w}\left(w, z\right)\left(\varphi\right) = J'\left(w\right)\left(\varphi\right) - A'_{kh}\left(w\right)\left(\varphi, z\right).$$

The error indentity (1.3.8) represents the discretisation error w.r.t. the functional $J$. It is based on the discrete primal residual $\rho\left(w_{kh}\right)$ and the discrete dual residual $\rho^{\star}\left(w_{kh}, z_{kh}\right)$. The residuals can be numerically evaluated. They are weighted with the dual approximation error $z - \tilde{z}_{kh}$ and with the primal approximation error $w - \tilde{w}_{kh}$, respectively. The approximation error cannot be evaluated exactly. In Section 1.3.3, we introduce a numerical evaluation method for the approximation error. Additionally, we do not know the value of $\mathcal{R}_{kh}$. However, it is of higher order in the error and can be neglected.

To prove Proposition 1.3.2, we need the following lemma, which goes back to Meidner [**85**] and is a generalisation of the classic result of Becker and Rannacher [**17**].

Lemma 1.3.3. *Suppose $L$ is a three times continuously Fréchet differentiable functional on the function space $Y$. Furthermore, let $y_1$ be a stationary point of $L$ on a subspace $Y_1 \subseteq Y$, i.e.*

$$\forall \delta y_1 \in Y_1 : \quad L'(y_1)(\delta y_1) = 0. \tag{1.3.9}$$

*The corresponding Galerkin solution $y_2 \in Y_2 \subseteq Y$ is defined by*

$$\forall \delta y_2 \in Y_2 : \quad L'(y_2)(\delta y_2) = 0. \tag{1.3.10}$$

*We assume*

$$L'(y_1)(y_2) = 0. \tag{1.3.11}$$

*Under these assumptions, we obtain for arbitrary $\tilde{y}_2 \in Y_2$ the error representation*

$$L(y_1) - L(y_2) = \frac{1}{2} L'(y_2)(y_1 - \tilde{y}_2) + \mathcal{R}.$$

*The remainder term $\mathcal{R}$ is*

$$\mathcal{R} = \frac{1}{2} \int_0^1 L'''(y_2 + se)(e, e, e) \cdot s \cdot (s - 1)\, ds$$

*with $e := y_1 - y_2$.*

Remark 1.3.4. In the classic result, $Y = Y_1$ is assumed. Then assumption (1.3.11) becomes trivial. Since we consider a nonconforming Petrov-Galerkin scheme, where $Y_2 \subseteq Y_1$ does not hold, the generalised form is needed.

We will use the ideas of the proof in Section 3.3 and 4.3 again. Hence we give the proof here.

Proof of Lemma 1.3.3. The main theorem of calculus leads to

$$L(y_1) - L(y_2) = \int_0^1 L'(y_2 + se)(e)\, ds.$$

We approximately evaluate this integral by the trapezoidal rule

$$\int_0^1 f(s)\, ds = \frac{1}{2} f(0) + \frac{1}{2} f(1) + \frac{1}{2}\int_0^1 f''(s) \cdot s \cdot (s-1)\, ds$$

and obtain

$$L(y_1) - L(y_2) = \frac{1}{2} L'(y_2)(e) + \frac{1}{2} L'(y_1)(e) + \mathcal{R}.$$

The assertions (1.3.9) and (1.3.11) yield $L'(y_1)(e) = 0$. Because of (1.3.10), we can substitute $L'(y_2)(e)$ by $L'(y_2)(y_1 - \tilde{y}_2)$ for any $\tilde{y}_2 \in Y_2$. □

After these preparations, we prove Proposition 1.3.2:

Proof of Proposition 1.3.2. We want to apply Lemma 1.3.3. For this purpose, we set

$$\begin{aligned} Y_1 &:= (U \times V) \times (V \times U), \\ Y_2 &:= (V_{kh} \times V_{kh}) \times (W_{kh} \times W_{kh}), \\ Y &:= (U \times V) \times ([V \cup W_{kh}] \times [U \cup W_{kh}]), \end{aligned}$$

and $L := \mathcal{L}_{kh}$. The assertions (1.3.4) and (1.3.5) of Assumption 1.3.1 ensure (1.3.9) and (1.3.10), respectively. Furthermore, we have $J(w) = \mathcal{L}_{kh}(w, z)$ and $J(w_{kh}) = \mathcal{L}_{kh}(w_{kh}, z_{kh})$, since

$$A(w)(z) = A_{kh}(w)(z) = A_{kh}(w_{kh})(z_{kh}) = 0$$

holds. It remains to verify (1.3.11). Since $V_{kh} \subseteq U$ and $V_{kh} \subseteq V$, we directly obtain

$$\mathcal{L}'_{kh,w}(w, z)(w_{kh}, z_{kh}) = J'(w)(w_{kh}) - A'(w)(w_{kh}, z) = 0.$$

The proof for the derivative w.r.t. $z$,

$$\mathcal{L}'_{kh,z}\left(w,z\right)\left(w_{kh},z_{kh}\right)=-A_{kh}\left(w\right)\left(z_{kh}\right)=-A\left(w\right)\left(z_{kh}\right),$$

is more involved, since $W_{kh}\nsubseteq U$ and $W_{kh}\nsubseteq V$. The function spaces $U$ and $V$ are dense in $L^2\left(I;H^1\left(\Omega,\Gamma_D\right)\right)$ and in $L^2\left(I;L^2\left(\Omega\right)\right)$ w.r.t. the $L^2\left(I;H^1\left(\Omega,\Gamma_D\right)\right)$- and the $L^2\left(I;L^2\left(\Omega\right)\right)$-norm, respectively. Furthermore, no time derivatives on the test function $\varphi$ are contained in $A\left(w\right)\left(\varphi\right)$. Consequently, $A\left(w\right)\left(\varphi\right)=0$ holds for all

$$\varphi=\left(\psi,\chi\right)\in L^2\left(I;L^2\left(\Omega\right)\right)\times L^2\left(I;H^1\left(\Omega,\Gamma_D\right)\right).$$

Since $W_{kh}\subseteq L^2\left(I;H^1\left(\Omega,\Gamma_D\right)\right)$ and $W_{kh}\subseteq L^2\left(I;L^2\left(\Omega\right)\right)$, we conclude $A\left(w\right)\left(z_{kh}\right)=0$ and with it (1.3.11) holds. Finally, we apply Lemma 1.3.3, from which the claim of Proposition 1.3.2 follows. □

**1.3.2. The Continuous and the Discrete Dual Problem.** The error identity (1.3.8) involves the quantities $z$ and $z_{kh}$, which are defined by the stationarity conditions (1.3.4) and (1.3.5) as solutions of the variational problems (1.3.6) and (1.3.7), respectively. These problems are called the continuous and the discrete dual problem. In this context, the original problems specified in the Definitions 1.1.5 and 1.2.1 are referred to as primal problems. We begin with the discussion of the continuous dual problem.

1.3.2.1. *The Continuous Dual Problem.* In view of (1.3.3), we define the continuous dual problem as follows:

DEFINITION 1.3.5. We say, $z=\left(\bar{u},\bar{v}\right)\in V\times U$ is a weak solution of the continuous dual problem, if

$$J'(w)(\varphi)=A'(w)(\varphi,z) \tag{1.3.12}$$

holds for all $\varphi=\left(\psi,\chi\right)\in V\times U$. Here, $w$ is the weak solution of the primal problem.

Let us now take a closer look at the continuous dual problem. The Fréchet derivative of $A$ w.r.t. $w$ is given by

$$\begin{aligned} A'(w)(\varphi, z) &= \left(\left(\dot{\psi} - \chi, \bar{u}\right)\right) + ((\dot{\chi}, \bar{v})) + \left(a'(u)\,(\psi, \bar{v})\right) \\ &\quad + (\psi(0), \bar{u}(0)) + (\chi(0), \bar{v}(0))\,. \end{aligned}$$

We use integration by parts to shift the time derivative from the test functions to the solution variables and obtain

$$\begin{aligned} A'(w)(\varphi, z) &= -\,((\dot{\bar{v}} + \bar{u}, \chi)) - ((\dot{\bar{u}}, \psi)) + \left(a'(u)\,(\psi, \bar{v})\right) \\ &\quad + (\psi(T), \bar{u}(T)) + (\chi(T), \bar{v}(T))\,. \end{aligned}$$

The Fréchet derivative of $J$ is

$$\begin{aligned} J'(w)(\varphi) &= J'_u(w)(\psi) + J'_v(w)(\chi) \\ &= \int_0^T J'_{1,u}(w)(\psi)\,dt + J'_{2,u}(w(T))(\psi(T)) \\ &\quad + \int_0^T J'_{1,v}(w)(\chi)\,dt + J'_{2,v}(w(T))(\chi(T)). \end{aligned}$$

Testing equation (1.3.12) with $\varphi_1 = (\psi, 0)$ and $\varphi_2 = (0, \chi)$ for arbitrary $\psi \in V$ and $\chi \in U$, we obtain the system

$$\begin{aligned} -\,((\dot{\bar{u}}, \psi)) + \left(a'(u)\,(\psi, \bar{v})\right) + (\bar{u}(T), \psi(T)) &= J'_u(w)(\psi) \\ -\,((\dot{\bar{v}} + \bar{u}, \chi)) + (\bar{v}(T), \chi(T)) &= J'_v(w)(\chi). \end{aligned}$$

Writing the first order system as a second order problem, we deduce that the strong solution of the continuous dual problem is characterised by the following relations:

Definition 1.3.6. The function $\bar{v} \in C^2(\Omega \times I)$ is a strong solution of the continuous dual problem, if

$$\begin{aligned} \ddot{\bar{v}} - \mathcal{A}'(w)\,(\bar{v}) &= J'_{1,u}(w) - \tfrac{\partial}{\partial t} J'_{1,v}(w) && \text{in } \Omega \times I \\ \bar{v} &= 0 && \text{on } \Gamma_D \times I \\ \tfrac{\partial \bar{v}}{\partial \nu} &= 0 && \text{on } \Gamma_N \times I \\ \bar{v}(T) &= J'_{2,v}(w(T)) && \text{in } \Omega \\ \dot{\bar{v}}(T) &= J'_{2,u}(w(T)) && \text{in } \Omega \end{aligned}$$

holds. Here, $w$ is the strong solution of the primal problem.

The strong problem formulation allows for more insight into the structure of the dual problem. The first important observation is that the dual problem starts at $T$ and runs backward in time to 0. Consequently, the initial values are specified for $T$. Furthermore, the dual problem is a linear hyperbolic equation of second order. By the substitution $\tilde{t} = T - t$, it can be transformed into a forward problem. Assumption 1.1.11 ensures that the theory mentioned in Section 1.1 is applicable. Thus, we obtain a well-posed problem. The homogeneous Dirichlet boundary conditions of the primal problem are transferred to the dual problem. The nonhomogeneous Neumann boundary conditions are transformed into homogeneous ones. The descriptive interpretation of the dual solution is that it represents the influence of a certain space-time point $(x, t)$ onto the error measured in the functional $J$.

1.3.2.2. *The Discrete Dual Problem.* Let us now take a closer look at the discrete dual problem, it is specified in the following definition:

Definition 1.3.7. The function $z_{kh} = (\bar{u}_{kh}, \bar{v}_{kh}) \in W_{kh} \times W_{kh}$ is a solution of the discrete dual problem, if

$$J'(w_{kh})(\varphi_{kh}) = A'_{kh}(w_{kh})(\varphi_{kh}, z_{kh}) \tag{1.3.13}$$

holds for all $\varphi_{kh} = (\psi_{kh}, \chi_{kh}) \in V_{kh} \times V_{kh}$. Here, $w_{kh}$ is the discrete solution of the quasilinear hyperbolic equation of second order specified in Definition 1.2.1.

We observe that the discrete solution $z_{kh}$ is contained in the test space $W_{kh} \times W_{kh}$ of the primal problem, i.e. $z_{kh}$ is a piecewise constant function in time. The interpolation by piecewise constant functions is of first order. Thus, the approximation is globally of maximum order $k$. The a priori error estimate is discussed later on in more detail. The time stepping scheme resulting from (1.3.13) is presented in Time Stepping Scheme 1.3.8. It differs from the primal Time Stepping Scheme 1.2.3. It should be remarked that we are not interested in an accurate numerical solution of the dual problem but in an accurate a posteriori error estimation, which is evaluable efficiently. For this purpose, it is advantageous to use the Time Stepping Scheme 1.3.8 to calculate an approximation of $z$.

For the derivation of the dual time stepping scheme from (1.3.13), we rewrite the discrete Fréchet derivative,

$$\begin{aligned} & A'_{kh}\left(w_{kh}\right)\left(\varphi_{kh}, z_{kh}\right) \\ = \; & \sum_{m=1}^{M} \left\{ \left(\left(\dot{\psi}_{kh} - \chi_{kh}, \bar{u}_{kh}\right)\right)_m + \left(a'\left(u_{kh}\right)\left(\psi_{kh}, \bar{v}_{kh}\right)\right)_m \right\} \\ & + \sum_{m=1}^{M} \left(\left(\dot{\chi}_{kh}, \bar{v}_{kh}\right)\right)_m + \left(\psi_{kh}(0), \bar{u}_{kh}(0)\right) + \left(\chi_{kh}(0), \bar{v}_{kh}(0)\right), \end{aligned}$$

by means of integration by parts as

$$\begin{aligned} & A'_{kh}\left(w_{kh}\right)\left(\varphi_{kh}, z_{kh}\right) \\ = \; & -\sum_{m=1}^{M} \left\{ \left(\left(\bar{u}_{kh} + \dot{\bar{v}}_{kh}, \chi_{kh}\right)\right)_m - \left(a'\left(u_{kh}\right)\left(\psi_{kh}, \bar{v}_{kh}\right)\right)_m \right\} \\ & -\sum_{m=1}^{M} \left\{ \left(\left(\left(\dot{\bar{u}}_{kh}, \psi_{kh}\right)\right)_m + \left[\bar{u}_{kh}\right]_m, \psi_{kh}\left(t_m\right)\right) + \left(\left[\bar{v}_{kh}\right]_m, \chi_{kh}\left(t_m\right)\right) \right\} \\ & + \left(\bar{u}^-_{kh}\left(T\right), \psi_{kh}(T)\right) + \left(\bar{v}^-_{kh}\left(T\right), \chi_{kh}(T)\right). \end{aligned}$$

Here, the jump of a possibly discontinuous function $\omega$ at a time instance $t_m$ is defined by

$$[\omega]_m := \lim_{t \downarrow t_m} \omega(t) - \lim_{t \uparrow t_m} \omega(t).$$

Since $z_{kh}$ is piecewise constant in time, the temporal derivative is zero. Thus, $A'_{kh}$ simplifies to

$$\begin{aligned} A'_{kh}\left(w_{kh}\right)\left(\varphi_{kh}, z_{kh}\right) \; = \; & \sum_{m=1}^{M} \left\{ \left(a'\left(u_{kh}\right)\left(\psi_{kh}, \bar{v}_{kh}\right)\right)_m - \left(\left(\bar{u}_{kh}, \chi_{kh}\right)\right)_m \right\} \\ & -\sum_{m=1}^{M} \left\{ \left(\left[\bar{u}_{kh}\right]_m, \psi_{kh}\left(t_m\right)\right) + \left(\left[\bar{v}_{kh}\right]_m, \chi_{kh}\left(t_m\right)\right) \right\} \\ & + \left(\bar{u}^-_{kh}\left(T\right), \psi_{kh}(T)\right) + \left(\bar{v}^-_{kh}\left(T\right), \chi_{kh}(T)\right). \end{aligned}$$

Testing equation (1.3.13) by $\varphi_1 = (\psi_{kh}, 0)$ and $\varphi_2 = (0, \chi_{kh})$ and using the simplified form of $A'_{kh}$, we obtain the system

$$\begin{aligned} & J'_u(w_{kh})(\psi_{kh}) \\ = \; & \sum_{m=1}^{M} \left\{ \left( a'(u_{kh})(\psi_{kh}, \bar{v}_{kh}) \right)_m - \left( [\bar{u}_{kh}]_m, \psi_{kh}(t_m) \right) \right\} \\ & + \left( \bar{u}_{kh}^{-}(T), \psi_{kh}(T) \right) \end{aligned}$$

and

$$\begin{aligned} & J'_v(w_{kh}) \\ = \; & -\sum_{m=1}^{M} \left\{ ((\bar{u}_{kh}, \chi_{kh}))_m + \left( [\bar{v}_{kh}]_m, \chi_{kh}(t_m) \right) \right\} \\ & + \left( \bar{v}_{kh}^{-}(T), \chi_{kh}(T) \right). \end{aligned}$$

The aim is to derive a time stepping scheme analogous to Section 1.2.2. Since the test space contains continuous functions, we cannot use the functions $\phi_m$ defined in Section 1.2.2. Instead, we specify the temporal test functions by

$$\bar{\phi}_M(t) := \begin{cases} \frac{t - t_{M-1}}{k_M} & \text{for } t \in I_M \\ 0 & \text{else,} \end{cases}$$

$$\bar{\phi}_m(t) := \begin{cases} \frac{t_{m+1} - t}{k_{m+1}} & \text{for } t \in I_{m+1} \\ \frac{t - t_{m-1}}{k_m} & \text{for } t \in I_m \\ 0 & \text{else,} \end{cases}$$

for $m = M-1, M-2, \ldots, 1$ and

$$\bar{\phi}_0(t) := \begin{cases} \frac{t_1 - t}{k_1} & \text{for } t \in I_1 \\ 0 & \text{else.} \end{cases}$$

We successively use the test functions $\phi_m \psi_h^m$ and $\phi_m \chi_h^m$ with $\psi_h^m \in V_h^m$ and $\chi_h^m \in V_h^m$ for $m = M, M-1, \ldots, 0$ and obtain the time stepping scheme:

Time Stepping Scheme 1.3.8. Find $z_{kh} = (\bar{u}_{kh}, \bar{v}_{kh}) \in W_{kh} \times W_{kh}$, where $z_{kh}^M = \left(\bar{u}_{kh}^M, \bar{v}_{kh}^M\right) \in V_h^M \times V_h^M$ is the solution of

$$\begin{aligned} \left(\bar{u}_{kh}^M, \psi_h\right) &= -\frac{1}{2} k_M a'\left(u_{kh}^M\right)\left(\psi_h, \bar{v}_{kh}^M\right) && (1.3.14)\\ &\quad \frac{1}{2} k_M J'_{1,u}\left(w_{kh}^M\right)(\psi_h) \\ &\quad + J'_{2,u}\left(w_{kh}^M\right)(\psi_h) \\ \left(\bar{v}_{kh}^M, \chi_h\right) - \frac{1}{2} k_M\left(\bar{u}_{kh}^M, \chi_h\right) &= \frac{1}{2} k_M J'_{1,v}\left(w_{kh}^M\right)(\chi_h) && (1.3.15)\\ &\quad + J'_{2,v}\left(w_{kh}^M\right)(\chi_h) \end{aligned}$$

for all $\psi_h$, $\chi_h \in V_h^M$. For $m = M-1, M-2, \ldots 1$, $z_{kh}^m = (\bar{u}_{kh}^m, \bar{v}_{kh}^m)$, $z_{kh}^m \in V_h^m \times V_h^m$ fulfils

$$\begin{aligned} \left(\bar{u}_{kh}^m - \bar{u}_{kh}^{m+1}, \psi_h\right) &= \frac{1}{2}(k_m + k_{m+1})\, a'(u_{kh}^m)(\psi_h, \bar{v}_{kh}^m) && (1.3.16)\\ &\quad \frac{1}{2}(k_m + k_{m+1})\, a'(u_{kh}^m)\left(\psi_h, \bar{v}_{kh}^{m+1}\right) \\ &\quad + \frac{1}{2}(k_m + k_{m+1})\, J'_{1,u}(w_{kh}^m)(\psi_h) \\ \left(\bar{v}_{kh}^m - \bar{v}_{kh}^{m+1}, \chi_h\right) &= \frac{1}{2}(k_m + k_{m+1})\left(\bar{u}_{kh}^m + \bar{u}_{kh}^{m+1}, \chi_h\right) && (1.3.17)\\ &\quad + \frac{1}{2}(k_m + k_{m+1})\, J'_{1,v}(w_{kh}^m)(\chi_h) \end{aligned}$$

for all $\psi_h, \chi_h \in V_h^m$. The function $z_{kh}^0 = \left(\bar{u}_{kh}^0, \bar{v}_{kh}^0\right) \in V_h^0 \times V_h^0$ is specified by the system

$$\begin{aligned} \left(\bar{u}_{kh}^0, \psi_h\right) &= \left(\bar{u}_{kh}^1, \psi_h\right) + \frac{1}{2} k_1 J'_{1,u}\left(w_{kh}^0\right)(\psi_h) && (1.3.18)\\ &\quad -\frac{1}{2} k_1 a'\left(u_{kh}^0\right)\left(\psi_h, \bar{v}_{kh}^1\right) \\ \left(\bar{v}_{kh}^0, \chi_h\right) &= \left(\bar{v}_{kh}^1, \chi_h\right) + \frac{1}{2} k_1 J'_{1,v}\left(w_{kh}^0\right)(\chi_h) + \frac{1}{2} k_1\left(\bar{u}_{kh}^1, \chi_h\right), && (1.3.19) \end{aligned}$$

which holds for all $\psi_h$, $\chi_h \in V_h^0 \times V_h^0$.

In the same way as in the primal time stepping scheme, we are able to decouple the systems (1.3.14-1.3.15) and (1.3.16-1.3.17). The system

(1.3.18-1.3.19) is already decoupled and consists of two $L^2$-projections. For the system (1.3.14-1.3.15), we obtain

$$\begin{aligned}
\left(\bar{v}_{kh}^M, \psi_h\right) &= -\frac{1}{4}k_M^2 a'\left(u_{kh}^M\right)\left(\psi_h, \bar{v}_{kh}^M\right) \\
&\quad + \frac{1}{4}k_M^2 J'_{1,u}\left(w_{kh}^M\right)\left(\psi_h\right) + \frac{1}{2}k_M J'_{2,u}\left(w_{kh}^M\right)\left(\psi_h\right) \\
&\quad + \frac{1}{2}k_M J'_{1,v}\left(w_{kh}^M\right)\left(\psi_h\right) + J'_{2,v}\left(w_{kh}^M\right)\left(\psi_h\right) \\
\left(\bar{u}_{kh}^M, \chi_h\right) &= \frac{2}{k_M}\left(\bar{v}_{kh}^M, \chi_h\right) - J'_{1,v}\left(w_{kh}^M\right)\left(\chi_h\right) \\
&\quad - \frac{2}{k_M} J'_{2,v}\left(w_{kh}^M\right)\left(\chi_h\right)
\end{aligned}$$

and for the system (1.3.16-1.3.17)

$$\begin{aligned}
\left(\bar{v}_{kh}^m, \psi_h\right) &= -\frac{1}{4}\left(k_m + k_{m+1}\right)^2 a'\left(u_{kh}^m\right)\left(\psi_h, \bar{v}_{kh}^m\right) \\
&\quad + \left(k_m + k_{m+1}\right)\left(\bar{u}_{kh}^{m+1}, \psi_h\right) + \left(\bar{v}_{kh}^{m+1}, \psi_h\right) \\
&\quad + \frac{1}{4}\left(k_m + k_{m+1}\right)^2 J'_{1,u}\left(w_{kh}^m\right)\left(\psi_h\right) \\
&\quad + \frac{1}{2}\left(k_m + k_{m+1}\right) J'_{1,v}\left(w_{kh}^m\right)\left(\psi_h\right) \\
&\quad - \frac{1}{4}\left(k_m + k_{m+1}\right)^2 a'\left(u_{kh}^m\right)\left(\psi_h, \bar{v}_{kh}^{m+1}\right) \\
\left(\bar{u}_{kh}^m, \chi_h\right) &= \frac{2}{k_m + k_{m+1}}\left(\bar{v}_{kh}^m - \bar{v}_{kh}^{m+1}, \chi_h\right) - J'_{1,v}\left(w_{kh}^m\right)\left(\chi_h\right) \\
&\quad - \left(\bar{u}_{kh}^{m+1}, \chi_h\right).
\end{aligned}$$

Thus, the systems (1.3.14-1.3.15) and (1.3.16-1.3.17) reduce to a linear Helmholtz equation and an $L^2$-projection. As in the primal time stepping scheme, the terms $a'\left(u_{kh}^m\right)\left(\psi_h, \bar{v}_{kh}^{m+1}\right)$, $\left(\bar{u}_{kh}^{m+1}, \psi_h\right)$, and $\left(\bar{v}_{kh}^{m+1}, \psi_h\right)$ have to be evaluated with special care, if $\mathbb{T}_h^m \neq \mathbb{T}_h^{m+1}$. Furthermore, the discrete primal solution $w_{kh}$ is needed to solve the discrete dual problem. In Section 2.5, we discuss these topics in detail.

At the best of the author's knowledge, there exists no analysis of the presented dual time stepping scheme, i.e. no convergence result and no a priori error estimate. Since this analysis is not important for the a posteriori error estimate, we only consider two numerical examples. In [**16**], this time stepping scheme is discussed in the context of parabolic problems. There, superconvergence effects in the temporal grid points are mentioned. But we are not able to confirm this claim. Since the temporal basis functions are piecewise constant, we cannot expect the error to be better than

$$\max_{t \in I} |z - z_{kh}|_1 = \mathcal{O}\,(h + k)$$

and

$$\max_{t \in I} \|z - z_{kh}\|_0 = \mathcal{O}\left(h^2 + k\right).$$

If we use the space $W_{kh} \times W_{kh}$ as test space, the standard discontinuous Galerkin method based on piecewise constant basis functions is obtained. This method converges of first order for all $t \in I$. See [**109**] for a discussion in the context of parabolic problems. However, we use linear polynomials in the test space $V_{kh} \times V_{kh}$ and come to the guess that superconvergence effects in the temporal grid points of the form

$$\max_{0 \le m \le M} |z\,(t_m) - z_{kh}\,(t_m)|_1 = \mathcal{O}\left(h + k^2\right)$$

occur. The standard derivation of superconvergence does not work, because the piecewise constant trial functions offer no possibility to enter an interpolation in the bilinear forms, which would lead to an additional power of $k$. Let us discuss this time stepping scheme in the context of ordinary differential equations (ODE) to illustrate this. We consider the ODE $\dot{u} + u = 0$ on $I$ with the intial data $u_s$. The analytical solution is given by $u = \exp(-t)u_s$. The approximation calculated on an equidistant temporal decomposition of length $k$ by the classic Crank-Nicholson scheme is

$$\tilde{u}^m = \left\{ \prod_{i=1}^{m} \left(1 - \frac{k}{2}\right) \left(1 + \frac{k}{2}\right)^{-1} \right\} u_s.$$

From

$$\left(1 + \frac{k}{2}\right)^{-1} = 1 - \frac{k}{2} + \frac{k^2}{4} - \dots,$$

we see

$$\left(1-\frac{k}{2}\right)\left(1+\frac{k}{2}\right)^{-1}=1-k+\frac{k^2}{2}+\mathcal{O}\left(k^3\right),$$

which is a third oder approximation of $\exp(-k)$. Consequently, the Crank-Nicolson scheme is of second order. For the dual time stepping scheme, we obtain the approximation

$$\tilde{u}^m=\left\{\prod_{i=2}^{m}\left(1+\frac{k}{2}\right)^{-1}\left(1-\frac{k}{2}\right)\right\}\left(1+\frac{k}{2}\right)^{-1}u_s.$$

The term $\left(1+\frac{k}{2}\right)^{-1}\left(1-\frac{k}{2}\right)$ is a third order approximation of $\exp(-k)$, as we have seen before. But the term $\left(1+\frac{k}{2}\right)^{-1}$ is only a first order approximation of $\exp(-k)$. Thus, $\tilde{u}^m$ is only a second order approximation of $\exp(-t_m)$. Therewith, the time stepping scheme is only consistent of first order.

Now, we consider a numerical example to show that no superconvergence in the temporal grid points exist in the context of linear hyperbolic problems of second order. Let $\mathcal{A}$ be the Laplacian $\Delta$. Then $a$ is the bilinear form $a\left(\nabla\cdot,\nabla\cdot\right)$. The basic domain $\Omega$ is given by $[0,1]^2$, $\Gamma_D=\partial\Omega$, $\Gamma_N=\emptyset$, and $I=[0,1]$. We assume the dual solution

$$z\left(x_1,x_2,t\right):=\sin\left(\pi x_1\right)\sin\left(\pi x_2\right)\cos\left(\pi\left(t-T\right)\right),$$

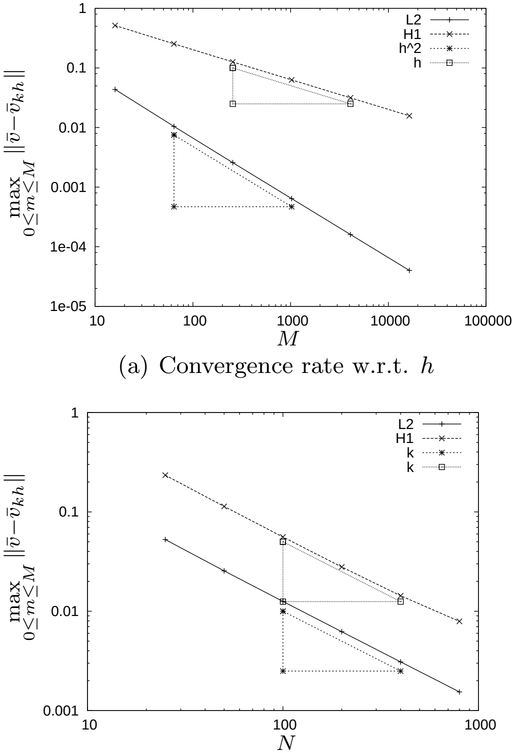
with $z\in C^{\infty}\left(\Omega\times I\right)$. With the functions

$$\begin{aligned} z_s\left(x_1,x_2,t\right) &:= \begin{pmatrix} 0 \\ \sin\left(\pi x_1\right)\sin\left(\pi x_2\right)\end{pmatrix} \\ z_f\left(x_1,x_2,t\right) &:= \begin{pmatrix} \pi^2 z\left(x_1,x_2,t\right) \\ -\pi\sin\left(\pi\left(t-T\right)\right)z\left(x_1,x_2,0\right)\end{pmatrix}, \end{aligned}$$

the output functionals $J_1$ and $J_2$ are defined as

$$\begin{aligned} J_1\left(w\right) &:= \int_{\Omega}\frac{1}{2}\left(z_f\cdot w\right)\,dx \\ J_2\left(w\right) &:= \int_{\Omega} z_s\cdot w\,dx. \end{aligned}$$

This output functional leads to the desired analytical solution.



(a) Convergence rate w.r.t. $h$

(b) Convergence rate w.r.t. $k$

FIGURE 1.3.1. Convergence rate for the dual solution $z$ w.r.t. $h$ and $k$

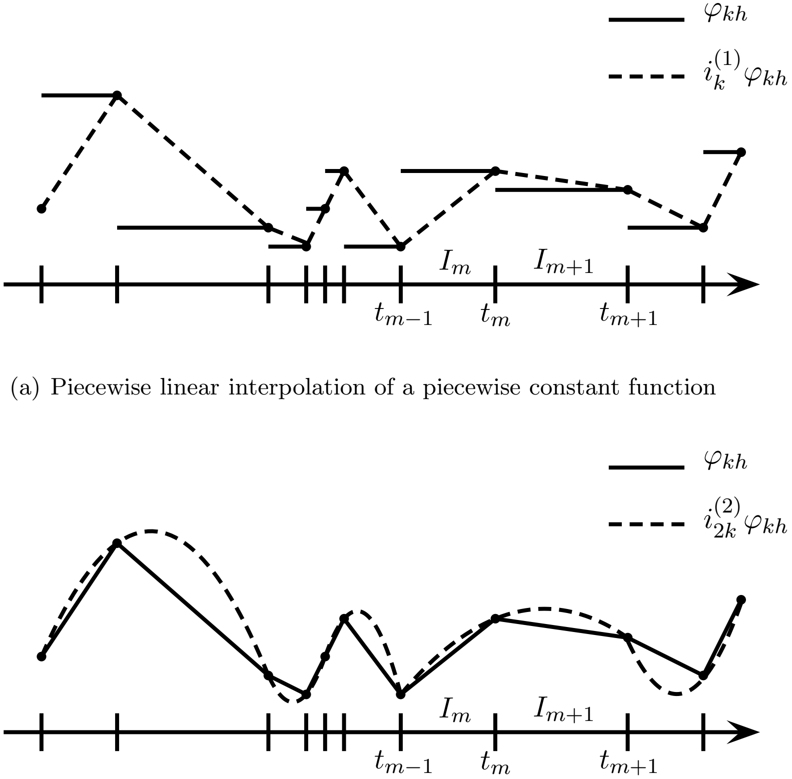
In Figure 1.3.1(a) the spatial convergence rate is depicted. The calculations have been performed with $M = 10000$ time steps. Here, $L2$ stands for the norm

$$\max_{0\le m\le M} \|z(t_m) - z_{kh}(t_m)\|_0$$

and $H1$ for

$$\max_{0\le m\le M} |z(t_m) - z_{kh}(t_m)|_1\,,$$

which would show superconvergence effects in the temporal grid points, if any exist. We observe linear convergence in the $H^1$-semi-norm and quadratic convergence in the $L^2$-norm w.r.t. $h$. This corresponds to the



(a) Piecewise linear interpolation of a piecewise constant function

(b) Piecewise quadratic interpolation of a piecewise linear function

FIGURE 1.3.2. Illustration of the interpolation operators $i_k^{(1)}$ and $i_{2k}^{(2)}$

expectations on bilinear finite elements. The convergence rate w.r.t. $k$ is presented in Figure 1.3.1(b). In these calculations, the spatial mesh consists of $N = 262144$ mesh cells. The convergence is linear in the $H^1$-semi-norm as well as in the $L^2$-norm. Since a smooth solution is considered in this example, we come to the conclusion that the dual time stepping scheme provides no superconvergence properties in the temporal grid points $t_m$.

**1.3.3. Approximate Evaluation of the Error Identity.** In the last section, we have discussed the numerical solution of the dual problem, which is needed to evaluate the error identity (1.3.8). The identity contains the weights $w - \tilde{w}_{kh}$ and $z - \tilde{z}_{kh}$, where $\tilde{w}_{kh} \in V_{kh} \times V_{kh}$ and $\tilde{z}_{kh} \in W_{kh} \times W_{kh}$ are arbitrary functions. The weights measure the approximation or interpolation error of the spaces $V_{kh}$ and $W_{kh}$ w.r.t. the continuous solutions $w$ and $z$. We are not able to evaluate these terms exactly. In literature, many approaches to approximate these terms are proposed, see, e.g., [**13**] for an overview. The idea to use a higher order interpolation of the discrete solutions $w_{kh}$ and $z_{kh}$ has turned out to be an accurate and efficient approximation. Thus, we also use this idea here. Furthermore, it allows us to split the spatial and the temporal part of the error, see Lemma 1.3.9.

In space, we need a higher order interpolation of the bilinear basis functions. Consequently, we work with biquadratic basis functions to construct the interpolation. For biquadratic basis functions, which shall be based on nodal values, nine nodal values are needed to determine the basis coefficients. We use the patch structure of the spatial meshes to obtain these nodal values, see Section 1.2.1 and Figure 1.2.2 for an illustration. The function space, which contains the interpolating functions, is

$$V_{2h}^{(2),m} := \left\{ \varphi \in C\left(\bar{\Omega}, \Gamma_D\right) \middle| \forall \mathcal{T} \in \mathbb{T}_{2h}^m : \varphi_{|\mathcal{T}} \in Q_2\left(\mathcal{T}; \mathbb{R}\right) \right\} \tag{1.3.20}$$

with $V_{2h}^{(2),m} \subset H^1\left(\Omega, \Gamma_D\right)$, where $\mathbb{T}_{2h}^m$ is the mesh of the patch elements in the $m^{\text{th}}$-time step and $Q_2\left(\mathcal{T}; \mathbb{R}\right)$ is the space of the biquadratic basis functions. Eventually, we obtain the operator $i_{2h}^{(2)} : V_h^m \to V_{2h}^{(2),m}$, which maps a finite element function from $V_h^m$ into the interpolation space $V_{2h}^{(2),m}$. Furthermore, we define the projection $\Pi_{2h}^{(2)} := i_{2h}^{(2)} - \text{id}$.

For the definition of the space time interpolations of higher order, we set $V_{kh}^{(1,1)} := V_{kh}$, $V_{kh}^{(0,1)} := W_{kh}$,

$$V_{kh}^{(0,2)} := \left\{ \varphi_{kh} \in L^2\left(I; H^1\left(\Omega, \Gamma_D\right)\right) \middle| \begin{array}{c} \varphi_{kh|I_m} \in \mathcal{P}_0\left(I_m; V_{2h}^{(2),m}\right), \\ m = 1, \ldots, M,\ \varphi_{kh}(0) \in V_h^0 \end{array} \right\}.$$

The temporal basis functions are $\tau_0^{(1)} = \tau_0$, $\tau_1^{(1)} = \tau_1$,

$$\begin{aligned}
\tau_0^{(2)}(t) &:= \frac{(t_m - t)(t_{m+1} - t)}{k_m (k_m + k_{m+1})}, \\
\tau_1^{(2)}(t) &:= \frac{(t - t_{m-1})(t_{m+1} - t)}{k_m k_{m+1}}, \\
\tau_2^{(2)}(t) &:= \frac{(t - t_{m-1})(t - t_m)}{(k_m + k_{m+1}) k_{m+1}}.
\end{aligned}$$

Furthermore, we define

$$\begin{aligned}
\tilde{\mathcal{P}}_{1,2}^m &:= \left\{ \tau_i \varphi_i \,\middle|\, \varphi_i \in V_{2h}^{(2),m-1+i},\ i = 0,1 \right\}, \\
\tilde{\mathcal{P}}_{2,1}^m &:= \left\{ \tau_i \varphi_i \,\middle|\, \varphi_i \in V_h^{m-1+i},\ i = 0,1,2 \right\}, \\
\tilde{\mathcal{P}}_{2,2}^m &:= \left\{ \tau_i \varphi_i \,\middle|\, \varphi_i \in V_{2h}^{(2),m-1+i},\ i = 0,1,2 \right\}.
\end{aligned}$$

Therewith, we denote

$$\begin{aligned}
V_{kh}^{(1,2)} &:= \left\{ \varphi_{kh} \in C\left(I; H^1(\Omega, \Gamma_D)\right) \,\middle|\, \varphi_{kh|I_m} \in \tilde{\mathcal{P}}_{1,2}^m,\ m = 1,2,\ldots,M \right\}, \\
V_{kh}^{(2,1)} &:= \left\{ \varphi_{kh} \in C\left(I; H^1(\Omega, \Gamma_D)\right) \,\middle|\, \varphi_{kh|I_m} \in \tilde{\mathcal{P}}_{2,1}^m,\ m = 1,2,\ldots,M \right\}, \\
V_{kh}^{(2,2)} &:= \left\{ \varphi_{kh} \in C\left(I; H^1(\Omega, \Gamma_D)\right) \,\middle|\, \varphi_{kh|I_m} \in \tilde{\mathcal{P}}_{2,2}^m,\ m = 1,2,\ldots,M \right\}.
\end{aligned}$$

In time, we use the spatial interpolation operator of higher order $i_{2h}^{(2)}$ transferred to the one dimensional case for the interpolation of piecewise linear and continuous functions. The approach is illustrated in Figure 1.3.2(b). The interpolation is named $i_{2k}^{(2)}$ and the exact terms are given in Section A.2, since we need them to specify the error estimator. For the piecewise constant temporal basis functions, we specify a linear interpolant $i_k^{(1)}$. The idea is exemplified in Figure 1.3.2(a) and the concrete terms are given in Section A.2, too. We define the projections

$$\Pi_k^{(1)} := i_k^{(1)} - \mathrm{id} \text{ and } \Pi_{2k}^{(2)} := i_{2k}^{(2)} - \mathrm{id}$$

for the temporal interpolants.

Now, we are able to define our space-time interpolations of higher order. For $i = 0,1,2$ and $j = 1,2$, let $\phi \in V_{kh}^{(i,j)}$. The function $\phi$ can be represented by the temporal basis functions $\tau^{(i),m}$ in the form

$$\phi(x,t) = \sum_{m=0}^{M} \tau^{(i),m}(t) \cdot \varphi^m(x)$$

with $\varphi^m(x) \in V_h^m$. We define three different space time interpolation operators: The first one only interpolates the spatial part and is given by

$$\begin{aligned} i_h^{(j,2)} \quad &: \quad V_{kh}^{(j,1)} \to V_{kh}^{(j,2)}, \\ i_h^{(j,2)}\phi(x,t) \quad &:= \quad \sum_{m=0}^{M} \tau^{(j),m}(t) \cdot i_{2h}^{(2)} \varphi^m(x), \end{aligned}$$

for $j = 0,1,2$. Let

$$\begin{aligned} i_k^{(j+1,i)} \quad &: \quad V_{kh}^{(j,i)} \to V_{kh}^{(j+1,i)}, \\ i_k^{(j+1,i)}\phi(x,t) \quad &:= \quad \sum_{m=0}^{M} i_k^{(j+1)} \tau^{(j),m}(t) \cdot \varphi^m(x), \end{aligned}$$

for $j = 0,1$ and $i = 1,2$ be the interpolation operator, which only interpolates the temporal part. The operator interpolating in space and time is

$$\begin{aligned} i_{hk}^{(j+1,2)} \quad &: \quad V_{kh}^{(j,1)} \to V_{kh}^{(j+1,2)}, \\ i_{kh}^{(j+1,2)}\phi(x,t) \quad &:= \quad \sum_{m=0}^{M} i_k^{(j+1)} \tau^{(j),m}(t) \cdot i_{2h}^{(2)} \varphi^m(x), \end{aligned}$$

for $j = 0,1$. We are now able to define the space time projections. They are given by

$$\Pi_{2h}^{(j,2)} : V_{kh}^{(j,1)} \to V_{kh}^{(j,2)}, \qquad \Pi_{2h}^{(j,2)} := i_h^{(j,2)} - \mathrm{id},$$

for $j = 0,1,2$, keep $V_{kh}^{(j,1)} \subset V_{kh}^{(j,2)}$ in mind,

$$\Pi_k^{(1,j)} : V_{kh}^{(0,j)} \to V_{kh}^{(0,j)} \cup V_{kh}^{(1,j)}, \qquad \Pi_k^{(1,j)} := i_k^{(1,j)} - \mathrm{id},$$

for $j = 1,2$,

$$\Pi_{2k}^{(2,j)} : V_{kh}^{(1,j)} \to V_{kh}^{(2,j)}, \qquad \Pi_{2k}^{(2,j)} := i_k^{(2,j)} - \mathrm{id},$$

for $j = 1, 2$, notice $V_{kh}^{(1,j)} \subset V_{kh}^{(2,j)}$,

$$\Pi_{k,2h}^{(1,2)} : V_{kh}^{(0,1)} \to V_{kh}^{(0,1)} \cup V_{kh}^{(1,2)}, \qquad \Pi_{k,2h}^{(1,2)} := i_{kh}^{(1,2)} - \mathrm{id},$$

and

$$\Pi_{2k,2h}^{(2,2)} : V_{kh}^{(1,1)} \to V_{kh}^{(2,2)}, \qquad \Pi_{2k,2h}^{(2,2)} := i_{kh}^{(2,2)} - \mathrm{id},$$

take notice of $V_{kh}^{(1,1)} \subset V_{kh}^{(2,2)}$. Now, we can state the approximation of the error identity (1.3.8):

$$\begin{aligned} & J(w) - J(w_{kh}) \\ = \; & \frac{1}{2}\left[\rho(w_{kh})(z - \tilde{z}_{kh}) + \rho^\star(w_{kh}, z_{kh})(w - \tilde{w}_{kh})\right] + \mathcal{R}_{kh} \\ \approx \; & \frac{1}{2}\left[\rho(w_{kh})\left(\Pi_{k,2h}^{(1,2)} z_{kh}\right) + \rho^\star(w_{kh}, z_{kh})\left(\Pi_{2k,2h}^{(2,2)} w_{kh}\right)\right] \\ =: \; & \eta \end{aligned}$$

Beside the approximation of the weights by the specified projections, we have neglected the remainder term $\mathcal{R}_{kh}$. It is of third order w.r.t. the error $e$, thus of higher order. Therewith, we are able to define the error estimator $\eta$. It should be remarked that the approximation sign "$\approx$" only occurs here in the derivation of the error estimate. In every other step, only real "$=$" signs occur. Furthermore, no other "higher order arguments" are involved. In [**99**], the derivation is based on the semi- and the full-discrete problem formulation to split the spatial and the temporal error estimator part. There, the unknown semi-discrete solution has to be approximated by the full-discrete one, which involves a higher order argument. In contrast to this, we use the following lemma to split the spatial and the temporal part of the a posteriori error estimate:

LEMMA 1.3.9. *The following identities hold:*

$$\begin{aligned} \Pi_{k,2h}^{(1,2)} &= i_k^{(1,2)} \Pi_{2h}^{(0,2)} + \Pi_k^{(1,1)} \\ &= i_h^{(1,2)} \Pi_k^{(1,1)} + \Pi_{2h}^{(0,2)}, \\ \Pi_{2k,2h}^{(2,2)} &= i_k^{(2,2)} \Pi_{2h}^{(1,2)} + \Pi_{2k}^{(2,1)} \\ &= i_h^{(2,2)} \Pi_{2k}^{(2,1)} + \Pi_{2h}^{(1,2)}. \end{aligned}$$

Proof of Lemma 1.3.9. Let $\varphi_{kh}$ be an arbitrary function from $V_{kh}^{(1,1)}$. It holds

$$\begin{aligned}\Pi_{2k,2h}^{(2,2)}\varphi_{kh} &= i_{kh}^{(2,2)}\varphi_{kh}-\varphi_{kh}\\ &= i_{kh}^{(2,2)}\varphi_{kh}-i_{k}^{(2,1)}\varphi_{kh}+i_{k}^{(2,1)}\varphi_{kh}-\varphi_{kh}.\end{aligned}$$

Since $\varphi_{kh}$ is a tensor product function in space and time, we obtain

$$\begin{aligned}\Pi_{2k,2h}^{(2,2)}\varphi_{kh} &= i_{kh}^{(2,2)}\varphi_{kh}-i_{k}^{(2,1)}\varphi_{kh}+i_{k}^{(2,1)}\varphi_{kh}-\varphi_{kh}\\ &= i_{k}^{(2,2)}\left(i_{h}^{(1,2)}\varphi_{kh}-\varphi_{kh}\right)+\Pi_{2k}^{(2,1)}\varphi_{kh}\\ &= i_{k}^{(2,2)}\Pi_{2h}^{(1,2)}\varphi_{kh}+\Pi_{2k}^{(2,1)}\varphi_{kh}.\end{aligned}$$

Furthermore, we have

$$\begin{aligned}\Pi_{2k,2h}^{(2,2)}\varphi_{kh} &= i_{kh}^{(2,2)}\varphi_{kh}-\varphi_{kh}\\ &= i_{kh}^{(2,2)}\varphi_{kh}-i_{h}^{(1,2)}\varphi_{kh}+i_{h}^{(1,2)}\varphi_{kh}-\varphi_{kh}\\ &= i_{h}^{(2,2)}\left(i_{k}^{(2,1)}\varphi_{kh}-\varphi_{kh}\right)+\Pi_{2h}^{(1,2)}\varphi_{kh}\\ &= i_{h}^{(2,2)}\Pi_{k}^{(1,1)}\varphi_{kh}+\Pi_{2h}^{(1,2)}\varphi_{kh}.\end{aligned}$$

The indentities for $\Pi_{k,2h}^{(1,2)}$ are derived analogously. □

Definition 1.3.10. Let us now define the spatial error estimator terms

$$\begin{aligned}\eta_h^n &:= \frac{1}{2}\left[\rho\left(w_{kh}\right)\left(\Pi_{2h}^{(0,2)}z_{kh}\right)+\rho^\star\left(w_{kh},z_{kh}\right)\left(\Pi_{2h}^{(1,2)}w_{kh}\right)\right],\\ \eta_h^i &:= \frac{1}{2}\left[\rho\left(w_{kh}\right)\left(i_{k}^{(1,2)}\Pi_{2h}^{(0,2)}z_{kh}\right)+\rho^\star\left(w_{kh},z_{kh}\right)\left(i_{k}^{(1,2)}\Pi_{2h}^{(1,2)}w_{kh}\right)\right],\end{aligned}$$

and the temporal error estimator terms

$$\begin{aligned}\eta_k^n &:= \frac{1}{2}\left[\rho\left(w_{kh}\right)\left(\Pi_{k}^{(1,1)}z_{kh}\right)+\rho^\star\left(w_{kh},z_{kh}\right)\left(\Pi_{2k}^{(2,1)}w_{kh}\right)\right],\\ \eta_k^i &:= \frac{1}{2}\left[\rho\left(w_{kh}\right)\left(i_{h}^{(1,2)}\Pi_{k}^{(1,1)}z_{kh}\right)+\rho^\star\left(w_{kh},z_{kh}\right)\left(i_{h}^{(2,2)}\Pi_{2k}^{(2,1)}w_{kh}\right)\right].\end{aligned}$$

Furthermore, we set

$$\begin{aligned}\eta_{nn} &:= \eta_h^n+\eta_k^n,\\ \eta_{ni} &:= \eta_h^n+\eta_k^i,\end{aligned}$$

$$\begin{aligned}\eta_{in} &:= \eta_h^i + \eta_k^n,\\ \eta_{ii} &:= \eta_h^i + \eta_k^i.\end{aligned}$$

Corollary 1.3.11. *It holds*

$$\eta = \eta_{ni} = \eta_{in}.$$

Proof of Corollary 1.3.11. Lemma 1.3.9 implies the assertion, since the residuals are linear in the second argument. □

The numerical results presented in the next section substantiate that both $\eta_h^n$ and $\eta_h^i$ measure the spatial error and that $\eta_k^n$ as well as $\eta_k^i$ represent the temporal error. The concrete form of $\eta_h^n$ is described in Section A.4.1, $\eta_h^i$ in Section A.4.2. The temporal error estimator terms $\eta_k^n$ and $\eta_k^i$ are worked out in Section A.3.1 and A.3.2, respectively. We exactly evaluate the temporal integrals in all terms except for

$$\left(\left(f, i_k^{(1,1)}\bar{v}_{kh} - \bar{v}_{kh}\right)\right)_m \text{ and } \left(\left(q, i_k^{(1,1)}\bar{v}_{kh} - \bar{v}_{kh}\right)_{\Gamma_N}\right)_m.$$

We consider $\eta_k^n$, the following discussion directly carries over to $\eta_k^i$. The first idea is to use Simpson's rule to approximate

$$\left(\left(f, i_k^{(1,1)}\bar{v}_{kh} - \bar{v}_{kh}\right)\right)_m \text{ and } \left(\left(q, i_k^{(1,1)}\bar{v}_{kh} - \bar{v}_{kh}\right)_{\Gamma_N}\right)_m.$$

But this leads to a less accurate error estimation, since $\rho\left(w_{kh}\right)\left(z_{kh}\right) = 0$ does not hold anymore. Consequently, we use the trapezoidal rule to approximate the temporal integrals

$$\left(\left(f, \bar{v}_{kh}\right)\right)_m \text{ and } \left(\left(q, \bar{v}_{kh}\right)_{\Gamma_N}\right)_m.$$

Then, $\rho\left(w_{kh}\right)\left(z_{kh}\right) = 0$ holds. Simpson's formula is applied to evaluate

$$\left(\left(f, i_k^{(1,1)}\bar{v}_{kh}\right)\right)_m \text{ and } \left(\left(q, i_k^{(1,1)}\bar{v}_{kh}\right)_{\Gamma_N}\right)_m.$$

See Section A.1 for a short overview of quadrature rules.

For linear problems with linear output functionals $J$, it is sufficient to work with the primal residuals, since they are equal to the dual ones.

Thus, we define the primal spatial error estimator terms

$$\begin{aligned} \eta_h^{p,n} &:= \rho\left(w_{kh}\right)\left(\Pi_{2h}^{(0,2)} z_{kh}\right), \\ \eta_h^{p,i} &:= \rho\left(w_{kh}\right)\left(i_k^{(1,2)} \Pi_{2h}^{(0,2)} z_{kh}\right), \end{aligned}$$

and the primal temporal error estimator terms

$$\begin{aligned} \eta_k^{p,n} &:= \rho\left(w_{kh}\right)\left(\Pi_{k}^{(1,1)} z_{kh}\right), \\ \eta_k^{p,i} &:= \rho\left(w_{kh}\right)\left(i_h^{(1,2)} \Pi_{k}^{(1,1)} z_{kh}\right). \end{aligned}$$

Furthermore, we set

$$\begin{aligned} \eta_{nn}^p &:= \eta_h^{p,n} + \eta_k^{p,n}, & \eta_{ni}^p &:= \eta_h^{p,n} + \eta_k^{p,i}, \\ \eta_{in}^p &:= \eta_h^{p,i} + \eta_k^{p,n}, & \eta_{ii}^p &:= \eta_h^{p,i} + \eta_k^{p,i}. \end{aligned}$$

Since the primal residual $\rho$ and the dual residual $\rho^\star$ are linear in the second argument, Lemma 1.3.9 shows that $\eta_{ni}^p = \eta_{in}^p = \eta^p$ holds.

The error estimator $\eta_{nn}$ corresponds to the error estimator, which is obtained, if the aguments used in [**99**] are transferred to hyperbolic problems of second order. The definition of $\eta_{ii}$ is an ad hoc definition, which is an obvious idea proposed by the used concept. The error estimator $\eta_{nn}^p$ is similar to the one derived in [**9**].

In the subsequent chapters, we will write $i_{2h}^{(2)}$ for all spatial interpolations of higher order to ease the notation, because it is obvious from the argument, which interpolation one has to take. The same holds for the temporal interpolations of higher order $i_k^{(1)}$ and $i_{2k}^{(2)}$ as well as for the spatial and temporal projections $\Pi_{2h}^{(2)}$, $\Pi_k^{(1)}$ and $\Pi_{2k}^{(2)}$.

