

# Accurate FE-Simulation of Three-Dimensional Microstructures

Dissertation zur Erlangung des Doktorgrades  
der Fakultät für Angewandte Wissenschaften  
der Albert-Ludwigs Universität Freiburg im Breisgau

Jens Müller

2001

Dekan: Prof. Dr. G. Urban

Referenten: Prof. Dr. J. G. Korvink, Prof. Dr. W. Menz

Datum: 20. September 2001

---

# CONTENTS

<b>Contents</b>	<b>3</b>
<b>Abstract</b>	<b>1</b>
<b>Zusammenfassung</b>	<b>3</b>
<b>1 Introduction</b>	<b>5</b>
1.1 Microsystems and Simulation	5
1.2 Accuracy control and adaptive methods	6
1.3 Major Results	7
<b>2 Modeling</b>	<b>11</b>
2.1 Continuum Mechanics	11
2.2 Thermomechanics	20
2.3 Electro-thermo-mechanics	23
<b>3 Accuracy Control for thin structure simulation</b>	<b>31</b>
3.1 Mathematical preliminaries	31
3.2 Modeling thin structures	54
3.3 Error Estimation for second order problems	71
3.4 Error Estimation for multi-layer thin structures	85
<b>4 Object Oriented Implementation</b>	<b>101</b>
4.1 Object Oriented (OO) Analysis and Design	101
4.2 Design of the simulator tool ADAPTREF	104
4.3 Mesh and Mesh Modification	107
4.4 Materials and Thin Structures	111
4.5 Fields and Boundary Conditions	113
4.6 Error Estimators and Mesh Refinement	115

---

---

<b>5</b>	<b>Simulation</b>	<b>123</b>
5.1	Thermal Analysis and typical singularities	123
5.2	Balancing error contributions	128
5.3	Mechanical Analysis	136
<b>6</b>	<b>Conclusion and Outlook</b>	<b>163</b>
	<b>Appendix</b>	<b>167</b>
A.1	Tensor operations	167
A.2	Tensor Reduction for Plates	168
	<b>References</b>	<b>173</b>
		<b>181</b>
	<b>Acknowledgements</b>	<b>183</b>
	<b>Curriculum Vitae</b>	<b>185</b>
	<b>Abbreviations and Symbols</b>	<b>187</b>
	<b>Index</b>	<b>191</b>

---

---

# ABSTRACT

This thesis presents the development and the application of a software tool ADAPTREF with which to automatically control the accuracy of simulation results obtained by finite element (FE) computations. It is especially designed to meet the requirements that are present when simulating micro-electro-mechanical systems (MEMS). The software tool aims to improve the solution accuracy of the computed fields from the coupled electrical, thermal and mechanical domains.

The coupled electro-thermo-mechanical fields obey (non-linear) partial differential equations and are derived by a thermodynamic formalism. By taking micro- and thus quantum statistics as a starting point one obtains, in the thermodynamic limit, the constitutive relations which together with the conservation equations yield the coupled electro-thermo-mechanics partial differential equations.

We then provide a review of functional analytical tools which allow us to reformulate the differential equations in a weak form. As its name says, the weak form is less restrictive with respect to the function space the solution is sought in and thus is amenable to a formulation that allows that the solution be sought by computing machines, which is done here using the finite element method. Besides a finite element formulation for simple thermal or electrostatic problems we give a more general presentation also comprising coupled thermo-electro-mechanical multi-layered plate problems, thereby demonstrating the extensibility of the concepts.

Even though computational resources have been boosted tremendously during the last years, one has to bear in mind that the complexity of the problems to be solved has increased, too. Therefore, dealing economically with the computational resources, and speaking in terms of memory requirements and computational time, we are bound to choose the number of computational nodes as small as possible. This can be achieved by the software tool ADAPTREF. It adaptively introduces new nodes into the computational finite element (FE) mesh which describes the device geometry. The adaptivity is carried out according to the requirement to

---

reduce the error of the computed solution. The main tool in doing so is error estimation with which we can localize the magnitude of the errors introduced by the discretization of the physical problem and which, for multi-physically active thin structures, is derived on the basis of a functional analytical framework. In addition, a refinement strategy and a geometrical split pattern are required in order to be able to implement a reliable and stable software tool with which to control the accuracy of the solution fields.

By choosing an Object-Oriented architecture and implementation (C++), we assure a flexible extensibility of the software module ADAPTREF into a great range of directions. By providing a proper interface it can be used with virtually any finite element tool, or more general, with any numerical tool that requires a mesh modification in order to improve the computed solutions.

Towards the end of the thesis, the power of the ADAPTREF is illustrated by simulations of a selected set of microsystems.

---

# ZUSAMMENFASSUNG

Die vorliegende Dissertation beschreibt die Entwicklung, die Implementierung und die Anwendungsmöglichkeiten eines Werkzeuges zur effizienten numerischen Simulation von Mikrosystemen. Basierend auf elementaren quantenstatistischen Beziehungen wird zunächst die Herleitung einer thermodynamischen Formulierung der Kontinuumsmechanik vorgenommen. Diese beinhaltet die für die Mikrosystemtechnik wichtigen physikalischen Effekte der Elektro-Thermomechanik, welche durch gekoppelte partielle, und im allgemeinen nichtlineare Differentialgleichungen beschrieben werden. Für die Geometrie des betrachteten Mikrosystems sind diese unter Berücksichtigung von Randbedingungen dann zu lösen, um Kenntnis der unterschiedlichen Feldgrößen zu erlangen.

Die Lösung dieser Differentialgleichungen gestaltet sich für Geometrien, wie sie für Mikrosysteme typisch sind, auf analytischem Wege schwierig und nicht selten aussichtslos. Deshalb greift man auf numerische Werkzeuge zurück, von welchen die Finiten-Element (FE) Programme die bedeutendsten sind. Nachdem ein für die Mikromechanik relevantes Plattenmodell vorgestellt wird, wird der Weg von den partiellen Differentialgleichungen zu einer allgemeinen Formulierung der Finite Element Methode auf funktionalanalytischer Basis skizziert. Die mittels der numerischen Werkzeuge erzielten Lösungen der Differentialgleichungen sind im allgemeinen umso besser, je größer die Anzahl der Rechenknoten gewählt wird, können aber, bedingt durch die diskrete Struktur der Rechenmethode, das Kontinuum nie erreichen.

Wenngleich die Rechnerkapazitäten in den vergangenen Jahren bedeutend gestiegen sind, so werden deren Grenzen bei der Simulation immer komplizierterer Gerätschaften dennoch immer wieder erreicht. Es ist deshalb unabdingbar, eine intelligente Software einzusetzen, welche das Rechengitter der berechneten Lösung anpasst und damit in einem weiteren Schritt deren Genauigkeit zu verbessern erlaubt. In dieser Arbeit haben wir ein sich in diesem Sinne selbst korrigierendes Software-Modul entwickelt, welches einen solchen Einsatz bei der Simulation von Mikrostrukturen erlaubt. Das Verfahren einer sich selbst korrigierenden Soft-

---

ware ist in diesem Falle auch unter dem Namen der adaptiven Finite Element Methoden bekannt. Bisher jedoch beschränkt sich deren Behandlung im wesentlichen auf theoretische Aspekte, in kommerziell erhältlichen Programmen sind, wenn überhaupt, nur äusserst reduzierte und oft nicht nachvollziehbare Methoden implementiert.

Wir geben, nach einer gründlichen mathematischen Vorbereitung, die Herleitung einer Fehlerschätzung für die multiphysikalischen, mehrlagigen Plattengleichungen, welche es erlaubt, auf der Grundlage der hierdurch lokalisierbaren Fehler, Strategien und geometrische Methoden einzusetzen, die eine lokale Modifikation des Rechengitters ermöglichen. Diese Methoden sind in dem hier entwickelten Werkzeug ADAPTREF zusammengefasst, einem Programm, dessen Einsatz zusammen mit einem Finite Element Werkzeug unter größt möglicher Schonung der Rechnerkapazitäten eine optimale Genauigkeit der zu berechnenden numerischen Lösung erlaubt. Bei der Entwicklung des Werkzeugs ADAPREF wurde eine Objekt-Orientierte Design-Methode und Programmiersprache, C++, für die Implementierung gewählt, um eine flexible Erweiterbarkeit zu gewährleisten. Schließlich werden die Möglichkeiten von ADAPREF mittels ausgewählter Simulationsbeispiele von Mikrosystemen illustriert.

# 1 INTRODUCTION

This thesis presents the development and the implementation of ADAPTREF, a multi-physics simulation tool with which to control and improve the results of solutions obtained from numerical microsystem analysis. It is primarily designed to be used in conjunction with a finite element tool which serves as the computational kernel and thus ADAPTREF acts as a driver program. Virtually any finite element tool could be used as soon as the interfaces between the two modules are well defined. We have used object-oriented technology to design and implement this flexible and extensible device simulation tool.

## 1.1 Microsystems and Simulation

Microsystems are a combination of packaged microelectronics and sensor and actuator transducers at the micrometer to millimeter scale. As for the case of IC development, circuit simulation tools have proven to facilitate the designer's work in many aspects. First, by making the design process more efficient, and, second, by making it possible to learn about the physical models underlying our assumptions in an organized manner that is "close" to reality [1]. The scope of circuit simulation, however, is limited to the electrical domain. When it comes to simulate microsystems in general, including sensors and actuators, multiple domains of physics come into play. Thermal, electrical, piezo-electrical, mechanical and fluidic effects must be considered, and also the coupling among domains has to be taken into account.

It turns out that these phenomena are well described by continuum models that appear in the form of coupled differential equations. Including this aspect, three fundamental issues must be addressed [2]:

- How are physical phenomena modeled using differential equations?
- What are the properties of solutions of differential equations?
- How are approximate solutions of differential equations computed and how can the accuracy of the approximations be controlled?

As far as the first point is concerned, a vast literature flourished [3],[4],[5],[6],[7],[8],[9],[10],[11],[12],[13]. Especially for thin structures, which are central to MEMS, many models have been presented, of which historically [14],[15] are the most significant. More recently, [16],[17],[18],[19],[20] and [22],[23],[24],[25],[26],[27],[28],[29] are only a few of the most important. Models also including analytical solutions for simple geometries are presented in [31],[32],[33],[34]. In general, all models presented in the literature up to now only cover the treatment of simple geometries or are restricted to models that only include a small part of the physically possible material crystal classes.

Properties of differential equations' solutions and their approximation are well established from the theoretical point of view for many phenomena and as well when it comes to the commercialization of simulation software tools [35],[36], the quality control of numerical solutions is still a question at issue in many aspects such that, to our knowledge no commercially available software exists that allows to control the accuracy of differential equations' solutions reliably.

## 1.2 Accuracy control and adaptive methods

Numerical computations require that the domain which is to be analyzed by means of partial differential equations and boundary conditions is given some discrete structure in order to be accessible to "discrete" computing machines. One usually assumes that the solution gets more accurate the denser the computational mesh covers the domain at issue. In fact, under certain preliminaries this is true, however, if the mesh would be refined uniformly the limits of the computational resources are reached very quickly, depending on the complexity of the problem to solve.

A major task therefore is to develop a method with which to gain control of the accuracy of the computed solutions by locally detecting errors and improving the accuracy at a minimum cost by only locally refining the mesh. It is often sufficient

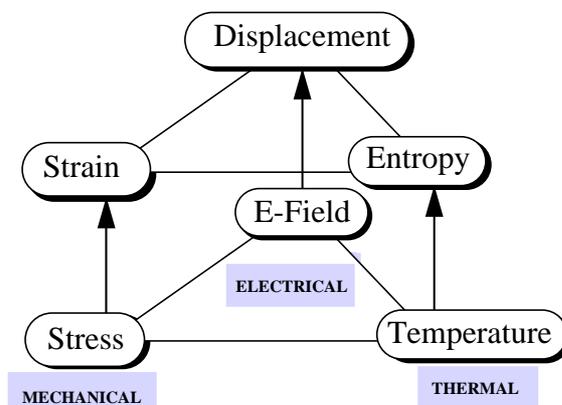
to take a rather coarse mesh as a starting point for the simulation cycle. Based on the pioneer works of [37],[39],[40],[41],[43],[43] several other works have been published treating a wide range of physical problems [44],[45],[46],[47] and [48],[49],[50],[51],[52],[53],[54],[55]. In general, for parabolic problems the theory is not as well developed as for elliptic problems, and for hyperbolic problems the field is still in its infancy [46].

For plate-like problems, one usually still refers to the classic Zienkiewicz paper [57], the Reissner-Mindlin approach to plate bending structures. For conforming Kirchhoff-Love plates however, only a basic, albeit mathematically sound, adaptive approach is given in [46], restricting the physical problem to the impractical case of single-layered, isotropic and uncoupled bending-only membranes. Only in recent years sound finite element models suited for multi-layered multi-physically active thin structures have been presented [58],[59],[60] on the basis of the Kirchhoff-Love theory, and accuracy control for these models has not been mentioned. A tool which incorporates an accuracy controlled treatment of such thin structure models therefore is highly desirable. ADAPTREF efficiently manages accuracy control of thin structure simulation and its performance is demonstrated in conjunction with the finite element software FEMEngine [61],[62],[63], an in-house simulation program.

## 1.3 Major Results

### THERMODYNAMICAL APPROACH TO ELECTRO-THERMO-MECHANICS

Based on equilibrium thermodynamics continuum relations are established out of a microscopic treatment which, by using a Lagrangian formulation, lead to the partial differential equations of electro-thermo-mechanics. This is done chapter 2.



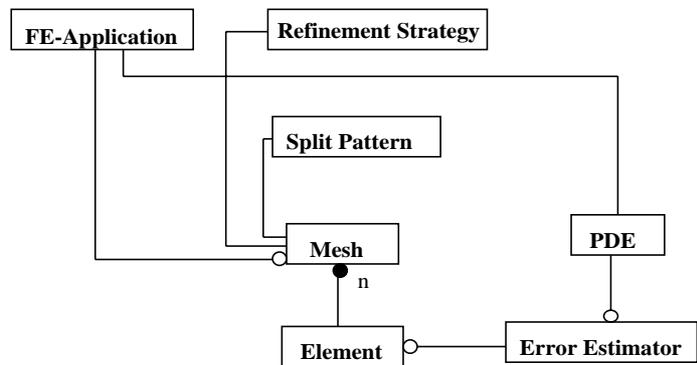
**ERROR ESTIMATION AND ACCURACY CONTROL**

On the basis of functional analytical methods we have developed error estimations for coupled multi-physically active multi-layered thin structures which are presented in chapter 3. The sources of error are separated into components that are useful from an engineering design viewpoint.

$$\|U - U_h\| \leq c \sqrt{\sum_{T \in S} (\eta_T^M)^2 + (\eta_T^P)^2}$$

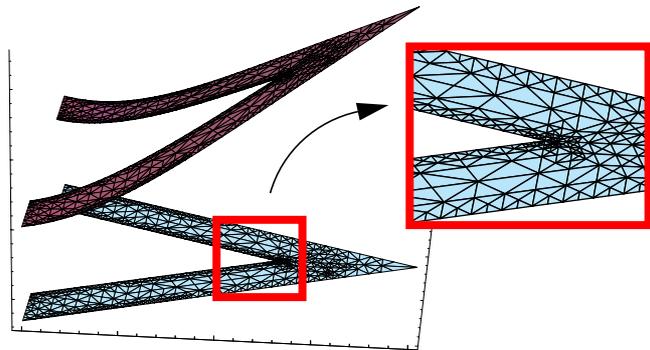
**IMPLEMENTATION**

We have implemented a novel software tool ADAPTREF that can be used in conjunction with virtually any finite element tool in order to control and improve the accuracy of the solutions of a wide range of physical problems by locally modifying the computational mesh and thus saving resources. The MEMS designer is thus relieved of the burden of finite element mesh design. Thanks to its Object-Oriented design it can be flexibly extended to cover even a broader range of problems. The architecture of ADAPTREF is presented in chapter 4.



## APPLICATIONS

The new simulation tool is applied to general structures which are typical for MEMS devices as well as to real-life microsystems. The examples show how, from a merely coarse grid, highly accurate solutions can be obtained, while keeping the computational effort at a minimum. The results of the simulations are presented in chapter 5.





## 2 MODELING

The objective of this chapter is to give a conceptual formulation of continuum mechanics as well as to illustrate the basic assumptions and models it is founded on. In recent years a vast literature has flourished illuminating continuum mechanics from broader and broader perspectives [3],[13],[28],[64]. While the work of Cauchy brought a temporary end to the mechanics of continuous solids in the middle of the nineteenth century, strong efforts in basic research have been made during and after the Second World War in order to meet the technical requirements at issue. With the immense growth of computer hardware resources during the last few years increasingly complex problems of continuum mechanics can be targeted.

### 2.1 Continuum Mechanics

Continuum mechanics is a phenomenological field theory. Mathematical models at a macroscopic level are established for the mechanical behaviour of matter based on experience. The behaviour includes effects introduced by extra-mechanical agencies such as electrical, thermal, or chemical resources. It is well known from physics that all matter has a discrete structure and its behaviour under external influences may be described by considering the interactions between atoms or molecules.

Quantum mechanics serves as a basis for phenomenological theories. In the context of many particle systems it is also known as quantum statistical mechanics [65],[110] taking into consideration the quantum behaviour of the small particles which we are dealing with. Utilizing this method one can derive a wide range of phenomena consistently which from a classical viewpoint do not seem to be connected, such as elasticity or magnetic effects. It is not in the scope of this work to give a consistent derivation of a complete set of phenomena. However, we will give an outline of the derivation of effects that are most important for the purpose of thin structure modeling. We start our exposition with a description of equilib-

rium [66]. Here, a quantum mechanical state, described by the density operator  $\rho$ , is conserved and thus  $d\rho/dt = 0$  or in other words, the Hamiltonian operator commutes with the density operator

$$[H, \rho] = 0. \quad (2.1)$$

If we virtually partition the system  $\Sigma_{12}$  spatially into two (almost) not interacting subsystems  $\Sigma_1$  and  $\Sigma_2$  we can expect independence and thus

$$\rho_{12} = \rho_1 \rho_2 \quad (2.2)$$

which is equivalent to

$$\log \rho_{12} = \log \rho_1 + \log \rho_2. \quad (2.3)$$

It follows that  $\log \rho$  is a linear function of the conserved quantities,

$$\log \rho = \text{constant} + \beta H,$$

$\beta$  denoting another constant. This is obvious in the light of the fact that extensive quantities such as energy and the number of particles sum up for separated systems. Using the *partition function*  $Z$  as a scaling factor in order to assure that  $\text{Trace}(\rho) = 1$ , we can rewrite the density matrix as

$$\rho = \frac{1}{Z} e^{-\beta H} \quad (2.4)$$

and thus

$$Z = \text{Trace} e^{-\beta H}. \quad (2.5)$$

The value of the parameter  $\beta$ , which can be shown to equal  $(k_B T)^{-1}$ ,  $k_B$  denoting the Boltzmann constant and  $T$  the absolute temperature, is fixed by the requirement that the expectation value of the Hamiltonian adopts a certain value:  $\langle H \rangle = E$ . The spectrum of the density operator  $\rho$  is purely discrete. The eigenstates are the states  $|E_i\rangle$  of  $H$  such that  $H|E_i\rangle = E_i|E_i\rangle$ . We only consider the case where the system is not extended infinitely but is spatially concentrated and thus

assuring its equilibrium. In turn, such systems possess discrete spectra. We therefore rewrite the density operator

$$\rho = \sum_i w_i |E_i\rangle\langle E_i| \quad (2.6)$$

where  $w_i$  is the occupation probability of the  $i$ -th state. Equivalently we have

$$\rho = \frac{1}{Z} \sum_i |E_i\rangle\langle E_i| e^{-\beta E_i} \quad (2.7)$$

with  $Z = \sum_i e^{-\beta E_i}$  and

$$w_i = \frac{1}{Z} e^{-\beta E_i}, \quad (2.8)$$

$e^{-\beta E_i}$  denoting the Boltzmann factor. The energy and the partition function are related by

$$E = -\frac{\partial}{\partial \beta} \ln Z. \quad (2.9)$$

We now introduce the concept of entropy  $S$  into the system by means of

$$S = -k_B \sum_i w_i \ln w_i. \quad (2.10)$$

It can be shown that entropy, a measure for the lack of knowledge of the micro state, increases monotonously with time and represents another extensive property of the system. With this concept of statistical mechanics the complete thermodynamics of a system can now be obtained. The expectation value of the energy  $E = \sum_i w_i E_i$  may be subject to the following variations:

- A change of the occupation probability
- A change  $dE_i$  of the energy eigenvalues  $E_i$ . This can be realized by varying external parameters  $X_\alpha$  in the Hamiltonian such as external magnetic fields or a change of the system's volume whereby the system's mechanism is changed.

The one-form of the energy eigenvalues towards equilibrium is then given by

$$dE_i = \sum_{\alpha} \frac{\partial E_i}{\partial X_{\alpha}} dX_{\alpha}, \quad (2.11)$$

such that the one-form of the energy reads

$$\begin{aligned} dE &= \sum_i dw_i E_i + \sum_i w_i dE_i \\ &= \sum_i dw_i E_i + \sum_i w_i \sum_{\alpha} \frac{\partial E_i}{\partial X_{\alpha}} dX_{\alpha} = \sum_i dw_i E_i + \sum_{\alpha} \xi_{\alpha} dX_{\alpha} \end{aligned} \quad (2.12)$$

where for abbreviation we have introduced

$$\xi_{\alpha} = \sum_i w_i \frac{\partial E_i}{\partial X_{\alpha}}. \quad (2.13)$$

The change of entropy in (2.10) is derived as

$$d\left(\frac{S}{k_B}\right) = -d\left(\sum_i w_i \ln w_i\right) = -\sum_i dw_i (\ln w_i + 1) = -\sum_i dw_i \ln w_i \quad (2.14)$$

since  $\sum_i dw_i$  vanishes identically. Introducing the distribution (2.8) we obtain

$$d\left(\frac{S}{k_B}\right) = \beta \sum_i dw_i E_i. \quad (2.15)$$

Recalling that  $\beta = 1/(k_B T)$  we end up with

$$dE = \frac{1}{k_B \beta} dS + \sum_{\alpha} \xi_{\alpha} dX_{\alpha} = T dS + \sum_{\alpha} \xi_{\alpha} dX_{\alpha} \quad (2.16)$$

which is also known as the Gibbs fundamental form. When incorporating the change of particle numbers one has to refer to the grand partition function as a starting point. The result is similar, only differing from the latter by the addition of the particle number and the chemical potential as conjugate pair of variables, the first being an extensive and the latter an intensive variable. As we will see later in this section, a set of additional thermodynamic parameters may be specified

such as the elastic strain or the electrical displacement. Once the partition function is known, a Gibbs function can be derived and thus the foundations for thermodynamics are laid [66]. Besides the internal energy  $E$ , auxiliary *thermodynamic potentials* might be defined such as the Helmholtz free energy

$$F = E - TS. \quad (2.17)$$

Recovering the energy can be achieved by means of a Legendre transformation

$$E = F - T \frac{\partial F}{\partial T}. \quad (2.18)$$

Dependencies between the different thermodynamic parameters are collected in a class of relations which are known as Maxwell relations. We point out that the first derivatives of the corresponding thermodynamic potentials lead to the corresponding energy conjugated system variables. In addition to the relations among system variables, one can also derive parameters which measure the response of a system variable to a change in a second system variable and thus are called response functions or susceptibilities [67]. As an example, we will demonstrate how to derive the most important response functions, namely the heat capacities  $C_\Lambda$ ,  $\Lambda$  denoting the quantity that is kept at a constant value when measuring  $C$ . When the system volume, the number of particles and the other variables considered are kept constant we can introduce

$$C_V = T \frac{\partial S(T, V, N, \Lambda_1 \dots)}{\partial T} = -T \frac{\partial^2 F(T, V, N, \Lambda_1 \dots)}{\partial T^2}, \quad (2.19)$$

the heat capacity at constant volume. Since for constant volume the Gibbs fundamental form reads

$$dE = TdS. \quad (2.20)$$

Therefore, the heat absorbed is identical to the change of energy and thus

$$C_V = \frac{\partial E(T, V, N, \Lambda_1 \dots)}{\partial T}. \quad (2.21)$$

By virtue of (2.9) the last identity can be recast into

$$C_V = -\frac{\partial}{\partial T} \frac{\partial}{\partial \beta} (\ln Z) = -\frac{\partial \beta}{\partial T} \frac{\partial}{\partial \beta} \frac{\partial}{\partial \beta} (\ln Z) = \frac{1}{kT^2} \frac{\partial^2}{\partial \beta^2} (\ln Z). \quad (2.22)$$

The fact that the susceptibilities only depend on the partition function, and thus on the hamiltonian of the system, becomes even more obvious when we rewrite equation (2.22) in the form

$$C_V = \frac{1}{kT^2} \left( \frac{1}{Z} \frac{\partial^2 Z}{\partial \beta^2} - \frac{1}{Z^2} \left( \frac{\partial Z}{\partial \beta} \right)^2 \right) = \frac{1}{kT^2} (\langle H^2 \rangle - \langle H \rangle^2) = \frac{1}{kT^2} \text{Var}(H). \quad (2.23)$$

We clearly see that the heat capacity depends linearly on the variance of the hamiltonian.

### ELASTIC ENERGY DENSITIES

Without referring to quantum mechanics one can derive an expression for the stored energy in a solid. We closely follow the exposition in [95]. In a crystal the ions are assumed to perform harmonic oscillations about an equilibrium position  $\mathbf{R}$  of a Bravais lattice site. The position of the ion, at any given time may be written as

$$\mathbf{r}(\mathbf{R}) = \mathbf{R} + \mathbf{u}(\mathbf{R}), \quad (2.24)$$

where  $\mathbf{u}(\mathbf{R})$  is the ion's deviation from equilibrium. We assume the amount of the deviations to be small compared to inter-ionic distances. We further consider internal forces to be present inside the lattice such that the overall potential field is composed of a set of pairpotentials, each being identical. In the case that each atom remained fixed at its Bravais lattice site this would read

$$U = \sum_{\substack{l, k = 1 \\ l < k}}^N V_{lk}(\mathbf{R}_l - \mathbf{R}_k) = \frac{1}{2} \sum_{l, k} V(\mathbf{R}_l - \mathbf{R}_k). \quad (2.25)$$

This expression, however, has to be replaced when we allow for a movement of the atoms to a general position  $\mathbf{r}(\mathbf{R}) \neq \mathbf{R}$  by

$$U = \frac{1}{2} \sum_{l,k} V(\mathbf{r}(\mathbf{R}_l) - \mathbf{r}(\mathbf{R}_k)) = \frac{1}{2} \sum_{l,k} V(\mathbf{R}_l - \mathbf{R}_k + \mathbf{u}(\mathbf{R}_l) - \mathbf{u}(\mathbf{R}_k)). \quad (2.26)$$

We further assume that the quantities  $\mathbf{u}(\mathbf{R}_l) - \mathbf{u}(\mathbf{R}_k)$  are small such that we can expand the potential into a Taylor series about the equilibrium position  $\mathbf{R}_l - \mathbf{R}_k$ , which for a general function  $v$  in multiple dimensions reads

$$v(\mathbf{x} + \mathbf{h}) = v(\mathbf{x}) + \nabla v(\mathbf{x})\mathbf{h} + \frac{1}{2}\mathbf{h}^T \mathbf{Hess}(v(\mathbf{x}))\mathbf{h} + O(\mathbf{h}^3), \quad (2.27)$$

where  $\mathbf{Hess}(v(\mathbf{x})) = \left( \frac{\partial^2 v}{\partial x_i \partial x_j} \right)_{ij}$  is the Hessian matrix of the field  $v$ . Applying this to (2.26) we have

$$\begin{aligned} U &= \frac{1}{2} \sum_{l,k} V(\mathbf{R}_l - \mathbf{R}_k + \mathbf{u}(\mathbf{R}_l) - \mathbf{u}(\mathbf{R}_k)) \\ &= \frac{1}{2} \sum_{l,k} V(\mathbf{R}_l - \mathbf{R}_k) + \frac{1}{4} \sum_{l,k} (\mathbf{u}(\mathbf{R}_l) - \mathbf{u}(\mathbf{R}_k))^T \mathbf{Hess}(V(\mathbf{R}_l - \mathbf{R}_k)) (\mathbf{u}(\mathbf{R}_l) - \mathbf{u}(\mathbf{R}_k)) \\ &\quad + O((\mathbf{u}(\mathbf{R}_l) - \mathbf{u}(\mathbf{R}_k))^3) \end{aligned} \quad (2.28)$$

since per definition of the equilibrium the potential is at a minimum and thus the gradient term vanishes. When envisaging the Lagrangian function which we will return to subsequently, it is obvious why this method also is denoted as a linearization about the equilibrium: the potential is bilinear in the displacement and thus the equations of motion form a linear system (the kinetic energy also is bilinear in the first time derivatives of the displacement such that the equations of motion in fact are linear. Moreover the constant term on the right hand side of (2.28) need not to be considered in the further analysis since it does not effect the equations of motion for the system. We therefore are left with the bilinear term which usually is referred to as the *harmonic potential*

$$U^{\text{harm}} = \frac{1}{4} \sum_{l,k} (\mathbf{u}(\mathbf{R}_l) - \mathbf{u}(\mathbf{R}_k))^T \mathbf{Hess}(V(\mathbf{R}_l - \mathbf{R}_k)) (\mathbf{u}(\mathbf{R}_l) - \mathbf{u}(\mathbf{R}_k)). \quad (2.29)$$

The harmonic potential as derived here also forms the basis for the concept of phonons describing the lattice dynamics from a microscopical and thus quantum mechanical point of view. As a next step we consider only displacements whose variations are small along  $\mathbf{R}_l - \mathbf{R}_k$  and which do not vary rapidly within the range of the pair potential, such that the displacement may be approximated linearly

$$\mathbf{u}(\mathbf{R}_l) = \mathbf{u}(\mathbf{R}_k) + D\mathbf{u}(\mathbf{r})|_{\mathbf{r}=\mathbf{R}_k}(\mathbf{R}_l - \mathbf{R}_k), \quad (2.30)$$

$D\mathbf{u}(\mathbf{r})|_{\mathbf{r}=\mathbf{R}_k}$  denoting the Jacobian matrix of  $\mathbf{u}$  at position  $\mathbf{r} = \mathbf{R}_k$ .

Inserting this into (2.29) we end up with

$$U^{\text{harm}} = \frac{1}{4} \sum_{l,k} (D\mathbf{u}(\mathbf{R}_k)(\mathbf{R}_l - \mathbf{R}_k))^T \text{Hess}(V(\mathbf{R}_l - \mathbf{R}_k))(D\mathbf{u}(\mathbf{R}_k)(\mathbf{R}_l - \mathbf{R}_k)) \quad (2.31)$$

and, after reordering the sum, we obtain the following expression for the harmonic potential

$$U^{\text{harm}} = \frac{1}{2} \sum_k (D\mathbf{u}(\mathbf{R}_k))^T : \left\{ \frac{1}{2} \sum_l (\mathbf{R}_l - \mathbf{R}_k)^T \text{Hess}(V(\mathbf{R}_l - \mathbf{R}_k)) (\mathbf{R}_l - \mathbf{R}_k) \right\} : D\mathbf{u}(\mathbf{R}_k), \quad (2.32)$$

where we have adopted tensor notation for the sake of readability, see A.1. The interior bracket forms a tensor of fourth rank  $\tilde{\mathbf{C}}(\mathbf{R}_k)$ , which, divided by the volume of the primitive lattice cell  $v_c$  is the elastic tensor  $\mathbf{C}$ . Since the displacements are slowly varying, the sum may be expressed as an integral

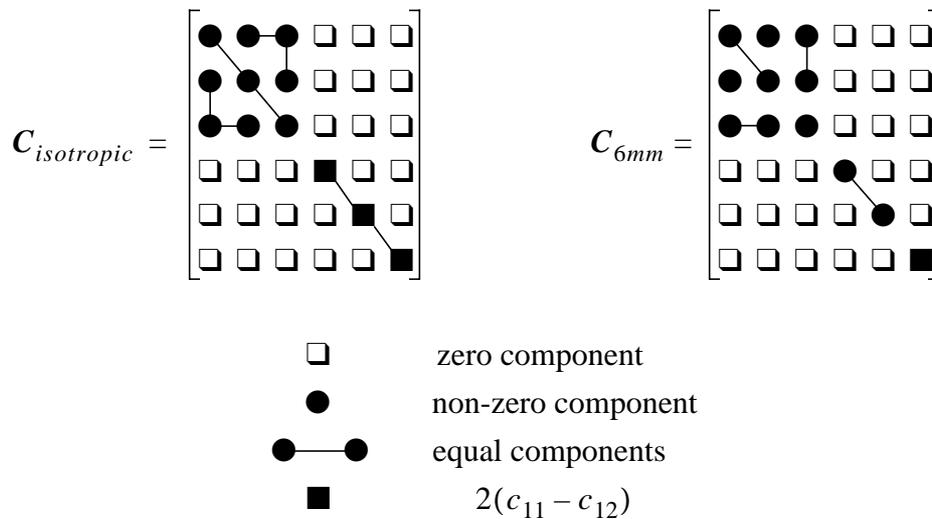
$$\begin{aligned} U^{\text{harm}} &= \frac{1}{2} \sum_{\mathbf{R}} \int (D\mathbf{u}(\mathbf{r}))^T : \mathbf{C}(\mathbf{R}) : D\mathbf{u}(\mathbf{r}) d\mathbf{r} \\ &= \frac{1}{2} \sum_{\mathbf{R}, \sigma\mu\tau\nu} \int \left( \frac{\partial}{\partial r_\sigma} u_\mu(\mathbf{r}) \right) C_{\sigma\mu\tau\nu} \left( \frac{\partial}{\partial r_\nu} u_\tau(\mathbf{r}) \right) d\mathbf{r}. \end{aligned} \quad (2.33)$$

The elastic tensor  $\mathbf{C}$  possesses a range of symmetries that mainly emerge from the fact that every Bravais lattice has inversion symmetry and since the harmonic potential is represented by a quadratic form which also is symmetric. Finally, the number of independent components can be reduced from 81 to 21. Further dependencies of the elastic constants turn up when taking into account crystal symme-

tries. These symmetries give rise to a convenient notation of the tensor components often seen in engineering literature and therefore is called engineering notation. This representation can be obtained by virtue of an index mapping where a pair of indexes is collapsed into one:

$$\begin{aligned}
 11 &\rightarrow 1 \\
 22 &\rightarrow 2 \\
 33 &\rightarrow 3 \\
 23, 32 &\rightarrow 4 \\
 31, 13 &\rightarrow 5 \\
 12, 21 &\rightarrow 6 .
 \end{aligned}
 \tag{2.34}$$

As an example, Figure 2.1. displays the symmetry relations among the tensor components for two of the crystal classes, the isotropic (cubic system) and the  $6mm$  class (hexagonal system, the crystal type of ZnO).



**Figure 2.1.** Symmetries of the elastic tensor exemplarily for two crystal classes, the isotropic and  $6mm$ . The latter describing the crystal properties of ZnO or PZT-4, materials used for piezoelectrically active MEMS. The full representation for all crystal classes can be found in [97].

In the isotropic case only two independent compounds are left, these are known as the Lamé constants  $\lambda$  and  $\mu$ , or, as a combination of these, the Young's modulus  $E$  and the Poisson number  $\nu$ , for which the relation

$$E = \frac{\mu(3\lambda + 2\mu)}{\lambda + \mu} \quad \nu = \frac{\lambda}{2(\lambda + \mu)} \quad (2.35)$$

holds.

## 2.2 Thermomechanics

With the concept of thermodynamic state variables introduced earlier in this chapter we derive the thermomechanics equations of motion using a Lagrangian field formalism for thermodynamic potentials and the associated state variables [58]. Having defined the second order symmetric strain tensor  $\boldsymbol{\varepsilon}$  as

$$\boldsymbol{\varepsilon} = (\nabla \mathbf{u})^S + \frac{1}{2} \nabla \mathbf{u} \cdot (\nabla \mathbf{u})^T \quad (2.36)$$

being a non-linear function of  $\nabla \mathbf{u}$ , the Helmholtz free energy density  $\hat{F}$  depends on the strain and on the temperature and the following Gibbs relations hold [68]

$$\begin{aligned} d\hat{E} &= Td\hat{S} + \boldsymbol{\sigma}:d\boldsymbol{\varepsilon} \\ d\hat{F} &= -\hat{S}dT + \boldsymbol{\sigma}:d\boldsymbol{\varepsilon} \end{aligned} \quad (2.37)$$

where we have introduced  $\boldsymbol{\sigma}$  as the energy conjugate state variable to the strain and thus subject to the relation

$$\boldsymbol{\sigma} = \frac{\partial \hat{F}}{\partial \boldsymbol{\varepsilon}}. \quad (2.38)$$

The state variables here appear in terms of densities for the same reason as we have introduced the continuum limit into the harmonic potential for crystal lattices earlier in this chapter. The identification scheme relating classical mechanical methods and thermodynamics is well known since [69], and was resurrected in [70], [71], or [58]. The analogy then suggests to identify the Lagrangian density  $\hat{L}$  with the negative of the free energy density plus the kinetic contribution. Recovering the internal energy density and the other potentials then of course can be achieved by means of a Legendre transform as it is the case when interchanging

between Hamiltonian and Lagrangian densities. We thus can write,  $\rho$  denoting the mass density,

$$\hat{L}(\mathbf{q}, \dot{\mathbf{q}}, \nabla \mathbf{q}) = \frac{1}{2} \rho \dot{\mathbf{u}}^2 - \hat{F}(\boldsymbol{\varepsilon}, T), \quad (2.39)$$

having merged the time primitive  $\theta$  of the temperature and the displacement field together into the vector  $\mathbf{q} = (\mathbf{u}, \theta)$ . Hamilton's principle then states that the motion of the system from time  $t_1$  to  $t_2$  is such that the integral

$$I = \int_{t_1}^{t_2} L dt, \quad (2.40)$$

where  $L$  is the Lagrange function associated to the Lagrange density, has a stationary value for the correct path of the motion [72]. In other words, the variation of the line integral has to vanish:

$$\delta \int_{t_1}^{t_2} L dt = \delta \int_{t_1}^{t_2} \int_{\Omega} \hat{L} dx dt = 0. \quad (2.41)$$

Carrying out the variational derivative for (2.41) yields the Euler-Lagrange equations

$$\frac{d}{dt} \left( \frac{\partial \hat{L}}{\partial \dot{\mathbf{q}}} \right) + \nabla \cdot \left( \frac{\partial \hat{L}}{\partial \nabla \mathbf{q}} \right) - \frac{\partial \hat{L}}{\partial \mathbf{q}} = 0 \quad (2.42)$$

which applied to the Lagrangian density results in a system of PDEs

$$\begin{cases} \rho \ddot{\mathbf{u}} - \nabla \cdot \left( \frac{\partial \hat{F}}{\partial \nabla \mathbf{u}} \right) = 0 \\ \dot{\hat{S}} = 0. \end{cases} \quad (2.43)$$

The second equation expresses the fact that the entropy density is conserved, the process is isentropic, no irreversible processes have been considered yet. The first

equation gives evidence of conservation of momentum and can be further resolved by using the identity

$$\left( \frac{\partial \hat{F}}{\partial \nabla \mathbf{u}} \right)_{ij} = \frac{\partial \hat{F}}{\partial \varepsilon_{kl}} \frac{\partial \varepsilon_{kl}}{\partial u_{i,j}} = [\boldsymbol{\sigma} \cdot (1 + \nabla \mathbf{u})]_{ij} \quad (2.44)$$

such that the first part of (2.43) reads

$$\rho \ddot{\mathbf{u}} - \nabla \cdot [\boldsymbol{\sigma} \cdot (1 + \nabla \mathbf{u})] = 0, \quad (2.45)$$

which is a non-linear equation with respect to the displacement fields. An equivalent expression for the entropy density change can be obtained by taking its total time derivative

$$\dot{\hat{S}}(\boldsymbol{\varepsilon}, T) = \frac{\partial \hat{S}}{\partial T} \dot{T} + \frac{\partial \hat{S}}{\partial \boldsymbol{\varepsilon}} : \dot{\boldsymbol{\varepsilon}} = \frac{\partial \hat{S}}{\partial T} \dot{T} - \frac{\partial^2 \hat{F}}{\partial T \partial \boldsymbol{\varepsilon}} : \dot{\boldsymbol{\varepsilon}} = \frac{\partial \hat{S}}{\partial T} \dot{T} - \frac{\partial \boldsymbol{\sigma}}{\partial T} : \dot{\boldsymbol{\varepsilon}} \quad (2.46)$$

where we have used that second derivatives are interchangeable. The first addend can be further simplified by using a thermodynamic susceptibility, the heat capacity, here, the heat capacity at constant strain derived by virtue of (2.18)

$$C_{\boldsymbol{\varepsilon}} = \left. \frac{\partial \hat{E}}{\partial T} \right|_{\boldsymbol{\varepsilon}} = \frac{\partial \hat{F}}{\partial T} + \hat{S} + T \frac{\partial \hat{S}}{\partial T} = T \frac{\partial \hat{S}}{\partial T} \quad (2.47)$$

and (2.37). Equation (2.46) then reads

$$\dot{\hat{S}}(\boldsymbol{\varepsilon}, T) = \frac{C_{\boldsymbol{\varepsilon}}}{T} \dot{T} - \frac{\partial \boldsymbol{\sigma}}{\partial T} : \dot{\boldsymbol{\varepsilon}}. \quad (2.48)$$

When considering heat conduction, which is an irreversible process, entropy is produced according to Fourier's law

$$\dot{\hat{S}} = \frac{\nabla \cdot (\boldsymbol{\kappa} \cdot \nabla T)}{T}, \quad (2.49)$$

$\kappa$  denoting the heat conductivity tensor. The entropy generation rate is due to an irreversible flux of heat from a hotter part of the solid to a colder part. Thus, in the presence of heat conduction, the second equation of (2.43) becomes

$$C_\varepsilon \dot{T} - T \frac{\partial \sigma}{\partial T} : \dot{\varepsilon} - \nabla \cdot (\kappa \cdot \nabla T) = 0. \quad (2.50)$$

In the absence of material compression or extension the second addend vanishes and we are left with the well known heat conduction equation. Moreover, in case some externally driven heat generation is present within the solid, equation (2.50) reads

$$C_\varepsilon \dot{T} - T \frac{\partial \sigma}{\partial T} : \dot{\varepsilon} - \nabla \cdot (\kappa \cdot \nabla T) = f \quad (2.51)$$

where  $f$  is the density of the local heat source. In general, many effects can be explained when considering the time dependent (and thus non-equilibrium) behaviour of thermodynamical systems as it was done here. Introducing generalized forces and fluxes by means of transport equations one can gain knowledge of various transport phenomena of which we only mention the Peltier or the Seebeck effect [73].