## 1.4. Numerical Results

In this section, we test the a posteriori error estimation by an example, for which the analytical solution is known. The differential operator is $\mathcal{A}(u) = \Delta u + u^3$. The spatial domain is $\Omega = [0,1]^2$, where $\Gamma_N = \{x = (x_1, x_2) \in \partial\Omega \,|\, x_1 = 1\}$ and $\Gamma_D = \partial\Omega \backslash \Gamma_N$. The time interval is $I =$

| $M$ | $N$ | $E_{\text{rel}}$ | $I_{\text{eff}}^{nn}$ | $I_{\text{eff}}^{in}$ | $I_{\text{eff}}^{ni}$ | $I_{\text{eff}}^{ii}$ |
|---|---|---|---|---|---|---|
| 50 | 64 | $-4.512 \cdot 10^{-3}$ | 0.534 | 0.561 | 0.561 | 0.591 |
| 100 | 256 | $-9.168 \cdot 10^{-4}$ | 0.808 | 0.845 | 0.845 | 0.884 |
| 200 | 1024 | $-2.164 \cdot 10^{-4}$ | 0.955 | 0.979 | 0.979 | 1.004 |
| 400 | 4096 | $-5.335 \cdot 10^{-5}$ | 1.001 | 1.014 | 1.014 | 1.026 |
| 800 | 16384 | $-1.344 \cdot 10^{-5}$ | 1.022 | 1.029 | 1.029 | 1.035 |
| 1600 | 65536 | $-3.398 \cdot 10^{-6}$ | 1.037 | 1.040 | 1.040 | 1.043 |

TABLE 1.4.1. Effectivity indices for different estimators based on the primal and dual residual w.r.t. uniform refinement

| $M$ | $N$ | $E_{\text{rel}}$ | $I_{\text{eff}}^{p,nn}$ | $I_{\text{eff}}^{p,in}$ | $I_{\text{eff}}^{p,ni}$ | $I_{\text{eff}}^{p,ii}$ |
|---|---|---|---|---|---|---|
| 50 | 64 | $-4.512 \cdot 10^{-3}$ | 0.564 | 0.629 | 0.629 | 0.710 |
| 100 | 256 | $-9.168 \cdot 10^{-4}$ | 0.907 | 1.005 | 1.005 | 1.125 |
| 200 | 1024 | $-2.164 \cdot 10^{-4}$ | 1.126 | 1.195 | 1.195 | 1.273 |
| 400 | 4096 | $-5.335 \cdot 10^{-5}$ | 1.209 | 1.247 | 1.247 | 1.286 |
| 800 | 16384 | $-1.344 \cdot 10^{-5}$ | 1.247 | 1.266 | 1.266 | 1.285 |
| 1600 | 65536 | $-3.398 \cdot 10^{-6}$ | 1.269 | 1.278 | 1.278 | 1.288 |

TABLE 1.4.2. Effectivity indices for different estimators based on the primal residual w.r.t. uniform refinement

| $M$ | $N$ | $\eta_k^n\ (\cdot 10^{-7})$ | $\eta_k^i (\cdot 10^{-7})$ | $\eta_k^{p,n} (\cdot 10^{-7})$ | $\eta_k^{p,i} (\cdot 10^{-7})$ |
|---|---|---|---|---|---|
| 200 | 64 | 2.828 | 12.38 | 7.641 | 26.77 |
| 200 | 256 | 2.540 | 5.270 | 7.378 | 12.85 |
| 200 | 1024 | 2.487 | 3.182 | 7.327 | 8.719 |
| 200 | 4096 | 2.472 | 2.647 | 7.313 | 7.664 |
| 200 | 16384 | 2.468 | 2.512 | 7.310 | 7.398 |
| 200 | 65536 | 2.467 | 2.478 | 7.309 | 7.331 |

TABLE 1.4.3. Behaviour of $\eta_k$ w.r.t. uniform spatial refinement

$[0, 1]$. We choose the analytical solution

$$u\left(x_1, x_2, t\right) := \sin\left(\pi t\right) \sin\left(\frac{1}{4}\pi x_1\right) \sin\left(\pi x_2\right).$$

| $M$ | $N$ | $\eta_h^n\ (\cdot 10^{-6})$ | $\eta_h^i (\cdot 10^{-6})$ | $\eta_h^{p,n} (\cdot 10^{-6})$ | $\eta_h^{p,i} (\cdot 10^{-6})$ |
|---|---|---|---|---|---|
| 50 | 1024 | −2.933 | −2.563 | −3.130 | −2.385 |
| 100 | 1024 | −3.048 | −2.895 | −3.146 | −2.839 |
| 200 | 1024 | −3.103 | −3.033 | −3.153 | −3.014 |
| 400 | 1024 | −3.127 | −3.093 | −3.151 | −3.085 |
| 800 | 1024 | −3.138 | −3.122 | −3.149 | −3.117 |
| 1600 | 1024 | −3.144 | −3.136 | −3.149 | −3.133 |
| 3200 | 1024 | −3.147 | −3.136 | −3.149 | −3.141 |

TABLE 1.4.4. Behaviour of $\eta_h$ w.r.t. uniform temporal refinement

Hence, $f$ and $q$ are set to

$$f := \ddot{u} - \Delta u - u^3, \quad q := \frac{\partial u}{\partial x_1}.$$

The initial values are given by $u_s = u\,(x_1, x_2, 0)$ and $v_s = \dot{u}\,(x_1, x_2, 0)$. As functional of interest, we choose

$$J_1\,(w) = \frac{1}{|B|}\int_B \left(u^2 + v^2\right)\,dx, \quad J_2(w) = 0, \quad B := B^\infty_{\frac{1}{8}}\left(\frac{1}{8}, \frac{1}{8}\right).$$

The effectivity index is defined as usual by

$$I_{\text{eff}} := \frac{J\,(w) - J\,(w_{kh})}{\eta}.$$

We add the indices of the different estimators $\eta$ to $I_{\text{eff}}$ in order to distinguish the effectivity indices. For instance, $I_{\text{eff}}^{nn}$ represents the effectivity index w.r.t. $\eta_{nn}$. The relative error is given by

$$E_{\text{rel}} := \frac{|J\,(w) - J\,(w_{kh})|}{|J\,(w)|}.$$

In Table 1.4.1 and Table 1.4.2, the effectivity indices for different error estimators are presented, where the temporal and the spatial mesh are uniformly refined. We observe that the effectivity indices of $\eta_{nn}$, $\eta_{ni}$, $\eta_{in}$, and $\eta_{ii}$ are similar, those of $\eta_{ni}$ and $\eta_{in}$ are equal as expected. For these error estimators, we obtain a very good estimate of the error with effectivity indices in the range of 1.0. The error estimators based only on

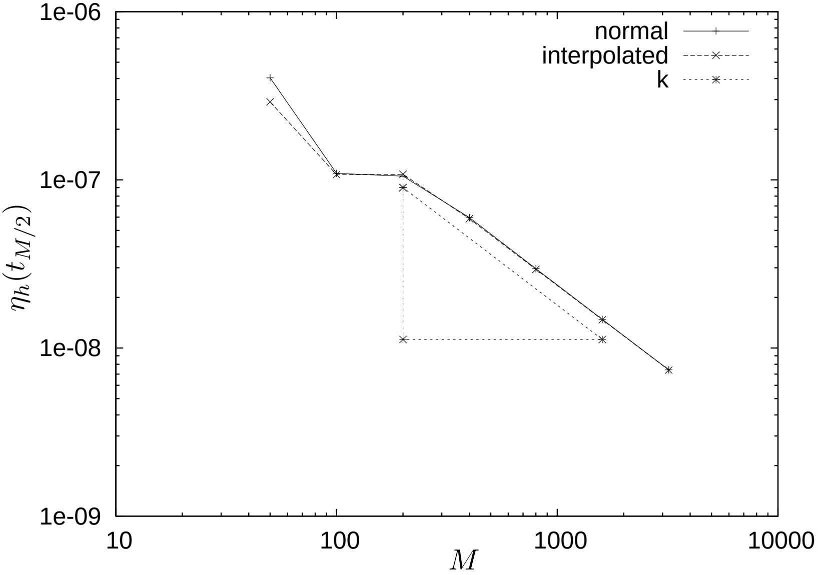


FIGURE 1.4.1. Development of $\eta_h^n(t_{M/2})$, $\eta_h^i(t_{M/2})$ w.r.t. $k$

the primal residuals show worse results, which was expected, because we consider a nonlinear problem here.

The reasonability of the definitions of $\eta_k$ and $\eta_h$ are supported in the Tables 1.4.3 and 1.4.4. We observe that $\eta_h$ is almost constant, if the temporal mesh is refined uniformly, while the spatial mesh is kept constant. There exists a dependence only of higher order. Vice versa, $\eta_k$ is almost constant, if the spatial mesh is refined uniformly and the temporal mesh is left unchanged. We observe the same behaviour of $\eta_k$ and $\eta_h$, if the meshes are adaptively refined. A direct consequence of this property of $\eta_h$ is shown in Figure 1.4.1. If $\eta_h$ is constant w.r.t. temporal refinement, then the terms of $\eta_h$ in the single time steps $\eta_h^m$ have to depend on $k$. We observe a linear depence on $k$ for every fixed temporal grid point $t_m$. Figure 1.4.1 shows a representative example. We have to keep this dependence in mind, when we develop the adaptive spatial refinement strategy in Section 2.2.

CHAPTER 2

# Adaptive Mesh Refinement

In the preceding chapter, we have derived an a posteriori error estimator, which accurately estimates the discretisation error w.r.t. an output functional $J$. Now, a second property of the error estimator is utilised. It reflects the spatial and temporal distribution of the discretisation error. Thus, we improve the solution by refining only those mesh cells, where the estimated error is large. We have to adapt the error estimator in such a way that it can serve as basis of the adaptive refinement process, i.e. we have to derive appropriate refinement indicators. This is the topic of the first section.

In Section 2.2, we present the adaptive refinement algorithm, which is based on the space-time equilibration strategy [**99**] and the optimal mesh strategy [**97, 98**]. Mesh refinement is only considered, algorithms for mesh coarsening are not included. The mesh properties, like patch structure, only spatial and temporal hanging nodes of degree one, are not conserved during the adaptive refinement of the spatial meshes. Consequently, we have to regularise the mesh, i.e. additional mesh cells are refined such that the mentioned properties hold. The regularisation algorithms are presented and analysed in Section 2.3. Combining all presented algorithms, we describe the whole adaptive solution algorithm. Its implementation is the topic of Section 2.5. We present a heuristic error estimator in Section 2.6, which is compared with the derived error estimator. In numerical examples, we observe that the presented adaptive solution algorithm leads to an efficient discretisation scheme.

In general, adaptive algorithms for dynamic problems are based on refinement strategies, which are known from static problems, see, for instance, [**13**], Section 4.2, and [**111**], Section 4.1, for a survey of adaptive algorithms for static problems. Commonly used adaptive algorithms for time dependent problems, see, e.g., [**100**], Section 1.5.4, and [**78**], perform an adaptive refinement process using a prescribed tolerance in every time step. This refinement process is independent of previous and subsequent time steps. The crucial point is that the time interval is passed only once. The tolerance limit cannot be reached, if the solution in the previous time step has not been calculated exactly enough. Moreover, the difference of the meshes of two successive time steps may significantly increase the error. Usually, rapid changes of the problem parameters are the reason for this behaviour. An alternative is given by algorithms based on the ideas in [**12, 99**], where a more global approach is used. The time interval is passed multiple times and the adaptive refinement strategies take the refinement indicators of several or all time steps into account. Here, we significantly extend these ideas to adapt them to our setting.

## 2.1. Refinement Indicators

In this section, we derive refinement indicators for an adaptive refinement process on the basis of the presented error estimator. The first step is the localisation of the error estimator to the single cells. Furthermore, we have to compensate the time dependence of the local spatial error estimates and the definition of the spatial higher order interpolation at hanging nodes is discussed. The last step is smoothing the indicators. We end up with appropriate refinement indicators, i.e. they lead to good adaptive meshes, as we will see in Section 2.7.

**2.1.1. Localisation of the Error Estimator.** In this chapter, we present all results for the error estimator $\eta_{ni}$. The derivation is analogous for $\eta_{nn}$, $\eta_{in}$, and $\eta_{ii}$. The localisation in temporal direction is easy. We split the integral over $I$ into the sum over all subintervals $I_m$, $0 < m \leq M$

and the error estimate in the initial values. For this purpose, we define

$$\eta_h^n := \sum_{jm=0}^{M} \eta_h^{n,m} \text{ and } \eta_k^i := \sum_{m=1}^{M} \eta_k^{i,m}.$$

The detailed form of $\eta_k^{i,m}$ is presented in Section A.3 and of $\eta_h^{n,m}$ in Section A.4.

Up to now, we have localised the error estimate to the single temporal subintervals. This localisation is sufficient for the adaptive temporal refinement, where the set

$$\tilde{\Theta}_k := \left\{ \eta_k^{i,m} \middle| m = 1, 2, \ldots, M \right\}$$

serve as basis for the refinement strategy. We have to localise the spatial error estimator $\eta_h^{n,m}$ to the single mesh cells in every time step. In literature, two different techniques to realise the spatial localisation are known: The first one is the cellwise integration by parts of the differential operator and the other one is based on a filtering operator. To be complete, we briefly present both methods:

The method based on integration by parts is the “classic” choice, see, for instance, [**4, 111**]. In this approach, only the terms

$$a\left(u_{kh}^m\right)\left(i_{2h}^{(2)}\bar{v}_{kh}^m - \bar{v}_{kh}^m\right) \text{ and } a'\left(u_{kh}^m\right)\left(i_{2h}^{(2)}u_{kh}^m - u_{kh}^m, \bar{v}_{kh}^m\right)$$

have to be modified. We exemplarily study the term

$$a\left(u_{kh}^m\right)\left(i_{2h}^{(2)}\bar{v}_{kh}^m - \bar{v}_{kh}^m\right),$$

the detailed expressions are given in Section (A.5). The part

$$\left(b\left(u_{kh}^m\right), i_{2h}^{(2)}\bar{v}_{kh}^m - \bar{v}_{kh}^m\right)$$

is left unchanged, thus we assume $b = 0$ for a moment. We obtain

$$a\left(u_{kh}^m\right)\left(\varphi_{kh}^m\right) = \left(\bar{a}\left(u_{kh}^m\right), \nabla\varphi_{kh}^m\right),$$

where

$$\bar{a}\left(u_{kh}^m\right) := \begin{pmatrix} a_1\left(x, t, u_{kh}^m, \nabla u_{kh}^m\right) \\ a_2\left(x, t, u_{kh}^m, \nabla u_{kh}^m\right) \end{pmatrix}.$$

The integral over $\Omega$ is split into

$$a\left(u_{kh}^m\right)\left(\varphi_{kh}^m\right) = \sum_{\mathcal{T}\in\mathbb{T}_h^m} \left(\bar{a}\left(u_{kh}^m\right), \nabla\varphi_{kh}^m\right)_{\mathcal{T}}.$$

Integration by parts on each $\mathcal{T}$ leads to

$$\left(\bar{a}\left(u_{kh}^m\right), \nabla\varphi_{kh}^m\right)_{\mathcal{T}} = \left(-\mathcal{A}\left(u_{kh}^m\right), \varphi_{kh}^m\right)_{\mathcal{T}} + \sum_{E\in\partial\mathcal{T}} \left(\mathcal{B}\left(u_{kh}^m\right), \varphi_{kh}^m\right)_E.$$

Summing up, we obtain

$$\begin{aligned} a\left(u_{kh}^m\right)\left(\varphi_{kh}^m\right) &= \sum_{\mathcal{T}\in\mathbb{T}_h^m} \left(-\mathcal{A}\left(u_{kh}^m\right), \varphi_{kh}^m\right)_{\mathcal{T}} \\ &\quad + \frac{1}{2}\sum_{\mathcal{T}\in\mathbb{T}_h^m}\sum_{E\in\partial\mathcal{T}} \left(\left[\mathcal{B}\left(u_{kh}^m\right)\right], \varphi_{kh}^m\right)_E, \end{aligned} \tag{2.1.1}$$

where

$$\begin{aligned} &\left[\mathcal{B}\left(u_{kh}^m\right)\right] \\ :=& \begin{cases} \mathcal{B}\left(u_{kh}^m\right)_{|\mathcal{T}} - \mathcal{B}\left(u_{kh}^m\right)_{|\mathcal{T}'}, \ E\subset\mathcal{T}', \mathcal{T}\neq\mathcal{T}', & \text{if } E\not\subset\partial\Omega \\ 0, & \text{if } E\subset\Gamma_D, \\ 2\left(q-\mathcal{B}\left(u_{kh}^m\right)\right), & \text{if } E\subset\Gamma_N. \end{cases} \end{aligned} \tag{2.1.2}$$

Equation (2.1.1) shows that we have to evaluate the strong differential operator and the jump terms on the boundary of the mesh cells. This considerably complicates the calculations, since we have to provide the second spatial derivative of the solution and the evaluation of the gradient operator $\mathcal{B}$ on the edges. Furthermore, a loop over all edges has to be realised, where it is very complicated due to the mesh changes to ensure that all integrals are correctly evaluated.

An alternative is given by the following method, which goes back to [**29**]. Since the description of this method in the context of the presented error estimator leads to a complex notation, we restrict ourselves to the term

$$a\left(u_{kh}^m\right)\left(i_{2h}^{(2)}\bar{v}_{kh}^m - \bar{v}_{kh}^m\right)$$

again. The application to the other terms is carried out in exactly the same manner.

Let

$$\left\{\alpha_j : \Omega \to \mathbb{R} \,\middle|\, j = 1, 2, \ldots, \bar{N}^m := \dim V_h^m \right\}$$

be the nodal Lagrange basis of $V_h^m$ and

$$\left\{\beta_j = i_{2h}^{(2)} \alpha_j \,\middle|\, j = 1, 2, \ldots, \bar{N}^m \right\}$$

the basis of $V_{2h}^{(2),m}$. Furthermore, $\bar{V}_j \in \mathbb{R}$, $j = 1, 2, \ldots, \bar{N}^m$, are the coefficients of $\bar{v}_{kh}^m$, i.e.

$$\bar{v}_{kh}^m = \sum_{j=1}^{\bar{N}^m} \bar{V}_j \alpha_j \quad \text{and} \quad i_{2h}^{(2)} \bar{v}_{kh}^m = \sum_{j=1}^{\bar{N}^m} \bar{V}_j \beta_j .$$

We set $\bar{V} = \left(\bar{V}_1, \ldots, \bar{V}_{\bar{N}^m}\right)^\top$. The value

$$\Psi_j := a\left(u_{kh}^m\right)\left(\beta_j - \alpha_j\right)$$

represents the term $a\left(u_{kh}^m\right)(\cdot)$ w.r.t. the difference of the biquadratic basis $\{\beta_j\}$ and the bilinear one $\{\alpha_j\}$. The vector $\Psi = \left(\Psi_1, \ldots, \Psi_{\bar{N}^m}\right)^\top$ is assembled in the same as, i.e., the right hand side. The value of

$$a\left(u_{kh}^m\right)\left(i_{2h}^{(2)} \bar{v}_{kh}^m - \bar{v}_{kh}^m\right)$$

is given by the Euclidean inner product in $\mathbb{R}^{\bar{N}^m}$ of $\Psi$ and $\bar{V}$, i.e.

$$a\left(u_{kh}^m\right)\left(i_{2h}^{(2)} \bar{v}_{kh}^m - \bar{v}_{kh}^m\right) = \Psi^\top \bar{V} = \sum_{j=1}^{\bar{N}^m} \Psi_j \bar{V}_j .$$

The space

$$V_{2h}^m := \left\{\varphi \in C\left(\bar{\Omega}, \Gamma_D\right) \,\middle|\, \forall \mathcal{T} \in \mathbb{T}_{2h}^m : \varphi_{|\mathcal{T}} \in Q_1\left(\mathcal{T}; \mathbb{R}\right)\right\} \subset H^1\left(\Omega, \Gamma_D\right)$$

consists of bilinear basis functions on patches. Because of the patch structure of the meshes, we have $V_{2h}^m \subseteq V_h^m$. The operator $i_{2h}^{(1)} : V_h^m \to V_{2h}^m$ interpolates a function from $V_h^m$ in $V_{2h}^m$. We define the operator $\pi := \mathrm{id} - i_{2h}^{(1)}$ and call $\pi$ filtering operator. The nodal vector $\bar{V}^\pi$ denotes the coefficients of the filtered function $\pi\bar{v}_{kh}^m$ w.r.t. the basis $\alpha_j$, i.e.

$$\pi\bar{v}_{kh}^m = \sum_{j=1}^{\bar{N}^m} \bar{V}_j^\pi \alpha_j.$$

The interpolation operator $i_{2h}^{(2)}$ is the identity on $V_{2h}^m$. Thus, we obtain

$$i_{2h}^{(2)}\pi\alpha_j - \pi\alpha_j = i_{2h}^{(2)}\alpha_j - \alpha_j = \beta_j - \alpha_j.$$

The linearity of the second argument leads to

$$a\left(u_{kh}^m\right)\left(i_{2h}^{(2)}\bar{v}_{kh}^m - \bar{v}_{kh}^m\right) = \sum_{j=1}^{\bar{N}^m} \Psi_j \bar{V}_j = \sum_{j=1}^{\bar{N}^m} \Psi_j \bar{V}_j^\pi,$$

see [**29**] for the detailed calculation. We end up with the nodal values $\Psi_j\bar{V}_j^\pi$, which provide a sufficient localisation of the term

$$a\left(u_{kh}^m\right)\left(i_{2h}^{(2)}\bar{v}_{kh}^m - \bar{v}_{kh}^m\right),$$

c.f. [**29**]. Finally, the values $\Psi_j\bar{V}_j^\pi$ have to be shifted from the nodes to the cells. We use the method presented in [**29**], i.e. we simply take the mean value of the values $\Psi_j\bar{V}_j^\pi$ of all nodes, which are the vertices of the mesh cell.

The filtering approach includes the application of the filtering operator and the shift to the mesh cells as additional operations. These additional operations lead to a small computational effort, especially in node oriented finite element codes.

We work with two different approaches to the spatial adaptivity. In the first approach, we set $V_h^m = V_h$ for all $m = 0, 1, \dots, M$, i.e. the meshes do not change from time step to time step. However, the underlying mesh $\mathbb{T}_h$ is adaptively refined. This method is called constant mesh approach (CM). The alternative is that the meshes can change from time step to time step. We call this the dynamic mesh approach (DM). Using one of the presented localisation methods, we obtain error indicators $\eta_T^{n,m}$ for

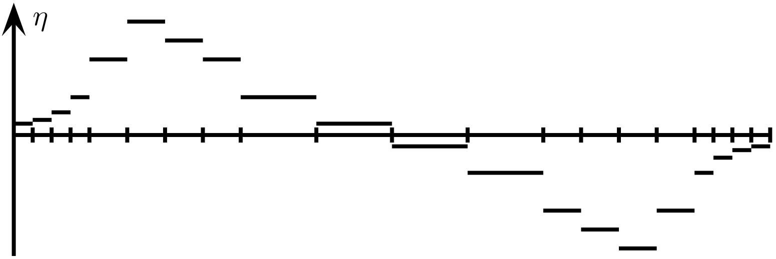


FIGURE 2.1.1. Illustration of the behaviour of a refinement indicator

all $\mathcal{T} \in \mathbb{T}_h^m$ and all $m = 0, 1, \ldots, M$. The sets

$$\begin{aligned}\tilde{\Theta}_h^{CM} &:= \left\{ \eta_{\mathcal{T}}^n := \sum_{m=0}^{M} \eta_{\mathcal{T}}^{n,m} \,\middle|\, \mathcal{T} \in \mathbb{T}_h \right\},\\ \tilde{\Theta}_h^{DM} &:= \left\{ \eta_{\mathcal{T}}^{n,m} \,\middle|\, \mathcal{T} \in \mathbb{T}_h^m,\ m = 0, 1, \ldots, M \right\}\end{aligned}$$

are the basis for the spatial adaptive mesh refinement.

**2.1.2. Adaption of the Refinement Indicators to the Refinement Strategy.** In the last section, we have localised the error estimate to single mesh cells. We want to use these values to decide, which elements are refined and which are left unchanged. For this end, the refinement strategies discussed in Section 2.2 are applied. All refinement strategies known from literature, just like the ones we use, are based on sorting the indicators by size, i.e. we need to take the absolute value of the indicators. The first and "classic" idea is to take the absolute value of the refinement indicator on a mesh element, i.e. to consider for instance $\iota_{\mathcal{T},o}^{n,m} := \left|\eta_{\mathcal{T}}^{n,m}\right|$. We call this approach "outside". However, as we will see in Section 2.7.1, this approach does not necessarily lead to good adaptive meshes. The reason is illustrated in Figure 2.1.1, where a typical distribution of refinement indicators in hyperbolic problems is depicted. We get small absolute values of the indicators, where the sign of the indicators changes. However, a sign change of the indicator does not necessarily imply that the error is small. Our approach to overcome this difficulty is to take the absolute value in different ways. We test the following three additional alternatives, which we explain by the example of $\eta_{\mathcal{T}}^{n,m}$ : The first one is named “residual”, where the absolute value of the primal residual $\rho$ and the dual residual $\rho^\star$ is added, i.e.

$$\iota_{\mathcal{T},r}^{n,m} := \frac{1}{2}\left(\left|\rho\left(w_{kh}\right)\left(\Pi_{k,2h}^{(1,2)} z_{kh}\right)_{|\mathcal{T}}\right| + \left|\rho^\star\left(w_{kh}, z_{kh}\right)\left(\Pi_{2k,2h}^{(2,2)} w_{kh}\right)_{|\mathcal{T}}\right|\right).$$

In the approach named “terms”, we take the absolute value of every contributing term:

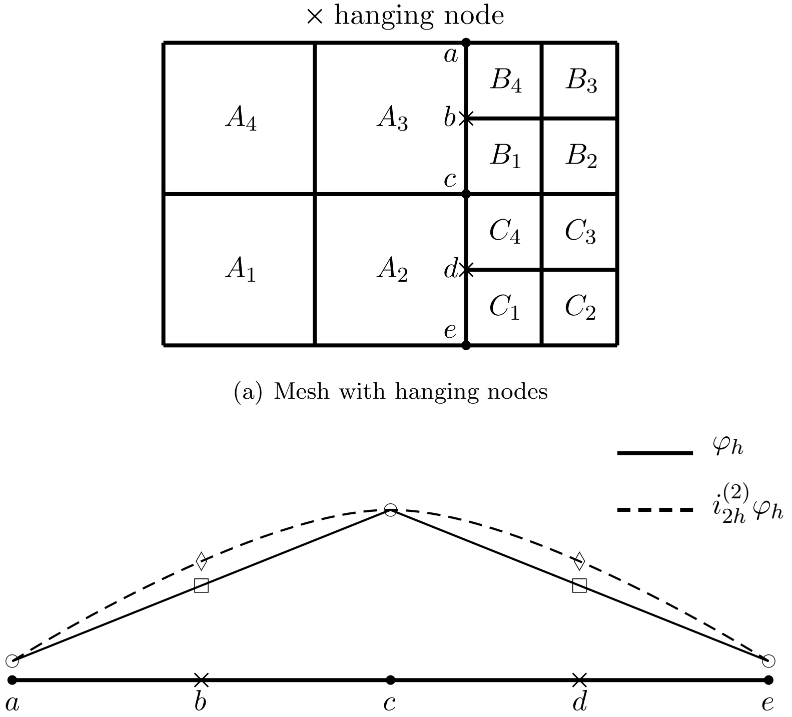
$$\iota_{\mathcal{T},t}^{n,m} := \sum_{i=1}^{6}\left|\eta_{\mathcal{T},i}^{n,m}\right|,$$

see Section A.4.1 for the definition of $\eta_{h,i}^{n,m}$. Here, $\eta_{\mathcal{T},i}^{n,m}$ is the contribution on a mesh cell $\mathcal{T}$. The last one is called “summand”, there we integrate over the absolute value of the single contributing terms. Thus,

$$\begin{aligned}\iota_{\mathcal{T},r}^{n,m} : &= \frac{1}{2}\rho\left(\left|w_{kh}\right|\right)\left(\left|\Pi_{k,2h}^{(1,2)} z_{kh}\right|\right)_{|\mathcal{T}}\\ &+\frac{1}{2}\rho^\star\left(\left|w_{kh}\right|, \left|z_{kh}\right|\right)\left(\left|\Pi_{2k,2h}^{(2,2)} w_{kh}\right|\right)_{|\mathcal{T}}.\end{aligned}$$

The decision, which way to take the absolute value is the best one, can only be made based on numerical results, see Section 2.7. We test the four alternatives, compare the results, and choose the alternative, which lead to the best results. The indicators contained in $\tilde{\Theta}_k$ can be used for the adpative refinement in time, if we take the absolute value. The resulting set of refinement indicators is called $\Theta_k := \left\{\tilde{\eta}_k^{i,m}\,\middle|\, m = 1, 2, \ldots, M\right\}$ with the modified refinement indicators $\tilde{\eta}_k^{i,m} \geq 0$.

For the spatial refinement indicators, we have to take some other effects into account. Functions in $V_{2h}^{(2),m}$ are continuous. Consequently, we have to ensure this continuity along edges with hanging nodes. For this end, we have to calculate the function values in the hanging nodes w.r.t. the biquadratic interpolation. However, these values differ from the values determined by the bilinear trial functions. In Figure 2.1.2, this is illustrated. On the patch $A$, which conists of the mesh cells $A_1$, $A_2$, $A_3$ and $A_4$, the quadratic interpolation on the edges $a$, $c$ and $c$, $e$ is determined



(a) Mesh with hanging nodes

(b) Piecewise quadratic interpolation on the edges with hanging nodes of the mesh depicted in (a)

FIGURE 2.1.2. Illustration of $i_{2h}^{(2)}$ on edges with hanging nodes

by the function values in $a$, $b$, and $c$ (Marked by small circles in Figure 2.1.2 (b)). For the hanging nodes $b$ and $d$, we obtain the function values marked with the small diamonds. They differ from the original function values (Marked with a small square), which are determined by the linear trial functions. This approach leads to an accurate estimation of the error, but it causes problems in the definition of the refinement indicators on the fine cells. Let us explain this with the help of Figure 2.1.2 again. If we use the globally continuous interpolation of higher order, we obtain large refinement indicators on the cells $B_1$, $B_4$, $C_1$, and $C_4$. Consequently, these cells are refined during the adaptive refinement process. However, the indicators are large on these cells not necessarily because of the large discretisation error but due to the global continuity of the higher order interpolation. To understand this, let us consider the patch $B$. If we only regard $B$ in the definition of the higher order interpolation, it is determined by the function values in the points $a$, $b$, and $c$. They form an affine linear function and the higher order interpolation includes affine linear functions. Thus, the higher order interpolation is equal to the original function and the difference is zero. But we have learned that the function value in $b$ w.r.t. the quadratic interpolation is determined by the function values in $a$, $c$, and $e$ and is not equal to the original function value. Consequently, we do not obtain the affine linear function as interpolation of higher order but a quadratic function and the difference is not longer zero. This leads to the large refinement indicator and the unnecessary refinements. To circumvent this, we use the higher order interpolation locally on every patch indepent from the other patches, when we calculate the refinement indicators. For the calculation of the error estimation, we use the globally continuous higher order interpolation. By using the local higher order interpolation and taking the absolute value, we obtain the set of refinement indicators for the CM approach

$$\Theta_h^{CM} := \left\{ \tilde{\eta}_{\mathcal{T}}^{n} := \sum_{m=0}^{M} \tilde{\eta}_{\mathcal{T}}^{n,m} \,\middle|\, \mathcal{T} \in \mathbb{T}_h = \mathbb{T}_h^m,\, m = 0, 1, \dots, M \right\},$$

where $\tilde{\eta}_{\mathcal{T}}^{n,m} \geq 0$ are the modified refinement indicators.

In Section 1.4, we have seen that $\eta_h^{n,m}$ linearly depends on $k$. In the set $\tilde{\Theta}_h^{DM}$, we have to compare refinement indicators over all time steps, where the comparison is disturbed by the dependence of $\eta_{\mathcal{T}}^{n,m}$ on $k$. We define the reference time step length $\hat{k} := T/M$ and scale all indicators by the factor $\hat{k}/k_m$ to compensate for this disturbance.

The numerical results in Section 2.7.1 show that the refinement indicators in the set $\tilde{\Theta}_h^{DM}$ do not lead to reasonable adaptive meshes. Despite all presented modifications, we observe that the refined regions of the meshes are fragmented as well as that the meshes rapidly change between two time steps. To further compensate this, we smooth the refinement

**Algorithm 2.1.1** Smoothing of refinement indicators

```
Input: Mesh $\mathbb{T}_h$=$\{\mathcal{T}_j|\, j = 1, 2, \ldots, N_h\}$, number of smoothing iterations
       $S$, set of indicators $\Theta = \{\eta^0_{\mathcal{T}} \,|\, \mathcal{T} \in \mathbb{T}_h\}$
    (1) Set $s = 1$.
    (2) Set $j = 1$.
    (3) Determine
            $$\eta^s_j = \frac{1}{|\mathrm{ad}\,(\mathcal{T}_j)|} \sum_{\mathcal{T}' \in \mathrm{ad}(\mathcal{T}_j)} \eta^{s-1}_{\mathcal{T}'}$$
    (4) Set $j \leftarrow j + 1$. If $j \leq N_h$ go to (3).
    (5) Set $s \leftarrow s + 1$. If $s \leq S$ go to (2).
  Output: Set of smoothed indicators $\tilde{\Theta} = \left\{ \eta^S_j \,\middle|\, j = 1, 2, \ldots, N_h \right\}$
```

indicators in every time step. The set of the adjacent mesh cells of a mesh cell $\mathcal{T} \in \mathbb{T}_h$ is defined by

$$\mathrm{ad}\,(\mathcal{T}) := \left\{ \mathcal{T}' \in \mathbb{T}_h \,\middle|\, \bar{\mathcal{T}} \cap \overline{\mathcal{T}'} \neq \emptyset \right\}.$$

The smoothing algorithm is present in Algorithm 2.1.1. Finally, we obtain the set of refinement indicators

$$\Theta^{DM}_h := \left\{ \tilde{\eta}^{n,m}_{\mathcal{T}} \,\middle|\, \mathcal{T} \in \mathbb{T}^m_h,\, m = 0, 1, \ldots, M \right\},$$

where we apply all mentioned operations to determine $\tilde{\eta}^{n,m}_{\mathcal{T}} \geq 0$. The numerical results presented in Section 2.7 substantiate that the definition of $\tilde{\eta}^{n,m}_{\mathcal{T}}$ lead to appropriate refinement indicators and therewith to reasonable adaptive meshes.

## 2.2. Adaptive Refinement Strategy

The space-time Galerkin method contains the spatial mesh sequence $\mathbb{M}_h$ and the temporal mesh $\mathbb{T}_k$. First, we develop a strategy, called space-time refinement strategy, to decide, whether the spatial mesh sequence $\mathbb{M}_h$, the temporal mesh $\mathbb{T}_k$, or both should be adaptively refined.

The overall aim of the adaptive discretisation is to achieve a desired accuracy, whereas the numerical effort attains its minimum. In view of

**Algorithm 2.2.1** Space-time refinement strategy

```
Input: Error estimates $\eta_h$ and $\eta_k$, equilibration factor $c > 1$
(1) If $|\eta_h| > c\,|\eta_k|$, then refine the spatial mesh sequence
(2) If $|\eta_k| > c\,|\eta_h|$, then refine the temporal mesh
(3) If $c^{-1} \leq \frac{|\eta_h|}{|\eta_k|} \leq c$, then refine the spatial mesh sequence and the temporal mesh
Output: Meshes to be refined
```

**Algorithm 2.2.2** Refinement strategy

```
Input: $\Theta := \{\eta_1, \eta_2, \dots, \eta_R\}$
(1) Determine a permutation $(i_1, i_2, \dots, i_R)$ of $(1, 2, \dots, R)$ such that
        $\eta_{i_1} \geq \eta_{i_2} \geq \dots \geq \eta_{i_R}.$
(2) Solve the optimisation problem
        $r = \arg \min_{1 \leq r \leq R} \mathcal{E}(r) \mathcal{N}(r)^\delta.$
(3) Set $\Theta_r := \{\eta_{i_1}, \eta_{i_2}, \dots, \eta_{i_r}\}$.
Output: $\Theta_r$
```

this aim, the spatial and temporal error should be equal, i.e., speaking in the terms of the error estimator,

$$|\eta_h| \approx |\eta_k|$$

should hold. Thus, the space-time refinement strategy must lead to a discretisation, where the spatial and the temporal error estimators are approximately equal. This property has to be conserved during further refinements. In this thesis, the space-time refinement strategy presented in [**99**] is used, which is simple as well as efficient. It is described in Algorithm 2.2.1. The equilibration factor $c$ measures, when we accept a discretisation to be equilibrated. In practical experiments, $c = 5$ has turned out to be a good choice. Smaller values of $c$ lead to a lot of refinement cycles, which are time consuming. However, larger values of $c$ lead to inefficient discretisations.

Up to now, we have presented a strategy to decide, whether the spatial mesh sequence $\mathbb{M}_h$, the temporal mesh $\mathbb{T}_k$, or both should be adaptively refined. In a second step, we select those spatial mesh cells or those temporal subintervals, repectively, which should be refined. Since the same refinement strategy is applied to the spatial mesh sequence as well as to the temporal mesh, we present it in a general setting. The basis is a set of refinement indicators $\Theta = \{\eta_i \geq 0 | \, i = 1, 2, \dots, R\}$, e.g., $\Theta_k$, $\Theta_h^{CM}$, or $\Theta_h^{DM}$. The refinement indicator $\eta_i$ is connected to a mesh cell $\mathcal{T}_i$. The refinement strategy has to determine a subset

$$\Theta_r := \left\{ \eta_{i_j} \,\middle|\, j = 1, 2, \dots, r \right\} \subseteq \Theta,$$

where the mesh cells $\mathcal{T}_{i_j}$ should be refined. And the set $\Theta_r$ has to be chosen in such a way that a maximum reduction of the error by a minimum additional numerical effort is achieved. Throughout this thesis, we apply the refinement strategy presented in [**97, 98**]. It is very flexible and leads to a uniform refinement, once the error indicators are equilibrated. Assuming some regularity properties, this approach has been justified in [**28**].

We briefly describe the refinement strategy here in order to give a complete overview. It is presented in Algorithm 2.2.2. There, $\mathcal{E}(r)$ is a prediction of the discretisation error, if $r$ cells are refined. It is based on the error estimates $\eta_i$ and on the convergence rate of the underlying discretisation. The computational effort is measured by $\mathcal{N}(r)$. We use the number of degrees of freedom to measure the effort. The convergence rate of the underlying discretisation and the dimension of the discretised domain are expressed by $\delta$. We solve the optimisation problem by testing all values of $r$, which is efficiently implemented and does not need much computing time.

The application of this refinement strategy to the spatial and temporal discretisation is straightforward. In the DM approach, we compare all refinement indicators of the spatial mesh sequence $\mathbb{M}_h$ with each other, not only the indicators of one time step. This leads to a very efficient adaptive refinement, which we will see in the Sections 2.7, 3.4, and 4.4.

## 2.3. Mesh Regularisation

During the adaptive refinement process, unstructured meshes are created. However, we have to ensure several properties of the meshes. The first one is the patch structure of the spatial meshes, which is needed to define the higher order interpolation $i_{2h}^{(2)}$, the filtering operator $\pi$, as well as the discrete Lagrange multipliers in Chapter 3 and 4. We also need a kind of patch structure of the temporal mesh, i.e. it has to contain an even number of time steps, in order to evaluate the higher order interpolation $i_{2k}^{(2)}$. We ensure this directly in the refinement strategy. There, we refine an additional element, if the adaptive refinement led to an odd number of mesh cells. Since spatial hanging nodes may lead to oscillations in the discrete solution, we allow only one hanging node per edge. The same restriction is enforced in static calculations, too. In [**35**], an a priori and a posteriori error analysis for triangulations with spatial hanging nodes of maximum degree one is presented. There, static problems are considered.

In the DM approach, we have to pay special attention to the mesh changes between the time steps. In view of Theorem 1.2.5, Proposition 1.2.7, and Remark 1.2.8, the differences between the meshes of two consecutive time steps should be small. The difference corresponds to the number of hanging nodes in time and to their degree. We do not control the number of hanging nodes. But we ensure that only hanging nodes of degree one in time occur. In the first part of this section, we present the regularisation algorithms and show in the second one that they lead to the desired mesh structure.

### 2.3.1. Regularisation Algorithms.

Before we present the regularisation algorithms, we introduce some notation:

- We denote an *arbitrary spatial mesh* by $\mathbb{T}_h$. We will make additional assumptions on the meshes in the following items. They especially ensure that all meshes are hierarchic structured.
- The operator $\text{ref}\,(\mathcal{T})$ denotes the symmetric and regular *refinement* of a cell $\mathcal{T} \in \mathbb{T}_h$, where we decompose the quadrilateral $\mathcal{T}$ into four new quadrilaterals $\mathcal{T}_N^1, \ldots, \mathcal{T}_N^4$ by bisection of all four edges of $\mathcal{T}$, $\text{ref}\,(\mathcal{T}) := \left\{\mathcal{T}_N^1, \ldots, \mathcal{T}_N^4\right\}$.
- By $\text{refine}\,(\mathbb{T}_h, \Theta)$, we denote the *refinement* of the mesh $\mathbb{T}_h$, where all cells in $\mathcal{T} \in \Theta \subseteq \mathbb{T}_h$ are refined by $\text{ref}\,(\mathcal{T})$,

$$\text{refine}\,(\mathbb{T}_h, \Theta) := (\mathbb{T}_h \backslash \Theta) \cup \left( \bigcup_{\mathcal{T} \in \Theta} \text{ref}\,(\mathcal{T}) \right)$$

- The new cells, which are generated during the refinement $\text{ref}\,(\mathcal{T})$, are collected in the set

$$\text{Ch}\,(\mathcal{T}) := \text{ref}\,(\mathcal{T}) = \left\{\mathcal{T}_N^1, \ldots, \mathcal{T}_N^4\right\}.$$

  They are called the *children of a mesh cell* $\mathcal{T}$. We remark that the children of $\mathcal{T}$ always exist.
- The *children of a mesh* $\text{Ch}\,(\mathbb{T}_h)$ are the set

$$\text{Ch}\,(\mathbb{T}_h) := \bigcup_{\mathcal{T} \in \mathbb{T}_h} \text{Ch}\,(\mathcal{T}).$$

- All spatial meshes are constructed by adaptive refinement of a fixed *initial mesh* $\mathbb{I}_h$, $\mathbb{I}_h = \text{Ch}\,(\mathbb{I}_h^\star)$, where $\mathbb{I}_h^\star$ is a mesh without hanging nodes. Consequently, $\mathbb{I}_h$ has patch structure.
- Let $\mathbb{U}_h^0 := \mathbb{I}_h$ be the initial mesh and define recursively $\mathbb{U}_h^l := \text{Ch}\left(\mathbb{U}_h^{l-1}\right)$ for $l = 1, 2, \ldots$. Furthermore, we set

$$\bar{\mathbb{U}}_h := \bigcup_{l=0}^{\infty} \mathbb{U}_h^l.$$

  For all meshes $\mathbb{T}_h$, $\mathbb{T}_h \subset \bar{\mathbb{U}}_h$ holds.
- If $\mathcal{T} \in \mathbb{U}_h^l$, we say the cell $\mathcal{T}$ has the *refinement level* $\text{RL}\,(\mathcal{T}) := l$. We remark that $\text{RL}\,(\text{P}\,(\mathcal{T})) = \text{RL}\,(\mathcal{T}) - 1$ and $\text{RL}\,(\mathcal{T}') = \text{RL}\,(\mathcal{T}) + 1$ for all $\mathcal{T}' \in \text{Ch}\,(\mathcal{T})$.

- We say a mesh $\mathbb{T}_h$ is of the *refinement level* $l$, if and only if
$$l = \mathrm{RL}\left(\mathbb{T}_h\right) := \max_{\mathcal{T} \in \mathbb{T}_h} \mathrm{RL}\left(\mathcal{T}\right).$$
- Let $\mathcal{T}$ be an arbitrary mesh cell with $\mathrm{RL}(\mathcal{T}) = l$. The *parent cell* of $\mathcal{T}$, $\mathrm{P}\left(\mathcal{T}\right)$, is the unique cell with $\mathrm{RL}\left(\mathrm{P}\left(\mathcal{T}\right)\right) = l - 1$, for which $\mathcal{T} \in \mathrm{Ch}\left(\mathrm{P}\left(\mathcal{T}\right)\right)$ holds.
- We set $\mathrm{Ch}^1\left(\mathcal{T}\right) := \mathrm{Ch}\left(\mathcal{T}\right)$ and recursively define
$$\mathrm{Ch}^p\left(\mathcal{T}\right) := \bigcup_{\mathcal{T}' \in \mathrm{Ch}^{p-1}(\mathcal{T})} \mathrm{Ch}\left(\mathcal{T}'\right)$$
with $p \in \mathbb{N}$, $p > 1$.
- The *set of all children* of a mesh element $\overline{\mathrm{Ch}}\left(\mathcal{T}\right)$ is then
$$\overline{\mathrm{Ch}}\left(\mathcal{T}\right) := \bigcup_{p=1}^{\infty} \mathrm{Ch}^p\left(\mathcal{T}\right).$$
- The *set of all parents* of a mesh element $\mathcal{T}$ is given by
$$\overline{\mathrm{P}}\left(\mathcal{T}\right) := \left\{ \mathcal{T}' \in \mathbb{U}_h^l \middle| \, 0 \leq l < \mathrm{RL}\left(\mathcal{T}\right), \mathcal{T} \in \overline{\mathrm{Ch}}\left(\mathcal{T}'\right) \right\}.$$
- The *refinement history* of a mesh element $\mathcal{T}$ is the set
$$\mathrm{RH}\left(\mathcal{T}\right) := \overline{\mathrm{P}}\left(\mathcal{T}\right) \cup \left\{\mathcal{T}\right\} \cup \overline{\mathrm{Ch}}\left(\mathcal{T}\right).$$
- The *neighbourhood* of a mesh element $\mathcal{T}$ in the mesh $\mathbb{T}_h$ is given by
$$\mathrm{N}(\mathcal{T}, \mathbb{T}_h) := \left\{ \mathcal{T}' \in \mathbb{T}_h \middle| \dim\left(\bar{\mathcal{T}} \cap \overline{\mathcal{T}'}\right) \geq 1 \right\}.$$
- A mesh cell $\mathcal{T} \in \mathbb{T}_h$ has a *spatial hanging node* of degree $d$, if and only if
$$d = \mathrm{hang}(\mathcal{T}, \mathbb{T}_h) := \max\left\{\mathrm{RL}(\mathrm{N}\left(\mathcal{T}\right)) - \mathrm{RL}(\mathcal{T}), 0\right\}.$$
- We say a mesh $\mathbb{T}_h$ has only *spatial hanging nodes* of degree $d$, if all $\mathcal{T} \in \mathbb{T}_h$ have only hanging nodes of degree less or equal $d$.
- The *patch* of a mesh element $\mathcal{T}$ is given by
$$\mathrm{patch}(\mathcal{T}) := \mathrm{Ch}\left(\mathrm{P}\left(\mathcal{T}\right)\right).$$

In particular, it holds $\mathcal{T} \in \operatorname{patch}(\mathcal{T})$.

- A mesh $\mathbb{T}_h$ has *patch structure*, if and only if the relation

$$\operatorname{patch}(\mathcal{T}) \subseteq \mathbb{T}_h$$

holds for all $\mathcal{T} \in \mathbb{T}_h$.

- We denote a *mesh sequence* by $\mathbb{M}_h = (\mathbb{T}_h^m)_{0\leq m\leq M}$. All meshes of this sequence are constructed from the same initial mesh $\mathbb{I}_h$ and are contained in $\bar{\mathbb{U}}_h$.
- The set of the *corresponding elements* $\mathrm{CE}^{\pm}\left(\mathcal{T}, \mathbb{T}_h^{m\pm1}\right)$ of an element $\mathcal{T} \in \mathbb{T}_h^m$ is defined as

$$\mathrm{CE}^{\pm}(\mathcal{T}, \mathbb{T}_h^{m\pm1}) := \left\{\mathcal{T}' \in \mathbb{T}_h^{m\pm1} \middle| \, \mathcal{T}' \in \mathrm{RH}(\mathcal{T})\right\}.$$

- The element $\mathcal{T} \in \mathbb{T}_h^m$ has a *forward (backward) temporal hanging node* of degree $d$, where

$$\begin{aligned} d &:= \operatorname{hang}^{\pm}\left(\mathcal{T}, \mathbb{T}_h^{m\pm1}\right) \\ &:= \max\left\{\mathrm{RL}(\mathrm{CE}^{\pm}(\mathcal{T}, \mathbb{T}_h^{m\pm1})) - \mathrm{RL}(\mathcal{T}), 0\right\}. \end{aligned}$$

- We say a sequence of meshes $\mathbb{M}_h$ has only *temporal hanging nodes* of degree $d$ in time, if all $\mathcal{T} \in \mathbb{T}_h^m$ have only forward and backward hanging nodes of maximum degree $d$ in time for all $0 \leq m \leq M$.
- A mesh sequence $\mathbb{M}_h$ is called *patch regular*, if and only if all meshes $\mathbb{T}_h^m$ in $\mathbb{M}_h$ have only hanging nodes of degree one in space and patch structure and if the sequence $\mathbb{M}_h$ has only hanging nodes of degree one.

With this notation, we are now able to introduce the regularisation algorithms. Here, regularisation means that additional mesh cells are refined to recover the desired mesh structure. Algorithm 2.3.1 presents the regularisation of one spatial mesh $\mathbb{T}_h$. It is directly used in static problems and in the CM approach. Lemma 2.3.1 ensures that we obtain a mesh, which has patch structure and no hanging nodes of degree greater than one, after a finite number of iterations. The algorithm to remove hanging nodes of degree greater than one in time is introduced in Algorithm 2.3.2.

**Algorithm 2.3.1** Spatial regularisation

```
Input: Spatial mesh $\mathbb{T}_h^0$
(1) Build $\Theta^0 = \{\mathcal{T} \in \mathbb{T}_h^0 \,|\mathrm{patch}(\mathcal{T}) \not\subseteq \mathbb{T}_h^0\}$.
(2) Set $\mathbb{T}_h^1 \leftarrow \mathrm{refine}\left(\mathbb{T}_h^0, \Theta^0\right)$, $p = 1$
(3) Build $\tilde{\Theta}^p = \{\mathcal{T} \in \mathbb{T}_h^l \,|\mathrm{hang}(\mathcal{T}, \mathbb{T}_h^p) > 1\}$.
(4) If $\tilde{\Theta}^p = \emptyset$ then $\mathbb{T}_h^\star = \mathbb{T}_h^p$ and STOP.
(5) Set $\Theta^p = \bigcup_{\mathcal{T} \in \tilde{\Theta}^p} \mathrm{patch}(\mathcal{T})$.
(6) Set $\mathbb{T}_h^{p+1} \leftarrow \mathrm{refine}\left(\mathbb{T}_h^p, \Theta^p\right)$.
(7) Set $p \leftarrow p + 1$ and go to (3).
Output: Spatial mesh $\mathbb{T}_h^\star$
```

If we enter an admissible mesh sequence in this algorithm, we end up with a patch regular mesh sequence, see Lemma 2.3.4. The interaction of the Algorithms 2.3.1 and 2.3.2 to realise an adaptive refinement of the mesh sequence $\mathbb{M}_h$ is described in Algorithm 2.3.3. There, we put the cells, which are chosen for refinement, into the set $\Theta_r$. Proposition 2.3.8 ensures that we obtain an adaptively refined patch regular mesh sequence. After the proof of these properties of the regularisation algorithms, we present the adaptive solution algorithm in Section 2.4.

**2.3.2. Analysis of the Regularisation Algorithms.** In this section, we analyse the regularisation Algorithm 2.3.3. As in the description of this algorithm, we split up the analysis into two parts. In the first part, we analyse Algorithm 2.3.1 and in the second one Algorithm 2.3.2. We begin with the following result for Algorithm 2.3.1:

Lemma 2.3.1. *Let $\tilde{\mathbb{T}}_h$ be a mesh of refinement level $l-1$, which has patch structure and only hanging nodes of degree one in space. The mesh $\mathbb{T}_h^0$ is constructed from $\tilde{\mathbb{T}}_h$ by refinement, in particular $\mathcal{T} \in \tilde{\mathbb{T}}_h$ or $\mathrm{P}\left(\mathcal{T}\right) \in \tilde{\mathbb{T}}_h$ holds for all $\mathcal{T} \in \mathbb{T}_h^0$. Then Algorithm 2.3.1 terminates after a maximum of $l-1$ iterations and the output mesh $\mathbb{T}_h^\star$ has patch structure and has only spatial hanging nodes of maximum degree one.*

Remark 2.3.2. The mesh $\mathbb{T}_h^\star$ is of refinement level $l$ or $l-1$. If we adaptively refine $\mathbb{T}_h^\star$, the resulting mesh fulfils the assumptions of Lemma

**Algorithm 2.3.2** Temporal regularisation

```
Input: Mesh sequence $\mathbb{M}_h = \left(\mathbb{T}_h^{m,0}\right)_{0\leq m\leq M}$
(1) Set $\hat{\mathbb{T}}_h^{0,0} := \mathbb{T}_h^{0,0}$ and $m = 1$.
(2) Build the set $\tilde{\Theta}_{f,m}^0 = \left\{\mathcal{T} \in \mathbb{T}_h^{m,0} \middle| \text{hang}^- \left(\mathcal{T}, \hat{\mathbb{T}}_h^{m-1,0}\right) > 1\right\}$.
(3) Set $\Theta_{f,m}^0 = \bigcup_{\mathcal{T}\in\tilde{\Theta}_{f,m}^0} \text{patch}(\mathcal{T})$ .
(4) Set $\mathbb{T}_h^{m,1} \leftarrow \text{refine}\left(\mathbb{T}_h^{m,0}, \Theta_{f,m}^0\right)$ and $p = 1$.
(5) Build $\tilde{\Theta}_{f,m}^p = \left\{\mathcal{T} \in \mathbb{T}_h^{m,p} \middle| \text{hang}\left(\mathcal{T}, \mathbb{T}_h^{m,p}\right) > 1\right\}$.
(6) If $\tilde{\Theta}_{f,m}^p = \emptyset$ then set $\hat{\mathbb{T}}_h^{m,0} = \mathbb{T}_h^{m,p}$ and go to (10).
(7) Set $\Theta_{f,m}^p = \bigcup_{\mathcal{T}\in\tilde{\Theta}_m^p} \text{patch}(\mathcal{T})$.
(8) Set $\mathbb{T}_h^{m,p+1} \leftarrow \text{refine}\left(\mathbb{T}_h^{m,p}, \Theta_{f,m}^p\right)$.
(9) Set $p \leftarrow p + 1$ and go to (5).
(10) If $m < M$, set $m \leftarrow m + 1$ and go to (2)
(11) Set $\mathbb{T}_h^{M,\star} := \hat{\mathbb{T}}_h^M$ and $m = M - 1$
(12) Build the set $\tilde{\Theta}_{b,m}^0 = \left\{\mathcal{T} \in \hat{\mathbb{T}}_h^{m,0} \middle| \text{hang}^+ \left(\mathcal{T}, \mathbb{T}_h^{m+1,\star}\right) > 1\right\}$.
(13) Set $\Theta_{b,m}^0 = \bigcup_{\mathcal{T}\in\tilde{\Theta}_m^0} \text{patch}(\mathcal{T})$ .
(14) Set $\hat{\mathbb{T}}_h^{m,1} \leftarrow \text{refine}\left(\hat{\mathbb{T}}_h^{m,0}, \Theta_{b,m}^0\right)$ and $p = 1$.
(15) Build $\tilde{\Theta}_{b,m}^p = \left\{\mathcal{T} \in \hat{\mathbb{T}}_h^{m,p} \middle| \text{hang}\left(\mathcal{T}, \hat{\mathbb{T}}_h^{m,p}\right) > 1\right\}$.
(16) If $\tilde{\Theta}_{b,m}^p = \emptyset$ then set $\mathbb{T}_h^{m,\star} = \hat{\mathbb{T}}_h^{m,p}$ and go to (20).
(17) Set $\Theta_{b,m}^p = \bigcup_{\mathcal{T}\in\tilde{\Theta}_{b,m}^p} \text{patch}(\mathcal{T})$.
(18) Set $\hat{\mathbb{T}}_h^{m,p+1} \leftarrow \text{refine}\left(\hat{\mathbb{T}}_h^{m,p}, \Theta_{b,m}^p\right)$.
(19) Set $p \leftarrow p + 1$ and go to (15).
(20) If $m > 0$, set $m \leftarrow m - 1$ and go to (12)
Output: Mesh sequence $\mathbb{M}_h^\star = \left(\mathbb{T}_h^{m,\star}\right)_{0\leq m\leq M}$
```

2.3.1. Consequently, the assumptions of Lemma 2.3.1 hold during an adaptive refinement process consisting of several adaptive refinements, if we apply Algorithm 2.3.1 after each adaptive refinement and if the initial mesh has patch structure and only hanging nodes of degree one in space.

**Algorithm 2.3.3** Adaptive spatial refinement algorithm with regularisation

```
Input: Mesh sequence $\mathbb{M}_h = (\mathbb{T}_h^m)_{0\leq m\leq M}$ and refinement indicators
       $\Theta_r = (\Theta_r^m)_{0\leq m\leq M}$
(1) Set $m = 0$.
(2) Set $\hat{\mathbb{T}}_h^m \leftarrow \mathrm{refine}\,(\mathbb{T}_h^m, \Theta_r^m)$.
(3) Regularise $\hat{\mathbb{T}}_h^m$ in space by Algorithm 2.3.1 and call the result
    $\tilde{\mathbb{T}}_h^m$.
(4) If $m < M$, set $m \leftarrow m+1$ and go to (2).
(5) Regularise the mesh sequence $\tilde{\mathbb{M}}_h = \left(\tilde{\mathbb{T}}_h^m\right)_{0\leq m\leq M}$ by Algorithm
    2.3.2 and call the result $\mathbb{M}_h^\star$.
Output: Mesh sequence $\mathbb{M}_h^\star = \left(\mathbb{T}_h^{m,\star}\right)_{0\leq m\leq M}$
```

Proof of Lemma 2.3.1. We define the sets

$$\begin{aligned}\tilde{\mathbb{T}}_R^p &:= \left\{\mathcal{T}\in\tilde{\mathbb{T}}_h \,\middle|\, \mathrm{Ch}\,(\mathcal{T}) \subseteq \mathbb{T}_h^p\right\},\\ \tilde{\mathbb{T}}_N^p &:= \left\{\mathcal{T}\in\tilde{\mathbb{T}}_h \,\middle|\, \mathcal{T} \in \mathbb{T}_h^p\right\}.\end{aligned}$$

It holds $\tilde{\mathbb{T}}_h = \tilde{\mathbb{T}}_R^0 \cup \tilde{\mathbb{T}}_N^0$, $\tilde{\mathbb{T}}_N^0 \cap \tilde{\mathbb{T}}_R^0 = \emptyset$, $\mathbb{T}_h^0 = \tilde{\mathbb{T}}_N^0 \cup \mathrm{Ch}\left(\tilde{\mathbb{T}}_R^0\right)$, and $\tilde{\mathbb{T}}_N^0 \cap \mathrm{Ch}\left(\tilde{\mathbb{T}}_R^0\right) = \emptyset$.

In the first step, we prove that $\mathbb{T}_h^1$ has patch structure. The set $\Theta^0$ is a subset of $\tilde{\mathbb{T}}_N^0$, since $\mathrm{patch}\,(\mathcal{T}) \subseteq \mathbb{T}_h^0$ for all $\mathcal{T} \in \mathrm{Ch}\left(\tilde{\mathbb{T}}_R^0\right)$. Consequently, we have $\tilde{\mathbb{T}}_N^1 = \tilde{\mathbb{T}}_N^0 \backslash \Theta^0$ and $\tilde{\mathbb{T}}_R^1 = \tilde{\mathbb{T}}_R^0 \cup \Theta^0$. Furthermore, it holds $\tilde{\mathbb{T}}_h = \tilde{\mathbb{T}}_R^1 \cup \tilde{\mathbb{T}}_N^1$, $\tilde{\mathbb{T}}_N^1 \cap \tilde{\mathbb{T}}_R^1 = \emptyset$, $\mathbb{T}_h^1 = \tilde{\mathbb{T}}_N^1 \cup \mathrm{Ch}\left(\tilde{\mathbb{T}}_R^1\right)$, and $\tilde{\mathbb{T}}_N^1 \cap \mathrm{Ch}\left(\tilde{\mathbb{T}}_R^1\right) = \emptyset$ again. The mesh $\mathbb{T}_h^1$ has patch structure, since $\mathrm{patch}\,(\mathcal{T}) \subseteq \mathbb{T}_h^1$ for all $\mathcal{T} \in \mathrm{Ch}\left(\tilde{\mathbb{T}}_R^1\right)$ by construction and for all $\mathcal{T} \in \tilde{\mathbb{T}}_N^1$ by definition. From the definition of the hanging nodes in space and of the set $\tilde{\Theta}$, as well as from $\mathrm{RL}\left(\mathbb{T}_h^1\right) \leq l$, we conclude $\mathrm{RL}\left(\tilde{\Theta}^1\right) \leq l-2$ .

The second step is to prove that $\mathrm{RL}\left(\tilde{\Theta}^2\right) \leq l-3$. Then we iterate the argument $l-2$ times and obtain $\mathrm{RL}\left(\tilde{\Theta}^l\right) < 0$, which is impossible by the definition of the refinement level. Consequently, $\tilde{\Theta}^l = \emptyset$ and Algorithm

2.3.1 terminates. If $\tilde{\Theta}^p = \emptyset$ for one $1 \leq p < l$, Algorithm 2.3.1 terminates in the $p^{\text{th}}$ iteration. The mesh $\mathbb{T}_h^\star$ has patch structure, since all $\mathbb{T}_h^p$, $p \geq 1$, have patch structure. This claim is shown during the proof of $\mathrm{RL}\left(\tilde{\Theta}^2\right) \leq l-3$. Furthermore, it does not contain any hanging nodes in space of degree two by definition of the stopping criterion.