## 2.3 Electro-thermo-mechanics

Going back to thermodynamically irreversible changes of states we may introduce additional properties of solids and their connections to those previously introduced. That is to say, we discuss properties that may be measured with the crystal in equilibrium with its surroundings, so that neither the state of the crystal nor that of its surroundings changes with time [97]. The quantities we shall be concerned with are the temperature  $T$ , the electrical field  $\mathbf{E}$  and the mechanical stress  $\sigma$ . The relations between these properties is sketched in Figure 2.2. A more specific information about the coupling between the fields is given in Figure 2.3., there, the field's names are replaced by their symbols. The free energy density  $\hat{F}$  then depends on the temperature, the strain and the electrical field [74]

$$\hat{F} = \hat{F}(T, \boldsymbol{\varepsilon}, \mathbf{E}). \quad (2.52)$$

The differential form for  $\hat{F}$  then reads

$$d\hat{F} = -\hat{S}dT - \mathbf{D} \cdot d\mathbf{E} + \boldsymbol{\sigma} : d\boldsymbol{\varepsilon}, \quad (2.53)$$

where  $\mathbf{D}$  is the electrical displacement. The total linear differential of the independent variables  $(\boldsymbol{\varepsilon}, \mathbf{E}, T)$  can be expressed as [97]

$$\left\{ \begin{array}{l} \begin{array}{lll} \text{elasticity} & \text{inverse} & \text{thermal} \\ & \text{piezoelectricity} & \text{expansion} \end{array} \\ d\boldsymbol{\sigma}_{ij} = \left( \frac{\partial \boldsymbol{\sigma}_{ij}}{\partial \boldsymbol{\varepsilon}_{kl}} \right)_{\mathbf{E}, T} d\boldsymbol{\varepsilon}_{kl} + \left( \frac{\partial \boldsymbol{\sigma}_{ij}}{\partial E_k} \right)_{\boldsymbol{\varepsilon}, T} dE_k + \left( \frac{\partial \boldsymbol{\sigma}_{ij}}{\partial T} \right)_{\boldsymbol{\varepsilon}, \mathbf{E}} dT \\ \\ \begin{array}{lll} \text{direct} & & \\ \text{piezoelectricity} & \text{permittivity} & \text{pyroelectricity} \end{array} \\ d\mathbf{D}_i = \left( \frac{\partial D_i}{\partial \boldsymbol{\varepsilon}_{jk}} \right)_{\mathbf{E}, T} d\boldsymbol{\varepsilon}_{jk} + \left( \frac{\partial D_i}{\partial E_j} \right)_{\boldsymbol{\varepsilon}, T} dE_j + \left( \frac{\partial D_i}{\partial T} \right)_{\boldsymbol{\varepsilon}, \mathbf{E}} dT \\ \\ \begin{array}{lll} \text{piezocaloric} & \text{electrocaloric} & \text{heat} \\ \text{effect} & \text{effect} & \text{capacity} \end{array} \\ d\hat{S} = \left( \frac{\partial \hat{S}}{\partial \boldsymbol{\varepsilon}_{ij}} \right)_{\mathbf{E}, T} d\boldsymbol{\varepsilon}_{ij} + \left( \frac{\partial \hat{S}}{\partial E_i} \right)_{\boldsymbol{\varepsilon}, T} dE_i + \left( \frac{\partial \hat{S}}{\partial T} \right)_{\boldsymbol{\varepsilon}, \mathbf{E}} dT . \end{array} \right. \quad (2.54)$$

This is a total of 13 equations, each index ranging from one to three. Each of the differential coefficients represents the dependence of the thermodynamic state variables and therefore describes a physical effect. Including higher order derivatives than the linear ones would give rise to additional effects [74], such as electro-optical effects (non-linear optics), piezo-optical effects such as electrostriction or even elastic moduli of higher order. The coefficients on the leading diagonal of the equations (2.54) measure the principal effects while the others measure the coupled effects. When deriving symmetries of the coefficients representing the various effects we refer to equation (2.53) which can be rewritten as

$$d\hat{F} = \left( \frac{\partial \hat{F}}{\partial T} \right)_{\boldsymbol{\varepsilon}, \mathbf{E}} dT + \left( \frac{\partial \hat{F}}{\partial E_i} \right)_{\boldsymbol{\varepsilon}, T} dE_i + \left( \frac{\partial \hat{F}}{\partial \boldsymbol{\varepsilon}_{ij}} \right)_{\mathbf{E}, T} d\boldsymbol{\varepsilon}_{ij}. \quad (2.55)$$

Comparing the coefficients in both equations (2.53) and (2.55) yields

$$\left(\frac{\partial \hat{F}}{\partial T}\right)_{\boldsymbol{\varepsilon}, \mathbf{E}} = -\hat{S} \quad \left(\frac{\partial \hat{F}}{\partial E_i}\right)_{\boldsymbol{\varepsilon}, T} = -D_i \quad \left(\frac{\partial \hat{F}}{\partial \varepsilon_{ij}}\right)_{\mathbf{E}, T} = \sigma_{ij}. \quad (2.56)$$

Exploiting the second derivatives of the free energy density provides the following Maxwell relations

$$\left(\frac{\partial^2 \hat{F}}{\partial \varepsilon_{ij} \partial E_k}\right)_T = \left(\frac{\partial \sigma_{ij}}{\partial E_k}\right)_{\boldsymbol{\varepsilon}, T} = -\left(\frac{\partial D_k}{\partial \varepsilon_{ij}}\right)_{\mathbf{E}, T} = \pi_{kij} \quad (2.57)$$

and similarly

$$\left(\frac{\partial^2 \hat{F}}{\partial \varepsilon_{ij} \partial T}\right)_{\mathbf{E}} = \left(\frac{\partial \sigma_{ij}}{\partial T}\right)_{\boldsymbol{\varepsilon}, \mathbf{E}} = -\left(\frac{\partial \hat{S}}{\partial \varepsilon_{ij}}\right)_{\mathbf{E}, T} = \alpha_{ij} \quad (2.58)$$

and

$$\left(\frac{\partial^2 \hat{F}}{\partial E_i \partial T}\right)_{\boldsymbol{\varepsilon}} = -\left(\frac{\partial D_i}{\partial T}\right)_{\boldsymbol{\varepsilon}, \mathbf{E}} = -\left(\frac{\partial \hat{S}}{\partial E_i}\right)_{\boldsymbol{\varepsilon}, T} = -p_i. \quad (2.59)$$

When comparing the last three sets of equations, (2.57)-(2.59), we state that the matrix defined by the right hand side of the system (2.54) is symmetric. In detail, the following results have been established:

- the absolute values of  $\boldsymbol{\pi}$  for the inverse and the direct piezoelectric effect are numerically equal
- the coefficients  $\boldsymbol{\alpha}$  for the piezocaloric effect are the same as those for the thermal expansion except for the sign; the thermal expansion coefficient in this case refers to the thermal stress, the same symbol (here denoted as  $\mathbf{a}$ ) sometimes is used for the thermal expansion without directly relating it to the thermal stress in the way  $\boldsymbol{\alpha} = \mathbf{C} : \mathbf{a}$
- the coefficients  $\mathbf{p}$  describing the pyroelectric effect are identical with those describing the electrocaloric effect

The constitutive relations now can be displayed as the integrated form of (2.54)

$$\begin{aligned}
 \sigma_{ij} &= C_{ijkl}\varepsilon_{kl} - \pi_{kij}E_k + \alpha_{ij}\Delta T \\
 D_i &= \pi_{ijk}\varepsilon_{jk} + \chi_{ij}E_j + p_i\Delta T \\
 \Delta\hat{S} &= -\alpha_{ij}\varepsilon_{ij} + p_iE_i + (C_\sigma/T)\Delta T.
 \end{aligned} \tag{2.60}$$

where we haven taken into account that changes of states and thus of the variables representing them are dependent on the temperature only in case a temperature change occurs.

We would like to point out that the values of the susceptibilities in general could be derived beginning with a microscopic view of the world. As we saw earlier in this section, an equilibrium state is fully determined as soon as the partition function is deduced. In other words, the knowledge of the hamiltonian operator is sufficient for the complete description of the macro physical equilibrium state. However, evaluation of the partition function is possible only for the most simple cases and usually one has to rely on approximations. As an example, the coefficients of thermal expansion  $\alpha$  are due to an auxiliary anharmonic term in the potential of the Hamiltonian describing the ionic interaction energy in a solid.

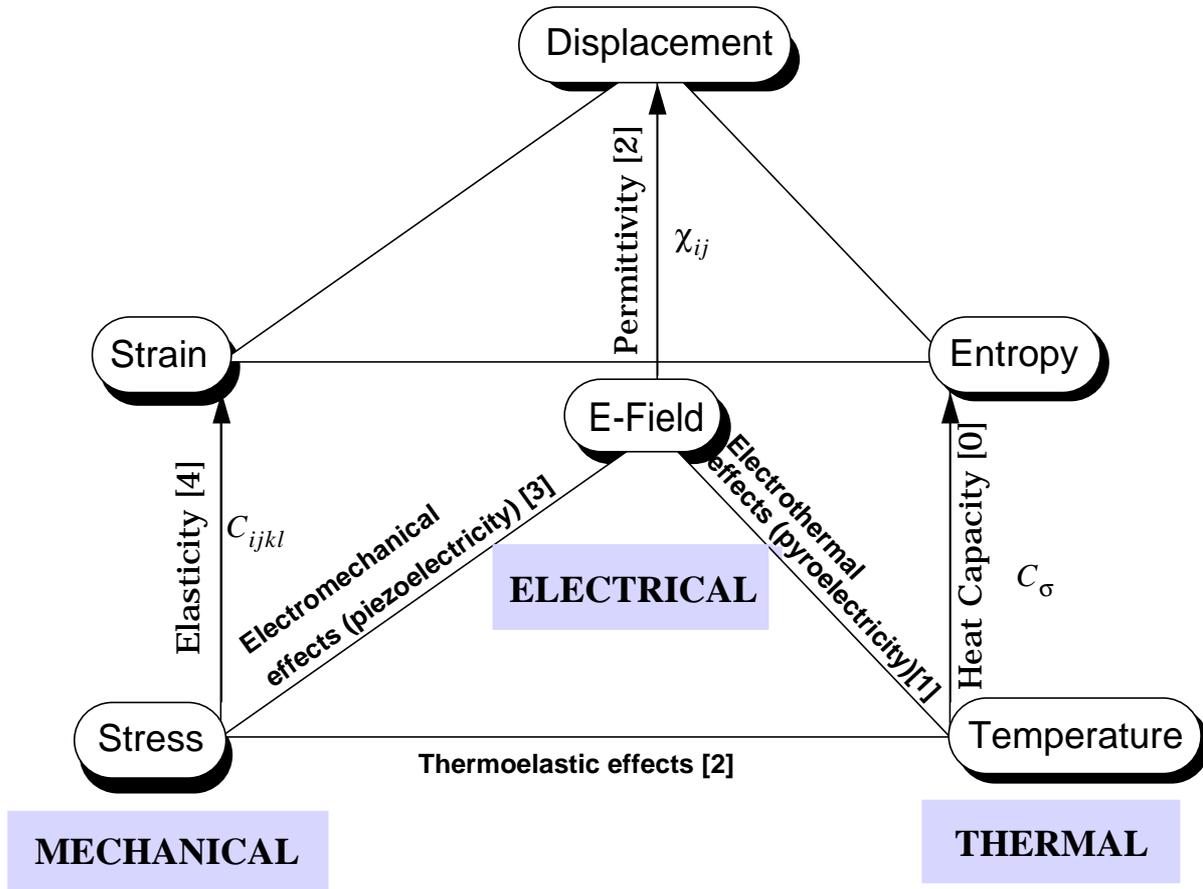
### DIFFERENTIAL EQUATIONS FOR ELECTRO-THERMOMECHANICAL SOLIDS

With the thermodynamical framework developed up to now, we are able to extend the equations of motion derived in (2.43). Therefore we introduce the stress of (2.60) into the first part of (2.43) in order to obtain

$$\rho\ddot{\mathbf{u}} - \nabla \cdot [\mathbf{C} : \boldsymbol{\varepsilon} - \boldsymbol{\pi}^T \cdot \mathbf{E} + \boldsymbol{\alpha}\Delta T \cdot (1 + \nabla\mathbf{u})] = \mathbf{f} \tag{2.61}$$

where  $\mathbf{f}$  is a three dimensional source function, in the engineering literature often referred to as a volume force acting within a solid and caused by, for example, a gravitational field. Introducing the relation for the strain, (2.36), and the fact that the electric field can be expressed as the negative gradient of a scalar electric potential, the equation of motion can be recast into

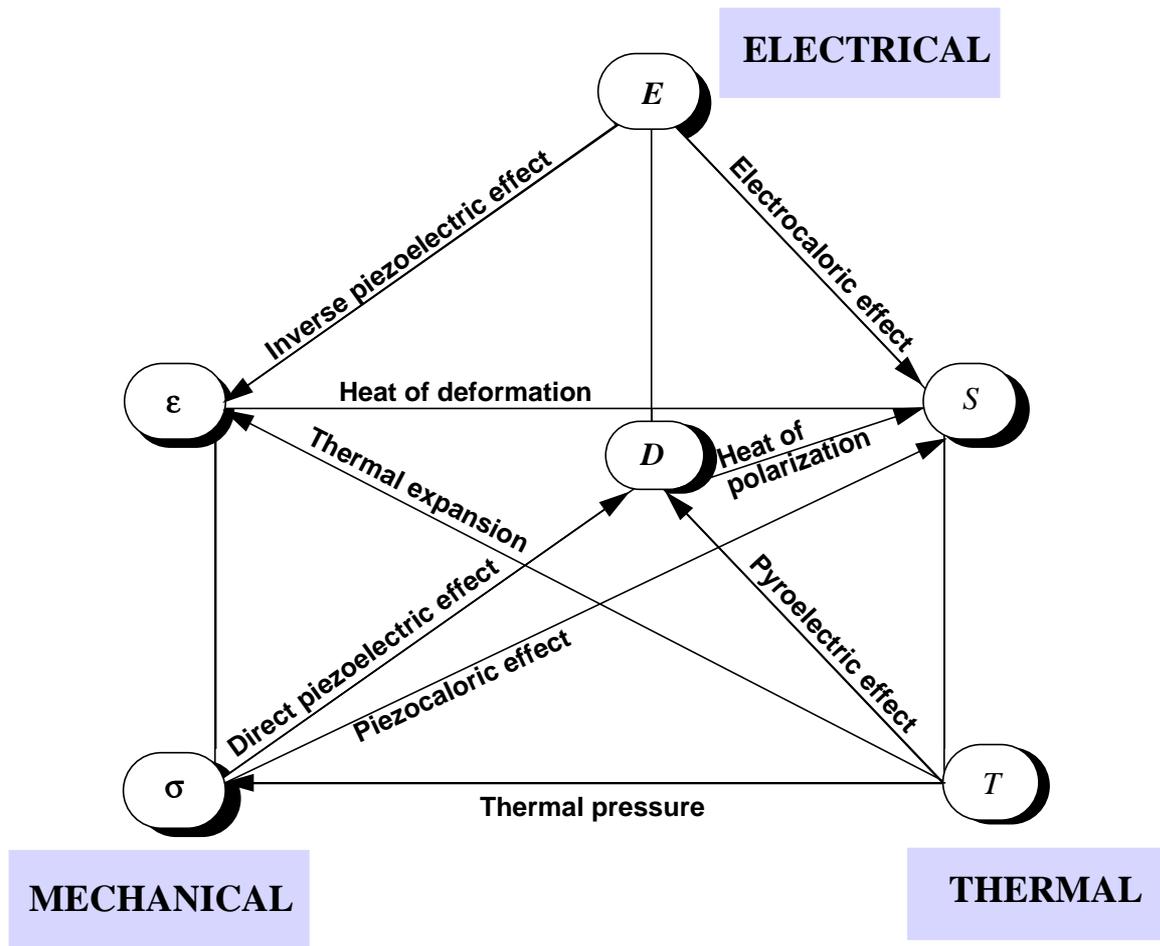
$$\rho\ddot{\mathbf{u}} - \nabla \cdot \left[ \mathbf{C} : \left( (\nabla\mathbf{u})^S + \frac{1}{2}\nabla\mathbf{u} \cdot (\nabla\mathbf{u})^T \right) + \boldsymbol{\pi}^T \cdot \nabla\phi + \boldsymbol{\alpha}\Delta T \cdot (1 + \nabla\mathbf{u}) \right] = \mathbf{f}. \tag{2.62}$$



**Figure 2.2.** The relations between different physical domains, namely the electrical, thermal and mechanical. The tensor rank of the susceptibilities is shown in square brackets whereas the tensors connecting the properties is given in index notation. Note that the intensive and extensive thermodynamic variables are situated on the lower and upper level of the prism, respectively.

Of course, we also take into account the electrostatic Gauss law [75] for the electrostatic displacement

$$\nabla \cdot \mathbf{D} = \rho, \quad (2.63)$$



**Figure 2.3.** Coupling effects between the fields. For clarity principal effects have been omitted in this Figure, for those refer to Figure 2.2. The fields are expressed in terms of their symbols.

$\rho$  in this context denoting the electrical charge density. Replacing the displacement by the expression derived in (2.36), (2.63) yields

$$\nabla \cdot \left( \boldsymbol{\pi} : \left( (\nabla \mathbf{u})^S + \frac{1}{2} \nabla \mathbf{u} \cdot (\nabla \mathbf{u})^T \right) + \boldsymbol{\chi} \cdot \nabla \phi + p \Delta T \right) = \rho. \quad (2.64)$$

This is a generalized equation of electrostatics, also comprising piezoelectric and pyroelectric effects. Analogously one could consider the rate of entropy change

with the additional independent variable  $\mathbf{E}$  representing the electrical field such that (2.46) reads

$$\dot{\hat{S}}(\boldsymbol{\epsilon}, T, \mathbf{E}) = \frac{\partial \hat{S}}{\partial T} \dot{T} + \frac{\partial \hat{S}}{\partial \boldsymbol{\epsilon}} : \dot{\boldsymbol{\epsilon}} + \frac{\partial \hat{S}}{\partial \mathbf{E}} \dot{\mathbf{E}}. \quad (2.65)$$

This identity could further be used in order to derive a more general heat conduction equation also comprising thermo-electrical effects. Making the solution of equations (2.62), (2.64) and (2.65) possible by means of a computer is the topic of the next chapter.



# 3 ACCURACY CONTROL FOR THIN STRUCTURE SIMULATION

In this chapter, we consider the numerical solution of the partial differential equations derived in the previous chapter. The reason for choosing numerical methods in deriving the physical fields simply is that the geometries of MEMS structures usually are too complex to be amenable to analytical solutions. However, when applying numerical methods one should care about the fact that these are approximation methods and thus errors are introduced inevitably. We begin this chapter by presenting the basic mathematical tools which allow for a reformulation of the partial differential equations and the corresponding boundary value problems (BVP) in a weak sense. Therefore we introduce a set of mathematical notions and techniques with which to correctly formulate the physical problems at a high level of abstraction. By doing so, we provide a wide range of physical models that can be dealt with numerically. The approximative solution method will be that of the finite elements which can be well established with the preparatory work. On the basis of this formulation we then are able to carry out a sound error analysis and propose methods with which these errors can be reduced at a minimum cost in terms of computational resources.

## 3.1 Mathematical preliminaries

A proper understanding of the theory of boundary value problems requires some background in functional analysis. In fact, many simulation techniques such as, for example, finite elements, do not require, at first sight, deeper understanding of functional analytical methods. But some background in functional analysis is an essential prerequisite for those who wish to gain a proper insight of qualitative

aspects of BVPs, or of aspects of the finite element method such as those that lead to the development of error estimates. In this section, by following the exposition of [76] we provide the basic facts about functional analysis which lead a reasonable in-depth study of BVPs describing MEMS physics and their approximation.

#### 3.1.1 Inner Product Spaces and Linear Operators

The idea of a vector space with the well known objects and operations and its generalization can be viewed as a starting point for functional analysis. Given a vector space  $X$  one can easily fix an inner product  $(u, v)$  of  $u, v \in X$  which satisfies the following axioms, for all  $u, v, w \in X$  and  $\alpha, \beta \in \mathbb{C}$ ,  $\mathbb{C}$  denoting the set of complex numbers:

- **A1:**  $(u, v) \in \mathbb{C}$  (the inner product is complex valued)
- **A2:**  $(u, v) = \overline{(v, u)}$  (the operation is Hermitian)
- **A3:**  $(\alpha u + \beta v, w) = \alpha(u, w) + \beta(v, w)$  (the inner product is linear in the first slot)
- **A4:**  $(u, u) \geq 0$  and  $(u, u) = 0$  if and only if  $u = 0$  (the inner product is positive definite)

Analogously one can define an inner product space on a real valued vector space with the slight change that hermiticity turns into symmetry in the second axiom. A standard example of an inner product space, besides the conventional or Euclidean scalar product is the space of square-integrable functions  $L^2(a, b)$  on an interval  $(a, b)$  and is defined as follows:

$$(u, v) = \int_a^b v(x) \overline{u(x)} dx. \quad (3.1)$$

An important property of vectors, or, at this stage of abstraction, of elements of a more general inner product space is that of orthogonality. Two vectors (or functions) are said to be *orthogonal* if

$$(u, v) = 0. \quad (3.2)$$

Another important characteristic that holds in any inner product space is the *Cauchy-Schwarz inequality*

$$|(u, v)| \leq (u, u)^{1/2} (v, v)^{1/2} \quad (3.3)$$

where  $| \cdot |$  denotes the absolute value.

By drawing an analogy with the notion of the length of a vector we can introduce the concept of a *norm*. Again we start from scratch with an arbitrary vector space  $X$  and declare an operation which we call *norm*  $\| \cdot \|$  on  $X$  and which satisfies the following axioms for any members  $u, v$  of  $X$ , and scalars (real or complex)  $\alpha$  :

- **N1:**  $\|u\| \in \mathbb{R}$  .
- **N2:**  $\|u\| \geq 0$  and  $\|u\| = 0$  if and only if  $u = 0$  (positive definiteness).
- **N3:**  $\|\alpha u\| = |\alpha| \|u\|$  (positive homogeneity).
- **N4:**  $\|u + v\| \leq \|u\| + \|v\|$  (triangle inequality).

A vector space  $X$  which is equipped with a norm is called a *normed space*. Even though the norm is a primitive concept and does not require for its definition the existence of an inner product, one usually uses the inner product in order to *generate the norm*:

$$\|u\| = \sqrt{(u, u)}. \quad (3.4)$$

Norms are an integral part of this work and are presented during the next sections.

The first norm is defined for any member belonging to the space  $X = L^p$ ,  $1 \leq p < \infty$ , of functions  $f$  for which the integral

$$\int_a^b |f(x)|^p dx$$

exists or, in other words, is smaller than infinity. The standard norm on  $L^p$  is defined by

$$\|u\|_{L^p} = \left[ \int_{\Omega} |u(x)|^p dx \right]^{1/p}, \quad (3.5)$$

where  $\Omega$  denotes some measurable, for practical purpose some bounded open set in  $\mathbb{R}^n$ . Usually one simply writes  $\|u\|_p$  instead of  $\|u\|_{L^p}$  to make things handier.

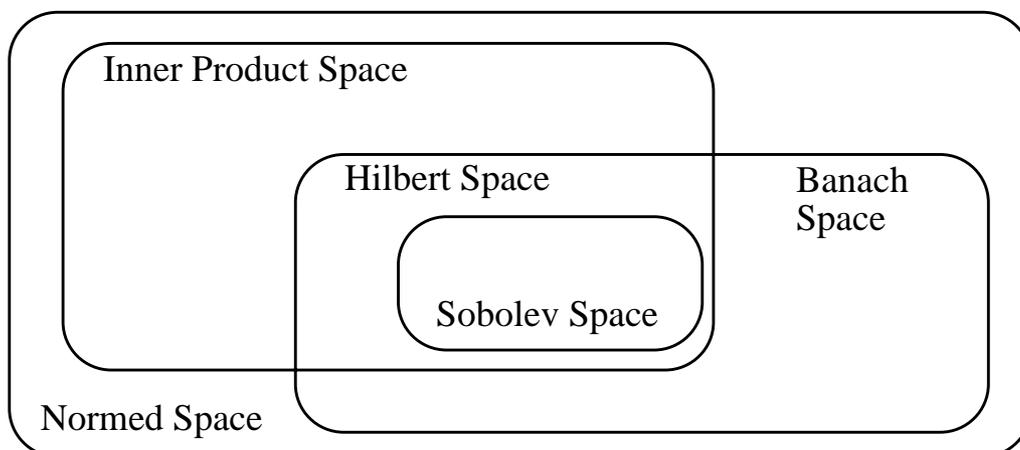
Considering the space  $L^\infty(\Omega)$  of bounded measurable functions, i.e. functions  $u$  that satisfy  $|u(x)| \leq \text{const}$  almost everywhere (a.e.) on  $\Omega$  we can define the norm  $\| \cdot \|_{L^\infty}$  by

$$\|u\|_{L^\infty} = \inf\{k: |u(x)| \leq k \text{ a.e.}\}, \quad (3.6)$$

the greatest lower bound of the constants  $k$  that bound  $|u|$  almost everywhere. Since for any given vector space the definition of a norm is not unique, the concept of *equivalence of norms* is useful. Two alternative norms  $\| \cdot \|_A$  and  $\| \cdot \|_B$  are said to be equivalent to each other if there are positive constants  $m$  and  $M$  such that

$$m\|u\|_A \leq \|u\|_B \leq M\|u\|_A. \quad (3.7)$$

The equivalence of norms is a helpful property with which to master extensive estimations much more easier, as we will see in section 3.3.1. Normed spaces that are complete in the sense that every Cauchy sequence converges are given a special name, namely *Banach spaces*. A special kind of a Banach space is a *Hilbert space*, the inner product space which is complete. Every Hilbert space is a Banach space since every inner product induces a norm. For the hierarchies of the different spaces see Figure 3.1.



**Figure 3.1** The different function subspaces.

Another fundamental concept in functional analysis is that of a mapping or *operator* from one space to another. Operators that map members of a specified space into the real or complex numbers are denoted *functionals* whereas linear operators that map pairs of elements into the real or complex numbers are called *bilinear forms*. The former two types of operators play a central role in the study of linear boundary value problems and thus are inevitable when sophisticated numerical simulation methods have to be developed.

A linear operator  $T$  is a mapping from a vector space  $X$ , the domain, into some other space and which is

- additive:  $T(u + v) = T(u) + T(v)$  for all  $u, v \in X$ ; and
- homogenous:  $T(\alpha u) = \alpha T(u)$ , where  $\alpha$  is either real or complex valued.

We say that a linear operator  $T: X \rightarrow Y$  is bounded, if it is possible to find a number  $K > 0$  such that

$$\|Tu\| \leq K\|u\| \quad \text{for all } u \text{ in } X. \quad (3.8)$$

For  $u \neq 0$  we define the norm of a linear operator  $T$  by taking the least upper bound of all  $K \geq \|Tu\|/\|u\|$ , taken over all members  $u$  of  $X$ . That is,

$$\|T\| = \sup\{\|Tu\|/\|u\|, u \neq 0\}. \quad (3.9)$$

Linear operators that map elements of a vector space into the set of the real or complex numbers (in the following we will restrict ourselves to the case of the real numbers since they are sufficient for our purpose) are called linear functionals

$$l: X \rightarrow \mathbb{R}.$$

Since the set of all bounded linear operators of a normed space  $X$  to a normed space  $Y$  is itself a normed space with the norm defined above in (3.9) we can introduce a special vector space  $L(X, \mathbb{R})$ , the space of bounded linear functionals on  $X$ . This space also is called the dual space of  $X$  and is denoted by

$$X^* = L(X, \mathbb{R}) . \quad (3.10)$$

From the boundedness of the linear functionals, that is, of each member  $l$  of  $X^*$  we have

$$\|l(u)\| = |l(u)| \leq K\|u\| \quad \text{for all } u \text{ in } X \quad (3.11)$$

with some constant  $K$ . The action of such a functional on an element  $u$  usually is denoted  $\langle l, u \rangle$  instead of  $l(u)$ . Referring to the definition of an operator's norm in (3.9), we see that the norm of a linear functional is given by

$$\|l\|_{X^*} = \sup \frac{|\langle l, u \rangle|}{\|u\|} \quad u \neq 0. \quad (3.12)$$

A well known example of a non-trivial linear functional is the Dirac delta “function” occurring in many fields of physics and engineering as an important concept. Here one usually refers to a quantity which is zero everywhere except at the origin where it takes the value infinity, that is,  $\delta(x) = 0$  for  $x \neq 0$  and  $\delta(x) \rightarrow \infty$  at  $x = 0$ . Moreover,  $\delta$  is assumed to have the following property

$$\int_{-\infty}^{\infty} \delta(x)u(x)dx = u(0) \quad (3.13)$$

for any continuous function  $u$ . It is, in fact, impossible to construct a function with these properties in the ordinary sense. To overcome this difficulty the Dirac delta function is better and more correctly defined as a bounded linear functional acting on the space of continuous functions  $C([a,b])$ ,  $[a,b]$  denoting a real interval,

$$\delta: C([a,b]) \rightarrow \mathbb{R}, \quad \langle \delta, u \rangle = u(0). \quad (3.14)$$

Having introduced the concept in this way, there is no difficulty in dealing with the Dirac delta. It is simply an operator acting on the set of continuous functions and samples the value at the origin. With the definition of a dual space above we see that the Dirac delta is a member of the dual of the continuous functions on an interval,  $\delta \in C([a,b])^*$ . A common physical interpretation of the delta function is a pointlike heat source acting within a given region or a located mechanical point force acting on some solid. The usefulness of this concept is evident when it

comes to the numerical simulation of systems where such types of interactions occur.

The most important operators when dealing with linear multiphysics problems and their related boundary value problems are the ones which map a pair of elements to the real numbers and which are linear in both arguments: the *bilinear forms*. For vector spaces  $X$  and  $Y$ , we may define a bilinear form  $a$  to be an operator  $a: X \times Y \rightarrow \mathbb{R}$  with the following properties:

$$\begin{aligned} a(\alpha u + \beta v, w) &= \alpha a(u, w) + \beta a(v, w) & u, w \in X, \quad v \in Y \\ a(u, \alpha v + \beta w) &= \alpha a(u, v) + \beta a(u, w) & u \in X, \quad v, w \in Y \end{aligned} \quad (3.15)$$

where  $\alpha, \beta$  are real numbers.

Bilinearforms will turn out to be closely related to partial differential equations and their numerical solutions. It is, in a certain sense, essential that bilinear forms possess a characteristic called *continuity* for the problem which they describe to be well-posed. A bilinear form is said to be continuous if there is a constant  $K$  such that

$$|a(u, v)| \leq K \|u\| \|v\| \quad \text{for all } u \in X, \quad v \in Y, \quad (3.16)$$

where  $X$  and  $Y$  are normed vector spaces. We mention another property of bilinear forms which is called *H-ellipticity*. Given an inner product space  $H$  and a bilinear form  $a: H \times H \rightarrow \mathbb{R}$  we say that  $a$  is *H-elliptic* if there is a constant  $\alpha$ , the ellipticity-constant, such that

$$a(v, v) \geq \alpha \|v\|^2 \quad \text{for all } v \in H. \quad (3.17)$$

In other words, an *H-elliptic* bilinear form always is non-negative or, *positive definite*.

With the abstract framework of functional analytical tools which we have set up by now, we are able to state a result that has turned out to be a cornerstone in science and engineering [2], the *Lax-Milgram theorem*. The power of abstraction guarantees that some basic models of science including electrostatic problems or linear elasticity have a satisfactory mathematical form and may be solved by

appropriate numerical methods. Under the assumption that a given bilinear form  $a$ , operating on a Hilbert space,  $a: H \times H \rightarrow \mathbb{R}$  is continuous and  $H$ -elliptic and that there is any given bounded linear functional  $l$  on  $H$ , there exists a unique element  $u$  in  $H$  such that

$$a(u, v) = \langle l, v \rangle \quad \text{for all } v \in H. \quad (3.18)$$

It seems advisable to anticipate that the bilinear form  $a$  represents the energy of a linear physical system, if there is an energy principle, and the linear functional has the meaning of some external source function. So, in other words, the Lax-Milgram theorem states that for a given external excitation there is a unique state that is adopted by the linear physical system by means of its energy. Equivalently one could formulate this fact via a minimization problem where the minimum energy is sought [77].

#### 3.1.2 Sobolev Spaces

In the previous chapter we derived partial differential equations which from a mathematical point of view require their solutions to be as smooth as the highest order of the acting differential operator indicates. This is often a drawback when modeling effects such as point forces acting on some elastic structures and, a similar example, a point-like heatsource in heat conduction. Some other physical problems may well require that data be modeled representing discontinuous material properties as it is the case for composite materials. Whereas the classical, i.e. partial differential formulation does not permit a treatment of such problems, the variational formulation offers a natural setting, since here we work in larger spaces. In this section we address ourselves to the task of developing a framework of a mathematically sound formulation of variational boundary value problems (VBVP).

The central concept will be that of Sobolev spaces, since these provide a means of characterizing the degree of smoothness of functions. And, probably the most important fact about Sobolev spaces is that numerical approximation methods such as the finite element method are most conveniently and correctly formulated in machine computable finite dimensional subspaces of Sobolev spaces.

We start the exposition by introducing the term of the *weak derivative* of a function. We call a function  $D^\alpha u$  the weak derivative of the function  $u$  if the following relation holds:

$$\int_{\Omega} D^\alpha u(x) \phi(x) dx = (-1)^{|\alpha|} \int_{\Omega} u(x) D^\alpha \phi(x) dx. \quad (3.19)$$

Recall that for the multi index notation  $|\alpha| = \alpha_1 + \dots + \alpha_n$  where  $\alpha = (\alpha_1, \dots, \alpha_n)$ . The *test function*  $\phi$  is chosen to be from the space  $C_0^\infty(\Omega)$  of infinitely differentiable functions with compact support. In case  $u$  is sufficiently smooth to belong to  $C^m(\bar{\Omega})$ ,  $\bar{\Omega}$  denoting the closure of the open set  $\Omega$ , then its weak derivatives  $D^\alpha u$  coincide with its classical derivatives for  $|\alpha| = m$ . To illustrate the concept of a weak derivative we refer to the following example. The function  $u(x) = |x|$  is continuous in the interval  $[-1,1]$ , that is, it belongs to  $C([-1,1])$ . Its classical derivative does not exist in the sense that it is not defined at the origin. The weak derivative, however, is the function

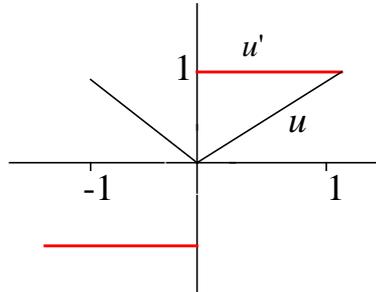
$$u' = \begin{cases} -1 & \text{for } -1 \leq x < 0 \\ 1 & \text{for } 0 \leq x < 1 \end{cases}$$

as sketched in Figure 3.2, since the equation

$$\int_{-1}^1 u' \phi dx = - \int_{-1}^1 u \phi' dx$$

holds.

The fundamental difference between the classical derivative of a function and the weak derivative is that the former is defined pointwise, i.e. for each point of its domain, whereas the latter only requires to be integrable. A function to be differentiable in the classical sense at least has to be continuous. The condition of weak differentiability thus invokes that a function  $v$  differing from a weak derivative of  $u$  on a set of measure zero is itself a weak derivative of  $u$ . One could guess that this concept is quite useful as soon as it comes to the numerical simulation of systems that possess discontinuities in their constitutive properties.



**Figure 3.2** The function  $u$  which is not differentiable in the conventional sense, whereas its weak derivative  $u'$  exists.

The *Sobolev space* of order  $m$ , denoted by  $H^m(\Omega)$ ,  $m$  being a non negative integer, is defined to be the space of all functions that are square integrable, i.e. that are in  $L^2(\Omega)$ , together with all their weak partial derivatives up to and including those of order  $m$ , belong to  $L^2(\Omega)$ :

$$H^m(\Omega) = \{u: D^\alpha u \in L^2(\Omega) \text{ for all } \alpha \text{ such that } |\alpha| \leq m\}. \quad (3.20)$$

With that, we may define an inner product space with the Sobolev inner product  $(\cdot, \cdot)_{H^m}$  as follows:

$$(u, v)_{H^m} = \int_{\Omega} \sum_{|\alpha| \leq m} (D^\alpha u)(D^\alpha v) dx \quad \text{for } u, v \in H^m(\Omega). \quad (3.21)$$

As we saw earlier an inner product generates a norm, here it is called the Sobolev norm defined by:

$$\|u\|_{H^m}^2 = (u, u)_{H^m} = \int_{\Omega} \sum_{|\alpha| \leq m} (D^\alpha u)^2 dx. \quad (3.22)$$

This relation can be recast into

$$\|u\|_{H^m}^2 = \sum_{|\alpha| \leq m} \|D^\alpha u\|_{L^2}^2 = \|D^{\alpha, |\alpha|=0} u\|_{L^2}^2 + \dots + \|D^{\alpha, |\alpha|=m} u\|_{L^2}^2. \quad (3.23)$$

When written out in full for the case  $m = 1$ , as it occurs when dealing with second order problems such as the Poisson equation, the Sobolev norm reads

$$\|u\|_{H^1}^2 = \int_{\Omega} \left( u^2 + \left( \frac{\partial u}{\partial x} \right)^2 + \left( \frac{\partial u}{\partial y} \right)^2 \right) dx. \quad (3.24)$$

A concept that is quite similar to that of a norm for Sobolev spaces is that of a semi- norm. A seminorm  $|\cdot|$  satisfies all the norm axioms stated in section 3.1.1 except that of positive definiteness, meaning that  $|u| \geq 0$  but  $|u| = 0$  does not necessarily imply that also  $u$  vanishes. The seminorm on  $H^m(\Omega)$  is defined as

$$|u|_{H^m}^2 = \int_{\Omega} \sum_{|\alpha|=m} (D^{\alpha}u)^2 dx. \quad (3.25)$$

In fact,  $|\cdot|_{H^m}$  is a seminorm, however, it is not a norm since if the weak derivatives vanish for  $\alpha = m$  this does not mean for  $u$  to vanish itself.

Obviously we have the relation

$$L^2(\Omega) = H^0(\Omega) \supset H^1(\Omega) \supset H^2(\Omega) \dots \quad (3.26)$$

There are, in fact, properties of Sobolev spaces that highly depend on the dimension of the domain where its functions are defined on. As one would expect, for a real interval  $[a,b]$  the functions in  $H^1([a,b])$  are continuous. However, this is not the case anymore for  $H^1$ -functions defined on an open subset  $\Omega$  in a space with a dimension higher than one as the following example clarifies. We consider the function  $u(\mathbf{x}) = r^{-\alpha}$ ,  $\alpha < 1/2$  where  $\|\mathbf{x}\|^2 \equiv r^2 = x^2 + y^2 + z^2$  and  $\Omega$  is a ball containing the origin. Then

$$\|u\|_{H^1}^2 = \int_{\Omega} (u^2 + (\nabla u)^2) dx = \int_{\Omega} (\|\mathbf{x}\|^{-2\alpha} + (\nabla(\|\mathbf{x}\|^{-\alpha}))^2) dx$$

which, when introducing spherical coordinates yields

$$\|u\|_{H^1}^2 = \iiint (\|\mathbf{x}\|^{-2\alpha} + \alpha^2 \|\mathbf{x}\|^{-2\alpha-2}) \|\mathbf{x}\|^2 \sin \theta d\theta dr d\phi.$$

Now, since  $\alpha$  has been assumed to be smaller than one half, we clearly see that the integral is bounded whereas the function  $u$  is singular at the origin. A more general statement is given by the *Sobolev embedding theorem* which gives a result for a bounded domain  $\Omega$  of  $\mathbb{R}^n$  with a boundary that is smooth enough (that is “Lipschitz” to be more correctly). Then, if  $m - k > n/2$ , every function in  $H^1(\Omega)$  belongs to  $C^k(\bar{\Omega})$ . Taking into account this fact becomes of prominent importance when dealing with functions which represent physical fields in a more general sense, including the occurrence of singularities. Therefore, some care has to be exercised in the approximation of such fields by approximation functions that usually are taken to be non-singular. We will pick this up in section 3.3.2.

#### 3.1.3 Elliptic Boundary Value Problems

As we have pointed out previously, the aim is that a desired solution of a physical boundary value problem is not necessarily required to be continuously differentiable, that is, in the space  $C^{2m}(\Omega)$  where  $2m$  is the order of the partial differential equation but in a less restrictive space  $H^n(\Omega)$  of a certain order  $n$ . And in turn, to make the calculation accessible to computing machines, only a subspace of  $H^m(\Omega)$  is desired. In this section we show the connection with the originally stated partial differential equations in chapter 2 and the corresponding variational boundary value problems. As having stated earlier, the shape of the simulation domain is expected to be reasonable in the sense that it is bounded and that its boundary is enough smooth. The shape of the linear partial differential equation with operator  $L$  is basically of the shape

$$Lu = \sum_{|\alpha|, |\beta| \leq m} (-1)^m D^\alpha (a_{\alpha\beta}(\mathbf{x}) D^\beta u) = f \quad \text{in } \Omega \quad (3.27)$$

and of the order  $2m$ . The set of boundary conditions can be partitioned into two subsets: The *essential boundary conditions* are those which are of order  $< m$  and the *natural boundary conditions* are those of order  $\geq m$ . In the case that there are mixed boundary conditions specified, that is, the boundary condition operator  $B_j$  contains differential operators of more than one order, the type of boundary conditions are to be determined by the highest order differential operator involved. The boundary conditions read as follows:

$$\begin{array}{rcl}
 B_0 u = 0 & \left. \vphantom{B_0 u = 0} \right\} & \text{(essential)} \\
 \dots & & \\
 B_{p-1} u = 0 & \left. \vphantom{B_{p-1} u = 0} \right\} & \\
 \\
 B_p u = g_p & \left. \vphantom{B_p u = g_p} \right\} & \text{(natural)} \\
 \dots & & \\
 B_{m-1} u = g_{m-1} & \left. \vphantom{B_{m-1} u = g_{m-1}} \right\} & 
 \end{array} \tag{3.28}$$

We would like to remark that further on we confine ourselves to the case of homogenous essential boundary conditions. But this is not a real restriction since it is a straightforward matter to convert any problem with non-vanishing boundary values to one whose boundary conditions are homogenous: For simplicity we consider a PDE of the same structure as in (3.27) which is of order two:

$$\begin{array}{ll}
 Lu = f & \text{in } \Omega \\
 u = g & \text{on } \partial\Omega .
 \end{array}$$

Then, assuming that there is a function  $u_0$  in  $\Omega$  which coincides with  $g$  on the boundary we define  $w \equiv u - u_0$  and  $f_1 \equiv f - Lu_0$  for which

$$\begin{array}{ll}
 Lw = f_1 & \text{in } \Omega \\
 w = 0 & \text{on } \partial\Omega
 \end{array}$$

holds. We shall point out that up to now we have restricted ourselves to the case where only scalar valued functions are treated. However, the physical fields of practical interest often are vector-valued as for example, the mechanical displacement fields. Moreover, there are sets of coupled fields, such as the temperature field caused by a mechanical displacement as we saw in the first chapter. In these cases the precedent formulations first have to be modified in terms of the notation: vector valued functions will be represented in bold case, i.e.,  $\mathbf{u}$ . The multi-physical fields in turn are merged into  $\mathbf{U} = (\mathbf{u}, w)$ , a vector valued field with a dimension according to the dimension of its components.

A crucial step then is to define a space  $X$  of admissible functions in which the solution of the variational boundary value problem is to be sought. For the above problem (3.27) and (3.28) this space is defined by

$$X = \{v \in H^m(\Omega): B_j = 0 \text{ on } \Gamma, j = 1, \dots, p-1\}, \quad (3.29)$$

$\Gamma$  representing the part of the boundary where essential, or *Dirichlet* boundary conditions are specified, or in other words

$$X = \{v \in H^m(\Omega): v \text{ satisfies all essential boundary conditions}\}. \quad (3.30)$$

We shall emphasize that only essential boundary conditions are taken care of by the requirement that the solution sought for be in the space  $X$ . Sobolev spaces that directly handle sets of homogenous essential boundary conditions are often denoted by  $H_0^m(\Omega)$ . Natural or Neumann boundary conditions are to be treated in another way which will be demonstrated below. The transition from the classical BVP to the VBVP then is performed by multiplying both sides of (3.27) by an arbitrary function  $v$  from  $X$ , integrating and using Green's theorem to further reduce the expression to one of the form

$$a(u, v) = \langle l, v \rangle. \quad (3.31)$$

In the case of a Dirichlet problem for the Poisson equation with a given source function  $f$  that is square integrable, i.e.  $f \in L^2(\Omega)$

$$-\nabla^2 u = f \quad \text{in } \Omega \quad (3.32)$$

$$u = 0 \quad \text{on } \Gamma. \quad (3.33)$$

the first step results in

$$-\int_{\Omega} (\nabla^2 u) v dx = \int_{\Omega} f v dx \quad (3.34)$$

where the order of the Sobolev space has to be fixed as one,  $H_0^1(\Omega)$ , taking into account that first order derivatives must be square integrable. Then, partial integration gives

$$-\int_{\Gamma} (\mathbf{n} \cdot \nabla u) v d\Gamma + \int_{\Omega} \nabla u \cdot \nabla v dx = \int_{\Omega} f v dx. \quad (3.35)$$

where  $\mathbf{n}$  denotes the surface normal vector pointing to the outside of the domain. Now, the reason why boundary conditions involving derivatives of order  $m$  are called natural becomes quite clear: They arise in a natural way when performing the transition from the classical boundary value problem to the variational problem. Since  $v$  was chosen to be from the space  $H_0^m(\Omega)$  the first integral on the left hand side vanishes and we are left with

$$\int_{\Omega} \nabla u \cdot \nabla v dx = \int_{\Omega} f v dx \quad (3.36)$$

which can be recast into

$$a(u, v) = \langle l, v \rangle, \quad (3.37)$$

when the bilinear form  $a$  is defined as

$$a(u, v) \equiv \int_{\Omega} \nabla u \cdot \nabla v dx \quad (3.38)$$

and the linear functional  $l$  by

$$\langle l, v \rangle \equiv \int_{\Omega} f v dx. \quad (3.39)$$

In the case where non vanishing natural boundary conditions such as

$$\mathbf{n} \cdot \nabla u = g \quad \text{on } \Gamma_N \quad (3.40)$$

are specified on a part of the domain boundary  $\Gamma_N$  with a non-vanishing function  $g$  that is square integrable, the linear form has to be modified accordingly:

$$\langle \tilde{l}, v \rangle \equiv \int_{\Omega} f v dx + \int_{\Gamma_N} g v d\Gamma = \langle l, v \rangle + \int_{\Gamma_N} g v d\Gamma. \quad (3.41)$$

With that, the boundary value problem reads

$$a(u, v) = \langle \tilde{l}, v \rangle. \quad (3.42)$$

Recall that, since we are dealing with spaces that are a special kind of those introduced in section 3.1.1 existence and uniqueness of a solution turn out under the same conditions of  $X$ -ellipticity and continuity that have to be met by the bilinear form and the continuity of the linear functional. This follows from the Lax-Milgram theorem.

#### 3.1.4 The Galerkin Method

Except for problems involving very simple PDEs and geometries, it is quite impossible to obtain exact solutions in either the classical or variational formulations. The idea then is to *approximate* the solution. The idea of the Galerkin approximation method, that later on leads to finite element methods is quite simple, when disposing of the framework established in the previous sections. We would like to emphasize that the nomination of this section's method by no means is uniformly handled in the literature. Again, we follow the notation of the authors of [78] or [76]. Consider the variational boundary problem of finding a solution  $u \in V$  that satisfies

$$a(u, v) = \langle l, v \rangle \quad \text{for all } v \in V \quad (3.43)$$

where  $V$  is some subspace of a Hilbert space  $H$ . The difficulty when searching a solution, however, is that the space  $V$  is infinite-dimensional with the result that it is impossible to set up a practical method to solve (3.43). Therefore one chooses a finite dimensional subspace  $V_h$  of the space  $V$  where the solution has to be sought for. Thus, a basis of  $V_h$  has to be fixed which spans any function in  $V_h$ . In other words, a finite number  $N$  of linearly independent functions  $\phi_i$  are chosen such that

$$V_h \subset V \quad \text{and} \quad \text{span}\{\phi_i\}_{i=1}^N = V_h. \quad (3.44)$$

The index  $h$  of the space  $V_h$  in that sense refers to the term of an approximation or more correctly a discretization parameter as it gets smaller as larger  $N$  and thus the number of basis functions gets. One would expect that the approximation is the better the higher the dimensionality of the discrete subspace  $V_h$  is taken. In other words, when taking the limit of  $h$  tending towards zero one expects  $V_h$  to

approach  $V$ . Disposing of the space  $V_h$ , problem (3.43) is now posed in  $V_h$  instead of  $V$ . Thus, we try to find a function  $u_h \in V_h$  that satisfies

$$a(u_h, v_h) = \langle l, v_h \rangle \quad \text{for all } v_h \in V_h. \quad (3.45)$$

The ansatz for solving for  $u_h$  is as follows:  $u_h$  and  $v_h$  must be linear combinations of the basis functions of  $V_h$ ,

$$u_h = \sum_{i=1}^N c_i \phi_i \quad \text{and} \quad v_h = \sum_{j=1}^N d_j \phi_j. \quad (3.46)$$

Inserted into (3.45) this yields

$$a\left(\sum_{i=1}^N c_i \phi_i, \sum_{j=1}^N d_j \phi_j\right) = \langle l, \sum_{j=1}^N d_j \phi_j \rangle \quad \text{for any coefficients } d_j \quad (3.47)$$

or

$$\sum_{i=1}^N \sum_{j=1}^N a(\phi_i, \phi_j) c_i d_j = \sum_{j=1}^N \langle l, \phi_j \rangle d_j \quad \text{for any coefficients } d_j \quad (3.48)$$

where the linearity of  $l$  and the bilinearity of  $a$  has been used. Defining

$$K_{ij} = a(\phi_i, \phi_j) \quad \text{and} \quad F_j = \langle l, \phi_j \rangle \quad (3.49)$$

(3.48) can be recast into

$$\sum_{j=1}^N d_j \left( \sum_{i=1}^N K_{ij} c_i - F_j \right) = 0 \quad \text{for any coefficients } d_j \quad (3.50)$$

which is equivalent to

$$\sum_{i=1}^N K_{ij} c_i = F_j, \quad j = 1, \dots, N \quad (3.51)$$

or in the classical finite element matrix-vector notation,

$$K \cdot c = F \quad (3.52)$$

representing a set of simultaneous linear equations. The desired field's approximation is obtained by inserting the linear equations' solution into (3.46).

The fact that we only have approximated the solution sought in (3.43) in a natural way leads to the question how to characterize the error  $e$  introduced by the approximation,

$$e = u - u_h. \quad (3.53)$$

We exploit the property of  $V_h$  being a subspace of  $V$ , so in (3.43) we can choose  $v$  to be from that subspace,  $v \in V_h$  and considering that by denoting this member by  $v_h$  yields

$$a(u, v_h) = \langle l, v_h \rangle \quad \text{for all } v_h \in V_h. \quad (3.54)$$

The problem (3.45), in the following also referred to as the discrete problem, when subtracted from (3.54),

$$a(u, v_h) - a(u_h, v_h) = \langle l, v_h \rangle - \langle l, v_h \rangle \quad \text{for all } v_h \in V_h, \quad (3.55)$$

results in

$$a(u - u_h, v_h) = a(e, v_h) = 0 \quad \text{for all } v_h \in V_h, \quad (3.56)$$

having used the bilinearity of the form  $a$ . This property commonly is labelled as the *orthogonality of the error* to  $V_h$  in accord with the geometrical interpretation of orthogonality of inner products in the event that  $a$  is symmetric.

When assuming  $V$ -ellipticity and continuity of the bilinearform  $a$  and taking into account that  $u - u_h$  also lies in  $V$ , we have the estimate

$$\begin{aligned}
 \alpha \|u - u_h\| &\leq a(u - u_h, u - u_h) \\
 &= a(u - u_h, u - u_h - v_h + v_h) \\
 &= a(u - u_h, u - v_h) - a(e, u_h - v_h) .
 \end{aligned} \tag{3.57}$$

Then, applying the continuity of the bilinearform and considering that the error is orthogonal to  $V_h$  gives

$$\alpha \|u - u_h\| \leq a(u - u_h, u - v_h) \leq M \|u - u_h\| \|u - v_h\| . \tag{3.58}$$

This estimate can be expressed as

$$\|u - u_h\| = \frac{M}{\alpha} \inf_{v_h \in V_h} \|u - v_h\| \tag{3.59}$$

which is also known as Céa's lemma. Therefore, determining how big the error is, can be transformed into the question of estimating the distance of  $u$  from the subspace  $V_h$ . We will see later that this result can be substantiated in that sense that convergence becomes more obvious when the dimension of the subspace  $V_h$  is increased.

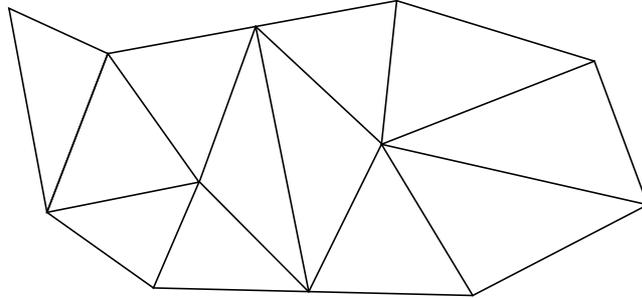
### 3.1.5 The Finite Element Method (FEM)

The practical aspects of the Galerkin method lead to the finite element method. Central to this technique is that it provides for types of basis functions suitable for domains with virtually arbitrary shape. These are piecewise polynomials that are non-vanishing only on a small part of the simulation domain, their support is local and overlaps only with few of the other basis functions' supports. The aim of this section is provide for the methods with which to construct those special bases.

Starting point is the decomposition  $S$  of the domain  $\Omega$  into a finite number  $v$  of non-overlapping subdomains  $\Omega_1, \Omega_2, \dots, \Omega_v$ , that cover  $\Omega$ :

$$\Omega_\lambda \cap \Omega_\mu = \emptyset \quad \text{for } \lambda \neq \mu, \quad \bigcup_{\mu=1}^v \Omega_\mu = \bar{\Omega}.$$

Here, the symbol  $\emptyset$  just stands for a set of measure zero in the space considered. We assume the domain boundary to be polygonal if the domain is in a more than one dimensional real space. That is, the domain boundary is made up of piecewise polygons, in  $\Omega \subset \mathbb{R}^2$  they are straight lines. Thus, the entire domain can be covered by polygonal elements, as illustrated in Figure 3.3.



**Figure 3.3** Admissible subdivision of a domain into triangular elements.

Further criteria which we shall impose on the subdivision of the 2D domain into triangles or quadrilaterals include the following:

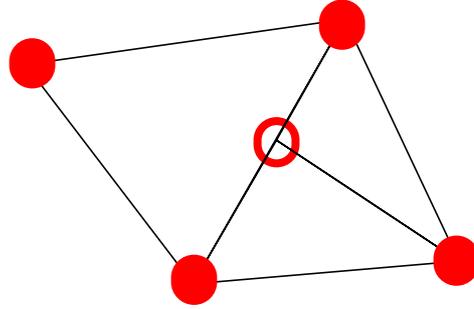
- if the intersection of two elements  $\Omega_\lambda \cap \Omega_\mu$  consists of exactly one point, then this is a corner point both of  $\Omega_\lambda$  and  $\Omega_\mu$ .
- if the intersection of two elements  $\Omega_\lambda \cap \Omega_\mu$  for  $\lambda \neq \mu$  consists of more than one point, then  $\Omega_\lambda \cap \Omega_\mu$  is an edge of  $\Omega_\lambda$  as well as of  $\Omega_\mu$ .
- a subdivision  $S$  sometimes is called shape regular, if there exists a constant  $\kappa > 0$  such that each of the elements  $\Omega_\lambda$  contains a circle of radius  $\rho_\Omega$  and

$$\rho_\Omega > \frac{h_\Omega}{\kappa}$$

holds,  $h_\lambda = \max \{ |\mathbf{x} - \mathbf{y}|, \mathbf{x}, \mathbf{y} \in \Omega_\lambda \}$  denoting the diameter of  $\Omega_\lambda$ .

An admissible triangulation therefore does not allow for hanging or dangling nodes, as sketched in Figure 3.4.

Bearing in mind that the basis functions span a subspace of the solution space  $X$  that satisfies all essential boundary conditions we are now ready to describe how



**Figure 3.4** An inadmissible triangulation. The open circle indicates the hanging node.

the finite element basis functions  $\{N_i\}_{i=1}^n$ ,  $n$  denoting the dimension of the finite element subspace, in general are formed.

- The functions  $N_i$  are bounded and belong to a subspace of the continuous functions on  $\bar{\Omega}$ , that is,

$$N_i \in C(\bar{\Omega}) \quad (3.60)$$

- The functions  $N_i$  are piecewise polynomials, that is, the restriction  $N_i^{(e)}$  of  $N_i$  to an element  $\Omega_e$  is a polynomial with degree  $k$ :

$$N_i|_{\Omega_e} \equiv N_i^{(e)}, \quad N_i^{(e)} \in \mathbb{P}_k(\Omega_e) \quad \text{for some } k \geq 1. \quad (3.61)$$

The  $N_i^{(e)}$  are called local basis functions since they are defined in order to have local (elementwise) support

$$\text{supp } N_i^{(e)} = \overline{\{\mathbf{x}, N_i^{(e)}(\mathbf{x}) \neq 0\}}. \quad (3.62)$$

Therefore the Matrix (3.51) has only few non-vanishing components. In the case where the  $\Omega_e$  are triangles and  $k = 1$ , that is, the polynomials are linear, the basis functions are defined as

$$N_i(\mathbf{x}_j) = \delta_{ij}, \quad (3.63)$$

$\delta_{ij}$  denoting the Kronecker delta, and  $x_j$  the geometrical nodes such that each basis function's support consists of all triangles containing node  $x_i$ .

Finite elements are said to be *conforming* if the basis functions are contained in the space in which the variational problem is posed. The task of finding the degree of the elementwise polynomials suited for conforming treatment of a given variational boundary value problem is not a trivial matter. It is facilitated by the result that for a function  $u: \bar{\Omega} \rightarrow \mathbb{R}$  which is piecewise arbitrarily often differentiable the following equivalence holds:

$$u \in H^k(\Omega) \Leftrightarrow u \in C^{k-1}(\bar{\Omega}). \quad (3.64)$$

In other words, when dealing with the Poisson equation which represents a second order problem and thus, according to section 3.1.3, the desired space is  $H^1(\Omega)$ . The polynomials therefore only have to be continuous, i.e.,  $\in C^0(\bar{\Omega})$ , for the ansatz to be conforming. This is not true any longer for fourth order problems representing a part of the plate equations. Since second order weak derivatives must be square integrable the appropriate space is  $H^2(\Omega)$  and thus the polynomials must be continuously differentiable, i.e. belong to  $C^1(\bar{\Omega})$ .

Having established the mechanisms used to set up finite elements we will now present the main approximation results also known as *a priori error estimates* which for a given choice of finite elements specify how far in some sense, the distance from the weak VBVP solution to the approximate or computed solution is. We define the interpolant of some continuous function on an element  $\Omega_h$  by

$$I_h: C(\Omega_h) \rightarrow X_h, \quad I_h v = \sum_{i=1}^N v(x_i) N_i^{(h)} \quad (3.65)$$

where  $X_h = \text{span} \{N_i^{(h)}\}$  and  $N_i^{(h)}$  are the local basis functions restricted to the elements  $\Omega_h$ . Recalling that in (3.59) we have established a Galerkin error estimate which now can be reduced to investigating the convergence of interpolates. With the choice (3.65) inequality (3.59) can be extended to

$$\|u - u_h\| = \frac{M}{\alpha} \inf_{v_h \in X_h} \|u - v_h\| \leq C \|u - I_h v\|. \quad (3.66)$$

Then, assuming the domain is decomposed into triangles according to the criteria given above, the standard error estimation reads

$$\|D^r(u - I_h v)\|_{L^2(T)} \leq c h_T^{q+1-r} \|D^{q+1} u\|_{L^2(T)} \quad (3.67)$$

where  $u$  has to be smooth enough, i.e.  $u \in H^{q+1}(T)$  and  $0 \leq r \leq q$ ,  $q$  denoting the polynomial degree of the interpolant  $I_h v$ . The quantity  $h_T$  is the diameter of the element  $T$ . Recall that  $D^r$  denotes the weak derivative operator using the multi index notation. This estimate of course can be substantiated further. Setting  $r$  to zero and choosing the elements to be linear, that is, setting the polynomial degree to one, we are left with

$$\|u - I_h v\|_{L^2(T)} \leq c h_T^2 \|D^2 u\|_{L^2(T)}. \quad (3.68)$$

If a bound for second order derivatives of  $u$  could be determined in some way, which is often done in regularity theorems, see for example [77], [78], or [80], then the convergence goes with the square of the triangle diameter. Interpolating with higher order polynomials naturally yields higher order convergence. However, this is only the case when the weak solution is smooth enough, see 5.2. This is a quantitative explanation of the well known and widely used practice of improving simulation results by either refining the mesh geometrically or increasing the polynomial order of the approximation space. We won't go into further details but want to state that this relation also holds globally, that is, on the whole domain

$$\|D^r(u - I_h v)\|_{L^2(\Omega)} \leq c h^{q+1-r} \|D^{q+1} u\|_{L^2(\Omega)} \quad (3.69)$$

where  $h$  denotes the maximum diameter of all elements. A detailed study of convergence results is found for example, in [78]. Summing up the results we can identify three factors bounding the error. Having fixed the order of the weak derivative,  $r$ , the power of  $h$  depends only on the degree of the polynomials and indicates the rate of convergence as the mesh is refined. This effect might be well observed in practical numerical studies. The constant  $c$  mainly depends on the geometrical shape of the elements and its nodal parameters. It increases as ele-

ments get worse, that is, tend to have small angles and high aspect ratios. The reason for choosing a highly regular mesh in the above specified sense now becomes quite obvious from a quantitative point of view. The third factor reflects the properties of the problem itself: the degree to which the solution is smooth and therefore easy to approximate accurately.

## 3.2 Modeling thin structures

### 3.2.1 Linear Elements for Thermal Problems

In chapter 2 we derived the partial differential equations for heat conduction (2.51). In the following we assume the heat transfer to be stationary and the mechanical stress not to be dependent on the temperature such that the heat equation reads

$$\nabla \cdot (\boldsymbol{\kappa} \cdot \nabla T(x, y, z)) = f, \quad (3.70)$$

which is also valid for several other physical problems described by a Poisson equation, such as, e.g. electrostatical problems (when symbols are exchanged accordingly). Within the sound mathematical framework established in the beginning of this chapter it is only a short way to obtain numerical results of practical interest.

We begin by defining the simulation domain  $\Omega \subset \mathbb{R}^3$  describing the device geometry, which is three dimensional. Boundary conditions that usually are applied comprise Dirichlet boundary conditions as well as Neumann boundary conditions. The latter are natural BCs and describe some thermal flux across the domain boundary. If these are zero the boundary part where they are specified is thermally insulated. The former are essential BCs and reflect the fact that some constant temperature reservoir is in contact with the device. We state these conditions as

$$\begin{aligned} T(x, y, z) &= T_0 \quad \text{on } \Gamma_D, \\ \mathbf{n} \cdot \boldsymbol{\kappa} \cdot \nabla T(x, y, z) &= g_N \quad \text{on } \Gamma_N, \end{aligned} \quad (3.71)$$

where  $\kappa$  denotes the thermal conductivity. We fix the solution space  $V$  as the Sobolev space containing all functions belonging to  $H^1(\Omega)$  that satisfy the essential boundary conditions

$$V = \{T(x, y, z) \in H^1(\Omega) : T(x, y, z) = T_0 \quad \text{on } \Gamma_D\} \quad (3.72)$$

such that we may rewrite (3.38) as

$$a(T, U) \equiv \int_{\Omega} \nabla T \cdot \kappa \cdot \nabla U \, dx \quad (3.73)$$

and the right hand side source term (3.41) as

$$\langle \tilde{l}, U \rangle \equiv \int_{\Omega} fU \, dx + \int_{\Gamma_N} g_N U \, d\Gamma. \quad (3.74)$$

The continuous weak problem of finding the temperature distribution then reads

$$a(T, U) = \langle \tilde{l}, U \rangle \quad \text{for all } U \in V. \quad (3.75)$$

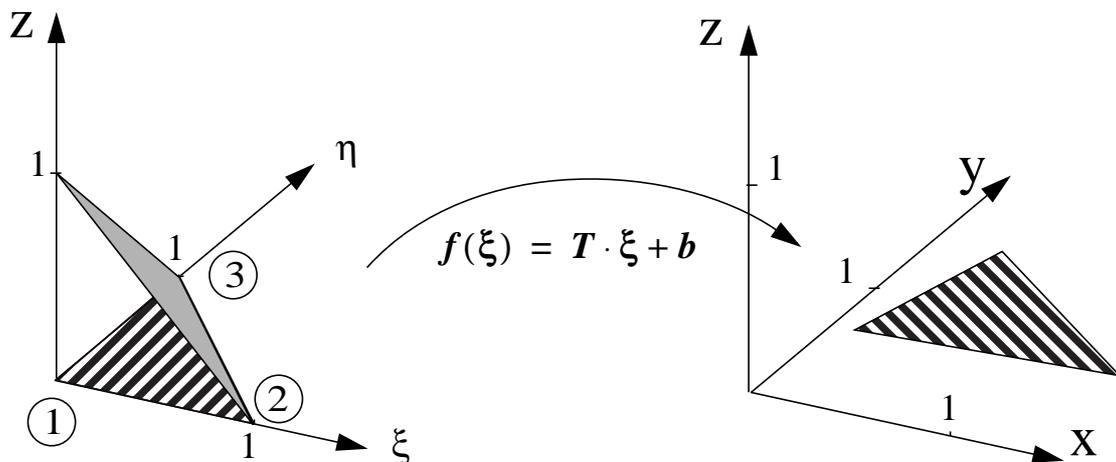
By applying the galerkin method 3.1.4, the next step would be, according to 3.1.5, choosing a discretization of the domain, a shape regular triangulation for example. Moreover we have to fix the finite dimensional subspace of the solution space  $V$  to be the space of piecewise polynomials, that is,

$$V_h = \left\{ T_h \in C^0(\Omega), T_h|_T \in \mathbb{P}_1, T_h|_{\Gamma_D} = T_0 \right\}, \quad (3.76)$$

and thus each polynomial is a linear function on each element. A convenient way for practical implementation issues of finite elements is first to transform an element to a reference element, in the two dimensional simplicial case this is the triangle  $T((0, 0), (1, 0), (0, 1))$ , sketched in Figure 3.5., by virtue of the affine mapping

$$f(\xi) = T \cdot \xi + b = x. \quad (3.77)$$

Here we have merged the coordinates in the reference frame into  $\xi = (\xi, \eta)$  and



**Figure 3.5** The reference triangle  $\hat{T}$  of a triangular finite element mesh (striped) and the shape function  $\hat{N}_1$  (dotted). The mapping  $f$  acts from the reference triangle to the triangle considered. Additionally we have numbered the nodes in the reference triangle on the left hand side. For each of the nodes a shape function is clearly defined (3.78).

analogously in the mapped domain. The local basis functions on the reference element for  $V_h$  must satisfy (3.63) and therefore read

$$\begin{aligned}\hat{N}_1(\xi) &= 1 - \xi - \eta \\ \hat{N}_2(\xi) &= \xi \\ \hat{N}_3(\xi) &= \eta.\end{aligned}\tag{3.78}$$

The contribution of triangle  $T$  to the  $ij$ -component of the system matrix has to be evaluated as

$$\begin{aligned}K_{ij}^{(T)} &= \int_{\hat{T}} \nabla(\hat{N}_i \circ f^{-1}) \cdot \kappa \cdot \nabla(\hat{N}_j \circ f^{-1}) |\text{Det} Df| d\xi d\eta \\ &= \int_T \nabla N_i(\mathbf{x}) \cdot \kappa \cdot \nabla N_j(\mathbf{x}) dx dy,\end{aligned}\tag{3.79}$$

where we have made use of the integral transformation theorem which holds since  $f$  is diffeomorphic [115]. Each component of the global stiffness matrix then is the sum over all elements for which the  $i$ -th and  $j$ -th nodal shape function's support intersection does not vanish

$$K_{ij} = \sum_{\{T, \text{supp}(N_i) \cap \text{supp}(N_j) \neq \emptyset\}} K_{ij}^{(T)}. \quad (3.80)$$

The right hand side  $\mathbf{F}$  of the algebraic equation that has to be set up to form a vector and is determined by the distribution of the heat source  $f$

$$F_i = \sum_{\{T, \text{supp}(N_i) \neq \emptyset\}} \int_T f(\mathbf{x}) N_i(\mathbf{x}) d\mathbf{x} \quad (3.81)$$

plus the contribution due to the prescribed flux on a part of the boundary

$$\tilde{F}_i = F_i + \int_{\partial\Gamma_N \cap \text{supp}(N_i)} g_N(\mathbf{x}) N_i(\mathbf{x}) d\Gamma. \quad (3.82)$$

Thus, the linear system to be solved reads

$$\mathbf{K} \cdot \mathbf{c} = \tilde{\mathbf{F}}, \quad (3.83)$$

where  $\mathbf{c}$  consists of the approximated nodal temperature values and thus represents the solution of our discrete problem. There are various methods of solving linear systems, ranging from classical Gaussian elimination algorithms (slow and memory consuming) to the more sophisticated iterative methods such as preconditioned conjugate gradient method (pcg) or the generalized minimal residual [81],[82] while the discussion of the advantages of linear system solvers is beyond the scope of this work. Yet, since the order of the equation system increases with the number of nodes, it is mandatory to bear in mind that computational resources always are limited, and, especially for very large systems equally distributed computational nodes would be too costly. The overall temperature field is, not surprisingly, locally linear (since the shape functions have been chosen in that way) and therefore can be formulated as

$$T_h(\mathbf{x}) = \sum_{i=1}^{\text{Number of Nodes}} c_i N_i(\mathbf{x}). \quad (3.84)$$

We want to point out that by referring to 3.1.2, equation (3.70) does not allow for discontinuities in the heat conductivity tensor  $\kappa$ . The reason is that the divergence operator requires global differentiability of the heat flux  $\kappa \cdot \nabla T$ , a condition, which of course, for a discontinuous heat conductivity is violated. Formulation (3.75) therefore is the only setting by which composite materials can be properly modeled.

#### 3.2.2 Multi-Layer Plate Models

Mechanical models derived from theories where the dimensionality of the problem has been reduced from three to two are called plate models. The general assumption here is that when a solid's lateral dimension exceeds its thickness by far, in other words, a high aspect ratio is present within the structure, then the effects along the thickness dimension can be neglected. Among others [27],[28], a key virtue of these lower-dimensional theories is their far better amenability to numerical computations. A vast literature has been flourished proposing plate theories, and their justifications. While some numerical approximations are by now on essentially safe theoretical grounds, there remains an abundance of challenging open problems mainly concerning existence results for various plate equations [28]. For a detailed survey of the matter we refer to the monographs of Ciarlet given in the reference list.

As in section 3.2.1 we introduce a weak or variational formulation of three dimensional elasticity based on the equations derived in chapter 2. In particular, we consider the time independent and linearized equation (2.59)

$$-\nabla \cdot \boldsymbol{\sigma} = \mathbf{f} \quad (3.85)$$

having reassembled the constitutive relationships into the overall mechanical stress  $\boldsymbol{\sigma}$ . We now apply the procedure developed in section 3.1.3 for the stress field for which a vector-valued differential equation holds. We identify the differential equation (3.85) with (3.27) in each of its components and fix a test function space coinciding with the solution space as in (3.30) to be

$$X = \{\delta\boldsymbol{\varepsilon} \in (H^0(\Omega))^3 : \delta\boldsymbol{\varepsilon} \text{ satisfies all essential boundary conditions}\}. \quad (3.86)$$

The variable  $\delta\boldsymbol{\varepsilon}$  denotes the variational strain and emerges in a natural way when considering the deformation virtual work of an elastic body caused by internal stresses [68]. Furthermore, the primitive of the strain is the displacement field itself such that the former only requires to be integrable, i.e.,  $\in H^0(\Omega) = L^2(\Omega)$ . Allowing surface traction on a part of the boundary

$$\boldsymbol{\sigma} \cdot \mathbf{n} = \mathbf{f}_s \quad \text{on } \Gamma_N \quad (3.87)$$

as a set of natural boundary conditions like in (3.28) and integrating (3.85) after multiplying it with a test function belonging to (3.86) results in a variational boundary problem analogously to (3.42)

$$a(\boldsymbol{\sigma}, \delta\boldsymbol{\varepsilon}) = \langle \mathbf{f}, \delta\mathbf{u} \rangle + \int_{\Gamma_N} \mathbf{f}_s \delta\mathbf{u} d\Gamma, \quad (3.88)$$

the second addend denoting a surface integral along the boundary where surface traction is specified. This weak formulation of electro-thermo-mechanics is taken as the starting point for the derivation of our plate model by re-introducing the non-linear relations such that (3.88) turns into the non-linear virtual work

$$W_E = (\boldsymbol{\sigma}, \delta\boldsymbol{\varepsilon}) = \langle \mathbf{f}, \delta\mathbf{u} \rangle + \int_{\Gamma_N} \mathbf{f}_s \delta\mathbf{u} d\Gamma. \quad (3.89)$$

Our plate models are obtained as the asymptotic case for one of the dimensions of the solid vanishing. In the asymptotic case for plates, the unknown field is assumed, *a priori*, to have a polynomial dependence with respect to the transverse variable  $z$ . The degree of this polynomial is specified but does not have to be the same for all components of the unknown field. In general, each component is interpolated using a finite set of linearly-independent functions as in the ansatz

$$w_{3D}(\mathbf{x}, z) = \sum_i w_i(\mathbf{x}) v_i(z). \quad (3.90)$$

Across the plate thickness, the field is interpolated with the set of shape functions  $v_i$ . Their coefficients  $w_i$  vary depending on the position  $\mathbf{x}$  of their projection on the middle-plate surface. Increasing the number of these functions (i.e., increasing the degree in the case of polynomials) yields a hierarchy of models of increasing order, whence the name hierarchic plate theories [24], in a very similar fashion to hierarchic finite elements [105]. The Kirchhoff-Love theory, [14],[98],[99], represents the lowest level of this hierarchy, as the transverse displacement  $w$  is kept constant across the thickness and the in-plane displacement is taken with a linear dependency on  $z$ . Additionally, it is necessary to satisfy the constraint that the coefficient of  $z$  be the opposite of the gradient of  $w$ , expressing the geometrical statement that lines normal to the middle plane keep their orthogonality in the deformed state. The ansatz for the three-dimensional displacement is

$$\mathbf{u}_{3D} = \begin{bmatrix} \mathbf{u}(\mathbf{x}) - z\nabla w(\mathbf{x}) \\ w(\mathbf{x}) \end{bmatrix}. \quad (3.91)$$

For Reissner-Mindlin plates, the through-thickness behavior is assumed as given by

$$\mathbf{u}_{3D} = \begin{bmatrix} \mathbf{u}(\mathbf{x}) - z\vartheta(\mathbf{x}) \\ w(\mathbf{x}) \end{bmatrix}, \quad (3.92)$$

which differs from the Kirchhoff-Love model, [15],[100],[101],[102], in the independence of the  $z$ -coefficient of the in-plane displacement. The Reissner-Mindlin-based plate model has received much attention because, contrary to Kirchhoff plate theory,  $C^0$ -continuity is sufficient to formulate finite element discretizations. This approximation requires a reduced-order integration to retain proper flexibility for the plate. This means that, instead of using a Gaussian quadrature over the element, where the order of the quadrature is chosen to give exact results for the highest polynomial degree of the shape functions, a quadrature that guarantees exact results only for a lower polynomial degree is chosen, thus suppressing higher-order displacement behaviour. Indeed, without uniform or selective reduced integration, elements incur “locking”, a phenomenon in which the normal-integrated  $C^0$ -element fails to reproduce the Kirchhoff solution expected in the thin-plate limit. However, wider computing experience with the reduced-

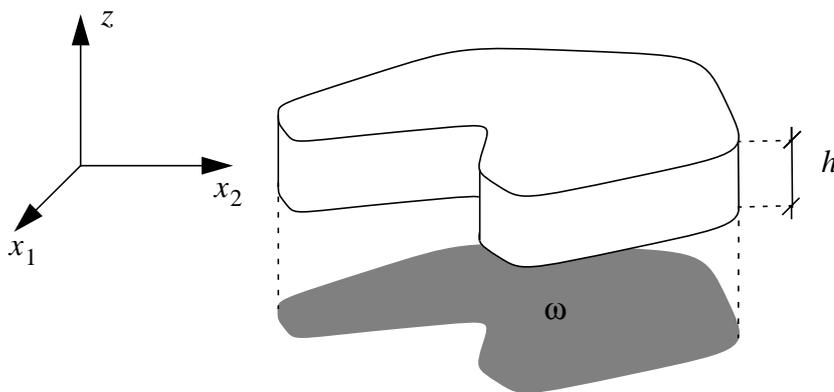
integration elements disclosed ill-conditioned behavior that is erratically dependent on element shapes and mesh patterns, which is an undesirable characteristic for elements intended for use in general-purpose software. Consequently, a rich literature has flourished suggesting several methods to overcome these problems. On the other hand, the common place that the Reissner-Mindlin theory is “better” than the Kirchhoff-Love theory, is not yet fully substantiated [28].

Another significant argument to choose the Kirchhoff-Love model is the fact that the expressions for the displacement field can be found without resorting to the Kirchhoff-Love hypothesis mentioned above, which is actually an *a priori* assumption of geometrical nature. This is achieved by applying asymptotic methods and expanding the three-dimensional solution in a formal power series [27].

Therefore, when using the more sophisticated finite elements of class  $C^1$  applied to the simpler Kirchhoff-Love theory which is mathematically sounder, a higher degree of reliability of the simulated results can be expected. Additionally, a conforming ansatz for this model has the advantage that error estimators are much simpler to formulate and thus adaptive computations are much faster and more effective in terms of memory requirement. This is in direct contrast to the supposedly efficient non-conforming approaches where the treatment of consistency errors becomes necessary [44],[46].

### THE 3D THEORY IN THE LANGUAGE OF PLATES

Describing the three-dimensional geometry of the plate as in Fig. 3.6, or



**Figure 3.6** The three-dimensional geometry of a plate

### 3 Accuracy Control for thin structure simulation

---

$$\Omega = \Omega_{3D} = \omega \times [-h/2, h/2] = \{(\mathbf{x}, z) | \mathbf{x} \in \omega, -h/2 \leq z \leq h/2\}, \quad (3.93)$$

the  $L^2$  product for an arbitrary pair of fields  $u, v$  on the solid plate is decomposed as

$$(u, v)_{\Omega_{3D}} = \int_{-h/2}^{h/2} (u, v)_{\omega} dz. \quad (3.94)$$

Henceforth the subscript  $\omega$  for the inner product and the integration extrema  $\pm h/2$  shall be omitted where there is no ambiguity. The boundary of the solid plate is decomposed as

$$\partial\Omega = \partial\Omega_{3D} = \omega \times \{-h/2\} \cup \omega \times \{h/2\} \cup \partial\omega \times [-h/2, h/2], \quad (3.95)$$

i.e., into its lower and upper planes and its perimeter wall. For any tensor field  $A(\mathbf{x}, z)$  defined on the solid plate, we define its  $n$ -th order moment by

$$A_n(\mathbf{x}) = \int z^n A(\mathbf{x}, z) dz. \quad (3.96)$$

For a body force decomposed in its transverse and in-plane components as  $\mathbf{f}_{3D} = (\mathbf{f}, g)$  the virtual work reads

$$\begin{aligned} W_B &= (\mathbf{f}_{3D}, \delta\mathbf{u}_{3D})_{\Omega_{3D}} \\ &= (\mathbf{f}, \mathbf{v} - z\nabla v)_{\Omega_{3D}} + (g, v)_{\Omega_{3D}} \\ &= (\int \mathbf{f} dz, \mathbf{v}) - (\int z\mathbf{f} dz, \nabla v) + (\int g dz, v) \\ &= (\mathbf{f}_0, \mathbf{v}) - (\mathbf{f}_1, \nabla v) + (g_0, v). \end{aligned} \quad (3.97)$$

where we have introduced the variation of the displacement field  $\delta\mathbf{u}_{3D} \equiv (v_1, v_2, v) \equiv (\mathbf{v}, v)$ . Recall that the (Green-St. Venant) three-dimensional strain is defined as

$$\boldsymbol{\varepsilon} = (\nabla\mathbf{u})^S + \frac{1}{2}(\nabla\mathbf{u} \cdot (\nabla\mathbf{u})^T). \quad (3.98)$$

Using the displacement decomposition, we rewrite (regroup) the strain as

$$\boldsymbol{\varepsilon}_{3D} = \begin{bmatrix} \boldsymbol{\varepsilon} & \boldsymbol{\gamma}/2 \\ \boldsymbol{\gamma}^T/2 & \varepsilon_{33} \end{bmatrix}, \quad (3.99)$$

where the terms are defined as follows

$$\boldsymbol{\varepsilon} = (\nabla \mathbf{u})^S + (\nabla w)(\nabla w)/2 - z\nabla\nabla w + (\nabla \mathbf{u}) \cdot (\nabla \mathbf{u})^T/2 - (z\nabla\nabla w)^2 \quad (3.100)$$

$$\boldsymbol{\gamma} = (\nabla \mathbf{u} - z\nabla\nabla w) \cdot \nabla w \quad (3.101)$$

$$\varepsilon_{33} = (\nabla w)^2. \quad (3.102)$$

This decomposition is useful in keeping the notation compact, and aids in the interpretation of terms later on.

### TAKING THE PLATE LIMIT

In the limit case, where the plate thickness vanishes, the in-plane strain reduces to

$$\boldsymbol{\varepsilon} = (\nabla \mathbf{u})^S + (\nabla w)(\nabla w)/2 - z\nabla\nabla w \quad (3.103)$$

which, at the plate's middle plane, gives

$$\boldsymbol{\varepsilon}_m = (\nabla \mathbf{u})^S + (\nabla w)(\nabla w)/2. \quad (3.104)$$

The variational field of the in-plane strain is

$$\delta \boldsymbol{\varepsilon} = (\nabla \mathbf{v})^S + (\nabla w)(\nabla v) - z\nabla\nabla v. \quad (3.105)$$

We now impose the additional condition that the elastic reaction vanishes in the transverse direction of the plate. This implies that only the in-plane components of the stress can be different from zero, so that we may write that

$$\boldsymbol{\sigma}_{3D} = \begin{bmatrix} \boldsymbol{\sigma} & 0 \\ 0 & 0 \end{bmatrix}. \quad (3.106)$$

A tensor reduction here has to be taken into account such that a reduced constitutive relation, see A.2, holds

$$\boldsymbol{\sigma} = \mathbf{A} : \hat{\boldsymbol{\varepsilon}} \quad (3.107)$$

which becomes clear when rearranging the tensor components in matrices by means of a mapping often referred to as the engineering notation (2.34),

$$\begin{bmatrix} \boldsymbol{\sigma}(1, 2) \\ \boldsymbol{\sigma}(\cdot, 3) \end{bmatrix} = \begin{bmatrix} \mathbf{C}_{11} & \mathbf{C}_{12} \\ \mathbf{C}_{21} & \mathbf{C}_{22} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\varepsilon}}_1 \\ \hat{\boldsymbol{\varepsilon}}_2 \end{bmatrix}. \quad (3.108)$$

Condition (3.106) requires the components of  $\boldsymbol{\sigma}$  which have an index ‘3’ to vanish and therefore the plane stress  $\boldsymbol{\sigma}(1, 2)$  in the first set of equations in (3.108) can be expressed in terms of the in plane strain  $\hat{\boldsymbol{\varepsilon}}_1$  only. The reduced elastic tensor  $\mathbf{A}$  then is composed by the  $\mathbf{C}_{ij}$  sub-matrices. We give a more detailed exposition of tensor reduction in the appendix. Unless otherwise stated the following tensor quantities denote those for the reduced case. Next we form the elastic virtual work which coincides with the previously defined bilinear form in case linear relations are assumed, using the definitions (3.103), (3.105) and (3.106), to get

$$\begin{aligned} W_E &= (\boldsymbol{\sigma}_{3D}, \delta \boldsymbol{\varepsilon}_{3D})_{\Omega_{3D}} \\ &= (\boldsymbol{\sigma}, (\nabla \mathbf{v})^S)_{\Omega_{3D}} - (z \boldsymbol{\sigma}, \nabla \nabla \mathbf{v})_{\Omega_{3D}} + (\nabla w \cdot \boldsymbol{\sigma}, \nabla \mathbf{v})_{\Omega_{3D}} \\ &= \left( \int \boldsymbol{\sigma} \, dz, (\nabla \mathbf{v})^S \right) - \left( \int z \boldsymbol{\sigma} \, dz, \nabla \nabla \mathbf{v} \right) + (\nabla w \cdot \int \boldsymbol{\sigma} \, dz, \nabla \mathbf{v}) \\ &= (\boldsymbol{\sigma}_0, (\nabla \mathbf{v})^S) - (\boldsymbol{\sigma}_1, \nabla \nabla \mathbf{v}) + (\nabla w \cdot \boldsymbol{\sigma}_0, \nabla \mathbf{v}), \end{aligned} \quad (3.109)$$

which, by introducing the in-plane stress-strain relation,

$$\boldsymbol{\sigma} = \mathbf{A} : \boldsymbol{\varepsilon} + \boldsymbol{\sigma}^{\text{ext}} \quad (3.110)$$

where  $\mathbf{A}$  is the reduced elastic tensor and  $\boldsymbol{\sigma}^{\text{ext}}$  the externally impressed prestress, yields

$$\begin{aligned}
 W_E = & (\mathbf{A}_0 : \boldsymbol{\varepsilon}_m - \mathbf{A}_1 : \nabla \nabla w + \boldsymbol{\sigma}_0^{\text{ext}}, (\nabla \mathbf{v})^S) \\
 & - (\mathbf{A}_1 : \boldsymbol{\varepsilon}_m - \mathbf{A}_2 : \nabla \nabla w + \boldsymbol{\sigma}_1^{\text{ext}}, \nabla \nabla v) \\
 & (\nabla w \cdot (\mathbf{A}_0 : \boldsymbol{\varepsilon}_m - \mathbf{A}_1 : \nabla \nabla w + \boldsymbol{\sigma}_0^{\text{ext}}), \nabla v) .
 \end{aligned} \tag{3.111}$$

### LINEARIZATION

The elastic virtual work (3.111) is non-linear. This means that the variational problem resulting by adding it to the body force virtual work is also a non-linear expression. The linearization acts on the in-plane strain at the middle surface,  $\boldsymbol{\varepsilon}_m$ , and on the third row of equation (3.111):

$$\begin{aligned}
 W_{E, \text{lin}} = & (\mathbf{A}_0 : (\nabla \mathbf{u})^S - \mathbf{A}_1 : \nabla \nabla w + \boldsymbol{\sigma}_0^{\text{ext}}, (\nabla \mathbf{v})^S) \\
 & - (\mathbf{A}_1 : (\nabla \mathbf{u})^S - \mathbf{A}_2 : \nabla \nabla w + \boldsymbol{\sigma}_1^{\text{ext}}, \nabla \nabla v) + (\nabla w \cdot \boldsymbol{\sigma}_0^{\text{ext}}, \nabla v) .
 \end{aligned} \tag{3.112}$$

The resulting weak formulation for the mechanical plate is then given by equating the elastic and body-force (or external) virtual work terms

$$W_E = W_B. \tag{3.113}$$

The linearity of (3.113) depends on the elastic virtual work  $W_E$  only. Explicitly, the linearized virtual work is

$$\begin{aligned}
 & (\nabla w \cdot \boldsymbol{\sigma}_0^{\text{ext}}, \nabla v) + (\mathbf{A}_0 : (\nabla \mathbf{u})^S - \mathbf{A}_1 : \nabla \nabla w + \boldsymbol{\sigma}_0^{\text{ext}}, (\nabla \mathbf{v})^S) \\
 & - (\mathbf{A}_1 : (\nabla \mathbf{u})^S - \mathbf{A}_2 : \nabla \nabla w + \boldsymbol{\sigma}_1^{\text{ext}}, \nabla \nabla v) = \langle \mathbf{f}_0, \mathbf{v} \rangle - \langle \mathbf{f}_1, \nabla v \rangle + \langle g_0, v \rangle.
 \end{aligned} \tag{3.114}$$

The left-hand side (LHS) of (3.114) forms a bilinear form in the space

$$X = H^1(\omega) \times H^1(\omega) \times H^2(\omega) \tag{3.115}$$

such that  $\mathbf{u}$  belongs to  $H^1(\omega) \times H^1(\omega)$  and  $w$  belongs to  $H^2(\omega)$ , as will be detailed below. So far, we have only considered constant external stress  $\boldsymbol{\sigma}^{\text{ext}}$ . When disassembling the extended constitutive relation for the stress in (3.85)

$$\boldsymbol{\sigma} = \mathbf{C} : \boldsymbol{\varepsilon} - \boldsymbol{\pi}^T \cdot \mathbf{E} + \alpha \Delta T \quad (3.116)$$

which also can be subjected to the tensor reduction detailed in the appendix we can rewrite the plane stress (3.107) in terms of

$$\boldsymbol{\sigma} = \mathbf{A} : \hat{\boldsymbol{\varepsilon}} - \hat{\boldsymbol{\pi}}^T \cdot \mathbf{E} + \hat{\alpha} \Delta T. \quad (3.117)$$

The weak plate equations (3.114) therefore can be extended, using  $\mathbf{E} = -\nabla\phi$ , to

$$\begin{aligned} & (\nabla w \cdot (\hat{\boldsymbol{\pi}}^T \cdot \nabla\phi + \hat{\alpha}\Delta T)_0, \nabla v) \\ & + (\mathbf{A}_0 : (\nabla \mathbf{u})^S - \mathbf{A}_1 : \nabla \nabla w + (\hat{\boldsymbol{\pi}}^T \cdot \nabla\phi)_0 + (\hat{\alpha}\Delta T)_0, (\nabla v)^S) \\ & - (\mathbf{A}_1 : (\nabla \mathbf{u})^S - \mathbf{A}_2 : \nabla \nabla w + (\hat{\boldsymbol{\pi}}^T \cdot \nabla\phi)_1 + (\hat{\alpha}\Delta T)_1, \nabla \nabla v) \\ & = \langle \mathbf{f}_0, \mathbf{v} \rangle - \langle \mathbf{f}_1, \nabla v \rangle + \langle g_0, v \rangle \end{aligned} \quad (3.118)$$

if there are no additional sources considered such as charge sources or heat sources. However, this equation is only the mechanical part of the coupled physical problem. The purely thermal part has been discussed in 3.2.1 whereas the electrical part has to be set up according to the procedure in section 3.1.3 using the generalized Gauss law (2.64). Besides the purely electrostatical terms the direct piezo-electric contribution has to be considered as in (2.60) which with the linearized in-plane strain (3.103) and using the reduced quantities (see Appendix A.2) yields

$$\mathbf{D} = \hat{\boldsymbol{\pi}} : ((\nabla \mathbf{u})^S - z \nabla \nabla w) + \hat{\boldsymbol{\chi}} \cdot \mathbf{E} \quad (3.119)$$

such that another bilinear form can be defined

$$\begin{aligned} & a(\boldsymbol{\psi}, \mathbf{D})_\Omega \\ & = (\nabla \boldsymbol{\psi}, \hat{\boldsymbol{\pi}} : (\nabla \mathbf{u})^S - z \hat{\boldsymbol{\pi}} : \nabla \nabla w - \hat{\boldsymbol{\chi}} \cdot \nabla \boldsymbol{\psi})_\Omega \end{aligned} \quad (3.120)$$

having neglected the electrocaloric terms and thus the contribution of a temperature change to the electrical displacement. The right hand side linear functional comprises the charge density and some prescribed electrical displacement on a part of the surface  $\mathbf{n} \cdot \mathbf{D} = h$

$$\langle \tilde{l}, \psi \rangle = \int_{\Omega} \rho \psi dx + \int_{\Gamma_N} h \psi d\Gamma. \quad (3.121)$$

It is important to note that for the extended Gauss law integration is taken over the whole three-dimensional domain whereas the mechanical part is fully restricted to the plate's middle plane. The reason is obvious since one should be able to specify boundary conditions of electrical nature also on the top and the bottom face of the plate. Merging the displacement fields, the temperature and the electrostatic potential together into one field  $\mathbf{U} = (\mathbf{u}, w, T, \phi)$  together with its variation  $\mathbf{V} = (\mathbf{v}, v, S, \psi)$  we may rewrite the linear plate equations in the form

$$a(\mathbf{U}, \mathbf{V}) = \langle \tilde{l}, \mathbf{V} \rangle, \quad (3.122)$$

where on the left hand side all terms we collected from the linearized forms of (3.73), (3.118) and (3.120) in order to define a new extensive bilinear form. In detail it reads

$$\begin{aligned} & ((\hat{\boldsymbol{\pi}} \cdot \nabla \phi + \hat{\boldsymbol{\alpha}} \Delta T)_0, \nabla v)_{\omega} \\ - & (\mathbf{A}_0 : (\nabla \mathbf{u})^S - \mathbf{A}_1 : \nabla \nabla w + (\hat{\boldsymbol{\pi}}^T \cdot \nabla \phi)_0 + (\hat{\boldsymbol{\alpha}} \Delta T)_0, (\nabla \mathbf{v})^S)_{\omega} \\ + & (\mathbf{A}_1 : (\nabla \mathbf{u})^S - \mathbf{A}_2 : \nabla \nabla w + (\hat{\boldsymbol{\pi}}^T \cdot \nabla \phi)_1 + (\hat{\boldsymbol{\alpha}} \Delta T)_1, \nabla \nabla v)_{\omega} \\ + & (\nabla \psi, \hat{\boldsymbol{\pi}} : (\nabla \mathbf{u})^S - z \hat{\boldsymbol{\pi}} : \nabla \nabla w - \hat{\boldsymbol{\chi}} \cdot \nabla \phi)_{\Omega} \\ + & (\nabla T, \boldsymbol{\kappa} \cdot \nabla S)_{\Omega} \\ = & \langle \mathbf{f}_0, \mathbf{v} \rangle_{\omega} - \langle \mathbf{f}_1, \nabla v \rangle_{\omega} + \langle g_0, v \rangle_{\omega} \\ & + \langle \rho, \psi \rangle_{\Omega} + \langle f, S \rangle_{\Omega} + \langle h, \psi \rangle_{\Gamma_{N, electrical}} + \langle g_N, S \rangle_{\Gamma_{N, thermal}} \end{aligned} \quad (3.123)$$

where the right hand side represents the source terms as well as the natural boundary conditions. The temperature difference contains the temperature field by means of  $\Delta T = T - T_0$ ,  $T_0$  being a reference temperature. The first three addends on both sides represent an integral extending over the two dimensional domain whereas the others extend over the whole three dimensional domain. The space in which problem (3.123) is posed is given by

$$X = \{ \mathbf{U} \in (H^1(\omega))^2 \times H^2(\omega) \times H^1(\Omega) \times H^1(\Omega), \mathbf{U} \text{ satisfies all essential BCs} \}. \quad (3.124)$$

Obviously, the in-plane displacement field requires only that its first derivatives be square integrable whereas the out-of-plane displacement field requires that also its second derivatives be square integrable and therefore belong to  $H^2(\omega)$ . As it is well known, the temperature and the electrostatic field obey second order differential equations and thus, as we have detailed in 3.1.3, belong to  $H^1(\Omega)$ . Existence and uniqueness of the solution  $\mathbf{U}$ , which has to be sought for is guaranteed by virtue of the ellipticity and continuity of the sub-problems.