To complete the proof, we show $\mathrm{RL}\left(\tilde{\Theta}^2\right) \leq l-3$. First of all, we prove

$$\forall \mathcal{T} \in \mathrm{Ch}\left(\tilde{\mathbb{T}}_R^1\right): \quad 0 \leq \mathrm{hang}\left(\mathcal{T}, \tilde{\mathbb{T}}_h\right) \leq 1. \tag{2.3.1}$$

Let $\mathcal{T}$ be an arbitrary element of $\mathrm{Ch}\left(\tilde{\mathbb{T}}_R^1\right)$. Then $\mathrm{P}\left(\mathcal{T}\right) \in \tilde{\mathbb{T}}_h$. Furthermore, we have

$$0 \leq \mathrm{hang}\left(\mathrm{P}\left(\mathcal{T}\right), \tilde{\mathbb{T}}_h\right) \leq 1, \tag{2.3.2}$$

since $\tilde{\mathbb{T}}_h$ contains only spatial hanging nodes of degree one. We define the sets

$$\begin{aligned}
\mathrm{I} &:= \left\{\mathcal{T}' \in \mathrm{N}\left(\mathcal{T}, \mathbb{T}_h^1\right) \middle| \, \mathcal{T}' \in \tilde{\mathbb{T}}_h\right\}, \\
\mathrm{II} &:= \left\{\mathcal{T}' \in \mathrm{N}\left(\mathcal{T}, \mathbb{T}_h^1\right) \middle| \, \mathrm{P}\left(\mathcal{T}'\right) \in \tilde{\mathbb{T}}_h \text{ and } \mathrm{P}\left(\mathcal{T}'\right) = \mathrm{P}\left(\mathcal{T}\right)\right\}, \\
\mathrm{III} &:= \left\{\mathcal{T}' \in \mathrm{N}\left(\mathcal{T}, \mathbb{T}_h^1\right) \middle| \, \mathrm{P}\left(\mathcal{T}'\right) \in \tilde{\mathbb{T}}_h \text{ and } \mathrm{P}\left(\mathcal{T}'\right) \neq \mathrm{P}\left(\mathcal{T}\right)\right\}.
\end{aligned}$$

By construction, it holds

$$\mathrm{N}\left(\mathcal{T}, \mathbb{T}_h^1\right) = \mathrm{I} \cup \mathrm{II} \cup \mathrm{III}.$$

From $\mathcal{T}' \in \mathrm{I}$, we conclude $\mathcal{T}' \in \mathrm{N}\left(\mathrm{P}\left(\mathcal{T}\right), \tilde{\mathbb{T}}_h\right)$. Inequality (2.3.2) yields

$$\mathrm{RL}\left(\mathcal{T}'\right) \leq \mathrm{RL}\left(\mathrm{P}\left(\mathcal{T}\right)\right) + 1 = \mathrm{RL}\left(\mathcal{T}\right). \tag{2.3.3}$$

For $\mathcal{T}' \in \mathrm{II}$, it holds

$$\mathrm{RL}\left(\mathcal{T}'\right) = \mathrm{RL}\left(\mathcal{T}\right) \tag{2.3.4}$$

by the definition of II. Let $\mathcal{T}'$ be an element of III. Thus,

$$\mathrm{P}\left(\mathcal{T}'\right) \in \mathrm{N}\left(\mathrm{P}\left(\mathcal{T}\right), \tilde{\mathbb{T}}_h\right)$$

holds. Then inequality (2.3.2) implies

$$\mathrm{RL}\left(\mathcal{T}'\right) - 1 = \mathrm{RL}\left(\mathrm{P}\left(\mathcal{T}'\right)\right) \leq \mathrm{RL}\left(\mathrm{P}\left(\mathcal{T}\right)\right) + 1 = \mathrm{RL}\left(\mathcal{T}\right). \tag{2.3.5}$$

The relations (2.3.3-2.3.5) yield (2.3.1).

From (2.3.1), we conclude

$$\tilde{\Theta}^1 \subseteq \tilde{\mathbb{T}}_N^1 \tag{2.3.6}$$

and show now

$$\tilde{\Theta}^1 = \left\{ \mathcal{T} \in \tilde{\mathbb{T}}_N^1 \middle| \operatorname{hang}\left(\mathcal{T}, \mathbb{T}_h^1\right) = 2 \right\}. \tag{2.3.7}$$

Let $\mathcal{T}$ be an arbitrary element of $\tilde{\mathbb{T}}_N^1$ and $\mathcal{T}' \in \mathrm{N}\left(\mathcal{T}, \mathbb{T}_h^1\right)$. Since $\mathbb{T}_h^1 = \tilde{\mathbb{T}}_N^1 \cup \mathrm{Ch}\left(\tilde{\mathbb{T}}_R^1\right)$ and $\tilde{\mathbb{T}}_N^1 \cap \mathrm{Ch}\left(\tilde{\mathbb{T}}_R^1\right) = \emptyset$, $\mathcal{T}'$ is either an element of $\tilde{\mathbb{T}}_N^1$ or $\mathrm{Ch}\left(\tilde{\mathbb{T}}_R^1\right)$. We begin with the case $\mathcal{T}' \in \tilde{\mathbb{T}}_N^1$. Then $\mathcal{T}' \in \mathrm{N}\left(\mathcal{T}, \tilde{\mathbb{T}}_h\right)$ holds. Since $\tilde{\mathbb{T}}_h$ contains only hanging nodes in space of degree one, we obtain $\mathrm{RL}\left(\mathcal{T}'\right) \leq \mathrm{RL}\left(\mathcal{T}\right) + 1$. In the other case $\mathcal{T}' \in \mathrm{Ch}\left(\tilde{\mathbb{T}}_R^1\right)$, we have $\mathrm{P}\left(\mathcal{T}'\right) \in \mathrm{N}\left(\mathcal{T}, \tilde{\mathbb{T}}_h\right)$. From (2.3.2), we deduce

$$\mathrm{RL}\left(\mathcal{T}'\right) = \mathrm{RL}\left(\mathrm{P}\left(\mathcal{T}'\right)\right) + 1 \leq \mathrm{RL}\left(\mathcal{T}\right) + 2.$$

Altogether, we obtain $\operatorname{hang}\left(\mathcal{T}, \mathbb{T}_h^1\right) \leq 2$ for all $\mathcal{T} \in \tilde{\mathbb{T}}_N^1$, i.e. only spatial hanging nodes of degree two occur. Therewith, (2.3.7) becomes obvious.

In step (5) of Algorithm 2.3.1, the set $\tilde{\Theta}^1$ is extended to the set $\Theta^1$. For an arbitrary element $\mathcal{T} \in \tilde{\mathbb{T}}_N^1$, we have $\mathcal{T} \in \tilde{\mathbb{T}}_h \cap \mathbb{T}_h^1$. Since the meshes $\tilde{\mathbb{T}}_h$ and $\mathbb{T}_h^1$ have patch structure, we have $\operatorname{patch}\left(\mathcal{T}\right) \subseteq \tilde{\mathbb{T}}_h$ and $\operatorname{patch}\left(\mathcal{T}\right) \subseteq \mathbb{T}_h^1$, i.e. $\operatorname{patch}\left(\mathcal{T}\right) \subseteq \tilde{\mathbb{T}}_h \cap \mathbb{T}_h^1 = \tilde{\mathbb{T}}_N^1$. Thus, $\Theta^1 \subseteq \tilde{\mathbb{T}}_N^1$. It holds $\mathrm{RL}\left(\Theta^1\right) = \mathrm{RL}\left(\tilde{\Theta}^1\right)$ by definition.

The refinement of $\mathbb{T}_h^1$ w.r.t. $\Theta^1$ leads to the mesh $\mathbb{T}_h^2$ with $\tilde{\mathbb{T}}_R^2 = \tilde{\mathbb{T}}_R^1 \cup \Theta^1$ and $\tilde{\mathbb{T}}_N^2 = \tilde{\mathbb{T}}_N^1 \backslash \Theta^1$. To ensure the patch structure of $\mathbb{T}_h^2$, we have to show that $\tilde{\mathbb{T}}_N^2$ has patch structure. Since $\tilde{\mathbb{T}}_N^1$ has patch structure, the negation of $\tilde{\mathbb{T}}_N^2$ having patch structure is that there exists an elment $\mathcal{T} \in \tilde{\mathbb{T}}_N^2$ with $\operatorname{patch}\left(\mathcal{T}\right) \cap \Theta^1 \neq \emptyset$. Let us assume that such an element $\mathcal{T} \in \tilde{\mathbb{T}}_N^2$ exists. Then an element $\mathcal{T}' \in \operatorname{patch}\left(\mathcal{T}\right) \cap \Theta^1$ exists, too. Because of the patch structure of $\Theta^1$, we have $\operatorname{patch}\left(\mathcal{T}'\right) \subseteq \Theta^1$. This yields

$$\operatorname{patch}\left(\mathcal{T}'\right) \cap \tilde{\mathbb{T}}_N^2 = \emptyset.$$

Hence, $\mathcal{T} \notin \operatorname{patch}(\mathcal{T}')$. This is a contradiction to the fact that $\bar{\mathcal{T}} \in \operatorname{patch}\left(\bar{\mathcal{T}}'\right)$ holds for all $\bar{\mathcal{T}}' \in \operatorname{patch}\left(\bar{\mathcal{T}}\right)$. Consequently, the meshes $\tilde{\mathbb{T}}_N^2$ and $\mathbb{T}_h^2$ have patch structure.

As for $\tilde{\Theta}^1$, we can show that

$$\tilde{\Theta}^2 \subseteq \tilde{\mathbb{T}}_N^2, \tag{2.3.8}$$

i.e. in particular

$$\tilde{\Theta}^2 \cap \Theta^1 = \tilde{\Theta}^2 \cap \tilde{\Theta}^1 = \emptyset.$$

For arbitrary $\mathcal{T} \in \tilde{\Theta}^2$ and $\mathcal{T}' \in \operatorname{patch}(\mathcal{T})$, this yields $\mathcal{T}' \notin \Theta^1$, i.e. $\operatorname{hang}\left(\mathcal{T}', \mathbb{T}_h^1\right) \leq 1$. Consequently, it holds

$$\mathrm{RL}\left(\mathrm{N}\left(\mathcal{T}', \mathbb{T}_h^1\right)\right) \leq \mathrm{RL}\left(\mathcal{T}'\right) + 1. \tag{2.3.9}$$

Inequality (2.3.9) holds for all $\mathcal{T}' \in \operatorname{patch}(\mathcal{T})$, in particular for $\mathcal{T} \in \tilde{\Theta}^2$. Since $\mathrm{RL}\left(\tilde{\Theta}^2\right) \leq l-2$ holds by definition of $\tilde{\Theta}^2$ and of hanging nodes in space, we obtain

$$\mathrm{RL}\left(\mathrm{N}\left(\mathcal{T}, \mathbb{T}_h^1\right)\right) \leq l-1. \tag{2.3.10}$$

Since inequality (2.3.10) holds for all $\mathcal{T} \in \tilde{\Theta}^2$, it is equivalent to

$$\mathrm{RL}\left(\tilde{\Theta}^2\right) \leq l-3$$

and we have finished the proof. □

During the proof of Lemma 2.3.1, we have shown $\tilde{\theta}^p \subset \tilde{\mathbb{T}}_N^p$, see (2.3.6) and (2.3.8), from which the following corollary follows. It provides an important observation for the forthcoming analysis.

COROLLARY 2.3.3. *It holds $\mathcal{T} \in \tilde{\mathbb{T}}_h$ or $\mathrm{P}(\mathcal{T}) \in \tilde{\mathbb{T}}_h$ for all $\mathcal{T} \in \mathbb{T}_h^\star$, i.e. during the adaptive refinement and the regularisation by Algorithm 2.3.1, a mesh cell is refined only once.*

The following result holds for Algorithm 2.3.2:

LEMMA 2.3.4. *Let*

$$\tilde{\mathbb{M}}_h = \left(\tilde{\mathbb{T}}_h^m\right)_{0 \leq m \leq M}$$

*be a patch regular mesh sequence of refinement level* $l-1$. *The input mesh sequence*

$$\mathbb{M}_h = \left(\mathbb{T}_h^{m,0}\right)_{0\le m\le M}$$

*in Algorithm 2.3.2 results from a refinement of* $\tilde{\mathbb{M}}_h$ *and all meshes* $\mathbb{T}_h^{m,0}$ *have patch structure and only spatial hanging nodes of degree one, in particular* $\mathcal{T} \in \tilde{\mathbb{T}}_h^m$ *or* $\mathrm{P}\left(\mathcal{T}\right) \in \tilde{\mathbb{T}}_h^m$ *holds for all* $\mathcal{T} \in \mathbb{T}_h^{m,0}$ *and all* $0 \le m \le M$. *Then the output mesh sequence*

$$\mathbb{M}_h^\star = \left(\mathbb{T}_h^{m,\star}\right)_{0\le m\le M}$$

*is patch regular.*

In the proof of Lemma 2.3.4, we need the following result:

Lemma 2.3.5. *Under the assumptions of Lemma 2.3.4, the mesh sequence*

$$\hat{\mathbb{M}}_h = \left(\hat{\mathbb{T}}_h^{m,0}\right)_{0\le m\le M}$$

*contains only backward hanging nodes of degree one. Furthermore, all* $\hat{\mathbb{T}}_h^{m,0}$, $0 \le m \le M$, *have patch structure and only spatial hanging nodes of degree one. It holds* $\mathcal{T} \in \tilde{\mathbb{T}}_h^m$ *or* $\mathrm{P}\left(\mathcal{T}\right) \in \tilde{\mathbb{T}}_h^m$ *for all* $\mathcal{T} \in \hat{\mathbb{T}}_h^{m,0}$ *in each time step* $m$, $0 \le m \le M$.

Proof of Lemma 2.3.5. The mesh $\hat{\mathbb{T}}_h^{0,0}$ has patch structure and only spatial hanging nodes of degree one by definition. Furthermore, it cannot contain backward hanging nodes in time and $\mathcal{T} \in \tilde{\mathbb{T}}_h^0$ or $\mathrm{P}\left(\mathcal{T}\right) \in \tilde{\mathbb{T}}_h^0$ holds for all $\mathcal{T} \in \hat{\mathbb{T}}_h^{0,0}$. We assume that $\hat{\mathbb{T}}_h^{m-1,0}$ has patch structure, only spatial hanging nodes of degree one, and only backward hanging nodes of degree one and that

$$\forall \mathcal{T} \in \hat{\mathbb{T}}_h^{m-1,0} : \quad \mathcal{T} \in \tilde{\mathbb{T}}_h^{m-1} \text{ or } \mathrm{P}\left(\mathcal{T}\right) \in \tilde{\mathbb{T}}_h^{m-1} \tag{2.3.11}$$

holds. If we show that $\hat{\mathbb{T}}_h^{m,0}$ has the same properties, the claim of the lemma holds by induction.

We define the sets

$$\begin{aligned}\tilde{\mathbb{T}}_R^m &:= \left\{\mathcal{T} \in \tilde{\mathbb{T}}_h^m \middle| \mathrm{Ch}\left(\mathcal{T}\right) \subseteq \mathbb{T}_h^{m,0}\right\},\\ \tilde{\mathbb{T}}_N^m &:= \left\{\mathcal{T} \in \tilde{\mathbb{T}}_h^m \middle| \mathcal{T} \in \mathbb{T}_h^{m,0}\right\}.\end{aligned}$$

It holds $\mathbb{T}_h^{m,0} = \tilde{\mathbb{T}}_N^m \cup \mathrm{Ch}\left(\tilde{\mathbb{T}}_R^m\right)$, $\tilde{\mathbb{T}}_N^m \cap \mathrm{Ch}\left(\tilde{\mathbb{T}}_R^m\right) = \emptyset$, $\tilde{\mathbb{T}}_h^m = \tilde{\mathbb{T}}_N^m \cup \tilde{\mathbb{T}}_R^m$, and $\tilde{\mathbb{T}}_N^m \cap \tilde{\mathbb{T}}_R^m = \emptyset$. Let $\mathcal{T}$ be an arbitrary element of $\mathbb{T}_h^{m,0} \cap \tilde{\mathbb{T}}_N^m$. Since $\tilde{\mathbb{M}}_h$ contains only backward hanging nodes of degree one, we know

$$\mathrm{RL}\left(\mathcal{T}\right) + 1 \geq \mathrm{RL}\left(\mathrm{CE}^-\left(\mathcal{T}, \tilde{\mathbb{T}}_h^{m-1}\right)\right). \tag{2.3.12}$$

From (2.3.11), we deduce

$$\mathrm{RL}\left(\mathrm{CE}^-\left(\mathcal{T}, \tilde{\mathbb{T}}_h^{m-1}\right)\right) + 1 \geq \mathrm{RL}\left(\mathrm{CE}^-\left(\mathcal{T}, \hat{\mathbb{T}}_h^{m-1,0}\right)\right). \tag{2.3.13}$$

The inequalities (2.3.12) and (2.3.13) yield

$$\forall \mathcal{T} \in \mathbb{T}_h^{m,0} \cap \tilde{\mathbb{T}}_N^m : \quad \mathrm{hang}^-\left(\mathcal{T}, \mathbb{T}_h^{m,0}, \hat{\mathbb{T}}_h^{m-1,0}\right) \leq 2. \tag{2.3.14}$$

For $\mathcal{T} \in \mathbb{T}_h^{m,0} \cap \mathrm{Ch}\left(\tilde{\mathbb{T}}_R^m\right)$, we have by similar arguments

$$\begin{aligned}\mathrm{RL}\left(\mathcal{T}\right) + 1 &= \mathrm{RL}\left(\mathrm{P}\left(\mathcal{T}\right)\right) + 2\\ &\geq \mathrm{RL}\left(\mathrm{CE}^-\left(\mathrm{P}\left(\mathcal{T}\right), \tilde{\mathbb{T}}_h^{m-1}\right)\right) + 1\\ &\geq \mathrm{RL}\left(\mathrm{CE}^-\left(\mathrm{P}\left(\mathcal{T}\right), \hat{\mathbb{T}}_h^{m-1,0}\right)\right)\\ &\geq \mathrm{RL}\left(\mathrm{CE}^-\left(\mathcal{T}, \hat{\mathbb{T}}_h^{m-1,0}\right)\right).\end{aligned}$$

Here, we have used

$$\mathrm{CE}^-\left(\mathcal{T}, \hat{\mathbb{T}}_h^{m-1,0}\right) \subseteq \mathrm{CE}^-\left(\mathrm{P}\left(\mathcal{T}\right), \hat{\mathbb{T}}_h^{m-1,0}\right).$$

Consequently, it holds

$$\forall \mathcal{T} \in \mathbb{T}_h^{m,0} \cap \mathrm{Ch}\left(\tilde{\mathbb{T}}_R^m\right) : \quad \mathrm{hang}^-\left(\mathcal{T}, \hat{\mathbb{T}}_h^{m-1,0}\right) \leq 1. \tag{2.3.15}$$

The inequalities (2.3.14) and (2.3.15) lead to

$$\tilde{\Theta}_{f,m}^0 = \left\{\mathcal{T} \in \mathbb{T}_h^{m,0} \middle| \mathrm{hang}^-\left(\mathcal{T}, \hat{\mathbb{T}}_h^{m-1,0}\right) = 2\right\} \subseteq \tilde{\mathbb{T}}_N^m. \tag{2.3.16}$$

We have seen in the proof of Lemma 2.3.1 that the mesh $\tilde{\mathbb{T}}_N^m$ has patch structure. Thus, the set $\Theta_{f,m}^0$ is a subset of $\tilde{\mathbb{T}}_N^m$, too.

Since we have to regularise only backward hanging nodes of degree two, see (2.3.16), and because $\Theta_{f,m}^0 \subseteq \tilde{\mathbb{T}}_N^m$, the mesh $\mathbb{T}_h^{m,1}$ contains only backward temporal hanging nodes of degree one and $\mathcal{T} \in \tilde{\mathbb{T}}_h^m$ or $\mathrm{P}(\mathcal{T}) \in \tilde{\mathbb{T}}_h^m$ holds for all $\mathcal{T} \in \mathbb{T}_h^{m,1}$. Furthermore, the mesh $\mathbb{T}_h^{m,1}$ has patch structure by construction of $\Theta_{f,m}^0$. We have already discussed this in the proof of Lemma 2.3.1. Due to the refinement in step (4) of Algorithm 2.3.2, additional forward temporal hanging nodes in $\hat{\mathbb{T}}_h^{m-1,0}$ and additional backward temporal hanging nodes in $\mathbb{T}_h^{m+1,0}$ may be created.

The removal of the spatial double hanging nodes in the steps (5)-(9) of Algorithm 2.3.2 is carried out in the same manner as in Algorithm 2.3.1. We pass on the ensuring of the patch structure, because the mesh $\mathbb{T}_h^{m,1}$ has already patch structure. Furthermore, the mesh $\mathbb{T}_h^{m,1}$ fulfills the assumptions of Lemma 2.3.1. We apply Lemma 2.3.1 and obtain that $\hat{\mathbb{T}}_h^{m,0}$ contains only spatial hanging nodes of degree one and has patch structure. Corollary 2.3.3 ensures $\mathcal{T} \in \tilde{\mathbb{T}}_h^m$ or $\mathrm{P}(\mathcal{T}) \in \tilde{\mathbb{T}}_h^m$ for all $\mathcal{T} \in \hat{\mathbb{T}}_h^{m,0}$. During the spatial regularisation of $\mathbb{T}_h^{m,1}$, we can only create temporal forward hanging nodes of degree two in $\hat{\mathbb{T}}_h^{m-1,0}$ and backward temporal hanging nodes of degree two in $\mathbb{T}_h^{m+1,0}$. But it is impossible to create any additional backward hanging nodes of degree two in $\hat{\mathbb{T}}_h^{m,0}$. Consequently, the proof is complete. □

By the same arguments as in the proof of Lemma 2.3.5, we obtain the following result concerning forward hanging nodes in time:

LEMMA 2.3.6. *Suppose that all meshes $\hat{\mathbb{T}}_h^{m,0}$, $0 \le m \le M$, of $\hat{\mathbb{M}}_h$ have patch structure and only spatial hanging nodes of degree one. Suppose further $\mathcal{T} \in \tilde{\mathbb{T}}_h^m$ or $\mathrm{P}(\mathcal{T}) \in \tilde{\mathbb{T}}_h^m$ for all $\mathcal{T} \in \hat{\mathbb{T}}_h^{m,0}$ and all $0 \le m \le M$. Then the mesh sequence*

$$\mathbb{M}_h^\star = \left(\mathbb{T}_h^{m,\star}\right)_{0 \le m \le M}$$

*contains only forward hanging nodes of degree one. Furthermore, all* $\mathbb{T}_h^{m,\star}$*,* $0 \leq m \leq M$*, have patch structure and only spatial hanging nodes of degree one.*

The following result describes the situation around a forward hanging node of degree two:

LEMMA 2.3.7. *Let* $0 \leq m < M$ *and*

$$\mathcal{T} \in \bigcup_p \Theta_{b,m}^p \subseteq \hat{\mathbb{T}}_h^{m,0}.$$

*Then*

$$\forall \mathcal{K} \in \mathrm{CE}^+\left(\mathcal{T}, \mathbb{T}_h^{m+1,\star}\right): \quad \mathrm{RL}\left(\mathcal{K}\right) \geq \mathrm{RL}\left(\mathcal{T}\right) + 1 \tag{2.3.17}$$

*holds.*

PROOF OF LEMMA 2.3.7. The first case is $\mathcal{T} \in \Theta_{b,m}^0$. By definition, there exists a mesh cell $\mathcal{T}' \in \mathrm{patch}\left(\mathcal{T}\right)$ with

$$\mathrm{hang}^+\left(\mathcal{T}', \mathbb{T}_h^{m+1,\star}\right) = 2. \tag{2.3.18}$$

The condition $\mathcal{T}' \in \mathrm{patch}\left(\mathcal{T}\right)$ implies $\mathrm{RL}\left(\mathcal{T}'\right) = \mathrm{RL}\left(\mathcal{T}\right)$ and $\mathrm{P}\left(\mathcal{T}'\right) = \mathrm{P}\left(\mathcal{T}\right)$. Because of the definition of the corresponding elements, we have

$$\mathrm{CE}^+\left(\mathcal{T}, \mathbb{T}_h^{m+1,\star}\right) \subseteq \mathrm{CE}^+\left(\mathrm{P}\left(\mathcal{T}\right), \mathbb{T}_h^{m+1,\star}\right) \tag{2.3.19}$$

and

$$\mathrm{CE}^+\left(\mathcal{T}', \mathbb{T}_h^{m+1,\star}\right) \subseteq \mathrm{CE}^+\left(\mathrm{P}\left(\mathcal{T}\right), \mathbb{T}_h^{m+1,\star}\right). \tag{2.3.20}$$

From (2.3.18) and (2.3.20),

$$\mathrm{RL}\left(\mathrm{CE}^+\left(\mathrm{P}\left(\mathcal{T}\right), \mathbb{T}_h^{m+1,\star}\right)\right) \geq \mathrm{RL}\left(\mathcal{T}\right) + 2$$

follows. This implies

$$\mathrm{CE}^+\left(\mathrm{P}\left(\mathcal{T}\right), \mathbb{T}_h^{m+1,\star}\right) \subset \overline{\mathrm{Ch}}\left(\mathrm{P}\left(\mathcal{T}\right)\right)$$

and

$$\forall \mathcal{K} \in \mathrm{CE}^+\left(\mathrm{P}\left(\mathcal{T}\right), \mathbb{T}_h^{m+1,\star}\right): \quad \mathrm{RL}\left(\mathcal{K}\right) \geq \mathrm{RL}\left(\mathcal{T}\right).$$

Since $\mathbb{T}_h^{m+1,\star}$ has patch structure, $\mathrm{CE}^+\left(\mathrm{P}\left(\mathcal{T}\right), \mathbb{T}_h^{m+1,\star}\right)$ has patch structure, too. Thus,

$$\mathrm{CE}^+\left(\mathrm{P}\left(\mathcal{T}\right), \mathbb{T}_h^{m+1,\star}\right) = \mathrm{patch}\left(\mathcal{T}\right)$$

holds, if

$$\mathrm{CE}^+\left(\mathrm{P}\left(\mathcal{T}\right), \mathbb{T}_h^{m+1,\star}\right) \cap \mathrm{patch}\left(\mathcal{T}\right) \neq \emptyset.$$

Together with (2.3.18), this implies

$$\forall \mathcal{K} \in \mathrm{CE}^+\left(\mathrm{P}\left(\mathcal{T}\right), \mathbb{T}_h^{m+1,\star}\right): \quad \mathrm{RL}\left(\mathcal{K}\right) \geq \mathrm{RL}\left(\mathcal{T}\right) + 1.$$

Finally, (2.3.19) yields the assertion (2.3.17) for $\mathcal{T} \in \Theta_{b,m}^0$.

Now, we consider $\mathcal{T} \in \Theta_{b,m}^1$. Then, a mesh element

$$\mathcal{T}' \in \mathrm{patch}\left(\mathcal{T}\right) \subset \hat{\mathbb{T}}_h^{m,1}$$

exists, where

$$\mathcal{T}' \in \mathrm{N}\left(\tilde{\mathcal{T}}, \hat{\mathbb{T}}_h^{m,1}\right)$$

holds with $\tilde{\mathcal{T}} \in \hat{\mathbb{T}}_h^{m,1}$, $\mathrm{P}\left(\tilde{\mathcal{T}}\right) \in \Theta_{b,m}^0$, and

$$\mathrm{RL}\left(\mathcal{T}'\right) = \mathrm{RL}\left(\mathcal{T}\right) = \mathrm{RL}\left(\tilde{\mathcal{T}}\right) - 2.$$

Since $\mathrm{P}\left(\tilde{\mathcal{T}}\right) \in \Theta_{b,m}^0$, we know

$$\forall \mathcal{K}' \in \mathrm{CE}^+\left(\tilde{\mathcal{T}}, \mathbb{T}_h^{m+1,\star}\right): \quad \mathrm{RL}\left(\mathcal{K}'\right) \geq \mathrm{RL}\left(\tilde{\mathcal{T}}\right).$$

The mesh $\mathbb{T}_h^{m+1,\star}$ contains only spatial hanging nodes of degree one, i.e.

$$\forall \tilde{\mathcal{K}} \in \mathrm{N}\left(\mathcal{K}', \mathbb{T}_h^{m+1,\star}\right): \quad \mathrm{RL}\left(\tilde{\mathcal{K}}\right) \geq \mathrm{RL}\left(\mathcal{K}'\right) - 1$$

holds. Furthermore, it holds

$$\mathrm{CE}^+\left(\mathrm{P}\left(\mathcal{T}\right), \mathbb{T}_h^{m+1,\star}\right) \cap \bigcup_{\mathcal{K}'} \mathrm{N}\left(\mathcal{K}', \mathbb{T}_h^{m+1,\star}\right) \neq \emptyset.$$

This yields

$$\begin{aligned}\mathrm{RL}\left(\mathrm{CE}^{+}\left(\mathrm{P}\left(\mathcal{T}\right), \mathbb{T}_h^{m+1,\star}\right)\right) &\geq \mathrm{RL}\left(\mathcal{K}'\right) - 1\\ &\geq \mathrm{RL}\left(\tilde{\mathcal{T}}\right) - 1\\ &= \mathrm{RL}\left(\mathcal{T}\right) + 1.\end{aligned}$$

Using the same arguments as in the previous case, we obtain the assertion (2.3.17) for $\mathcal{T} \in \Theta_{b,m}^{1}$. The cases with $p > 1$ can be proven in the same manner. We omit the details here. □

Now, we prove Lemma 2.3.4:

Proof of Lemma 2.3.4. Lemma 2.3.5 shows that $\hat{\mathbb{M}}_h$ fulfils the assumptions of Lemma 2.3.6. Thus, we know that $\mathbb{M}_h^{\star}$ contains only forward hanging nodes of degree one and that all $\mathbb{T}_h^{m,\star}$, $0 \leq m \leq M$, have patch structure and only spatial hanging nodes of degree one. What remains to show is that $\mathbb{M}_h^{\star}$ contains only backward hanging nodes of degree one.

We define the sets

$$\begin{aligned}\tilde{\mathbb{T}}_R^m &:= \left\{\mathcal{T} \in \tilde{\mathbb{T}}_h^m \,\middle|\, \mathrm{Ch}\left(\mathcal{T}\right) \subseteq \mathbb{T}_h^{m,\star}\right\},\\ \tilde{\mathbb{T}}_N^m &:= \left\{\mathcal{T} \in \tilde{\mathbb{T}}_h^m \,\middle|\, \mathcal{T} \in \mathbb{T}_h^{m,\star}\right\}.\end{aligned}$$

From Lemma 2.3.5 and Lemma 2.3.6, we deduce $\mathbb{T}_h^{m,0} = \tilde{\mathbb{T}}_N^m \cup \mathrm{Ch}\left(\tilde{\mathbb{T}}_R^m\right)$, $\tilde{\mathbb{T}}_N^m \cap \mathrm{Ch}\left(\tilde{\mathbb{T}}_R^m\right) = \emptyset$, $\tilde{\mathbb{T}}_h^m = \tilde{\mathbb{T}}_N^m \cup \tilde{\mathbb{T}}_R^m$, and $\tilde{\mathbb{T}}_N^m \cap \tilde{\mathbb{T}}_R^m = \emptyset$. Consequently, $\mathbb{M}_h^{\star}$ fulfills the assumptions of Lemma 2.3.5. The same agruments as in the derivation of (2.3.16) lead to

$$\begin{aligned}\Theta_{\star}^{m} &:= \left\{\mathcal{T} \in \mathbb{T}_h^{m,\star} \,\middle|\, \mathrm{hang}^{-}\left(\mathcal{T}, \mathbb{T}_h^{m-1,\star}\right) > 1\right\}\\ &= \left\{\mathcal{T} \in \mathbb{T}_h^{m,\star} \,\middle|\, \mathrm{hang}^{-}\left(\mathcal{T}, \mathbb{T}_h^{m-1,\star}\right) = 2\right\} \subseteq \tilde{\mathbb{T}}_N^m\end{aligned}$$

for $0 < m \leq M$. We have to show $\Theta_{\star}^{m} = \emptyset$ for $0 < m \leq M$.

We assume that an element $\mathcal{T} \in \Theta_{\star}^{m}$ exists. Then $\mathcal{T}$ also is an element of $\tilde{\mathbb{T}}_{N}^{m}$, $\mathbb{T}_{h}^{m,0}$, and $\hat{\mathbb{T}}_{h}^{m,0}$. Furthermore, we know $\operatorname{hang}^{-}\left(\mathcal{T}, \mathbb{T}_{h}^{m-1,\star}\right) = 2$, i.e.

$$\exists \mathcal{T}' \in \mathrm{CE}^{-}\left(\mathcal{T}, \mathbb{T}_{h}^{m-1,\star}\right): \quad \mathrm{RL}\left(\mathcal{T}'\right) = \mathrm{RL}\left(\mathcal{T}\right) + 2. \tag{2.3.21}$$

Since $\mathcal{T} \in \hat{\mathbb{T}}_{h}^{m,0}$ and since $\hat{\mathbb{M}}_{h}$ contains only backward hanging nodes of degree one, we deduce

$$\operatorname{hang}^{-}\left(\mathcal{T}, \hat{\mathbb{T}}_{h}^{m-1,0}\right) \leq 1,$$

i.e.

$$\mathrm{RL}\left(\mathrm{CE}^{-}\left(\mathcal{T}, \hat{\mathbb{T}}_{h}^{m-1,0}\right)\right) \leq \mathrm{RL}\left(\mathcal{T}\right) + 1. \tag{2.3.22}$$

From (2.3.21) and (2.3.22), $\mathcal{T}' \notin \mathrm{CE}^{-}\left(\mathcal{T}, \hat{\mathbb{T}}_{h}^{m-1,0}\right)$ follows. Consequently,

$$\mathrm{P}\left(\mathcal{T}'\right) \in \mathrm{CE}^{-}\left(\mathcal{T}, \hat{\mathbb{T}}_{h}^{m-1,0}\right)$$

by construction and

$$\mathrm{RL}\left(\mathrm{CE}^{-}\left(\mathcal{T}, \hat{\mathbb{T}}_{h}^{m-1,0}\right)\right) = \mathrm{RL}\left(\mathcal{T}\right) + 1.$$

The cell $\mathrm{P}\left(\mathcal{T}'\right)$ is refined during the steps (12)-(19) of Algorithm 2.3.2, i.e.

$$\mathrm{P}\left(\mathcal{T}'\right) \in \bigcup_{p} \Theta_{b,m-1}^{p}.$$

Because of $\mathcal{T}' \in \mathrm{CE}^{-}\left(\mathcal{T}, \mathbb{T}_{h}^{m-1,\star}\right)$, it holds $\mathcal{T} \in \mathrm{CE}^{+}\left(\mathrm{P}\left(\mathcal{T}'\right), \mathbb{T}_{h}^{m,\star}\right)$. From Lemma 2.3.7, we know

$$\forall \mathcal{K} \in \mathrm{CE}^{+}\left(\mathrm{P}\left(\mathcal{T}'\right), \mathbb{T}_{h}^{m,\star}\right): \quad \mathrm{RL}\left(\mathcal{K}\right) \geq \mathrm{RL}\left(\mathcal{T}'\right).$$

Thus, $\mathrm{RL}\left(\mathcal{T}\right) \geq \mathrm{RL}\left(\mathcal{T}'\right)$. This is a contradiction to (2.3.21) and the proof is complete. □

The following proposition ensures the desired mesh properties in the DM approach:

Proposition 2.3.8. *If the input mesh sequence*

$$\mathbb{M}_{h} = \left(\mathbb{T}_{h}^{m}\right)_{0 \leq m \leq M}$$

*in Algorithm 2.3.3 is patch regular, then the output mesh sequence*

$$\mathbb{M}_h^\star = \left(\mathbb{T}_h^{m,\star}\right)_{0 \le m \le M}$$

*is patch regular, too.*

Proof of Proposition 2.3.8. In Algorithm 2.3.3, we adaptively refine meshes $\mathbb{T}_h^m$, which have patch structure and only spatial hanging nodes of degree one. Consequently, Lemma 2.3.1 can be applied. Lemma 2.3.1 and Corollary 2.3.3 ensure the assumptions of Lemma 2.3.4, which implies the claim of the proposition. □

## 2.4. Adaptive Solution Algorithm

Up to now, we have presented all parts of the adaptive solution algorithm. In this section, we put them together. The adaptive solution algorithm is given in Algorithm 2.4.1. It is illustrated in Figure 2.4.1. We discuss the complex adaptive solution algorithm of the DM approach. The modifications for the CM approach are straightforward and left out.

The first step is to specifiy the initial data. The initial mesh $\mathbb{I}_h$ and an initial decomposition $\mathbb{T}_k^0$ of the time interval $I$ are chosen. For the calculation of the refinement indicators, the way of taking the absolute value and the number of smoothing iterations are determined. Furthermore, the constants of the refinement strategies, Algorithm 2.2.2, are specified. We need the number of new mesh cells, which are created, if the cell is refined. It depends on the refinement and on $\dim(\Omega)$. To predict the reduction of the error, we use the convergence rate of the space-time finite element ansatz. The equilibration constant $c > 1$ in Algorithm 2.2.1 has to be chosen, too. The number of the adaptive solution iteration is $l$. Furthermore, a stopping criterion is specified. It decides, when the adaptive solution algorithm terminates. Possible choices are: A fixed number of iterations $l^\star$ has been reached or the estimated error is smaller than a given tolerance tol. Based on $\mathbb{I}_h$, the initial mesh sequence $\mathbb{M}_h^0$ is created (Step (1) in Algorithm 2.4.1).

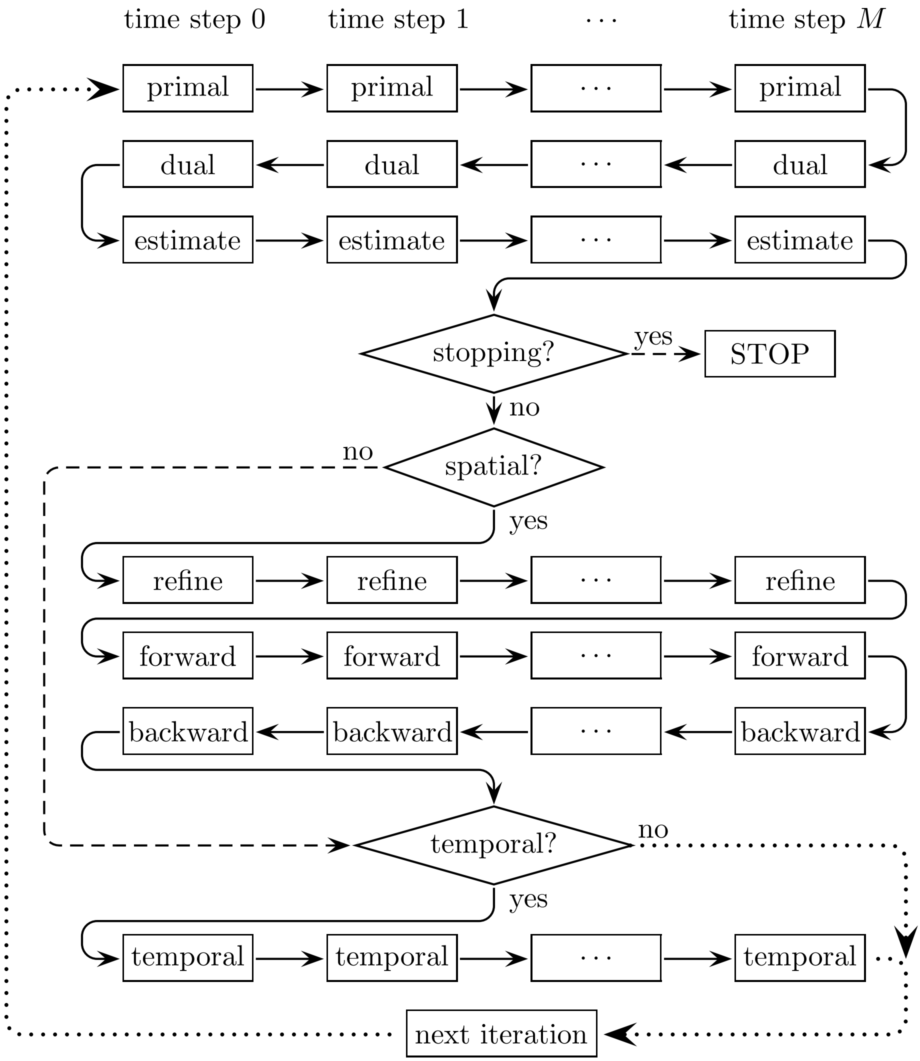


FIGURE 2.4.1. Illustration of the adaptive solution algorithm

The first step is the determination of the solution $w^l_{kh} = \left(u^l_{kh}, v^l_{kh}\right)$ of the primal problem (Definition 1.2.1). For this purpose, we use the Time Stepping Scheme 1.2.3. Since we need $w_{kh}$ during the solution of the dual problem and the evaluation of the error estimator, we have to save

**Algorithm 2.4.1** Adaptive solution algorithm for the DM approach

```
Input: Stopping criterion, initial mesh $\mathbb{I}_h$, initial temporal mesh $\mathbb{T}_k^0$ with $M^0$ time steps, equilibration constant $c$, number of smoothing iterations $S$, the functions $\mathcal{E}(r)$ and $\mathcal{N}(r)$
 (1) Set $l = 0$ and $\mathbb{M}_h^l = (\mathbb{I}_h)_{0 \leq m \leq M^0}$.
 (2) Solve the primal problem by the Time Stepping Scheme 1.2.3. Save $u_{kh}^l$ and $v_{kh}^l$.
 (3) Solve the dual problem by the Time Stepping Scheme 1.3.8. Save $\bar{u}_{kh}^l$ and $\bar{v}_{kh}^l$.
 (4) Evaluate the error estimator given in Definition 1.3.10 and calculate the error indicators as outlined in Section 2.1.
 (5) If the stopping criterion is fulfilled, then set $u_{kh}^\star = u_{kh}^l$ and $v_{kh}^\star = v_{kh}^l$ and STOP.
 (6) Use Algorithm 2.2.1 to determine, whether the spatial, the temporal or both meshes have to be refined.
 (7) If only temporal refinement, then set $\tilde{\mathbb{M}}_h^{l+1} = \mathbb{M}_h^l$ and go to (10).
 (8) Use Algorithm 2.2.2 to determine $\Theta_{h,r}^{DM,l}$.
 (9) Call Algorithm 2.3.3 with $\mathbb{M}_h^l$ and $\Theta_{h,r}^{DM,l}$. Call the result $\tilde{\mathbb{M}}_h^{l+1}$.
(10) If only spatial refinement, then set $\mathbb{M}_h^{l+1} = \tilde{\mathbb{M}}_h^{l+1}$ and $\mathbb{T}_k^{l+1} = \mathbb{T}_k^l$. Go to (13)
(11) Use Algorithm 2.2.2 to determine $\Theta_{k,r}^l$.
(12) Refine the temporal mesh $\mathbb{T}_k^l$ and modify the spatial mesh sequence $\tilde{\mathbb{M}}_h^{l+1}$ according to $\Theta_{k,r}^l$. Call the results $\mathbb{T}_k^{l+1}$ and $\mathbb{M}_h^{l+1}$.
(13) Set $l \leftarrow l + 1$ and go to (2).
Output: Desired solution $u_{kh}^\star$ and $v_{kh}^\star$.
```

it. By this, a huge amount of data can be created[1] and it can become a time consuming operation. These operations are referred to as “primal” in Figure 2.4.1 and is step (2) in Algorithm 2.4.1.

The backward or dual problem (Definition 1.3.7) is solved in step (3) of Algorithm 2.4.1 (“dual” in Figure 2.4.1). The Time Stepping Scheme 1.3.8 is applied to solve the dual problem. The dual solution $z_{kh}^l = \left(\bar{u}_{kh}^l, \bar{v}_{kh}^l\right)$ is saved, because we need it in the evaluation of the error estimate.

[1] E.g. if we have 2000000 unknowns and 10000 time steps, roughly speaking, a data set of 160GB is created just for the primal solution. The same amount of data is needed for the dual solution.

The evaluation of the error estimate and the calculation of the refinement indicators is the next step (Step (4) in Algorithm 2.4.1 and "estimate" in Figure 2.4.1). On one hand, we evaluate the error estimate specified in Definition 1.3.10. For this purpose, we evaluate the terms given in Section A.3 and A.4. On the other hand, we calculate the refinement indicators on each cell, i.e. we have to evaluate the localised form of the error estimator, taking the absolute value and to smooth the data according to Section 2.1. The refinement indicators are saved for their use in the refinement strategies.

After the evaluation of the error estimate, we check the stopping criterion ("stopping?" in Figure 2.4.1). If it is fulfilled, we stop the iteration with the desired solution $w_{kh}^{\star}$ (Step (5) in Algorithm 2.4.1). If it is not, we proceed with the adaptive refinement.

The first step of the adaptive refinement procedure is to decide, whether the temporal mesh $\mathbb{T}_k^l$, the spatial mesh sequence $\mathbb{M}_h^l$, or both should be refined. This question is answered by Algorithm 2.2.1 (Step (6) in Algorithm 2.4.1). In Figure 2.4.1, we illustrate this step by "spatial?" and "temporal?". If no spatial refinement is needed, we set $\mathbb{M}_h^l = \tilde{\mathbb{M}}_h^{l+1}$ and skip the spatial refinement (Step (7) in Algorithm 2.4.1).

We begin the spatial refinement with the determination of the set $\Theta_{h,r}^{DM,l}$. It contains all mesh cells, which are chosen for refinement by Algorithm 2.2.2. For the determination of $\Theta_{h,r}^{DM,l}$, all refinement indicators of the mesh sequence $\mathbb{M}_h^l$ are compared. This should lead to a maximum efficient discretisation, since all available information is used in the refinement strategy and it is not restricted to a single time step. The determination of $\Theta_{h,r}^{DM,l}$ is step (8) of Algorithm 2.4.1.

The adaptive refinement is carried out by Algorithm 2.3.3. The resulting mesh sequence is chosen as intermediate mesh sequence $\tilde{\mathbb{M}}_h^{l+1}$. This is step (9) of Algorithm 2.4.1. The detailed adaptive refinement method is described in Section 2.3. In Figure 2.4.1, we refer to the adaptive refinement and the spatial regularisation (Algorithm 2.3.1) by "refine", to the forward regularisation (Algorithm 2.3.2, (1)-(10)) by "forward" and to the backward regularisation (Algorithm 2.3.2, (11)-(20)) by "backward".

If it has been decided in step (6) not to refine the temporal mesh, we set $\mathbb{M}_h^{l+1} = \tilde{\mathbb{M}}_h^l$, $\mathbb{T}_k^{l+1} = \mathbb{T}_k^l$ and skip the temporal refinement (Step (10) of Algorithm 2.4.1). Otherwise, we use Algorithm 2.2.2 to determine the set $\Theta_{k,r}^l$. It contains the temporal mesh cells, which are marked for refinement. This is step (11) of Algorithm 2.4.1. Then the temporal mesh is adaptively refined in step (12) of Algorithm 2.4.1. There, we have to modify the spatial mesh sequence $\tilde{\mathbb{M}}_h^{l+1}$, since the number of time steps is changed and a specific spatial mesh is connected to each time step. If a time step is refined, we have to add a spatial mesh, call it $\mathbb{T}_h^{l+1,m+1/2}$, in $\tilde{\mathbb{M}}_h^{l+1}$. Possible choices for $\mathbb{T}_h^{l+1,m+1/2}$ are $\mathbb{T}_h^{l+1,m}$, $\mathbb{T}_h^{l+1,m+1}$ or, e.g., the mesh consisting of the finest cells of $\mathbb{T}_h^{l+1,m}$ and $\mathbb{T}_h^{l+1,m+1}$. We simply insert $\mathbb{T}_h^{l+1,m+1}$. This is referred to as "temporal" in Figure 2.4.1.

After the adaptive refinement, we increase the number of the iteration cycle $l$ and restart the iteration with the solution of the primal problem. This is step (13) of Algorithm 2.4.1. Up to now, we have presented the adaptive solution algorithm. In the upcoming section, we will discuss the implementation of Algorithm 2.4.1.

## 2.5. Implementation Issues

The adaptive solution algorithm has been implemented in the finite element library SOFAR (Scientific Object oriented Finite element library for Application and Research) [**22**]. SOFAR has been basically designed for adaptive finite element methods in the context of structural mechanics. It includes a flexible mesh and degree of freedom management. Furthermore, the inclusion of different basis functions is an easy task, such that *hp*-adaptive finite element methods can efficiently be realised. During the last years, a lot of solution algorithms for time dependent problems, e.g. parabolic, hyperbolic and mixed parabolic-hyperbolic problems, have been added to the library, where we have used an object oriented concept. In this approach, all algorithms are created on the basis of general interfaces and can be used for different classes, which only have to implement these interfaces. Thus, all adaptive techniques mentioned in this thesis can easily be applied to other time dependent problem classes than hyperbolic problems of second order. During the implementation of the adaptive solution algorithm, one faces three major problems: The management and storage of the mesh sequence $\mathbb{M}_h^l$, the evaluation of integrals on different meshes and the storage of the primal and dual solution. One after the other, we discuss our solution approaches to these problems.

During the calculation, we have to construct the meshes $\mathbb{T}_h^{l,m}$ in each time step. One idea is to individually save these meshes. There exist several disadvantages: Since the whole geometry and history has to be stored, a big amount of data arises. The creation of the mesh is time consuming. Furthermore, the comparison of different meshes is complicated. To circumvent all these problems, we have developed the following approach: We create a so called global mesh $\mathbb{G}_h$, which contains all mesh cells ever used during the calculation. In the worst case, there holds

$$\mathbb{G}_h = \bigcup_{j=0}^{l} \mathbb{U}_h^j.$$

To construct the single meshes $\mathbb{T}_h^{l,m}$, we take the necessary elements out of $\mathbb{G}_h$. Every mesh cell in $\mathbb{G}_h$ has a unique index. To store the mesh $\mathbb{T}_h^{l,m}$, we only have to save the indices of the mesh cells in $\mathbb{T}_h^{l,m}$. Thus, a mesh $\mathbb{T}_h^{l,m}$ is identified with its index vector $\bar{\mathbb{T}}^{l,m}$. Whenever we cannot perform an operation with the index vector and consequently need the mesh, we recreate it from the index vector. Furthermore, the comparison of two meshes becomes very easy. Let $\mathbb{T}_h^{l,m_1}$ and $\mathbb{T}_h^{l,m_2}$ be two meshes in $\mathbb{M}_h^l$. If we want to check, whether an element $\mathcal{T} \in \mathbb{T}_h^{l,m_1}$ also is an element of $\mathbb{T}_h^{l,m_2}$ or not, we only check, if the index of $\mathcal{T}$ is contained in $\bar{\mathbb{T}}^{l,m_2}$ or not. In order to determine the corresponding elements $\mathrm{CE}^{\pm}(\mathcal{T})$, we perform this check with the refinement history $\mathrm{RH}(\mathcal{T})$.

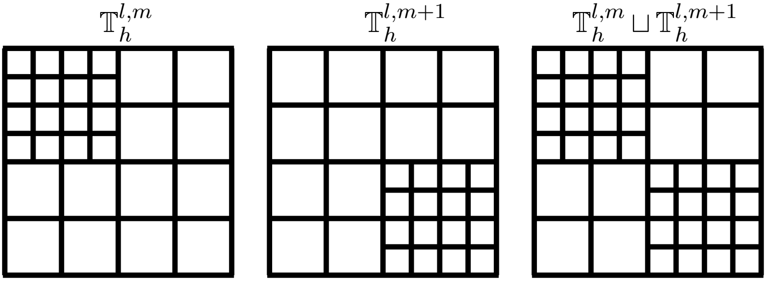


FIGURE 2.5.1. Illustration of the assembling on two different meshes

Since the regularisation w.r.t. hanging nodes in time relies on the determination of the corresponding elements $\mathrm{CE}^{\pm}(\mathcal{T})$, an efficient implementation of $\mathrm{CE}^{\pm}(\mathcal{T})$ is indispensable. It is also needed, when we calculate an integral of two finite element functions, which are based on different meshes, e.g., the $L^2$-scalar product of $u_{kh}^m$ and $u_{kh}^{m-1}$, $\left(u_{kh}^m, u_{kh}^{m-1}\right)$, with $\mathbb{T}_h^m \neq \mathbb{T}_h^{m-1}$ or when we assemble the terms in the time stepping schemes, which are conntected to the solution in the previous time step, e.g. $\left(u_{kh}^{m-1}, \varphi_h\right)$ with $\varphi \in V_h^m$. Let us exemplarily study $\left(u_{kh}^{m-1}, \varphi_h\right)$ here. We calculate the integral by applying numerical quadrature rules, e.g. the 2 × 2 Gauß formula, on the single mesh cells $\mathcal{T} \in \mathbb{T}_h^m$ and summing up the single results. But $u_{kh}^{m-1}$ is not necessarily continuously differentiable on $\mathcal{T}$. Consequently, the approximation of the integral on $\mathcal{T}$ introduces additional errors in the calculation. Because of the low regularity, the use of higher order quadrature rules does not lead to a more exact approximation. However, we know where $u_{kh}^{m-1}$ is not continuously differentiable, namely on the edges of $\mathbb{T}_h^{m-1}$. To circumvent this problem, we have to calculate the integral on the finest cells of $\mathbb{T}_h^m$ and $\mathbb{T}_h^{m-1}$. The mesh containing the finest cells of $\mathbb{T}_h^{m-1}$ and $\mathbb{T}_h^m$ is denoted by $\mathbb{T}_h^{m-1} \sqcup \mathbb{T}_h^m$ and is illustrated in Figure 2.5.1. For the integration, we locally create the mesh $\mathbb{T}_h^{m-1} \sqcup \mathbb{T}_h^m$. Here locally means that we numerically integrate on every mesh cell $\mathcal{T} \in \mathbb{T}_h^m$. If $\left|\mathrm{CE}^{-}(\mathcal{T})\right| > 1$, we split the integration up, integrate on every element of $\mathrm{CE}^{-}(\mathcal{T})$ and sum up the results. See, e.g., [**94**], Section 5.3, for a detailed discussion.

The last topic, which we discuss, is the storage of the primal and dual solution. The primal solution is needed during the solution of the dual problem and during the evaluation of the error estimator. The dual solution enters in the evaluation of the error estimator, too. Consequently, we have to store the primal and dual solutions. The first idea is to save them in the main memory. But then the available main memory is exceeded quickly, since we also have to store the matrices, auxiliary vectors, the mesh and so on. Another possibility is to store the data on the hard disc. This approach is used here. Numerical experiments show that the additional amount of computing time is negligible, since the difference is smaller than the natural fluctuation in the computing time. Other alternatives are checkpointing or windowing. In the checkpointing approach, the solutions are only stored in some particular time points, so called checkpoints. The values between the checkpoints are recalculated, whenever they are needed, see for instance [**85**]. Beside the difficult organisation of the data storage, the recalculation also needs a lot of computing time. Consequently, this approach turned out to be inefficient [**95**].

## 2.6. A Heuristic Error Indicator

The error estimator, which we have presented, is very complex and the evaluation is time consuming. We want to compare it to a heuristic error indicator, which can be evaluated easily and quickly. We do not request an exact error estimate in the functional $J$ from the heuristic error indicator. It should only provide refinement indicators for the adaptive refinement process.

For the derivation of the error indicator, we consider the presented space-time finite element discretisation in the view of Rothe's method. There, we discretise the temporal direction first and obtain a so called semi discrete solution $w_k$. The arising spatial problems are solved by the finite element method approximatively . For the temporal discretisation by the continuous Galerkin method based on linear polynomials, we have the following a posteriori error estimate, see [**96**], Chapter 4:

$$|(e_M,\varphi)| \le C_s C_I \max_{1\le m\le M} \left\{ k_m \sup_{t\in I_m} |R\left(t,w_k(t),\varphi\right) - \mathrm{P}_0 R\left(t,w_k(t),\varphi\right)| \right\},$$

which holds for all $\varphi \in H^1\left(\Omega,\Gamma_D\right)$. In this estimate, we control the error $e = w - w_k$ at the final time $t_M = T$. The stability constant $C_S > 0$ results from the estimate of the corresponding dual problem, $C_I > 0$ is the interpolation constant. The residual $R\left(t,w_k(t),\varphi\right)$ is given by

$$\begin{aligned} R\left(t,w_k(t),\varphi\right) \quad &:= \quad \left(\dot{u}_k(t) - v_k(t) + \dot{v}_k(t),\varphi\right) + a\left(u_k(t)\right)(\varphi) \\ &\qquad - \left(f(t),\varphi\right) - \left(q(t),\varphi\right)_{\Gamma_N}. \end{aligned}$$

The operator $\mathrm{P}_0$ is the temporal projection on piecewise constant functions. To obtain an evaluable error indicator, we approximate $w_k$ by $w_{kh}$ and $\varphi$ by $\varphi_h \in V_h^m$. Then we can evaluate

$$\eta_{I,k}^m \quad \approx \quad k_m \sup_{t\in I_m} |R\left(t,w_k(t),\varphi\right) - \mathrm{P}_0 R\left(t,w_k(t),\varphi\right)|,$$

where

$$\begin{aligned} &\eta_{I,k}^m \\ := \quad & k_m \left| \begin{array}{c} \frac{1}{2}\left(v_{kh}^m - v_{kh}^{m-1},\varphi_h\right) - \frac{1}{2}\left[a\left(u_{kh}^m\right)(\varphi_h) - a\left(u_{kh}^{m-1}\right)(\varphi_h)\right] \\ +\frac{1}{2}\left(f^m - f^{m-1},\varphi_h\right) + \frac{1}{2}\left(q^m - q^{m-1},\varphi_h\right)_{\Gamma_N} \end{array} \right|. \end{aligned}$$

The term $\eta_{I,k}^m$ provides the temporal refinement indicator in the $m^{\mathrm{th}}$ time step.

For the spatial discretisation, we use the residual error estimator presented in [**111**], Section 3.3. We obtain

$$\begin{aligned} \eta_{I,h}^m \quad &:= \quad \left\{ \sum_{\mathcal{T}\in\mathbb{T}_h^m} \left(\eta_{I,\mathcal{T}}^m\right)^2 \right\}^{1/2}, \\ \left(\eta_{I,\mathcal{T}}^m\right)^2 \quad &:= \quad h_{\mathcal{T}}^2 \left\|R_T\left(w_{kh}^m\right)\right\|_{0,\mathcal{T}}^2 + \frac{1}{2} \sum_{E\in\partial\mathcal{T}} h_E \left\|R_E\left(u_{kh}^m\right)\right\|_{0,E}^2, \end{aligned}$$

where

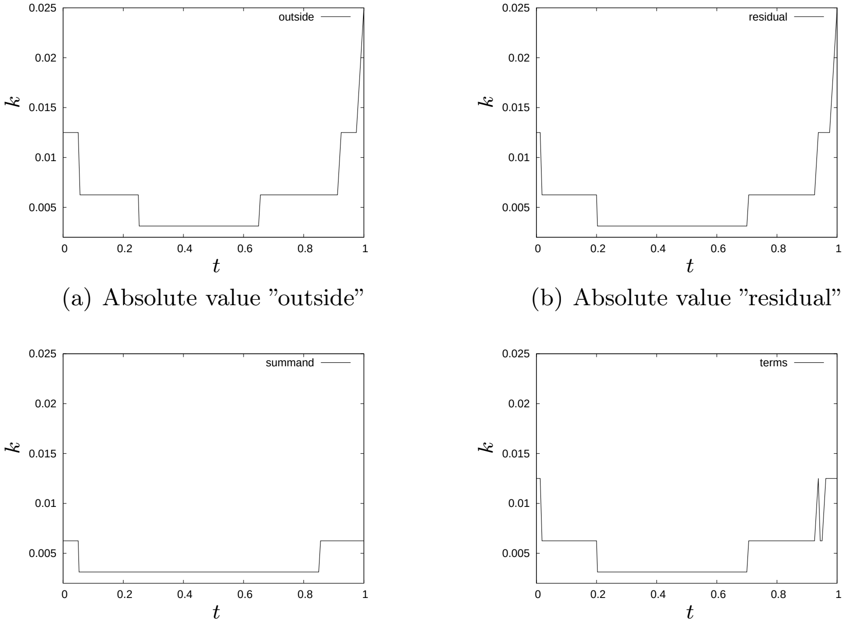
$$\begin{aligned}
R_T\left(w_{kh}^m\right) &:= u_{kh}^m - u_{kh}^{m-1} + \left(1 - \frac{1}{2}k_m\right) v_{kh}^m \\
&\quad - \left(1 + \frac{1}{2}k_m\right) v_{kh}^{m-1} - \frac{1}{2}k_m\left(f^m + f^{m-1}\right) \\
&\quad + \frac{1}{2}k_m\left(-\mathcal{A}\left(u_{kh}^m\right) - \mathcal{A}\left(u_{kh}^{m-1}\right)\right), \\
R_E\left(u_{kh}^m\right) &:= \frac{1}{2}k_m\left(\left[\mathcal{B}\left(u_{kh}^m\right)\right] + \left[\mathcal{B}\left(u_{kh}^{m-1}\right)\right]\right),
\end{aligned}$$

and $[\mathcal{B}]$ is defined in (2.1.2). This error indicator is not exact, since we know only approximations of $u_{kh}^{m-1}$ and $v_{kh}^{m-1}$. See [**25**] for a detailed discussion. Since the error estimators involve unknown constants and provide only upper bounds for the error, we are not able to apply the space-time refinement strategy presented in Algorithm 2.2.1. It is based on the fact that we know approximations of the spatial and the temporal error or upper and lower bounds, which is not given in this case. Thus, we refine both, the spatial and the temporal, mesh in every refinement interation.

## 2.7. Numerical Results

Up to now, we have presented an a posteriori error estimation and have derived refinement indicators for the refinement strategies. In this section, we study the effects of the different refinement indicators and choose the parameters for the subsequent examples. We consider a kind of penalty discretisation for a dynamic contact problem and a 2D linear elastic example.

**2.7.1. Comparison of Different Refinement Indicators.** In this section, we consider the basic domain $\Omega = [0,1]^2$ and $I = [0,1]$. The differential operator is given by $\mathcal{A}(u) = \Delta u + u^3$. We prescribe homogeneous Dirichlet boundary conditions on $\Gamma_D = \partial\Omega$. The initial values and the right hand side are given by means of the prescribed analytical solution,



(a) Absolute value ”outside”

(b) Absolute value ”residual”

(c) Absolute value ”summand”

(d) Absolute value ”terms”

FIGURE 2.7.1. Temporal meshes for different ways to take the absolute value

i.e.

$$\begin{aligned} u_s &= u(0), \\ v_s &= \dot{u}(0), \\ f &= \ddot{u} - \Delta u - u^3. \end{aligned}$$

We begin with the discussion of the temporal refinement indicators. For this purpose, we set

$$\begin{aligned} \tilde{u}\left(t\right) &:= \begin{cases} \tilde{u}_1(t), & t \in [0, 0.375] \\ \tilde{u}_2(t), & t \in (0.375, 0.625) \\ \tilde{u}_3(t), & t \in [0.625, 1]\,, \end{cases} \\ u\left(x_1, x_2, t\right) &:= \tilde{u}(t) \sin\left(\pi x_1\right) \sin\left(\pi x_2\right), \end{aligned}$$

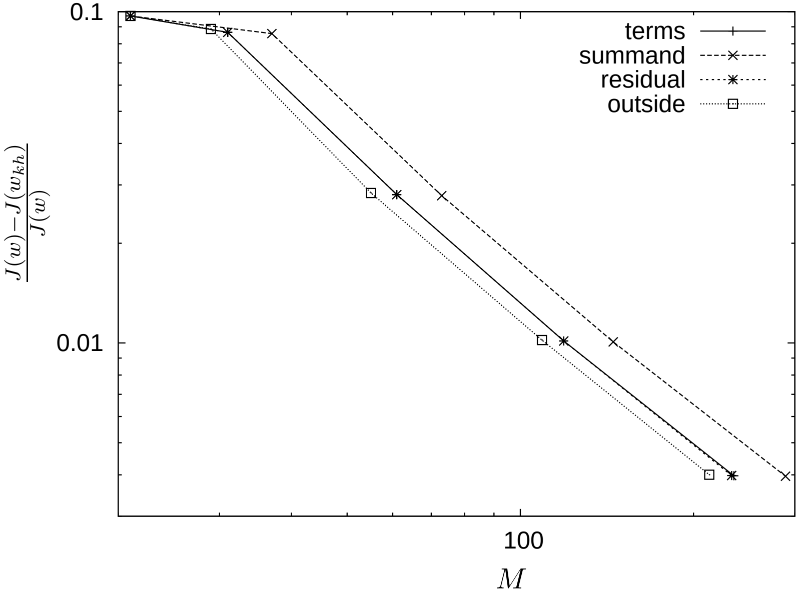
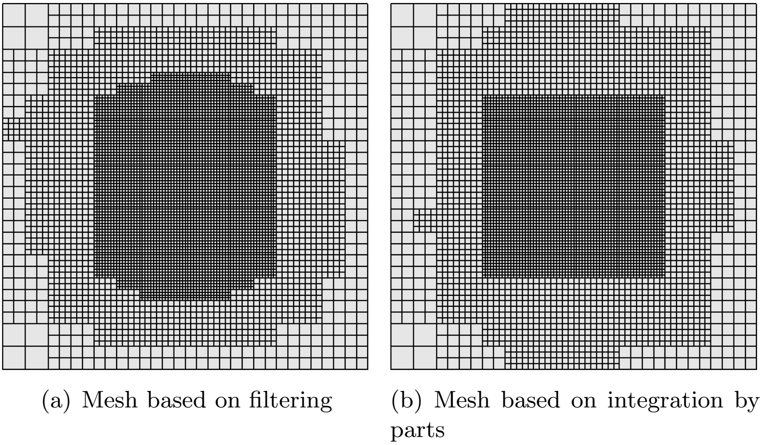


FIGURE 2.7.2. Error in the output funcitonal w.r.t. adaptive temporal refinement



(a) Mesh based on filtering

(b) Mesh based on integration by parts

FIGURE 2.7.3. Meshes in the CM approach

with

$$\tilde{u}_1(t) \quad := \quad \exp\left(8\left(t - 0.375\right)\right),$$

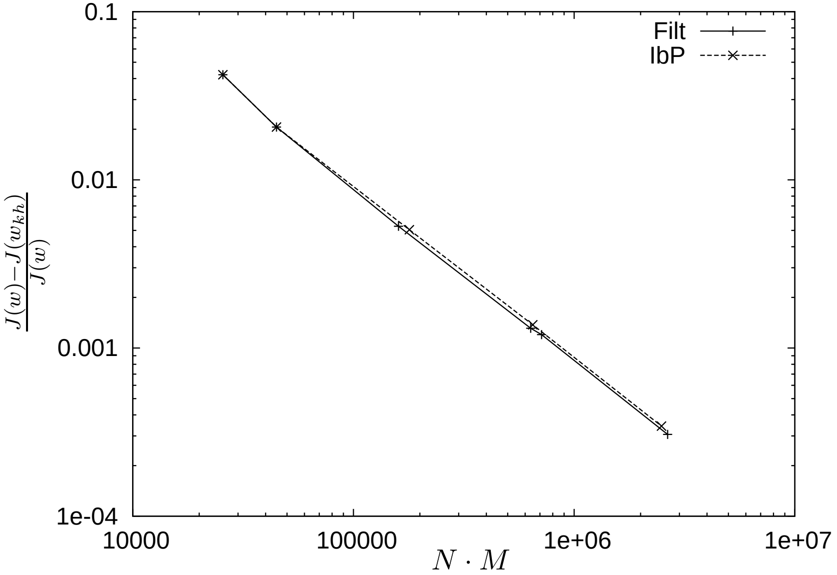


FIGURE 2.7.4. Developement of the error in $J$ w.r.t. spatial adaptive refinement based on the CM approach. Here, localisation by filtering is named “Filt” and localisation by integration by parts “IbP”

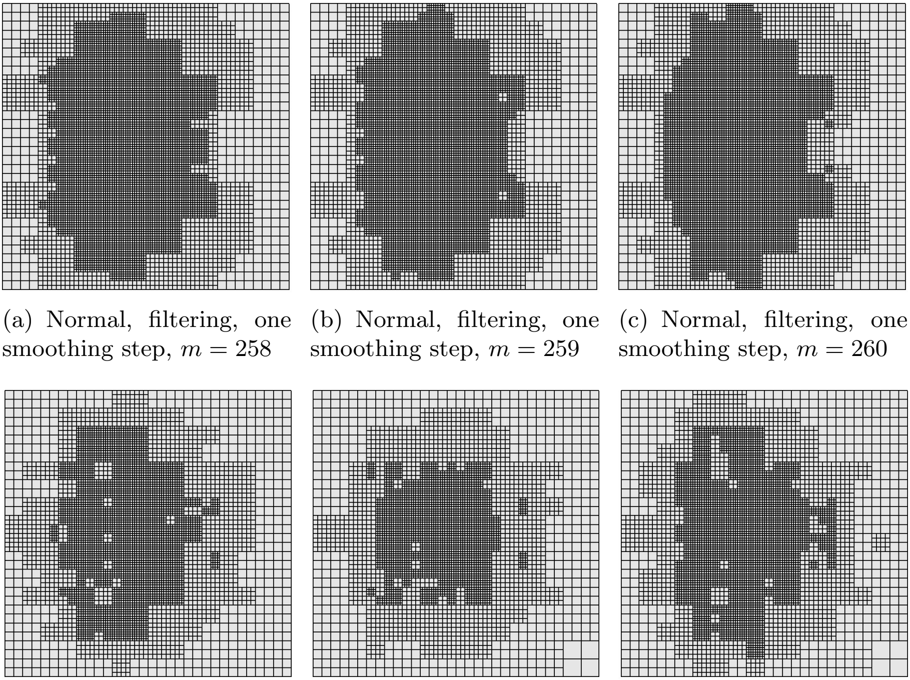
$$\begin{aligned}\tilde{u}_2(t) &:= 1024t^4 - 2048t^3 + 1472t^2 - 448t + 49.75,\\ \tilde{u}_3(t) &:= \exp\left(-8\left(t-0.625\right)\right).\end{aligned}$$

The functional of interest is given by

$$J\left(w\right) = \frac{8}{3|B|}\int_{0.625}^{1}\int_{B} u\,dx\,dt$$

with $B = [0.25, 0.75]^2$.

Based on a uniform spatial mesh with 4096 cells and an initial temporal mesh with 20 time steps, we perform 5 adaptive solution cycles, where only temporal refinements are permitted. In Figure 2.7.1, the temporal meshes are depicted. For $\eta_k^i$ and $\eta_k^n$, we obtain the same temporal meshes. They only differ, if we change the method of taking the absolute value. The general structure of all meshes is similar; we have larger time steps at the beginning and at the end and smaller time steps in the middle.



(a) Normal, filtering, one smoothing step, $m = 258$

(b) Normal, filtering, one smoothing step, $m = 259$

(c) Normal, filtering, one smoothing step, $m = 260$

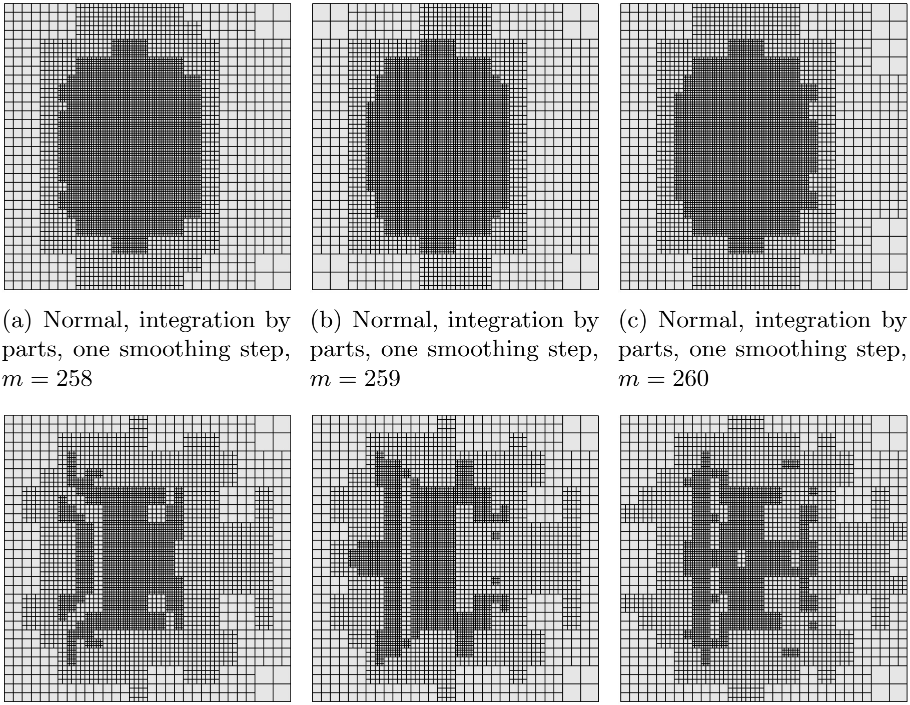
(d) Interpolated, filtering, no smoothing, $m = 258$

(e) Interpolated, filtering, no smoothing, $m = 259$

(f) Interpolated, filtering, no smoothing, $m = 260$

FIGURE 2.7.5. Meshes created during the adaptive refinement process based on different refinement indicators, localisation by filtering

Looking at Figure 2.7.2, we observe that the results for the "summand" method of taking the absolute value are worse than the other results. This also is the case in any other example, which we have considered. The results for the "terms" and the "residual" method of taking the absolute value are identical. The "outside" method is the best one, which is supported by other examples, too. Consequently, we will use the "outside" method of taking the absolute value. Since the results are the same w.r.t. $\eta_k^i$ and $\eta_k^n$, we have the free choice and choose $\eta_k^i$. The reason will become apparent after the discussion of the spatial adaptive refinement.



(a) Normal, integration by parts, one smoothing step, $m = 258$

(b) Normal, integration by parts, one smoothing step, $m = 259$

(c) Normal, integration by parts, one smoothing step, $m = 260$

(d) Interpolated, integration by parts, no smoothing, $m = 258$

(e) Interpolated, integration by parts, no smoothing, $m = 259$

(f) Interpolated, integration by parts, no smoothing, $m = 260$

FIGURE 2.7.6. Meshes created during the adaptive refinement process based on different refinement indicators, localisation by integration by parts

For the discussion of the spatial refinements indicators, we set

$$u\left(x_1, x_2, t\right) := \begin{cases} \sin^4\left(2\pi x_1 + \pi t\right)\sin\left(\pi x_2\right), & x_1 \in (0.5 - 0.5t, 1 - 0.5t) \\ 0, & \text{else.} \end{cases}$$

Here, we use a uniform temporal mesh with 400 time steps. Let us begin with the CM approach. In Figure 2.7.3, the adaptive meshes created based on filtering and on integration by parts are depicted. These meshes are representative for both methods. Only small differences occur, if we use other ways to take the absolute value or exchange $\eta_h^i$ and $\eta_h^n$.

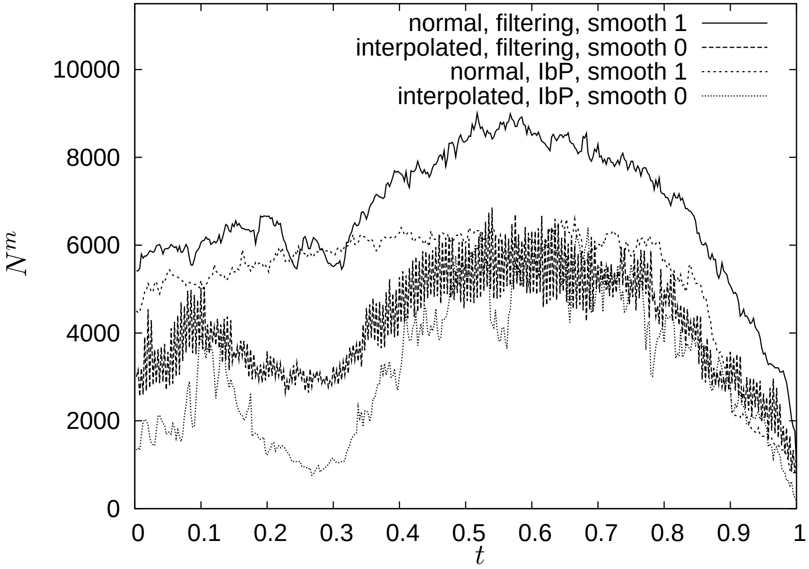


FIGURE 2.7.7. Number of mesh elements in the single time steps

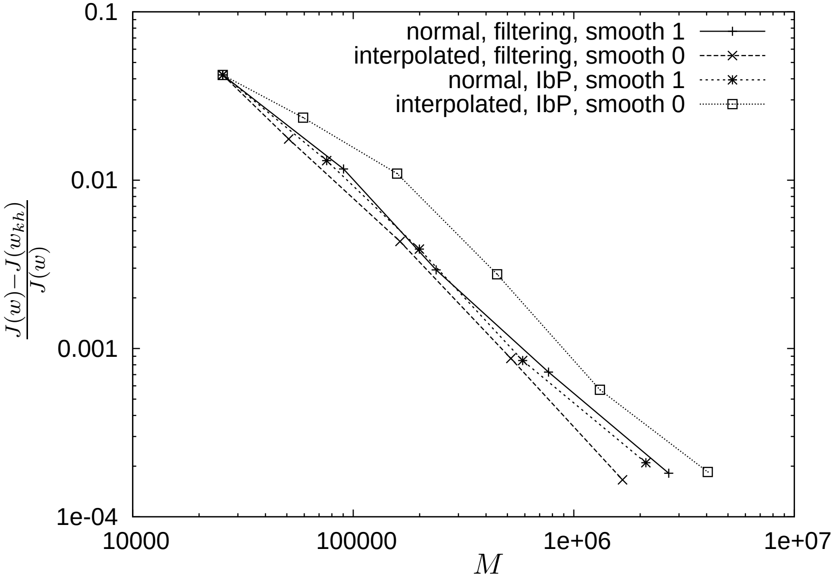


FIGURE 2.7.8. Convergence in the output functional $J$ for different refinement indicators

The adaptive meshes shown in Figure 2.7.3 look similar, too. This is supported by the convergence behaviour presented in Figure 2.7.4, where nearly no difference between the two methods arise. The CM approach is not sensitive to these modifications. Consequently, we will choose the refinement indicators based on the results of the DM approach.

The Figures 2.7.5 and 2.7.6 show adaptive meshes created by different refinement indicators in the DM approach. Because of the large amount of possible refinement indicators, we only show some typical examples. We observe that the meshes change rapidly from one time step to the other, if the indicators are not smoothed. In particular, this occurs for refinement indicators based on $\eta_h^i$. This behaviour is also shown in Figure 2.7.7, where the number of mesh elements in the different time steps is depicted. The oscillations in the graphs connected to the non-smoothed indicators express this behaviour. Furthermore, connected areas are refined for smoothed refinement indicators. If we do not smooth the indicators, the refined zones are more fragmented. Since the results for refinement indicators based on integration by parts strongly depend on the way of taking the absolute value, we work with the filtering approach. If we use filtering, the results are the same for additional smoothing steps, i.e. it is sufficient to apply one smoothing step. However, we have to apply one or up to three smoothing steps to obtain reasonable results based on integration by parts. The disadvantage of the additional smoothing is that it leads to a larger number of refined cells. Consequently, the meshes can be less effective. This is shown in Figure 2.7.8, where the convergence of the adaptive methods based on different refinement indicators is shown.

**2.7.2. Example 1: Penalty Method for Dynamic Contact.** Dynamic contact problems are discussed in Chapter 4. They lead to constrained optimisation problems, which can be solved by the penalty method. This approach simplifies the structure of the problems. If the penalty parameter is well chosen, this method leads to a reasonable numerical solution of the dynamic contact problem, see for instance [**54**].

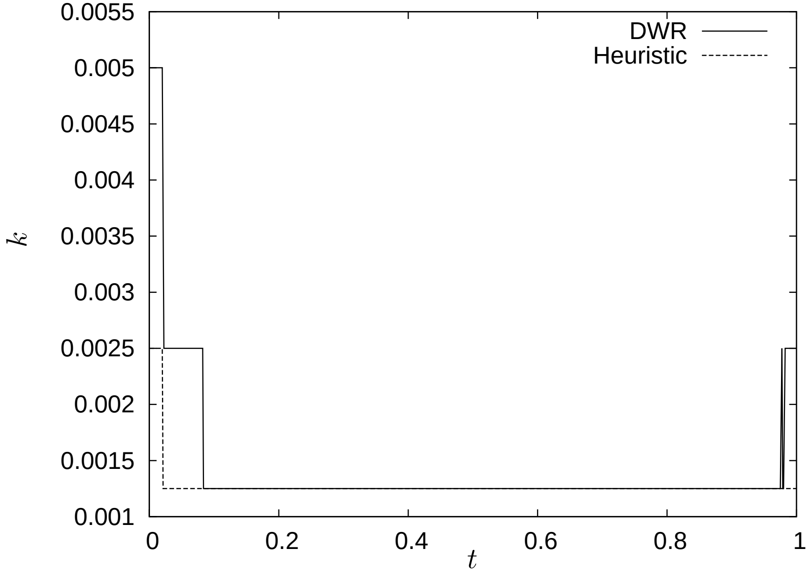


FIGURE 2.7.9. Temporal meshes based on the heuristic error indicator and on the DWR error estimator

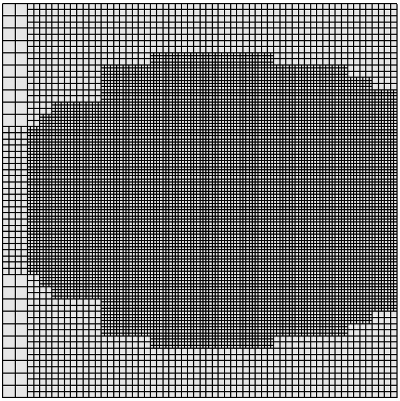


FIGURE 2.7.10. Mesh created by the adaptive refinement in the CM approach

The basic domain is $\Omega = [0,1]^2$ and $I = [0,1]$. Homogeneous Dirichlet boundary conditions are prescribed on $\Gamma_D := \{(x_1,x_2) \in \partial\Omega | \, x_1 = 0\}$ and homogeneous Neumann boundary conditions on $\Gamma_N := \partial\Omega \backslash \Gamma_D$. On the set $\Gamma_c := \{(x_1,x_2) \in \partial\Omega | \, x_1 = 1\}$, the displacement is restricted by

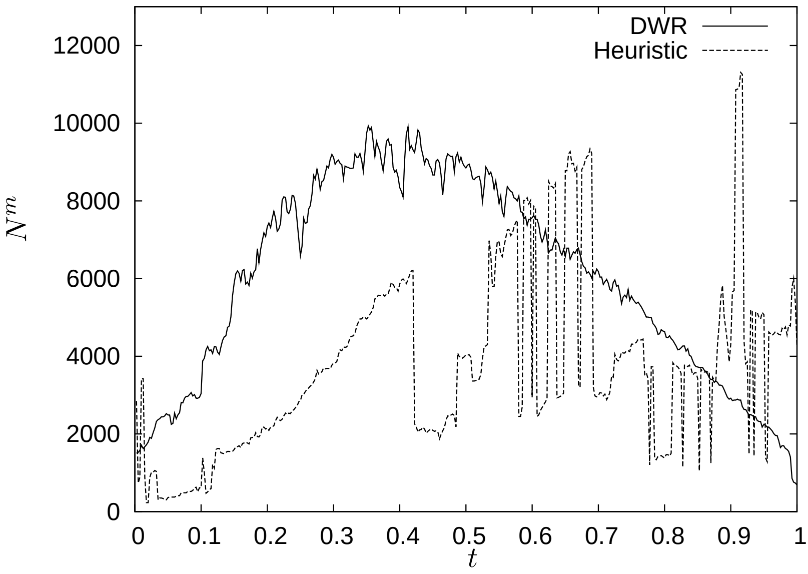


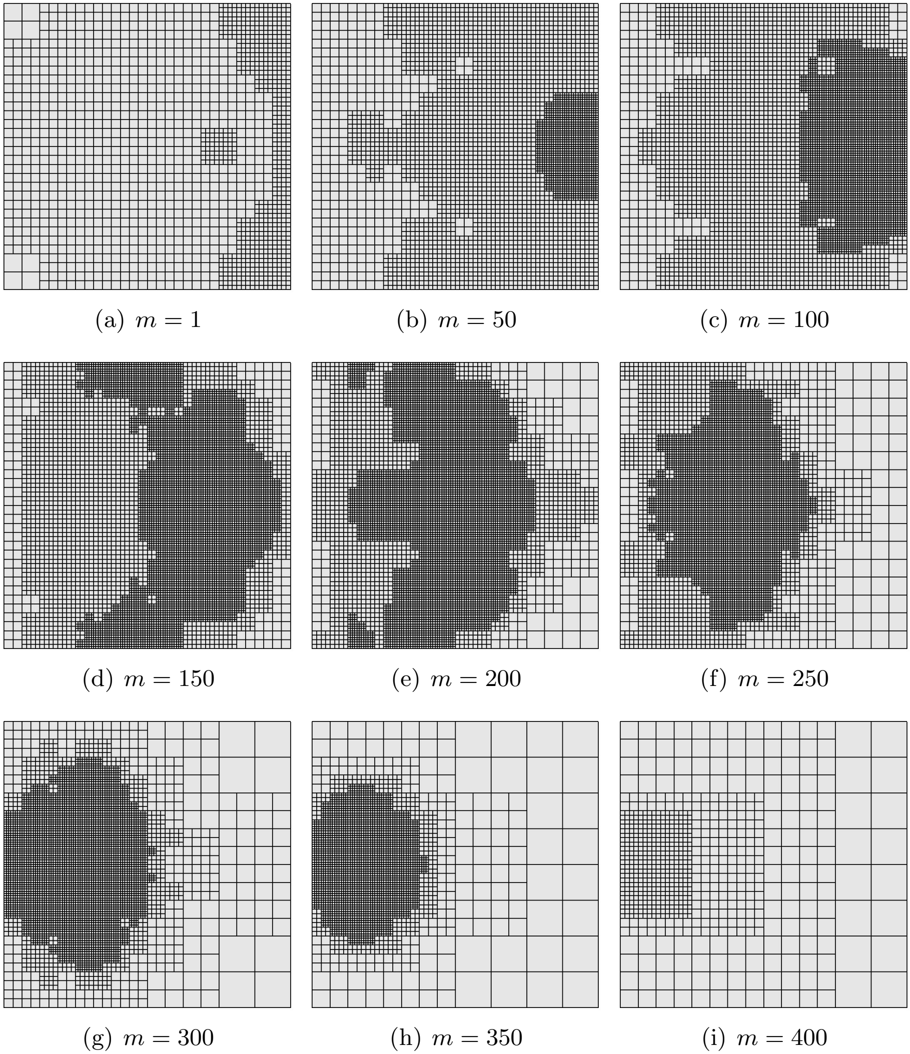
FIGURE 2.7.11. Distribution of the number of mesh elements based on the DWR error estimator and the heuristic error indicator in the DM approach

| $M$ | $\sum_{m=1}^{M} N^M$ | $E_{\text{rel}}$ | $I_{\text{eff}}$ |
|---|---|---|---|
| 50 | 3200 | $1.887 \cdot 10^{-2}$ | 1.423 |
| 50 | 8864 | $6.330 \cdot 10^{-3}$ | 1.245 |
| 96 | 53496 | $1.871 \cdot 10^{-3}$ | 0.846 |
| 190 | 360208 | $5.101 \cdot 10^{-4}$ | 0.875 |
| 378 | 2385216 | $1.360 \cdot 10^{-4}$ | 0.850 |
| 752 | 17370620 | $3.273 \cdot 10^{-5}$ | 0.805 |

TABLE 2.7.1. Adaptive refinement based on the DM approach with regularisation

the function $\hat{g}(x_2) := -(x_2 - 0.5)^2$. We define the function $g$ on the set $\Gamma_c^{\varepsilon} := [1-\varepsilon, 1] \times \Gamma_C$ by $g(x_1, x_2) = \hat{g}(x_2)$. Let

$$\mathcal{A}(u) := -\Delta u - \varrho\,[g-u]_{+,\Gamma_c^{\varepsilon}}^2$$



(a) $m = 1$ (b) $m = 50$ (c) $m = 100$

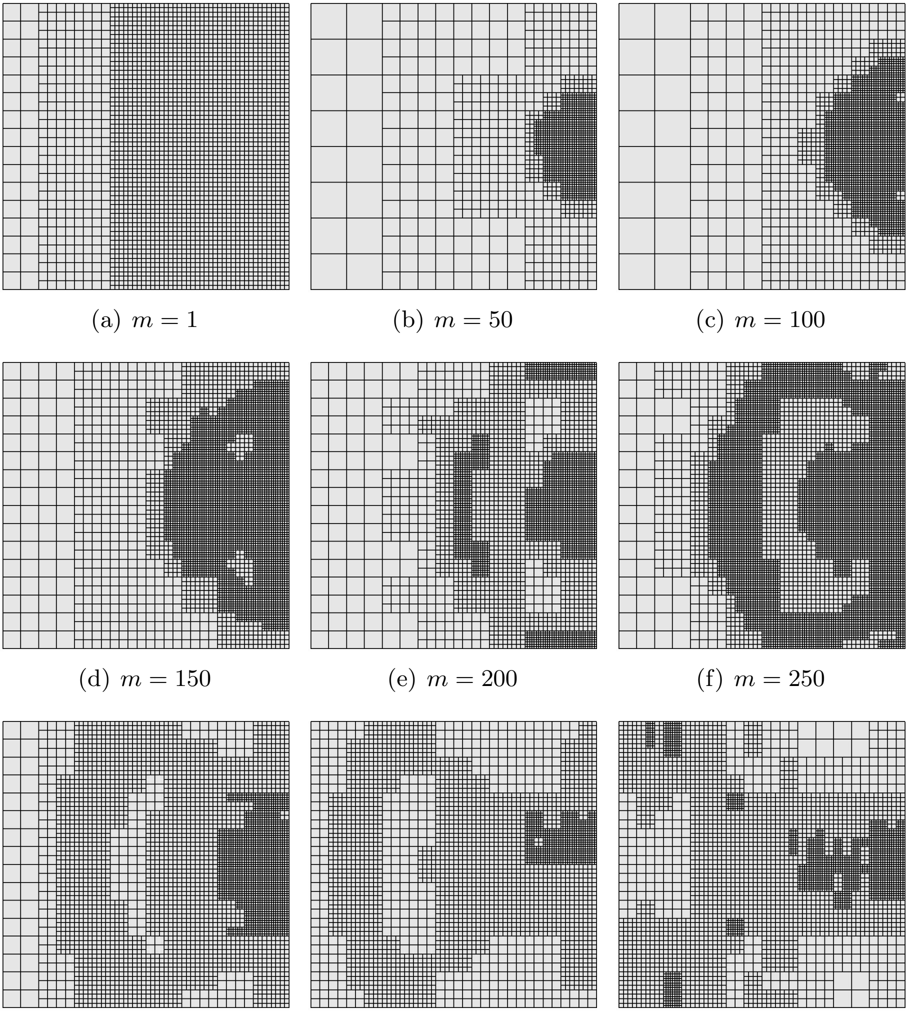
(d) $m = 150$ (e) $m = 200$ (f) $m = 250$

(g) $m = 300$ (h) $m = 350$ (i) $m = 400$

FIGURE 2.7.12. Meshes created during the adaptive refinement process based on the DWR error estimator in the DM approach

be the differential operator with

$$[g-u]_{+,\Gamma_c^\varepsilon}(x) := \begin{cases} 0, & \text{if } x \notin \Gamma_c^\varepsilon, \\ 0, & \text{if } g(x) - u(x) \leq 0, \\ g(x) - u(x), & \text{else.} \end{cases}$$



(a) $m = 1$

(b) $m = 50$

(c) $m = 100$

(d) $m = 150$

(e) $m = 200$

(f) $m = 250$

(g) $m = 300$

(h) $m = 350$

(i) $m = 400$

FIGURE 2.7.13. Meshes created during the adaptive refinement process based on the heuristic error indicator in the DM approach

The penalty parameter is given by $\varrho \in \mathbb{R}_+$. We have chosen a form of the penalty term, which is not usual. However, it fits better in our general framework than penalty terms on the boundary. Furthermore, the

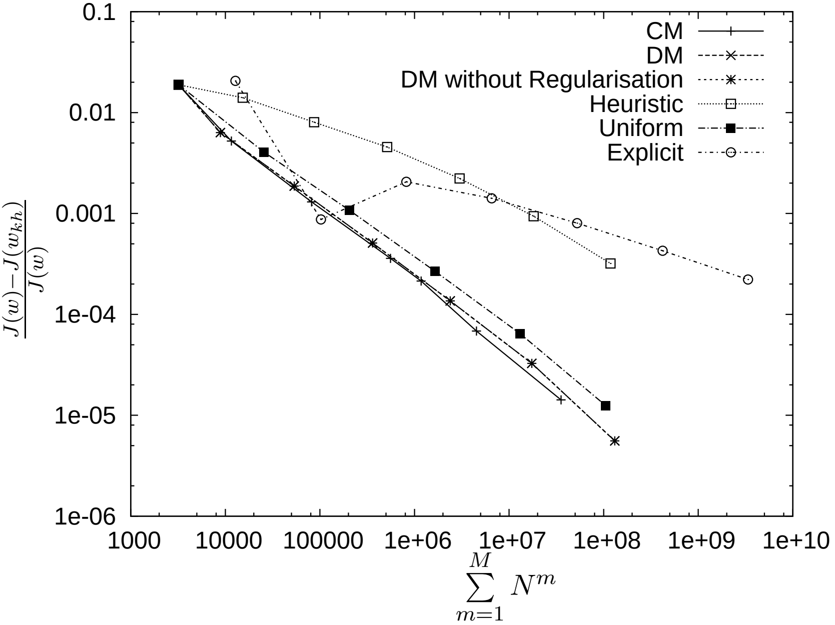


FIGURE 2.7.14. Convergence in the output functional $J$ w.r.t. the total number of mesh cells for different refinement techniques

operator $\mathcal{A}$ is not three times continuously Fréchet differentiable. Thus, we cannot expect the remainder terms to be small. However, the adaptive concept leads to reasonable results. To obtain a reasonable approximation of the underlying dynamic contact problem, the parameters $\varrho$ and $\varepsilon$ have to be chosen in dependence of $h$ and $k$, where $\varrho$ tends towards infinity and $\varepsilon$ towards zero, if $h$ and $k$ tend to zero. In this example, we are interested in the discretisation error for fixed $\varrho = 1000$ and $\varepsilon = 0.125$. The initial conditions are $u_s = 0$ and $v_s := -0.25 \sin(0.5\pi x_1)$. The right hand side $f$ is set to zero. The output functional is given by

$$J(w) := \frac{1}{|B|} \int_B u^2 \, dx$$

with $B = [0, 0.25] \times [0.375, 0.625]$.

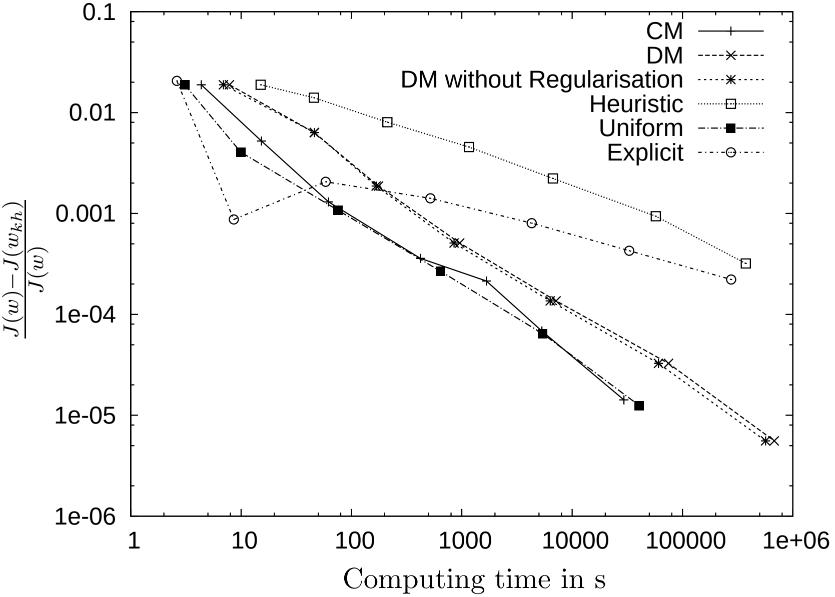


FIGURE 2.7.15. Convergence in the output functional $J$ w.r.t. computing time for different refinement techniques

We begin with the comparison of temporal meshes, which are created during the adaptive refinement process based on the heuristic error indicator and on $\eta_k^i$. We choose a fixed spatial mesh with 4096 elements and an initial time step length of 0.02. We perform five adaptive refinement cycles, where only temporal refinements are permitted. In Figure 2.7.9, the resulting temporal meshes are depicted. At the beginning and at the end of the time interval, the temporal mesh based on $\eta_k^i$ consists of larger time steps. In the middle of the time interval, the same length of the time steps is used.

To compare the different adaptive spatial refinement techniques, we choose a fixed time step length $k = 0.0025$ and also perform five adaptive refinement cycles. In Figure 2.7.10, the mesh created by the CM approach is presented. We observe that the region with $[g-u]_{+,\Gamma_c^\varepsilon} > 0$, $B$, and the connection between these two regions are well resolved. In Figure

2.7.12, meshes of different time steps, which are created by the DM approach, are depicted. In the beginning, the region $[g-u]_{+,\Gamma_c^\varepsilon} > 0$ is well resolved. There, a wave arises. It moves through $\Omega$ from the right to the left. The mesh follows this wave. As soon as the wave enters $B$, only $B$ is well resolved. Looking at Figure 2.7.11, we see that the number of mesh elements also corresponds to this behaviour. In the beginning and in the end, we have a small number of mesh cells, since only small regions have to be refined. In the middle, larger regions have to be resolved and consequently the number of mesh elements increases. The adaptive meshes created by the heuristic error indicator are depicted in Figure 2.7.13. During the whole calculation, the region with $[g-u]_{+,\Gamma_c^\varepsilon} > 0$ is well resolved. Furthermore, we observe additional refinements to resolve the arising wave. At the end, the mesh becomes more and more unstructured. During the whole calculation, the region of interest $B$ is not refined. Looking at Figure 2.7.11, we observe oscillations in the first time steps and in the second half of the calculation.

In Figure 2.7.14, the development of the error measured in $J$ w.r.t. the total number of mesh elements is depicted. We use the numerical solution of the CM approach with $M = 1502$ and

$$\sum_{m=0}^{M} N^m = 287236472$$

as reference value for the calculation of the error. We observe the same convergence rate for the adaptive methods based on the presented error estimator and for the uniform refinement. However, we need less mesh elements in the adaptive approach to achieve the same error. The CM and the DM approach lead to nearly the same results. We observe that no temporal hanging nodes of degree two occur. Consequently, no differences between the regularised adaptive method and the one without regularisation occur. The heuristic error estimator lead to a slower convergence in the first adaptive refinement iterations. Furthermore, it needs much more unknowns to obtain the same error than the adaptive methods based on the presented error estimator. The reason is that the region of interest $B$ is not well resolved. Using the left box rule in the derivation of the cg(1)cg(1) method, an explicit time stepping scheme can be derived. By considering a lumped mass matrix, we obtain a time stepping scheme, which leads to a linear system of equations with a diagonal matrix instead of a nonlinear system of equations. However, this method is not stable and we have to ensure the stability by choosing a smaller time step size, which depends on the spatial mesh width. The explicit form leads to a slower convergence and needs more elements to achieve the same accuracy as the adaptive methods.

In Figure 2.7.14, we measure the computational effort in the total number of mesh elements and observe that the adaptive refinement methods based on the presented error estimator are most efficient. However, we have to perform many additional operations, which need computing time, in the adaptive algorithm. In Figure 2.7.15, we compare the different refinement techniques w.r.t. computing time. We measure the computing time to calculate one result, if the cg(1)cg(1) method or its explicit version are considered. For the adaptive methods, the computing time of the whole adaptive solution algorithm up to a certain result is measured. We roughly need the same computing time to achieve the same accuracy by the CM approach and by uniform refinement. The smaller number of mesh elements suffices to compensate for the additional effort in the adaptive algorithm. The DM approach needs more computing time because of the additional effort for the mesh changes. The comparison of the DM approach with and without regularisation shows that the effort for the mesh regularisation is small.

Table 2.7.1 shows the developement of the adaptive solution process based on the DM approach. In the first iteration only the spatial mesh sequence is refined. In all other iterations the temporal mesh as well as the spatial mesh sequence are refined. The effectivity indices show the accuracy of the error estimate.

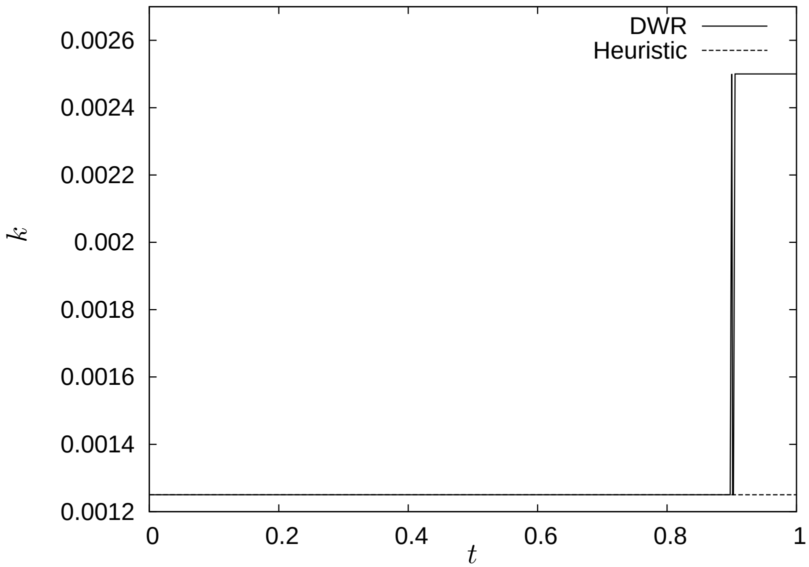


FIGURE 2.7.16. Temporal meshes based on the heuristic error indicator and on the DWR error estimator

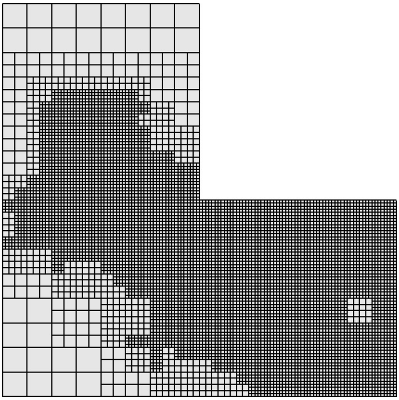


FIGURE 2.7.17. Mesh created by the adaptive refinement based on the CM approach

**2.7.3. Example 2: Spindle-Grinding Wheel-System Model Case.** Before we discuss the example, the constitutive equations of linear elasticity are introduced. As usual, we consider a spatial domain $\Omega \subset \mathbb{R}^2$. The function $u : \Omega \to \mathbb{R}^2$ describes the deformation of the body

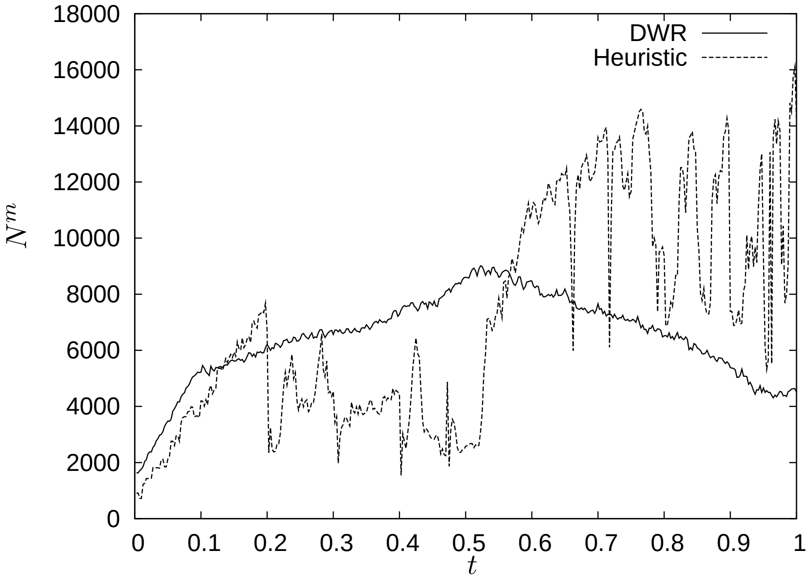


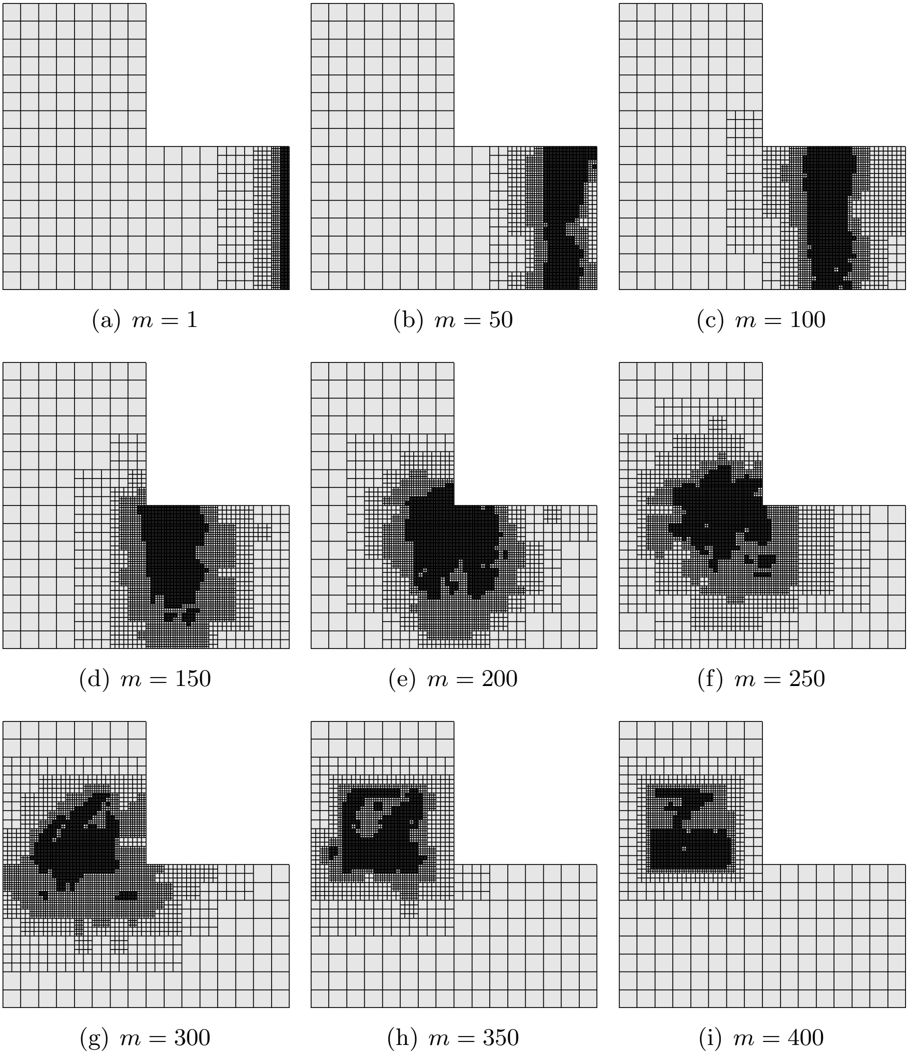
FIGURE 2.7.18. Distribution of the number of mesh elements in the DM approach based on the DWR error estimator and on the heuristic error indicator

| $M$ | $\sum_{m=1}^{M} N^M$ | $E_{\text{rel}}$ | $I_{\text{eff}}$ |
|---|---|---|---|
| 50 | 9600 | $1.063 \cdot 10^{-1}$ | 18.74 |
| 100 | 57600 | $3.483 \cdot 10^{-1}$ | 171.9 |
| 200 | 400800 | $1.570 \cdot 10^{-1}$ | 124.8 |
| 400 | 2976000 | $7.174 \cdot 10^{-2}$ | 77.15 |
| 800 | 22944000 | $2.876 \cdot 10^{-2}$ | 135.3 |
| 1600 | 176140800 | $9.161 \cdot 10^{-4}$ | 88.83 |
| 3196 | 363193440 | $6.921 \cdot 10^{-4}$ | 16.10 |

TABLE 2.7.2. Adaptive refinement based on the DWR error estimator in the CM approach

$\Omega$ under the body force $f$ and the boundary load $q$. We assume small deformations here. Consequently, we work with linear strains, i.e.

$$\varepsilon(u) := \frac{1}{2}\left(\nabla u + \nabla u^T\right) \in \mathbb{R}^{2\times 2}.$$



(a) $m = 1$

(b) $m = 50$

(c) $m = 100$

(d) $m = 150$

(e) $m = 200$

(f) $m = 250$

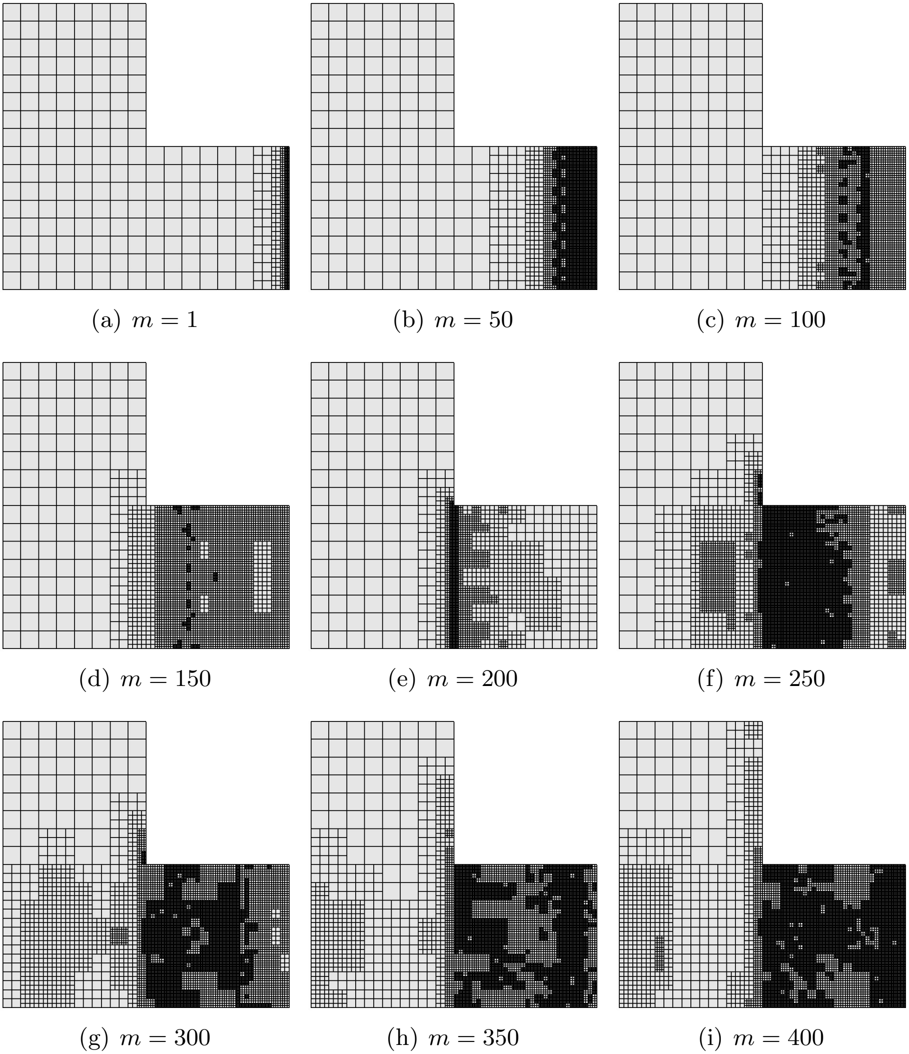
(g) $m = 300$

(h) $m = 350$

(i) $m = 400$

FIGURE 2.7.19. Meshes created during the adaptive refinement process based on the DWR error estimator in the DM approach

The relationship between the strains $\varepsilon(u)$ and the stresses $\sigma(u) \in \mathbb{R}^{2\times 2}$ is assumed to be linear and the plain stress assumption is made. The matrices $\varepsilon$ and $\sigma$ are symmetric. Based on the modulus of elasticity $E$



(a) $m = 1$ (b) $m = 50$ (c) $m = 100$

(d) $m = 150$ (e) $m = 200$ (f) $m = 250$

(g) $m = 300$ (h) $m = 350$ (i) $m = 400$

FIGURE 2.7.20. Meshes created during the adaptive refinement process based on the heuristic error indicator in the DM approach

and Poisson's ratio $\nu \in [0, 0.5)$, we have

$$\begin{pmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{12} \end{pmatrix} = \frac{E}{1-\nu^2} \begin{pmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & 1-\nu \end{pmatrix} \begin{pmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{12} \end{pmatrix}.$$

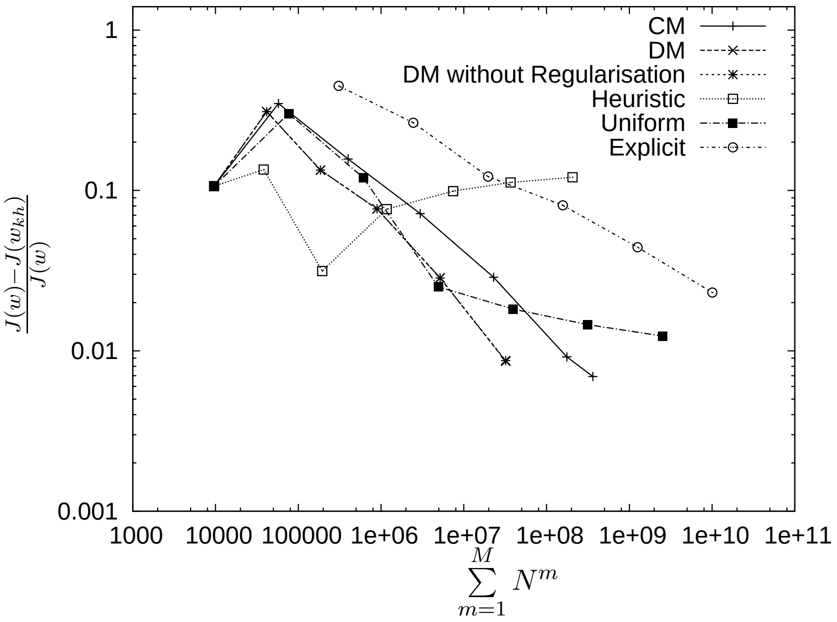


FIGURE 2.7.21. Convergence in the output functional $J$ w.r.t. the total number of mesh cells for different refinement techniques

The bilinear form $a$ is then given by $a(\varphi, \psi) := (\sigma(\varphi), \varepsilon(\psi))$. From the engineering point of view, one is often interested in the so called equivalent stress, which is used to determine the state of the material. A popular one is the von Mises equivalent stress $\sigma_v$. It is defined as

$$\sigma_v = \sqrt{\sigma_{11}^2 + \sigma_{22}^2 + 3\sigma_{12}^2}$$

and used as functional of interest in different examples.