For the linear problem (3.88), the former can be proven by Korn's inequality which holds as soon as a pure rigid body motion is excluded by applying Dirichlet boundary conditions on a part of the boundary that has not measure zero [77]. For the slightly more complicated (pure) plate case the solution's existence is proved in [103] and that of the coupled piezoelectrical plate case for example in [30]. For the second order problems treating electrostatics and thermo-statics the well known Poincaré -Friedrichs inequality is used to show the existence of solutions, see [76] or [77].

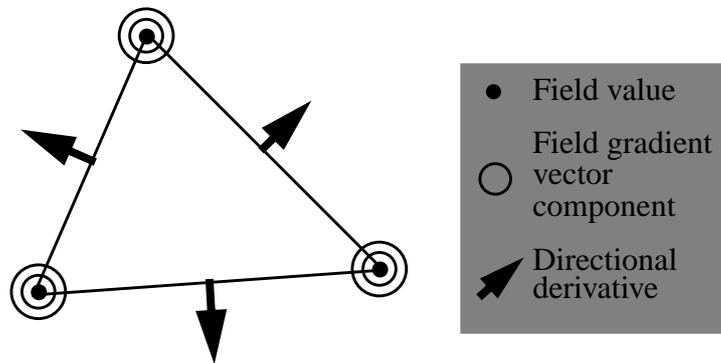
#### **FINITE ELEMENTS FOR THERMOMECHANICAL PLATES**

Equation (3.114) is discretized using finite elements by interpolating the fields  $\mathbf{u}$  and  $w$  with a set of shape functions, and according to the Galerkin method, using each of these shape functions as the test functions  $\mathbf{v}$  and  $v$ . For the transverse displacement the unknown field is interpolated as

$$\mathbf{u}(\mathbf{x}) = \sum_i u_i N_i(\mathbf{x}) \quad . \quad (3.125)$$

As we have seen, the term  $(A_2: \nabla \nabla w, \nabla \nabla v)$ , containing the transverse displacement requires that second derivatives also must be square integrable, i.e.,  $w \in H^2(\omega)$ . In order to have a conforming discretization, the shape functions must also be elements of  $H^2(\omega)$  and thus the shape functions must be of class  $C^1(\omega)$  according to (3.64). This can be achieved by implementing the Argyris element [104], see Fig. 3.7.

The Argyris element can be represented by a fifth order polynomial and is also used to interpolate the in-plane displacement fields, although the solution has to be found in the "less restrictive" space of (3.115). A review of the Argyris element is given in [106]. In case an additional temperature field has to be computed

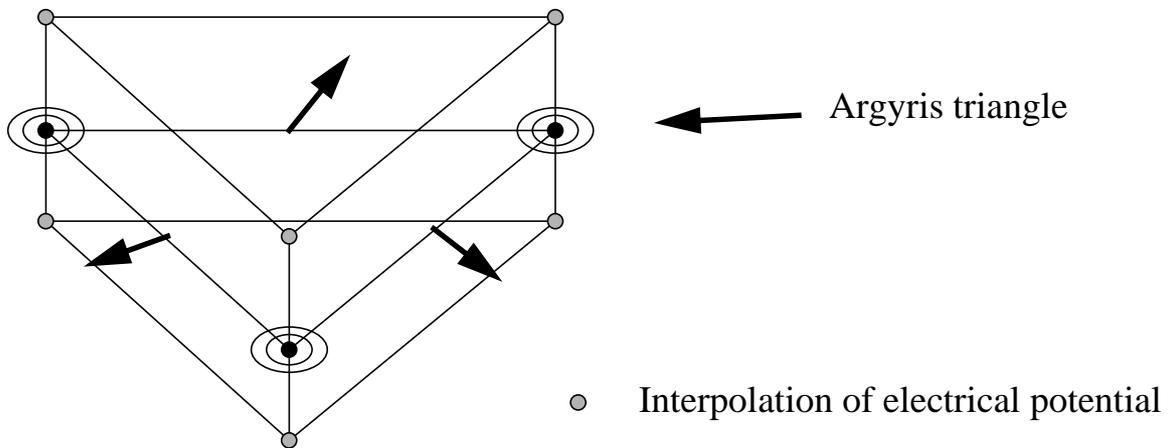


**Figure 3.7** The Argyris Element [104]

(neglecting electrostatic effects) the plate equations (3.123) can be decoupled in that sense that the mechanical problem can be solved after a solution for the temperature field has been obtained. We would like to point out that thermal effects across the thickness are neglected, see Appendix, and therefore the temperature field can be interpolated on the same 2D domain as the mechanical displacement fields. As usual when treating a dimensionally reduced thermal problem, the zeroth order moment of the (reduced) heat conductivity tensor  $\kappa_0$  has to be considered.

### FINITE ELEMENTS FOR PIEZO-ELECTRICAL PLATES

The situation turns more delicate when an electrical field comes into play. Inspecting equation (3.123), we realize that this field is defined over the full three-dimensional domain whereas the mechanical field is defined on some midplane. A new element therefore has been proposed [59],[60] which is called the Argyris prism. It takes into account that each of the displacement fields are interpolated continuously differentiable on a two dimensional domain whereas the electrostatic field has to be interpolated only continuously, but on a three dimensional domain. Figure 3.8 shows the setup of this combined element. Since the plate model allows for the treatment of multi-layer stacks we make the arrangement that for a given composite multilayer structure the prisms' electrical nodes are located in a plane for both the top and the bottom surface. The height of all prisms is determined by the largest stack extension present within the multilayer plate. The nodes for the electrical field for that largest stack then coincide with its vertical margins, as sketched in Figure 3.9.



**Figure 3.8** The Argyris Prism finite element [59]. An Argyris triangle is placed exactly on the half way through the thickness of the structure. The electrical field is placed at the outmost vertical ends of the structure and thus allows for the specification of boundary conditions that differ at either the top or the bottom.

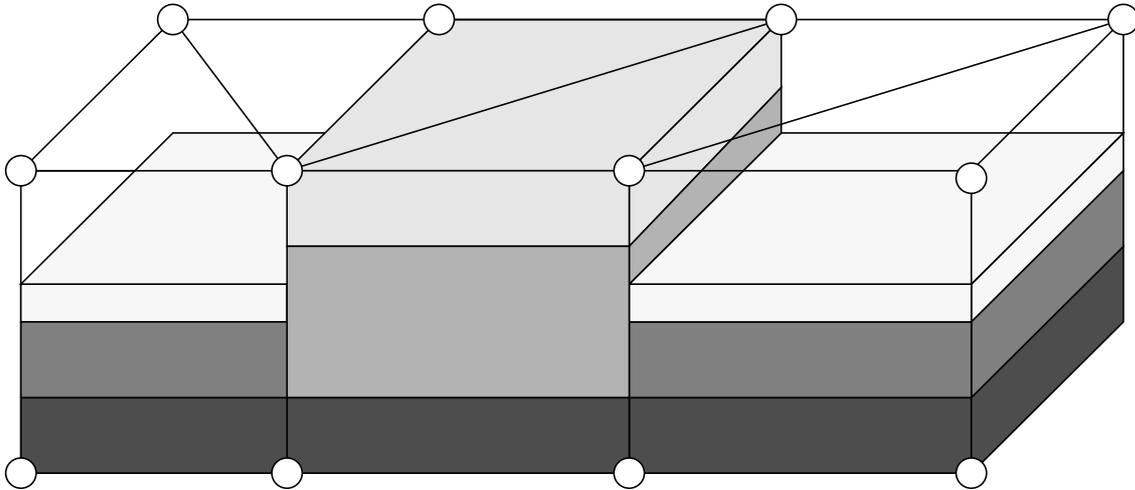
Again, we can define a reference prism with the nodes and the shape functions as listed in Table 3.1. The polynomial then to be built for each prism then is of the

**Table 3.1** Shape functions for the reference prism  $\hat{P}$ .

<i>Node Number</i>	<i>Node coordinates</i>	<i>Shape function</i>
0	(0, 1, -1)	$\frac{1}{2}(1-x-y)(1-z)$
1	(1, 0, -1)	$\frac{1}{2}x(1-z)$
2	(0, 1, -1)	$\frac{1}{2}y(1-z)$
3	(0, 0, 1)	$\frac{1}{2}(1-x-y)(1+z)$
4	(1, 0, 1)	$\frac{1}{2}x(1+z)$

general shape

$$p(x, y, z) = a_0 + a_1x + a_2y + a_3z + a_4xz + a_5yz. \quad (3.126)$$



**Figure 3.9** Arrangement of the computational nodes in prisms which interpolate the electrical field. The prismatic mesh stretches over a composite multilayer structure which is typical in MEMS. In case layerstacks occur that have different vertical dimensions i.e. are variably high, the prisms have to be ‘filled’ with the air gaps. The middle plane of the structure where the displacement fields are interpolated is not shown for the sake of clarity.

### 3.3 Error Estimation for second order problems

#### 3.3.1 Abstract error estimation

The central difficulty that one has to face when obtaining solutions by numerical methods such as those presented up to now is the fact that these solutions are nothing but approximations. One could overcome this problem by simply approaching the continuous world, but the discrete structure of the computer can not be circumvented and moreover, in contrast to mathematically taking the limit, computer resources in fact *are* limited. To be on the safe side all the same, errors can be

computed and reduced at least locally. To make the point clear we reconsider the difference of two fields, the first field defined by (3.43), the weak form expressed by means of a bilinear form and a linear functional and the second field represented by its discretized form (3.45)

$$\| \| u - u_h \| \| . \quad (3.127)$$

This is an error, a measure of how good the approximation and thus the computed solution is. The way in which we measure the error, namely by an integral norm, emerges from the fact that we want to obtain information about how the error is distributed over the simulation domain. Disposing of such a distribution one can decide where the domain geometry should be modified. The smallest units where the pointwise errors can be averaged are the elements  $T$  themselves such that a local element error estimate might be stated as

$$\| \| u - u_h \| \|_T \leq \eta_T(u_h) . \quad (3.128)$$

What we would like to find is an expression as on the right hand side, *depending only on the computed solution itself* which bounds the error. For the derivation of such an a posteriori error estimation we use section 3.1 and start with a given continuous, elliptic and symmetric bilinearform  $a$  (the left hand side of the weak form) and a Hilberspace  $X$  with scalar product  $( \cdot , \cdot )_X$  and the corresponding Norm

$$\| v \|_X = \sqrt{(v,v)_X} . \quad (3.129)$$

For  $X$ , the space the weak solution has to be sought for, we can define its dual as in (3.10),  $X^* = L(X, \mathbb{R})$  with the duality pairing

$$\langle \cdot , \cdot \rangle_{X^* \times X}: X^* \times X \rightarrow \mathbb{R} \quad (3.130)$$

$$\langle l, v \rangle_{X^* \times X} = l(v) \quad \forall l \in X^*, v \in X . \quad (3.131)$$

We have seen that  $X$  has to be defined as some Sobolev space or the product space of Sobolev spaces of appropriate order. We now can define a differential operator associated to the bilinear form acting as follows:

$$L: X \rightarrow X^* \quad (3.132)$$

$$\langle Lw, v \rangle_{X^* \times X} = a(w, v) \quad \forall w, v \in X. \quad (3.133)$$

By means of the bilinear form we define a norm also known as the energy norm (and thus representing the internal energy of a system multiplied by a factor of two)

$$\|v\| = \sqrt{a(v, v)}. \quad (3.134)$$

The assumptions made on the bilinear form, namely that it be continuous (3.16) and elliptic (3.17) will enable us to deal with the different norms in a straightforward manner: the Hilbertspace norm and the energy norm then are equivalent,

$$\alpha \|v\|_X^2 \leq \|v\|^2 \leq K \|v\|_X^2. \quad (3.135)$$

The continuous FE-problem then can be formulated in the following way: for a given source function  $f \in X^*$  we are looking for the solution  $u$  in  $X$  such that

$$a(u, v) = \langle f, v \rangle_{X^* \times X} \quad \forall v \in X, \quad (3.136)$$

or, making use of (3.133),

$$Lu = f \quad \text{in } X^*. \quad (3.137)$$

The existence and uniqueness of solutions are then assured by functional analytical results such as the Lax-Milgram Theorem, for example, as we have seen in 3.1. Discretizing the problem (3.136) means choosing a finite dimensional subspace  $X_h$  of  $X$ , and looking for an  $u_h$  in  $X_h$  where

$$a(u_h, v_h) = \langle f, v_h \rangle_{X^* \times X} \quad \forall v_h \in X_h. \quad (3.138)$$

This formulation of the FE-problem forms the basis of the following abstract a posteriori error analysis. When rewriting (3.136) with  $u - u_h$  instead of  $u$  we have for any  $v \in X$  by virtue of (3.133) and the differential equation (3.137)

$$a(u - u_h, v) = \langle L(u - u_h), v \rangle = \langle f - Lu_h, v \rangle. \quad (3.139)$$

For any  $w \in X, w \neq 0$  we can write

$$\|w\| = \frac{1}{\|w\|} \|w\|^2 = \frac{1}{\|w\|} a(w, w) = a(w, \frac{w}{\|w\|}) \leq \sup_{v \in X, \|v\|=1} a(w, v). \quad (3.140)$$

Combining the last two equations yields the estimate

$$\|u - u_h\| \leq \sup_{v \in X, \|v\|=1} a(u - u_h, v) = \sup_{v \in X, \|v\|=1} \langle f - Lu_h, v \rangle. \quad (3.141)$$

Inequality (3.141) is an abstract a posteriori error estimation: only known data are required to compute an upper bound for the error  $u - u_h$  in the energy norm. The term  $f - Lu_h$  defines the residual of the strong form of the partial differential equation. Error estimation based on these techniques thus is known as residual error estimation. The aim now is to find upper bounds of the RHS of inequality (3.141) which are local, that is, may be expressed elementwise and invoke only the computed solution field and known data. How to explicitly derive a computable expression in the following is demonstrated for a second order problem.

#### 3.3.2 Error estimation for a Poisson equation

By choosing a Poisson-type equation to derive an error estimator we gain insight into a mathematical technique without demanding too much formal expenses as it would be the case for the rather complicated coupled plate equations. Later then we are able to treat the plate case just by drawing analogies to the steps which allow of it. We refer to 3.2.1 and introduce the weak form of a general Poisson problem

$$a(u, v) = \int_{\Omega} \nabla u \cdot \boldsymbol{\kappa} \cdot \nabla v \, d\mathbf{x} = \langle \tilde{f}, v \rangle = \langle f, v \rangle + \int_{\Gamma_N} g_N v \, d\Gamma \quad \forall v \in X, \quad (3.142)$$

where  $\Omega$  is a bounded domain with polygonal boundary. The part of the boundary where Dirichlet boundary conditions are specified is expected to be non-empty whereas normal fluxes  $g_N$  across the domain border are specified on the part of the boundary which is denoted as  $\Gamma_N$ . By discretizing the domain according to 3.1.5 we can state the finite element problem as

$$a(u_h, v_h) = \int_{\Omega} f v_h d\mathbf{x} + \int_{\Gamma_N} g_N v_h d\Gamma \quad \forall v_h \in X_h \quad (3.143)$$

assuming that given sources coincide with their interpolations, in other words,

$$f = f_h \quad \text{and} \quad g_N = g_{N,h}. \quad (3.144)$$

Before we apply the abstract a posteriori error estimation of the previous section to this problem, we introduce the jump of some function  $w$  along an edge  $E$  by

$$[w(\mathbf{x})] = \lim_{\delta \rightarrow 0} w(\mathbf{x} + \delta \mathbf{n}_{\Gamma}) - \lim_{\delta \rightarrow 0} w(\mathbf{x} - \delta \mathbf{n}_{\Gamma}) \quad (3.145)$$

where  $\mathbf{n}_{\Gamma}$  is a fixed outward normal of a given triangle's edge. The scalar product in the RHS of (3.141) can be recast as

$$\langle \tilde{f} - Lu_h, v \rangle = \langle \tilde{f} - Lu_h, v - v_h \rangle \quad (3.146)$$

since  $X_h \subset X$  and thus the error  $v - v_h$  is orthogonal to  $X_h$  as we have seen earlier in (3.49). Making use of the definition of  $L$  in (3.133) and the definition of  $a$  in (3.142) together with partial integration turns (3.146) into

$$\begin{aligned} & \langle \tilde{f} - Lu_h, v - v_h \rangle \\ &= \langle \tilde{f}, v - v_h \rangle - a(u_h, v - v_h) \\ &= \langle \tilde{f}, v - v_h \rangle - \int_{\Omega} \nabla u_h \cdot \boldsymbol{\kappa} \cdot \nabla (v - v_h) d\mathbf{x}. \end{aligned} \quad (3.147)$$

The reason for having introduced the term  $v - v_h$  instead of  $v$  is the fact that we have to estimate the interpolation error  $v - v_h$  in a stronger norm than in the  $L^2$ -norm which will be explained later in detail. Partial integration and domain decomposition  $S, \bar{\Omega} = \bigcup_{T \in S} T$ , then yields the expression

$$\begin{aligned} & \langle \tilde{f} - Lu_h, v - v_h \rangle \\ &= \langle \tilde{f}, v - v_h \rangle - \sum_{T \in S} \left( \int_T -\nabla \cdot \boldsymbol{\kappa} \cdot \nabla u_h (v - v_h) d\mathbf{x} + \int_{\partial T} \mathbf{n} \cdot \boldsymbol{\kappa} \cdot \nabla u_h (v - v_h) d\partial T \right). \end{aligned} \quad (3.148)$$

Using definition (3.142) and taking into account that integration over edges  $E$  in the domain interior occurs twice the above equation becomes

$$\begin{aligned}
 \langle \tilde{f} - Lu_h, v - v_h \rangle &= \sum_{T \in \mathcal{S}} \left\{ \int_T (f + \nabla \cdot \boldsymbol{\kappa} \cdot \nabla u_h)(v - v_h) dx \right. \\
 &\quad + \sum_{E \in \partial T \cap \Omega} \frac{1}{2} \int_E [\mathbf{n}_E \cdot \boldsymbol{\kappa} \cdot \nabla u_h](v - v_h) d\Gamma \\
 &\quad \left. + \sum_{E \in \partial T \cap \Gamma_N} \int_E (g_N - \mathbf{n}_E \cdot \boldsymbol{\kappa} \cdot \nabla u_h)(v - v_h) d\Gamma \right\}. \tag{3.149}
 \end{aligned}$$

Applying the Cauchy-Schwarz inequality yields

$$\begin{aligned}
 \langle \tilde{f} - Lu_h, v - v_h \rangle &\leq \sum_{T \in \mathcal{S}} \left\{ \|f + \nabla \cdot \boldsymbol{\kappa} \cdot \nabla u_h\|_{L^2(T)} \|v - v_h\|_{L^2(T)} \right. \\
 &\quad + \sum_{E \in \partial T \cap \Omega} \frac{1}{2} \|[\mathbf{n}_E \cdot \boldsymbol{\kappa} \cdot \nabla u_h]\|_{L^2(E)} \|v - v_h\|_{L^2(E)} \\
 &\quad \left. + \sum_{E \in \partial T \cap \Gamma_N} \|g_N - \mathbf{n}_E \cdot \boldsymbol{\kappa} \cdot \nabla u_h\|_{L^2(E)} \|v - v_h\|_{L^2(E)} \right\}. \tag{3.150}
 \end{aligned}$$

We now have to introduce estimates for the terms

$$\|v - v_h\|_{L^2(T)} \quad \text{and} \quad \|v - v_h\|_{L^2(E)}$$

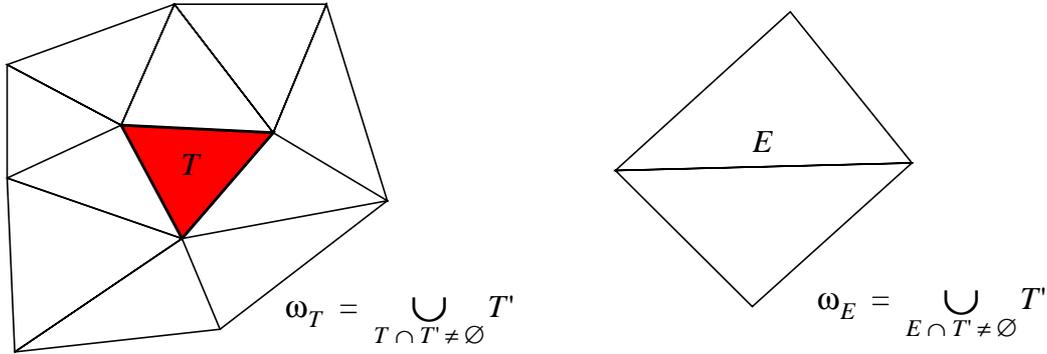
where  $v_h$  denotes some interpolant. As a first difficulty we have already mentioned that the interpolation error has to be estimated in a stronger norm than in the  $L^2$ -norm due to (3.141) which means that also *derivatives* of the error have to be measured. Moreover we have to cope with the fact that functions in  $H^1$  are not necessarily continuous so that the usual Lagrange interpolation is not valid: point-wise evaluation does not make sense anymore for a field with possible singularities as we have seen in section 3.1.2 on Sobolev spaces. We avoid this by introducing a special interpolation operator  $I_h$  named after Clément [107]. Setting  $v_h = I_h v$  we then have the following interpolation estimates:

$$\|v - v_h\|_{L^2(T)} \leq ch_T \|\nabla v\|_{L^2(\omega_T)} \quad (3.151)$$

and

$$\|v - v_h\|_{L^2(E)} \leq ch_E^{1/2} \|\nabla v\|_{L^2(\omega_T)} \quad (3.152)$$

where  $c$  denotes some constant and  $\omega_T$  the patch of all triangles whose intersection with  $T$  is non empty as sketched in Figure 3.10.



**Figure 3.10** The domains  $\omega_T$  and  $\omega_E$ .

A main tool in deriving equation (3.152) is a trace theorem. Applying these estimates to (3.150) and making use of the Cauchy-Schwarz inequality once more yields

$$\begin{aligned} \langle \tilde{f} - Lu_h, v \rangle^2 &\leq c \left( \sum_{T \in S} \left( \|f + \nabla \cdot \boldsymbol{\kappa} \cdot \nabla u_h\|_{L^2(T)}^2 h_T^2 \right. \right. \\ &\quad + \sum_{E \in S, E \subset \Omega} \frac{1}{4} \|[ \mathbf{n}_E \cdot \boldsymbol{\kappa} \cdot \nabla u_h ]\|_{L^2(E)}^2 h_E \\ &\quad \left. \left. + \sum_{E \in S, E \subset \Gamma_N} \|g_N - \mathbf{n}_E \cdot \boldsymbol{\kappa} \cdot \nabla u_h\|_{L^2(E)}^2 h_E \right) \right) \left( \sum_{T \in S} \|\nabla v\|_{L^2(\omega_T)}^2 \right), \end{aligned} \quad (3.153)$$

where  $h_T, h_E$  denote the triangle diameter and the edge length respectively.

We then note that

$$\sum_{T \in S} \|\nabla v\|_{L^2(\omega_T)}^2 \leq c \|\nabla v\|_{L^2(\Omega)}^2 \quad (3.154)$$

and consider the fact that the  $X$ -norm in the space  $X$  is equivalent to the energy norm of the problem as we saw in (3.135). Thus we can take the supremum of the estimate (3.153) which yields the following local element error estimator

$$\begin{aligned} \eta_T^2(u_h) &= \|f + \nabla \cdot \boldsymbol{\kappa} \cdot \nabla u_h\|_{L^2(T)}^2 h_T^2 \\ &+ \sum_{E \in S, E \subset \Omega} \frac{1}{4} \|\mathbf{n}_E \cdot \boldsymbol{\kappa} \cdot \nabla u_h\|_{L^2(E)}^2 h_E \\ &+ \sum_{E \in S, E \subset \Gamma_N} \|g_N - \mathbf{n}_E \cdot \boldsymbol{\kappa} \cdot \nabla u_h\|_{L^2(E)}^2 h_E \end{aligned} \quad (3.155)$$

only depending on the computed solution  $u_h$  and the given data of the problem. Thus we have shown the a posteriori error estimation

$$\|u - u_h\| \leq c \left( \sum_{T \in S} \eta_T^2(u_h) \right)^{1/2} \quad (3.156)$$

for the energy norm of the error which is reliable and computable. The constant  $c$  depends mainly on the smallest angle occurring in the triangulation  $S$ . The first term on the RHS of (3.155) specifies an element residual with respect to the strong form of the problem. In case there is zero load specified, i.e. no heat source or electric charge density is specified only the second addend has to be considered. If in addition linear polynomials are chosen to interpolate the field  $u$ , the element residual vanishes completely since second order derivatives of linear functions are identically zero. The second term describes jumps of the gradient across a given edge in the domain interior and might be viewed as a measure of the “smoothness” of the solution whereas the last term considers an error introduced by the deviation of the approximated from the prescribed normal flux along non-essential bound-

aries. It will be shown in the next section that the error estimator (3.155) not only is reliable in the sense of (3.156), but also efficient which means that it is bounded from above by the real error.

### 3.3.3 Efficiency of the Error Estimator

In the previous chapter we have shown the error estimator  $\eta_T(u_h)$  to be reliable, in other words, that there really is an upper bound for the quantity which we denote as the error, and that this bound, to some extent, also is computable. However, we still do not know how *good* this bound is, meaning it could be undesirably far away from the true error. To make the problem clear we mention that arbitrarily high values also could be a bound of the error, although not useful for practical purpose. In fact, the error estimator itself can be proved to be bounded by the real error

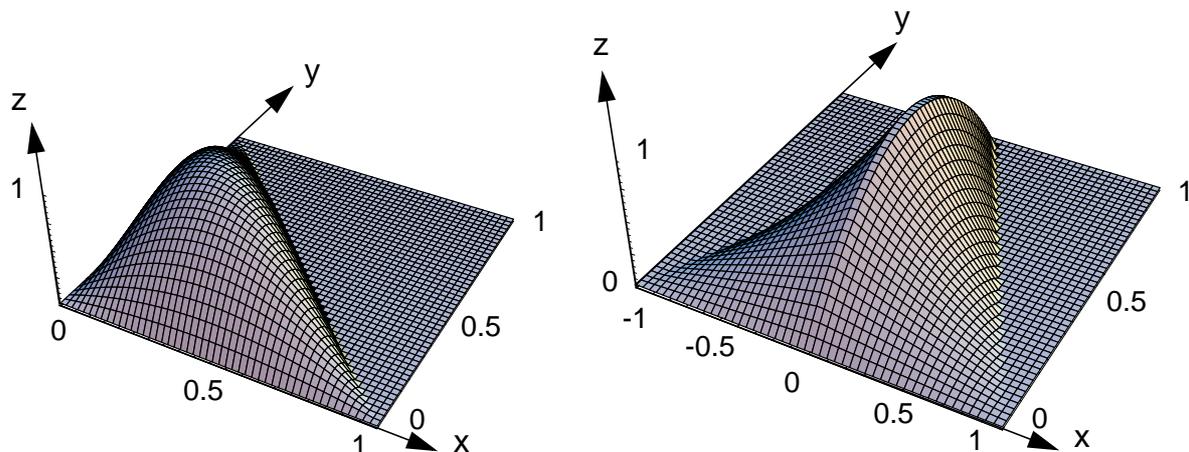
$$\eta_T(u_h) \leq c \|u - u_h\|. \quad (3.157)$$

We show this by displaying the most significant steps. For more details, see for example [46],[56],[77].

We start by defining bubble functions that are polynomials and have local support. The reason why these functions are given that special name becomes clear when looking at Figure 3.11. Consider for each triangle  $T$  of the triangulation  $S$  its barycentric coordinates  $\lambda_{1,T}, \lambda_{2,T}, \lambda_{3,T}$ . For any point  $P$ , these are defined to be the weights  $\lambda_i, i = 1, \dots, 3$ , such that, when applied to three given arbitrary points  $P_i, i = 1, \dots, 3$ ,  $P$  is the centre of mass of the triangle generated by  $P_i, i = 1, \dots, 3$ . For the reference triangle, see Figure 3.5, these coincide with the shape functions (3.78). The *triangle bubble function* then is defined as a third order polynomial

$$\psi_T = \begin{cases} 27\lambda_{1,T}(\mathbf{x})\lambda_{2,T}(\mathbf{x})\lambda_{3,T}(\mathbf{x}) & \text{for } \mathbf{x} \in T \\ 0 & \text{for } \mathbf{x} \notin T \end{cases} \quad (3.158)$$

and the edge bubble function as



**Figure 3.11** The triangle and the edge bubble functions  $\psi_T$  and  $\psi_E$  respectively. The triangle bubble function (left) is displayed for the reference triangle  $\hat{T}$ , whereas the edge bubble function (right) is displayed for the edge which bounds the reference triangle and its reflection triangle along the  $y$ -axis. Both bubble functions have local support, see the text for more details.

$$\psi_E = \begin{cases} 4 \prod_{j=0, j \neq k}^2 \lambda_{j,T}(x) & \text{for } x \in T \\ 4 \prod_{j=0, j \neq i}^2 \lambda_{j,T}(x) & \text{for } x \in T' \\ 0 & \text{for } x \in \Omega \setminus \omega_E \end{cases} \quad (3.159)$$

where coordinates' numbering is in accordance with the sketch in Figure 3.5. It is found that these functions are continuous on the domain and each function's range is between zero and one. An additional mapping is required that carries out a continuation of a function defined on an edge to one that is defined also on the neighbouring triangles  $\Upsilon: L^2(E) \rightarrow L^2(\omega_E)$ . We set for any polynomial  $\sigma$  of degree  $\geq 0$

$$\Upsilon(\sigma(x)) = \sigma(x') \quad \text{for } x \in T \quad (3.160)$$

and  $\mathbf{x}'$  being a point on  $E$  such that  $\lambda_{j,T}(\mathbf{x}) = \lambda_{j,T}(\mathbf{x}')$ , and  $\lambda_{j,T}$  one of the two coordinate functions that are non-constant along the edge  $E$ . For polynomials  $v$  we then have the following relations:

$$\|\Psi_T^{1/2} v\|_{L^2(T)} \geq c \|v\|_{L^2(T)} \quad (3.161)$$

$$\|\Psi_T v\|_{L^2(T)} \leq \|v\|_{L^2(T)} \quad (3.162)$$

$$\|\nabla(\Psi_T v)\|_{L^2(T)} \leq \frac{c}{h_T} \|v\|_{L^2(T)} \quad (3.163)$$

$$\|\Psi_E^{1/2} \sigma\|_{L^2(E)} \geq c \|\sigma\|_{L^2(E)} \quad (3.164)$$

$$ch^{1/2} \|\sigma\|_{L^2(E)} \leq \|\Psi_E \Upsilon(\sigma)\|_{L^2(T)} \leq ch^{1/2} \|\sigma\|_{L^2(E)} \quad (3.165)$$

$$\|\nabla(\Psi_E \Upsilon(\sigma))\|_{L^2(T)} \leq ch^{-1} \|\Psi_E \Upsilon(\sigma)\|_{L^2(T)} \leq ch^{-\frac{1}{2}} \|\sigma\|_{L^2(E)}. \quad (3.166)$$

The first relation follows from the fact that norms on finite dimensional spaces are equivalent ( $\Psi_T$  is a polynomial) and by bidirectional transformation of  $v$  onto the reference triangle. The second inequality takes into account that  $\Psi_T$  is bounded by one. Relation (3.163) is an inverse inequality [77], together with (3.162). The last three inequalities can be shown in a similar fashion, see for example, [46]. We then define the element residuum

$$r_T(u_h) = f + \nabla \cdot \boldsymbol{\kappa} \cdot \nabla u_h \quad (3.167)$$

and the edge residuum

$$r_E(u_h) = \begin{cases} [\mathbf{n}_E \cdot \boldsymbol{\kappa} \cdot \nabla u_h] & E \in \partial T \cap \Omega \\ g_N - (\mathbf{n}_E \cdot \boldsymbol{\kappa} \cdot \nabla u_h) & E \in \partial T \cap \Gamma_N \\ 0 & \text{else} \end{cases} \quad (3.168)$$

In order to keep the presentation clear we define an operator  $G: H^1(\Omega) \rightarrow \mathbb{R}$  that is related to the element and edge residuals as

$$\begin{aligned}
 G(v) &= \sum_{T \in \mathcal{T}_h} \int_T r_T(u_h) v dx + \sum_{E \in \partial \mathcal{T}} \int_E r_E(u_h) v d\Gamma \\
 &= \sum_{T \in \mathcal{T}_h} \int_T (f + \nabla \cdot \boldsymbol{\kappa} \cdot \nabla u_h) v dx + \sum_{E \in \partial \mathcal{T} \cap \Omega_E} \int_E [\mathbf{n}_E \cdot \boldsymbol{\kappa} \cdot \nabla u_h] v d\Gamma \\
 &\quad + \sum_{E \in \partial \mathcal{T} \cap \Gamma_N} \int_E (g_N - (\mathbf{n}_E \cdot \boldsymbol{\kappa} \cdot \nabla u_h)) v d\Gamma \\
 &= \int_{\Omega} f v dx - \sum_{T \in \mathcal{T}_h} \left( \int_T (-\nabla \cdot \boldsymbol{\kappa} \cdot \nabla u_h) v dx + \sum_{E \in \partial T \cap \Omega_E} \int_E (\mathbf{n}_E \cdot \boldsymbol{\kappa} \cdot \nabla u_h) v d\Gamma \right. \\
 &\quad \left. - \sum_{E \in \partial T \cap \Gamma_N} \int_E g_N v d\Gamma \right) \\
 &= \int_{\Omega} f v dx - \sum_{T \in \mathcal{T}_h} \int_T (\nabla v \cdot \boldsymbol{\kappa} \cdot \nabla u_h) dx + \int_{\Gamma_N} g_N v d\Gamma \\
 &= \sum_{T \in \mathcal{T}_h} \int_{\Omega} \nabla v \cdot \boldsymbol{\kappa} \cdot \nabla (u - u_h) dx \\
 &= a(u - u_h, v)
 \end{aligned} \tag{3.169}$$

where partial integration has been applied for the transformation. The element error estimator then can be expressed as

$$\eta_T^2(u_h) = c \left( h_T^2 \|r_T(u_h)\|_{L^2(T)}^2 + \sum_{E \in \partial T} h_T \|r_E(u_h)\|_{L^2(T)}^2 \right). \tag{3.170}$$

Noting that the element bubble function  $\psi_T$  defined above has local support we choose

$$v = \psi_T r_T(u_h) \tag{3.171}$$

which is identically zero outside the interior of  $T$ . Then, taking into account that  $r_T(u_h)$  is polynomial, estimate (3.161) yields

$$G(\Psi_T r_T(u_h)) = \int_T \Psi_T r_T^2(u_h) = \|\Psi_T^{1/2} r_T(u_h)\|_{L^2(T)}^2 \geq c \|r_T(u_h)\|_{L^2(T)}^2 \quad (3.172)$$

where we have used the fact that the second integral vanishes since it extends over the element edges. This inequality is recast into

$$\begin{aligned} & c \|r_T(u_h)\|_{L^2(T)}^2 \\ & \leq G(\Psi_T r_T(u_h)) \\ & = \int_T \nabla(\Psi_T r_T(u_h)) \cdot \boldsymbol{\kappa} \cdot \nabla(u - u_h) dx \\ & \leq c \|\nabla(\Psi_T r_T(u_h))\|_{L^2(T)} \|u - u_h\|_T \end{aligned} \quad (3.173)$$

using the definition (3.169) and the continuity of the bilinear form  $a$ . Hence

$$c \|r_T(u_h)\|_{L^2(T)}^2 \leq c \|u - u_h\|_T \frac{1}{h_T} \|r_T(u_h)\|_{L^2(T)} \quad (3.174)$$

by virtue of (3.163). Dividing the last inequality by  $\|r_T(u_h)\|_{L^2(T)}$ , squaring the result and multiplying both sides by  $h_T^2$  yields the estimate

$$h_T^2 \|r_T(u_h)\|_{L^2(T)}^2 \leq c \|u - u_h\|_T^2. \quad (3.175)$$

An estimate for the edge residual is derived in a similar fashion. We therefore set

$$v = \Psi_E \Upsilon(r_E(u_h)) \quad (3.176)$$

such that  $v$  vanishes outside the interior of the patch  $\omega_E$ . Then, by picking out the relevant elements from (3.169) we have

$$\begin{aligned}
G(\Psi_E \Upsilon(r_E(u_h))) &= \sum_{T \in \omega_{ET}} \int r_T(u_h) \Psi_E \Upsilon(r_E(u_h)) dx + \int_E r_E^2(u_h) \Psi_E d\Gamma \\
&= \sum_{T \in \omega_{ET}} \int r_T(u_h) \Psi_E \Upsilon(r_E(u_h)) dx + \|\Psi_E^{1/2} r_E(u_h)\|_{L^2(E)}^2 \\
&\geq \sum_{T \in \omega_{ET}} \int r_T(u_h) \Psi_E \Upsilon(r_E(u_h)) dx + c \|r_E(u_h)\|_{L^2(E)}^2
\end{aligned} \tag{3.177}$$

since  $r_E(u_h)$  only is defined on the edge  $E$  and vanishing elsewhere. The inequality is a direct result of (3.164). Reorganizing (3.177) yields

$$\begin{aligned}
c \|r_E(u_h)\|_{L^2(E)}^2 &\leq |G(\Psi_E \Upsilon(r_E(u_h)))| + \sum_{T \in \omega_E} \|r_T(u_h)\|_{L^2(T)} \|\Psi_E \Upsilon(r_E(u_h))\|_{L^2(T)} \\
&\leq |G(\Psi_E \Upsilon(r_E(u_h)))| + \|r_T(u_h)\|_{L^2(\omega_E)} h_T^{1/2} \|r_E(u_h)\|_{L^2(E)}
\end{aligned} \tag{3.178}$$

exploiting the Cauchy-Schwarz inequality and relation (3.165). For the first addend in (3.178) we have

$$\begin{aligned}
|G(\Psi_E \Upsilon(r_E(u_h)))| &= a(u - u_h, \Psi_E \Upsilon(r_E(u_h))) \leq c \|u - u_h\|_{\omega_E} \|\Psi_E \Upsilon(r_E(u_h))\|_{L^2(\omega_E)}
\end{aligned} \tag{3.179}$$

due to the continuity of the bilinearform  $a$ . Applying (3.166) to the above inequality yields

$$|G(\Psi_E \Upsilon(r_E(u_h)))| \leq c h_T^{-1/2} \|u - u_h\|_{\omega_E} \|r_E(u_h)\|_{L^2(E)} \cdot \tag{3.180}$$

Then, combining (3.178) and (3.180) the estimate for the edge residual turns into

$$\begin{aligned}
 & h_T \|r_E(u_h)\|_{L^2(E)}^2 \\
 & \leq c \left( \|u - u_h\|_{\omega_E}^2 + h_T^2 \|r_T(u_h)\|_{L^2(\omega_E)}^2 \right) \\
 & \leq c \|u - u_h\|_{\omega_E}^2
 \end{aligned} \tag{3.181}$$

where (3.178) has been divided by  $\|r_E(u_h)\|$ , using (3.180) for the second addend. The result thus has been squared and multiplied by  $h_T$ . To get the second inequality in (3.181) we use the estimate (3.175). Hence, the final estimate is obtained by returning to (3.170) and simply applying both estimates (3.175) and (3.181)

$$\begin{aligned}
 \eta_T^2(u_h) & \leq c \left( h_T^2 \|r_T(u_h)\|_{L^2(T)}^2 + \sum_{E \in \partial T} h_T \|r_E(u_h)\|_{L^2(T)}^2 \right) \\
 & \leq c \|u - u_h\|_{\omega_E}^2.
 \end{aligned} \tag{3.182}$$

This is exactly (3.157), a bound for the error estimator has been established. The computable expression  $\eta_T(u_h)$  not only represents a bound for the solution error but itself is bounded by the true error. The undesirable case that the error estimator is too far away from the real error is thus avoided.