In the introduction, we have mentioned the spindle-grinding wheel-system. The geometry of this system includes several re-entrant corners. Furthermore, it includes jumping material coefficients. As model case, we study an L-shaped domain

$$\Omega := [-0.5, 0] \times [-0.5, 0.5] \cup [0, 0.5] \times [-0.5, 0]$$

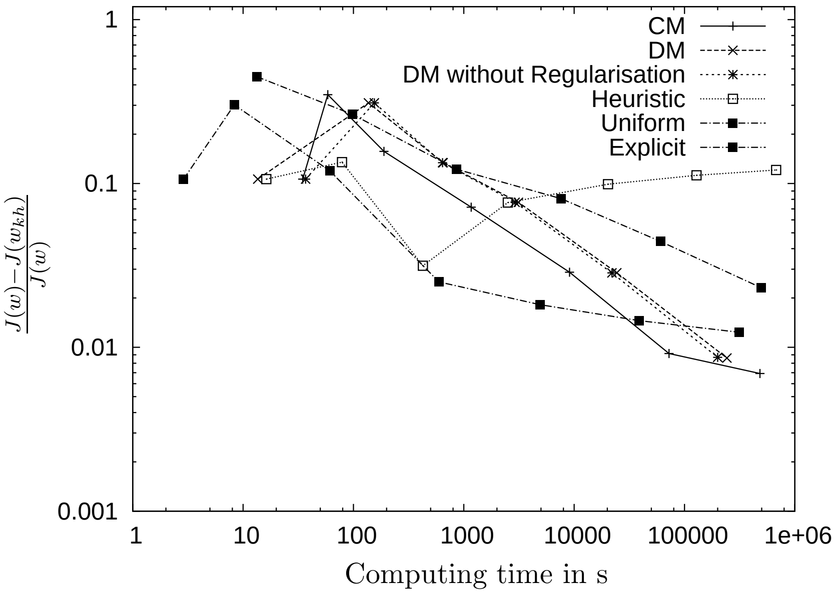


FIGURE 2.7.22. Convergence in the output functional $J$ w.r.t. computing time for different refinement techniques

with

$$\Gamma_D := \{x \in \partial\Omega \,|x_2 = 0.5\}$$

and $\Gamma_N := \partial\Omega\backslash\Gamma_D$ here. The time interval is $I = [0, 1]$. We choose

$$E = E(x) = \rho = \rho(x) = \begin{cases} 10, & \text{if } x_1 < 0, \\ 100, & \text{else,} \end{cases}$$

and set $\nu = 0.33$. The initial conditions are $u_s = v_s = 0$. The body force $f$ is set to zero. We apply the boundary load

$$q(x,t) = \begin{cases} (-100, 0)^\top, & \text{if } t \leq 0.05 \text{ and } x_1 = 0.5, \\ 0, & \text{else.} \end{cases}$$

The functional of interest is

$$J_1\left(w_{kh}\right) := \frac{1}{|B|}\int_B \sigma_v\left(u\right)\, dx$$

with $B := [-0.375, -0.125] \times [0, 0.25]$.

We begin with the comparison of temporal meshes, which are created during the adaptive refinement process based on the heuristic error indicator and on $\eta_k^i$. We choose a fixed spatial mesh with 3072 elements and an initial time step length of 0.02. We perform five adaptive refinement cycles, where only temporal refinements are permitted. In Figure 2.7.16, the resulting temporal meshes are depicted. The heuristic error indicator leads to a global uniform refinement of the temporal mesh. However, larger time step sizes are created at the end of the time interval by the adaptive refinement based on $\eta_k^i$

To compare the different adaptive spatial refinement techniques, we choose a fixed time step length $k = 0.0025$ and perform five adaptive refinement cycles. In Figure 2.7.17, the mesh created by the CM approach is presented. Let

$$\Gamma_N^1 := \{x \in \partial\Omega \,|x_1 = 0.5\}$$

be the boundary, where nonhomogeneous Neumann boundary conditions are prescribed. The part of the boundary $\Gamma_N^1$, $B$, and the connection between these two regions are well resolved. In Figure 2.7.19, meshes of different time steps, which are created by the DM approach, are depicted. In the beginning, the part of the boundary $\Gamma_N^1$ is well resolved. There, a wave arises. It moves through $\Omega$ from the right to the left. The mesh follows this wave. At the discontinuity of the material coefficients, the main part of the wave is reflected. The mesh does not follow the main part, but the smaller part, which moves further to the left. The region $B$ also becomes an important factor. The mesh is more and more concentrated in $B$. Looking at Figure 2.7.18, we see that the number of mesh elements is small in the beginning and in the end. There, only small regions have to be refined. In the middle, larger regions have to be resolved and consequently the number of mesh elements increases. The adaptive meshes created by the heuristic error indicator are depicted in Figure 2.7.20. In the beginning, $\Gamma_N^1$ is well resolved. Then the mesh follows the outgoing wave. In contrast to the meshes created by the DM approach based on $\eta$, the meshes based on the heuristic error indicator resolve the reflected wave. We mainly observe refinements for $x_1 \geq 0$. In the second part of the calculation, the meshes become more unstructured. This is observed in Figure 2.7.18, too. There, large oscillations in the number of mesh elements are depicted, especially, for $t \geq 0.5$.

In Figure 2.7.21, the development of the relative error $E_{\text{rel}}$ w.r.t. to the total number of mesh cells is depicted. We use the numerical solution with $M = 3197$ time steps and a total number of unknowns of

$$\sum_{m=1}^{M} N^m = 1415035392$$

in the CM approach as reference value. In this example, the adaptive method based on the DM approach is most efficient. Only few temporal hanging nodes of degree two occur. Consequently, the difference between the DM approach with and without regularisation is small. The CM approach is more efficient than every other method except the DM approach. The adaptive method based on the heuristic error indicator does not provide a good approximation of the functional of interest $J$, since it does not lead to refinements in the neighbourhood of $B$, see Figure 2.7.20. The different methods are compared in terms of computing time in Figure 2.7.22. The adaptive methods based on the DM approach as well as on the CM approach need less computing time than the explicit method, the method based on uniform refinement, and the adaptive method based on the heuristic error indicator. The computing time is comparable for the CM and the DM approach. We also observe in this example that a small amount of computing time is needed for the temporal regularisation. Looking at Table 2.7.2, we observe that the error estimate is not as accurate as in the other examples. The reason is the complex functional of interest and the relatively coarse discretisation.

CHAPTER 3

# Adaptive Finite Elements for Static Contact Problems

In this chapter, an adaptive finite element method for static contact problems is discussed. We consider the static case in order to illustrate the approach to a posteriori error estimation, which we will use in the next chapter for the error control of dynamic contact problems. The aim is to derive an a posteriori error bound w.r.t. an arbitrary output functional. We use the dual problem known from elliptic equations for the a posteriori error estimate. The same dual problem was used in [**107**], but there a different and less accurate a posteriori error bound was derived. One way to derive goal-oriented a posteriori error estimates for contact problems is based on a dual problem, which was first introduced by Natterer [**86**]. Contributions to this technique are made in [**26, 103, 106**]. In all mentioned references, linear output functionals are considered in contrast to our approach, which can be applied to nonlinear output functionals. In our derivation, we employ an auxiliary problem, which corresponds to an elliptic variational equality. The idea of using this auxiliary problem was introduced by Braess [**31**] for static obstacle problem and generalised by Schröder [**102, 104**], who even applied it to static frictional contact problems. The error, however, was only estimated in the norms corresponding to the trial spaces. Other approaches to adaptive finite elements for static contact problems have been presented in a lot of contributions [**5, 14, 38, 59, 66, 88, 110**]. In particular, an adaptive scheme for two-body contact is contained in [**116**]. Convergence results for adaptive algorithms in the context of obstacle problems are proven in [**32, 33, 105**].

The outline of this chapter is as follows: After the introduction of the continuous formulation of the static contact problem in Section 3.1, a mixed finite element discretisation is presented. The goal-oriented a posteriori error estimator is derived in Section 3.3. Finally, we apply the presented error estimator to a simplified and a full Signorini problem and discuss the numerical results.

## 3.1. Continuous Formulation

Let $\Omega \subset \mathbb{R}^2$ be the basic domain. The boundary $\partial\Omega$ of $\Omega$ is divided into three mutually disjoint parts $\Gamma_D$, $\Gamma_C$ and $\Gamma_N$ with positive measure. Homogeneous Dirichlet and Neumann boundary conditions are prescribed on the closed set $\Gamma_D$ and on the relatively open set $\Gamma_N$, respectively. Contact may take place on the sufficiently smooth set $\Gamma_C$, $\bar{\Gamma}_C \subset \complement\Gamma_D$. See, for instance, [**70**], Section 5.3, for more details. The rigid foundation is parameterised by a function $g : \Gamma_C \to \mathbb{R} \cup \{-\infty\}$, $g \in H^{1/2}(\Gamma_c)$. We consider the restriction $u \geq g$ on $\Gamma_C$, $u \leq g$ can be treated analogously.

The strong formulation reads

$$\begin{aligned}
-\Delta u &= f && \text{in } \Omega \\
u &= 0 && \text{on } \Gamma_D \\
\frac{\partial u}{\partial \nu} &= 0 && \text{on } \Gamma_N \\
u - g &\geq 0 && \text{on } \Gamma_C && (3.1.1) \\
\frac{\partial u}{\partial \nu} &\leq 0 && \text{on } \Gamma_C && (3.1.2) \\
\frac{\partial u}{\partial \nu}(u - g) &= 0 && \text{on } \Gamma_C. && (3.1.3)
\end{aligned}$$

Relation (3.1.1) represents the non-penetration condition. The direction of the contact stresses is determined by (3.1.2). The complementarity condition is given by (3.1.3). It ensures that contact stresses only occur if the gap is closed.

If the solution $u \in H^1(\Omega, \Gamma_D) \cap C^2(\Omega)^1$, the strong formulation is equivalent to the variational inequality

$$\forall \varphi \in K: \quad a(u, \varphi - u) \geq (f, \varphi - u), \tag{3.1.4}$$

where $a(\cdot,\cdot) := (\nabla\cdot, \nabla\cdot)$ is a uniformly elliptic, symmetric, and continuous bilinear form. The solution $u$ is an element of

$$K := \left\{ \varphi \in V := H^1(\Omega, \Gamma_D) \middle| \, \gamma_{|\Gamma_C}(\varphi) \geq g \right\}.$$

One way of solving elliptic variational inequalities is given by their mixed formulation. The Lagrange multiplier may be interpreted as contact force. The variational inequality (3.1.4) is equivalent to the following mixed form: Find $(u, \lambda) \in V \times \Lambda$ with

$$\begin{aligned} \forall \varphi \in V: \quad a(u, \varphi) + \left\langle \lambda, \gamma_{|\Gamma_C}(\varphi) \right\rangle &= (f, \varphi) && (3.1.5) \\ \forall \mu \in \Lambda: \quad \left\langle \mu - \lambda, \gamma_{|\Gamma_C}(u) - g \right\rangle &\leq 0. && (3.1.6) \end{aligned}$$

Here, $\Lambda$ is the dual cone of the set

$$G := \left\{ \varphi \in H^{1/2}(\Gamma_C) \middle| \, \varphi \leq 0 \right\},$$

i.e.

$$\Lambda := \left\{ \mu \in H^{-1/2}(\Gamma_C) \, | \forall \varphi \in G: \, \langle \mu, \varphi \rangle \geq 0 \right\}.$$

The equivalence of the two formulations is a well-known conclusion from the general theory of minimisation problems in Hilbert spaces presented, e.g., in [**37, 46**]. Furthermore, the existence of a unique weak solution $(u, \lambda)$ of (3.1.5-3.1.6) is assured, if the inf-sup-condition

$$\alpha \|\mu\|_{-\frac{1}{2},\Gamma_C} \leq \sup_{v \in H^1(\Omega,\Gamma_D), \|v\|_1 = 1} \left\langle \mu, \gamma_{|\Gamma_C}(v) \right\rangle \tag{3.1.7}$$

holds for a constant $\alpha > 0$ and all $\mu \in H^{-1/2}(\Gamma_C)$. The inf-sup-condition (3.1.7) follows directly from the closed range theorem and the surjectivity of $\gamma_{|\Gamma_C}$, c.f. for instance [**104**].

[1] This can be generalised to $u \in H^1(\Omega, \Gamma_D) \cap H^2(\Omega)$.

## 3.2. Mixed Finite Element Discretisation

A finite element approach is applied to discretise (3.1.5-3.1.6). Bilinear basis functions on the mesh $\mathbb{T}_h$ are used for the finite element space

$$V_h := \left\{ \varphi \in C\left(\bar{\Omega}, \Gamma_D\right) \middle| \forall \mathcal{T} \in \mathbb{T}_h : \varphi_{|\mathcal{T}} \in Q_1\left(\mathcal{T}; \mathbb{R}\right) \right\}.$$

The discrete Lagrange multipliers are piecewise constant on the boundary mesh $\mathbb{B}_H$, the trial set is

$$\Lambda_H := \left\{ \varphi \in G \middle| \forall \mathcal{T} \in \mathbb{B}_H : \varphi_{|\mathcal{T}} \in \mathcal{P}_0\left(\mathcal{T}; \mathbb{R}\right) \right\}.$$

The index $H$ indicates that coarser meshes may be chosen for the Lagrange multiplier. In our calculations, we use $H = 2h$ for stability reasons. We assume that $\mathbb{T}_h$ has patch structure. Consequently, the mesh $\mathbb{B}_H$ is well defined and consists of the edges on $\Gamma_C$ of the patch elements. A detailed study of this finite element approach for solving contact problems is contained in [**104**].

The discrete problem reads: Find $(u_h, \lambda_H) \in V_h \times \Lambda_H$ with

$$\begin{aligned} \forall \varphi_h \in V_h : \quad & a\left(u_h, \varphi_h\right) + \left\langle \lambda_H, \gamma_{|\Gamma_C}\left(\varphi_h\right)\right\rangle &=& \left(f, \varphi_h\right) & (3.2.1) \\ \forall \mu_H \in \Lambda_H : \quad & \left\langle \mu_H - \lambda_H, \gamma_{|\Gamma_C}\left(u_h\right) - g\right\rangle &\leq& 0. & (3.2.2) \end{aligned}$$

The system (3.2.1-3.2.2) leads to the following saddle point problem in $\mathbb{R}^m$:

$$\begin{aligned} K\bar{u} + B\bar{\lambda} &= \bar{f} \\ \forall \bar{\mu} \in \mathbb{R}^{\tilde{m}}_{\leq 0} : \quad \left(\bar{\mu} - \bar{\lambda}\right)^T \left(B^T \bar{u} - \bar{g}\right) &\leq 0, \end{aligned}$$

Here, $K \in \mathbb{R}^{m \times m}$ is the stiffness matrix. The matrix $B \in \mathbb{R}^{m \times \tilde{m}}$ represents the dual pairing in (3.2.2). Using the Schur complement

$$\bar{u} := K^{-1}\left(\bar{f} - B\bar{\lambda}\right),$$

the saddle point problem is rewritten as the quadratic program (QP)

$$\begin{aligned} \min_{\bar{\lambda} \in \mathbb{R}^{\tilde{m}}} \quad & \tfrac{1}{2}\bar{\lambda}^T Q \bar{\lambda} - \bar{\lambda}^T \left[B^T K^{-1} \bar{f} - \bar{g}\right] \\ \text{s. t.} \quad & \bar{\lambda} \leq 0. \end{aligned}$$

The matrix $Q := B^T K^{-1} B$ is symmetric and positive semidefinite. Thus, a standard QP with simple sign constraints has to be solved. This solution approach has several advantages: There are only simple sign constraints. Furthermore, the number of active optimisation variables is much smaller than in a direct approach.

For the numerical solution of this QP any QP-solver can be used, which only requires a user-defined routine for the calculation of $Q\bar{\lambda}$. We use SQOPT [**51**]. The inner linear system of equations is solved either by the direct solver Umfpack, see [**43**], or by a multigrid method, see, e.g., [**30**].

A unique discrete solution $(u_h, \lambda_H)$ on a uniform mesh exists if the discrete analogon of (3.1.7) holds for a mesh independent constant $\alpha > 0$, i.e.

$$\forall \mu_H \in \Lambda_H : \quad \alpha \left\|\mu_H\right\|_{\frac{1}{2},\Gamma_C} \leq \sup_{v_h \in V_h, \|v_h\|_1 = 1} \left(\mu_H, \gamma_{|\Gamma_C}(v_h)\right)_{\Gamma_C}. \tag{3.2.3}$$

For our discretisation, the quotient $h/H$ has to be sufficiently small for (3.2.3) to hold [**58**]. Let $\mathbb{T}_h$ and $\mathbb{B}_H$ be regular and quasi-uniform meshes with $h/H$ constant and sufficiently small. Furthermore, $\gamma(u) \in H^{1,\infty}(\mathcal{T})$ holds for all $\mathcal{T} \in \mathbb{B}_H$ and the number of switching-points between $\gamma(u)$ and $g$ is finite. Then the a priori error estimate

$$\|u - u_h\|_1 + \|\lambda - \lambda_H\|_{-\frac{1}{2},\Gamma_C} \leq Ch \tag{3.2.4}$$

holds with a constant $C > 0$ for the presented discretisation (Theorem 4.1 in [**56**]). Here, some additional properties of the contact situation are assumed.

## 3.3. A Posteriori Error Estimation

In this section, we derive an a posteriori error estimate for the discretisation error w.r.t. some three times Fréchet differentiable output functional $J(\cdot) \in V^\star$, which may be nonlinear. For this purpose, we define the associated dual problem

$$\forall \varphi \in V : \quad a(\varphi, z) = J'(u)(\varphi)$$

with a solution $z \in V$. The corresponding finite element solution $z_h$ is given by

$$\forall \varphi_h \in V_h : \quad a(\varphi_h, z_h) = J'(u_h)(\varphi_h).$$

We specify a second auxiliary problem by

$$\forall \varphi \in V : \quad a(u_\star, \varphi) = (f, \varphi) - \left\langle \lambda_H, \gamma_{|\Gamma_C}(\varphi) \right\rangle \tag{3.3.1}$$

with a solution $u_\star \in V$. Equation (3.3.1) is a linear and elliptic variational equation with nonhomogeneous Neumann boundary conditions. The auxiliary problem (3.3.1) corresponds to (3.1.5), but with the discrete Lagrange multiplier $\lambda_H$ instead of $\lambda$. We observe that $u_h$ also is a discrete solution of (3.3.1). The dual problem associated to $u_\star$ has the solution $z_\star$ defined by

$$\forall \varphi \in V : \quad a(\varphi, z_\star) = J'(u_\star)(\varphi).$$

If $J$ is linear, $z_\star$ is equal to $z$. Since $u_{\star,h} = u_h$, it holds $z_{\star,h} = z_h$.

After these preliminary definitions, we are now able to derive the a posteriori error estimate. First, we split the error into two parts:

$$\begin{aligned} & J(u) - J(u_h) \\ = \quad & J(u) - J(u_\star) + J(u_\star) - J(u_h) \quad . \end{aligned}$$

Both parts are examined separately. The first one is written as

$$J(u) - J(u_\star) = \int_0^1 J'(u_\star + se_\star)(e_\star)\, ds$$

with $e_\star := u - u_\star$. The trapezoidal rule is used to approximate the integral, which leads to

$$\begin{aligned} & \textstyle\int_0^1 J'(u_\star + se_\star)(e_\star)\, ds \\ = \quad & \tfrac{1}{2}\left[J'(u_\star)(e_\star) + J'(u)(e_\star)\right] \\ & + \tfrac{1}{2}\textstyle\int_0^1 J'''(u_\star + se_\star)(e_\star, e_\star, e_\star)\, s(s-1)\, ds \\ =: \quad & \tfrac{1}{2}\left[J'(u_\star)(e_\star) + J'(u)(e_\star)\right] + \mathcal{R}_\star^{(3)}, \end{aligned}$$

where the remainder term $\mathcal{R}_\star^{(3)}$ is of third order in the error $e_\star$. Using the definitions of the dual solutions $z$ and $z_\star$ and the equations (3.2.1) and (3.3.1), we obtain

$$\begin{array}{rc} & J'(u_\star)(e_\star) \\ = & a(e_\star, z_\star) \\ = & (f, z_\star) - \left\langle \lambda, \gamma_{|\Gamma_C}(z_\star) \right\rangle - (f, z_\star) + \left\langle \lambda_H, \gamma_{|\Gamma_C}(z_\star) \right\rangle \\ = & -\left\langle \lambda - \lambda_H, \gamma_{|\Gamma_C}(z_\star) \right\rangle \end{array}$$

and

$$J'(u)(e_\star) = -\left\langle \lambda - \lambda_H, \gamma_{|\Gamma_C}(z) \right\rangle.$$

Defining the contact residual by

$$\rho_c(\lambda, \lambda_H)(z, z_\star) := -\left\langle \lambda - \lambda_H, \frac{1}{2}\gamma_{|\Gamma_C}(z_\star + z) \right\rangle,$$

we finally arrive at the error identity

$$J(u) - J(u_\star) = \rho_c(\lambda, \lambda_H)(z, z_\star) + \mathcal{R}_\star^{(3)}. \tag{3.3.2}$$

If $J$ is linear, the contact residual simplifies to

$$\rho_c(\lambda, \lambda_H)(z) := -\left\langle \lambda - \lambda_H, \gamma_{|\Gamma_C}(z) \right\rangle$$

and the remainder term is zero.

The second part $J(u_\star) - J(u_h)$ corresponds to the error in a linear and elliptic equation and can be handled by the standard DWR technique or by every other goal-oriented error estimator for linear elliptic variational equations. For the sake of completeness, we briefly outline the DWR approach. Using a result from [**17**], we obtain

$$\begin{array}{rc} & J(u_\star) - J(u_h) \\ = & \frac{1}{2}\left[(f, z - \tilde{z}_h) - \left\langle \lambda_H, \gamma_{|\Gamma_C}(z - \tilde{z}_h) \right\rangle - a(u_h, z - \tilde{z}_h)\right] \\ & +\frac{1}{2}\left[J'(u_h)(u - \tilde{u}_h) - a(u - \tilde{u}_h, z_h)\right] + \mathcal{R}_h^{(3)} \\ =: & \frac{1}{2}\left[\rho(u_h, \lambda_H)(z - \tilde{z}_h) + \rho^\star(z_h)(u - \tilde{u}_h)\right] + \mathcal{R}_h^{(3)} \end{array} \tag{3.3.3}$$

for arbitrary $\tilde{z}_h, \tilde{u}_h \in V_h$. The remainder term $\mathcal{R}_h^{(3)}$ is given by

$$\mathcal{R}_h^{(3)} = \frac{1}{2}\int_0^1 J'''(u_h + se)(e, e, e)\, s\,(s-1)\; ds$$

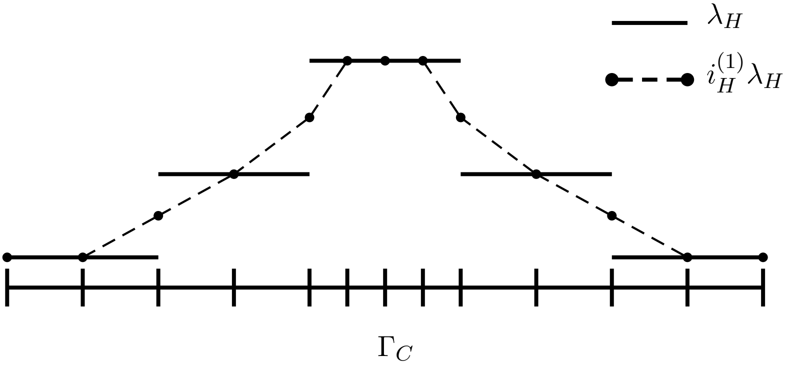


FIGURE 3.3.1. Linear interpolation of the Lagrange multiplier

with $e := u_\star - u_h$ and is of third order in the error $e$. For a linear $J$, the remainder term is zero.

In order to obtain a computationally evaluable error estimate, we approximately evaluate the error identities (3.3.2) and (3.3.3) by

$$\begin{aligned} & J(u) - J(u_h) \\ = \quad & \rho_c(\lambda, \lambda_H)(z, z_\star) + \mathcal{R}_\star^{(3)} \\ & + \tfrac{1}{2}\left[\rho(u_h, \lambda_H)(z - \tilde{z}_h) + \rho^\star(z_h)(u - \tilde{u}_h)\right] + \mathcal{R}_h^{(3)} \\ \approx \quad & \rho_c\left(i_H^{(1)}\lambda_H, \lambda_H\right)\left(i_{2h}^{(2)} z_h, i_{2h}^{(2)} z_h\right) \\ & + \tfrac{1}{2}\left[\rho(u_h, \lambda_H)\left(i_{2h}^{(2)} z_h - z_h\right) + \rho^\star(z_h)\left(i_{2h}^{(2)} u_h - u_h\right)\right] \\ =: \quad & \eta = \eta_c + \eta_e. \end{aligned}$$

Here, we made use of the fact that $z_{\star,h} = z_h$. Furthermore, $i_H^{(1)}$ defines a linear interpolant of the Lagrange multiplier $\lambda_H$ based on the idea used in the ZZ-error estimator [**120**]. The Lagrange multiplier $\lambda_H \in \Lambda_H$ is piecewise constant on possibly coarser mesh $\mathbb{B}_H$. The interpolant $i_H^{(1)}\lambda_H$ of $\lambda_H$ is built on the original mesh $\mathbb{B}_h$. The basis values, which correspond to the nodal values, are defined as the mean value of $\lambda_H$ on the two adjacent cells. The approach is illustrated in Figure 3.3.1. The operator $i_{2h}^{(2)}$ specifies a biquadratic interpolant of the discrete primal and dual solution. It is the same as in Chapter 1.

Another possibility for the approximate evaluation of $\eta_c$ is given by

$$\bar{\eta}_c := \rho_c \left( i_H^{(1)} \lambda_H, \lambda_H \right) (z_h, z_h) .$$

More sophisticated interpolation operators are possible. However, they do not significantly improve the results. We will compare the two approaches in the next section.

Since the error estimate will be used as basis for an adaptive refinement algorithm, the terms $\eta_c$ and $\eta_e$ have to be localised. The term $\eta_c$ is evaluated on the edges of the boundary and the value is then added to the estimated value of the adjacent element, which contains this edge. The term $\eta_e$ is localised by the techniques presented in Section 2.1.1.

It is possible to derive the error estimators $\eta$ and $\bar{\eta}$ without the help of the auxiliary problem (3.3.1). However, we would not be able to use an arbitrary goal-oriented error estimator but would have to use a fixed one.

## 3.4. Numerical Results

In this section, we discuss two examples of static contact problems. The first one is a simplified Signorini problem. To show that the presented approach also is applicable to full Signorini examples, we investigate such an example in the second part of this section.

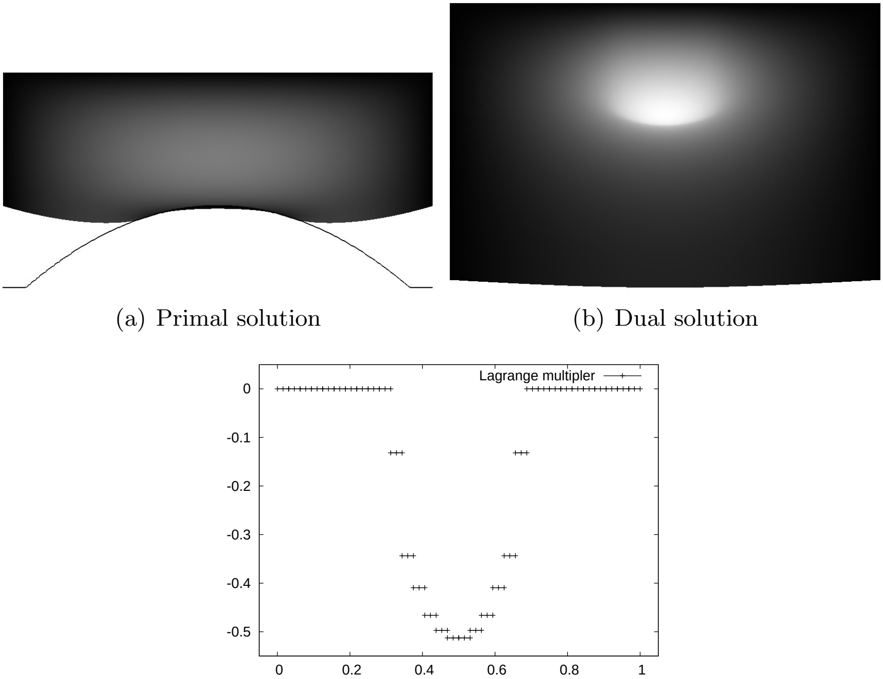
### 3.4.1. Example 1: Simplified Signorini Problem.

Let

$$\Omega := [0,1]^2$$

be the basic domain. The boundary is divided as follows:

$$\begin{aligned} \Gamma_D &:= \{ x = (x_1, x_2) \in \partial\Omega | \, x_1 = 0 \vee x_2 = 0 \vee x_2 = 1 \} , \\ \Gamma_N &:= \{ x = (x_1, x_2) \in \partial\Omega | \, x_1 = 1 \wedge (x_2 < 0.1 \vee x_2 > 0.9) \} , \\ \Gamma_C &:= \{ x = (x_1, x_2) \in \partial\Omega | \, x_1 = 1 \wedge 0.1 \leq x_2 \leq 0.9 \} . \end{aligned}$$



(a) Primal solution

(b) Dual solution

(c) Lagrange multiplier

FIGURE 3.4.1. Geometry of the simplified Signorini example based on $J_1$

The right hand side is given by $f \equiv -1$ and the rigid foundation by $g(x_2) = -(x_2 - 0.5)^2$. The output functional is set to

$$J_1(\varphi) := \frac{1}{|B|} \int_B \exp(-u)\, dx$$

with $B := [0, 0.25] \times [0.375, 0.625]$. The setting is illustrated in Figure 3.4.1. The reference value of the output functional is calculated on a uniform mesh with 1048576 cells. We use the error estimator presented in the last section as basis for an adaptive refinement process, where we localise the estimator by integration by parts as well as filtering. As refinement strategy, we apply the optimal mesh strategy, which is presented in Section 2.2. The error estimator presented in [**104**] is applied to

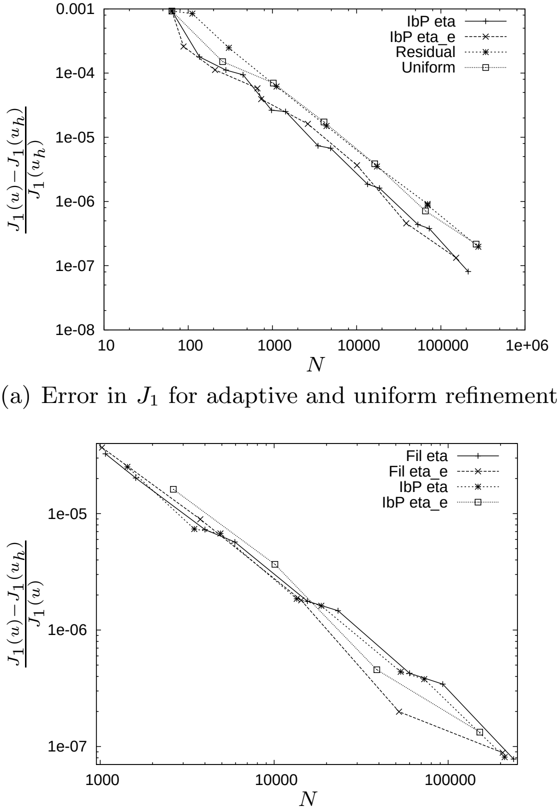
| $N$ | $E_{\mathrm{rel}}$ | $I_{\mathrm{eff}}$ | $\bar{I}_{\mathrm{eff}}$ | $I^e_{\mathrm{eff}}$ |
|---|---|---|---|---|
| 64 | $9.240 \cdot 10^{-4}$ | 0.910 | 0.912 | 0.988 |
| 136 | $1.791 \cdot 10^{-4}$ | 0.634 | 0.634 | 0.681 |
| 280 | $1.114 \cdot 10^{-4}$ | 1.147 | 1.147 | 1.210 |
| 448 | $9.402 \cdot 10^{-5}$ | 0.987 | 0.987 | 1.000 |
| 976 | $2.641 \cdot 10^{-5}$ | 1.091 | 1.091 | 1.106 |
| 1432 | $2.520 \cdot 10^{-5}$ | 1.034 | 1.034 | 1.047 |
| 3472 | $7.379 \cdot 10^{-6}$ | 1.231 | 1.231 | 1.248 |
| 4912 | $6.754 \cdot 10^{-6}$ | 1.136 | 1.136 | 1.153 |
| 13480 | $1.855 \cdot 10^{-6}$ | 1.212 | 1.212 | 1.230 |
| 18640 | $1.616 \cdot 10^{-6}$ | 1.145 | 1.145 | 1.157 |
| 53344 | $4.379 \cdot 10^{-7}$ | 1.182 | 1.182 | 1.195 |
| 72856 | $3.787 \cdot 10^{-7}$ | 1.092 | 1.092 | 1.095 |
| 211288 | $8.107 \cdot 10^{-8}$ | 0.883 | 0.883 | 0.886 |

TABLE 3.4.1. Error in $J_1$ and effectivity index for adaptive refinement based on integration by parts

| $N$ | $E_{\mathrm{rel}}$ | $I_{\mathrm{eff}}$ | $\bar{I}_{\mathrm{eff}}$ | $I^e_{\mathrm{eff}}$ |
|---|---|---|---|---|
| 64 | $9.240 \cdot 10^{-4}$ | 0.910 | 0.912 | 0.989 |
| 136 | $1.791 \cdot 10^{-4}$ | 0.634 | 0.634 | 0.681 |
| 256 | $1.452 \cdot 10^{-4}$ | 1.048 | 1.048 | 1.088 |
| 448 | $1.078 \cdot 10^{-4}$ | 1.023 | 1.023 | 1.036 |
| 1072 | $3.265 \cdot 10^{-5}$ | 0.991 | 0.991 | 1.000 |
| 1600 | $2.024 \cdot 10^{-5}$ | 1.051 | 1.051 | 1.068 |
| 4000 | $7.294 \cdot 10^{-6}$ | 1.185 | 1.185 | 1.202 |
| 5944 | $5.707 \cdot 10^{-6}$ | 1.144 | 1.144 | 1.159 |
| 15520 | $1.770 \cdot 10^{-6}$ | 1.178 | 1.178 | 1.192 |
| 23272 | $1.467 \cdot 10^{-6}$ | 1.121 | 1.121 | 1.134 |
| 60040 | $4.254 \cdot 10^{-7}$ | 1.166 | 1.165 | 1.178 |
| 93088 | $3.432 \cdot 10^{-7}$ | 1.053 | 1.053 | 1.056 |
| 237904 | $7.800 \cdot 10^{-8}$ | 0.863 | 0.863 | 0.866 |

TABLE 3.4.2. Error in $J_1$ and effectivity index for adaptive refinement based on filtering

the problem to compare the presented approach with estimators, which measure the error in a global norm.

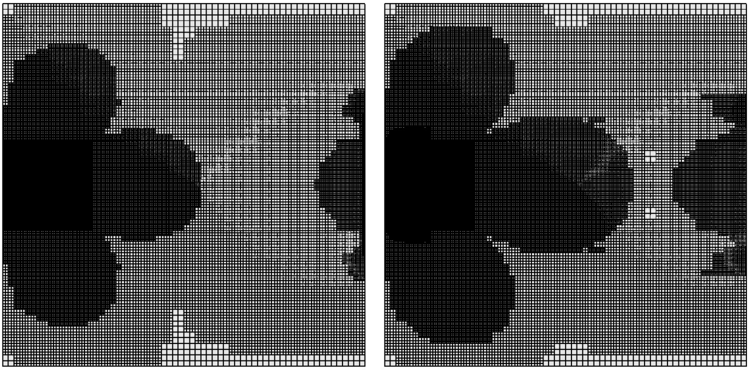


(a) Error in $J_1$ for adaptive and uniform refinement

(b) Comparison of $\eta_c$ and $\eta_e$ for $J_1$

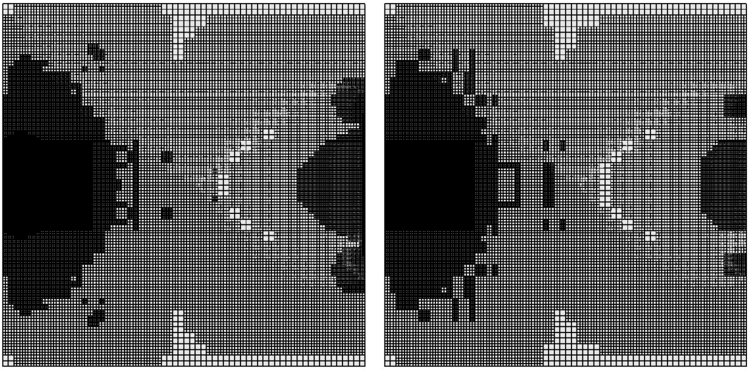
Figure 3.4.2. Convergence of different refinement methods

In Figure 3.4.2, the convergence behaviour of different refinement techniques is compared. Since we know that the finite element method is of optimal order in $h$, see (3.2.4), one cannot expect a better order of the adaptive methods. However, the adaptive methods should lead to a more accurate approximation than the uniform refinement. We see in Figure 3.4.2 that the adaptive methods based on the presented approach and based on $\eta_e$ lead to a smaller error than the uniform refinement. But with the adaptive refinement based on the residual error estimator, we obtain the same results as for the uniform refinement. This is not



(a) Based on $\eta$, 10th iteration (b) Based on $\eta_e$, 6th iteration

FIGURE 3.4.3. Adaptive meshes based on localisation by filtering



(a) Based on $\eta$, 11th iteration (b) Based on $\eta_e$, 7th iteration

FIGURE 3.4.4. Adaptive meshes based on localisation by integration by parts

astonishing, because the refinement strategy leads to a globally uniform refinement in this case after a first adaptive refinement in the contact zone, see Figure 3.4.5. The results based on $\eta_e$ and $\eta$ are nearly the same, which is explained by Figure 3.4.6(a). There, we observe that $\eta_c$ is a factor 10 smaller than $\eta_e$ and that $\eta_c$ and $\eta_e$ are of the same order.

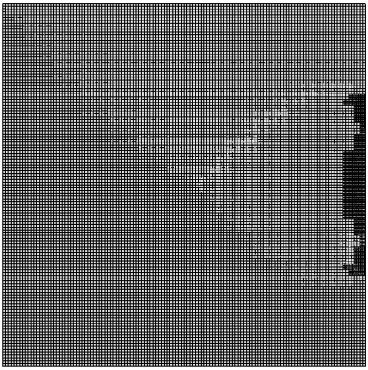
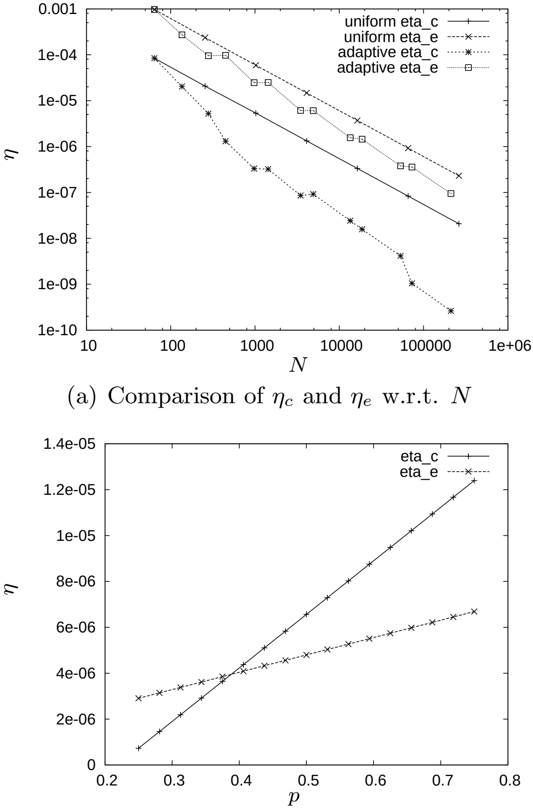


FIGURE 3.4.5. Adaptive mesh based on the residual error estimator, 6th iteration

| $N$ | $E_{\text{rel}}$ | $I_{\text{eff}}$ | $I^e_{\text{eff}}$ |
|---|---|---|---|
| 64 | $1.931 \cdot 10^{-2}$ | 16.70 | 27.07 |
| 136 | $9.619 \cdot 10^{-3}$ | 37.52 | 35.06 |
| 208 | $5.804 \cdot 10^{-3}$ | 27.04 | 28.67 |
| 352 | $3.461 \cdot 10^{-3}$ | 20.88 | 22.00 |
| 520 | $5.469 \cdot 10^{-4}$ | 3.572 | 3.632 |
| 784 | $4.284 \cdot 10^{-4}$ | 3.026 | 2.850 |
| 1432 | $4.755 \cdot 10^{-4}$ | 13.49 | 12.07 |
| 1888 | $1.236 \cdot 10^{-4}$ | 2.839 | 2.514 |
| 2464 | $4.770 \cdot 10^{-5}$ | 1.119 | 0.975 |
| 3400 | $5.500 \cdot 10^{-5}$ | 1.230 | 1.127 |
| 6208 | $2.261 \cdot 10^{-6}$ | 0.255 | 0.212 |
| 8080 | $2.231 \cdot 10^{-5}$ | 1.854 | 1.651 |
| 10792 | $3.455 \cdot 10^{-5}$ | 2.766 | 2.605 |
| 14128 | $2.541 \cdot 10^{-5}$ | 2.028 | 1.934 |
| 26752 | $6.989 \cdot 10^{-6}$ | 2.922 | 2.710 |
| 34720 | $6.974 \cdot 10^{-6}$ | 1.617 | 2.213 |

TABLE 3.4.3. Error in $J_{2,0.75}$ and effectivity index for adaptive refinement

In Figure 3.4.2(b), the different localisation methods are compared for $\eta$ and $\eta_e$. We observe no decisive differences. Thus, we will only consider localisation by integration by parts in the rest of this section. In



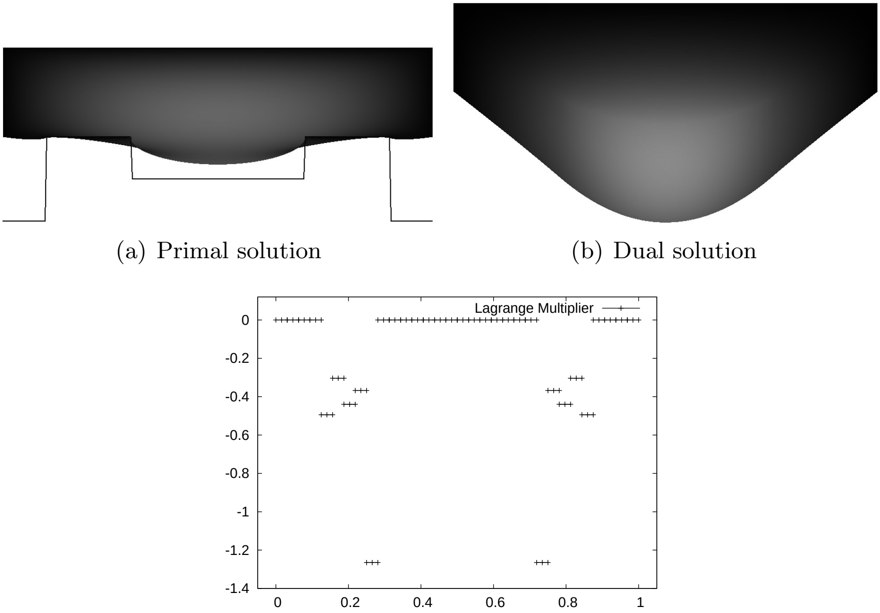
(a) Comparison of $\eta_c$ and $\eta_e$ w.r.t. $N$

(b) Comparison of $\eta_c$ and $\eta_e$ w.r.t. $p$

FIGURE 3.4.6. Comparison of $\eta_c$ and $\eta_{DWR}$

Figure 3.4.3, adaptive meshes based on $\eta$ and $\eta_e$ localised by filtering are compared. The overall structure of the meshes is the same for both estimators. The only difference is that the adaptive refinement based on $\eta$ needs twice the number of adaptive refinement iterations. The reason is the localisation of $\eta_c$, which leads to an adaptive refinement of the contact area in every second refinement iteration. The corresponding results for localisation by integration by parts are depicted in Figure 3.4.4.

In Table 3.4.1, the quantitative results for the adaptive refinement based on the presented error estimator localised by integration by parts are



(a) Primal solution

(b) Dual solution

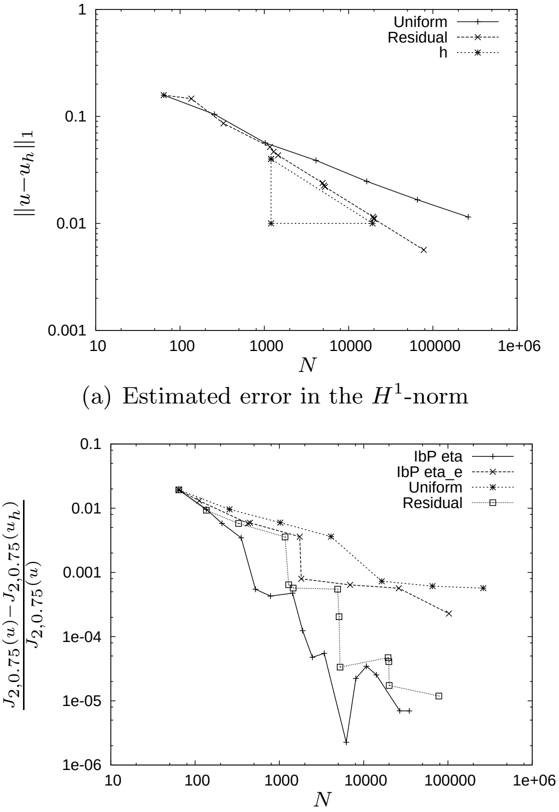
(c) Lagrange multiplier

FIGURE 3.4.7. Geometry of the simplified Signorini example based on $J_{2,0.75}$

shown. The overestimation of the error in the last step results from the fact that the solution is too close to the reference solution. We also observe that $\eta_c \ll \eta_e$, since the effectivity indices based on $\eta$ and $\eta_e$ do not differ much. Furthermore, we have tested $\bar{\eta}_c$. The effectivity indices $I_{\text{eff}}$ and $\bar{I}_{\text{eff}}$ as well as the adaptive meshes are the same for $\eta$ and $\bar{\eta} := \bar{\eta}_c + \eta_e$. Hence, it is sufficient to work with the computationally cheaper estimator $\bar{\eta}$. The quantitative results based on filtering are presented in Table 3.4.2. They correspond to the results based on integration by parts.

In Figure 3.4.6(a), we have seen that $\eta_c$ is a factor 10 smaller than $\eta_e$. Let us have a closer look at the ratio between $\eta_c$ and $\eta_e$. We consider the functional

$$J_{2,p}\left(\varphi\right) := \frac{1}{|B_p|} \int_{B_p} \exp\left(-u\right)\, dx,$$

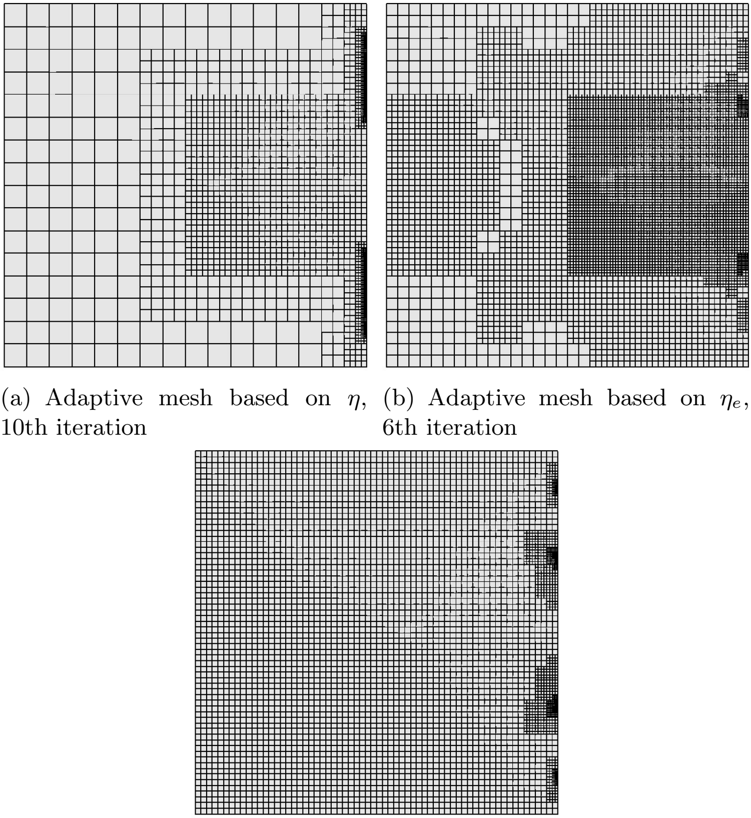


(a) Estimated error in the $H^1$-norm

(b) Error in $J_{2,0.75}$

FIGURE 3.4.8. Convergence of different refinement methods

where $B_p := [p-0.25, p+0.25] \times [0.25, 0.75]$ with $p \in [0.25, 0.75]$. The only difference between $J_1$ and $J_{2,p}$ is the carrier $B_p$. By the parameter $p$, we control the distance between the contact boundary $\Gamma_C$ and $B_p$. In Figure 3.4.6(b), $\eta_c$ and $\eta_e$ are compared w.r.t. $p$ on a uniform mesh with 16384 cells. Thus, the estimated error in the Lagrange multiplier $i_H^{(1)}\lambda_H - \lambda_H$ is constant and $\eta_c$ depends only on $z_h$. We observe that $\eta_c$ and $\eta_e$ increase linearly with increasing $p$. But the gradient of $\eta_c$ is greater than the gradient of $\eta_e$. For $p = 0.25$, $\eta_e$ dominates in $\eta$. But this changes and for $p \gtrsim 0.4$, $\eta_c$ dominates.



(a) Adaptive mesh based on $\eta$, 10th iteration

(b) Adaptive mesh based on $\eta_e$, 6th iteration

(c) Adaptive mesh based on residual estimator, 8th iteration

FIGURE 3.4.9. Adaptive meshes for different adaptive refinement methods

Finally, we consider a discontinuous rigid foundation

$$g(x_2) = \begin{cases} 0, & \text{for } x_2 < 0.3 \text{ and } x_2 > 0.7, \\ -0.1, & \text{else.} \end{cases}$$

The output functional is $J_{2,0.75}$. The numerical solution of this problem and of the corresponding dual problem are depicted in Figure 3.4.7. Since the rigid foundation is discontinuous, we do not find the optimal order of convergence in the $H^1$- norm, see Figure 3.4.8(a), where the estimated

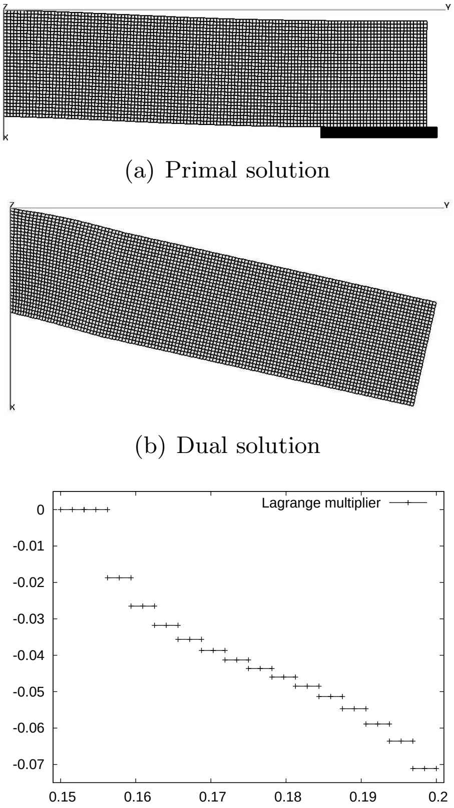
| $N$ | $E_{\text{rel}}$ | $I_{\text{eff}}$ | $I^e_{\text{eff}}$ |
|---|---|---|---|
| 64 | $1.768 \cdot 10^{-2}$ | 0.537 | 0.539 |
| 136 | $4.202 \cdot 10^{-3}$ | 0.348 | 0.349 |
| 340 | $3.711 \cdot 10^{-3}$ | 0.740 | 0.741 |
| 856 | $1.410 \cdot 10^{-3}$ | 0.783 | 0.784 |
| 1984 | $5.241 \cdot 10^{-4}$ | 0.657 | 0.657 |
| 4588 | $2.311 \cdot 10^{-4}$ | 0.602 | 0.602 |
| 11728 | $1.096 \cdot 10^{-4}$ | 0.741 | 0.741 |
| 32596 | $1.721 \cdot 10^{-5}$ | 0.346 | 0.346 |

TABLE 3.4.4. Relative error in $J$ and effectivity index for adaptive refinement based on integration by parts

error in the $H^1$-norm is depicted. By adaptive mesh refinement based on the residual error estimator, we obtain the optimal order $\mathcal{O}(h)$. This gain of the adaptive methods is also obvious in Figure 3.4.8(b), where the error is measured in $J_{2,0.75}$. In contrast to the first example, the adaptive method based on $\eta_e$ is insufficient, since it does not lead to an increased efficiency of the discretisation. The reason behind this is that the solution strongly depends on the discretisation of the contact area. As shown in Figure 3.4.9, this area is well resolved by the adaptive methods based on $\eta$ and the residual error estimator but not by the one based on $\eta_e$. The adaptive method based on the residual error estimator resolves the boundary of the contact zone, whereas the one based on $\eta$ resolves the whole contact zone. In Table 3.4.3, the quantitative results are presented. The effectivity indices are not as good as in the first example. But this could not be expected because of the underlying problem structure.

**3.4.2. Example 2: Full Signorini Problem.** We consider a 2D full Signorini problem in this section. The basic domain is

$$\Omega = [0, 0.05] \times [0, 0.2]$$



(a) Primal solution

(b) Dual solution

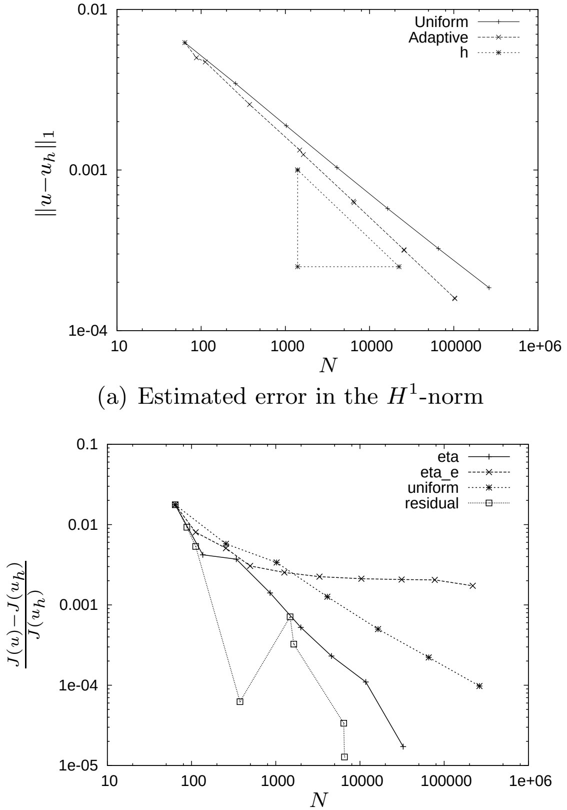
(c) Lagrange multiplier

FIGURE 3.4.10. Geometry of the Signorini example

with

$$\begin{aligned}
\Gamma_D &= \{x=(x_1,x_2)|\, x_2=0\}\,,\\
\Gamma_C &= \{x=(x_1,x_2)|\, x_1=0.05 \wedge x_2 \geq 0.15\}\,,\\
\Gamma_N &= \partial\Omega \backslash (\Gamma_D \cup \Gamma_C)\,.
\end{aligned}$$

We set $q=0$ and $f=(0.5,0)^T$. The material parameters are $E=10$ and $\nu=0.33$. The rigid foundation is given by $g=0.055$. We are interested in the von Mises equivalent stress $\sigma_v$, see Section 2.7.3, in the domain



(a) Estimated error in the $H^1$-norm

(b) Error in $J$

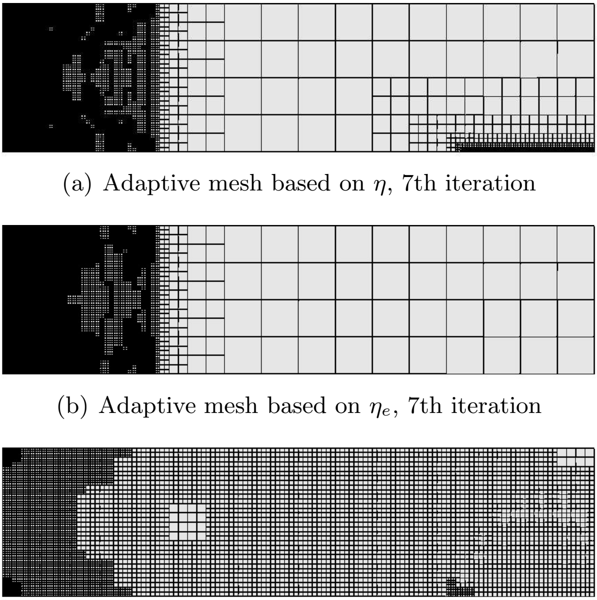
FIGURE 3.4.11. Convergence of different refinement methods

$B = [0, 0.05]^2$, i.e.

$$J(u) := \frac{1}{|B|} \int_B \sigma_v^2\,(u)\;dx.$$

The numerical solution is illustrated in Figure 3.4.10.

Figure 3.4.11(a) shows that we do not obtain the optimal order of convergence in the case of a uniform refinement. We consider the estimated $H^1$-norm here. However, it is knwon that the convergence rate usually is not optimal for Signorini problems and that the contact situation and the domain have to be smooth to obtain convergence of order $\mathcal{O}(h)$. It is recovered by the adaptive method based on the residual estimator. The



(a) Adaptive mesh based on $\eta$, 7th iteration

(b) Adaptive mesh based on $\eta_e$, 7th iteration

(c) Adaptive mesh based on residual estimator, 7th iteration

FIGURE 3.4.12. Adaptive meshes for different adaptive refinement methods

convergence behaviour in $J$ of different adaptive methods is compared in Figure 3.4.11(b). We observe that the adaptive method based on $\eta_e$ converges very slowly. The reason becomes apparent, if we look at the adaptive meshes (Figure 3.4.12(b)). The contact area is not refined at all, thus the main influence factor of this example is inaccurately calculated. The convergence of the adaptive method based on the residual estimator is the fastest one. It leads to a well resolved contact zone. Furthermore, $B$ is adaptively refined, because stress peaks occur in $(0, 0)$ and $(0.05, 0)$. The adaptive mesh is depicted in Figure 3.4.12(c). The error estimator $\eta$ leads to a well resolved contact area in contrast to $\eta_e$, see Figure 3.4.12(a). In Table 3.4.4, we observe that $\eta_c \ll \eta_e$. Consequently, we have to consider $\eta_c$ in the adaptive refinement process, although its contribution to $\eta$ is small.

CHAPTER 4

# Contact/Impact Problems

In the preceding chapter, we have discussed adpative finite elements for static contact problems. These problems are extended in this chapter by including inertia forces. A lot of numerical approaches to solve dynamic contact problems, which are based on different problem formulations, exist: In [**118**], the penalty-method is used to solve the discrete problems. Special contact elements in combination with Lagrange multipliers are presented in [**15**]. Other techniques for smoothing and stabilizing the computation with special finite elements, e.g., Mortar finite elements, are presented in [**55, 84, 92**]. In [**45**], the Newmark scheme is used with an additional $L^2$-projection for stabilisation. Algorithms for dynamic contact/impact problems based on the energy- and momentum conservation are derived in [**6, 76**]. An additive splitting of the acceleration into two parts, representing the interior forces and the contact forces, is the basis of the methods introduced in [**68, 90**]. In [**41, 93, 108**] algorithms based on variational inequalities and optimisation algorithms are presented. A space-time finite element approach is discussed in [**21**]. Detailed surveys of this topic can be found in the monographs [**75**], Chapter 7, and [**117**], Chapter 9.

But in contrast to the static case, the analysis of the continuous formulation and of the discretisation schemes is in the early stages. In the monographs [**75, 89**] and in [**53, 82**] as well as in the references therein, models for dynamic contact/impact problems are discussed. Beside the geometric contact conditions known from the static case, the dynamics of the contact are modelled by an impact law. The impact law specifies the development of the energy during the contact. If the energy is conserved, the contact is purely elastic. The contact is called inelastic, if energy is dissipated. We use a purely elastic model. Different possibilities to include the impact law in the problem formulation are discussed in [**53, 75**]. The existence and uniqueness of a solution is still an open question. In the linear viscoelastic case, the existence of at least one weak solution is assured [**3**]. In this case, the continuous dependence on the data is proven, too [**73**]. The convergence of the discretisation schemes is another open question. Recently, a consistency result, which is the basis for a priori error estimates, has been derived in the context of the stabilized Newmark method [**74**]. The development of adaptive methods for dynamic contact problems also is in the early stages. In [**19, 24, 25**], a posteriori error estimates in the $H^1$-norm based on the semi-discrete problem formulation are discussed.

## 4.1. Dynamic Contact/Impact Problems

In this section, we present the continuous problem formulation. The model presented, for instance, in [**40, 75**] is used. Especially, the discussion in [**75**] is helpful. As in Chapter 3, we restrict ourselves to simplified Signorini problems. The generalisation to Signorini problems is straightforward. We present such an example in Section 4.4.

The basic domain is $\Omega \subset \mathbb{R}^2$ and the boundary $\partial\Omega = \Gamma_D \cup \Gamma_N \cup \Gamma_C$ is subdivided in the same manner as in Section 3.1. The rigid foundation is

$$g : \Gamma_C \to \mathbb{R} \cup \{-\infty\}\,, \quad g \in H^{1/2}(\Gamma_C)\,.$$

Again, we consider the case $u \geq g$. In the description of dynamic contact problems, we assume homogeneous Neumann boundary conditions to ease the notation.

If the solution $u$ is sufficiently smooth, it fulfils the equations of the wave equation

$$\begin{aligned} \rho\ddot{u} - \Delta u &= f && \text{in } \Omega \times I && (4.1.1)\\ u &= 0 && \text{on } \Gamma_D \times I && (4.1.2)\\ \frac{\partial u}{\partial \nu} &= 0 && \text{on } \Gamma_N \times I && (4.1.3)\\ u(0) &= u_s && \text{in } \Omega && (4.1.4)\\ \dot{u}(0) &= v_s && \text{in } \Omega && (4.1.5) \end{aligned}$$

and the contact conditions

$$\begin{aligned} u - g &\geq 0 && \text{on } \Gamma_C \times I && (4.1.6)\\ \frac{\partial u}{\partial \nu} &\leq 0 && \text{on } \Gamma_C \times I && (4.1.7)\\ \frac{\partial u}{\partial \nu}(u-g) &= 0 && \text{on } \Gamma_C \times I && (4.1.8)\\ \frac{\partial u}{\partial \nu}\dot{u} &= 0 && \text{on } \Gamma_C \times I. && (4.1.9) \end{aligned}$$

We set $\rho \equiv 1$ for notational simplicity. In comparison to the static contact conditions (3.1.1-3.1.3), the persistency condition (4.1.9) has been added. It corresponds to the complementarity condition (4.1.8), only the gap $u - g$ is replaced by the gap rate $\dot{u}$. We see in Proposition 4.1.3 that the persistency condition ensures the conservation of energy. Therewith, the impact is purely elastic. To clarify the physical meaning of the persistency condition, we examine the following equivalent form (c.f. [**75**]) of the contact conditions (4.1.6-4.1.9) on $\Gamma_C \times I$:

$$u - g > 0 \;\Rightarrow\; \begin{cases} \frac{\partial u}{\partial \nu} = 0 \\ \dot{u} \text{ unconstrained,} \end{cases} \qquad (4.1.10)$$

$$u - g = 0 \;\Rightarrow\; \begin{cases} \frac{\partial u}{\partial \nu} \leq 0 \\ \dot{u} \geq 0 \\ \frac{\partial u}{\partial \nu}\dot{u} = 0. \end{cases} \qquad (4.1.11)$$

The condition (4.1.10) says that the movement of the membrane is free, if the gap is open. If the gap is closed, then condition (4.1.11) means that the contact stresses are negative, which is known from the static context. But now, the velocity has to be greater than zero, too. This ensures that $u - g \geq 0$ holds. Furthermore, we recover the persistency condition (4.1.9).

After having discussed the strong formulation, we now define the weak one. We use the notation from Section 1.1, if not stated otherwise.

DEFINITION 4.1.1. A function $u \in K := \left\{\varphi \in W \,\middle|\, \gamma_{|\Gamma_C}(\varphi) \geq g\right\}$ is a weak solution of the dynamic contact/impact problem if and only if

$$\begin{aligned} \forall \varphi \in \tilde{K}: \quad \langle \ddot{u}, \varphi - u\rangle + a\left(u, \varphi - u\right) \quad &\geq \quad \left(f, \varphi - u\right) && (4.1.12)\\ \left(\frac{\partial u}{\partial \nu}, \gamma_{|\Gamma_C}\left(\dot{u}\right)\right)_{\Gamma_C} \quad &= \quad 0 && (4.1.13)\\ u(0) \quad &= \quad u_s && (4.1.14)\\ \dot{u}(0) \quad &= \quad v_s && (4.1.15)\end{aligned}$$

holds for a.e. time $t \in I$ with

$$\begin{aligned} W \quad &:= \quad \left\{\varphi \in L^2\left(I; H^1\left(\Omega, \Gamma_D\right)\right) \,\middle|\, \begin{array}{l} \dot{\varphi} \in L^2\left(I; H^1\left(\Omega, \Gamma_D\right)\right), \\ \ddot{\varphi} \in L^2\left(I; \left(H^1\left(\Omega, \Gamma_D\right)\right)^\star\right) \end{array}\right\}, \\ \tilde{K} \quad &:= \quad \left\{\varphi \in H^1\left(\Omega, \Gamma_D\right) \middle| \gamma_{|\Gamma_C}(\varphi) \geq g\right\}. \end{aligned}$$

Furthermore, $f \in L^2\left(I; L^2\left(\Omega\right)\right)$, $u_s \in H^1\left(\Omega, \Gamma_D\right)$, and $v_s \in L^2\left(\Omega\right)$.

The bilinear form $a$ is given by $a(\cdot, \cdot) := (\nabla \cdot, \nabla \cdot)$. If the solution $u$ is sufficiently smooth, (4.1.12) and (4.1.14-4.1.15) are equivalent to (4.1.1-4.1.8), see for instance [**108**]. The generalised persistency condition (4.1.13) (c.f. [**73**]) corresponds to the pointwise persistency condition (4.1.9) due to the sign conditions in (4.1.10-4.1.11). Since the mixed formulation of the dynamic contact problem is more suited for the analysis of the problem and for its discretisation, we write the variational inequality (4.1.12) in the mixed form:

Definition 4.1.2. The tuple $(u, \lambda) \in W \times \tilde{\Lambda}$ is a weak solution of the dynamic contact/impact problem, if and only if

$$\begin{aligned} \langle \ddot{u}, \varphi \rangle + a(u, \varphi) + \left\langle \lambda, \gamma_{|\Gamma_C}(\varphi) \right\rangle &= (f, \varphi) && (4.1.16) \\ \left\langle \mu - \lambda, \gamma_{|\Gamma_C}(u) - g \right\rangle &\leq 0 \\ \left\langle \lambda, \gamma_{|\Gamma_C}(\dot{u}) \right\rangle &= 0 && (4.1.17) \\ u(0) &= u_s \\ \dot{u}(0) &= v_s \end{aligned}$$

hold for all $\varphi \in H^1(\Omega, \Gamma_D)$, all $\mu \in G^\star$, and a.e. $t \in I$. Here, $G^\star$ is the dual cone of the set

$$G := \left\{ \mu \in H^{1/2}(\Gamma_C) \middle| \mu \leq 0 \right\},$$

see Section 3.1, and $\tilde{\Lambda}$ is given by $\tilde{\Lambda} := L^2(I; G^\star)$.

The mixed formulation and its equivalence to the variational inequality formulation are discussed, e.g., in [**6, 91**]. In particular, the equality of $\frac{\partial u}{\partial \nu}$ and $\lambda$ is considered. Now, we examine the energy conservation:

Proposition 4.1.3. *If the right hand side $f$ is zero the total energy is conserved in the dynamic contact problem.*

Proof of Proposition 4.1.3. We test equation (4.1.16) by $\dot{u}$ and obtain

$$\begin{aligned} 0 &= \langle \ddot{u}, \dot{u} \rangle + a(u, \dot{u}) + \left\langle \lambda, \gamma_{|\Gamma_C}(\dot{u}) \right\rangle \\ &= \frac{\partial}{\partial t}\frac{1}{2}(\dot{u}, \dot{u}) + \frac{\partial}{\partial t}\frac{1}{2}a(u, u) + \left\langle \lambda, \gamma_{|\Gamma_C}(\dot{u}) \right\rangle . \end{aligned}$$

Hence, the temporal derivative of the total energy is given by

$$\frac{\partial}{\partial t} E_{tot} = - \left\langle \lambda, \gamma_{|\Gamma_C}(\dot{u}) \right\rangle .$$

Because of condition (4.1.17), the term on the right hand side vanishes and therewith, the total energy is constant. □

Remark 4.1.4. The linear and the angular momentum are also conserved, see [**75**].

## 4.2. Finite Element Discretisation in Space and Time

After the formulation of the weak problem in an adequate form, we apply a finite element approach in space and time for discretisation. An essential result is that all properties from the continuous case carry over to the discrete formulation. We present a solution method for the discrete problem and analyse it by means of a model problem. The space-time finite element approach is similar to the discretisation scheme developed in [**76**]. There a finite difference method is used for the temporal discretisation. Furthermore, the solution algorithms for the discrete optimisation problems are different.

**4.2.1. A Continuous Petrov-Galerkin Method.** The basis for the discretisation is the following weak problem formulation:

Definition 4.2.1. The functions $(w,\lambda) = ((u,v),\lambda) \in (U \times V) \times \Lambda$ are a weak solution of the dynamic contact/impact problem, if and only if

$$\begin{aligned} \forall \varphi = (\psi,\chi) \in U \times V: \qquad & A(w,\varphi) + \Big(\big(\lambda, \gamma_{|\Gamma_C}(\chi)\big)_{\Gamma_C}\Big) = 0 \qquad (4.2.1)\\ \forall \mu \in \bar{\Lambda},\ t \in I: \qquad & \big(\mu - \lambda(t), \gamma_{|\Gamma_C}(u(t)) - g\big)_{\Gamma_C} \leq 0\\ \forall t \in I: \qquad & \big(\lambda(t), \gamma_{|\Gamma_C}(\dot{u}(t))\big)_{\Gamma_C} = 0 \end{aligned}$$

holds with

$$\begin{aligned} A(w,\varphi) &:= ((v-\dot{u},\psi)) + ((\dot{v},\chi)) + (a\,(u,\chi)) - ((f,\chi))\\ &\quad + (u(0)-u_s,\chi(0)) + (v(0)-v_s,\psi(0))\,,\\ U &:= \Big\{\psi \in L^2\left(I;H^1\left(\Omega,\Gamma_D\right)\right) \Big| \dot{\psi} \in L^2\left(I;H^1\left(\Omega,\Gamma_D\right)\right)\Big\}\,,\\ V &:= \left\{\chi \in L^2\left(I;H^1\left(\Omega,\Gamma_D\right)\right) \middle| \dot{\chi} \in L^2\left(I;L^2\left(\Omega\right)\right)\right\}\,,\\ \Lambda &:= \Big\{\mu \in L^2\Big(I;H^{1/2}\left(\Gamma_C\right)\Big)\Big|\,\mu \leq 0\Big\}\,,\\ \bar{\Lambda} &:= \Big\{\mu \in H^{1/2}\left(\Gamma_C\right)\Big|\,\mu \leq 0\Big\}\,. \end{aligned}$$

For the discretisation of the displacement $u$ and the velocity $v$, we use the same spaces as in Section 1.2. The assumptions on the trial sets $U$, $V$, and

$\Lambda$ are somewhat stronger than the ones on $W$ and $\tilde{\Lambda}$ in Definition 4.1.2. They are only made to ease the notation. The spatial mesh sequence and the temporal mesh are created just like the ones in Section 1.2. The only new set is the discrete counterpart of $\Lambda$. With the spatial trial set

$$\Lambda_H^m := \left\{\mu \in \bar{\Lambda} \,\middle|\, \forall \mathcal{T} \in \mathbb{B}_H^m : \mu_{|\mathcal{T}} \in \mathcal{P}_0\left(\mathcal{T}, \mathbb{R}_{\leq 0}\right)\right\},$$

see Section 3.2, we define

$$\Lambda_{kH} := \left\{\mu_{kH} \in L^2\left(I; L^2\left(\Gamma_C\right)\right) \,\middle|\, \mu_{kH|_{I_m}} \in \mathcal{P}_0\left(I_m; \Lambda_H^m\right)\right\}.$$

Here, $\mathbb{B}_H^m$ is the boundary mesh connected to $\mathbb{T}_h^m$. As in Chapter 3, we choose $H = 2h$ for stability reasons. The discretisation reads:

Definition 4.2.2. The functions

$$\left(w_{kh}, \lambda_{kH}\right) = \left(\left(u_{kh}, v_{kh}\right), \lambda_{kH}\right) \in \left(V_{kh} \times V_{kh}\right) \times \Lambda_{kH}$$

are a discrete solution of the dynamic contact/impact problem, if and only if

$$\begin{aligned} A_{kh}(w_{kh}, \varphi_{kh}) + \left(\left(\lambda_{kH}, \gamma_{|\Gamma_C}(\chi_{kh})\right)_{\Gamma_C}\right) &= 0 && (4.2.2)\\ \left(\mu_H - \lambda_{kH}, \gamma_{|\Gamma_C}\left(u_{kh}\right) - g\right)_{\Gamma_C} &\leq 0 && (4.2.3)\\ \left(\lambda_{kH}, \gamma_{|\Gamma_C}\left(\dot{u}_{kh}\right)\right)_{\Gamma_C} &= 0 && (4.2.4) \end{aligned}$$

holds with

$$\begin{aligned} A_{kh}\left(w_{kh}, \varphi_{kh}\right) = & \sum_{m=1}^{M} \left\{\left(\left(v_{kh} - \dot{u}_{kh}, \psi_{kh}\right)\right)_{I_m} + \left(\left(\dot{v}_{kh}, \chi_{kh}\right)\right)_{I_m}\right\} \\ & + \sum_{m=1}^{M} \left\{\left(a\left(u_{kh}, \chi_{kh}\right)\right)_{I_m} - \left(\left(f, \chi_{kh}\right)\right)_{I_m}\right\} \\ & + \left(u_{kh}^0 - u_s, \chi_{kh}^0\right) + \left(v_{kh}^0 - v_s, \psi_{kh}^0\right). \end{aligned}$$

Equation (4.2.2) has to be valid for all $\varphi_{kh} = (\psi_{kh}, \chi_{kh}) \in W_{kh} \times W_{kh}$, inequality (4.2.3) for all $m \in \{0, 1, \ldots, M\}$, all $\mu_H \in \Lambda_H^m$, and all $t \in I_m$, and equation (4.2.4) for all $t \in I$.

We have discretised the dynamic contact/impact problem in the usual way. Thus, we expect as usual that the properties of the continuous solution carry over to the discrete one. In the following proposition, we discuss the conservation of the total energy:

PROPOSITION 4.2.3. *If the right hand side $f$ is zero and if $V_h^{m-1} \subseteq V_h^m$ for all $m \in \{0, 1, \ldots, M\}$, then the total energy is constant.*

As in Chapter 1, we assume $V_h^{m-1} \subseteq V_h^m$. Thus, the energy conservation is disturbed by the adaptive methods and we have to pay attention to this fact, i.e. the regularisation algorithm presented in Section 2.3 are applied.

PROOF OF PROPOSITION 4.2.3. We test equation (4.2.2) on a subinterval $I_m$ by $(\dot{v}_{kh}, \dot{u}_{kh})$[1] and obtain

$$\begin{aligned}
0 &= \left(\left(v_{kh}, \dot{v}_{kh}\right)\right)_{I_m} + \left(a\left(u_{kh}, \dot{u}_{kh}\right)\right)_{I_m} + \left(\left(\lambda_{kH}, \gamma_{|\Gamma_C}(\dot{u}_{kh})\right)_{\Gamma_C}\right)_{I_m} \\
&= \frac{1}{2}\left(v_{kh}^m, v_{kh}^m\right) - \frac{1}{2}\left(v_{kh}^{m-1}, v_{kh}^{m-1}\right) + \frac{1}{2}a\left(u_{kh}^m, u_{kh}^m\right) \\
&\quad - \frac{1}{2}a\left(u_{kh}^{m-1}, u_{kh}^{m-1}\right) + k_m\left(\lambda_{kH}^m, \gamma_{|\Gamma_C}(\dot{u}_{kh}^m)\right)_{\Gamma_C}.
\end{aligned}$$

Consequently, it holds

$$E_{tot}^m - E_{tot}^{m-1} = -k_m\left(\lambda_{kH}^m, \gamma_{|\Gamma_C}(\dot{u}_{kh}^m)\right) = 0$$

because of (4.2.4). □

We have seen that the energy conservation carries over to the discrete problem. The existence of a weak solution is an open question as well as the convergence of the discrete solution to the continuous one and an a priori error estimate.

### 4.2.2. Solution of the Discrete Problem.

In this section, we discuss the calculation of a discrete solution according to Definition 4.2.2.

[1] see Proposition 1.2.7 for the discussion of the admissibility of $(\dot{v}_{kh}, \dot{u}_{kh})$

The derivation of the time stepping scheme is carried out in the same way as in Section 1.2.2. We end up with the initial conditions

$$\begin{aligned}\left(u_{kh}^0 - u_s, \psi_h\right) &= 0\\ \left(v_{kh}^0 - v_s, \chi_h\right) &= 0\end{aligned}$$

and

$$\begin{aligned}
\left(v_{kh}^m, \psi_h\right) &= \frac{2}{k_m}\left(u_{kh}^m - u_{kh}^{m-1}, \psi_h\right) - \left(v_{kh}^{m-1}, \psi_h\right) && (4.2.5)\\
\frac{2}{k_m^2}\left(u_{kh}^m, \chi_h\right) &= \frac{1}{2}\left(f^m + f^{m-1}, \chi_h\right) - \left(\lambda_{kH}^m, \gamma_{|\Gamma_C}\left(\chi_{kh}\right)\right)_{\Gamma_C} && (4.2.6)\\
&\quad + \frac{2}{k_m^2}\left(u_{kh}^{m-1}, \chi_h\right) + \frac{2}{k_m}\left(v_{kh}^{m-1}, \chi_h\right)\\
&\quad - \frac{1}{2}a\left(u_{kh}^m + u_{kh}^{m-1}\right)\left(\chi_h\right)\\
0 &\geq \left(\mu_H - \lambda_{kH}^m, \gamma_{|\Gamma_C}\left(u_{kh}^m\right) - g\right)_{\Gamma_C} && (4.2.7)\\
0 &= \left(\lambda_{kH}^m, \gamma_{|\Gamma_C}\left(u_{kh}^m - u_{kh}^{m-1}\right)\right)_{\Gamma_C} && (4.2.8)
\end{aligned}$$

for all $\psi_h, \chi_h \in V_h^m$, all $\mu_H \in \Lambda_H^m$, and all $m = 1, \ldots, M$. We have multiplied equation (4.2.6) by $\frac{2}{k_m^2}$, because this form is more convenient for the solution of the discrete problem, as we will see later on. The pointwise condition in (4.2.3) reduces to the condition in the single time instances in (4.2.7), because $u$ is a continuous and piecewise linear function. Furthermore, the discrete persistency condition (4.2.4) is equivalent to condition (4.2.8), since all variables in (4.2.4) are constant in time.

The initial conditions are defined by $L^2$-projections, which are solved by the same techniques as in Section 1.2.2. The solution of the system (4.2.5-4.2.8) is more involved. Let $\bar{M}^m \in \mathbb{R}^{d^m \times d^m}$ be the mass matrix, $\bar{K}^m \in \mathbb{R}^{d^m \times d^m}$ the stiffness matrix, and

$$\bar{A}^m := \frac{2}{k_m^2}\bar{M}^m + \frac{1}{2}\bar{K}^m \in \mathbb{R}^{d^m \times d^m}$$

the generalised stiffness matrix. The vector $\bar{F}^m \in \mathbb{R}^{d^m}$ represents

$$\begin{aligned} &\frac{1}{2}\left(f^m + f^{m-1}, \chi_h\right) + \frac{2}{k_m^2}\left(u_{kh}^{m-1}, \chi_h\right) \\ &+\frac{2}{k_m}\left(v_{kh}^{m-1}, \chi_h\right) - \frac{1}{2}a\left(u_{kh}^{m-1}\right)(\chi_h). \end{aligned}$$

The right hand side of equation (4.2.5) is given by $\bar{G}^m$. Let $B \in \mathbb{R}^{d^m \times \tilde{d}^m}$ be the matrix associated to $\left(\cdot, \gamma_{|\Gamma_C}(\cdot)\right)_{\Gamma_C}$. The vector connected to the gap function $g$ is $\bar{g}$. The vectors $\bar{u}^m$, $\bar{u}^{m-1}$, $\bar{v}^m \in \mathbb{R}^{d^m}$ are connected to the functions $u_{kh}^m$, $u_{kh}^{m-1}$, $v_{kh}^m$, respectively. The vector $\bar{\lambda}^m \in \mathbb{R}_{\leq 0}^{\tilde{d}^m}$ represents the Lagrange multiplier $\lambda_{kH}^m$. The system (4.2.5-4.2.8) reads in matrix vector notation:

$$\begin{aligned} \bar{M}^m \bar{v}^m &= \bar{G}^m \\ \bar{A}^m \bar{u}^m + \bar{B}^m \bar{\lambda}^m &= \bar{F}^m \\ \forall \bar{\mu} \in \mathbb{R}_{\leq 0}^{\tilde{d}^m}: \quad \left(\bar{\mu} - \bar{\lambda}^m\right)^T \left(\left(\bar{B}^m\right)^T \bar{u}^m - \bar{g}\right) &\leq 0 \\ \left(\bar{\lambda}^m\right)^T \left(\bar{B}^m\right)^T \left(\bar{u}^m - \bar{u}^{m-1}\right) &= 0. \end{aligned}$$

Using $\bar{u}^m = \bar{A}^{-1}\left\{\bar{F}^m - \bar{B}^m \bar{\lambda}^m\right\}$, we write the last two lines in such a way that the Lagrange multiplier $\bar{\lambda}^m$ is the only unknown:

$$\begin{aligned} \bar{M}^m \bar{v}^m &= \bar{G}^m && (4.2.9) \\ \bar{A}^m \bar{u}^m + \bar{B}^m \bar{\lambda}^m &= \bar{F}^m && (4.2.10) \\ \forall \bar{\mu} \in \mathbb{R}_{\leq 0}^{\tilde{d}^m}: \quad \left(\bar{\mu} - \bar{\lambda}^m\right)^T \left(\bar{c}^m - \bar{Q}^m \bar{\lambda}^m\right) &\leq 0 \\ \left(\bar{\lambda}^m\right)^T \left(2\bar{d}^m - \bar{Q}^m \bar{\lambda}^m\right) &= 0. \end{aligned}$$

Here,

$$\bar{Q}^m := \left(\bar{B}^m\right)^T \left(\bar{A}^m\right)^{-1} \bar{B}^m \in \mathbb{R}^{\tilde{d}^m \times \tilde{d}^m}$$

is a symmetric and positive semidefinite matrix,

$$\begin{aligned} \bar{c}^m &:= \left(\bar{B}^m\right)^T \left(\bar{A}^m\right)^{-1} \bar{F}^m - \bar{g}^m, \\ \bar{d}^m &:= \frac{1}{2}\left(\bar{B}^m\right)^T \left\{\left(\bar{A}^m\right)^{-1} \bar{F}^m - \bar{u}^{m-1}\right\}. \end{aligned}$$

The last two lines correspond to a quadratic program with nonlinear constraints:

$$\begin{aligned} \min_{\bar{\mu}\in\mathbb{R}^{\tilde{d}^m}} \quad & \tfrac{1}{2}\bar{\mu}^T\bar{Q}^m\bar{\mu} - \bar{\mu}^T\bar{c}^m & \\ \text{s. t.} \quad & \bar{\mu} & \leq 0 \\ & \tfrac{1}{2}\bar{\mu}^T\bar{Q}^m\bar{\mu} - \bar{\mu}^T\bar{d}^m & = 0 \end{aligned} \tag{4.2.11}$$

The solution algorithm is now straightforward: First, the solution $\bar{\mu} = \bar{\lambda}^m$ of the quadratic program (4.2.11) is determined. Then, the displacement $\bar{u}^m$ is calculated on the basis of equation (4.2.10). Finally, equation (4.2.9) is used to determine $\bar{v}^m$.

The quadratic program (4.2.11) can be reduced to a linear program with the help of the nonlinear constraint. We use the optimisation software SNOPT [**52**] to solve the program. In principle, every optimisation software, which is designed for this class of program, could be used. The only restriction is that the matrix $\bar{Q}^m$ should not be needed explicitly, since only a matrix vector multiplication of $\bar{Q}^m$ can be computed. We use a direct solver, namely Umfpack [**43**], to factorise $\bar{Q}^m$ and efficiently evaluate the matrix vector multiplication by forward and backward substitution. The use of iterative solvers is possible and in particular multigrid solvers are an interesting alternative.

The advantage of solving in the dual variables is that the number of optimisation variables is small compared to the optimisation in the primal variables. Furthermore, we obtain a value for the Lagrange multiplier, which is a better approximation of the contact stress than the post processed value from the displacement $u$. The Lagrange multiplier is moreover an essential quantity in the presented a posteriori error estimate.

**4.2.3. Model Problem Analysis.** In this section, we test the presented discretisation scheme with an easy but representative model problem. This gives a first hint, whether a time stepping scheme is appropriate for the discretisation of dynamic contact problems or not. Since no mathematical theory exists to prove the convergence, we have to rely on the model problem analysis, which also is used in literature. There, optimal time integration parameters are calculated on the basis of the model problem, c.f. [**15, 39**]. We use the same problem as in [**93**], which is based on the ideas in [**15, 39**]. We consider a force free movement of a mass point of 1 kg. The initial position is $0 \leq u_s < -kv_s$, where $v_s < 0$ is the initial velocity and $k$ is the constant time step length. The condition $u_s < -kv_s$ is only needed to abbreviate the calculation. The rigid foundation is located at the coordinate 0. The analytical solution of this problem is

$$u(t) = |u_s + v_s t| \,.$$

The contact takes place for $t = t_c := -\frac{u_s}{v_s}$ and the velocity is given by

$$v(t) = \begin{cases} v_s & \text{for } t < t_c \\ -v_s & \text{for } t > t_c. \end{cases}$$

We observe that the displacement $u$ is only continuous but not continuously differentiable and the velocity $v$ is undefined for $t = t_c$ and discontinuous.

Now, we use the time stepping algorithm presented in the last section to calculate a numerical approximation to $u$. First, we have to determine $\lambda^1 \leq 0$ in such a way that $u^1 \geq 0$ and $\lambda^1 \left(u^1 - u_s\right) = 0$ holds. This is the case for $\lambda^1 = kv_s$. Then, we obtain $u^1 = u_s$ and $v^1 = -v_s$, which corresponds to the analytical solution. Consequently, the error in this case is 0 and we have obtained a perfect approximation.

## 4.3. A Posteriori Error Estimation

In this section, we derive an a posteriori error estimate for the discretisation error w.r.t. some output functional

$$J(w) = \int_0^T J_1(w)\,dt + J_2(w(T)),$$

see Section 1.3. For this purpose, we define the associated dual problem

$$\forall \varphi = (\psi, \chi) \in U \times V: \quad A'(\varphi, z) = J'(w)(\varphi)$$

with a solution $z = (\bar{u}, \bar{v}) \in V \times U$. The Fréchet derivative[2] of $A$ w.r.t. $w$ is

$$\begin{aligned} A'(\varphi, z) \quad = \quad & \left(\left(\chi - \dot{\psi}, \bar{u}\right)\right) + ((\dot{\chi}, \bar{v})) + (a(\chi, \bar{v})) \\ & + (\chi(0), \bar{v}(0)) + (\psi(0), \bar{u}(0)) \, . \end{aligned}$$

The corresponding finite element solution $z_{kh} = (\bar{u}_{kh}, \bar{v}_{kh}) \in W_{kh} \times W_{kh}$ is given by

$$\forall \varphi_{kh} = (\psi_{kh}, \chi_{kh}) \in V_{kh} \times V_{kh} : \quad A'_{kh}(\varphi_{kh}, z_{kh}) = J'(w_{kh})(\varphi_{kh}) \, .$$

The continuous and the discrete dual problem are equal to the dual problems considered in Section 1.3. Consequently, we have to use the same time stepping scheme etc. and can pass on the details.

We specify a second auxiliary problem by

$$\forall \varphi = (\psi, \chi) \in U \times V : \quad A(w_\star, \varphi) = -\left(\left(\lambda_{kH}, \gamma_{|\Gamma_C}(\psi)\right)_{\Gamma_C}\right) \qquad (4.3.1)$$

with a solution $w_\star = (u_\star, v_\star) \in U \times V$. It is a linear hyperbolic equation of second order, see Section 1.1. The auxiliary problem (4.3.1) corresponds to (4.2.1), but with the discrete Lagrange multiplier $\lambda_{kH}$ instead of $\lambda$. We observe that $w_{kh}$ also is a discrete solution of (4.3.1). The dual problem associated to $w_\star$ has the solution $z_\star = (\bar{u}_\star, \bar{v}_\star) \in U \times V$ defined by

$$\forall \varphi\,(\psi, \chi) \in U \times V : \quad A'(\varphi, z_\star) = J'(w_\star)(\varphi) \, .$$

If $J$ is linear, $z_\star$ is equal to $z$. Since $w_{\star,kh} = w_{kh}$, we obtain $z_{\star,kh} = z_{kh}$.

Now, we derive the a posteriori error estimate. The error is split up into two parts:

$$\begin{aligned} & J(w) - J(w_{kh}) \\ = \quad & J(w) - J(w_\star) + J(w_\star) - J(w_{kh}) \quad . \end{aligned}$$

[2] We skip the $(w)$ in the notation, since $A'$ does not depend on $w$.

First, we look at $J(w) - J(w_\star)$. Using the arguments from Section 3.3 and performing the same calculations, we obtain

$$J(w) - J(w_\star) = \frac{1}{2}\left[J'(w_\star)(e_\star) + J'(w)(e_\star)\right] + \mathcal{R}_\star^{(3)}$$

with $e_\star := w - w_\star$, where the remainder term $\mathcal{R}_\star^{(3)}$ is of third order in the error $e_\star$. Using the definitions of the dual solutions $z$ and $z_\star$ and the equations (4.2.1) and (4.3.1), we obtain

$$\begin{array}{rc} & J'(w_\star)(e_\star) \\ = & A'(e_\star, z_\star) \\ = & -\left(\left(\lambda - \lambda_{kH}, \gamma_{|\Gamma_C}(\bar{v}_\star)\right)_{\Gamma_C}\right) \end{array}$$

and

$$J'(w)(e_\star) = -\left(\left(\lambda - \lambda_{kH}, \gamma_{|\Gamma_C}(\bar{v})\right)_{\Gamma_C}\right).$$

We define the contact residual

$$\rho_c(\lambda, \lambda_{kH})(z, z_\star) := -\left(\left(\lambda - \lambda_{kH}, \frac{1}{2}\gamma_{|\Gamma_C}(\bar{v}_\star + \bar{v})\right)_{\Gamma_C}\right)$$

and obtain the error identity

$$J(w) - J(w_\star) = \rho_c(\lambda, \lambda_{kH})(z, z_\star) + \mathcal{R}_\star^{(3)}. \tag{4.3.2}$$

For linear $J$, the contact residual is given by

$$\rho_c(\lambda, \lambda_{kH})(z) := -\left(\left(\lambda - \lambda_{kH}, \gamma_{|\Gamma_C}(\bar{v})\right)_{\Gamma_C}\right)$$

and the remainder term vanishes.

The second part $J(w_\star) - J(w_{kh})$ corresponds to the problems discussed in Chapter 1. As outlined in the preceding chapter, we can choose any appropriate a posteriori error estimator to control this difference. We use the one presented in Chapter 1. Since $A$ is linear in this context, we obtain

$$\begin{array}{rc} & J(w_\star) - J(w_{kh}) \\ =: & \frac{1}{2}\left[\rho(w_{kh}, \lambda_{kH})(z - \tilde{z}_{kh}) + \rho^\star(z_{kh})(u - \tilde{u}_{kh})\right] + \mathcal{R}_{kh}^{(3)} \end{array} \tag{4.3.3}$$

for arbitrary $\tilde{z}_{kh} \in W_{kh} \times W_{kh}$ and $\tilde{u}_{kh} \in V_{kh} \times V_{kh}$. The remainder term $\mathcal{R}^{(3)}_{kh}$ is given by

$$\mathcal{R}^{(3)}_{kh} = \frac{1}{2} \int_0^1 J''' \left( w_{kh} + se \right) (e, e, e) \, s \, (s - 1) \; ds$$

with $e := w_\star - w_{kh}$ and is of third order in the error $e$. For a linear $J$, the remainder term is zero.

In order to obtain a computationally evaluable error estimate, we approximate the error identities (4.3.2) and (4.3.3) by

$$\begin{aligned} & J(w) - J(w_{kh}) \\ = \; & \rho_c \left( \lambda, \lambda_{kH} \right) (z, z_\star) + \mathcal{R}^{(3)}_\star \\ & + \tfrac{1}{2} \left[ \rho \left( w_{kh}, \lambda_{kH} \right) (z - \tilde{z}_{kh}) + \rho^\star \left( w_{kh}, z_{kh} \right) (w - \tilde{w}_{kh}) \right] + \mathcal{R}^{(3)}_{kh} \\ \approx \; & \rho_c \left( i^{(1,1)}_{k,H} \lambda_{kH}, \lambda_{kH} \right) \left( i^{(1,2)}_{k,2h} z_{kh}, i^{(1,2)}_{k,2h} z_{kh} \right) \\ & + \tfrac{1}{2} \left[ \rho \left( w_{kh}, \lambda_{kH} \right) \left( i^{(1,2)}_{k,2h} z_{kh} - z_{kh} \right) + \rho^\star \left( z_{kh} \right) \left( i^{(2,2)}_{2k,2h} w_{kh} - w_{kh} \right) \right] \\ =: \; & \eta = \eta_c + \eta_e. \end{aligned}$$

Here, we have used the fact that $z_{\star,kh} = z_{kh}$. As in the preceding chapter, it is possible to derive the error estimate without the use of the auxiliary problem (4.3.1). However, the use of the auxiliary problem offers the possibility to use different known error estimators to estimate the error of it.

We have obtained an evaluable a posteriori error estimate for the discretisation error in an arbitrary nonlinear output functional. But we also want to use the estimate as basis for an adaptive refinement algorithm. Hence, we have to split the spatial and the temporal error and to localise the error estimate. We have done this for $\eta_e$ in Chapter 1 and 2. For $\eta_c$, we use the same ideas. Since $\lambda_{kH}$ is a tensor product function, we obtain

$$\begin{aligned} i^{(1,1)}_{k,H} \lambda_{kH} - \lambda_{kH} &= i^{(1)}_k \left( i^{(1)}_H \lambda_{kH} - \lambda_{kH} \right) + i^{(1)}_k \lambda_{kH} - \lambda_{kH} \\ &= i^{(1)}_H \left( i^{(1)}_k \lambda_{kH} - \lambda_{kH} \right) + i^{(1)}_H \lambda_{kH} - \lambda_{kH} \end{aligned}$$

and define accordingly

$$\begin{aligned}\eta_c &= \eta^i_{c,H} + \eta^n_{c,k} \\ &= \eta^i_{c,k} + \eta^n_{c,H}.\end{aligned}$$

The interpolation $i_H^{(1)}$ has been defined in Section 3.3 and $i_k^{(1)}$ in Section 1.3. The terms $\eta^i_{c,H}$ and $\eta^n_{c,H}$ are evaluated on the edges or faces of the boundary and the value is then added to the estimated value of the adjacent element, which contains this edge or face. The values of $\eta^i_{c,k}$ and $\eta^n_{c,k}$ are simply added to the value of $\eta_{e,k}$ in the corresponding time step. The computationally evaluated form of $\eta_c$ is described in Section A.6.

In the preceding chapter, we have seen that it does not matter, whether we insert $i_{2h}^{(2)} z_h$ or $z_h$ into the contact residual. Consequently, we also specify the estimator

$$\bar{\eta}_c = \rho_c \left( i_{k,H}^{(1,1)} \lambda_{kH}, \lambda_{kH} \right) (z_{kh}, z_{kh})$$

with

$$\begin{aligned}\bar{\eta}_c &= \bar{\eta}^i_{c,H} + \bar{\eta}^n_{c,k} \\ &= \bar{\eta}^i_{c,k} + \bar{\eta}^n_{c,H}.\end{aligned}$$

## 4.4. Numerical Results

In this section, we discuss two applications of the presented error estimator. In the first example, we discuss a simplified Signorini problem, which corresponds to the example presented in Section 2.7.2 and has similarities to the example of Section 3.4.1. Afterwards, a full Signorini example is presented. We extend the example presented in Section 3.4.2 by considering dynamic effects.

### 4.4.1. Example 1: Dynamic Simplified Signorini Problem.

We consider a dynamic simplified Signorini problem on the basic domain $\Omega = [0,1]^2$ and the time interval $I = [0,1]$. The boundary is divided into

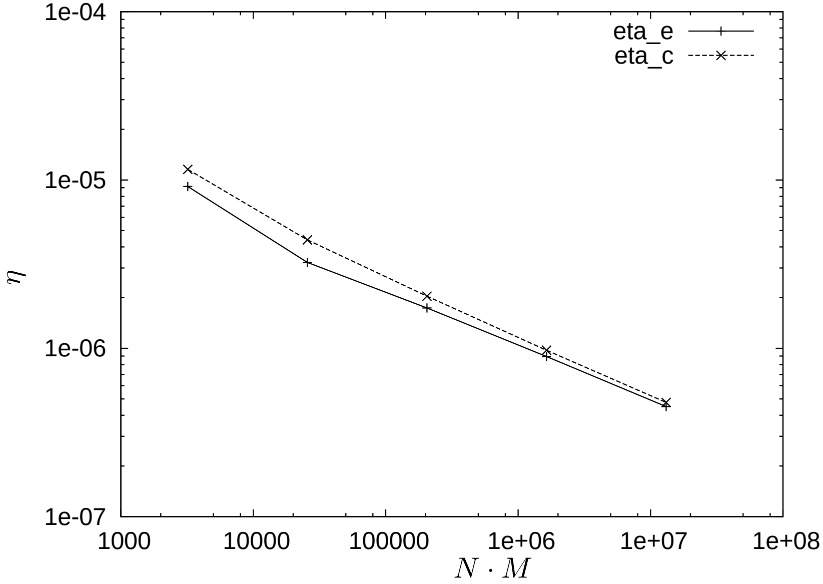


FIGURE 4.4.1. Comparison of $\eta_c^{ni}$ and $\eta_e^{ni}$ w.r.t. uniform spatial and temporal refinement

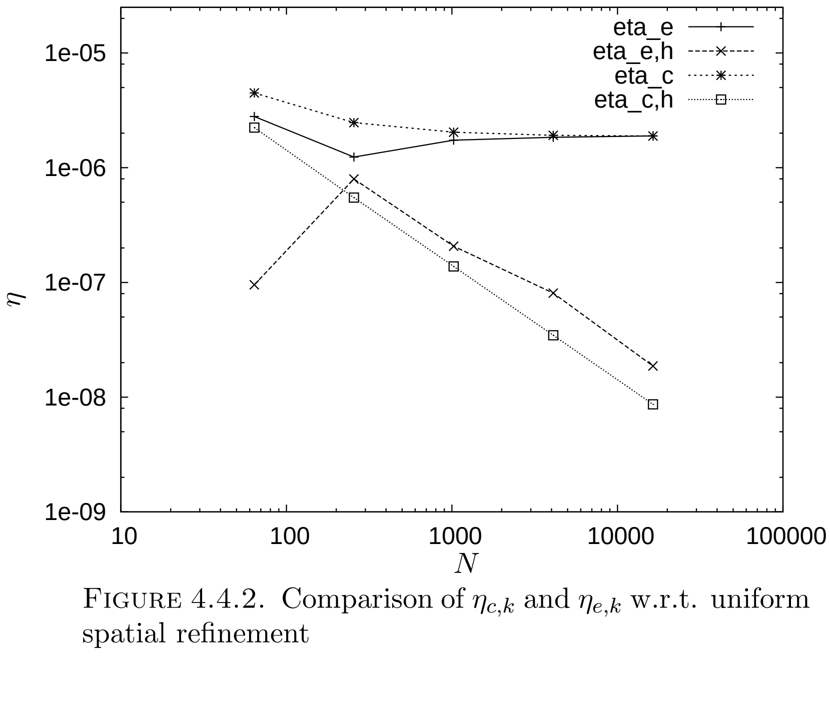


FIGURE 4.4.2. Comparison of $\eta_{c,k}$ and $\eta_{e,k}$ w.r.t. uniform spatial refinement

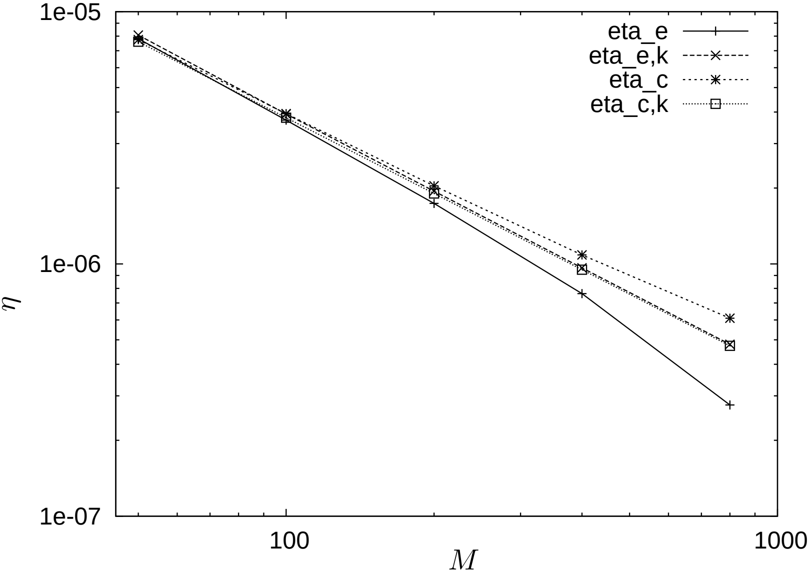


FIGURE 4.4.3. Comparison of $\eta_{c,H}$ and $\eta_{e,h}$ w.r.t. uniform temporal refinement

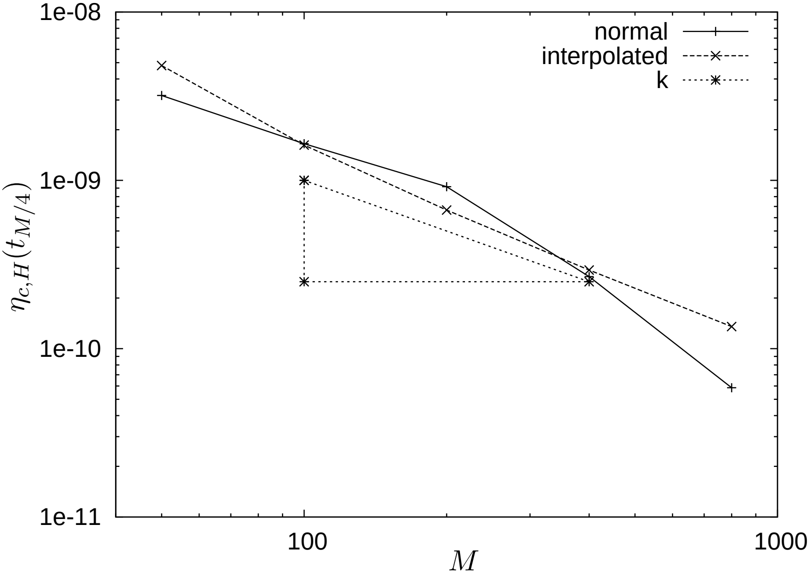


FIGURE 4.4.4. Development of $\eta^n_{c,H}(t_{M/4})$, $\eta^i_{c,H}(t_{M/4})$ over $k$

| $M$ | $N$ | $\eta_c^{nn}$ | $\eta_c^{ni}$ | $\eta_c^{in}$ | $\eta_c^{ii}$ | $\eta_e^{ni}$ |
|---|---|---|---|---|---|---|
| 50 | 64 | 118.4 | 115.7 | 115.7 | 113.0 | 91.46 |
| 100 | 256 | 44.32 | 44.07 | 44.07 | 43.82 | 32.38 |
| 200 | 1024 | 20.42 | 20.40 | 20.40 | 20.37 | 17.37 |
| 400 | 4096 | 9.736 | 9.733 | 9.733 | 9.729 | 8.937 |
| 800 | 16384 | 4.770 | 4.769 | 4.769 | 4.769 | 4.508 |

TABLE 4.4.1. Development of the contact error estimator $\eta_c$ w.r.t. uniform spatial and temporal refinement, all values of the error estimators are scaled by $10^7$

| $M$ | $N$ | $\bar{\eta}_c^{nn}$ | $\bar{\eta}_c^{ni}$ | $\bar{\eta}_c^{in}$ | $\bar{\eta}_c^{ii}$ | $\eta_e^{ni}$ |
|---|---|---|---|---|---|---|
| 50 | 64 | 100.7 | 98.37 | 98.37 | 96.06 | 91.46 |
| 100 | 256 | 41.75 | 41.52 | 41.52 | 41.29 | 32.38 |
| 200 | 1024 | 19.92 | 19.90 | 19.90 | 19.87 | 17.37 |
| 400 | 4096 | 9.631 | 9.628 | 9.628 | 9.624 | 8.937 |
| 800 | 16384 | 4.743 | 4.743 | 4.743 | 4.743 | 4.508 |

TABLE 4.4.2. Development of the simplified contact error estimator $\bar{\eta}_c$ w.r.t. uniform spatial and temporal refinement, all values of the error estimators are scaled by $10^7$

| $M$ | $N$ | $\eta_{c,k}^{i}$ | $\eta_{c,k}^{n}$ | $\bar{\eta}_{c,k}^{i}$ | $\bar{\eta}_{c,k}^{n}$ | $\eta_{e,k}^{i}$ |
|---|---|---|---|---|---|---|
| 200 | 64 | 2.246 | 2.317 | 2.149 | 2.211 | 2.697 |
| 200 | 256 | 1.921 | 1.934 | 1.864 | 1.876 | 2.036 |
| 200 | 1024 | 1.902 | 1.905 | 1.855 | 1.858 | 1.944 |
| 200 | 4096 | 1.880 | 1.881 | 1.839 | 1.840 | 1.921 |
| 200 | 16384 | 1.880 | 1.880 | 1.837 | 1.837 | 1.912 |

TABLE 4.4.3. Behaviour of $\eta_{c,k}$ w.r.t. uniform spatial refinement, all values of the error estimators are scaled by $10^6$

three parts:

$$\begin{aligned}
\Gamma_D &= \{x = (x_1, x_2) |\, x_1 = 0\}\,, \\
\Gamma_C &= \{x = (x_1, x_2) |\, x_1 = 1\}\,, \\
\Gamma_N &= \partial\Omega \backslash (\Gamma_D \cup \Gamma_C)\,.
\end{aligned}$$

| $M$ | $N$ | $\eta^i_{c,H}$ | $\eta^n_{c,H}$ | $\bar{\eta}^i_{c,H}$ | $\bar{\eta}^n_{c,H}$ | $\eta^n_{e,h}$ |
|---|---|---|---|---|---|---|
| 50 | 1024 | 1.357 | 1.466 | 1.241 | 1.345 | $-2.582$ |
| 100 | 1024 | 1.360 | 1.413 | 1.297 | 1.349 | $-2.164$ |
| 200 | 1024 | 1.354 | 1.380 | 1.320 | 1.345 | $-2.067$ |
| 400 | 1024 | 1.351 | 1.363 | 1.330 | 1.342 | $-2.027$ |
| 800 | 1024 | 1.350 | 1.356 | 1.335 | 1.341 | $-2.040$ |

TABLE 4.4.4. Behaviour of $\eta_{c,H}$ w.r.t. uniform temporal refinement, all values of the error estimators are scaled by $10^7$

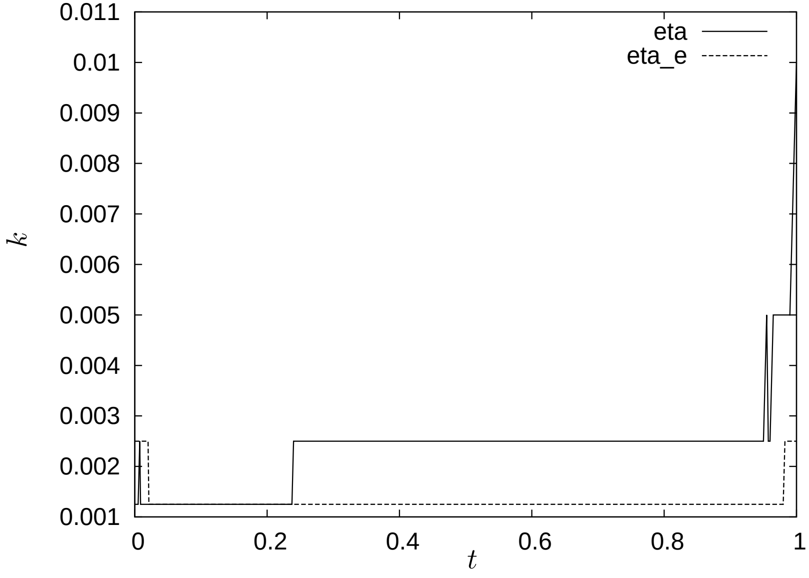


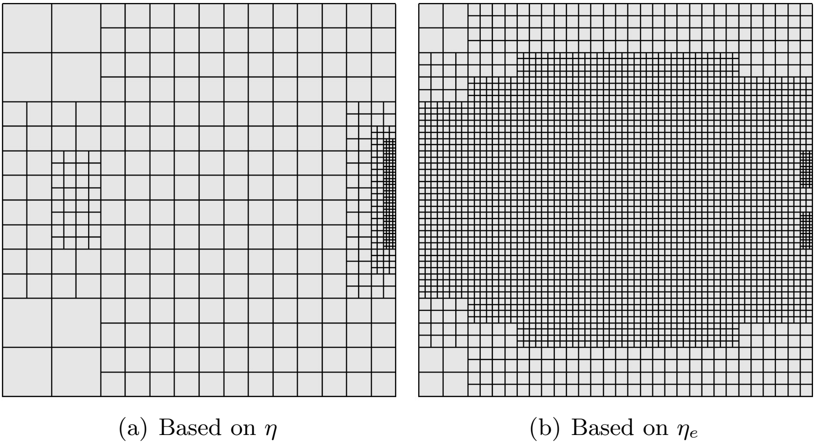
FIGURE 4.4.5. Temporal meshes based on $\eta$ and on $\eta_e$

Homogeneous Neumann boundary conditions are prescribed on $\Gamma_N$. The right hand side $f$ is set to zero. The initial conditions are $u_s = 0$ and

$$v_s(x) = -\frac{1}{4}\sin\left(\frac{1}{2}\pi x_1\right).$$

The rigid foundation is given by

$$g\left(x_2\right) = -\left(x_2 - \frac{1}{2}\right)^2.$$



(a) Based on $\eta$

(b) Based on $\eta_e$

FIGURE 4.4.6. Meshes created by adaptive refinement in the CM approach

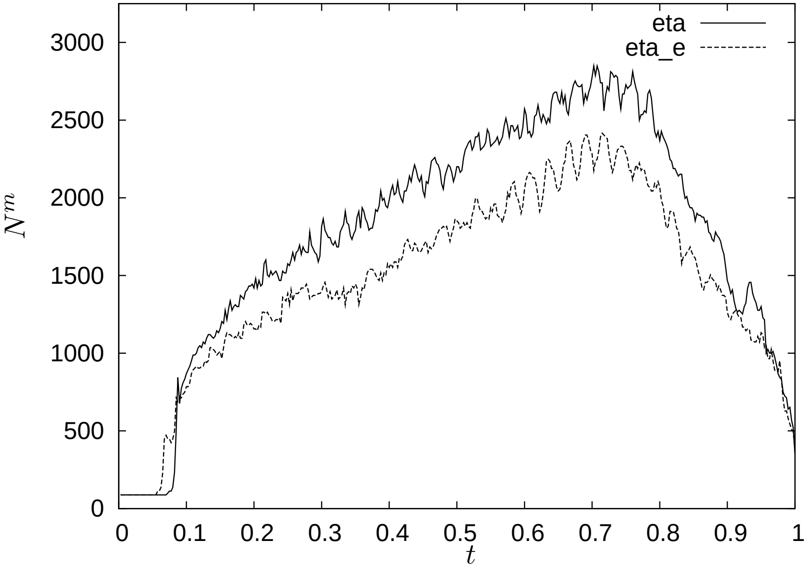
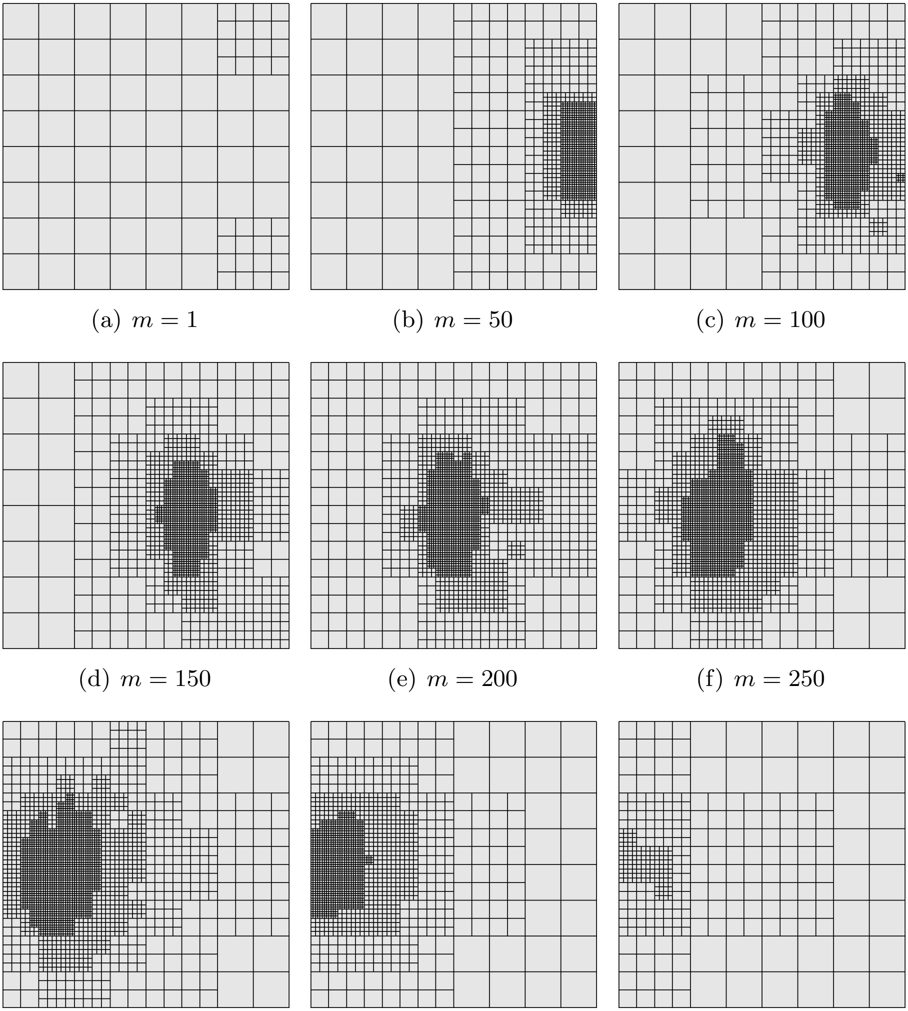


FIGURE 4.4.7. Distribution of the number of mesh elements in the DM approach based on $\eta$ and on $\eta_e$



(a) $m = 1$ (b) $m = 50$ (c) $m = 100$

(d) $m = 150$ (e) $m = 200$ (f) $m = 250$

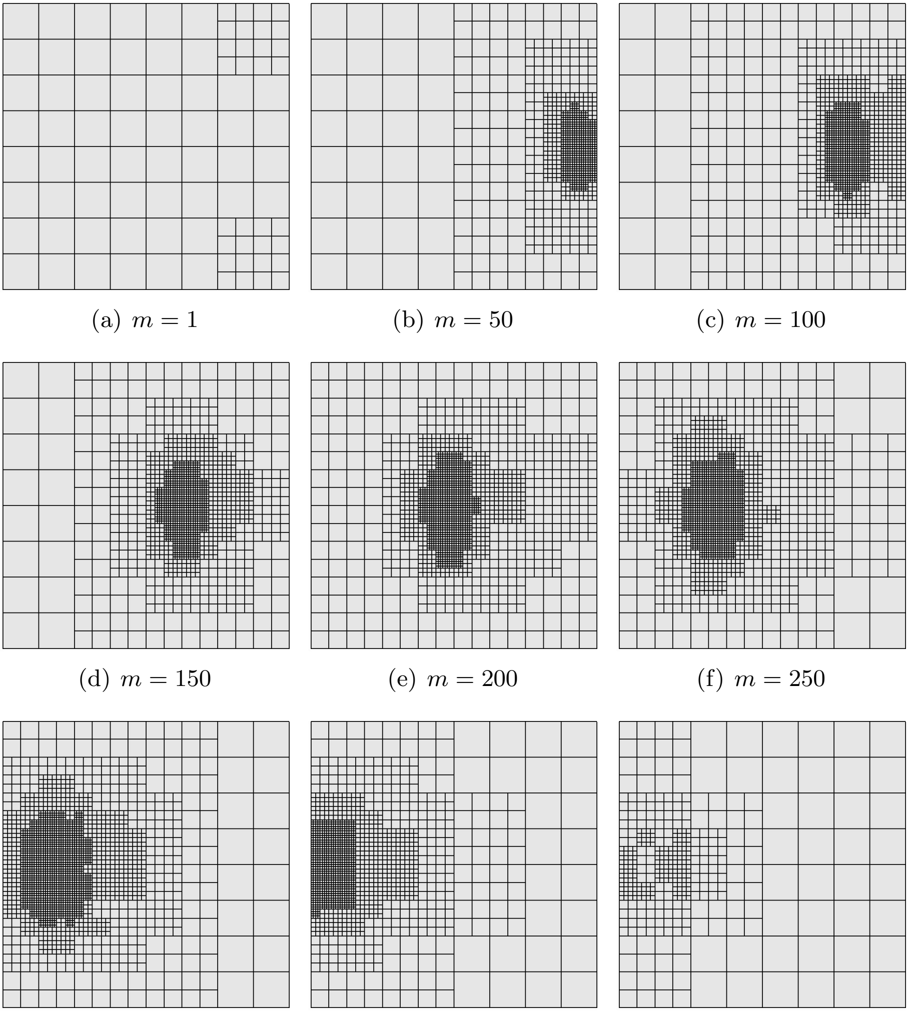
(g) $m = 300$ (h) $m = 350$ (i) $m = 400$

FIGURE 4.4.8. Meshes created during the adaptive refinement process based on $\eta$ in the DM approach

The functional of interest is

$$J_1(w) := \frac{1}{|B|}\int_B u^2\,dx$$

with $B = [0, 0.25] \times [0.375, 0.625]$ and $J_2(w(T)) = 0$.