### 3.4 Error Estimation for multi-layer thin structures

The complete a posteriori error analysis for electro-thermomechanically coupled multi-layer thin plate finite element simulations takes the weak form (3.122) as its basis. Merging together the source functions and the natural boundary conditions of the different physical domains, namely the mechanical, the thermal and the electrical sources into one field  $\tilde{F}$ , (3.122) can be rewritten as

$$a(\mathbf{U}, \mathbf{V}) = \langle \tilde{F}, \mathbf{V} \rangle \quad \forall \mathbf{V} \in X \tag{3.183}$$

where  $X$  is the space of admissible solutions as in (3.124)

$$X = \{ \mathbf{U} \in (H^1(\omega))^2 \times H^2(\omega) \times H^1(\Omega) \times H^1(\Omega), \mathbf{U} \text{ satisfies all essential BCs} \}. \tag{3.184}$$

Since the thermal problem can be fully decoupled from the electromechanical problem, that is, one first could solve the heat equation and then insert the calculated temperature field into the electromechanical part, we restrict our analysis to the equation where only the displacement fields and the electrostatical field is unknown. The error estimator for the temperature field then is exactly identical to that which we derived in the previous section 3.3. We now redefine  $\mathbf{U} = (\mathbf{u}, w, \phi)$  and its variation  $\mathbf{V} = (\mathbf{v}, v, \psi)$  such that the remaining variational problem reads

$$\begin{aligned}
 & (\nabla w \cdot \boldsymbol{\sigma}_0^{th}, \nabla v)_\omega \\
 & - (\mathbf{A}_0 : (\nabla \mathbf{u})^S - \mathbf{A}_1 : \nabla \nabla w + (\hat{\boldsymbol{\pi}}^T \cdot \nabla \phi)_0 + \boldsymbol{\sigma}_0^{th}, (\nabla \mathbf{v})^S)_\omega \\
 & + (\mathbf{A}_1 : (\nabla \mathbf{u})^S - \mathbf{A}_2 : \nabla \nabla w + (\hat{\boldsymbol{\pi}}^T \cdot \nabla \phi)_1 + \boldsymbol{\sigma}_1^{th}, \nabla \nabla v)_\omega \\
 & + (\nabla \psi, \hat{\boldsymbol{\pi}} : (\nabla \mathbf{u})^S - z \hat{\boldsymbol{\pi}} : \nabla \nabla w - \hat{\boldsymbol{\chi}} \cdot \nabla \phi)_\Omega \\
 & = \langle \mathbf{f}_0, \mathbf{v} \rangle_\omega - \langle \mathbf{f}_1, \nabla v \rangle_\omega + \langle g_0, v \rangle_\omega + \langle \rho, \psi \rangle_\Omega + \langle h, \psi \rangle_{\Gamma_{N, electrical}}.
 \end{aligned} \tag{3.185}$$

Here we have introduced a couple of modifications of which the most important is the conversion of the temperature dependent thermal expansion into a constant stress term  $\boldsymbol{\sigma}^{th}$  which is widely known as thermal prestress and is of immense importance when dealing with microstructures [88]. When it comes to a simulation where temperature dependent behaviour is required one could just replace the constant prestress by the thermal expansion tensor contracted with the previously calculated temperature. As another change we have omitted the electrostatic term in the first addend of (3.185) which is a remnant of the non-linear theory, see (3.111) and (3.112), and therefore could be neglected if non-linear effects modeled by this term do not play a dominant role [59]. With these changes of the variational form the solution space changes accordingly,

$$X = \{ \mathbf{U} \in (H^1(\omega))^2 \times H^2(\omega) \times H^1(\Omega), \mathbf{U} \text{ satisfies all essential BCs} \}. \tag{3.186}$$

We further extend (3.185) by introducing a set of natural boundary conditions for the out-of-plane displacement field  $w$ , only involving derivatives at least of order two, which for thin plate problems can be stated as follows:

$$\mathbf{n}_\Gamma \cdot \mathbf{A} : \nabla \nabla w = \mathbf{M}_b \quad \text{on } \Gamma_{b, w} \tag{3.187}$$

where  $\mathbf{M}_b$  are the vector-valued bending moments imposed on a part of the boundary  $\Gamma_{b,w} \subset \partial\Omega$ . Having fixed second order derivatives on a part of the domain boundary we may specify the bending moments with regard to the moment order of the elastic tensor in the following way:

$$\mathbf{n}_\Gamma \cdot \mathbf{A}_2 : \nabla \nabla w = \mathbf{M}_{b,2} \quad \text{on } \Gamma_{b,w} \quad (3.188)$$

and

$$\mathbf{n}_\Gamma \cdot \mathbf{A}_1 : \nabla \nabla w = \mathbf{M}_{b,1} \quad \text{on } \Gamma_{b,w}. \quad (3.189)$$

The scalar valued normal shear forces imposed on another part  $\Gamma_{s,w} \subset \partial\Omega$  of the boundary are given by

$$\mathbf{n}_\Gamma \cdot \nabla \cdot (\mathbf{A}_2 : \nabla \nabla w) = F_s \quad \text{on } \Gamma_{s,w}. \quad (3.190)$$

In fact, there are some additional choices of imposing natural boundary conditions to the thin plate problems. Since most of them are of less practical interest we assume them to be homogenous. As far as the in plane displacement field is concerned we only assume either exclusively essential boundary conditions or essential boundary conditions combined with homogenous natural boundary conditions. We thus can fix some part of the boundary where we define natural boundary conditions for the in-plane displacement fields  $\mathbf{u}$ , independently of the type of boundary condition for the out-of plane displacement. The homogeneous boundary condition then read

$$\mathbf{n} \cdot \mathbf{A}_0 : (\nabla \mathbf{u})^S = \mathbf{0} \quad \text{on } \Gamma_{c,u} \quad (3.191)$$

and

$$\mathbf{n} \cdot \mathbf{A}_1 : (\nabla \mathbf{u})^S = \mathbf{0} \quad \text{on } \Gamma_{c,u} \quad (3.192)$$

where the RHS of the both equations describes the imposed zero normal stress couples. Additionally we may specify homogeneous bending moments for the in-plane displacement field

$$\mathbf{n} \cdot (\nabla \cdot (\mathbf{A}_1 : (\nabla \mathbf{u})^S)) = 0 \quad \text{on } \Gamma_{b,u} \quad (3.193)$$

taking into account  $\Gamma_{c,u} \cup \Gamma_{b,u} \subset \partial\Omega$  and  $\Gamma_{c,u} \cup \Gamma_{b,u} \neq \partial\Omega$ . We should have noticed that homogenous boundary conditions of the kind (3.187) and (3.190) correspond to completely free edges of a plate. We now are able to set up the RHS of (3.183)

$$\langle \tilde{\mathbf{F}}, \mathbf{V} \rangle = \langle \mathbf{F}, \mathbf{V} \rangle + \int_{\Gamma_{b,w}} \mathbf{M}_b \cdot \nabla v d\Gamma + \int_{\Gamma_{s,w}} F_s v d\Gamma \quad (3.194)$$

thus enabling us to state the variational problem

$$a(\mathbf{U}, \mathbf{V}) = \langle \tilde{\mathbf{F}}, \mathbf{V} \rangle \quad \forall \mathbf{V} \in X. \quad (3.195)$$

Having fixed a finite dimensional sub-space  $X_h$  of  $X$  as

$$X_h = \left\{ \begin{array}{l} \mathbf{U}_h \in (C^1(\Omega))^3, \quad \mathbf{U}_h|_T \in (\mathbb{P}_5(\omega))^3 \times \mathbb{P}_2(\Omega), \\ \mathbf{U}_h \text{ satisfies all essential BCs} \end{array} \right\} \quad (3.196)$$

such that every displacement field is interpolated by piecewise fifth order polynomials [103],[104] and the electrical potential field by trilinear polynomials as defined in Table 3.1. The in-plane displacement field, of course, does not require to be interpolated as smooth as the flexure  $w$ . However, since the convergence rate of the solution in the context of (3.67) is determined by the lowest polynomial degree that occurs, it seems advisable to choose the in-plane approximation in a similar order as the out-of plane approximation [103]. We can reformulate the finite element problem (3.138)

$$a(\mathbf{U}_h, \mathbf{V}_h) = \langle \tilde{\mathbf{F}}, \mathbf{V}_h \rangle \quad \forall \mathbf{V}_h \in X_h. \quad (3.197)$$

Making use of the ellipticity of the bilinear-form  $a$  and recalling the abstract definition of the operator  $L$  in (3.132) and (3.133) we now can state the error estimation of the thin plate problem

$$\|\mathbf{U} - \mathbf{U}_h\| \leq \sup_{\mathbf{V} \in X, \|\mathbf{V}\|=1} a(\mathbf{U} - \mathbf{U}_h, \mathbf{V}) = \sup_{\mathbf{V} \in X, \|\mathbf{V}\|=1} \langle \mathbf{F} - L\mathbf{U}_h, \mathbf{V} \rangle$$

where the energy norm here again is induced by the bilinear-form

$$\|U - U_h\|^2 = a(U, U). \quad (3.198)$$

We obtain the counterpart of equation (3.147) according to the definition of the bilinear form  $a$

$$\begin{aligned} \langle \tilde{\mathbf{F}} - L\mathbf{U}_h, \mathbf{V} - \mathbf{V}_h \rangle &= R^P + \langle \tilde{\mathbf{F}}^M, \mathbf{V}^M - \mathbf{V}_h^M \rangle \\ &\quad - \int_{\Omega} (\mathbf{A}_0 : (\nabla \mathbf{u}_h)^S - \mathbf{A}_1 : \nabla \nabla w_h + \boldsymbol{\sigma}_0^{\text{ext}}) (\nabla(v - v_h))^S dx \\ &\quad + \int_{\Omega} (\mathbf{A}_1 : (\nabla \mathbf{u}_h)^S - \mathbf{A}_2 : \nabla \nabla w_h + \boldsymbol{\sigma}_1^{\text{ext}}) (\nabla \nabla(v - v_h)) dx \\ &\quad - \int_{\Omega} \nabla w_h \cdot \boldsymbol{\sigma}_0^{\text{ext}} \cdot \nabla(v - v_h) dx . \end{aligned} \quad (3.199)$$

To keep the exposition reasonably clear with respect to the number of terms describing different kinds of error contributions we confine ourselves to only a part of the problem for the time being. This is realized by means of splitting the linear form into a purely mechanical part and a part involving the piezo-electric effects

$$\begin{aligned} \langle \tilde{\mathbf{F}} - L\mathbf{U}_h, \mathbf{V} - \mathbf{V}_h \rangle &= \langle \tilde{\mathbf{F}}^M - L^M \mathbf{U}_h^M, \mathbf{V}^M - \mathbf{V}_h^M \rangle + \langle \tilde{\mathbf{F}}^P - L^P \mathbf{U}_h^P, \mathbf{V}^P - \mathbf{V}_h^P \rangle, \\ &\equiv \langle \tilde{\mathbf{F}}^M - L^M \mathbf{U}_h^M, \mathbf{V}^M - \mathbf{V}_h^M \rangle + R^P \end{aligned} \quad (3.200)$$

denoted as the piezoelectric residual  $R^P$ . Afterwards it will be clear in which way the contributions due to the piezoelectrical effects have to be set up. We make the following definitions to further simplify the exposition:

$$\begin{aligned} \boldsymbol{\sigma}_{u0} &= \mathbf{A}_0 : (\nabla \mathbf{u}_h)^S & \boldsymbol{\sigma}_{u1} &= \mathbf{A}_1 : (\nabla \mathbf{u}_h)^S \\ \boldsymbol{\sigma}_{w1} &= \mathbf{A}_1 : \nabla \nabla w_h & \boldsymbol{\sigma}_{w2} &= \mathbf{A}_2 : \nabla \nabla w_h . \\ \mathbf{e}_h &= (v - v_h) & \mathbf{e}_h &= (v - v_h) \end{aligned} \quad (3.201)$$

Assuming that the domain  $\omega$  is decomposed into a shape regular triangulation and partially integrating equation (3.199) yields

$$\begin{aligned}
\langle \tilde{\mathbf{F}} - LU_h, \mathbf{V} - \mathbf{V}_h \rangle &= R^P + \langle \tilde{\mathbf{F}}^M, \mathbf{V}^M - \mathbf{V}_h^M \rangle - \sum_{T \in \mathcal{S}} \left( - \int_T ((\nabla \cdot \boldsymbol{\sigma}_{u0}) \cdot \mathbf{e}_h) dx + \right. \\
&\quad \int_{\partial T} \mathbf{n} \cdot \boldsymbol{\sigma}_{u0} \cdot \mathbf{e}_h d\partial T + \int_T \nabla \cdot \boldsymbol{\sigma}_{w1} \cdot \mathbf{e}_h dx - \int_{\partial T} \mathbf{n} \cdot \boldsymbol{\sigma}_{w1} \cdot \mathbf{e}_h d\partial T + \\
&\quad \int_{\partial T} \mathbf{n} \cdot \boldsymbol{\sigma}_{w1} \cdot \mathbf{e}_h d\partial T - \int_{\partial T} \mathbf{n} \cdot \boldsymbol{\sigma}_0^{ext} \cdot \mathbf{e}_h d\partial T + \int_{\partial T} (\mathbf{n} \cdot \nabla \cdot (\nabla \cdot \boldsymbol{\sigma}_{u1})) e_h d\partial T - \\
&\quad \int_{\partial T} \mathbf{n} \cdot \boldsymbol{\sigma}_{u1} \cdot \nabla e_h d\partial T + \int_T (\nabla \cdot \nabla \cdot \boldsymbol{\sigma}_{w2}) e_h dx - \int_{\partial T} (\mathbf{n} \cdot \nabla \cdot \boldsymbol{\sigma}_{w2}) e_h d\partial T + \\
&\quad \int_{\partial T} \mathbf{n} \cdot \boldsymbol{\sigma}_{w2} \cdot \nabla e_h d\partial T - \int_{\partial T} \mathbf{n} \cdot \boldsymbol{\sigma}_1^{ext} \cdot \nabla e_h d\partial T - \int_T \nabla \cdot (\nabla w_h \cdot \boldsymbol{\sigma}_0^{ext}) e_h dx + \\
&\quad \left. \int_{\partial T} (\mathbf{n} \cdot \nabla w_h \cdot \boldsymbol{\sigma}_0^{ext}) e_h d\partial T \right). \tag{3.202}
\end{aligned}$$

Further expanding equation (3.202) and making use of the fact that integration over interior edges occurs twice results in

$$\begin{aligned}
 \langle \tilde{\mathbf{F}} - \mathbf{LU}_h, \mathbf{V} - \mathbf{V}_h \rangle &= R^P + \sum_{T \in \mathcal{S}} \left( \int_T (f_0 + \nabla \cdot \boldsymbol{\sigma}_{u0}) \cdot \mathbf{e}_h dx \right. \\
 &\quad - \frac{1}{2} \sum_{E \in \partial T \cap \omega_E} \int [n \cdot \boldsymbol{\sigma}_{u0}] \cdot \mathbf{e}_h d\Gamma - \frac{1}{2} \sum_{E \in \partial T \cap \Gamma_{c,u} E} \int [n \cdot \boldsymbol{\sigma}_{u0}] \cdot \mathbf{e}_h d\Gamma \\
 &\quad - \int_T \nabla \cdot \boldsymbol{\sigma}_{w1} \cdot \mathbf{e}_h dx + \frac{1}{2} \sum_{E \in \partial T \cap \omega_E} \int [n \cdot \boldsymbol{\sigma}_{w1}] \cdot \mathbf{e}_h d\Gamma \\
 &\quad + \frac{1}{2} \sum_{E \in \partial T \cap \Gamma_{s,w} E} \int (n \cdot \boldsymbol{\sigma}_{w1} + \mathbf{M}_{b,1}) \cdot \mathbf{e}_h d\Gamma - \frac{1}{2} \sum_{E \in \partial T \cap \Omega_E} \int [n \cdot \boldsymbol{\sigma}_0^{ext}] \cdot \mathbf{e}_h d\Gamma \\
 &\quad + \int_T (\nabla \cdot (\nabla \cdot \boldsymbol{\sigma}_{u1})) e_h dx - \frac{1}{2} \sum_{E \in \partial T \cap \omega_E} \int [n \cdot \nabla \cdot \boldsymbol{\sigma}_{u1}] e_h d\Gamma \\
 &\quad - \frac{1}{2} \sum_{E \in \partial T \cap \Gamma_{b,u} E} \int (n \cdot \nabla \cdot \boldsymbol{\sigma}_{u1}) e_h d\Gamma + \frac{1}{2} \sum_{E \in \partial T \cap \omega_E} \int ([n \cdot \nabla \cdot \boldsymbol{\sigma}_{u1}] \cdot \nabla e_h) d\Gamma \\
 &\quad + \sum_{E \in \partial T \cap \partial \omega_E} \int (n \cdot (\nabla \cdot \boldsymbol{\sigma}_{u1} - f_1)) e_h d\Gamma \\
 &\quad + \frac{1}{2} \sum_{E \in \partial T \cap \Gamma_{c,u} E} \int (n \cdot \nabla \cdot \boldsymbol{\sigma}_{u1}) \cdot \nabla e_h d\Gamma + \int_T (g_0 - \nabla \cdot (\nabla \cdot \boldsymbol{\sigma}_{w2})) e_h dx \\
 &\quad + \sum_{E \in \partial T \cap \Gamma_s E} \int (F_s + (n \cdot \nabla \cdot \boldsymbol{\sigma}_{w2})) e_h d\Gamma + \frac{1}{2} \sum_{E \in \partial T \cap \omega_E} \int [n \cdot \nabla \cdot \boldsymbol{\sigma}_{w2}] e_h d\Gamma \\
 &\quad + \sum_{E \in \partial T \cap \Gamma_b E} \int (\mathbf{M}_{b,2} - n \cdot \nabla \cdot \boldsymbol{\sigma}_{w2}) \cdot \nabla e_h d\Gamma + \frac{1}{2} \sum_{E \in \partial T \cap \partial \omega_E} \int [-n \cdot \boldsymbol{\sigma}_{w2}] \cdot \nabla e_h d\Gamma \\
 &\quad + \frac{1}{2} \sum_{E \in \partial T \cap \omega_E} \int [n \boldsymbol{\sigma}_1^{ext}] \cdot \nabla e_h d\Gamma + \int_T \nabla \nabla w_h \cdot \boldsymbol{\sigma}_0^{ext} e_h dx \\
 &\quad \left. - \frac{1}{2} \sum_{E \in \partial T \cap \omega_E} \int [n \cdot \nabla w_h \cdot \boldsymbol{\sigma}_0^{ext}] e_h d\Gamma \right). \tag{3.203}
 \end{aligned}$$

External stresses are assumed to be constant on each triangle and the following element integrals

$$\int_T (\nabla \cdot \boldsymbol{\sigma}_0^{ext}) \cdot \mathbf{e}_h dx = 0 \tag{3.204}$$

and

$$\int_T \nabla \cdot (\nabla \cdot \boldsymbol{\sigma}_1^{ext}) e_h d\mathbf{x} = 0 \quad (3.205)$$

and the boundary integral

$$\int_{\partial T} (\mathbf{n} \cdot \nabla \boldsymbol{\sigma}_1^{ext}) e_h d\Gamma = 0 \quad (3.206)$$

vanish. The last sum of equation (3.203) only adopts non zero values when the edge considered bounds two triangle elements associated with different material stacks since the gradient of the approximate out-of plane displacement field is continuous across element borders and thus does not contribute to a jump term. Applying the Cauchy-Schwarz inequality and rearranging the addends into element contributions and sums over internal and boundary edges separately then yields

$$\begin{aligned}
 & \langle \tilde{\mathbf{F}} - LU_h, \mathbf{V} - \mathbf{V}_h \rangle = \\
 & \sum_{T \in \mathcal{S}} \left\{ \left\| \nabla \cdot (\nabla \cdot \boldsymbol{\sigma}_{w2}) - g_0 \right\|_{L^2(T)} \|e_h\|_{L^2(T)} + \left\| \nabla \cdot \boldsymbol{\sigma}_{u0} + \mathbf{f}_0 \right\|_{L^2(T)} \|e_h\|_{L^2(T)} \right. \\
 & + \left\| \nabla \cdot (\nabla \cdot \boldsymbol{\sigma}_{u1}) \right\|_{L^2(T)} \|e_h\|_{L^2(T)} + \left\| \nabla \cdot \boldsymbol{\sigma}_{w1} \right\|_{L^2(T)} \|e_h\|_{L^2(T)} \\
 & + \left\| \nabla \cdot (\nabla_{w_h} \cdot \boldsymbol{\sigma}_0^{ext}) \right\|_{L^2(T)} \|e_h\|_{L^2(T)} + \frac{1}{2} \sum_{E \in \partial T \cap \Omega} \left\{ \left\| \mathbf{n} \cdot [\nabla \cdot \boldsymbol{\sigma}_{w2}] \right\|_{L^2(E)} \|e_h\|_{L^2(E)} \right. \\
 & + \left\| \mathbf{n} \cdot [\boldsymbol{\sigma}_{w2}] \right\|_{L^2(E)} \|\nabla e_h\|_{L^2(E)} + \left\| \mathbf{n} \cdot [\boldsymbol{\sigma}_{u0}] \right\|_{L^2(E)} \|e_h\|_{L^2(E)} \\
 & + \left\| \mathbf{n} \cdot [\boldsymbol{\sigma}_{u1}] \right\|_{L^2(E)} \|\nabla e_h\|_{L^2(E)} + \left\| \mathbf{n} \cdot [\nabla \cdot \boldsymbol{\sigma}_{u1}] \right\|_{L^2(E)} \|e_h\|_{L^2(E)} \\
 & + \left\| \mathbf{n} \cdot [\boldsymbol{\sigma}_{w1}] \right\|_{L^2(E)} \|e_h\|_{L^2(E)} + \left\| \mathbf{n} \cdot [\nabla_{w_h} \cdot \boldsymbol{\sigma}_0^{ext}] \right\|_{L^2(E)} \|e_h\|_{L^2(E)} \\
 & \left. + \left\| \mathbf{n} \cdot [\boldsymbol{\sigma}_0^{ext}] \right\|_{L^2(E)} \|e_h\|_{L^2(E)} + \left\| \mathbf{n} \cdot [\boldsymbol{\sigma}_1^{ext}] \right\|_{L^2(E)} \|\nabla e_h\|_{L^2(E)} \right\} \\
 & + \sum_{E \in \partial T \cap \Gamma_s} \left\| \mathbf{n} \cdot \nabla \cdot \boldsymbol{\sigma}_{w2} + F_s \right\|_{L^2(E)} \|e_h\|_{L^2(E)} + \sum_{E \in \partial T \cap \Gamma_b} \left( \left\| \mathbf{M}_{b,2} - \mathbf{n} \cdot \boldsymbol{\sigma}_{w2} \right\|_{L^2(E)} \|\nabla e_h\|_{L^2(E)} \right. \\
 & \left. + \left\| \mathbf{n} \cdot \boldsymbol{\sigma}_{w1} + \mathbf{M}_{b,1} \right\|_{L^2(E)} \|e_h\|_{L^2(E)} \right) + \sum_{E \in \partial T \cap \Gamma_{c,u}} \left( \left\| \mathbf{n} \cdot \boldsymbol{\sigma}_{u0} \right\|_{L^2(E)} \|e_h\|_{L^2(E)} \right. \\
 & \left. + \left\| \mathbf{n} \cdot \boldsymbol{\sigma}_{u1} \right\|_{L^2(E)} \|\nabla e_h\|_{L^2(E)} \right) + \sum_{E \in \partial T \cap \partial \omega} \left\| \mathbf{n} \cdot (\nabla \cdot \boldsymbol{\sigma}_{u1} - \mathbf{f}_1) \right\|_{L^2(E)} \|e_h\|_{L^2(E)} \\
 & \left. + \sum_{E \in \partial T \cap \Gamma_{b,u}} \left\| \mathbf{n} \cdot \nabla \cdot \boldsymbol{\sigma}_{u1} \right\|_{L^2(E)} \|e_h\|_{L^2(E)} \right\} + R^P.
 \end{aligned} \tag{3.207}$$

In order to perform a similar estimation as in (3.153) for the interpolation terms  $\|\mathbf{V} - \mathbf{V}_h\|_{L^2}$  we need some additional results for  $H^2$ -functions  $w$  which are given for example in [107]

$$\left\| w - w_h \right\|_{L^2(T)} \leq ch_T^2 \|w\|_{H^2(\omega_T)} \tag{3.208}$$

and

$$\|w - w_h\|_{L^2(E)} \leq ch_E^{3/2} \|w\|_{H^2(\omega_T)} \quad (3.209)$$

and for the first order derivatives along edges

$$\|\nabla(w - w_h)\|_{L^2(E)} \leq ch_E^{1/2} \|w\|_{H^2(\omega_T)}. \quad (3.210)$$

For the  $H^1$ -interpolation which is required for the in plane displacement field we refer to the above mentioned results in (3.151) and (3.152). Introducing these estimates and applying the Cauchy-Schwarz inequality once more gives an expression for an error estimator  $\eta_T^M$  for the pure mechanical part associated to each element  $T$  of the triangulation

$$\begin{aligned} \eta_T^M (U_h^M)^2 &= \|\nabla \cdot (\nabla \cdot \boldsymbol{\sigma}_{w2}) - g_0\|_{L^2(T)}^2 h_T^4 \\ &+ \|\nabla \cdot \boldsymbol{\sigma}_{u0} + \mathbf{f}_0\|_{L^2(T)}^2 h_T^4 + \|\nabla \cdot (\nabla \cdot \boldsymbol{\sigma}_{u1})\|_{L^2(T)}^2 h_T^4 \\ &+ \|\nabla \cdot \boldsymbol{\sigma}_{w1}\|_{L^2(T)}^2 h_T^4 + \|\nabla \cdot (\nabla w_h \cdot \boldsymbol{\sigma}_0^{ext})\|_{L^2(T)}^2 h_T^4 \\ &+ \frac{1}{4} \sum_{E \in \partial T \cap \Omega} \left\{ \|\mathbf{n} \cdot [\nabla \cdot \boldsymbol{\sigma}_{w2}]\|_{L^2(E)}^2 h_E^3 + \|\mathbf{n} \cdot [\boldsymbol{\sigma}_{w2}]\|_{L^2(E)}^2 h_E \right. \\ &+ \|\mathbf{n} \cdot [\boldsymbol{\sigma}_{u0}]\|_{L^2(E)}^2 h_E^3 + \|\mathbf{n} \cdot [\boldsymbol{\sigma}_{u1}]\|_{L^2(E)}^2 h_E \\ &+ \|\mathbf{n} \cdot [\nabla \cdot \boldsymbol{\sigma}_{u1}]\|_{L^2(E)}^2 h_E^3 + \|\mathbf{n} \cdot [\boldsymbol{\sigma}_{w1}]\|_{L^2(E)}^2 h_E^3 \\ &+ \|\mathbf{n} \cdot [\nabla w_h \cdot \boldsymbol{\sigma}_0^{ext}]\|_{L^2(E)}^2 h_E^3 + \|\mathbf{n} \cdot [\boldsymbol{\sigma}_0^{ext}]\|_{L^2(E)}^2 h_E^3 + \|\mathbf{n} \cdot [\boldsymbol{\sigma}_1^{ext}]\|_{L^2(E)}^2 h_E \left. \right\} \\ &+ \sum_{E \in \partial T \cap \Gamma_s} \|\mathbf{n} \cdot \nabla \cdot \boldsymbol{\sigma}_{w2} + F_s\|_{L^2(E)}^2 h_E^3 + \sum_{E \in \partial T \cap \Gamma_{b,w}} (\|\mathbf{M}_{b,2} - \mathbf{n} \cdot \boldsymbol{\sigma}_{w2}\|_{L^2(E)}^2 h_E \\ &+ \|\mathbf{n} \cdot \boldsymbol{\sigma}_{w1} + \mathbf{M}_{b,1}\|_{L^2(E)}^2 h_E^3) + \sum_{E \in \partial T \cap \Gamma_{c,u}} (\|\mathbf{n} \cdot \boldsymbol{\sigma}_{u0}\|_{L^2(E)}^2 h_E^3 \\ &+ \|\mathbf{n} \cdot \boldsymbol{\sigma}_{u1}\|_{L^2(E)}^2 h_E) + \sum_{E \in \partial T \cap \Gamma_{b,u}} \|\mathbf{n} \cdot \nabla \cdot \boldsymbol{\sigma}_{u1}\|_{L^2(E)}^2 h_E^3 \\ &+ \sum_{E \in \partial T \cap \partial \omega} \|\mathbf{n} \cdot (\nabla \cdot \boldsymbol{\sigma}_{u1} - \mathbf{f}_1)\|_{L^2(E)}^2 h_E^3. \end{aligned} \quad (3.211)$$

The first five addends describe errors introduced by the deviation of external loads from the computed loads and are thus closely connected to the element residual of the strong form of the differential equations of the plate problem. The first term measures the difference between the bi-laplacian of the out-of plane displacement field from some imposed pressure difference or gravitational force. This element residual for single-layer plates consisting of an isotropic material reads

$$D\Delta^2 w - g_0, \quad (3.212)$$

$D$  denoting the plate's flexural rigidity,  $D = (Eh^3)/(12(1 - \nu^2))$  in terms of Young's Modulus  $E$  and the Poisson number  $\nu$ , and  $\Delta^2$  the well known bi-laplacian operator.

The sum over all internal edges measures the jumps of higher order field derivatives across edges. The first term in this sum considers the jumps of the shear forces whereas the second one measures the jump error of the bending moments. Terms containing the in-plane displacement field take into account the jumps of the stress couples. In those cases where all elements consist of the same material stack, the terms involving the external prestresses only then do not vanish when the prestress tensor is non constant.

The last four terms measure errors introduced by the deviation of the computed from the prescribed normal essential boundary values. In case there are non-vanishing essential boundary conditions imposed on the in-plane displacement field, the error estimator has to be extended accordingly. Finally, we obtain the global error estimator, which is composed of locally computable error terms and requires only the computed solution and the known mesh as input data together with the contribution stemming from the piezoelectrical effects

$$\|U - U_h\| \leq \sup_{\|V\|=1} (\mathbf{F} - LU_h, V) \leq c \sqrt{\sum_{T \in S} \eta_T^M (U_h)^2 + (R^P)^2} \quad (3.213)$$

and yet is to be determined.

Analogously to the purely mechanical part we analyze the natural boundary conditions. Again, these re-arise in a natural way when partially integrating the variational form. Obviously, we can define an electrical current across the domain borders

$$\mathbf{n} \cdot \hat{\boldsymbol{\chi}} \cdot \nabla \phi = h \quad \text{on } \Gamma_{e1}, \quad (3.214)$$

$\Gamma_{e1}$  being a part of the domain's three-dimensional boundary and  $\mathbf{n}$  therefore is the three-dimensional surface normal. Moreover, we can specify a set of surface tractions and bending moments evoked by the inverse piezoelectric effect,

$$\mathbf{n}_{2D} \cdot \hat{\boldsymbol{\pi}}^T \cdot \nabla \phi = 0 \quad \text{and} \quad \mathbf{n}_{2D} \cdot z \hat{\boldsymbol{\pi}}^T \cdot \nabla \phi = 0 \quad \text{and} \quad \mathbf{n}_{2D} \cdot \nabla_{2D} \cdot z \hat{\boldsymbol{\pi}}^T \cdot \nabla \phi = 0 \quad (3.215)$$

on a part of the boundary. We assume them to be zero, but in case these become physically relevant they may be included into the error estimation as was demonstrated for the various natural mechanical BCs. The set stemming from the direct piezoelectric effect can be expressed as

$$\mathbf{n}_{3D} \cdot (\hat{\boldsymbol{\pi}} : (\nabla \mathbf{u})^S) = 0 \quad \text{and} \quad \mathbf{n}_{3D} \cdot (z \hat{\boldsymbol{\pi}} : \nabla \nabla w) = 0 \quad . \quad (3.216)$$

Each of the terms has the meaning of an electrical displacement applied on a part of the domain boundary. Analogously to (3.201) we introduce the following abbreviations with which the exposition becomes clearer:

$$\begin{aligned} \boldsymbol{\sigma}_{P\phi} &= -\hat{\boldsymbol{\pi}}^T \cdot \nabla \phi & \mathbf{D}_{Pw} &= \hat{\boldsymbol{\pi}} : \nabla \nabla w \\ \mathbf{D}_{Pu} &= \hat{\boldsymbol{\pi}} : (\nabla \mathbf{u})^S & \mathbf{D}_{P\phi} &= \hat{\boldsymbol{\chi}} \cdot \nabla \phi \\ \mathbf{e}_h &= (\mathbf{v} - \mathbf{v}_h) & e_h &= (v - v_h) & e_\psi &= (\Psi - \Psi_h) \quad . \end{aligned} \quad (3.217)$$

With that we only pick the piezoelectrical part of (3.185) and partially integrate in order to obtain

$$\begin{aligned}
 & \langle \tilde{\mathbf{F}}^P - L^P \mathbf{U}_h^P, \mathbf{V}^P - \mathbf{V}_h^P \rangle = R^P \\
 & = \langle \tilde{\mathbf{F}}^P, \mathbf{V}^P - \mathbf{V}_h^P \rangle \\
 & - \sum_{P \in \mathcal{S}} \left( \int_{\partial P} \mathbf{n}_{2D} \cdot \boldsymbol{\sigma}_{P\phi} \cdot \mathbf{e}_h d\partial P - \int_P (\nabla_{2D} \cdot \boldsymbol{\sigma}_{P\phi}) \cdot \mathbf{e}_h dx \right. \\
 & - \int_{\partial P} \mathbf{n}_{2D} \cdot z \boldsymbol{\sigma}_{P\phi} \cdot \nabla e_h d\partial P + \int_{\partial P} \mathbf{n}_{2D} \cdot \nabla_{2D} \cdot z \boldsymbol{\sigma}_{P\phi} e_h d\partial P - \int_P \nabla_{2D} \cdot (\nabla_{2D} \cdot z \boldsymbol{\sigma}_{P\phi}) e_h dx \\
 & + \int_{\partial P} (\mathbf{n}_{3D} \cdot \mathbf{D}_{Pu}) e_\psi d\partial P - \int_P (\nabla_{3D} \cdot \mathbf{D}_{Pu}) e_\psi dx \\
 & - \int_{\partial P} (\mathbf{n}_{3D} \cdot z \mathbf{D}_{Pw}) e_\psi d\partial P + \int_P (\nabla_{3D} \cdot z \mathbf{D}_{Pw}) e_\psi dx \\
 & \left. - \int_{\partial P} (\mathbf{n}_{3D} \cdot \mathbf{D}_{P\phi}) e_\psi d\partial P + \int_P (\nabla_{3D} \cdot \mathbf{D}_{P\phi}) e_\psi dx \right) . \tag{3.218}
 \end{aligned}$$

To make clear whether we deal with 2D or 3D quantities we have provided subscripts for the appropriate symbols. In contrast to (3.202) the domain integration here is partitioned into prisms  $P$ . Taking the sources and the fluxes into consideration results in

$$\begin{aligned}
 & \langle \tilde{\mathbf{F}}^P - L^P \mathbf{U}_h^P, \mathbf{V}^P - \mathbf{V}_h^P \rangle \\
 & = \sum_{P \in \mathcal{S}} \left( \int_P (\nabla_{2D} \cdot \boldsymbol{\sigma}_{P\phi}) \cdot \mathbf{e}_h dx + \int_P \nabla_{2D} \cdot (\nabla_{2D} \cdot z \boldsymbol{\sigma}_{P\phi}) e_h dx \right. \\
 & + \int_P (\nabla_{3D} \cdot \mathbf{D}_{Pu}) e_\psi dx - \int_P (\nabla_{3D} \cdot z \mathbf{D}_{Pw}) e_\psi dx - \int_P (\nabla_{3D} \cdot \mathbf{D}_{P\phi} + \rho) e_\psi dx \\
 & + \sum_{\partial P \in \Omega} \frac{1}{2} \left( \int_{\partial P} [\mathbf{n}_{2D} \cdot \boldsymbol{\sigma}_{P\phi}] \cdot \mathbf{e}_h d\partial P \right. \\
 & - \int_{\partial P} [\mathbf{n}_{2D} \cdot z \boldsymbol{\sigma}_{P\phi}] \cdot \nabla e_h d\partial P + \int_{\partial P} [\mathbf{n}_{2D} \cdot \nabla_{2D} \cdot z \boldsymbol{\sigma}_{P\phi}] e_h d\partial P \\
 & + \int_{\partial P} [\mathbf{n}_{3D} \cdot \mathbf{D}_{Pu}] e_\psi d\partial P - \int_{\partial P} [\mathbf{n}_{3D} \cdot z \mathbf{D}_{Pw}] e_\psi d\partial P + \int_{\partial P} [\mathbf{n}_{3D} \cdot \mathbf{D}_{P\phi}] e_\psi d\partial P \left. \right) \\
 & + \sum_{\partial P \cap \Gamma_{e_1}} \int_{\partial P} (\mathbf{n}_{3D} \cdot \mathbf{D}_{P\phi} + h) e_\psi d\partial P . \tag{3.219}
 \end{aligned}$$

The interpolation estimates (3.151) and (3.152) also hold in three dimensions such that, after applying the Cauchy-Schwarz inequality in order to separate the inter-

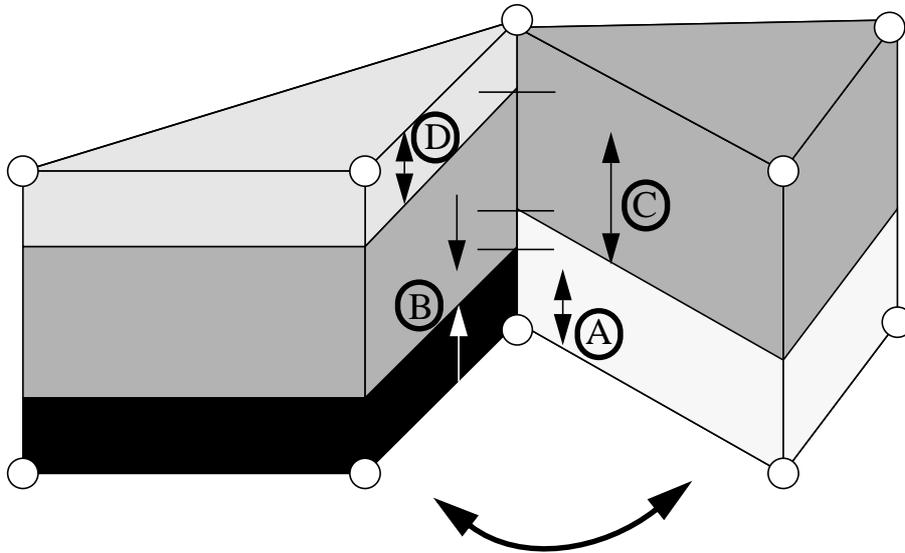
polation errors  $e$  from the other expressions by means of  $L^2$ -norms, the former can be used together with the Cauchy-Schwarz inequality for sums to yield an expression similar to that of (3.153). The error estimator which finally can be identified then reads:

$$\begin{aligned}
\eta_T^P(\mathbf{U}^P_h)^2 &= \|\nabla_{2D} \cdot \boldsymbol{\sigma}_{P\phi}\|_{L^2(P_{3D})}^2 h_P^5 + \|\nabla_{2D} \cdot (\nabla_{2D} \cdot z \boldsymbol{\sigma}_{P\phi})\|_{L^2(P_{3D})}^2 h_P^5 \\
&+ \|\nabla_{3D} \cdot \mathbf{D}_{Pu}\|_{L^2(P_{3D})}^2 h_P^2 + \|\nabla_{3D} \cdot z \mathbf{D}_{Pw}\|_{L^2(P_{3D})}^2 h_P^2 \\
&+ \|\nabla_{3D} \cdot \mathbf{D}_{Pw} + \rho\|_{L^2(P_{3D})}^2 h_P^2 \\
&+ \sum_{E \in P, P \cap \partial\Omega} \frac{1}{4} \left( \|\mathbf{n}_{2D} \cdot \boldsymbol{\sigma}_{P\phi}\|_{L^2(E_{3D})}^2 h_E^4 + \|\mathbf{n}_{2D} \cdot z \boldsymbol{\sigma}_{P\phi}\|_{L^2(E_{3D})}^2 h_E^2 \right. \\
&+ \|\mathbf{n}_{2D} \cdot \nabla_{2D} \cdot z \boldsymbol{\sigma}_{P\phi}\|_{L^2(E_{3D})}^2 h_E^4 + \|\mathbf{n}_{3D} \cdot \mathbf{D}_{Pu}\|_{L^2(E_{3D})}^2 h_E \\
&+ \left. \|\mathbf{n}_{3D} \cdot z \mathbf{D}_{Pw}\|_{L^2(E_{3D})}^2 h_E + \|\mathbf{n}_{3D} \cdot \mathbf{D}_{P\phi}\|_{L^2(E_{3D})}^2 h_E \right) \\
&+ \sum_{E \in P, E \cap \Gamma_{e1}} \|\mathbf{n}_{3D} \cdot \mathbf{D}_{P\phi} + h\|_{L^2(E_{3D})}^2 h_E
\end{aligned} \tag{3.220}$$

such that the overall error estimation can be stated as

$$\|\mathbf{U} - \mathbf{U}_h\| \leq c \sqrt{\sum_{T \in S} (\eta_T^M)^2 + (\eta_T^P)^2}. \tag{3.221}$$

The error estimator's efficiency can be proved by the same methods which we have used in section 3.3.3. It should be pointed out that the second addend in (3.220) vanishes identically because third order derivatives are acting on a trilinear polynomial which interpolates the electrical potential. Inspecting the addends that contain jumps of the according quantities across the prism-boundaries we see that  $L^2$ -integration has to be taken over rectangles. When computing these terms one has to take into account that the neighbouring layers usually are not justified and therefore the integrands change as soon as another layer appears on either side of the boundary when crossing the stack across the thickness. Figure 3.12 illustrates the situation by two opened prisms. Of course, managing this task is one of the software problems in this work which will be discussed in the next chapter.



**Figure 3.12** Computation of jump terms across prism boundaries. The prisms are the finite elements covering the layerstacks of the device. For illustration, two neighbouring prisms are opened along the rectangular edge which they share. Integration has to be performed piecewise since the jump quantities depend on both, the material properties and the approximation functions. In this example, 4 different integrands have to be treated.



# 4 OBJECT ORIENTED IMPLEMENTATION

Making the theoretical results available in terms of a computer program is a major task of this work. It is evident that assembling the various methods and tools with which to efficiently simulate microsystem components leads to a complex software system. Among the many software analysis and design methods the object oriented method seems to be the most appropriate for a flexible and extensible architecture of the simulator. This chapter describes the procedure with which the numerical tools are merged into a stable and effective software tool ADAPTREF.

## 4.1 Object Oriented (OO) Analysis and Design

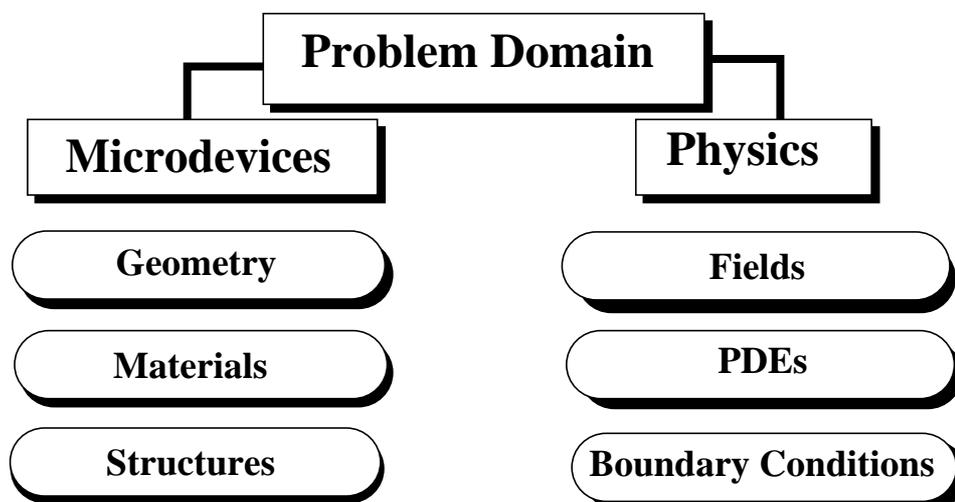
An object oriented view of the world of microsystem components, in contrast to the traditional structured analysis, emphasizes the creation of real-world models. According to [83], the method examines the requirements from the perspective of the classes and objects found in the vocabulary of the problem domain. The products of object oriented analysis (OOA) serve as the models from which an object oriented design (OOD) might be started. Then, in turn, the design products can be used as schemes for completely implementing a system using object oriented programming methods (OOP). The software system developed here, ADAPTREF, is implemented in C++.

The tool can be viewed as an entity of collaborating objects and each object is associated, in the sense of the OO paradigm, to a real world microsystem or numerical or physical object. The names are given accordingly and a typical example of an object is a physical field such as the temperature or the mechanical displacement field. A blueprint of an early stage OOA is sketched in Figure 4.1.

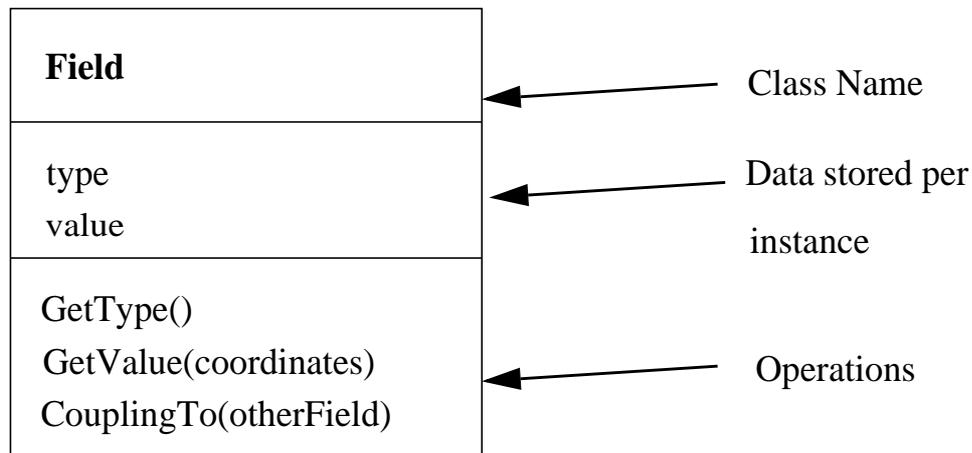
Objects like partial differential equations and a geometry obviously hold a key position in the whole software system. We further assume that an object can be

identified by having a state, behaviour and identity. These properties are defined in a class common to all objects of that type. The state of an object incorporates all the static properties of an object and the dynamic values of these properties. A property is an inherent characteristic, quality or feature that contributes to making an object uniquely that object. According to [83] the identity of an object may be defined as the property of an object which distinguishes it from all other objects. In order to avoid many kinds of errors in object-oriented programming such as, the occurrence of dangling pointers, it is immensely important to retain control of handling the objects' identities properly.