(a) $m = 1$ (b) $m = 50$ (c) $m = 100$

(d) $m = 150$ (e) $m = 200$ (f) $m = 250$

(g) $m = 300$ (h) $m = 350$ (i) $m = 400$

FIGURE 4.4.9. Meshes created during the adaptive refinement process based on $\eta_e$ in the DM approach

In Table 4.4.1 and 4.4.2, the values of various variants of the error estimators $\eta_c$ are given, where the temporal and the spatial mesh are uniformly refined. All values are similar. We will use $\eta_c^{ni}$, since we have chosen $\eta_e^{ni}$ in Chapter 2. Furthermore, the values of $\eta_e^{ni}$ are given. They are of the

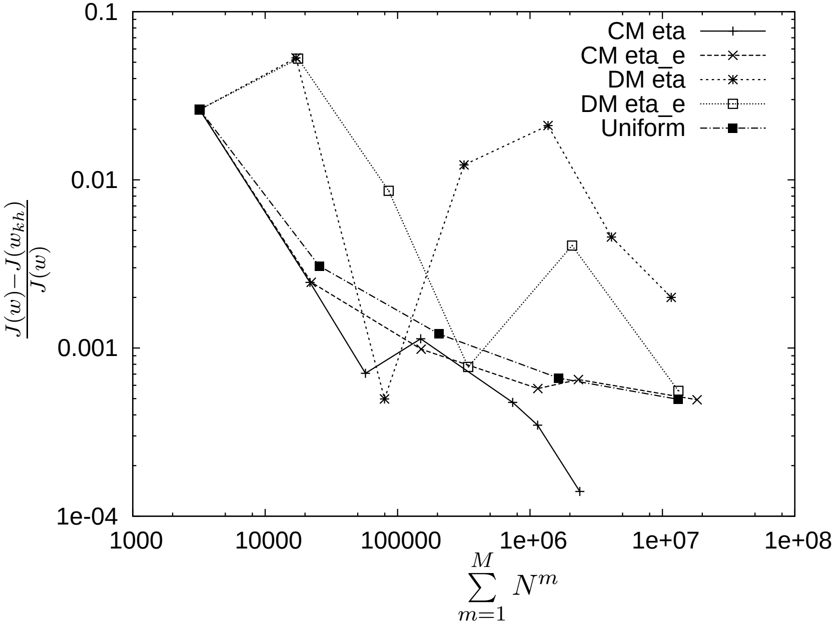


FIGURE 4.4.10. Convergence in the output functional $J$ w.r.t. the total number of mesh cells for different refinement techniques

| $M$ | $\sum_{m=1}^{M} N^M$ | $E_{\text{rel}}$ | $I_{\text{eff}}$ |
|---|---|---|---|
| 50 | 3200 | $2.615 \cdot 10^{-2}$ | 0.702 |
| 94 | 21808 | $2.454 \cdot 10^{-3}$ | 0.118 |
| 188 | 57152 | $7.073 \cdot 10^{-4}$ | 0.075 |
| 374 | 149600 | $1.135 \cdot 10^{-3}$ | 0.291 |
| 714 | 739704 | $4.747 \cdot 10^{-4}$ | 0.186 |
| 870 | 1141440 | $3.474 \cdot 10^{-4}$ | 0.314 |
| 1184 | 2377472 | $1.404 \cdot 10^{-4}$ | 0.855 |

TABLE 4.4.5. Adaptive refinement based on $\eta$ in the CM approach

same magnitude as $\eta_c$. In Figure 4.4.1, we observe that $\eta_c^{ni}$ and $\eta_e^{ni}$ are of the same order, too. Consequently, we consider both terms $\eta_e$ and $\eta_c$ in the error estimate and cannot neglect one.

To substantiate that $\eta_{c,k}$ depends on $h$ only of higher order, we choose a fixed $M$ and vary the spatial mesh width. The results of this calculation are given in Table 4.4.3. The value of $\eta_{c,k}$ is roughly constant just like the value of $\eta_{e,k}$. In Figure 4.4.2, the behaviour of $\eta_{c,k}$ and $\eta_{e,k}$ w.r.t. uniform spatial refinement is compared. They are of the same order and magnitude. The values of $\eta_c$ and $\eta_e$ are nearly constant in this case. In dynamic contact problems, the temporal convergence rate is often smaller than two because of the low temporal regularity of the analytical solution, see, for instance, [**21**]. Furthermore, the temporal discretisation error is mostly dominant.

For the results given in Table 4.4.4, we maintain the spatial mesh and vary the time step length. We observe that the value of $\eta_{c,H}$ is nearly constant. The term $\eta_{e,h}$ shows the same behaviour. Furthermore, it is of the same magnitude as $\eta_{c,H}$. These results substantiate the definition of $\eta_{c,H}$, which only depends of higher order on $k$. In Figure 4.4.3, we compare $\eta_{e,k}$ and $\eta_{c,k}$. They are of the same magnitude and order. The influence of $\eta_{c,H}$ on $\eta_c$ becomes greater while $M$ increases. Thus, the difference between $\eta_{c,k}$ and $\eta_c$ grows, where $\eta_c > \eta_{c,k}$. The same behaviour is observed for $\eta_e$ and $\eta_{e,k}$. But $\eta_e < \eta_{e,k}$ holds, since $\eta_{e,h} < 0$. In Figure 4.4.4, the developement of $\eta_{c,H}$ at a fixed time point is depicted, where the time step size is varied. The value of $\eta_{c,H}$ decreases almost linearly. This result correspond to the behaviour of $\eta_{e,h}$ as presented in Section 1.4.

Let us discuss the results of the adaptive solution process. We choose a fixed spatial mesh with 4096 cells and perform five adaptive refinement iterations. We use the estimators $\eta$ and $\eta_e$ as basis for the adaptive refinement. The created temporal meshes are depicted in Figure 4.4.5. The adaptive refinement based on $\eta$ leads to smaller time steps for small $t$ and larger time steps for $t > 0.2$.

To test the adaptive spatial refinement algorithm, we use a uniform temporal mesh with $M = 400$ time steps. In Figure 4.4.6, the meshes created in the CM approach are presented. In the mesh based on $\eta$, the contact zone is well resolved. Some additional refinements are located in $B$ and between $B$ and the contact zone. The adaptive algorithm based on $\eta_e$ leads to more refined cells, especially in the interior of $\Omega$. The basic structure of the mesh is equal to the structure created by the adaptive refinement algorithm based on $\eta$. The meshes created in the DM approach are depicted in Figure 4.4.8 and 4.4.9. They are similar for $\eta$ and $\eta_e$. For small $t$, the contact zone is well resolved. Then the mesh follows the arising wave from the right to the left. The distribution of the mesh cells over the time intervall are similar as shown in Figure 4.4.7. The adaptive refinement based on $\eta$ leads to meshes with slightly more cells.

In Figure 4.4.10, we compare different refinement methods. The numerical solution in the CM approach based on $\eta$ with $M = 2341$ and

$$\sum_{m=0}^{M} N^m = 7815600$$

is used as reference value. During the numerical experiments, it turned out that the ratio between $k$ and $h$ have to be bounded to circumvent numerical difficulties in the discrete solver. For this purpose, we refine the spatial mesh sequence and the temporal mesh in each refinement iteration. Furthermore, the DM approach is less effective in the presented situation. The time step, in which the first contact between the membrane and the rigid foundation is detected by the numerical algorithm, depends on the spatial mesh. The meshes are coarse in the contact zone in the beginning of the calculation, see Figure 4.4.7, 4.4.8, and 4.4.9. Thus, the first contact is detected later in the DM approach than in the CM approach. This fact leads to the reduced effectivity of the DM approach. The adaptive refinement based on $\eta$ in the CM approach is the most effective method. The adaptive refinement based on $\eta_e$ in the CM approach is as effective as the uniform refined method. The effectivity indices during the adaptive solution process based on $\eta$ in the CM approach are presented in Table 4.4.5.

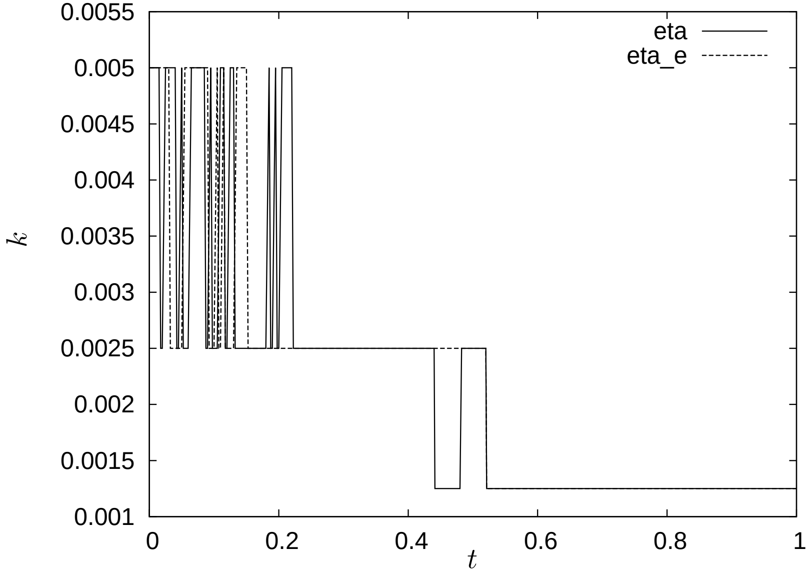
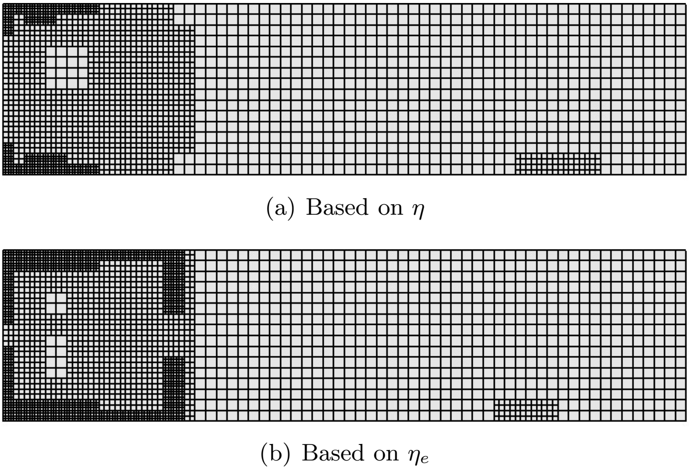


FIGURE 4.4.11. Temporal meshes based on $\eta$ and on $\eta_e$



(a) Based on $\eta$

(b) Based on $\eta_e$

FIGURE 4.4.12. Meshes created by adaptive refinement in the CM approach

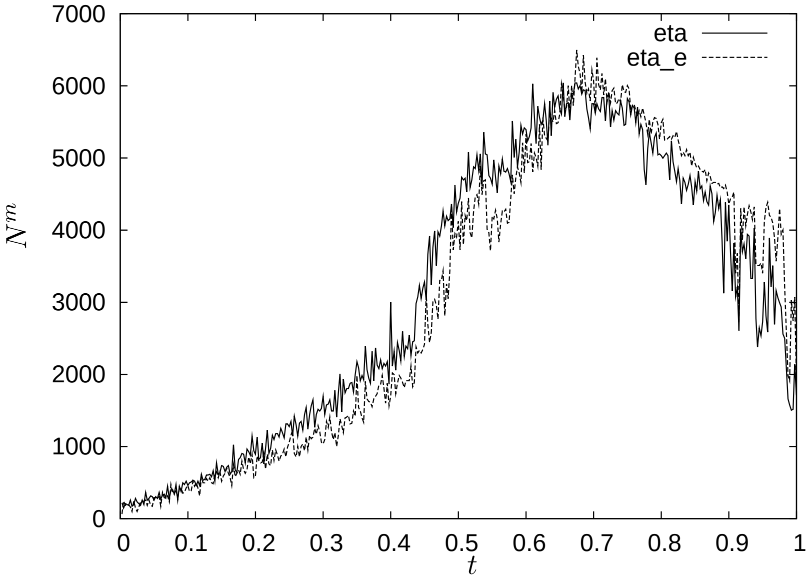


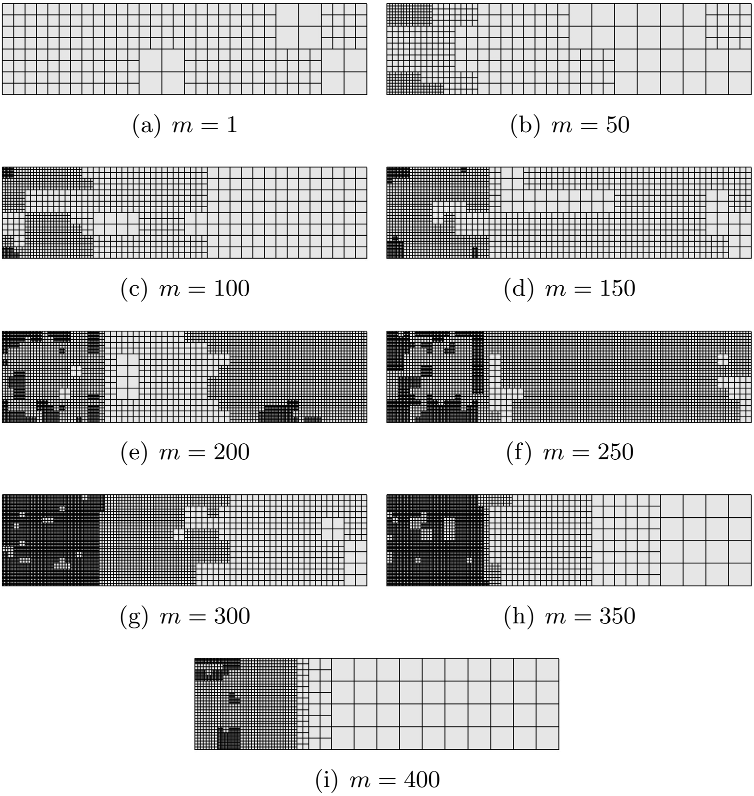
FIGURE 4.4.13. Distribution of the number of mesh elements in the DM approach based on $\eta$ and on $\eta_e$

| $M$ | $\sum_{m=1}^{M} N^M$ | $E_{\text{rel}}$ | $I_{\text{eff}}$ | $I^e_{\text{eff}}$ |
|---|---|---|---|---|
| 50 | 12800 | $3.725 \cdot 10^{-2}$ | 2.692 | 2.708 |
| 96 | 42912 | $1.700 \cdot 10^{-2}$ | 3.119 | 3.132 |
| 182 | 176552 | $6.339 \cdot 10^{-3}$ | 2.294 | 2.300 |
| 352 | 725908 | $3.351 \cdot 10^{-3}$ | 2.532 | 2.536 |
| 678 | 4177824 | $5.664 \cdot 10^{-4}$ | 1.049 | 1.050 |

TABLE 4.4.6. Adaptive refinement based on $\eta$ in the DM approach

**4.4.2. Example 2: Dynamic Full Signorini Problem.** A 2D full Signorini problem is considered in this section. The basic domain is given by $\Omega = [0, 0.05] \times [0, 0.2]$ with

$$\begin{aligned}
\Gamma_D &= \{x = (x_1, x_2)|\, x_2 = 0\}\,, \\
\Gamma_C &= \{x = (x_1, x_2)|\, x_1 = 0.05 \wedge x_2 \geq 0.15\}\,, \\
\Gamma_N &= \partial\Omega \backslash (\Gamma_D \cup \Gamma_C)\,.
\end{aligned}$$



(a) $m = 1$

(b) $m = 50$

(c) $m = 100$

(d) $m = 150$

(e) $m = 200$

(f) $m = 250$

(g) $m = 300$

(h) $m = 350$

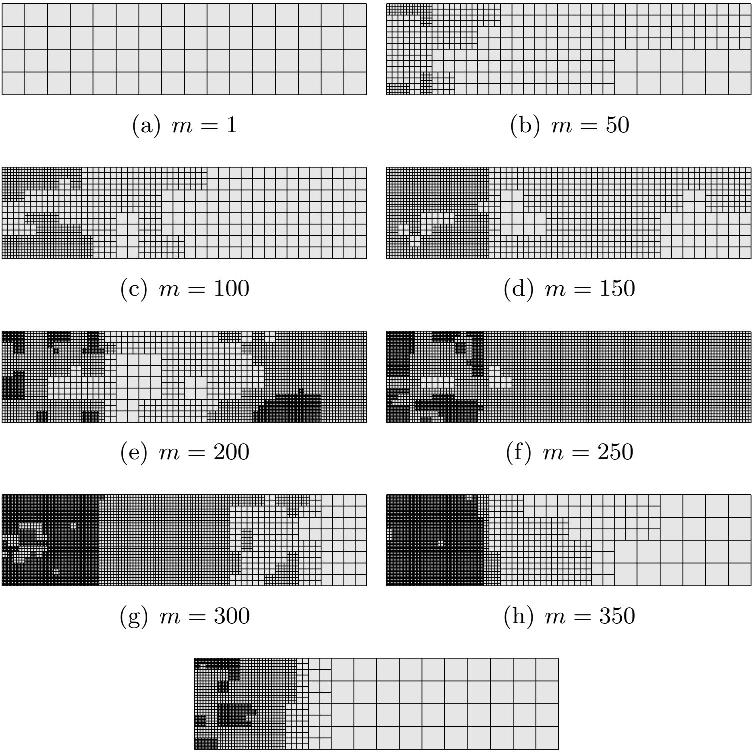
(i) $m = 400$

FIGURE 4.4.14. Meshes created during the adaptive refinement process based on $\eta$ in the DM approach

The time interval is $I = [0, 1]$. We set $q = 0$ and $f = (0.5, 0)^T$. Homogeneous initial conditions are prescribed, i.e. $u_s = v_s = 0$. The material parameters are $E = 10$, $\nu = 0.33$, and $\rho = 10$. The rigid foundation is given by $g = 0.055$. We are interested in the temporal mean value of the von Mises equivalent stress $\sigma_v$, see Section 2.7.3, in the domain $B = [0, 0.05]^2$, i.e.

$$J_1(w) := \frac{1}{|B|} \int_B \sigma_v^2 (u)\; dx$$

and $J_2(w) = 0$.



(a) $m = 1$

(b) $m = 50$

(c) $m = 100$

(d) $m = 150$

(e) $m = 200$

(f) $m = 250$

(g) $m = 300$

(h) $m = 350$

(i) $m = 400$

FIGURE 4.4.15. Meshes created during the adaptive refinement process based on $\eta_e$ in the DM approach

We use a fixed spatial mesh with 4096 cells and perform five adaptive refinement iterations. The temporal meshes created based on $\eta$ and $\eta_e$ are depicted in Figure 4.4.11. The meshes are similar. They use larger time steps for small $t$ and smaller time steps for large $t$.

For the discussion of the spatial meshes, we choose a uniform decomposition of the time interval $I$ with $M = 400$ time steps. The spatial meshes created in the CM approach are presented in Figure 4.4.12. They are similar. The left end of the contact zone and the left corners of $\Omega$ are well resolved. In the left corners, we observe stress singularities due to the geometry. In Figure 4.4.14 and 4.4.15, the adaptive meshes created in the

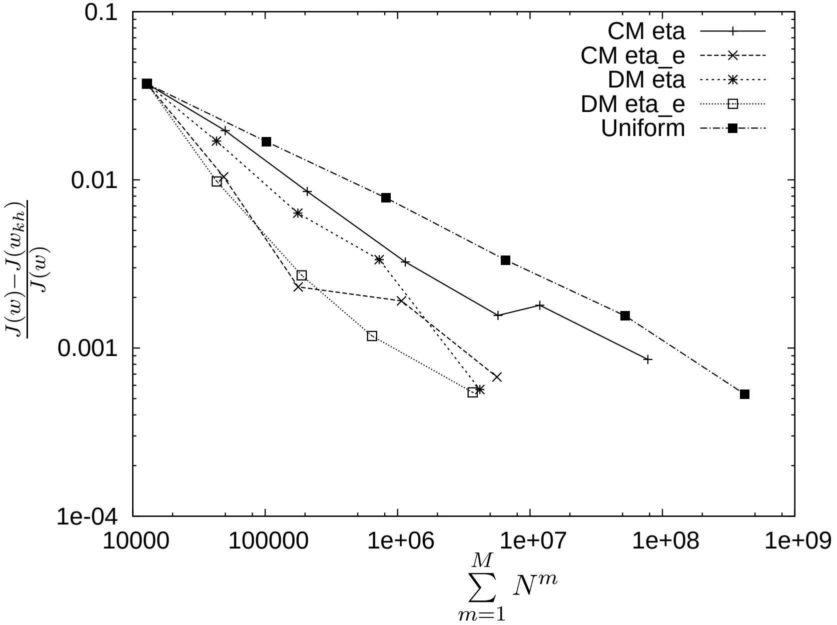
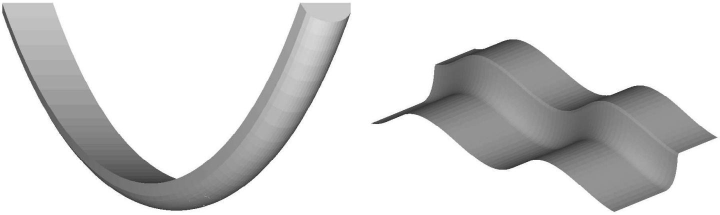


FIGURE 4.4.16. Convergence in the output functional $J$ w.r.t. the total number of mesh cells for different refinement techniques in the full Signorini example

DM approach are depicted. For $m < 200$, we mainly observe additional refinements in the left corners, where the stress singularities occur. In the $200^{\text{th}}$ time step, the left boundary of the contact zone and $B$ are well resolved. The additional refinements are more and more concentrated in $B$ for $m > 200$. In Figure 4.4.13, we see that the number of mesh cells increases for $t > 0.4$. It attains its maximum for $t \approx 0.7$ and then decreases.

In Figure 4.4.16, the convergence behaviour for different refinement techniques are compared, where we have chosen the numerical solution in the DM approach based on $\eta$ with $M = 1322$ and

$$\sum_{m=1}^{M} N^m = 25480148$$



(a) Grinding wheel (b) Workpiece

FIGURE 4.5.1. Geometry of the grinding wheel and the workpiece

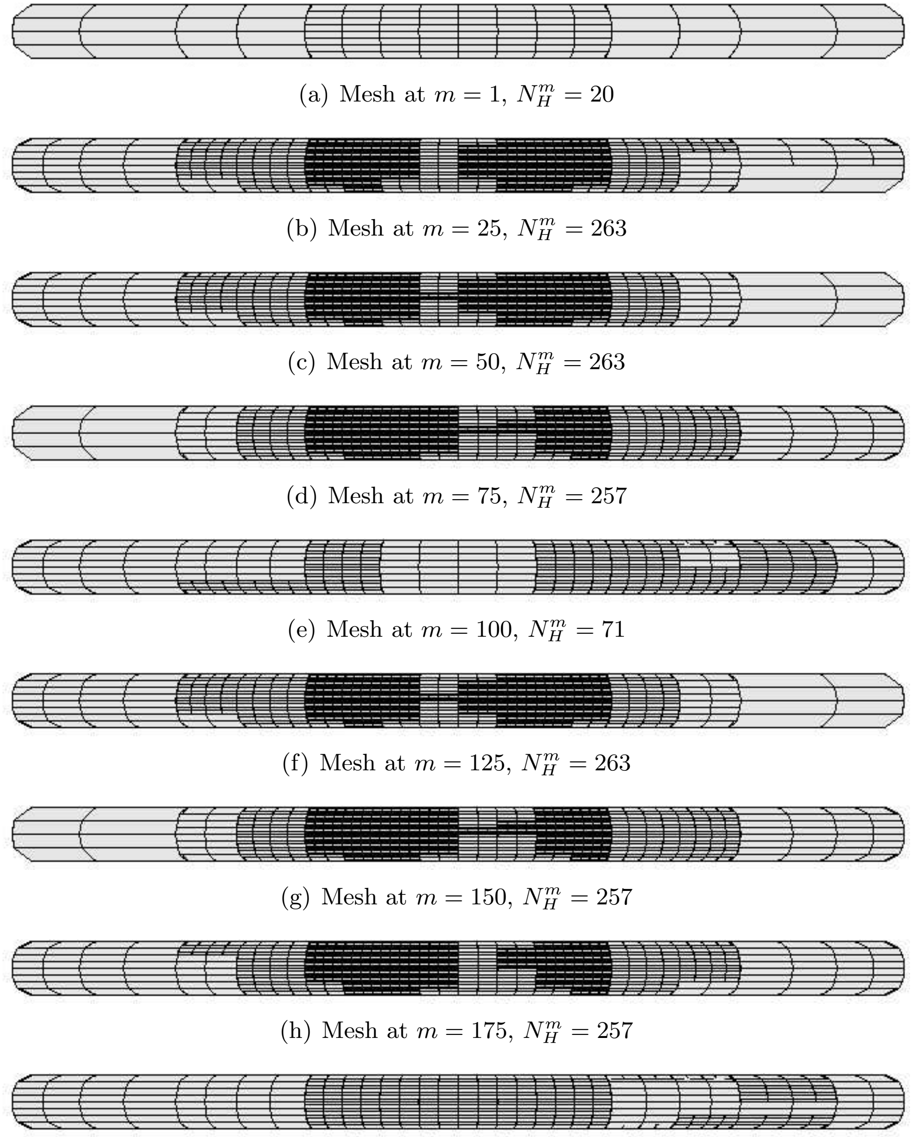
as reference solution. More detailed information about the DM approach based on $\eta$ are presented in Table 4.4.6. We observe that $\eta_c \ll \eta_e$ because $I_{\text{eff}} \approx I^e_{\text{eff}}$. Furthermore, the accuracy of the a posteriori error estimate is substantiated. The adaptive methods based on $\eta_e$ as well as the DM approach based on $\eta$ are very effective. They lead to comparable results. However, the CM approach based on $\eta$ is less effective. The results nearly correspond to the results based on uniform refinement.

## 4.5. Further Open Questions

This section is devoted to the discussion of further open questions in the context of the presented error estimators. Furthermore, the perspectives of the estimators are considered. In [**19**], a heuristic error indicator for dynamic contact problems is discussed. It is applied to the dynamic Signorini problem arising in the NC-shape grinding process. The basic domain is a quarter of the outer grinding wheel, see Figure 4.5.1(a). The rigid foundation is given by a free formed workpiece, which is depicted in Figure 4.5.1(b). The adaptive meshes created by the error indicator are presented in Figure 4.5.2, where

$$N^m_H := \dim\left(\Lambda^m_H\right).$$

See [**19**] for a detailed description of this example. The efficient application of the presented error estimator to such complex examples requires



(a) Mesh at $m = 1$, $N_H^m = 20$

(b) Mesh at $m = 25$, $N_H^m = 263$

(c) Mesh at $m = 50$, $N_H^m = 263$

(d) Mesh at $m = 75$, $N_H^m = 257$

(e) Mesh at $m = 100$, $N_H^m = 71$

(f) Mesh at $m = 125$, $N_H^m = 263$

(g) Mesh at $m = 150$, $N_H^m = 257$

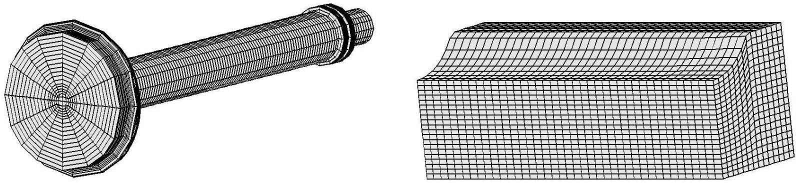
(h) Mesh at $m = 175$, $N_H^m = 257$

(i) Mesh at $m = 200$, $N_H^m = 71$

FIGURE 4.5.2. Meshes for different time steps of the dynamic Signorini problem, view on the contact zone of the grinding wheel

an improvement of the computational realisation. Parallelisation of different steps of the adaptive solution algorithm is one method to enhance the efficiency. Especially, the evaluation of the error estimator and the adaptive refinement are well suited for parallelisation, since the evaluation and the refinement in each time step can be performed independent of the other time steps. In linear problems with linear functionals of interest, the solution of the primal and dual problem may easily be calculated in parallel, since in this case the dual problem does not depend on the primal solution. The use of a multigrid scheme to solve the discrete problems is another possibility to enhance the adaptive method. Furthermore, a detailed analysis of the implementation is needed to find the time consuming operations, which should be enhanced. For instance, it could be more efficient to save the mass matrices during the solution of the primal problem and to reload them while solving the dual problem than to assemble them during the solution of the primal and of the dual problem. The mesh management and the calculation of integrals on different meshes can be enhanced further.

Beside the improvement of the realisation of the introduced adaptive method, more sophisticated adaptive methods are an interesting alternative. One possibility is given by multi-adaptive methods. They are discussed in the context of ODEs in [**80**]. In our temporal discretisation, the time step length $k$ does not depend on the spatial variable $x$. In multi-adaptive methods, the variation of $k$ with $x$ is permitted. Thus a very efficient discretisation is obtained. But the numerical solution of the discrete problem is problematic, since it does not lead to a classic time stepping scheme. We have derived the error estimate for a special space-time finite element approach, which is based on bilinear basis functions in space and linear basis functions in time. The use of higher order basis functions would be favoured. To obtain the maximum performance of the higher order ansatz, we have to apply *hp*-adaptive methods, i.e. we change the basis functions locally from mesh cell to mesh cell and from time step to time step. The derivation of the general a posteriori error estimate is the same, but the evaluation of the weights is much more



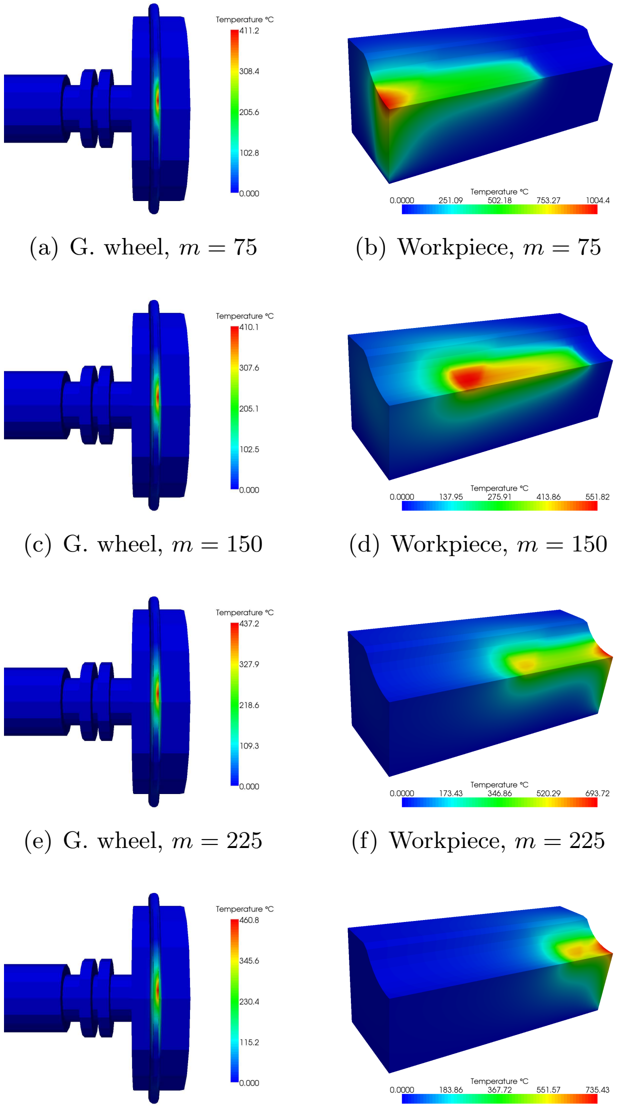
(a) Spindle-grinding wheel-system (b) Workpiece

Figure 4.5.3. Mesh of the spindle-grinding wheel-system and the workpiece

involved. Furthermore, the efficient implementation of such a method is complex. First results concerning *hp*-adaptvie methods for parabolic problems are described in [**101**].

In the introduction, the NC-shape grinding process has been discussed. From the engineering point of view, dynamic, rotational, damping, contact, frictional, and thermal effects have to be considered in the simulation of this process. The decisive parts of the grinding process, the grinding wheel and the workpiece, are simulated in detail. Furthermore, the spindle is explicitly included, where the stiffness of the grinding machine machine is modelled by elastic bearings. In Figure 4.5.3, the finite element meshes of the spindle-grinding wheel-system and of the workpiece are depicted. It should be mentioned that the geometry of the workpiece changes because of the material removal during the grinding process. Due to the different length scales, additional refinements in the contact zone between grinding wheel and workpiece are needed. In Figure 4.5.4, some simulation results are exemplarily presented. The model including all mentioned effects and the corresponding finite element discretisation scheme are explained in [**23**].

In this paragraph, we discuss the necessary extensions of the presented error estimator in order to apply it to the whole simulation of the NC-shape grinding process. The transfer of the presented approach to quasi-linear second order hyperbolic equations, which also includes a nonlinear term depending on the first temporal derivative of the solution, should be



(a) G. wheel, $m = 75$

(b) Workpiece, $m = 75$

(c) G. wheel, $m = 150$

(d) Workpiece, $m = 150$

(e) G. wheel, $m = 225$

(f) Workpiece, $m = 225$

(g) G. wheel, $m = 300$

(h) Workpiece, $m = 300$

FIGURE 4.5.4. Temperature distribution in the grinding wheel (view on the contact zone) and in the workpiece

straightforward. From the modelling point of view, this would allow to consider linear and nonlinear damping effects. Another important topic is the coupling of second order hyperbolic and parabolic problems. Such mixed problems arise, e.g., if thermal effects on mechanical structures are modelled. An appropriate coupling of the presented error estimator with the one described in [**99**] for parabolic problems should lead to a reasonable estimator for the mixed problems.

We restrict ourselves to the case of a simple linear differential operator in the contact problems. The extension to quasilinear differential operators is of great interest. The consideration of these operators in the presented error estimator for contact problems should be an easy task. Furthermore, we neglect frictional effects. The approach to derive the error estimators should be applicable to frictional contact problems, too. In the static case, the generalisation of this approach was done without any difficulty, see [**104**]. During frictional contact, energy is dissipated and usually converted into heat. This leads to a heat flux into the participating bodies. Thus, the consideration of thermal effects have to be integrated into the modelling and the discretisation of the contact problems. Combining an error estimator for mixed parabolic hyperbolic problems with an error estimator for frictional contact problems should lead to an adequate error estimator for thermomechanical contact problems. Our research on contact problems is based on the assumption that an elastic body hits a rigid foundation. In many situations, these assumption does not hold and we have to consider the contact between two or more elastic bodies. The application of the presented techniques to two body contact problems is a challenging task.

CHAPTER 5

# Conclusions

The aim of the thesis at hand was to develop space and time adaptive finite element methods for nonlinear hyperbolic problems of second order. The two main ingredients of the adaptive method, the underlying a posteriori error estimator and the adaptive refinement algorithm, were presented. We derived goal-oriented a posteriori error estimates for quasilinear hyperbolic equations of second order, static contact problems, and dynamic contact/impact problems on the basis of the DWR technique. Several numerical examples substantiated the accuracy of the estimate. Hence, the basic applicability of the presented error estimator to nonlinear hyperbolic problems was shown. However, we do not and possibly cannot prove that the numerically evaluated value of the error identity converges to the continuous value in the considered setting. As outlined in Section 4.5, the efficiency of the error estimation can and has to be improved further. Beside the acceleration of the implementation, simplifications of the error estimator are an alternative. For instance, we can confine ourselves to the evaluation of the primal residual, if we are only interested in adaptive refinement. The restriction to the primal residual reduces the computing time for the evaluation by more than 50 %. A second opportunity to speed up the evalution is provided by simplification of the discrete dual problem. The solution on coarser meshes does not enhance the efficiency, because the effort for the interpolation on the fine grids offsets the saved computing time during the solution process.

Refinement indicators, which are the basis of the adaptive refinement method, were deduced from the error estimator. In view of the a priori error estimates and of the definition of the higher order interpolation, it is required that the mesh sequence contains no temporal and spatial hanging nodes of degree greater than one and that all spatial meshes have patch structure. This structure is usually destroyed during the adaptive refinement. Thus, we presented a regularisation algorithm, which restores the mesh structure. The efficiency of the adaptive method w.r.t. the total number of mesh cells as well as computing time was shown by numerical examples. But we also obtain less efficient adaptive meshes, especially in the DM approach, if the solution on coarse meshes does not exactly represent the problem behaviour. In dynamic contact/impact problems, the solution algorithm detects the first contact between the elastic body and the rigid foundation too late. Consequently, additional refinements in the contact zone are only performed in later time steps, which leads to a less accurate approximation. The treatment of the described difficulty is a basic open question in adaptivity for nonlinear problems. A second conclusion from the numerical experiences is that we have to balance the flexibility of the adaptive approach and the complexity of the underlying algorithms. In some examples, the simpler CM approach is more efficient than the DM approach. Criteria for balancing flexibility and complexity in advance are needed.

The presented space and time adaptive finite element methods are the foundation of the adaptive simulation of the NC-shape grinding process. It relies on the accurate and efficient determination of the dynamic behaviour of the spindle-grinding wheel-system and of the contact between the grinding wheel and the workpiece, since all other effects depend on the dynamics and on the contact situation. The next steps towards the integral adaptive simulation of the NC-shape grinding process consist in the successive extension of the presented error estimator towards problems including damping, rotational, frictional, and thermal effects.

# Bibliography

[1] R. A. Adams. *Sobolev spaces.* Pure and Applied Mathematics. Academic Press, New York, 1975.

[2] S. Adjerid. A posteriori finite element error estimation for second-order hyperbolic problems. *Comput. Methods Appl. Mech. Engng.*, 191:4699–4719, 2002.

[3] J. Ahn and D. E. Stewart. Dynamic frictionless contact in linear viscoelasticity. *IMA J. Numer. Anal.*, 1:43–71, 2009.

[4] M. Ainsworth and J. T. Oden. *A Posteriori Error Estimation in Finite Element Analysis.* John Wiley & Sons, New York, 2000.

[5] M. Ainsworth, J. T. Oden, and C. Y. Lee. Local a posteriori error estimators for variational inequalities. *Numer. Methods Partial Differ. Equ.*, 9:23–33, 1993.

[6] F. Armero and E. Petőcz. Formulation and analysis of conserving algorithms for frictionless dynamic contact/impact problems. *Comput. Methods Appl. Mech. Engng.*, 158:269–300, 1998.

[7] D. Aubry, D. Lucas, and B. Tie. Adaptive strategy for transient/coupled problems. Applications to thermoelasticity and elastodynamics. *Comput. Methods Appl. Mech. Engng.*, 176:41–50, 1999.

[8] L. Bales and I. Lasiecka. Continuous finite elements in space and time for the nonhomogeneous wave equation. *Comput. Math. Appl.*, 27(3):91–102, 1994.

[9] W. Bangerth. Adaptive Finite-Elemente-Methoden zur Lösung der Wellengleichung mit Anwendung in der Physik der Sonne. Diploma thesis, Ruprecht-Karls-Universität Heidelberg, 1998.

[10] W. Bangerth. Mesh adaptivity and error control for a finite element approximation of the elastic wave equation. In *Proceedings of Fifth International Conference on Mathematical and Numerical Aspects of Wave Propagation (Waves2000)*, pages 725–729, Santiago de Compostela, Spain, 2000.

[11] W. Bangerth and R. Rannacher. Finite element approximation of the acoustic wave equation: Error control and mesh adaptation. *East-West J. Numer. Math.*, 7(4):263–282, 1999.

[12] W. Bangerth and R. Rannacher. Adaptive finite element techniques for the acoustic wave equation. *J. Comp. Acoustics*, 9(2):575–591, 2001.

[13] W. Bangerth and R. Rannacher. *Adaptive Finite Element Methods for Differential Equations*. Birkhäuser, Basel, 2003.

[14] S. Bartels and C. Carstensen. Averaging techniques yield reliable a posteriori finite element error control for obstacle problems. *Numer. Math.*, 99:225–249, 2004.

[15] K. J. Bathe and A. B. Chaudhary. A solution method for static and dynamic analysis of three-dimensional contact problems with friction. *Comput. Struct.*, 24(6):855–873, 1986.

[16] R. Becker. Adaptive finite elements for optimal control problems. Habilitation thesis, Ruprecht-Karls-Universität Heidelberg, 2001.

[17] R. Becker and R. Rannacher. An optimal control approach to a posteriori error estimation and mesh adaptation in finite element methods. In A. Iserles, editor, *Acta Numerica 2000*, pages 1–101. Cambridge University Press, 2001.

[18] C. Bernadi and E. Süli. Time and space adaptivity for the second-order wave equation. *Math. Models Methods Appl. Sci.*, 15(2):199–225, 2005.

[19] D. Biermann, H. Blum, T. Jansen, A. Rademacher, A. V. Scheidler, A. Schröder, and K. Weinert. Space adaptive finite element methods for dynamic signorini problems in the simulation of the NC-shape grinding process. In *Proceedings of the 1st CIRP International Conference on Process Machine Interactions (PMI)*, pages 309–316, 2008.

[20] D. Biermann, H. Blum, T. Jansen, A. Rademacher, A. V. Scheidler, and K. Weinert. Experimental analyses to develop models for NC-shape grinding with a toroid grinding wheel. In *Proceedings of the 1st CIRP International Conference on Process Machine Interactions (PMI)*, pages 279–288, 2008.

[21] H. Blum, T. Jansen, A. Rademacher, and K. Weinert. Finite elements in space and time for dynamic contact problems. *Int. J. Numer. Meth. Engng.*, 76:1632–1644, 2008.

[22] H. Blum, H. Kleemann, H. Ludwig, A. Rademacher, M. Schäckelhoff, A. Schröder, O. Schulz, M. Stiemer, K. Taebi, and S. Wiedemann. SOFAR: Scientific Object oriented Finite element library for Application and Research. www.mathematik.tu-dortmund.de/lsx/research/software/sofar/index.html, 2009.

[23] H. Blum, H. Kleemann, A. Rademacher, and A. Schröder. On solving frictional contact problems. Part II: Dynamic case. Ergebnisberichte Angewandte Mathematik, No. 377, www.mathematik.uni-dortmund.de/lsiii/static/preprintfb.nhtml, 2008.

[24] H. Blum, A. Rademacher, and A. Schröder. Space adaptive finite element methods for dynamic obstacle problems. *ETNA, Electron. Trans. Numer. Anal.*, 32:162–172, 2008.

[25] H. Blum, A. Rademacher, and A. Schröder. Space adaptive finite element methods for dynamic signorini problems. *Comput. Mech.*, 44(4):481–491, 2009.

[26] H. Blum and F. T. Suttmeier. An adaptive finite element discretisation for a simplified signorini problem. *Calcolo*, 37:65–77, 2000.

[27] F. A. Bornemann and M. Schemann. An adaptive Rothe method for the wave equation. *Comput. Visual Sci.*, 1:137–144, 1998.

[28] M. Braack. *An Adaptive Finite Element Method for Reactive Flow Problems.* PhD thesis, Ruprecht-Karls-Universität Heidelberg, 1998.

[29] M. Braack and A. Ern. A posteriori control of modeling errors and discretisation errors. *Multiscale Model. Simul.*, 1(2):221–238, 2003.

[30] D. Braess. *Finite Elemente. Theorie, schnelle Löser und Anwendungen in der Elastizitätstheorie.* Springer, Berlin, 3. edition, 2003.

[31] D. Braess. A posteriori error estimators for obstacle problems - another look. *Numer. Math.*, 101:415–421, 2005.

[32] D. Braess, C. Carstensen, and R. H. W. Hoppe. Convergence analysis of a conforming adaptive finite element method for an obstacle problem. *Numer. Math.*, 107:455–471, 2007.

[33] D. Braess, C. Carstensen, and R. H. W. Hoppe. Error reduction in adaptive finite element approximations of elliptic obstacle proplems. *J. Comput. Math.*, 27(2-3):148–169, 2009.

[34] G. F. Carey and J. T. Oden. *Finite Elements, volume 3: Computational aspects.* Prentice-Hall, Englewood Cliffs, 1984.

[35] C. Carstensen and J. Hu. Hanging nodes in the unifying theory of a posteriori finite element error control. *J. Comput. Math.*, 27(2-3):215–236, 2009.

[36] T. Cazenave and A. Haraux. *An Introduction to Semilinear Evolution Equations.* Oxford University Press, Oxford, 1998.

[37] J. Céa. *Lectures on optimization - theory and algorithms.* Springer, Berlin, 1978.

[38] Z. Chen and R. Nochetto. Residual type a posteriori error estimates for elliptic obstacle problems. *Numer. Math.*, 84:527–548, 2000.

[39] A. Czekanski, N. El-Abbasi, and S. A. Meguid. Optimal time integration parameters for elastodynamic contact problems. *Commun. Numer. Meth. Engng.*, 17:379–384, 2001.

[40] A. Czekanski, N. El-Abbasi, S. A. Meguid, and M. H. Refaat. On the elastodynamic solution of frictional contact problems using variational inequalities. *Int. J. Numer. Meth. Engng.*, 50:611–627, 2001.

[41] A. Czekanski and S. A. Meguid. Analysis of dynamic frictional contact problems using variational inequalities. *Finite Elem. Anal. Des.*, 37:6861–879, 2001.

[42] R. Dautray and J.-L. Lions. *Mathematical Analysis and Numerical Methods for Science and Technology: Evolution Problems I, Vol. 5*. Springer-Verlag, Berlin, 1992.

[43] T. A. Davis. UMFPACK version 4.4 user guide. Technical report, Dept. of Computer and Information Science and Engineering, University of Florida, 2005.

[44] P. Deuflhard. *Newton methods for nonlinear problems. Affine invariance and adaptive algorithms*, volume 35 of *Springer series in computational mathematics*. Springer, Berlin, 2004.

[45] P. Deuflhard, R. Krause, and S. Ertel. A contact-stabilized Newmark method for dynamical contact problems. *Int. J. Numer. Meth. Engng.*, 73:1274–1290, 2008.

[46] I. Ekeland and R. Temam. *Convex analysis and variational problems*. Studies in Mathematics and its Applications. North-Holland Publishing Company, Amsterdam, 1976.

[47] K. Eriksson, D. Estep, P. Hansbo, and C. Johnson. *Computational Differential Equations*. Cambridge University Press, Cambridge, 1996.

[48] L. C. Evans. *Partial Differential Equations*. American Mathematical Society, Providence, 1998.

[49] D. A. French and T. E. Peterson. A continuous space-time finite element method for the wave equation. *Math. Comput.*, 65(214):491–506, 1996.

[50] D. A. Geba. A local well-posedness result for the quasilinear wave equation in $\mathbb{R}^{2+1}$. *Commun. Partial Differ. Equations*, 29(3-4):323–360, 2004.

[51] P. E. Gill, W. Murray, and M. A. Saunders. User's guide for SQOPT version 5.3: A fortran package for large-scale linear and quadratic programming. Technical report, University of California, San Diego and Stanford University, Stanford, 1999.

[52] P. E. Gill, W. Murray, and M. A. Saunders. User's guide for SNOPT version 7: Software for large-scale nonlinear programming. Technical report, University of California, San Diego and Stanford University, Stanford, 2006.

[53] C. Glocker and P. D. Panagiotopoulos. Inequality constraints with elastic impacts in deformable bodies. The convex case. *Arch. Appl. Mech.*, 70:349–365, 2000.

[54] C. Großmann and H.-G. Roos. *Numerische Behandlung partieller Differentialgleichungen*. Teubner, 2005.

[55] C. Hager, S. Hüeber, and B. Wohlmuth. A stable energy conserving approach for frictional contact problems based on quadrature formulas. *Int. J. Numer. Meth. Engng.*, 73:205–225, 2008.

[56] J. Haslinger and J. Lovišek. Mixed variational formulation of unilateral problems. *Commentat. Math. Univ. Carol.*, 21:231–246, 1980.

[57] M. Hermann. *Numerische Mathematik*. Oldenbourg Verlag, München, Wien, 2001.

[58] I. Hlavaček, J. Haslinger, J. Nečas, and J. Lovišek. *Solution of Variational Inequalities in Mechanics*. Springer, Berlin, 1988.

[59] R. H. W. Hoppe and R. Kornhuber. Adaptive multilevel methods for obstacle problems. *SIAM J. Numer. Anal.*, 31:301–323, 1994.

[60] T. J. R. Hughes. *The Finite Element Method*. Dover Publications, Inc., Mineola, 2000.

[61] T. J. R. Hughes and G. M. Hulbert. Space-time finite element methods for elastodynamics: Formulations and error estimates. *Comput. Methods Appl. Mech. Engng.*, 66:339–363, 1988.

[62] T. J. R. Hughes and G. M. Hulbert. Space-time finite element methods for second-order hyperbolic equations. *Comput. Meth. Appl. Mech. Engng.*, 84:327–348, 1990.

[63] T. J. R. Hughes, T. Kato, and J. E. Marsden. Well-posed quasilinear second order hyperbolic systems with applications to nonlinear elastodynamics and general relativity. *Arch. Rational Mech. Anal.*, 63(3):273–294, 1976.

[64] G. M. Hulbert. Time finite element methods for structural dynamics. *Int. J. Numer. Meth. Engng.*, 33:307–331, 1992.

[65] T. Jansen. *Entwicklung einer Simulation für den NC-Formschleifprozess mit Torusschleifscheiben*. PhD thesis, Technische Universität Dortmund, 2007.

[66] C. Johnson. Adaptive finite element methods for the obstacle problem. *Math. Models Meth. Appl. Sci.*, 2:483–487, 1992.

[67] C. Johnson. Discontinuous Galerkin finite element methods for second order hyperbolic problems. *Comput. Meth. Appl. Mech. Engng.*, 107:117–129, 1993.

[68] C. Kane, J. E. Marsden, M. Ortiz, and E. A. Repetto. Finite element analysis of nonsmooth contact. *Comput. Meth. Appl. Mech. Engng.*, 180:1–26, 1999.

[69] O. Karakashian and C. Makridakis. Convergence of a continuous galerkin method with mesh modification for nonlinear wave equations. *Math. Comput.*, 74(249):85–102, 2005.

[70] N. Kikuchi and J. T. Oden. *Contact Problems in Elasticity: A Study of Variational Inequalities and Finite Element Methods*. SIAM Studies in Applied Mathematics. SIAM, Philadelphia, 1988.

[71] S. Kizio. *Adaptive Finite-Elemente-Algorithmen in der Strukturdynamik*. PhD thesis, Universität Fridericiana zu Karlsruhe, 2008.

[72] S. Kizio and K. Schweizerhof. Time integration error estimation for continuous galerkin schemes. *PAMM, Proc. Appl. Math. Mech.*, 5:675–676, 2005.

[73] C. Klapproth, P. Deuflhard, and A. Schiela. A perturbation result for dynamical contact problems. *Zuse-Institut Berlin*, Preprint 08-27, 2008. Accepted in Numer. Math. TMA.

[74] C. Klapproth, A. Schiela, and P. Deuflhard. Consistency results for the contact-stabilized newmark method. *Zuse-Institut Berlin*, Preprint 09-06, 2009.

[75] T. A. Laursen. *Computational Contact and Impact Mechanics*. Springer, Berlin, Heidelberg, New York, 2002.

[76] T. A. Laursen and V. Chawla. Design of energy conserving algorithms for frictionless dynamic contact problems. *Int. J. Numer. Meth. Engng.*, 40:863–886, 1997.

[77] X. Li and N. E. Wiberg. Structural dynamic analysis by a time-discontinuous galerkin finite element method. *Int. J. Numer. Meth. Engng.*, 39:2131–2152, 1996.

[78] X. Li and N. E. Wiberg. Implementation and adaptivity of a space-time finite element method for structural dynamics. *Comput. Meth. Appl. Mech. Engng.*, 156:211–229, 1998.

[79] X. Li, N. E. Wiberg, and L. F. Zeng. A posteriori local error estimation and adaptive time-stepping for Newmark integration in dynamic analysis. *Earth. Engng. Struct. Dyn.*, 21:555–571, 1992.

[80] A. Logg. Multi-adaptive galerkin methods for ODEs I. *SIAM J. Sci. Comput.*, 24(6):1879–1902, 2003.

[81] A. Logg. Multi-adaptive galerkin methods for ODEs III: A priori error estimates. *SIAM J. Numer. Anal.*, 43(6):2624–2646, 2006.

[82] J. A. C. Martins and J. T. Oden. Existence and uniqueness results for dynamic contact problems with nonlinear normal and friction interface laws. *Nonlinear Anal., Theory, Methods, Appl.*, 11(3):407–428, 1987.

[83] A. Maute. *Fehlerkontrolle bei Finite-Element-Methoden in der linearen Strukturdynamik*. PhD thesis, Universität Stuttgart, 2001.

[84] T. W. McDevitt and T. A. Laursen. A mortar-finite element formulation for frictional contact problems. *Int. J. Numer. Meth. Engng.*, 48:1525–1547, 2000.

[85] D. Meidner. *Adaptive Space-Time Finite Element Methods for Optimization Problems Governed by Nonlinear Parabolic Systems*. PhD thesis, Ruprecht-Karls-Universität Heidelberg, 2008.

[86] F. Natterer. Optimale $L^2$-Konvergenz finiter Elemente bei Variationsungleichungen. *Bonn. Math. Schrift.*, 89:1–12, 1976.

[87] J. Neumann. *Anwendung von adaptiven Finite Element Algorithmen auf Probleme der Strukturdynamik*. PhD thesis, Universität Fridericiana zu Karlsruhe, 2004.

[88] R. H. Nochetto, K. G. Siebert, and A. Veeser. Pointwise a posteriori error control for elliptic obstacle problems. *Numer. Math.*, 95:163–195, 2003.

[89] P. D. Panagiotopoulos. *Inequality Problems in Mechanics and Applications, Convex and Nonconvex Energy Functions*. Birkhäuser, Basel, 1985.

[90] A. Pandolfi, C. Kane, J. E. Marsden, and M. Ortiz. Time-discretized variational formulation of non-smooth frictional contact. *Int. J. Numer. Meth. Engng.*, 53:1801–1829, 2002.

[91] P. Papadopoulos and R. L. Taylor. On a finite element method for dynamic contact/impact problems. *Int. J. Numer. Meth. Engng.*, 36:2123–2140, 1993.

[92] M. Puso and T. A. Laursen. A 3D contact smoothing method using Gregory patches. *Int. J. Numer. Meth. Engng.*, 54:1161–1194, 2002.