An operation is a part of the behaviour of an object and represents its outwardly visible and testable activity. In other words, it is some action that one object performs upon another in order to elicit a reaction. As an example we consider the features of an object described in the class "field". A physical field is active on a certain part of a device and it may adopt certain values. Moreover it has an identity such as, temperature, electrostatical potential or mechanical displacement. A field may also act on other fields, due to coupling effects. Figure 4.2 represents a class diagram of the class "field", a common graphical aid during the OOD phase.



**Figure 4.1** Blueprint of an early stage OOA. The objects are taken from the real world problem domain.



**Figure 4.2** An example class “Field”

The operations of a field thus comprise

- passing of the type
- obtain the field’s value at given device coordinates
- enforcing the coupling to an other field (other fields)

The individual data like the field’s type and the field’s value are stored inside each instance or object of the class. The terms instance and object are used interchangeably as well as the terms message and operation. Message passing is one part of the equation that defines the behaviour of an object; on the other hand the state of an object affects the behaviour of an object as well.

The purpose of identifying classes and objects is to establish the boundaries of the problem at hand, namely a tool for controlling the accuracy of simulation results. Additionally, this activity is the first step in devising an object oriented decomposition of the system under development. A parallel operation is dealing with the question of classification which is fundamentally a problem of clustering, that is, seeking to group things that have a common structure or exhibit a common behaviour. Classification helps us to identify generalization, specialization, and aggregation hierarchies among classes. Modularization is the property of a system that has been decomposed into a set of cohesive and loosely coupled modules. This property of the software system also can be viewed as a result of classification. A

major difficulty when it comes to classification is the fact that a given set of objects may be classified in many equally proper ways. As an example we consider the simple geometrical class of a tetrahedron. A question to be posed then is whether it shall be directly grouped with triangles and lines, since these are also simplicial geometrical objects and thus share a couple of properties, or, if it should preferably be grouped with other three dimensional objects such as prisms or cuboids. A powerful tool of object oriented analysis is the description of scenarios. It is closely related to identifying mechanisms between the different objects, the mechanisms are the means whereby objects collaborate to provide some higher level behaviour.

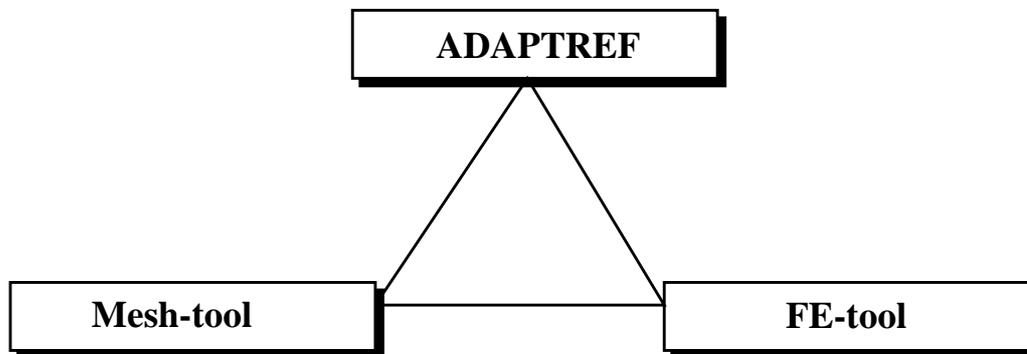
All these processes of developing software are done iteratively. At the beginning a system with a simplified architecture and only the most basic features is implemented and tested. Then, new requirements turn up and are implemented into classes whereas existing ones are improved and so the first architecture is extended. The aim then is to iterate in order to conform the requirements at the end of the process. The next sections focus on the software system ADAPTREF itself, including the classes, their relationships and their interactions in terms of the more abstract framework presented in this section.

## **4.2 Design of the simulator tool ADAPTREF**

### **4.2.1 Overall structure**

The software developed in this work either can be viewed as a stand alone module serving as a geometry manipulating tool, or it can be used in conjunction with a numerical simulation tool such as a finite element tool. Figure 4.3 shows a possible application and how it may interact with other software systems. When using it together with a mesh generator only, i.e. a software that generates a geometrical mesh on a prescribed domain, it serves as a device with which to modify the mesh according to certain criteria such as, for example, the maximum size of each mesh constituent. Additionally, the software-system ADAPTREF emerges as a natural driver for finite element programs. The latter generally lack of the control of the accuracy of the computed solution of the physical fields. On the basis of the

numerical framework developed in chapter 3 we now are able to overcome this long existing incompleteness in microsystem simulation. Figure 4.4 illustrates how the software system ADAPTREF can be put over a finite element software. Together with properly defined interfaces, its core operations consist of three

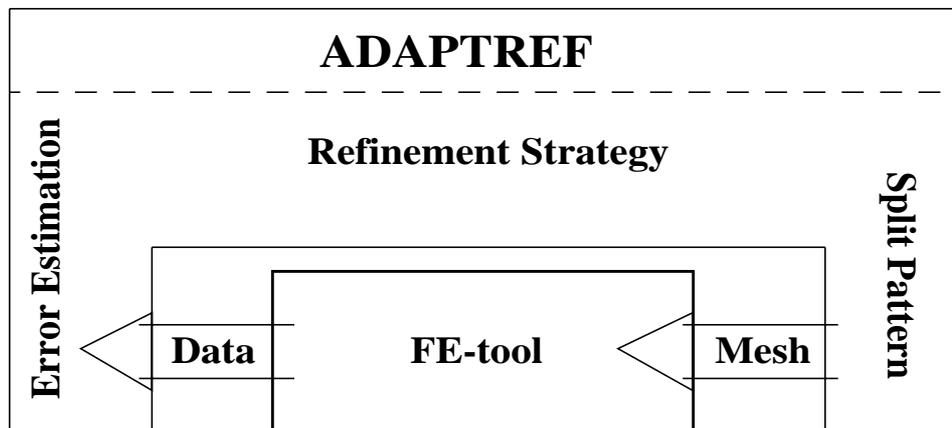


**Figure 4.3** Configuration of the global use-cases of the software system ADAPTREF

blocks:

- Error estimation of the solution computed by the FE-program
- Refinement strategy that manages the decision where the computational mesh has to be modified
- Split patterns which perform the geometrical modification of the mesh

The interfaces mainly serve as transfer gates for solution data and geometry data respectively. The architecture of the software system ADAPTREF is kept flexible such that it serves as a driver program for virtually any FE-tool or meshing tool as soon as the interfaces can be clearly defined. Moreover, the application of ADAPTREF is not restricted to the usage of numerical finite element programs, other numerical solvers such as, boundary element tools, can be used as well. The global architecture in terms of classes and their collaboration is displayed in Figure 4.5. The notation for documenting the classes and their relationships are taken from [83]. A straight line represents an association between classes. In C++ this means that in one class there is a reference of an object or an object is instantiated by its value from the other class and vice versa. The full bullet represents the fact that



**Figure 4.4** Using the software system ADAPTREF as an overhead to existing finite element tools. Major operations consist of error estimation, applying refinement strategies and applying split patterns. The most significant interfaces consist of solution data transfer from- and mesh data transfer to the FE-tool.

one class “has” an object of the other class, in the Booch terminology also known as aggregation. In C++ two classes are related by aggregation if one class has stored an object of the other class by instance. The third relationship is illustrated by an empty bullet and stands for the fact that one class uses another class. The class being used usually serves as a parameter of an operation of the other class. Central to the system is the class Mesh describing the geometry of the MEMS-device and serving as a framework for numerical simulation of multi-physical effects by a finite element tool. The mesh in turn is composed of a set of elements, by the notation introduced, there is an aggregation between the mesh and the elements. Each element then uses an error estimator based on the numerical solution of a PDE that is generated by the FE-Application class. The error estimator, of course, can not be set up without the knowledge of the PDE it corresponds to, it therefore uses the same PDE as the finite element application did before. The refinement strategy and the split pattern classes associated to the mesh, operate on the latter in such a way that it is modified. The full understanding of what is happening during the application of an ADAPTREF run can be obtained when considering an interaction diagram, sketched in Figure 4.10. Initially, elements are created such that a finite element mesh is set up representing the device geometry.

Of course, material properties are assigned within the geometry but this operation for clarity has been omitted in the figure. This point will be detailed later in section 4.4. At the same time a finite element application is instantiated which calls the mesh and the PDE in terms of the type and the boundary conditions.

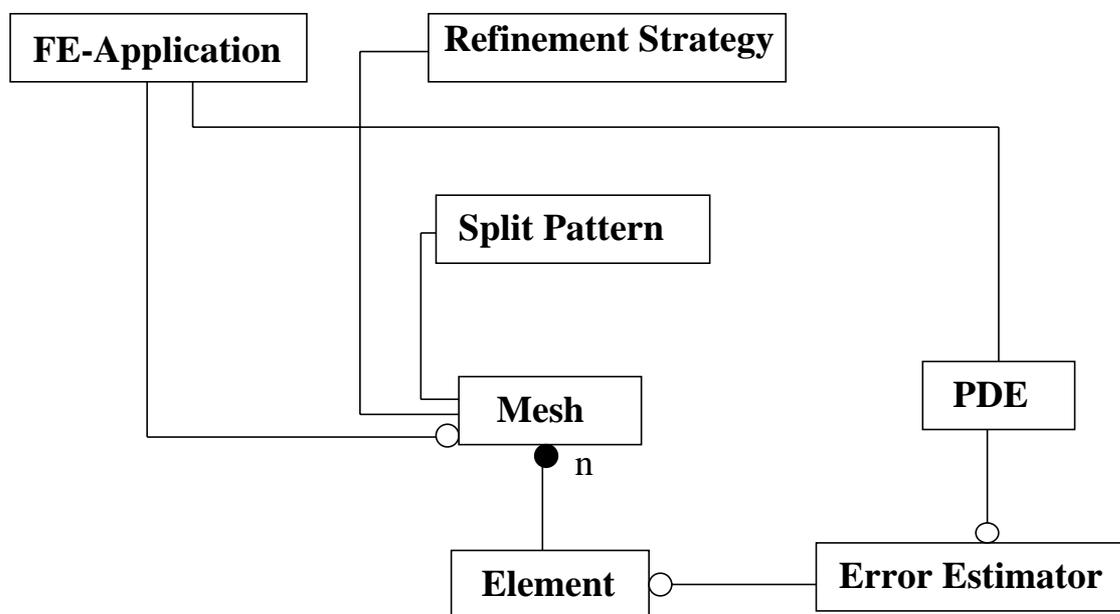
A run of the finite element tool is performed resulting in a set of solution data which then are passed over to the mesh. This might be well substantiated since a one to one relationship between physical field values and the geometry can easily be established. The object PDE remains until the object error estimator obtains knowledge of the partial differential equation and the boundary conditions associated to it (for simplicity we here assume the PDE to contain the boundary conditions as well, effectively there is a separate class for each of them). In the contrary, the lifetime of the FE-Application ends as soon as the solution data are passed to the Mesh which in turn hands it over to each of its constituents.

Along with the destruction of the FE-Application comes the instantiation of the element error estimators which compute each element's error value based on the PDE and the approximated physical field values now attached to the mesh constituents. The geometrical mesh data and the physical field data are separated in different classes as will be detailed below. Then, a refinement strategy object is evoked, calling the elements' error estimators and applying the information of their distribution to the mesh. In other words, regions within the mesh are identified to have higher or lower error values and thus the local element error information is made available globally. In the end of the cycle a SplitPattern is instantiated updating the old mesh by applying its geometrical routines to the mesh and thus generating a modified one. The cycle may be repeated until a stop condition is imposed. The major operations of the software system ADAPTREF sketched in Figure 4.10 are operations of the software system's main object AdaptrefApplication itself. Figure 4.6 displays the class diagram and the example code of how to apply the software on the highest level.

### 4.3 Mesh and Mesh Modification

The core of the ADAPTREF is the geometrical mesh and its constituents. We only have implemented simplicial elements and their extensions, that is, lines, triangles, tetrahedrons and prisms although the overall structure supports the imple-

mentation of any kind of polytopes in a straightforward manner. A design has been chosen to conform memory and speed requirements on the one hand and on the other hand the need to make mesh modifying easy has been met. Figure 4.7 shows the class hierarchy and the class relationships. The head of the hierarchy is represented by the class `ListItem`, a vector-type structure needed to assure fast access to the objects stored within the list. Moreover memory leakage is avoided if objects are stored in accurately defined blocks. A major issue when it comes to



**Figure 4.5** Major classes and their relations of the ADAPTREF software system.

the classification of the elements is how to ensure fast access to each single object and how to make the splitting of the geometrical object simple. By introducing an inheritance relationship between the two dimensional simplices, the triangles/tetrahedrons, and the parent class `BinaryTree` we are able to arrange the whole mesh in a binary tree structure. This in turn guarantees quick access to and generation of new mesh hierarchies when bisecting triangles or tetrahedrons. The single trees then are stored as list object such that a 2D or 3D simplicial mesh is represented as a list of binary trees. The reason for choosing that combination of binary trees and linear lists is closely related to the issue of keeping shape regularity of the

mesh as we have seen in Figure 3.4. A more general design would evoke the question of how to deal with constraints such as the interpolation on hanging nodes which was discussed in earlier works [63], [84].

Matters are not as simple anymore when dealing with prismatic elements since simple bisection usually generates hanging nodes and thus a binary tree structure would be useless. For thin structure simulation we only have to handle one-layered prismatic meshes such that bisection of elements does not result in the production of hanging nodes. Supporting the axiom of re-usage of existing classes we have combined the classes triangle and prism without imposing an inheritance structure on the latter into a class SuperTriangle. It will later turn out that this class is perfectly suited for the treatment of the mixed dimensional piezoelectrically active membrane structure where two dimensional triangles and three dimensional prisms are combined to form a single element. Of course, lower level

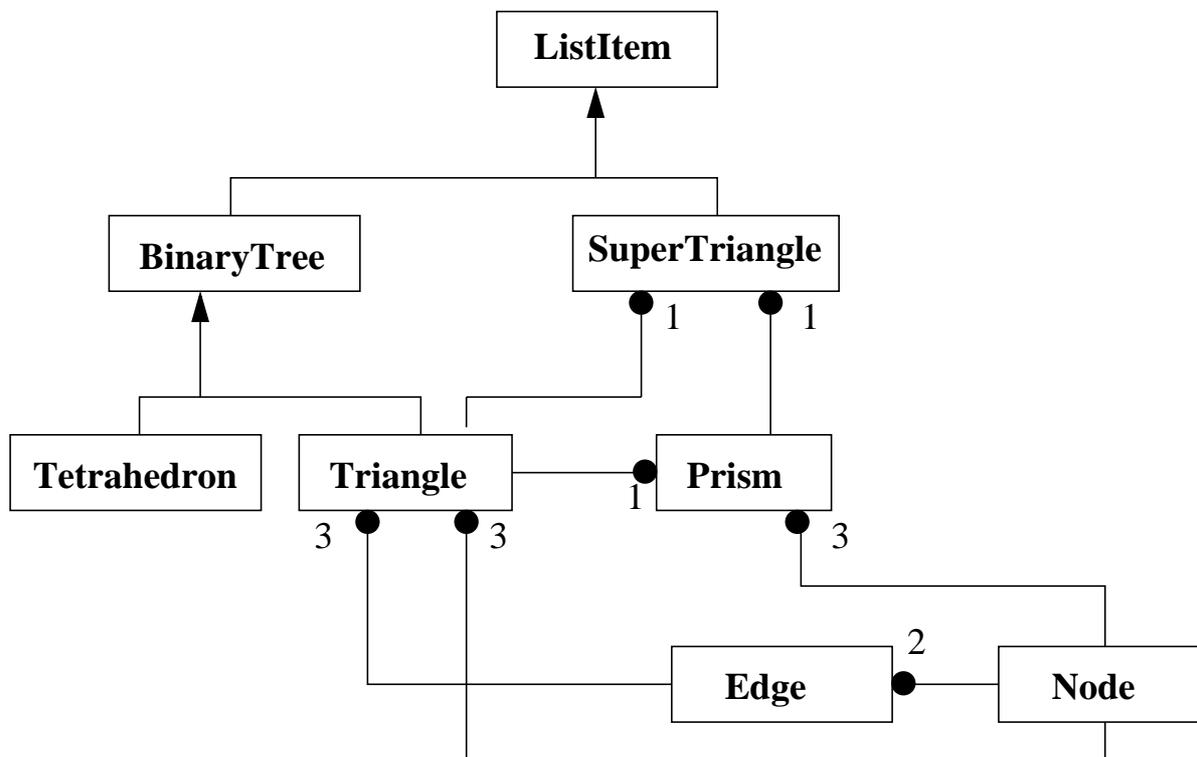
<b>AdaptrefApplication</b>	<i>RefinementStrategy myStrategy;</i> <i>SplitPattern mySplitPattern;</i>
Refinement Strategy	
SplitPattern	<i>AdaptRefApplication myAppl(</i> <i>myStrategy,</i> <i>mySplitPattern);</i>
Read/Write(FE-Application)	
Run(FE-Application)	<i>myAppl.Run( FE-Application);</i>

**Figure 4.6** Highest level object and example high level source code application

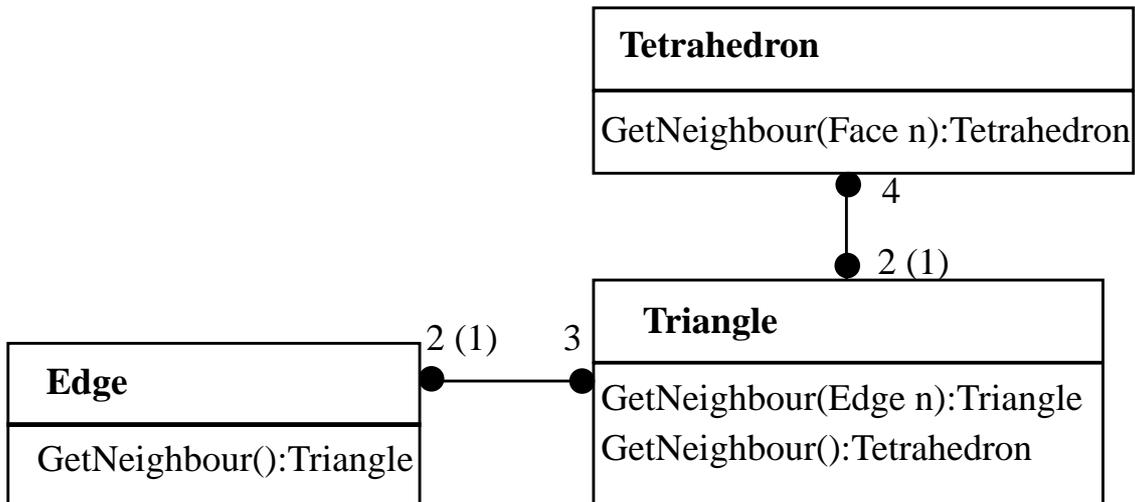
objects, such as Nodes and Edges are also stored in linear lists and these are related to the higher level geometrical objects by aggregation. To keep the illustration clear, the aggregation relations for the tetrahedron have been omitted. Besides the relations of the geometrical objects sketched above we should point out that there has to be introduced another connection between certain objects in order to perform the mesh modification properly: these are the neighbouring relationships. When splitting a simplex along its longest edge one or more neighbour simplices

have to be accessed in order to avoid the introduction of hanging nodes. The number of neighbours of a triangle is limited to exactly one while in three dimensions there is no limit for the numbers of simplices of a tetrahedron as shown in Figure 4.11. Therefore the mesh constituents have to be designed to allow access to all of their neighbours.

This is realized by assigning to each of the bounding  $D-1$ -simplices the  $D$ -simplex itself,  $D$  denoting the dimension of the simplex,  $D=2,3$ . Each of the bounding ( $D-1$ )-simplices thus has at least one and at most two  $D$ -simplices depending on whether it is on the domain boundary or not. Figure 4.10 shows the Booch notation of this important aspect of mesh representation. Note that a triangle's role is



**Figure 4.7** Class hierarchy and class relationship between major mesh constituents. The arrow denotes inheritance relationship according to the Booch notation. Aggregation relationships for the tetrahedron class are omitted for clarity reasons.



**Figure 4.8** Class diagram displaying the relation of the simplicial geometrical objects. Since accessing neighbouring elements quickly is essential to the software tool ADAPTREF, not only are edges attached to triangles and are triangles attached to tetrahedrons but also the converse is true: depending whether the simplex under consideration is on the domain boundary or not it is assigned one or two of the higher dimensional one, respectively.

twofold: in the first case it is an element itself and in the second case it only serves as a part of the boundary of a tetrahedron. The operation GetTetrahedron of course is disabled when the mesh is only two dimensional.

## 4.4 Materials and Thin Structures

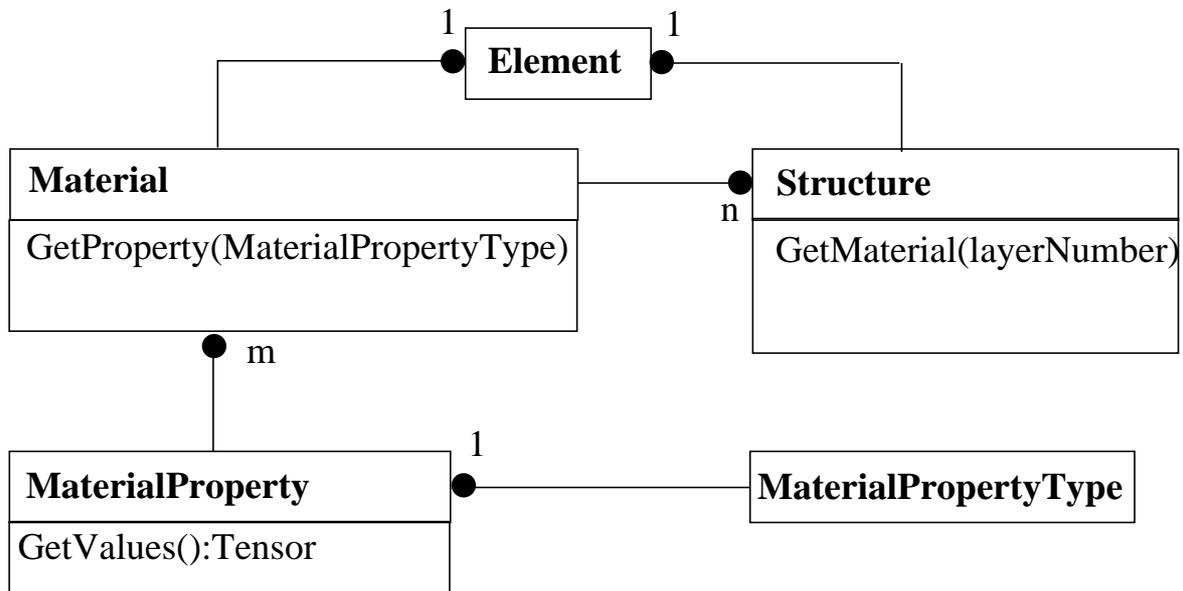
The design of the material database basically follows the one described in [63]. In the case that other FE-simulation tools than FEMEngine are used as a computational kernel, an appropriate interface has to be implemented without changing the structure of ADAPTREF's material database. Simulating microsystem components requires the following tasks to be met:

- material properties are scalars or tensors
- one selected material usually has many properties
- modeling and simulation of thin structures requires support for multi-material layers

The architecture of the material database is shown in Figure 4.9. The responsibilities of each class are:

Structure	A structure is a composite material. Within a structure each material component is assumed to be immobile with regard to the other materials, in other words, the components are supposed to be glued together. The main application of structures are multi-layered plates, membrane-like devices consisting of different material stacks with different thicknesses sandwiched together. The class Structure contains a set of Material objects which can be requested by the class user. Also the class offers information about component thicknesses and positions within the structure. A chief feature of the class is the computation of material tensor moments of a certain order across its vertical dimension such that a material property can be returned as if the structure consisted of only one material.
Material	This class is central to the material database. It is the representant of a single material and comprises all those properties that have been specified. In turn these properties can be accessed by the class user. An error is reported if for a given physical problem a certain property has not been defined. A key feature of the material class is its ability to determine the reduced values of a specified property. This is required when dealing with thin structures, as we saw in chapter 3.
MaterialPropertyType	This is an enumeration type and is used to identify one particular MaterialProperty such as heat conductivity or thermal expansion. When requesting the value of a material property from a material object a MaterialPropertyType is passed to state the type of property that is requested.
MaterialProperty	This class denotes the value of one particular property of the material. The associated MaterialPropertyType enables the user of the class to determine what kind of property this object is.

The material data are made available to the numerical error estimating modules by the elements. The elements are assigned either one particular material or one particular structure. It is also possible to extend the material database to environment dependent material properties, for example a Young's modulus which is temperature dependent [63].



**Figure 4.9** Class relationship of the material database. A material might possess different material properties such as thermal expansion or thermal conductivity. A structure consists of a stack of different Materials of which each has a certain position within the stack identified by the layerNumber. An element, the building block of a device geometry, may be either assigned a structure or a material.

## 4.5 Fields and Boundary Conditions

A major difference between a common finite element tool and the software module ADAPTREF presented here is the fact that the mesh is dynamic in the latter case. An important aspect thus is how to prolongate persistent information from a previous mesh to a new one. To illustrate this we take the example of an element which is split: Here the material properties of the new elements simply obtain those adherent to the parent element. Or, consider, by way of illustration, a spatially defined source function, such as a heat source, which also has to be present at the same spatial positions in a new mesh.

The situation is quite similar when it comes to treating boundary conditions. As soon as a new node is introduced on the domain boundary the correct boundary conditions have to be assigned to it. This is easy when the new node's old neigh-

bours share the same conditions for a given field (as a convention we assume that boundary conditions in many finite element tools are defined on nodes). However, things turn more complicated when the neighbouring old nodes do not share the same conditions for a particular field, for instance the first one is assigned natural BCs whereas the second one has essential BCs. The management of difficulties of that kind is left to the class `Comp(utational)Node` and its associates as illustrated in Figure 4.12. A `CompNode` includes all the information the split pattern needs to equip newly generated nodes with the new boundary conditions on the one hand and with information concerning predefined regions a certain set of nodes belong to, on the other hand. It is noteworthy that, while the relation between the boundary condition class and the `CompNode` class is one to one, there is an aggregation between the `CompNodes` and the geometrical nodes such that several `CompNodes` may be attached to one single node. This is because different boundary conditions associated to different physical fields are specified at the same geometrical position. In the following we give a listing of the individual class items and the tasks they have been assigned.

<code>CompNode</code>	Derived from the expression <code>computational node</code> the class serves as an interface class between the persistent physical properties defined within specified spatial regions and the evolving mesh structure. It is thus responsible for the correct prolongation of boundary data and information about regions a node may belong to. Major operations consist of passing the field type and the boundary conditions the <code>CompNode</code> is linked with to the user of this class, mainly the geometrical split pattern.
<code>Field</code>	An instance of this class denotes one particular physical field such as the temperature or one of the displacement fields. A field is one to one related with a boundary condition and thus with a <code>CompNode</code> . For each node it has a defined value.
<code>FieldType</code>	An enumeration type needed when a field is requested from a <code>CompNode</code> object.
<code>BoundaryCondition</code>	The class <code>boundary condition</code> represents the boundary values specified for a given physical problem at a certain part of the device geometry's boundary. It is thus closely related to a set of nodes (in case the finite element tool requires BCs to be specified at nodes) which are combined on their part in an object of the class <code>Region</code> . A boundary condition has one particular field type and it is aggregated to one region which means that all nodes in that region share the same boundary condition. Since it is con-

nected to exactly one field type the main issue of an object of this class is passing the BC's value.

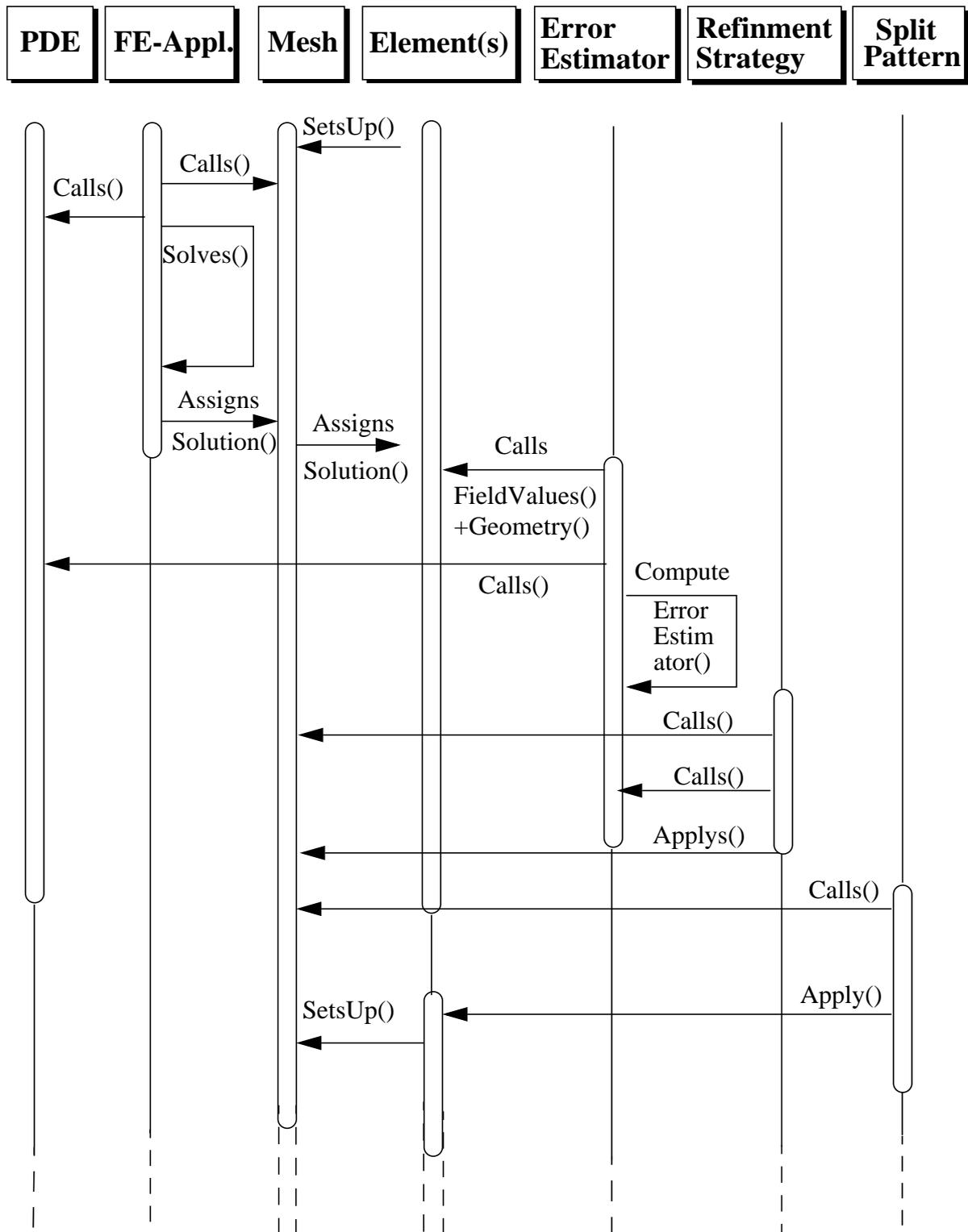
**BoundaryConditionType** An enumeration type needed to decide whether we deal with natural BCs or essential BCs for a given node. Neumann or Dirichlet Boundary Conditions are BoundaryConditionTypes for instance. Some difficulties arise when introducing a new node into the mesh in between two old nodes having different BoundaryConditionTypes for a given field. Natural conditions then are chosen as a default for the new node in case the edge's first old node has natural BCs whereas the second old node has essential BCs for a given field.

**Region** The region class usually serves as a container for a set of nodes which share some specific properties. It is not restricted to nodes that share a certain boundary condition nor is it restricted to nodes as such: also elements might be collected within an instance of this class. Creating new nodes and elements requires them to be assigned the regions (by the appropriate identifiers) in order to provide a correct interface for the finite element application.

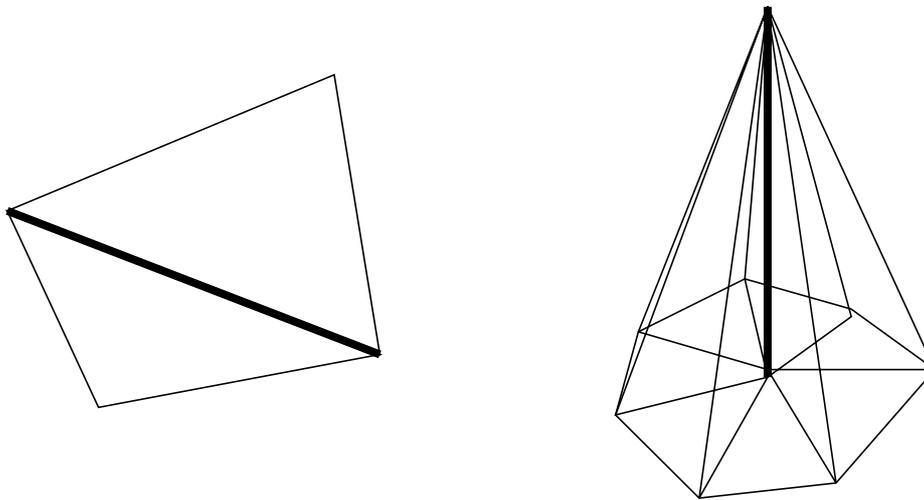
We finally want to point out that when new mesh constituents are generated in the split pattern all the important properties connected with boundary conditions and regional information are transferred automatically to the new nodes. This is why the split pattern not only uses the pure geometrical information of the nodes but as well has to be aware of the data which we collected in the CompNodes. Implementation of this procedure in a sophisticated way presented here thus releases the program end user from tediously writing program macros for the prolongation of data, a common but poorly way in commercial software tools such as ANSYS [35].

## 4.6 Error Estimators and Mesh Refinement

The main feature of the software system ADAPTREF is its ability to automatically modify a mesh describing a device geometry. The mesh modification is solely based on the optimization of the accuracy of the physical field to be computed. And at the same time, computational resources such as processor time and memory requirements are kept at a minimum. This can only be achieved by exploiting the interaction of a system pillared by three classes, namely the refinement strategy class, the split pattern class and the class responsible for setting up contributions to the element's error estimator, denoted by Comp(utational)Ele-



**Figure 4.10** Interaction diagram of the major objects used during a simulation run cycle. The lifetime of each object(s) is highlighted by an empty thicker bar.



**Figure 4.11** Accessing neighbours of simplicial elements along a predefined edge (bold): An issue of OO software design.

ment. The latter class takes into account that an element has two sets of responsibilities: first, the element is the building block of the geometry of the device being simulated. Second, the element is used as a finite element and thus it must provide for a convenient representation of finite dimensional function spaces.

While finite elements in the conventional sense require the support for shape functions, we here only need a representation of polynomials of different orders. This means that an element, before some error can be retrieved, has to be equipped with solution data representing some field's values by a finite dimensional functional object, in our case, a polynomial. To one and the same geometrical element there might be attached several polynomials each representing a different physical field. The polynomials and the operations that are in charge of setting up the elements error contribution are collected in the class `CompElement`.

Disposing of element error values we are able to set up some refinement strategy, a conception to decide where the mesh should be modified. Finally, a split pattern can be instantiated that performs the mesh modification from a geometrical point of view, see also Figure 4.10. Figure 4.13 describes the mechanisms with which these classes collaborate. The responsibilities of each class are as follows:

<code>CompElt</code>	This is the class which comprises the non- geometrical aspects of the finite element. In contrast to the conventional finite element
----------------------	--

tools fields are just represented by polynomials which in turn are reconstructed by the finite element definition and the field's values. This requires the solution of a linear system which is at most 21x21 for the highly sophisticated Argyris triangle. Moreover, an instance of this class sets up the individual error contribution for the field it is (non-ambiguously) associated to. An object of the class element has as many CompElts as there are fields defined for it. Thus, when calculating the element's entire error contribution each CompElt is requested to hand over its part. For instance, the CompElement "LinearTriangle" associated to the field Temperature has to provide for the jump errors of the heat flux,

$$\|[\mathbf{n} \cdot \boldsymbol{\kappa} \cdot \nabla T_h]\|_{L^2(E)}$$

across the element interfaces. Each CompElt, independently of which field it is associated to, has member functions that provide for different differential operators acting on the functions which represent the field. The elements do not need to know of which kind the CompElts are, as an example of polymorphism, the correct operation for every CompElt is determined not until runtime. The class CompElt is a base class to its children classes ranging from linear tetrahedrons over to Argyris triangles to linear prisms.

### RefinementStrategy

An instance of the class RefinementStrategy operates on the geometrical elements and determines whether they should be split or not. Usually this is done on the basis of an element error estimator computed by the CompElements. Of course, global strategies are allowed to be applied where no error estimator is required. Among others, a common strategy is the maximum strategy, and is done in the following way [46]: Suppose that for a given mesh a solution and an error estimator for each element has been computed. Put

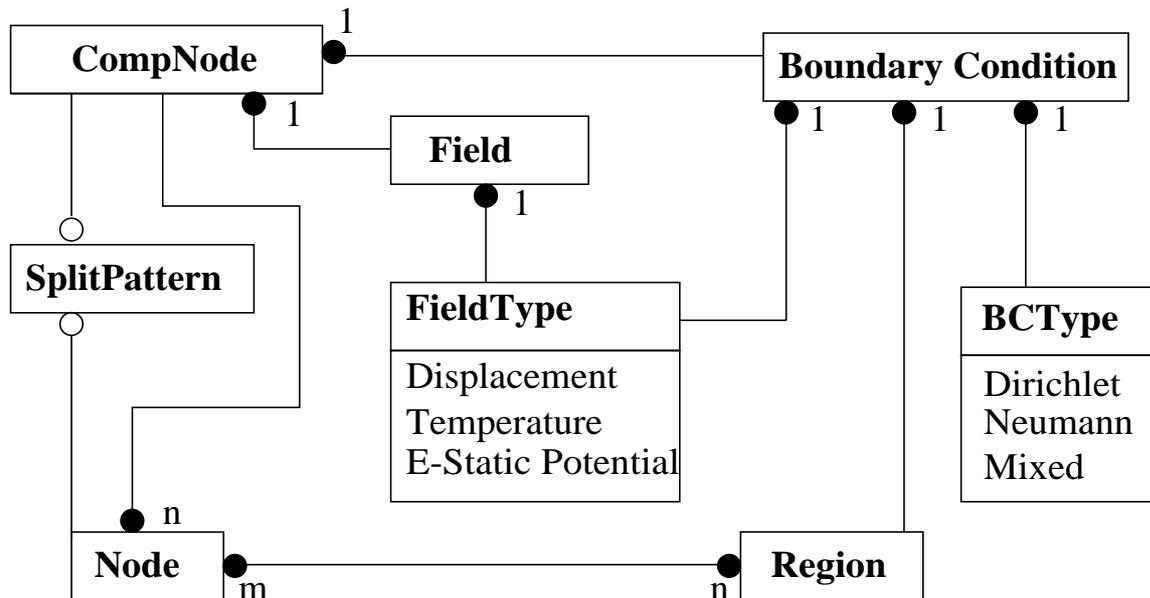
$$\eta := \max_{T \in \mathcal{S}} \eta_T$$

and split an element  $T$  if  $\eta_T \geq \zeta \eta$  where  $\zeta$  is a prescribed threshold,  $0 < \zeta < 1$ . This strategy, applied iteratively, would continue indefinitely. A halting condition is simple to add: stop if  $\eta \leq \eta_{\text{accept}}$ . From heuristic arguments we know that among all partitions of a linear finite element discretization, that one is optimal which equilibrates the error. I.e., the errors in all elements should be made equal. This can be achieved by the maximum strategy. All different types of strategies are predefined by the program user and require the enum-type **StrategyType**.

### SplitPattern

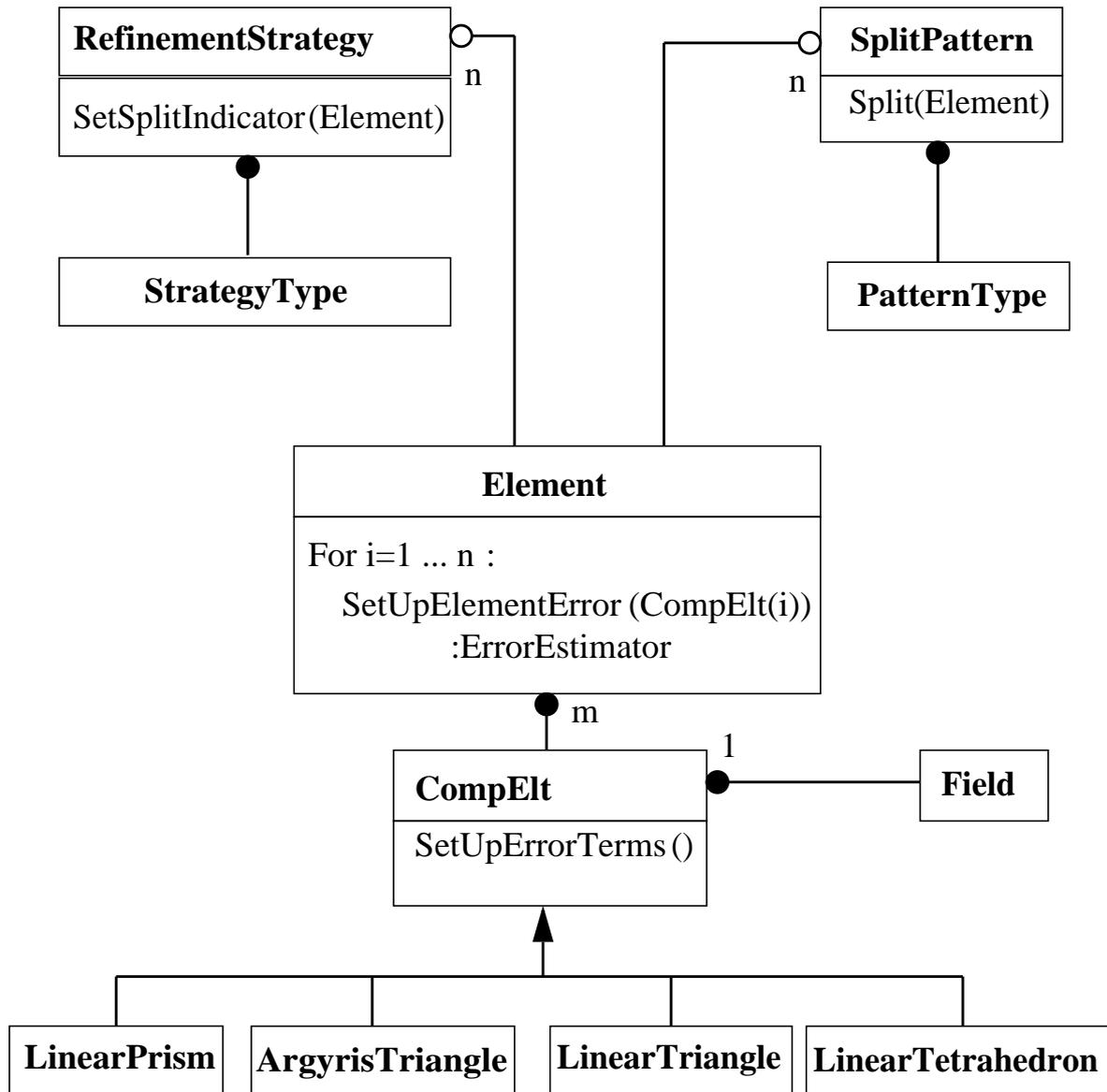
This class is responsible for the methods how to modify the mesh from a geometrical point of view. The way in which the refinement process is performed, from a geometrical point of view, depends on the kind of elements present in the mesh. Difficulties arise from keeping shape regularity and from handling hanging nodes. Many rules have been established for the splitting of simplicial mesh constituents, such as e.g. the “red”, “green” or “blue” refinement for triangles. We use a recursive algorithm for triangles which is based on the longest edge bisection. It has been shown that only a finite number of different angles occur during the refinement process and therefore shape regularity is guaranteed [111],[112],[113]. The algorithm is shown in Figure 4.14, whereas the splitting of a 2D triangular patch is displayed in Figure 4.15. As for the RefinementStrategy a SplitPattern has to be declared by the program user by an enum type **PatternType**.

We would like to point out that a recursive split pattern has also been implemented



**Figure 4.12** Information that has to be prolonged during the adaptive refinement procedure is stored in a class called **Comp(utational)Node**. One particular geometrical **Node** may carry many different **CompNodes** according of the numbers of the physical fields defined on it. A major task of the **CompNodes** is to handle the boundary conditions by entering them into the split pattern.

for tetrahedral grids, but here one has to encounter the difficulty that convergence of the algorithm is restricted to a small class of geometries the tetrahedrons are embedded in, see [114]. This in turn means that the variety of such applications is limited. Nevertheless we present a thermal analysis of an elbow in the next chapter where the recursive split pattern for tetrahedrons has been applied.



**Figure 4.13** Relationships among classes representing the numerical core of the software on the one hand and the impact of the mesh manipulating classes on the other hand. The generation of each element's error estimator is a nice illustration of an OO-feature called polymorphism: An element does not need to know what special kind of CompElts it is dealing with, at run time each of them hands over its own contribution to the total element error autonomously.

```

recursiveRefine(triangle_element)
{
  do
  {
    if neighbour has non-compatible
      refinement- edge

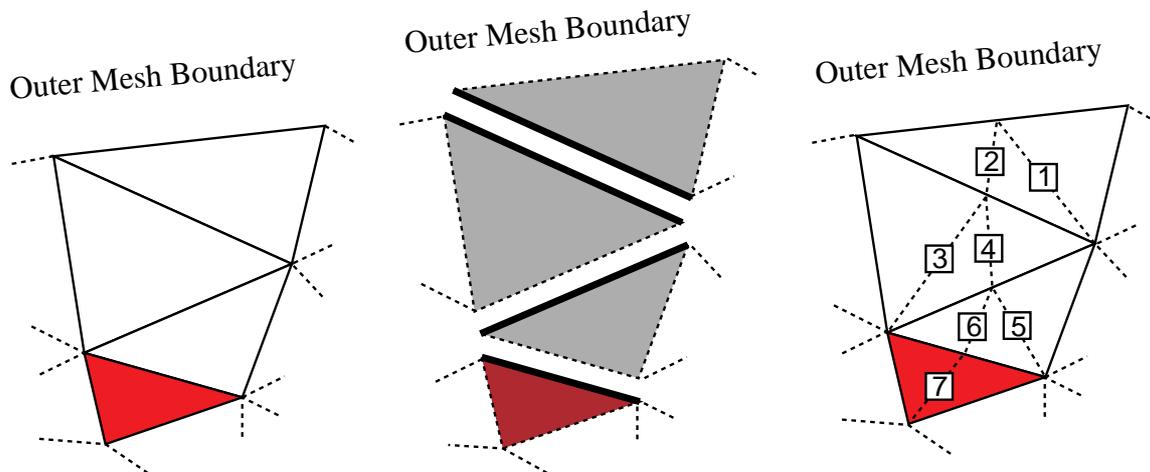
      recursiveRefine(neighbour)

  }until neighbor has compatible refinement-edge

  bisect both triangles at the refinement-edge
}

```

**Figure 4.14** Recursive refinement algorithm for triangular grids.



**Figure 4.15** The elementary mesh at the left consists of 4 triangles. The shaded triangle is marked to be split along its longest edge. The longest edges of each triangle are marked by thick lines in the center figure. The dashed lines at the right show how the recursive bisection algorithm produces new triangles, continuing through the neighbourhood until a first edge can be split (in the worst case, this happens only when the boundary edge is reached), then backtracking. The boxed numbers label the order in which new edges are introduced on the return path of the recursive algorithm (depth 4).

# 5 SIMULATION

This chapter is dedicated to present the areas of application of the software module ADAPTREF. It serves as a tool with which to accurately simulate the physical behaviour of microsystem components. Through its ability to perform auto-correction in the sense that it adapts the computational mesh to the physical fields that are desired, it relieves the MEMS designer of the burden of finite element mesh design. Moreover, with this software at our disposal we overcome the general drawback when using common finite element tools such as [35],[36]. These disregard field singularities introduced by the mere shape of the device or such that are due to material interfaces. We demonstrate the action of ADAPTREF and thus the accuracy controlled MEMS simulation procedure starting with thermal problems and closing with a piezo-electrically driven membrane problem.

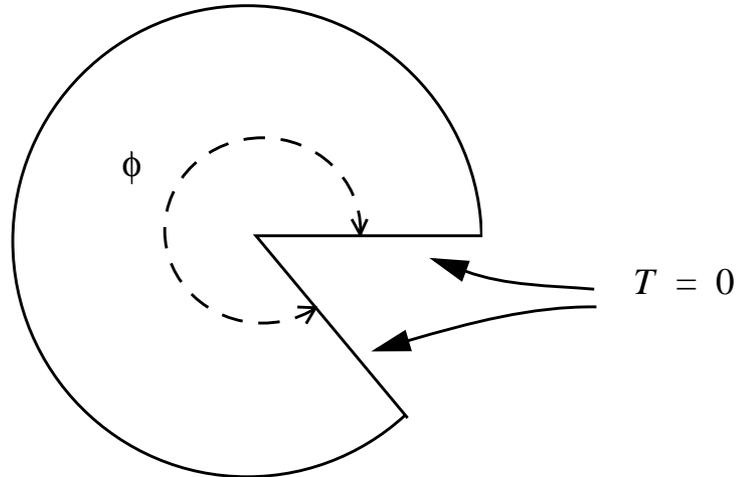
## 5.1 Thermal Analysis and typical singularities

It is well known that simulation domains with concave corners give rise to field singularities [78],[2]. We will use this fact in order to provide an illustrative entry into what the adaptive finite element tool ADAPTREF is able to perform. We consider a general wedge shaped domain  $\Omega$  as sketched in Figure 5.1. We assume that there is no heat source within the domain and the Poisson equation describing the temperature distribution reduces to a Laplace equation for which we fix the following boundary conditions:

$$\begin{aligned} -(\nabla \cdot \nabla T) &= -\Delta T = 0 & \text{on } \Omega &= \{(r, \theta): 0 \leq r < 1, 0 < \theta < \phi\} \\ T(r, 0) &= T(r, \phi) = 0 & 0 \leq r < 1 \\ T(1, \theta) &= \sin\left(\frac{\pi}{\phi}\theta\right) & 0 \leq \theta \leq \phi. \end{aligned} \tag{5.1}$$

Since for the function

$$T(r, \theta) = r^{\frac{\pi}{\phi}} \sin\left(\frac{\pi}{\phi}\theta\right) \quad (5.2)$$



**Figure 5.1** A wedge shaped domain for thermal analysis. The angle  $\phi$  is a variable and thus the domain gives rise to field singularities in the very corner of the domain.

the equation

$$\begin{aligned} \Delta_{r, \theta} T(r, \theta) &= \frac{\partial^2 T}{\partial r^2} + \frac{1}{r} \frac{\partial T}{\partial r} + \frac{1}{r^2} \frac{\partial^2 T}{\partial \theta^2} \\ &= \frac{\pi}{\phi} \left( \frac{\pi}{\phi} - 1 \right) r^{\pi/\phi - 2} \sin\left(\frac{\pi}{\phi}\theta\right) + \frac{\pi}{\phi} r^{\pi/\phi - 2} \sin\left(\frac{\pi}{\phi}\theta\right) - \left(\frac{\pi}{\phi}\right)^2 r^{\pi/\phi - 2} \sin\left(\frac{\pi}{\phi}\theta\right) = 0 \end{aligned} \quad (5.3)$$

holds for any  $\phi$  we see, together with the boundary conditions, that the harmonic function (5.2) is a solution to problem (5.1). However, difficulties arise when one wants to approximate the temperature field by the governing PDE and the boundary conditions in the context of (3.68) where it was assumed that a bound for the linear interpolation error is given by some  $L^2$ -integral of the function's second derivatives. In our case, the derivatives of the function considered become singular at the origin as soon as  $\phi$  exceeds the value of  $\pi$  since then

$$\frac{\partial}{\partial r} T(r, \theta) = \frac{\pi}{\phi} r^{\frac{\pi}{\phi}-1} \sin\left(\frac{\pi}{\phi}\theta\right) = \frac{\pi}{\phi} r^{\lambda} \sin\left(\frac{\pi}{\phi}\theta\right) \quad (5.4)$$

with  $\lambda < 0$ . Still, first derivatives are square-integrable and thus the function is in  $H^1(\Omega)$ , as we saw for example in 3.1.2, but second derivatives are not square integrable anymore since

$$\iint_{r=0} \left(\frac{\partial^2 T}{\partial r^2}\right)^2 r d\theta dr \propto \int r^{2(\pi/\phi-2)} r dr = \int_{r=0} r^{\frac{2\pi}{\phi}-3} dr \quad (5.5)$$

does not exist for values of  $\phi$  exceeding  $\pi$ .

Although the a priori error estimation fails we can perform some a posteriori error estimation for the poisson equation with a zero heat source according to 3.3.2. Since the interpolation functions which we choose are linear, the error estimator (3.155) only consists of terms involving jumps of the heat flux across element edges and deviations from the prescribed flux at the Neumann boundary

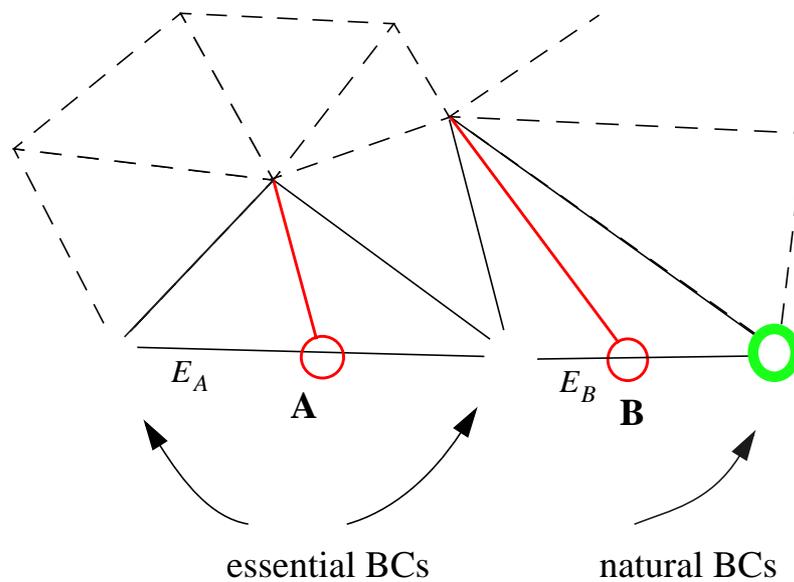
$$\begin{aligned} \eta_T^2(T_h) = & \sum_{E \in S, E \subset \Omega} \frac{1}{4} \|[n_E \cdot \kappa \cdot \nabla T_h]\|_{L^2 E}^2 h_E \\ & + \sum_{E \in S, E \subset \Gamma_N} \|g_N - n_E \cdot \kappa \cdot \nabla T_h\|_{L^2 E}^2 h_E, \end{aligned} \quad (5.6)$$

$T_h$  denoting the interpolated temperature field. One should note that this type of error also could be interpreted as discontinuities of thermal fluxes, or when dealing with an electrostatic problem, jumps of the electrical field which both are forbidden by the laws of physics.

Figure 5.3 displays the thermal analysis for a simple L-shaped single material geometry with different Dirichlet boundary conditions applied on the outward edges and the inward edges respectively. The absolute values of the field gradients and thus the jumps of the thermal fluxes across element boundaries are displayed in the last row. The initial triangulation shows large jumps of the gradient values between interelement borders. During the refinement process the gradient value field is smoothed and thus jump errors are reduced. Although some values seem to be high we can state that the error contribution also takes into account the

dimension of the elements, see (5.6). Therefore, the connection between the strategy to equidistribute the element's errors and the mesh refinement can be clearly identified.

Another important type of singularities emerges when different types or values of boundary conditions meet. As an example we consider the polygonal boundary of some domain which has been assigned different boundary conditions: essential BCs directly border on natural BCs at a given point of the triangulation. This difficulty evokes the treatment of *open sets* in a mathematical sense since it would have no meaning specifying both natural *and* essential BCs in one single point. The matter becomes of a tremendous importance when it comes to the prolongation of boundary data during the refinement process. Which type of boundary conditions should be applied to newly created nodes, **A**, **B** as in Figure 5.2?



**Figure 5.2** Singularities due to the coincidence of different BCs. According to ADAPTREF's convention the new node **A** will have the same essential BCs as the corner nodes on the edge  $E_A$ . The newly created node **B** however, is assigned natural BCs, so the edge  $E_B$  is treated as an open set.

The software tool ADAPTREF has an internal definition of open and closed sets and rules how to treat them during the refinement process as we have detailed in section 4.5. By default, natural boundary conditions always correspond to open sets unless otherwise stated. The reason simply is that natural boundary conditions enter the finite element formulation by an integral to be taken over the boundary whereas essential boundary conditions directly influence the finite dimensional ansatz-space which is defined via the nodal values.

We illustrate this topic by a coaxial-shaped structure having constant temperature on the inner boundary circle and another constant temperature on the most part of the outer circle except a closure where no heat flux is allowed and thus homogeneous Neumann BCs apply, as can be seen in Figure 5.4. Besides the singularities caused by concave corners we can observe two other singularities at the points where different BCs meet. During the refinement process the temperature jump gets steeper and steeper in the vicinity of the nodes under consideration. This is for the reason that mathematically the natural BC forms an open set that extends to the essential BC node infinitely close. When looking at the triangulation of the simulation domains in Figure 5.4 we observe that the mesh quality improves remarkably by means of the smallest angles which occur. This is due to the usage of the recursive splitting algorithm presented in Figure 4.14.

A recursive splitting algorithm is also used for adaptive mesh refinement in three dimensions, displayed in Figure 4.11. The splitting procedure, however, is much more complicated since for a given tetrahedron's longest edge there might be an arbitrary number of other tetrahedrons which form the neighbours at that given edge. We present the results for the same thermal problem as we already did in this chapter's first example together with an extension into the third dimension. Here we have to deal with the difficulty that a part of the boundary is assigned natural BCs although all of its corner nodes are determined to have essential BCs. This is the case for the upper left triangular face being a part of the elbow's thermally insulated face as can be seen in Figure 5.5. Not surprisingly, major refinement occurs at the re-entrant corner, or, more precisely, along the straight line that represents the 3D-version of the concave corner. A major drawback when it comes to the refining of three dimensional simplicial grids is the fact that the recursive algorithm presented in Figure 4.14 which is smart in the sense that it keeps shape regularity and avoids hanging nodes, only terminates for a small class of initial tri-

angulations [114]. One could overcome this difficulty by carefully choosing the initial discretization of the device. This can be achieved for example by embedding the tetrahedrons into cubes which requires the rather restrictive decomposition of the domain into equally sized cubes.

Another way how to handle the refinement of simplicial meshes in three dimensions would be the enormously time consuming re-meshing of the whole device geometry as it is done for example in [35]. With these considerations it comes in handy that certain aspects of structural behaviour can be analyzed by using the reduced two-dimensional thin plate model presented above.

### 5.2 Balancing error contributions

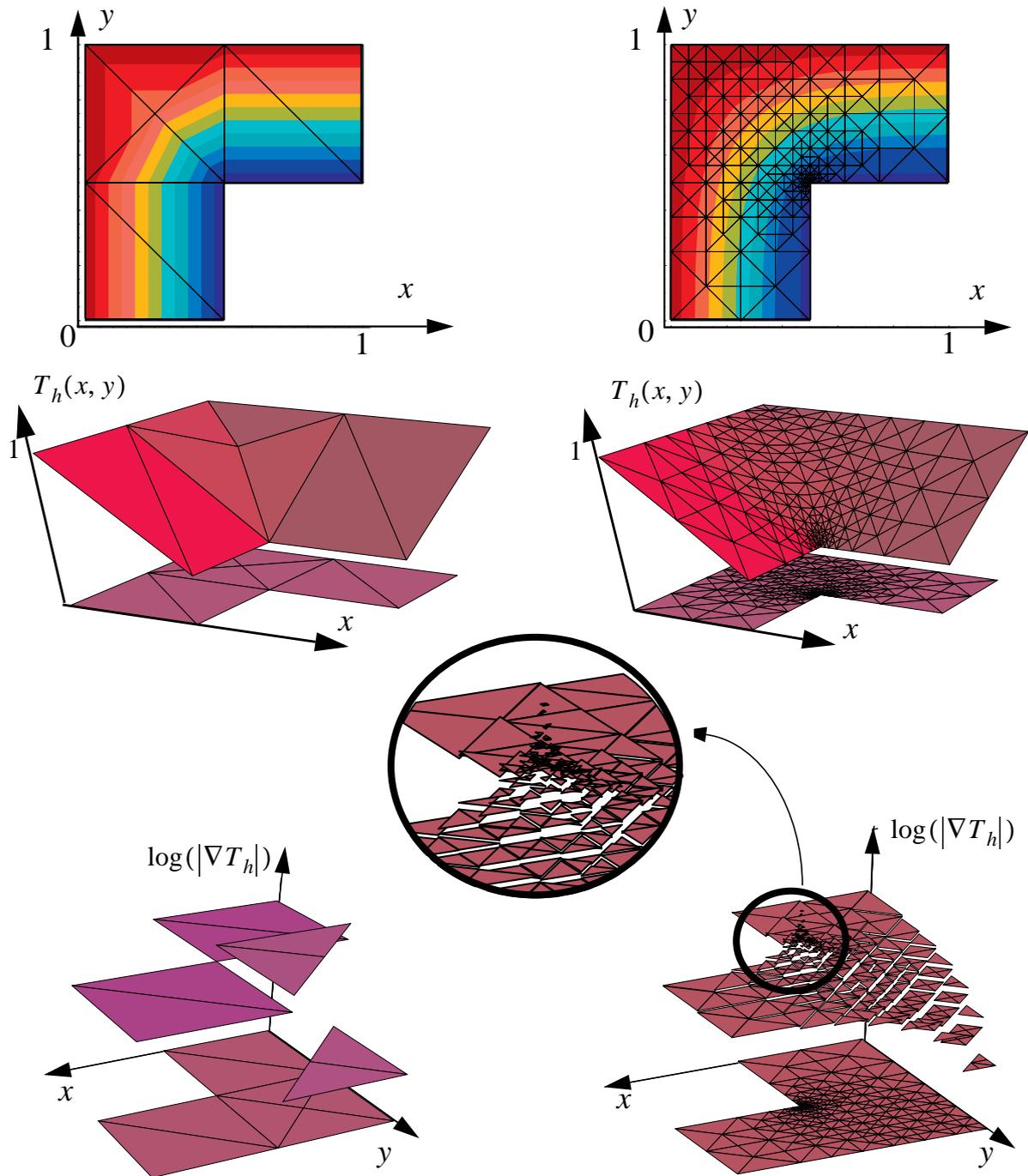
The choice of the finite elements has to meet the following fundamental requirements:

- they have to be in the same space in which the variational problem is posed
- physical conservation principles must not be violated

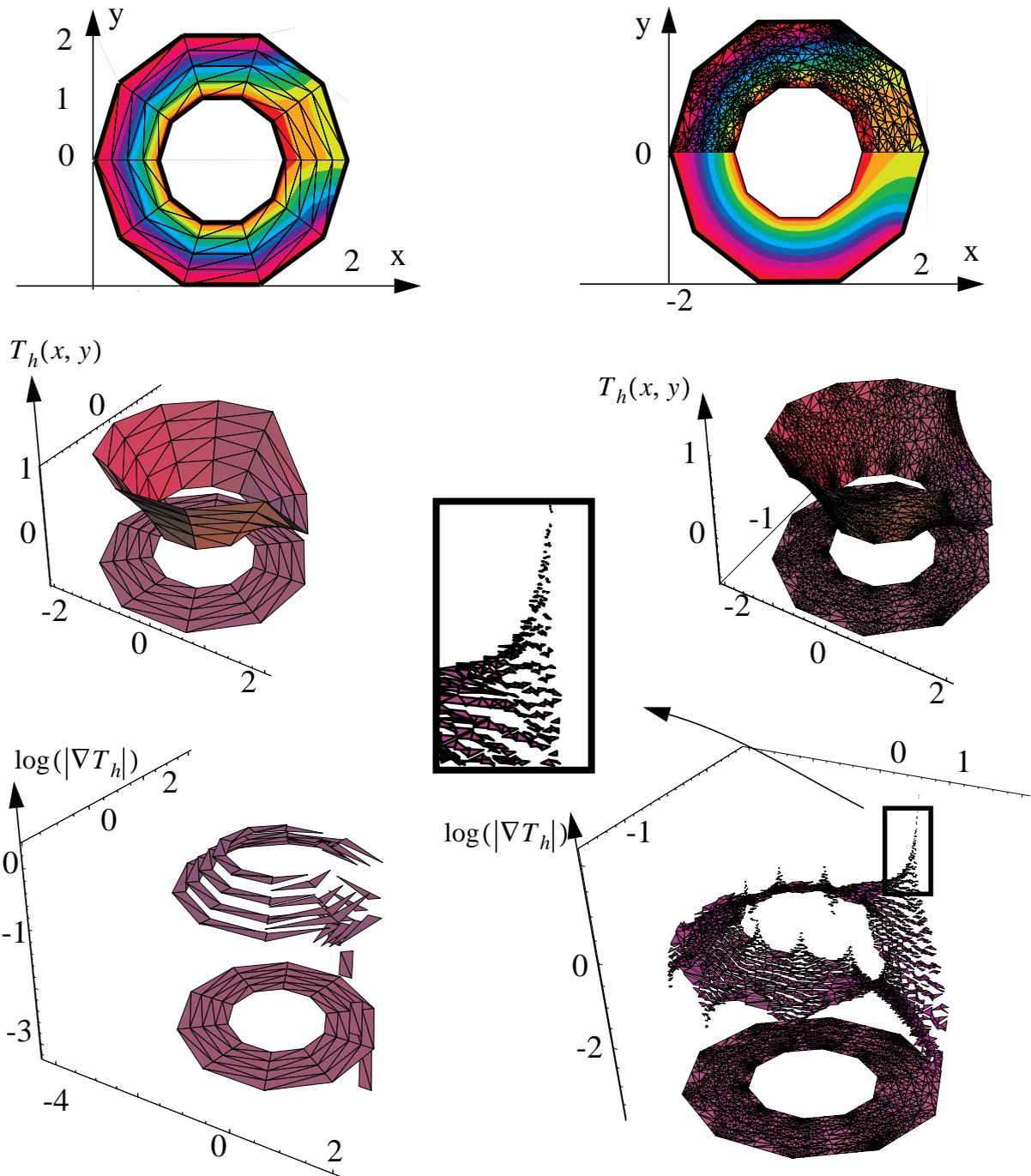
When it comes to the simulation of thermo-electro-mechanically active thin structures one has to encounter the difficulty that on the one hand elements of class  $C^1$  have to be used in order to match the conforming requirements of the Kirchhoff-Love plate model. This model is perfectly suited for very thin structure bending behaviour and thus a priori avoids the locking phenomenon. On the other hand, when using for example, the Argyris triangle, or the Bell triangle, first and second order derivatives are enforced to be continuous at the element nodes. Computing the temperature field on a 2D domain by using the Argyris element -what would be breaking a fly on the wheel- therefore would enforce continuous temperature gradients which at a material interface  $MI$  would violate the principle of thermal flux conservation since there the relation

$$\kappa_1 \nabla T_h|_{MI} \neq \kappa_2 \nabla T_h|_{MI} \quad (5.7)$$

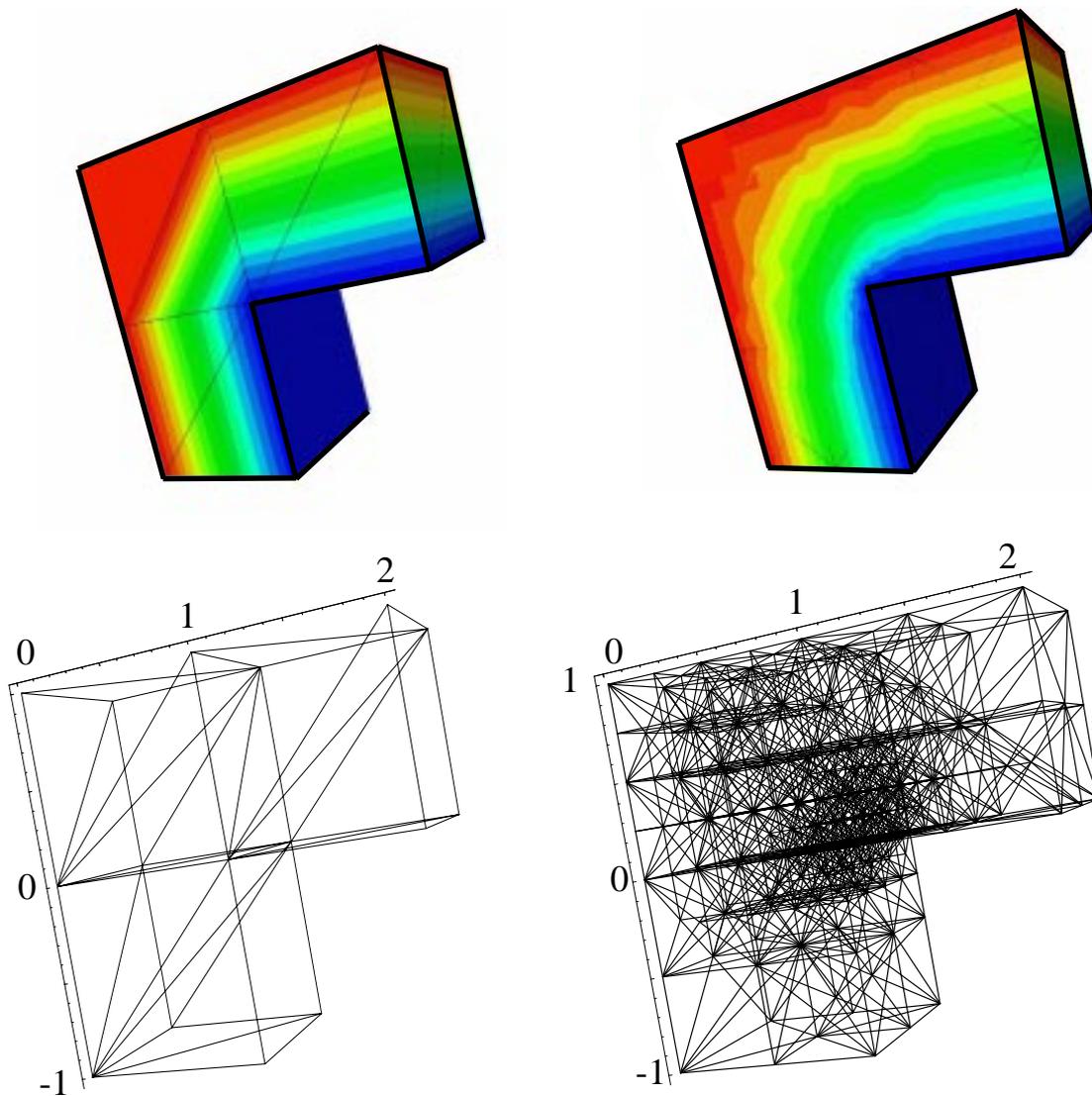
holds,  $\kappa_1$ ,  $\kappa_2$  denoting the heat conductivities of the different materials. Nevertheless, the residual error estimator presented in the previous chapter can cope with



**Figure 5.3** Thermal analysis on a domain geometry with a re-entrant, concave corner. Both columns display the interpolated temperature field as a projection of the field into the plane, a 3D graph, and a plot of the gradient jumps across inter-element boundaries, from top to bottom. The left column represents the starting coarse mesh and the right column the highly refined mesh.



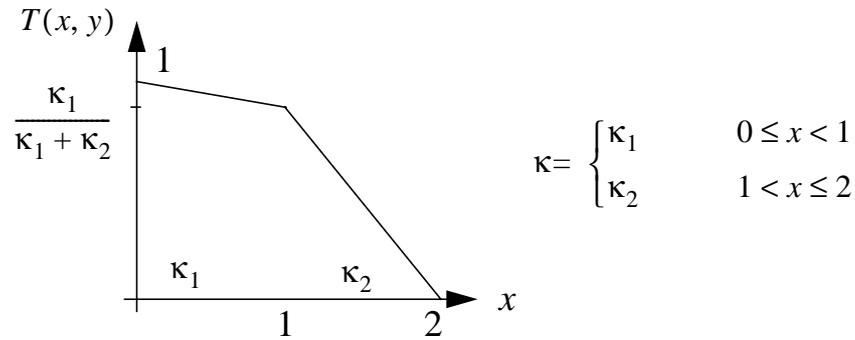
**Figure 5.4** Thermal analysis on a coaxial shaped domain. Besides singularities caused by concave corners the major errors occur in regions where different BCs meet. The adaptive procedure “resolves” these singularities as can be seen in the bottom right graph. To clearly display the temperature in the vicinity of one boundary singularity a part of the refined mesh has not been displayed.



**Figure 5.5** 3D-thermal analysis of an elbow. Dirichlet BCs have been applied on the outer and on the inner boundaries whereas homogenous Neumann BCs apply on the back face and on the front face. On the left the initial temperature distribution and the underlying coarse mesh is displayed. On the right hand side the field and the corresponding tetrahedral mesh after 9 refinement steps is shown.

errors introduced by elements which are too smooth. We demonstrate this for a simple Dirichlet thermal problem where constant temperatures of  $1K$  and  $0K$  are applied on the left and the right hand side of a rectangular 2D domain which con-

sists of two different materials and has a length of  $2m$ . The exact solution is sketched in Fig. 5.6



**Figure 5.6** Exact solution of a symmetric 2D thermal problem. The temperature field's gradient is discontinuous at the material interface.

and is piecewise linear

$$T(x, y) = \begin{cases} \left( \frac{\kappa_1}{\kappa_1 + \kappa_2} - 1 \right) x + 1 & 0 \leq x < 1 \\ \left( \frac{-\kappa_1}{\kappa_1 + \kappa_2} \right) x + \frac{2\kappa_1}{\kappa_1 + \kappa_2} & 1 < x \leq 2 \end{cases} . \quad (5.8)$$

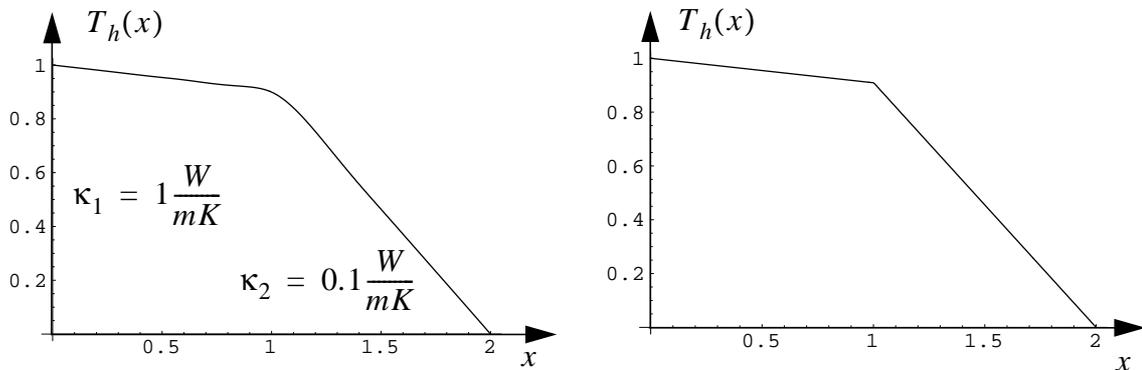
This exact solution can be obtained by linear triangular elements. Since function (5.8) is identical with the one sought in the VBVP (3.73) given by the bilinear form

$$a(T, U) \equiv \int_{\Omega} \nabla T \cdot \boldsymbol{\kappa} \cdot \nabla U dx = (\nabla T, \boldsymbol{\kappa} \cdot \nabla U)_{\Omega} = 0 \quad (5.9)$$

there is no difference between the weak solution and the approximated solution and thus the error estimator vanishes.

Matters are different when this problem is approximated by the continuously differentiable Argyris triangle. As in (5.7), errors are introduced by imposing discon-

tinuous fluxes at the material interface nodes and across elements. The detection of the error and how the solution is improved is displayed in Figure 5.7.



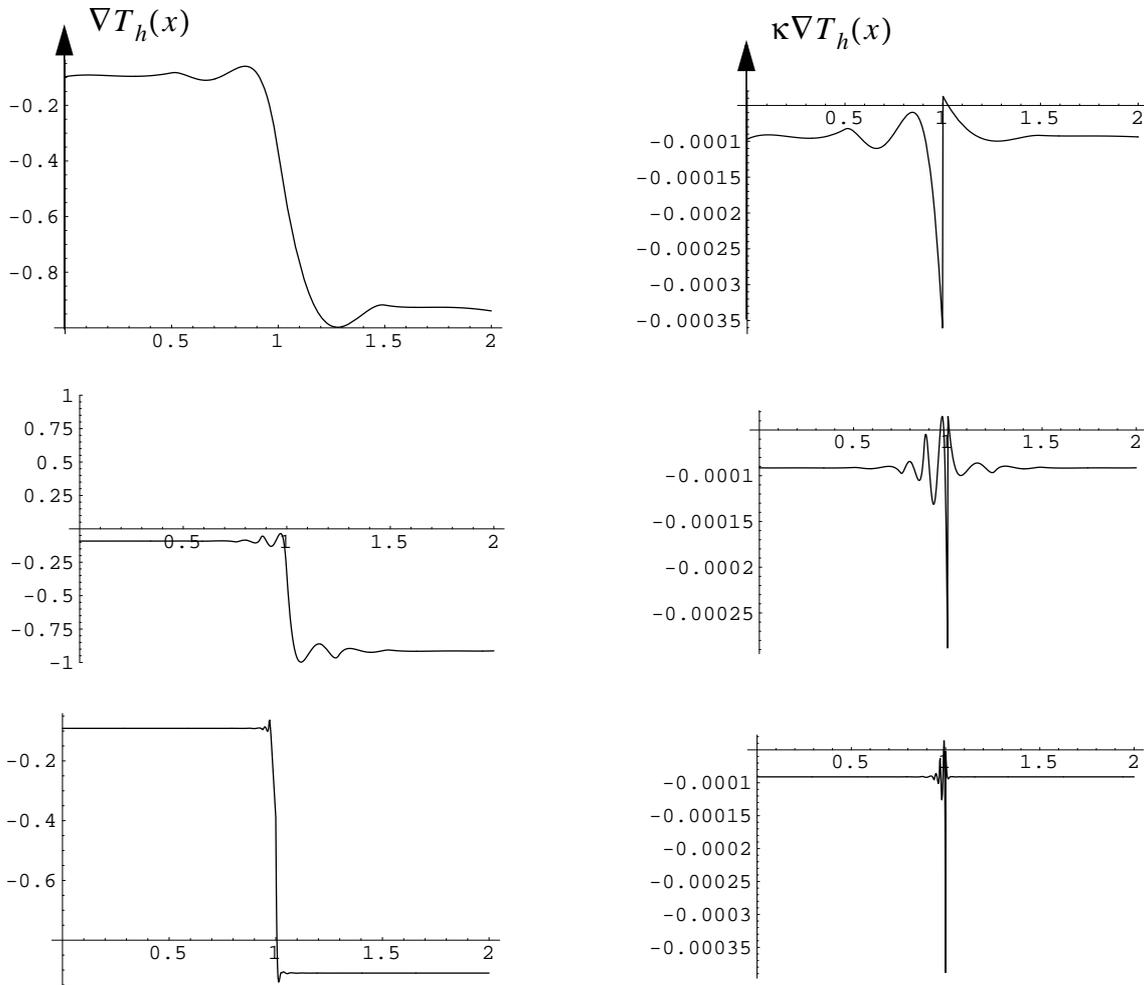
**Figure 5.7** Temperature along a cross section of the rectangular domain interpolated with smooth Argyris triangles. The left curve is obtained by a coarse grid and the right curve is the temperature after 10 refinement steps.

Not only the temperature field itself but also the first order derivatives tend towards the exact solution during the refinement process. The thermal fluxes, which according to (5.8) are supposed to be constant over the whole domain, and the temperature gradients are displayed for different refinement steps in Figure 5.8. One should note that in the same way in which the continuous temperature gradient is attempted to be eliminated, the jump of the heat flux reduces to a set of measure zero. The main error contributions in this case are inner element force deviations

$$\|\nabla \cdot \boldsymbol{\kappa} \cdot \nabla T_h\|_{L^2(T)}^2, \quad (5.10)$$

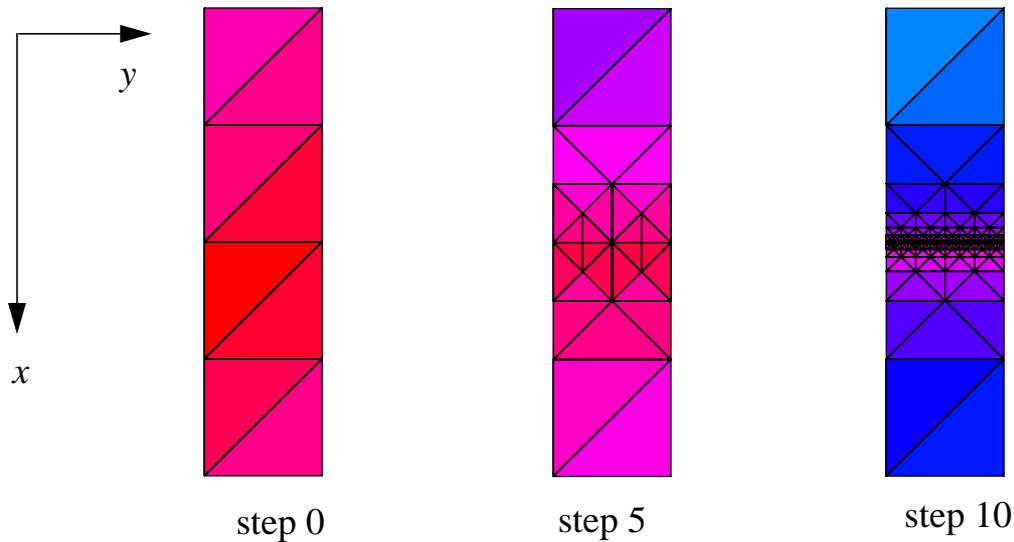
since, in contrast to linear elements, second order derivatives of the fifth order polynomial do not vanish. These error contributions are displayed in Figure 5.9.

Another example where physical continuity requirements are violated by elements too smooth can be demonstrated by a cantilever, clamped at the left end and



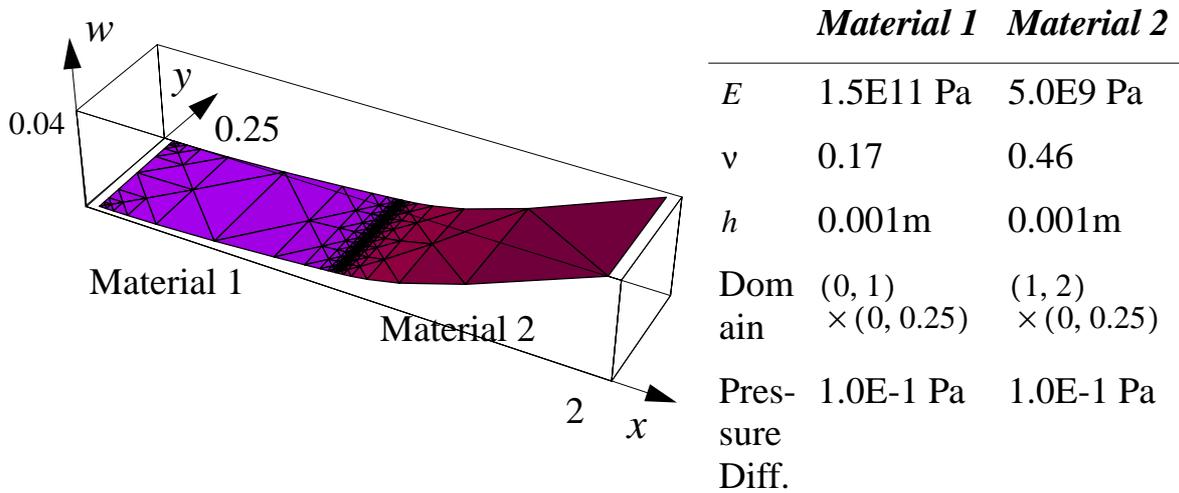
**Figure 5.8** Temperature gradients (left) and thermal fluxes (right) along the rectangular domain's cross section at  $y = 0$ . The curves display the fields in the starting triangulation and after 5 and 10 steps, from top to bottom. The initial thermal flux misleadingly is discontinuous and is squeezed into the correct continuity in the refinement limit.

consisting of two different materials which is deformed due to a constant pressure difference acting in vertical direction. When computing the displacement field by conforming Argyris triangles, the imposed continuity of second order derivatives at the triangles' corners results in discontinuities of bending moments that physically are not allowed. However, the adaptive mechanism of ADAPTREF based on residual error estimation tries to balance the errors caused by discontinuities in order to yield the optimal solution. The displacement of the cantilever is displayed



**Figure 5.9** Inner element force deviations for the thermal Dirichlet problem for different refinement steps. The error values are highest for the initial mesh (reddish) and are reduced during the refinement process (bluish).

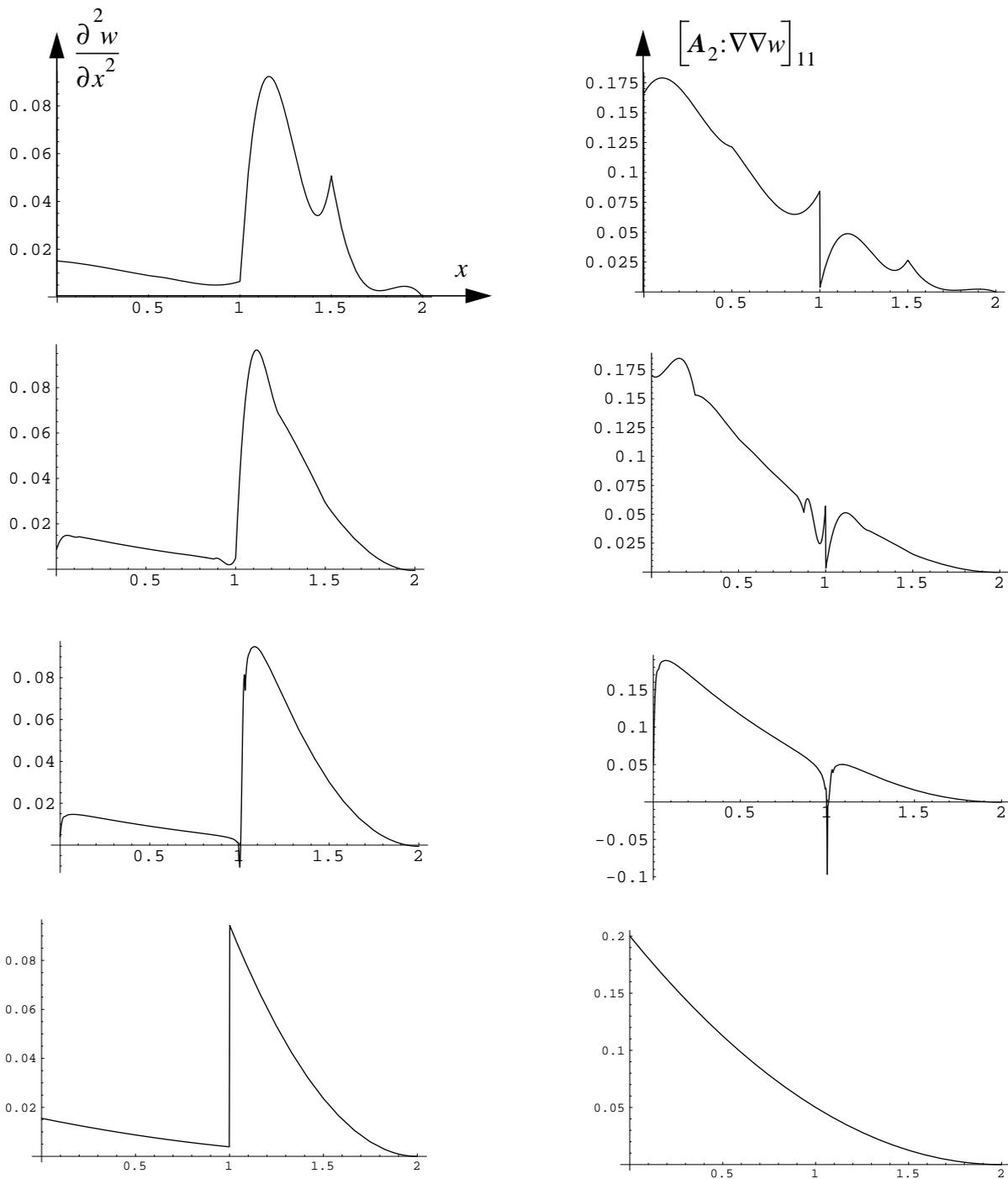
in Figure 5.10. The second order derivatives of the out-of-plane displacement field are continuous whereas the corresponding bending moments have jumps as displayed in Figure 5.11. In the limit of the adaptive refinement however, the approximated bending moments tend towards that of the local analytical solution. The latter, on a one dimensional domain can be obtained by imposing the appropriate boundary conditions and continuity requirements to a fourth order polynomial which obeys a bi-laplacian equation. Its second order derivatives and the corresponding bending moments are sketched in the last row of Figure 5.11. We finally want to point out that besides the previously discussed singularities also those inconsistencies can be handled that originate from finite element subspaces which are too smooth. Therefore the user of the software ADAPTREF also is relieved from the burden of thinking about the smoothness of the finite element polynomials as soon as conformity is guaranteed.