[93] A. Rademacher. Finite-Elemente-Diskretisierungen für dynamische Probleme mit Kontakt. Diploma thesis, Universität Dortmund, 2005.

[94] R. Rannacher. Numerische Mathematik 2: Numerik partieller Differentialgleichungen. Lecture notes, Ruprecht-Karls-Universität Heidelberg, http://numerik.iwr.uni-heidelberg.de/ lehre/notes/, 2006.

[95] R. Rannacher. A posteriori Fehlerschätzung, Gitteradaption und darüber hinaus. Presentation: Arbeitskreis für numerische Mathematik Dortmund-Siegen-Wuppertal, Universität Siegen, 2006.

[96] R. Rannacher. Numerische Mathematik 1: Numerik gewöhnlicher Differentialgleichungen. Lecture note, Ruprecht-Karls-Universität Heidelberg, http://numerik.iwr.uni-heidelberg.de/ lehre/notes/, 2009.

[97] T. Richter. Funktionalorientierte Gitteroptimierung bei der Finite-Elemente-Approximation elliptischer Differentialgleichungen. Diploma thesis, Ruprecht-Karls-Universität Heidelberg, 2001.

[98] T. Richter. *Parallel Multigrid Method for Adaptive Finite Elements with Application to 3D Flow Problems*. PhD thesis, Ruprecht-Karls-Universität Heidelberg, 2005.

[99] M. Schmich and B. Vexler. Adaptivity with dynamic meshes for space-time finite element discretizations of parabolic equations. *SIAM J. Sci. Comput.*, 30:369–393, 2008.

[100] A. Schmidt and K. G. Siebert. *Design of Adaptive Finite Element Software*. Lecture Notes in Computational Science and Engineering. Springer, Berlin, 2004.

[101] D. Schötzau and C. Schwab. Time discretization of parabolic problems by the hp-version of the discontinuous galerkin finite element method. *SIAM J. Numer. Anal.*, 38(3):837–875, 2000.

[102] A. Schröder. Error control in *h*- and *hp*-adaptive FEM for Signorini's problem. *Institute for Mathematics, Humboldt Universität zu Berlin, Preprint*, 09-05, 2009.

[103] A. Schröder. Goal-oriented error control in *h*- and *hp*- adaptive FEM for Signorini's problem. In *Proceedings of Computer Methods in Mechanics*, 2009.

[104] A. Schröder. *Fehlerkontrollierte adaptive h- und hp-Finite-Elemente-Methoden für Kontaktprobleme mit Anwendungen in der Fertigungstechnik*. PhD thesis, Universität Dortmund, 2005.

[105] K. Siebert and A. Veeser. A unilaterally constrained quadratic minimization with adaptive finite elements. *SIAM J. Optim.*, 18:260–289, 2007.

[106] F. T. Suttmeier. On a direct approach to adaptive FE-discretisations for elliptic variational inequalities. *J. Numer. Math.*, 13:73–80, 2005.

[107] F. T. Suttmeier. Adaptive computational methods for variational inequalities based on mixed formulations. *Int. J. Numer. Meth. Engng*, 68:1180–1208, 2006.

[108] D. Talaslidis and P. D. Panagiotopoulos. A linear finite element approach to the solution of the variational inequalities arising in contact problems of structural dynamics. *Int. J. Numer. Meth. Engng.*, 18:1505–1520, 1982.

[109] V. Thomée. *Galerkin finite element methods for parabolic problems.* Number 1054 in Lecture notes in mathematics. Springer, Berlin, 1984.

[110] A. Veeser. Efficient and reliable a posteriori error estimators for elliptic obstacle problems. *SIAM J. Numer. Anal.*, 39:146–167, 2001.

[111] R. Verfürth. *A Review of A Posteriori Error Estimation and Adaptive Mesh-Refinement Technique.* Wiley and Teubner, Chichester, 1996.

[112] K. Weinert, H. Blum, T. Jansen, T. Mohn, M. Noyen, and A. Rademacher. Verfahrensspezifische Modellbildung für die Belastung beim Schleifen. In H.-W. Hoffmeister and B. Denkena, editors, *Jahrbuch Schleifen, Honen, Läppen und Polieren*, volume 63, pages 24–38. Vulkan-Verlag, Essen, 2007.

[113] K. Weinert, H. Blum, T. Jansen, T. Mohn, and A. Rademacher. Angepasste Simulationstechnik zur Analyse NC-gesteuerter Formschleifprozesse. *ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb*, 101(7-8):422–425, 2006.

[114] K. Weinert, H. Blum, T. Jansen, and A. Rademacher. Simulation based optimization of the NC-shape grinding process with toroid grinding wheels. *Production Engineering - Research and Development*, 1:245–252, 2007.

[115] D. Werner. *Funktionalanalysis.* Springer, Berlin, 2002.

[116] B. I. Wohlmuth. An a posteriori error estimator for two-body contact problems on non-matching meshes. *J. Sci. Comput.*, 33:25–45, 2007.

[117] P. Wriggers. *Computational Contact Mechanics.* John Wiley & Sons Ltd, Chichester, 2002.

[118] P. Wriggers, T. Vu Van, and E. Stein. Finite element formulation of large deformation impact-contact problems with friction. *Comput. Struct.*, 37:319–331, 1990.

[119] Y. M. Xie and O. C. Zienkiewicz. A simple error estimator and adaptive time stepping procedure for dynamic analysis. *Earth. Engrg. Struct. Dyn.*, 20:871–887, 1991.

[120] O. C. Zienkiewicz and J. Z. Zhu. The superconvergent patch recovery and a posteriori error estimates. Part I, II. *Int. J. Numer. Meth. Engng.*, 33(7):1331–1364, 1365–1382, 1992.

CHAPTER A

# Practical Realisation of the General Concepts

## A.1. Quadrature Formulas

Throughout this thesis, integrals of the form $\int_{I_m} f(t)\,dt$ have to be evaluated numerically. For this purpose, we use quadrature formulas $Q$ with

$$\int_{I_m} f(t)\,dt = Q(f) + E(f),$$

where $E$ is the error term. The following three quadrature formulas of different order are applied:

(1) For approximation of integrals of piecewise constant functions, the box formula

$$Q(f) = k_m f^m$$

is used. The error term is given by

$$E(f) = -k_m^2 f^{(1)}(\xi), \quad \xi \in I_m.$$

(2) The trapezoidal rule

$$Q(f) = \frac{1}{2} k_m \left[ f^m + f^{m-1} \right]$$

is exact for piecewise linear functions. The error equation

$$E(f) = -\frac{1}{12} k_m^3 f^{(2)}(\xi), \quad \xi \in I_m,$$

holds.

(3) The evaluation of integrals of piecewise quadratic polynomials is done by the Simpson formula

$$Q(f) = \frac{1}{6} k_m \left[ f^m + 4 f^{m-\frac{1}{2}} + f^{m-1} \right].$$

The error term is given by

$$E(f) = -\frac{1}{2880} k_m^5 f^{(4)}(\xi), \quad \xi \in I_m.$$

The Simpson formula is of fifth order in $k_m$ and is exact for polynomials of maximum degree three.

## A.2. Interpolation of Higher Order in Time

For the evaluation of the a posteriori error estimate, two different interpolation methods of higher order in time are needed. The first one is $i_k^{(1)}$, which is linear and is used in the case of piecewise constant trial functions. The other one is $i_{2k}^{(2)}$, which is quadratic. Piecewise linear functions are extrapolated by this method. The interpolation operator $i_k^{(1)}$ is defined as

$$i_k^{(1)} v_{kh}(t) := \frac{t_m - t}{k_m} v_{kh}^{m-1} + \frac{t - t_{m-1}}{k_m} v_{kh}^m$$

with $t \in I_m$ and $v_{kh} \in W_{kh}$. The evaluation of the interpolation at different time instances results in the terms

$$\begin{aligned} i_k^{(1)} v_{kh}\left(t_m\right) &= v_{kh}^m \\ i_k^{(1)} v_{kh}\left(t_{m-1}\right) &= v_{kh}^{m-1} \\ i_k^{(1)} v_{kh}\left(t_{m-\frac{1}{2}}\right) &= \frac{1}{2} k_m \left[ v_{kh}^m + v_{kh}^{m-1} \right]. \end{aligned}$$

The interpolation operator $i_{2k}^{(2)}$ is defined as

$$\begin{aligned} i_{2k}^{(2)} v_{kh}(t) := & \frac{\left(t_m - t\right)\left(t_{m+1} - t\right)}{k_m\left(k_m + k_{m+1}\right)} v_{kh}^{m-1} + \frac{\left(t - t_{m-1}\right)\left(t_{m+1} - t\right)}{k_m k_{m+1}} v_{kh}^m \\ & + \frac{\left(t - t_{m-1}\right)\left(t - t_m\right)}{\left(k_m + k_{m+1}\right) k_{m+1}} v_{kh}^{m+1} \end{aligned}$$

for $t \in I_m \cup I_{m+1}$ and $v_{kh} \in V_{kh}$. The basic evaluation terms are

$$i_{2k}^{(2)} v_{kh}\left(t_{m-1}\right) = v_{kh}^{m-1}, \quad i_{2k}^{(2)} v_{kh}\left(t_m\right) = v_{kh}^m, \quad i_{2k}^{(2)} v_{kh}\left(t_{m+1}\right) = v_{kh}^{m+1}$$

and

$$\begin{aligned}
i_{2k}^{(2)} v_{kh}\left(t_{m-\frac{1}{2}}\right) &= \frac{1}{2} \frac{k_{m+1} + \frac{1}{2} k_m}{k_m + k_{m+1}} v_{kh}^{m-1} + \frac{1}{2}\left(1 + \frac{k_m}{2k_{m+1}}\right) v_{kh}^m \\
&\quad - \frac{1}{4} \frac{k_m^2}{\left(k_m + k_{m+1}\right) k_{m+1}} v_{kh}^{m+1} \\
i_{2k}^{(2)} v_{kh}\left(t_{m+\frac{1}{2}}\right) &= -\frac{1}{4} \frac{k_{m+1}^2}{k_m\left(k_m + k_{m+1}\right)} v_{kh}^{m-1} + \frac{1}{2}\left(1 + \frac{k_{m+1}}{2k_m}\right) v_{kh}^m \\
&\quad + \frac{1}{2} \frac{k_m + \frac{1}{2} k_{m+1}}{k_m + k_{m+1}} v_{kh}^{m+1}.
\end{aligned}$$

## A.3. Temporal Error Estimate

The temporal part of the discretisation error is estimated by $\eta_k$. It is derived in Section 1.3. Here, the general form is concretised for the implementation.

### A.3.1. Terms in the Temporal Error Estimator.

The temporal error estimator $\eta_k^{n,m}$ is given by

$$\begin{aligned}
\eta_k^{n,m} = \sum_{j=1}^{6} \eta_{k,j}^{n,m} = & -\frac{1}{2} A\left(w_{kh}\right)\left(\Pi_k^{(1)} z_{kh}\right)_m \\
& + \frac{1}{2} J'\left(w_{kh}\right)\left(\Pi_{2k}^{(2)} w_{kh}\right)_m \\
& - \frac{1}{2} A'\left(w_{kh}\right)\left(\Pi_{2k}^{(2)} w_{kh}, z_{kh}\right)_m
\end{aligned}$$

for $m = 1, \ldots, M$. The single terms are

$$\begin{aligned}
\eta_{k,1}^{n,m} &:= -\frac{1}{2}\left(\left(\dot{u}_{kh} - v_{kh}, i_k^{(1)} \bar{u}_{kh} - \bar{u}_{kh}\right)\right)_m \\
&\quad - \frac{1}{2}\left(\left(\dot{v}_{kh}, i_k^{(1)} \bar{v}_{kh} - \bar{v}_{kh}\right)\right)_m, \\
\eta_{k,2}^{n,m} &:= -\frac{1}{2}\left(a\left(u_{kh}\right)\left(i_k^{(1)} \bar{v}_{kh} - \bar{v}_{kh}\right)\right)_m + \frac{1}{2}\left(\left(f, i_k^{(1)} \bar{v}_{kh} - \bar{v}_{kh}\right)\right)_m
\end{aligned}$$

$$
\begin{aligned}
& +\frac{1}{2}\left(\left(q, i_k^{(1)}\bar{v}_{kh}-\bar{v}_{kh}\right)_{\Gamma_N}\right)_m, \\
\eta_{k,3}^{n,m} &:= \frac{1}{2}\left(J_{1,u}'\left(w_{kh}\right)\left(i_{2k}^{(2)}u_{kh}-u_{kh}\right)\right)_m, \\
\eta_{k,4}^{n,m} &:= \frac{1}{2}\left(J_{1,v}'\left(w_{kh}\right)\left(i_{2k}^{(2)}v_{kh}-v_{kh}\right)\right)_m, \\
\eta_{k,5}^{n,m} &:= \frac{1}{2}\left(\left(\dot{\bar{v}}_{kh}+\bar{u}_{kh}, i_{2k}^{(2)}v_{kh}-v_{kh}\right)\right)_m \\
& +\frac{1}{2}\left(\left[\bar{u}\right]_{m+1}, \left(i_{2k}^{(2)}u_{kh}-u_{kh}\right)\left(t_m\right)\right) \\
& +\frac{1}{2}\left(\left(\dot{\bar{u}}_{kh}, i_{2k}^{(2)}u_{kh}-u_{kh}\right)\right)_m \\
& +\frac{1}{2}\left(\left[\bar{v}\right]_{m+1}, \left(i_{2k}^{(2)}v_{kh}-v_{kh}\right)\left(t_m\right)\right), \\
\eta_{k,6}^{i,m} &:= -\frac{1}{2}\left(a'\left(u_{kh}\right)\left(i_{2h}^{(2)}\left(i_{2k}^{(2)}u_{kh}-u_{kh}\right), \bar{v}_{kh}\right)\right)_m .
\end{aligned}
$$

Using appropriate quadrature rules, we obtain the following form of the terms:

$$
\begin{aligned}
\eta_{k,1}^{n,m} &= \frac{1}{4}\left(u_{kh}^m-u_{kh}^{m-1}, \bar{u}_{kh}^m-\bar{u}_{kh}^{m-1}\right) \\
& -\frac{k_m}{12}\left(v_{kh}^m+2v_{kh}^{m-1}, \bar{u}_{kh}^m-\bar{u}_{kh}^{m-1}\right) \\
& +\frac{1}{4}\left(v_{kh}^m-v_{kh}^{m-1}, \bar{v}_{kh}^m-\bar{v}_{kh}^{m-1}\right), \\
\eta_{k,2}^{n,m} &= -\frac{k_m}{12}\left[a\left(u_{kh}^{m-1}\right)\left(\bar{v}_{kh}^{m-1}-3\bar{v}_{kh}^m\right)-2a\left(u_{kh}^m\right)\left(\bar{v}_{kh}^m\right)\right] \\
& -\frac{k_m}{6}a\left(\frac{1}{2}\left(u_{kh}^m+u_{kh}^{m-1}\right)\right)\left(\bar{v}_{kh}^m+\bar{v}_{kh}^{m-1}\right) \\
& +\frac{k_m}{12}\left[-2\left(f^m, \bar{v}_{kh}^m\right)+\left(f^{m-1}, \bar{v}_{kh}^{m-1}-3\bar{v}_{kh}^m\right)\right] \\
& +\frac{k_m}{6}\left(f^{m-\frac{1}{2}}, \bar{v}_{kh}^m+\bar{v}_{kh}^{m-1}\right) \\
& +\frac{k_m}{12}\left[-2\left(q^m, \bar{v}_{kh}^m\right)_{\Gamma_N}+\left(q_{kh}^{m-1}, \bar{v}_{kh}^{m-1}-3\bar{v}_{kh}^m\right)_{\Gamma_N}\right] \\
& +\frac{k_m}{6}\left(q^{m-\frac{1}{2}}, \bar{v}_{kh}^m+\bar{v}_{kh}^{m-1}\right)_{\Gamma_N},
\end{aligned}
$$

$$\begin{aligned}
\eta_{k,3}^{n,m} &= -\frac{k_m}{6}\left[J_{1,u}'\left(w_{kh}^m\right)\left(u_{kh}^m\right) + J_{1,u}'\left(w_{kh}^{m-1}\right)\left(u_{kh}^{m-1}\right)\right] \\
&\quad + \frac{k_m}{3}J_{1,u}'\left(\frac{1}{2}\left(w_{kh}^m + w_{kh}^{m-1}\right)\right)\left(i_{2k}^{(2)}u_{kh}^{m-\frac{1}{2}}\right), \\
\eta_{k,4}^{n,m} &= -\frac{k_m}{6}\left[J_{1,v}'\left(w_{kh}^m\right)\left(v_{kh}^m\right) + J_{1,v}'\left(w_{kh}^{m-1}\right)\left(v_{kh}^{m-1}\right)\right] \\
&\quad + \frac{k_m}{3}J_{1,v}'\left(\frac{1}{2}\left(w_{kh}^m + w_{kh}^{m-1}\right)\right)\left(i_{2k}^{(2)}v_{kh}^{m-\frac{1}{2}}\right), \\
\eta_{k,5}^{n,m} &= -\frac{k_m}{6}\left(\bar{u}_{kh}^m, v_{kh}^m + v_{kh}^{m-1}\right) + \frac{k_m}{3}\left(\bar{u}_{kh}^m, i_{2k}^{(2)}v_{kh}^{m-\frac{1}{2}}\right), \\
\eta_{k,6}^{n,m} &= \frac{k_m}{6}\left[a'\left(u_{kh}^m\right)\left(u_{kh}^m, \bar{v}_{kh}^m\right) + a'\left(u_{kh}^{m-1}\right)\left(u_{kh}^{m-1}, \bar{v}_{kh}^m\right)\right] \\
&\quad - \frac{k_m}{3}a'\left(\frac{1}{2}\left(u_{kh}^m + u_{kh}^{m-1}\right)\right)\left(i_{2k}^{(2)}u_{kh}^{m-\frac{1}{2}}, \bar{v}_{kh}^m\right),
\end{aligned}$$

where

$$u_{kh}^{m-\frac{1}{2}} = u_{kh}\left(t_{m-\frac{1}{2}}\right), \quad t_{m-\frac{1}{2}} = \frac{1}{2}\left(t_{m-1} + t_m\right),$$

is the evaluation of the specified function at the time instance $t_{m-\frac{1}{2}}$.

### A.3.2. Terms in the Interpolated Temporal Error Estimator.

In Section 1.3, the interpolated temporal error estimator

$$\begin{aligned}
\eta_k^{i,m} = \sum_{j=1}^{6}\eta_{k,j}^{i,m} = &-\frac{1}{2}A\left(w_{kh}\right)\left(i_{2h}^{(2)}\Pi_k^1 z_{kh}\right)_m \\
&+\frac{1}{2}J'\left(w_{kh}\right)\left(i_{2h}^{(2)}\Pi_{2k}^2 w_{kh}\right)_m \\
&-\frac{1}{2}A'\left(w_{kh}\right)\left(i_{2h}^{(2)}\Pi_{2k}^2 w_{kh}, z_{kh}\right)_m
\end{aligned}$$

has been derived for $m = 1, \ldots, M$. The single terms $\eta_{k,j}^{i,m}$ are

$$\begin{aligned}
\eta_{k,1}^{i,m} &:= -\frac{1}{2}\left(\left(\dot{u}_{kh} - v_{kh}, i_{2h}^{(2)}\left(i_k^{(1)}\bar{u}_{kh} - \bar{u}_{kh}\right)\right)\right)_m \\
&\quad -\frac{1}{2}\left(\left(\dot{v}_{kh}, i_{2h}^{(2)}\left(i_k^{(1)}\bar{v}_{kh} - \bar{v}_{kh}\right)\right)\right)_m, \\
\eta_{k,2}^{i,m} &:= -\frac{1}{2}\left(a\left(u_{kh}\right)\left(i_{2h}^{(2)}\left(i_k^{(1)}\bar{v}_{kh} - \bar{v}_{kh}\right)\right)\right)_m \\
&\quad +\frac{1}{2}\left(\left(f, i_{2h}^{(2)}\left(i_k^{(1)}\bar{v}_{kh} - \bar{v}_{kh}\right)\right)\right)_m
\end{aligned}$$

$$
\begin{aligned}
& && +\frac{1}{2}\left(\left(q, i_{2h}^{(2)}\left(i_{k}^{(1)}\bar{v}_{kh}-\bar{v}_{kh}\right)\right)_{\Gamma_N}\right)_m ,\\
\eta_{k,3}^{i,m} &:= && \frac{1}{2}\left(J_{1,u}'\left(w_{kh}\right)\left(i_{2h}^{(2)}\left(i_{2k}^{(2)}u_{kh}-u_{kh}\right)\right)\right)_m ,\\
\eta_{k,4}^{i,m} &:= && \frac{1}{2}\left(J_{1,v}'\left(w_{kh}\right)\left(i_{2h}^{(2)}\left(i_{2k}^{(2)}v_{kh}-v_{kh}\right)\right)\right)_m ,\\
\eta_{k,5}^{i,m} &:= && \frac{1}{2}\left(\left(\dot{\bar{v}}_{kh}+\bar{u}_{kh}, i_{2h}^{(2)}\left(i_{2k}^{(2)}v_{kh}-v_{kh}\right)\right)\right)_m\\
& && +\frac{1}{2}\left(\left[\bar{u}\right]_{m+1}, i_{2h}^{(2)}\left(i_{2k}^{(2)}u_{kh}-u_{kh}\right)\left(t_m\right)\right)\\
& && +\frac{1}{2}\left(\left(\dot{\bar{u}}_{kh}, i_{2h}^{(2)}\left(i_{2k}^{(2)}u_{kh}-u_{kh}\right)\right)\right)_m\\
& && +\frac{1}{2}\left(\left[\bar{v}\right]_{m+1}, i_{2h}^{(2)}\left(i_{2k}^{(2)}v_{kh}-v_{kh}\right)\left(t_m\right)\right),\\
\eta_{k,6}^{i,m} &:= && -\frac{1}{2}\left(a'\left(u_{kh}\right)\left(i_{2h}^{(2)}\left(i_{2k}^{(2)}u_{kh}-u_{kh}\right), \bar{v}_{kh}\right)\right)_m .
\end{aligned}
$$

Evaluation of the time integrals by appropriate quadrature rules leads to

$$
\begin{aligned}
\eta_{k,1}^{i,m} &= && \frac{1}{4}\left(u_{kh}^{m}-u_{kh}^{m-1}, i_{2h}^{(2)}\left(\bar{u}_{kh}^{m}-\bar{u}_{kh}^{m-1}\right)\right)\\
& && -\frac{k_m}{12}\left(v_{kh}^{m}+2v_{kh}^{m-1}, i_{2h}^{(2)}\left(\bar{u}_{kh}^{m}-\bar{u}_{kh}^{m-1}\right)\right)\\
& && +\frac{1}{4}\left(v_{kh}^{m}-v_{kh}^{m-1}, i_{2h}^{(2)}\left(\bar{v}_{kh}^{m}-\bar{v}_{kh}^{m-1}\right)\right),\\
\eta_{k,2}^{i,m} &= && -\frac{k_m}{12}a\left(u_{kh}^{m-1}\right)\left(i_{2h}^{(2)}\left(\bar{v}_{kh}^{m-1}-3\bar{v}_{kh}^{m}\right)\right)\\
& && +\frac{k_m}{6}a\left(u_{kh}^{m}\right)\left(i_{2h}^{(2)}\bar{v}_{kh}^{m}\right)\\
& && -\frac{k_m}{6}a\left(\frac{1}{2}\left(u_{kh}^{m}+u_{kh}^{m-1}\right)\right)\left(i_{2h}^{(2)}\left(\bar{v}_{kh}^{m}+\bar{v}_{kh}^{m-1}\right)\right)\\
& && +\frac{k_m}{12}\left[-2\left(f^{m}, i_{2h}^{(2)}\bar{v}_{kh}^{m}\right)+\left(f^{m-1}, i_{2h}^{(2)}\left(\bar{v}_{kh}^{m-1}-3\bar{v}_{kh}^{m}\right)\right)\right]\\
& && +\frac{k_m}{6}\left(f^{m-\frac{1}{2}}, i_{2h}^{(2)}\left(\bar{v}_{kh}^{m}+\bar{v}_{kh}^{m-1}\right)\right)\\
& && -\frac{k_m}{6}\left(q^{m}, i_{2h}^{(2)}\bar{v}_{kh}^{m}\right)_{\Gamma_N}\\
& && +\frac{k_m}{12}\left(q^{m-1}, i_{2h}^{(2)}\left(\bar{v}_{kh}^{m-1}-3\bar{v}_{kh}^{m}\right)_{\Gamma_N}\right)
\end{aligned}
$$

$$\begin{aligned}
& +\frac{k_m}{6}\left(q^{m-\frac{1}{2}}, i_{2h}^{(2)}\left(\bar{v}_{kh}^{m}+\bar{v}_{kh}^{m-1}\right)_{\Gamma_N}\right), \\
\eta_{k,3}^{i,m} &= -\frac{k_m}{6}\left[J_{1,u}'\left(w_{kh}^{m}\right)\left(i_{2h}^{(2)}u_{kh}^{m}\right)+J_{1,u}'\left(w_{kh}^{m-1}\right)\left(i_{2h}^{(2)}u_{kh}^{m-1}\right)\right] \\
&\quad +\frac{k_m}{3}J_{1,u}'\left(\frac{1}{2}\left(w_{kh}^{m}+w_{kh}^{m-1}\right)\right)\left(i_{2h}^{(2)}i_{2k}^{(2)}u_{kh}^{m-\frac{1}{2}}\right), \\
\eta_{k,4}^{i,m} &= -\frac{k_m}{6}\left[J_{1,v}'\left(w_{kh}^{m}\right)\left(i_{2h}^{(2)}v_{kh}^{m}\right)+J_{1,v}'\left(w_{kh}^{m-1}\right)\left(i_{2h}^{(2)}v_{kh}^{m-1}\right)\right] \\
&\quad +\frac{k_m}{3}J_{1,v}'\left(\frac{1}{2}\left(w_{kh}^{m}+w_{kh}^{m-1}\right)\right)\left(i_{2h}^{(2)}i_{2k}^{(2)}v_{kh}^{m-\frac{1}{2}}\right), \\
\eta_{k,5}^{i,m} &= -\frac{k_m}{6}\left(\bar{u}_{kh}^{m}, i_{2h}^{(2)}\left(v_{kh}^{m}+v_{kh}^{m-1}\right)\right) \\
&\quad +\frac{k_m}{3}\left(\bar{u}_{kh}^{m}, i_{2h}^{(2)}i_{2k}^{(2)}v_{kh}^{m-\frac{1}{2}}\right), \\
\eta_{k,6}^{i,m} &= \frac{k_m}{6}\left[a'\left(u_{kh}^{m}\right)\left(i_{2h}^{(2)}u_{kh}^{m},\bar{v}_{kh}^{m}\right)+a'\left(u_{kh}^{m-1}\right)\left(i_{2h}^{(2)}u_{kh}^{m-1},\bar{v}_{kh}^{m}\right)\right] \\
&\quad -\frac{k_m}{3}a'\left(\frac{1}{2}\left(u_{kh}^{m}+u_{kh}^{m-1}\right)\right)\left(i_{2h}^{(2)}i_{2k}^{(2)}u_{kh}^{m-\frac{1}{2}},\bar{v}_{kh}^{m}\right).
\end{aligned}$$

## A.4. Spatial Error Estimate

In this section, the concrete terms of the spatial error estimator are discussed. The spatial error estimator in the first time step is

$$\begin{aligned}
\eta_h^0 &= -\frac{1}{2}\left(u_{kh}^0-u_s, \Pi_{2h}^{(2)}\bar{u}_{kh}^0\right)-\frac{1}{2}\left(v_{kh}^0-v_s, \Pi_{2h}^{(2)}\bar{v}_{kh}^0\right) \\
&\quad +\frac{1}{2}\left(\bar{u}_{kh}^1-\bar{u}_{kh}^0, \Pi_{2h}^{(2)}u_{kh}^0\right)+\frac{1}{2}\left(\bar{v}_{kh}^1-\bar{v}_{kh}^0, \Pi_{2h}^{(2)}v_{kh}^0\right).
\end{aligned}$$

It represents the error in the initial conditions and in the jump terms of the dual problem.

**A.4.1. Terms in the Spatial Error Estimator.** The spatial error estimator for $0 < m < M$ is

$$\begin{aligned}
\eta_h^{n,m} \quad = \sum_{i=1}^{6} \eta_{h,i}^{n,m} = \quad & -\frac{1}{2} A\left(w_{kh}\right)\left(\Pi_{2h}^{(2)} z_{kh}\right)_m \\
& +\frac{1}{2} J'\left(w_{kh}\right)\left(\Pi_{2h}^{(2)} w_{kh}\right)_m \\
& -\frac{1}{2} A'\left(w_{kh}\right)\left(\Pi_{2h}^{(2)} w_{kh}, z_{kh}\right)_m .
\end{aligned}$$

The single $\eta_{h,i}^{n,m}$ are given by

$$\begin{aligned}
\eta_{h,1}^{n,m} \quad &:= \quad -\frac{1}{2}\left(\left(\dot{u}_{kh} - v_{kh}, \Pi_{2h}^{(2)} \bar{u}_{kh}\right)\right)_m - \frac{1}{2}\left(\left(\dot{v}_{kh}, \Pi_{2h}^{(2)} \bar{v}_{kh}\right)\right)_m , \\
\eta_{h,2a}^{n,m} \quad &:= \quad -\frac{1}{2}\left(a\left(u_{kh}\right)\left(\Pi_{2h}^{(2)} \bar{v}_{kh}\right)\right)_m , \\
\eta_{h,2b}^{n,m} \quad &:= \quad \frac{1}{2}\left(\left(f, \Pi_{2h}^{(2)} \bar{v}_{kh}\right)\right)_m + \frac{1}{2}\left(\left(q, \Pi_{2h}^{(2)} \bar{v}_{kh}\right)_{\Gamma_N}\right)_m , \\
\eta_{h,3}^{n,m} \quad &:= \quad \frac{1}{2}\left(J'_{1,u}\left(w_{kh}\right)\left(\Pi_{2h}^{(2)} u_{kh}\right)\right)_m , \\
\eta_{h,4}^{n,m} \quad &:= \quad \frac{1}{2}\left(J'_{1,v}\left(w_{kh}\right)\left(\Pi_{2h}^{(2)} v_{kh}\right)\right)_m , \\
\eta_{h,5}^{n,m} \quad &:= \quad \frac{1}{2}\left(\left(\dot{\bar{v}}_{kh} + \bar{u}_{kh}, \Pi_{2h}^{(2)} v_{kh}\right)\right)_m + \frac{1}{2}\left(\left(\dot{\bar{u}}_{kh}, \Pi_{2h}^{(2)} u_{kh}\right)\right)_m \\
& \quad\quad +\frac{1}{2}\left(\left[\bar{u}\right]_{m+1}, \Pi_{2h}^{(2)} u_{kh}\left(t_m\right)\right) + \frac{1}{2}\left(\left[\bar{v}\right]_{m+1}, \Pi_{2h}^{(2)} v_{kh}\left(t_m\right)\right) , \\
\eta_{h,6}^{n,m} \quad &:= \quad -\frac{1}{2}\left(a'\left(u_{kh}\right)\left(\Pi_{2h}^{(2)} u_{kh}, \bar{v}_{kh}\right)\right)_m .
\end{aligned}$$

Using suitable quadrature formulas, we obtain

$$\begin{aligned}
\eta_{h,1}^{n,m} \quad &= \quad -\frac{1}{2}\left(u_{kh}^m - u_{kh}^{m-1}, \Pi_{2h}^{(2)} \bar{u}_{kh}^m\right) + \frac{k_m}{4}\left(v_{kh}^m + v_{kh}^{m-1}, \Pi_{2h}^{(2)} \bar{u}_{kh}^m\right) \\
& \quad\quad -\frac{1}{2}\left(v_{kh}^m - v_{kh}^{m-1}, \Pi_{2h}^{(2)} \bar{v}_{kh}^m\right) , \\
\eta_{h,2a}^{n,m} \quad &= \quad -\frac{k_m}{4}\left[a\left(u_{kh}^{m-1}\right)\left(\Pi_{2h}^{(2)} \bar{v}_{kh}^m\right) + a\left(u_{kh}^m\right)\left(\Pi_{2h}^{(2)} \bar{v}_{kh}^m\right)\right] , \\
\eta_{h,2b}^{n,m} \quad &= \quad \frac{k_m}{4}\left[\left(f^{m-1}, \Pi_{2h}^{(2)} \bar{v}_{kh}^m\right) + \left(f^m, \Pi_{2h}^{(2)} \bar{v}_{kh}^m\right)\right]
\end{aligned}$$

$$
\begin{aligned}
&\quad +\frac{k_m}{4}\left[\left(q^{m-1},\Pi_{2h}^{(2)}\bar{v}_{kh}^m\right)_{\Gamma_N}+\left(q^m,\Pi_{2h}^{(2)}\bar{v}_{kh}^m\right)_{\Gamma_N}\right],\\
\eta_{h,3}^{n,m} &= \frac{k_m}{4}\left[J_{1,u}'\left(w_{kh}^{m-1}\right)\left(\Pi_{2h}^{(2)}u_{kh}^{m-1}\right)+J_{1,u}'\left(w_{kh}^m\right)\left(\Pi_{2h}^{(2)}u_{kh}^m\right)\right],\\
\eta_{h,4}^{n,m} &= \frac{k_m}{4}\left[J_{1,v}'\left(w_{kh}^{m-1}\right)\left(\Pi_{2h}^{(2)}v_{kh}^{m-1}\right)+J_{1,v}'\left(w_{kh}^m\right)\left(\Pi_{2h}^{(2)}v_{kh}^m\right)\right],\\
\eta_{h,5}^{n,m} &= \frac{1}{2}\left(\bar{u}_{kh}^{m+1}-\bar{u}_{kh}^m,\Pi_{2h}^{(2)}u_{kh}^m\right)+\frac{1}{2}\left(\bar{v}_{kh}^{m+1}-\bar{v}_{kh}^{m+1},\Pi_{2h}^{(2)}v_{kh}^m\right)\\
&\quad +\frac{k_m}{4}\left(\bar{u}_{kh}^m,\Pi_{2h}^{(2)}\left(v_{kh}^{m-1}+v_{kh}^m\right)\right),\\
\eta_{h,6}^{n,m} &= -\frac{k_m}{4}a'\left(u_{kh}^m\right)\left(\Pi_{2h}^{(2)}u_{kh}^m,\bar{v}_{kh}^m\right)\\
&\quad -\frac{k_m}{4}a'\left(u_{kh}^{m-1}\right)\left(\Pi_{2h}^{(2)}u_{kh}^{m-1},\bar{v}_{kh}^m\right).
\end{aligned}
$$

In the last time step, only the term $\eta_{h,5}^{n,M}$ has to be modified. It is of the form

$$
\begin{aligned}
\eta_{h,5}^M &= \frac{1}{2}J_{2,u}'\left(w_{kh}^M\right)\left(\Pi_{2h}^{(2)}u_{kh}^M\right)+\frac{1}{2}J_{2,v}'\left(w_{kh}^M\right)\left(\Pi_{2h}^{(2)}v_{kh}^M\right)\\
&\quad -\frac{1}{2}\left(\bar{u}_{kh}^M,\Pi_{2h}^{(2)}u_{kh}^M\right)-\frac{1}{2}\left(\bar{v}_{kh}^M,\Pi_{2h}^{(2)}v_{kh}^M\right)\\
&\quad +\frac{k_M}{4}\left(\bar{u}_{kh}^M,\Pi_{2h}^{(2)}\left(v_{kh}^M+v_{kh}^{M-1}\right)\right).
\end{aligned}
$$

### A.4.2. Terms in the Interpolated Spatial Error Estimator.

For $0 < m < M$, the interpolated spatial error estimator $\eta_h^{i,m}$ is given by

$$
\begin{aligned}
\eta_h^{i,m} = \sum_{j=1}^{6}\eta_{h,j}^{i,m} = &-\frac{1}{2}A\left(w_{kh}\right)\left(i_k^{(1)}\Pi_{2h}^{(2)}z_{kh}\right)_m\\
&+\frac{1}{2}J'\left(w_{kh}\right)\left(i_{2k}^{(2)}\Pi_{2h}^{(2)}w_{kh}\right)_m\\
&-\frac{1}{2}A'\left(w_{kh}\right)\left(i_{2k}^{(2)}\Pi_{2h}^{(2)}w_{kh},z_{kh}\right)_m.
\end{aligned}
$$

Here, $\eta_{h,j}^{i,m}$ are

$$
\eta_{h,1}^{i,m} := -\frac{1}{2}\left(\left(\dot{u}_{kh}-v_{kh},i_k^{(1)}\Pi_{2h}^{(2)}\bar{u}_{kh}\right)\right)_m
$$

$$
\begin{aligned}
&& -\frac{1}{2}\left(\left(\dot{v}_{kh}, i_k^{(1)}\Pi_{2h}^{(2)}\bar{v}_{kh}\right)\right)_m, \\
\eta_{h,2a}^{i,m} &:=& -\frac{1}{2}\left(a\left(u_{kh}\right)\left(i_k^{(1)}\Pi_{2h}^{(2)}\bar{v}_{kh}\right)\right)_m, \\
\eta_{h,2b}^{i,m} &:=& \frac{1}{2}\left(\left(f, i_k^{(1)}\Pi_{2h}^{(2)}\bar{v}_{kh}\right)\right)_m + \frac{1}{2}\left(\left(q, i_k^{(1)}\Pi_{2h}^{(2)}\bar{v}_{kh}\right)_{\Gamma_N}\right)_m, \\
\eta_{h,3}^{i,m} &:=& \frac{1}{2}\left(J_{1,u}'\left(w_{kh}\right)\left(i_{2k}^{(2)}\Pi_{2h}^{(2)}u_{kh}\right)\right)_m, \\
\eta_{h,4}^{i,m} &:=& \frac{1}{2}\left(J_{1,v}'\left(w_{kh}\right)\left(i_{2k}^{(2)}\Pi_{2h}^{(2)}v_{kh}\right)\right)_m, \\
\eta_{h,5}^{i,m} &:=& \frac{1}{2}\left(\left(\dot{\bar{v}}_{kh}+\bar{u}_{kh}, i_{2k}^{(2)}\Pi_{2h}^{(2)}v_{kh}\right)\right)_m + \frac{1}{2}\left(\left(\dot{\bar{u}}_{kh}, i_{2k}^{(2)}\Pi_{2h}^{(2)}u_{kh}\right)\right)_m \\
&& +\frac{1}{2}\left(\left[\bar{u}\right]_{m+1}, i_{2k}^{(2)}\Pi_{2h}^{(2)}u_{kh}\left(t_m\right)\right) \\
&& +\frac{1}{2}\left(\left[\bar{v}\right]_{m+1}, i_{2k}^{(2)}\Pi_{2h}^{(2)}v_{kh}\left(t_m\right)\right), \\
\eta_{h,6}^{i,m} &:=& -\frac{1}{2}\left(a'\left(u_{kh}\right)\left(i_{2k}^{(2)}\Pi_{2h}^{(2)}u_{kh}, \bar{v}_{kh}\right)\right)_m.
\end{aligned}
$$

Numerical integration leads to

$$
\begin{aligned}
\eta_{h,1}^{i,m} &=& -\frac{1}{4}\left(u_{kh}^m - u_{kh}^{m-1}, \Pi_{2h}^{(2)}\left(\bar{u}_{kh}^m + \bar{u}_{kh}^{m-1}\right)\right) \\
&& +\frac{k_m}{6}\left[\left(v_{kh}^{m-1}, \Pi_{2h}^{(2)}\bar{u}_{kh}^{m-1}\right) + \left(v_{kh}^m, \Pi_{2h}^{(2)}\bar{u}_{kh}^m\right)\right] \\
&& +\frac{k_m}{12}\left[\left(v_{kh}^m, \Pi_{2h}^{(2)}\bar{u}_{kh}^{m-1}\right) + \left(v_{kh}^{m-1}, \Pi_{2h}^{(2)}\bar{u}_{kh}^m\right)\right] \\
&& -\frac{1}{4}\left(v_{kh}^m - v_{kh}^{m-1}, \Pi_{2h}^{(2)}\left(\bar{v}_{kh}^m + \bar{v}_{kh}^{m-1}\right)\right), \\
\eta_{h,2a}^{i,m} &=& -\frac{k_m}{12}\left[a\left(u_{kh}^{m-1}\right)\left(\Pi_{2h}^{(2)}\bar{v}_{kh}^{m-1}\right) + a\left(u_{kh}^m\right)\left(\Pi_{2h}^{(2)}\bar{v}_{kh}^m\right)\right] \\
&& -\frac{k_m}{6}a\left(\frac{1}{2}\left(u_{kh}^m + u_{kh}^{m-1}\right)\right)\left(\Pi_{2h}^{(2)}\left(\bar{v}_{kh}^m + \bar{v}_{kh}^{m-1}\right)\right), \\
\eta_{h,2b}^{i,m} &=& \frac{k_m}{12}\left[\left(f^{m-1}, \Pi_{2h}^{(2)}\bar{v}_{kh}^{m-1}\right) + \left(f^m, \Pi_{2h}^{(2)}\bar{v}_{kh}^m\right)\right] \\
&& +\frac{k_m}{6}\left(f^{m-\frac{1}{2}}, \Pi_{2h}^{(2)}\left(\bar{v}_{kh}^m + \bar{v}_{kh}^{m-1}\right)\right) \\
&& \frac{k_m}{12}\left[\left(q^{m-1}, \Pi_{2h}^{(2)}\bar{v}_{kh}^{m-1}\right)_{\Gamma_N} + \left(q^m, \Pi_{2h}^{(2)}\bar{v}_{kh}^m\right)_{\Gamma_N}\right]
\end{aligned}
$$

$$
\begin{aligned}
& +\frac{k_m}{6}\left(q^{m-\frac{1}{2}}, \Pi_{2h}^{(2)}\left(\bar{v}_{kh}^m+\bar{v}_{kh}^{m-1}\right)\right)_{\Gamma_N}, \\
\eta_{h,3}^{i,m} = & \frac{k_m}{12}\left[J_{1,u}^{\prime}\left(w_{kh}^{m-1}\right)\left(\Pi_{2h}^{(2)} u_{kh}^{m-1}\right)+J_{1,u}^{\prime}\left(w_{kh}^m\right)\left(\Pi_{2h}^{(2)} u_{kh}^m\right)\right] \\
& +\frac{k_m}{3} J_{1,u}^{\prime}\left(\frac{1}{2}\left(w_{kh}^m+w_{kh}^{m-1}\right)\right)\left(i_{2k}^{(2)} \Pi_{2h}^{(2)} u_{kh}^{m-\frac{1}{2}}\right), \\
\eta_{h,4}^{i,m} = & \frac{k_m}{12}\left[J_{1,v}^{\prime}\left(w_{kh}^{m-1}\right)\left(\Pi_{2h}^{(2)} v_{kh}^{m-1}\right)+J_{1,v}^{\prime}\left(w_{kh}^m\right)\left(\Pi_{2h}^{(2)} v_{kh}^m\right)\right] \\
& +\frac{k_m}{3} J_{1,v}^{\prime}\left(\frac{1}{2}\left(w_{kh}^m+w_{kh}^{m-1}\right)\right)\left(i_{2k}^{(2)} \Pi_{2h}^{(2)} v_{kh}^{m-\frac{1}{2}}\right), \\
\eta_{h,5}^{i,m} = & \frac{1}{2}\left(\bar{u}_{kh}^{m+1}-\bar{u}_{kh}^m, \Pi_{2h}^{(2)} u_{kh}^m\right)+\frac{1}{2}\left(\bar{v}_{kh}^{m+1}-\bar{v}_{kh}^m, \Pi_{2h}^{(2)} v_{kh}^m\right) \\
& +\frac{k_m}{12}\left(\bar{u}_{kh}^m, \Pi_{2h}^{(2)}\left(v_{kh}^m+v_{kh}^m\right)\right) \\
& +\frac{k_m}{3}\left(\bar{u}_{kh}^m, i_{2k}^{(2)} \Pi_{2h}^{(2)} v_{kh}\left(t_{m-\frac{1}{2}}\right)\right), \\
\eta_{h,6}^{i,m} = & -\frac{k_m}{12}\left[a^{\prime}\left(u_{kh}^m\right)\left(\Pi_{2h}^{(2)} u_{kh}^m, \bar{v}_{kh}^m\right)+a^{\prime}\left(u_{kh}^{m-1}\right)\left(\Pi_{2h}^{(2)} u_{kh}^{m-1}, \bar{v}_{kh}^m\right)\right] \\
& -\frac{k_m}{3} a^{\prime}\left(\frac{1}{2}\left(u_{kh}^m+u_{kh}^{m-1}\right)\right)\left(i_{2k}^{(2)} \Pi_{2h}^{(2)} u_{kh}^{m-\frac{1}{2}}, \bar{v}_{kh}^m\right).
\end{aligned}
$$

We obtain

$$
\begin{aligned}
\eta_{h,5}^{i,M} = & \frac{1}{2} J_{2,u}^{\prime}\left(w_{kh}^M\right)\left(\Pi_{2h}^{(2)} u_{kh}^M\right)+\frac{1}{2} J_{2,v}^{\prime}\left(w_{kh}^M\right)\left(\Pi_{2h}^{(2)} v_{kh}^M\right) \\
& -\frac{1}{2}\left(\bar{u}_{kh}^M, \Pi_{2h}^{(2)} u_{kh}^M\right)-\frac{1}{2}\left(\bar{v}_{kh}^M, \Pi_{2h}^{(2)} v_{kh}^M\right) \\
& +\frac{k_M}{12}\left(\bar{u}_{kh}^M, \Pi_{2h}^{(2)}\left(v_{kh}^M+v_{kh}^{M-1}\right)\right) \\
& +\frac{k_M}{3}\left(\bar{u}_{kh}^M, i_{2k}^{(2)} \Pi_{2h}^{(2)} v_{kh}^{M-\frac{1}{2}}\right).
\end{aligned}
$$

in the last time step. All other terms have the same form as for

$$0 < m < M.$$

## A.5. Localised Spatial Error Estimator

The terms $\eta^m_{h,2a}$ and $\eta^m_{h,6}$ change for $m = 1, 2, \ldots, M$ compared to Section A.4, when the spatial error estimator is localised by integration by parts.

**A.5.1. Terms in the Spatial Error Estimator.** The term $\eta^{n,m}_{h,2}$ is written as

$$\eta^{n,m}_{h,2} = \tilde{\eta}^{n,m}_{h,2a} + \eta^{n,m}_{h,2b} + \eta^{n,m}_{h,2c}$$

with

$$\tilde{\eta}^{n,m}_{h,2a} = \frac{k_m}{4}\left[\left(\mathcal{A}\left(u^m_{kh}\right), \Pi^{(2)}_{2h}\bar{v}^m_{kh}\right) + \left(\mathcal{A}\left(u^m_{kh}\right), \Pi^{(2)}_{2h}\bar{v}^m_{kh}\right)\right],$$

and

$$\begin{aligned}
\eta^{n,m}_{h,2c} = & -\frac{k_m}{8}\sum_{\mathcal{T}\in\mathbb{T}^m_h}\sum_{E\in\partial\mathcal{T}}\left(\left[\mathcal{B}\left(u^{m-1}_{kh}\right)\right], \Pi^{(2)}_{2h}\bar{v}^m_{kh}\right)_E \\
& -\frac{k_m}{8}\sum_{\mathcal{T}\in\mathbb{T}^m_h}\sum_{E\in\partial\mathcal{T}}\left(\left[\mathcal{B}\left(u^{m}_{kh}\right)\right], \Pi^{(2)}_{2h}\bar{v}^m_{kh}\right)_E .
\end{aligned}$$

The term $\tilde{\eta}^{n,m}_{h,2a}$ represents the evaluation of the strong formulation and $\eta^{n,m}_{h,2c}$ the jump terms on the boundary of the mesh cells. Now, the term

$$\eta^{n,m}_{h,6} = \eta^{n,m}_{h,6a} + \eta^{n,m}_{h,6b}$$

is discussed. We end up with

$$\begin{aligned}
\eta^{n,m}_{h,6a} = & \frac{k_m}{4}\left(\mathcal{A}'\left(u^m_{kh}\right)\left(\bar{v}^m_{kh}\right), \Pi^{(2)}_{2h}u^m_{kh}\right) \\
& +\frac{k_m}{4}\left(\mathcal{A}'\left(u^{m-1}_{kh}\right)\left(\bar{v}^m_{kh}\right), \Pi^{(2)}_{2h}u^{m-1}_{kh}\right),
\end{aligned}$$

and

$$\begin{aligned}
\eta^{n,m}_{h,6b} = & -\frac{k_m}{8}\sum_{\mathcal{T}\in\mathbb{T}^m_h}\sum_{E\in\partial\mathcal{T}}\left(\left[\mathcal{B}'\left(u^{m-1}_{kh}\right)\left(\bar{v}^m_{kh}\right)\right], \Pi^{(2)}_{2h}u^{m-1}_{kh}\right)_E \\
& -\frac{k_m}{8}\sum_{\mathcal{T}\in\mathbb{T}^m_h}\sum_{E\in\partial\mathcal{T}}\left(\left[\mathcal{B}'\left(u^{m}_{kh}\right)\left(\bar{v}^m_{kh}\right)\right], \Pi^{(2)}_{2h}u^{m}_{kh}\right)_E .
\end{aligned}$$

**A.5.2. Terms in the Interpolated Spatial Error Estimator.** We write the term $\eta_{h,2}^{i,m}$ as

$$\eta_{h,2}^{i,m} = \tilde{\eta}_{h,2a}^{i,m} + \eta_{h,2b}^{i,m} + \eta_{h,2c}^{i,m}.$$

In detail, theses terms are given by

$$\begin{aligned}\tilde{\eta}_{h,2a}^{i,m} &= \frac{k_m}{12}\left[\left(\mathcal{A}\left(u_{kh}^{m-1}\right), \Pi_{2h}^{(2)}\bar{v}_{kh}^{m-1}\right) + \left(\mathcal{A}\left(u_{kh}^{m}\right), \Pi_{2h}^{(2)}\bar{v}_{kh}^{m}\right)\right] \\ &\quad + \frac{k_m}{6}\left(\mathcal{A}\left(\frac{1}{2}\left(u_{kh}^{m} + u_{kh}^{m-1}\right)\right), \Pi_{2h}^{(2)}\left(\bar{v}_{kh}^{m} + \bar{v}_{kh}^{m-1}\right)\right),\end{aligned}$$

and

$$\begin{aligned}\eta_{h,2c}^{i,m} &= -\frac{k_m}{24}\sum_{\mathcal{T}\in\mathbb{T}_h^m}\sum_{E\in\partial\mathcal{T}}\left(\left[\mathcal{B}\left(u_{kh}^{m-1}\right)\right], \Pi_{2h}^{(2)}\bar{v}_{kh}^{m-1}\right)_E \\ &\quad -\frac{k_m}{24}\sum_{\mathcal{T}\in\mathbb{T}_h^m}\sum_{E\in\partial\mathcal{T}}\left(\left[\mathcal{B}\left(u_{kh}^{m}\right)\right], \Pi_{2h}^{(2)}\bar{v}_{kh}^{m}\right)_E \\ &\quad -\frac{k_m}{12}\sum_{\mathcal{T}\in\mathbb{T}_h^m}\sum_{E\in\partial\mathcal{T}}\left(\left[\mathcal{B}\left(\frac{1}{2}\left(u_{kh}^{m} + u_{kh}^{m-1}\right)\right)\right], \Pi_{2h}^{(2)}\left(\bar{v}_{kh}^{m}\right)\right)_E \\ &\quad -\frac{k_m}{12}\sum_{\mathcal{T}\in\mathbb{T}_h^m}\sum_{E\in\partial\mathcal{T}}\left(\left[\mathcal{B}\left(\frac{1}{2}\left(u_{kh}^{m} + u_{kh}^{m-1}\right)\right)\right], \Pi_{2h}^{(2)}\left(\bar{v}_{kh}^{m-1}\right)\right)_E.\end{aligned}$$

The term $\eta_{h,6}^{i,m}$ is split up into

$$\eta_{h,6}^{i,m} = \eta_{h,6a}^{i,m} + \eta_{h,6b}^{i,m}$$

with

$$\begin{aligned}\eta_{h,6a}^{i,m} &= \frac{k_m}{12}\left(\mathcal{A}'\left(u_{kh}^{m}\right)\left(\bar{v}_{kh}^{m}\right), \Pi_{2h}^{(2)}u_{kh}^{m}\right) \\ &\quad + \frac{k_m}{12}\left(\mathcal{A}'\left(u_{kh}^{m-1}\right)\left(\bar{v}_{kh}^{m}\right), \Pi_{2h}^{(2)}u_{kh}^{m-1}\right) \\ &\quad + \frac{k_m}{3}\left(\mathcal{A}'\left(u_{kh}^{m-\frac{1}{2}}\right)\left(\bar{v}_{kh}^{m}\right), i_{2k}^{(2)}\Pi_{2h}^{(2)}u_{kh}^{m-\frac{1}{2}}\right),\end{aligned}$$

and

$$\eta_{h,6b}^{i,m} = -\frac{k_m}{24}\sum_{\mathcal{T}\in\mathbb{T}_h^m}\sum_{E\in\partial\mathcal{T}}\left(\left[\mathcal{B}'\left(u_{kh}^{m-1}\right)\left(\bar{v}_{kh}^{m}\right)\right], \Pi_{2h}^{(2)}u_{kh}^{m-1}\right)_E$$

$$-\frac{k_m}{24}\sum_{\mathcal{T}\in\mathbb{T}_h^m}\sum_{E\in\partial\mathcal{T}}\left(\left[\mathcal{B}'\left(u_{kh}^m\right)\left(\bar{v}_{kh}^m\right)\right],\Pi_{2h}^{(2)}u_{kh}^m\right)_E$$
$$-\frac{k_m}{6}\sum_{\mathcal{T}\in\mathbb{T}_h^m}\sum_{E\in\partial\mathcal{T}}\left(\left[\mathcal{B}'\left(u_{kh}^{m-\frac{1}{2}}\right)\left(\bar{v}_{kh}^m\right)\right],i_{2k}^{(2)}\Pi_{2h}^{(2)}u_{kh}^{m-\frac{1}{2}}\right)_E.$$

## A.6. Terms in the Error Estimator for Dynamic Contact

In this section, the terms in the error estimator for dynamic contact problems are evaluated. We begin with the terms for the temporal error:

**A.6.1. Temporal Error Estimator Terms.** The error estimator in time is given by

$$\eta_{c,k}^{n,m}=\frac{k_m}{3}\left(\lambda_{kH}^m-\lambda_{kH}^{m-1},\gamma_{|\Gamma_C}\left(i_{2h}^{(2)}\left(\bar{v}_{kh}^m+2\bar{v}_{kh}^{m-1}\right)\right)\right)_{\Gamma_C}.$$

It simplifies to

$$\bar{\eta}_{c,k}^{n,m}=\frac{k_m}{2}\left(\lambda_{kH}^m-\lambda_{kH}^{m-1},\gamma_{|\Gamma_C}\left(\bar{v}_{kh}^m\right)\right)_{\Gamma_C},$$

if we insert $z_{kh}$ instead of $i_{k,2h}^{(1,2)}z_{kh}$. The interpolated counterpart of $\eta_{c,k}^{n,m}$ is

$$\eta_{c,k}^{i,m}=\frac{k_m}{3}\left(i_H^{(1)}\left(\lambda_{kH}^m-\lambda_{kH}^{m-1}\right),\gamma_{|\Gamma_C}\left(i_{2h}^{(2)}\left(\bar{v}_{kh}^m+2\bar{v}_{kh}^{m-1}\right)\right)\right)_{\Gamma_C}.$$

The simplified form is given by

$$\bar{\eta}_{c,k}^{i,m}=\frac{k_m}{2}\left(i_H^{(1)}\left(\lambda_{kH}^m-\lambda_{kH}^{m-1}\right),\gamma_{|\Gamma_C}\left(\bar{v}_{kh}^m\right)\right)_{\Gamma_C}.$$

**A.6.2. Spatial Error Estimator Terms.** The spatial error estimator is

$$\eta_{c,H}^{n,m}=-\frac{k_m}{2}\left(i_H^{(1)}\lambda_{kH}^m-\lambda_{kH}^m,\gamma_{|\Gamma_C}\left(i_{2h}^{(2)}\left(\bar{v}_{kh}^m+\bar{v}_{kh}^{m-1}\right)\right)\right)_{\Gamma_C}.$$

If $z_{kh}$ is inserted instead of $i_{k,2h}^{(1,2)}z_{kh}$, we obtain

$$\bar{\eta}_{c,H}^{n,m}=-k_m\left(i_H^{(1)}\lambda_{kH}^m-\lambda_{kH}^m,\gamma_{|\Gamma_C}\left(\bar{v}_{kh}^m\right)\right)_{\Gamma_C}.$$

The interpolated spatial error estimator is given by

$$\begin{aligned}\eta_{c,H}^{i,m} \quad = \quad & -\frac{k_m}{6}\left(i_H^{(1)}\lambda_{kH}^{m-1} - \lambda_{kH}^{m-1}, \gamma_{|\Gamma_C}\left(2i_{2h}^{(2)}\bar{v}_{kh}^{m-1} + i_{2h}^{(2)}\bar{v}_{kh}^{m}\right)\right)_{\Gamma_C} \\ & -\frac{k_m}{6}\left(i_H^{(1)}\lambda_{kH}^{m} - \lambda_{kH}^{m}, \gamma_{|\Gamma_C}\left(2i_{2h}^{(2)}\bar{v}_{kh}^{m} + i_{2h}^{(2)}\bar{v}_{kh}^{m-1}\right)\right)_{\Gamma_C}.\end{aligned}$$

It has the simplified form

$$\bar{\eta}_{c,H}^{i,m} = -\frac{k_m}{2}\left(i_H^{(1)}\left(\lambda_{kH}^{m} + \lambda_{kH}^{m-1}\right) - \lambda_{kH}^{m} - \lambda_{kH}^{m-1}, \gamma_{|\Gamma_C}\left(\bar{v}_{kh}^{m}\right)\right)_{\Gamma_C}.$$