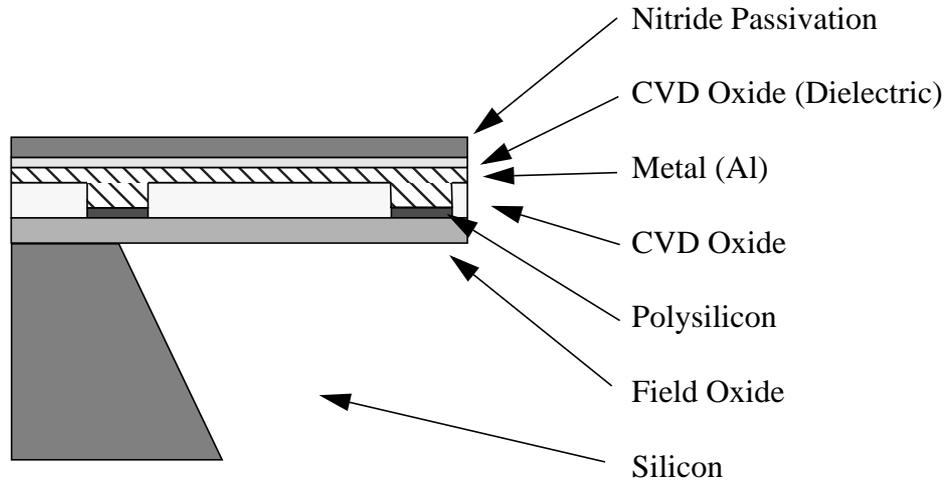
**Figure 5.10** Deflection of a beam consisting of two materials which is clamped at one side. The computation of the deflection is made on the mesh obtained after the 14th refinement step. The table shows the corresponding material properties.

### 5.3 Mechanical Analysis

Thin structures are central to micro-electro-mechanical systems. For example the micromachined complementary metal oxide semiconductor (CMOS) Atomic Force Microscope (AFM) beam [85], formed from the dielectric layers and some silicon, undergoes minute deflections. The stress caused by the deflections is detected in piezo-resistors at the beam's base. A typical example of thin structures widely used in MEMS-technology is given schematically in Figure 5.12. Multi-physically active thin structures here are fabricated by a CMOS process together with an etching step in order to release the beam or the membrane structure [86],[87].



**Figure 5.11** Second order derivatives of the out-of-plane displacement field in  $x$ -direction along a cross section of the cantilever at  $y = 0$ , left, in the 0th, 5th and 14th refinement step and the analytical result, from top to bottom. The right column represents the corresponding bending moments.



**Figure 5.12** Schematic view of a released beam/membrane structure fabricated by a CMOS process together with an etching step.

Typical layer thicknesses of the different layers are summarized in Table 5.1.

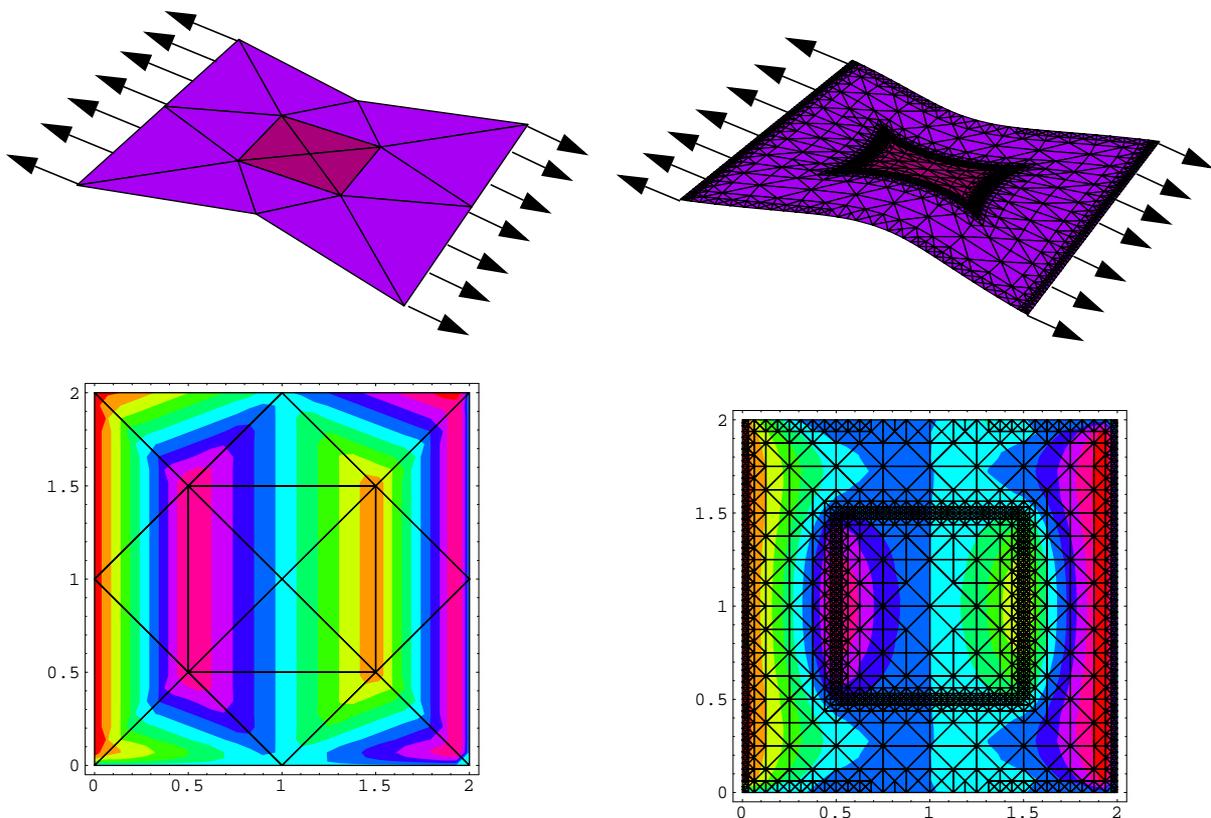
**Table 5.1** Typical thicknesses of layers resulting from a CMOS process.

<i>Layer</i>	<i>Thickness</i>
Field Oxide	$0.55\mu m$
Polysilicon	$0.3\mu m$
Metal (Al)	$0.6 - 1.0\mu m$
Passivation	$1.0\mu m$

Thicknesses of the oxide layers typically vary between  $0.05 - 1.0\mu m$ . The resulting thin plate structures to simulate then have an aspect ratio of the order of  $10^2$  when assuming a lateral dimension of the same order of magnitude in microns. The mechanical deflections are in the order of the thickness and thus the linear plate model presented in chapter 3 fits excellent [27]. In the following we are going to examine the accuracy control for different deformation behaviour of such multi-layer structures.

### 5.3.1 The stretched membrane

As a first example, we consider a homogenous single-layered rectangular membrane which is partly prestressed in its center region. Fig. 5.13 shows the



**Figure 5.13** Exaggerated deformation of a membrane together with the x-component  $u_x$  of the in-plane displacement field  $u$ . The membrane is under constant horizontal surface traction at the left and the right side of the structure, before (left) and after (right) 13 cycles of adaptive mesh refinement. The upper and lower edges are completely free. The center region is under tensile prestress. Significantly refined regions are visible at the transition from the prestressed to the non-prestressed region and where the vertical constraints are applied.

deformed structure modeled by the initial triangulation and the same structure

after 13 mesh refinement cycles. Constant surface traction is applied on the right and left boundary of the membrane (note that the surface in our flat model is reduced to a one dimensional line) and no load acts in the vertical direction. Since the structure is single-layered, all error terms appearing in (3.211) which include first order moments of the reduced elastic constants or prestresses, such as for example

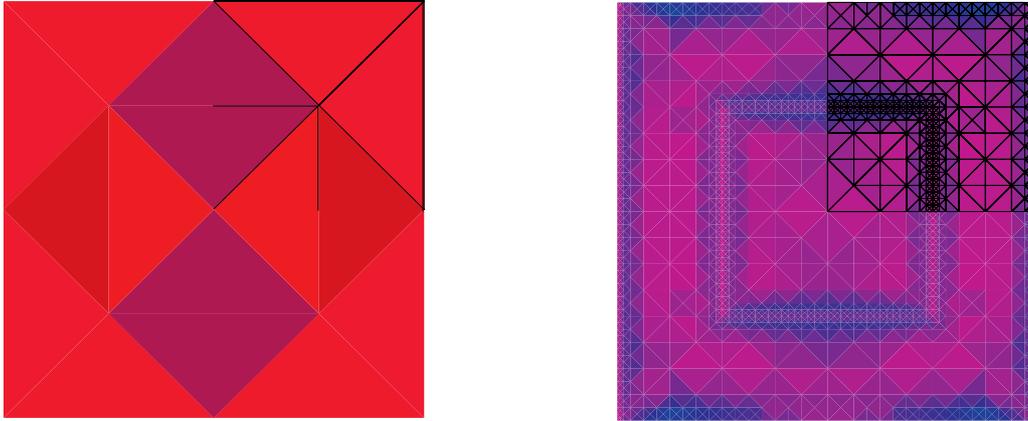
$$\|\mathbf{n} \cdot [\boldsymbol{\sigma}_{u_1}]\|_{L^2(E)}^2 = \|\mathbf{n} \cdot [\mathbf{A}_1 : (\nabla \mathbf{u}_h)^S]\|_{L^2(E)}^2 = 0 \quad (5.11)$$

identically vanish. This is because the first order moments are obtained by integrating an odd-ordered function over a transverse domain centered at the z-direction's origin. Moreover, as there is no force which causes the out-of-plane displacement to be nonzero, all terms involving the vertical displacement vanish identically. So we are left with terms which contain only derivatives of the in-plane displacement field  $\mathbf{u}_h$  contracted with zero order moments of the elasticity tensor and the jumps of the zeroth order prestress:

$$\begin{aligned} & \|\nabla \cdot (\mathbf{A}_0 : (\nabla \mathbf{u}_h)^S) + \mathbf{f}_0\|_{L^2(T)}^2 \\ & \|\mathbf{n} \cdot [\mathbf{A}_0 : (\nabla \mathbf{u}_h)^S]\|_{L^2(E)}^2 \\ & \|\mathbf{n} \cdot [\boldsymbol{\sigma}_0^{ext}]\|_{L^2(E)}^2. \end{aligned} \quad (5.12)$$

Taking into account that the horizontal displacement field  $\mathbf{u}$  is interpolated by the Argyris-polynomials and thus is continuously differentiable in the domain ( $\mathbf{u}_h \in C^1(\omega)$ ) no jumps along interior edges occur. The only non-vanishing error contributions in this case then are in-plane bending errors in the element interiors, and these decrease when we move from the starting coarse mesh to the level 13 refined mesh, see Fig. 5.14.

The errors due to deviations of computed boundary values from prescribed values at natural boundaries (situated at the top and the bottom of the membrane) are shown for mesh level 0 and 9 in Fig. 5.15. The errors due to jumps across the prestressed zone edges are shown in Fig. 5.16. The decrease of the total error (which is the sum of the single contributions) during the refinement process is illustrated in Fig. 5.17.

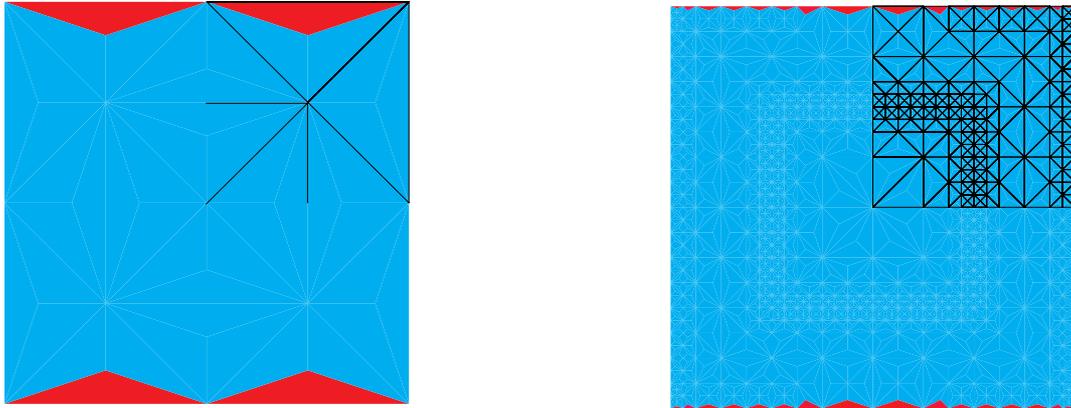


**Figure 5.14** Reduction of inner-element in-plane bending moment errors during the refinement process. For each triangle we assign one error colour of the specified quantity. Red shading denotes high error values whereas bluish denotes zero element errors. The adapted mesh plot at the right clearly shows the tendency of the refinement strategy to equilibrate the element errors when combined with the other error contributions. For clarity only a part of the underlying mesh is displayed.

### 5.3.2 Fully clamped membrane

Error contributions due to different material stacks and inconsistencies in approximating the out-of-plane displacement field are best demonstrated when a transverse pressure difference acts on a fully clamped membrane. The membrane consists of different material stacks which, for simplicity, we assume to be single-layered to make the first order moments of the in-plane tensors vanish. Slightly modified, this material configuration is often found in pressure sensor MEMS [90]. Fig. 5.18 shows the deformation of the membrane before and after 14 refinement cycles.

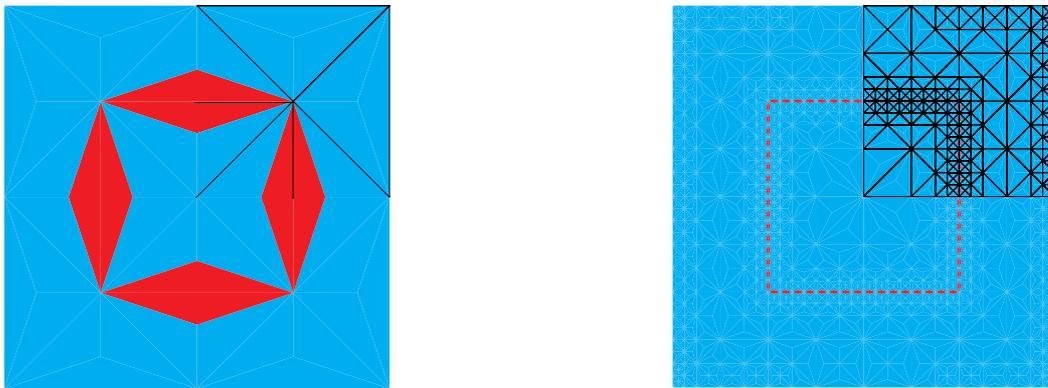
The in-plane displacement is assumed to be zero. Thus terms in (3.210) which contain  $\mathbf{u}_h$  are neglected. Since no prestress is specified, the corresponding terms




---

**Figure 5.15** Decrease of stress-couple errors across element edges up to refinement step 9. For each triangle we assign three error colours located at the triangle's edges and representing the jumps of the specified quantities across the edge. Here, only errors at the non-essential boundaries are non-zero.

---



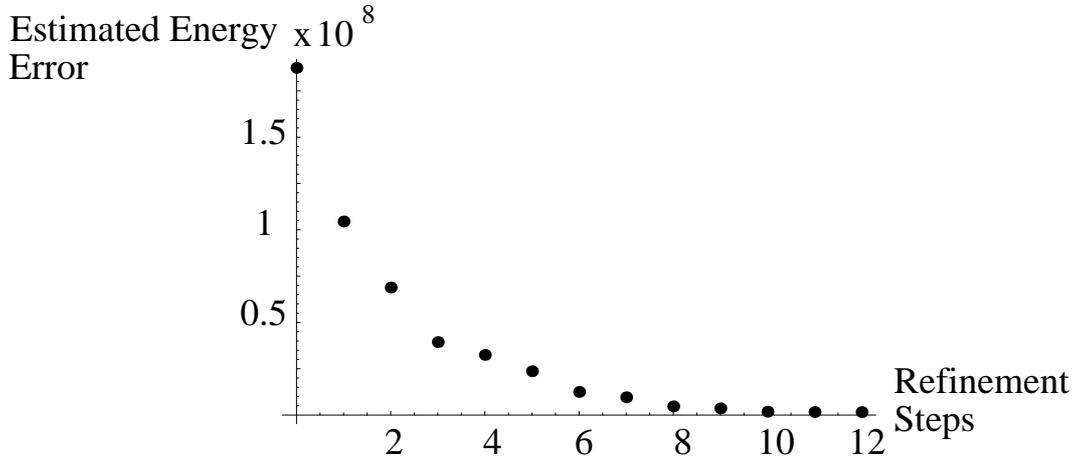

---

**Figure 5.16** Decrease of prestress jumps along the transition region of prestress to the non-prestress part of the membrane up to refinement step 9. For each triangle we assign three error colours located at the triangle's edges and representing the jumps of the specified quantities across the edge. Here, only errors at the transition edges are non-zero (red-dish).

---

also vanish. We are left with terms involving second order moments of the elasticity tensor contracted with the derivatives of the transverse displacement field.

---



**Figure 5.17** Decay of the total estimated error during the refinement process of the partly prestressed membrane of Fig. 5.13.

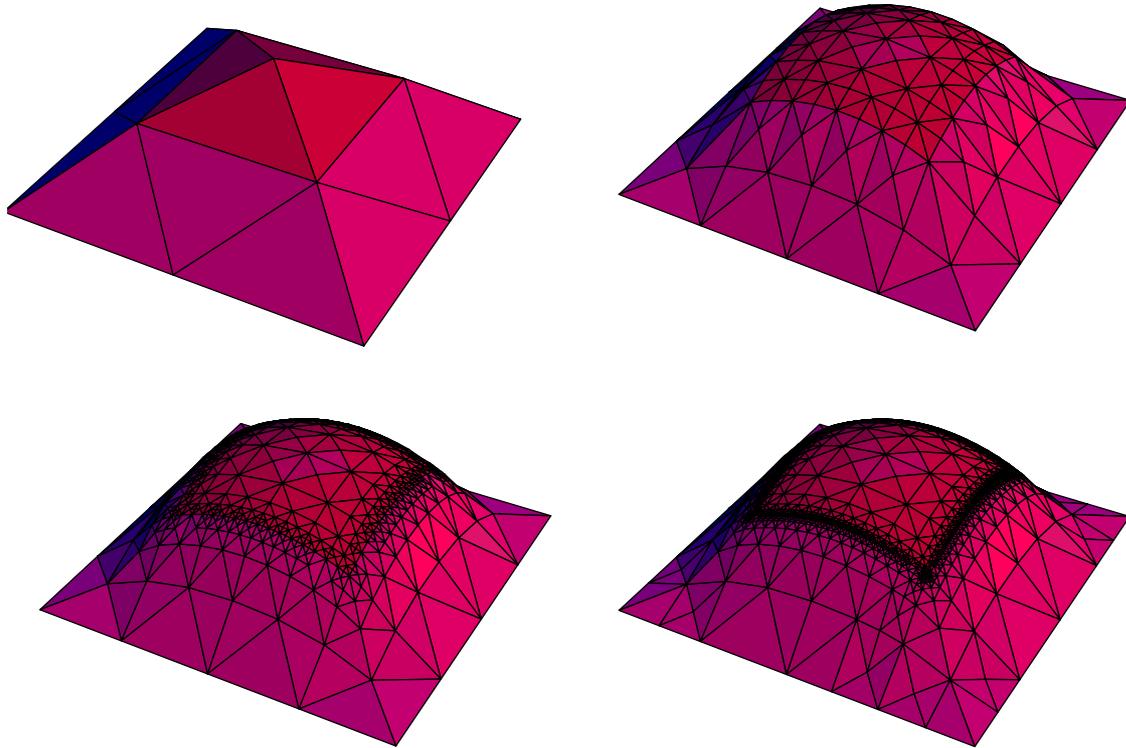
Clearly, contributions from edge-jumps of the bending moments and the shear forces (which physically are not permitted and hence spurious) are biggest in the region where the different material stacks meet. The error terms we are required to include are the element force deviations,

$$\|\nabla \cdot (\nabla \cdot \mathbf{A}_2 : \nabla \nabla w_h) - g_0\|_{L^2(T)}^2, \quad (5.13)$$

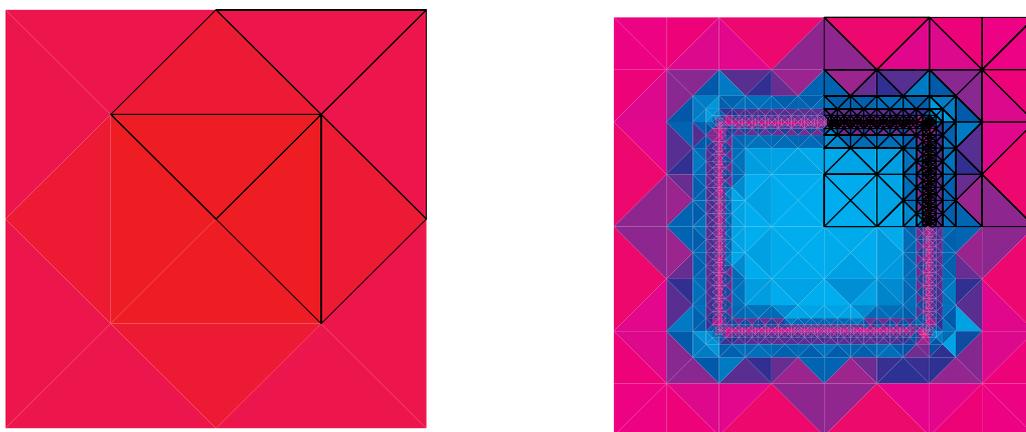
and the bending moment jumps and shear force jumps:

$$\begin{aligned} & \|\mathbf{n} \cdot [\mathbf{A}_2 : \nabla \nabla w_h]\|_{L^2(E)}^2, \\ & \|\mathbf{n} \cdot [\nabla(\mathbf{A}_2 : \nabla \nabla w_h)]\|_{L^2(E)}^2. \end{aligned} \quad (5.14)$$

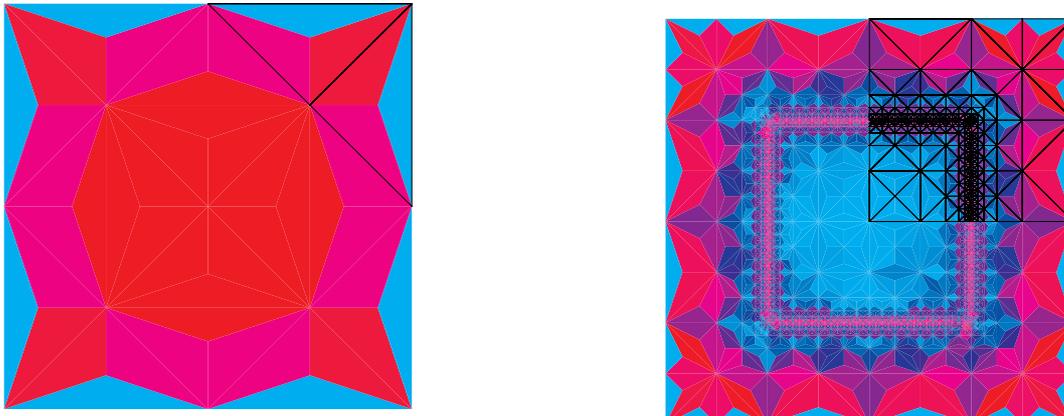
Fig. 5.19, Fig. 5.20 and Fig. 5.21 show these contributions at the start and for the refined mesh after the specified number of adaptivity cycles. The decrease of the computed energy error estimator during the adaptivity cycles is shown in Fig. 5.22.



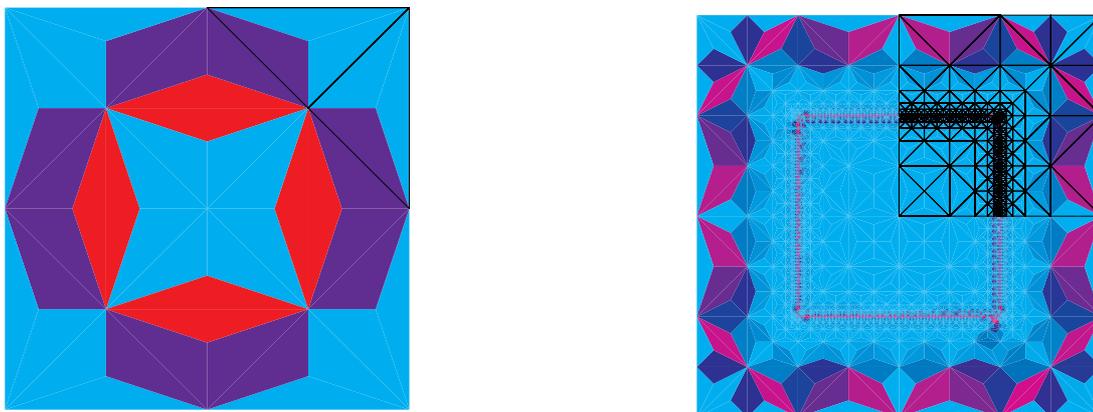
**Figure 5.18** Exaggerated deformation of a single-layered membrane consisting of two different materials under constant vertical surface pressure before (top-left), during, (top-right, step 5; bottom-left, step 9) and after 14 refinement cycles (bottom-right). Major refinement occurs at the material transition interface. We should note that the displacement field is interpolated by  $C^1$ -functions whereas the graphical interpolation is linear.



**Figure 5.19** The fully clamped membrane: Inner element force errors in the starting coarse mesh and after a refinement iteration of depth 14.



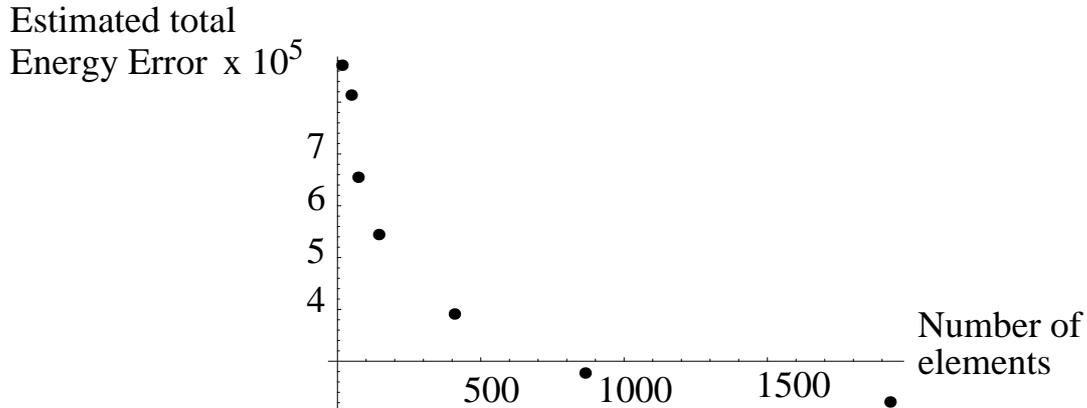
**Figure 5.20** The fully clamped membrane: Errors due to discontinuities of shear forces along element interfaces. After the 14th refinement step jumps in the inner region are almost extinguished (greenish).



**Figure 5.21** Reduction of the bending moment errors. These are largest at the material transitions in the starting coarse mesh and remain so in the refined mesh.

### 5.3.3 Double-layer stretched membrane

The third example serves to demonstrate the influence of the coupling of in-plane displacement and out-of-plane displacement fields on the solution errors. This only occurs for multi-layered structures, due to the fact that only then do first order moments of stresses take on non-zero values. As in the first example, a membrane is subjected to constant surface traction at two opposite boundary edges. A double layer structure is embedded in the membrane (see Fig. 5.23).



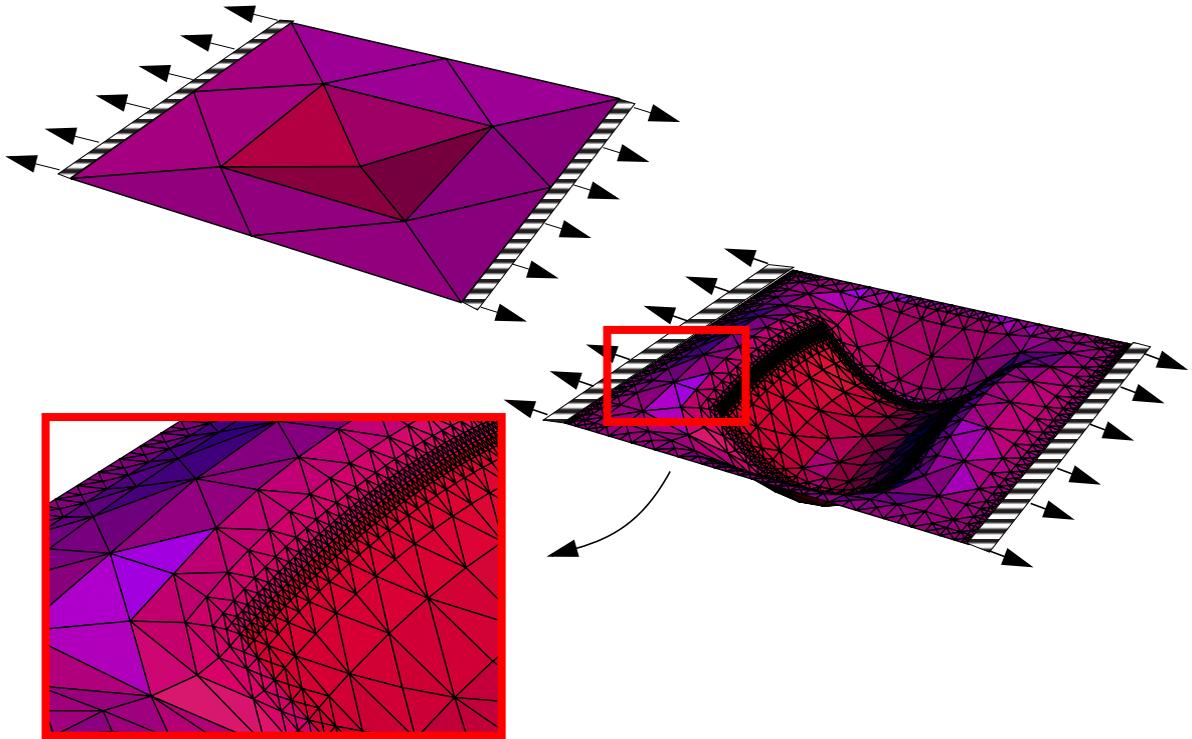
**Figure 5.22** Reduction of the total error of the clamped membrane. The error is displayed versus the number of elements during the refinement process.

Here, the out-of-plane displacement does not vanish and we are left with the following contributions: inner-element deviations of the computed forces from the prescribed forces where the latter are zero for both the transversal and horizontal displacement fields

$$\begin{aligned}
 & \|\nabla \cdot (\nabla \cdot \mathbf{A}_2 : \nabla \nabla w_h)\|_{L^2(T)}^2, \\
 & \|\nabla \cdot (\mathbf{A}_0 : (\nabla \mathbf{u}_h)^S)\|_{L^2(T)}^2, \\
 & \|\nabla \cdot (\nabla \cdot (\mathbf{A}_1 : (\nabla \mathbf{u}_h)^S))\|_{L^2(T)}^2
 \end{aligned} \tag{5.15}$$

and the sum of the jump terms

$$\begin{aligned}
 & \|\mathbf{n} \cdot [\nabla \cdot (\mathbf{A}_2 : \nabla \nabla w_h)]\|_{L^2(E)}^2 h_E^3 + \|\mathbf{n} \cdot [\mathbf{A}_2 : \nabla \nabla w_h]\|_{L^2(E)}^2 h_E \\
 & + \|\mathbf{n} \cdot [\mathbf{A}_0 : (\nabla \mathbf{u}_h)^S]\|_{L^2(E)}^2 h_E^3 + \|\mathbf{n} \cdot [\mathbf{A}_1 : (\nabla \mathbf{u}_h)^S]\|_{L^2(E)}^2 h_E \\
 & + \|\mathbf{n} \cdot [\nabla \cdot (\mathbf{A}_1 : (\nabla \mathbf{u}_h)^S)]\|_{L^2(E)}^2 h_E^3 + \|\mathbf{n} \cdot [\mathbf{A}_1 : \nabla \nabla w_h]\|_{L^2(E)}^2 h_E^3.
 \end{aligned} \tag{5.16}$$

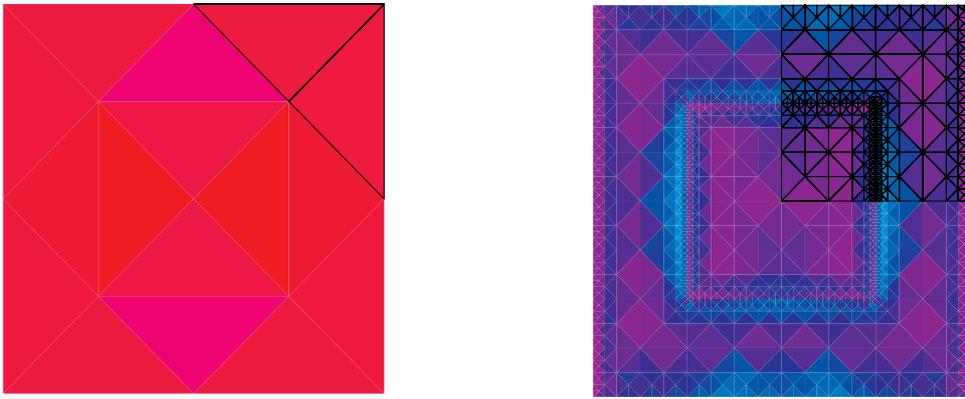


**Figure 5.23** Exaggerated deformation of a membrane with a double layer structure embedded in the center region. While the in-plane displacement is small and is hard to distinguish in the figure, we clearly see that out-of-plane displacement occurs in both the positive and the negative directions.

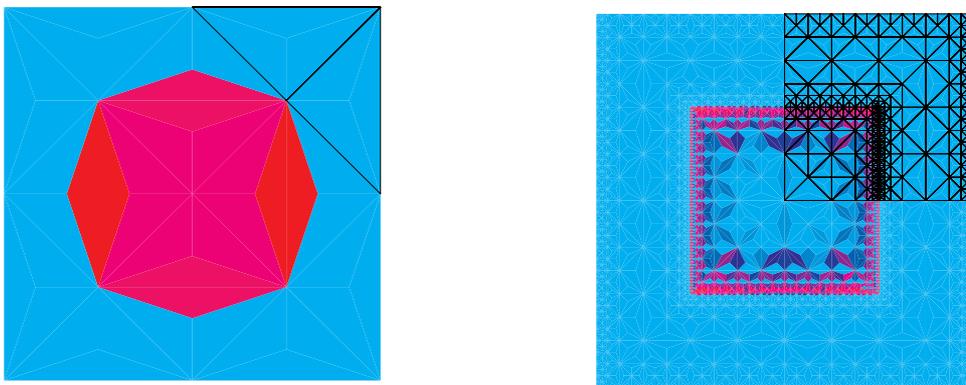
As mentioned before, terms involving first order derivatives of the in-plane displacement field only then do not vanish, when material properties across elements change. Jumps of first order moment stress-couples, i.e.,

$$\|n \cdot [A_1 : (\nabla u_h)^{S_1}]\|_{L^2(E)}^2 \quad (5.17)$$

obviously only exist along the layer-stack interface. Exemplarily we show some terms contributing to the total error of the FE-approximation of the field  $U$  in Fig. 5.24 and Fig. 5.25.



**Figure 5.24** Error of inner-element bending moments. Contributions in the same regions decrease at a similar rate, from reddish to greenish.



**Figure 5.25** Bending moment error contributions due to the coupling of the in-plane and out-of-plane fields. The only contributions occur where first order moments of the elasticity tensor exist. The highest contributions occur in the regions where second order derivatives of the  $z$ -displacement field are highest, also see Fig. 5.23.

### 5.3.4 Comparison of uniform and adaptive Refinement

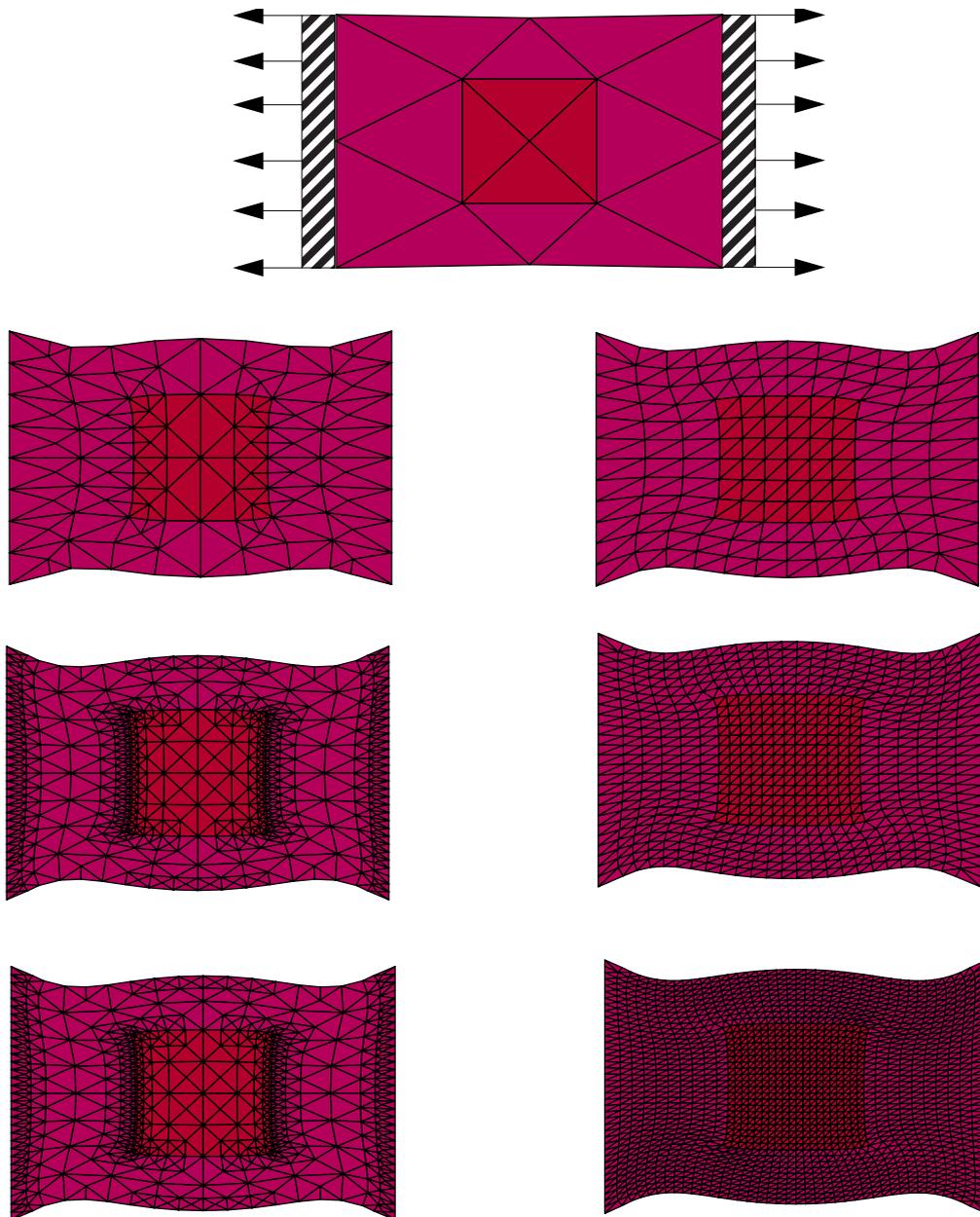
In this section we demonstrate how valuable in terms of computational resources, that is to say, memory requirements and computational time, the error adaptive procedure is. We thus compare the estimated error for a mesh that is uniformly refined, such that all elements present in the mesh are of the same size, and a mesh that is adaptively refined to gain a better accuracy. The BVP under consideration is a membrane-like structure consisting of two different material stacks, the stiffer

located in the center of the device. A surface traction is applied on the left and the right hand side both of the same magnitude but different directions. The upper and the lower side of the membrane are completely free. Figure 5.26 shows the meshes of uniform refinement and the adaptive refinement together with the deformed domain. The left column displays three adaptive meshes during the refinement cycle whereas the right column shows the membrane deformation with underlying uniform meshes. Even though in each row the number of elements are approximately the same for both the uniform and the adaptive refinement, the estimated error values corresponding to each mesh vary considerably, see Figure 5.27. We also see from that figure that in an early stage of the adaptive refinement process the results are much more accurate as if a uniform mesh with the same number of elements had been taken.

### 5.3.5 AFM-Beams

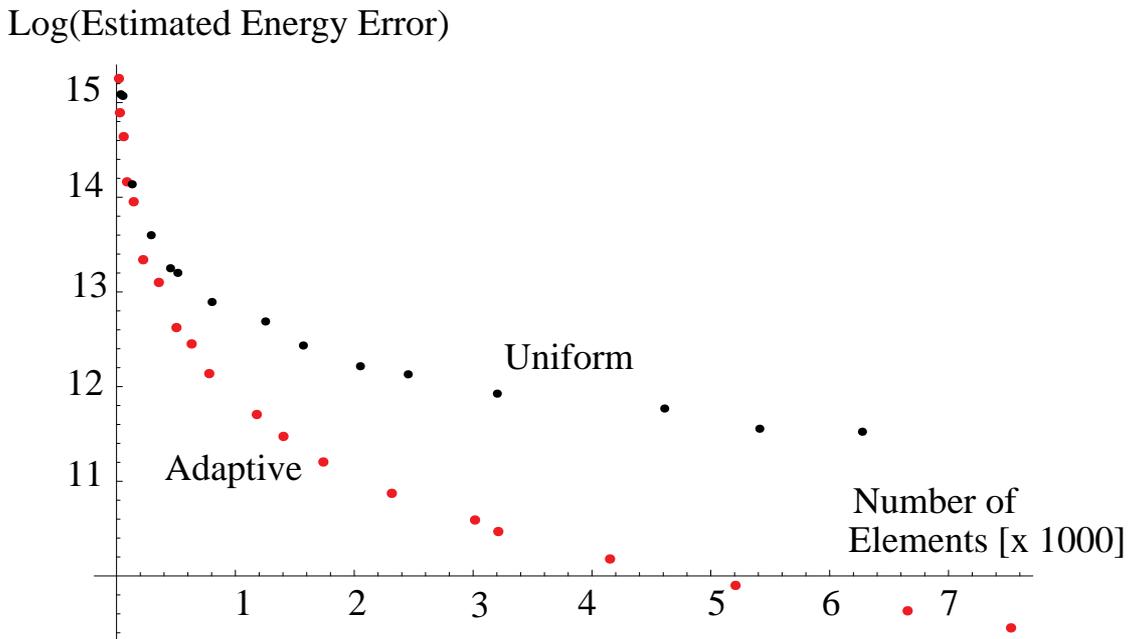
We demonstrate adaptive mesh refinement on two atomic force microscope (AFM) cantilevers [116],[117]. The basic operation of an AFM is dependent on the atomic forces that are exerted on the tip of a probe being sufficient to cause measurable deflection of the cantilever on which the probe is placed. Cantilevers for this purpose arise in a great variety of shapes [85], [91], [92]. Rectangular beams or cantilevers in simplified models, that is, only considering horizontally isotropic material properties, still are accessible by analytical solutions, even in the case of piezoelectrically sensing and actuating thin MEMS devices [33]. Complex geometry and horizontally non-homogenous material properties exceed the capabilities of the analytical approaches by far and even highly sophisticated commercial simulation software tools do not allow for appropriate consideration of the singularities of the computable (weak) solution.

A widely used atomic force microscope cantilever or beam which we consider here is V-shaped and reported of in [91] and [92]. The starting material is a SOI wafer, which is a buried oxide wafer. Fabrication of the AFM cantilevers with integrated Piezo-Resistive sensors consists of KOH etching of the SOI substrate in order to create a membrane, where the AFM cantilevers will be located. After the Piezo-Resistive are implanted on the cantilevers, the back thin silicon film is removed by reactive ion etching. Finally, the AFM cantilevers are released by buffered HF etching silicon dioxide on the cantilever and in the SOI wafer such



**Figure 5.26** The deformation of the doubly stretched membrane computed by varying triangulations. Below the initial meshing the adaptive mesh is shown in the left column for 220, 1176 and 3208 elements. The right column displays the uniform refinement for 288, 1250 and 3200 elements. Although in each row the number of elements is approximately the same, the estimated energy error differs considerably, see Figure 5.27.

that a typical cantilever thickness of  $2 - 3\mu m$  results. The cantilever's typical dimension is about 200 microns length and 80 microns width. Many efforts have been made to accurately model the mechanical behavior of such cantilevers [91], but lack of generality in case more sophisticated geometries will be designed.

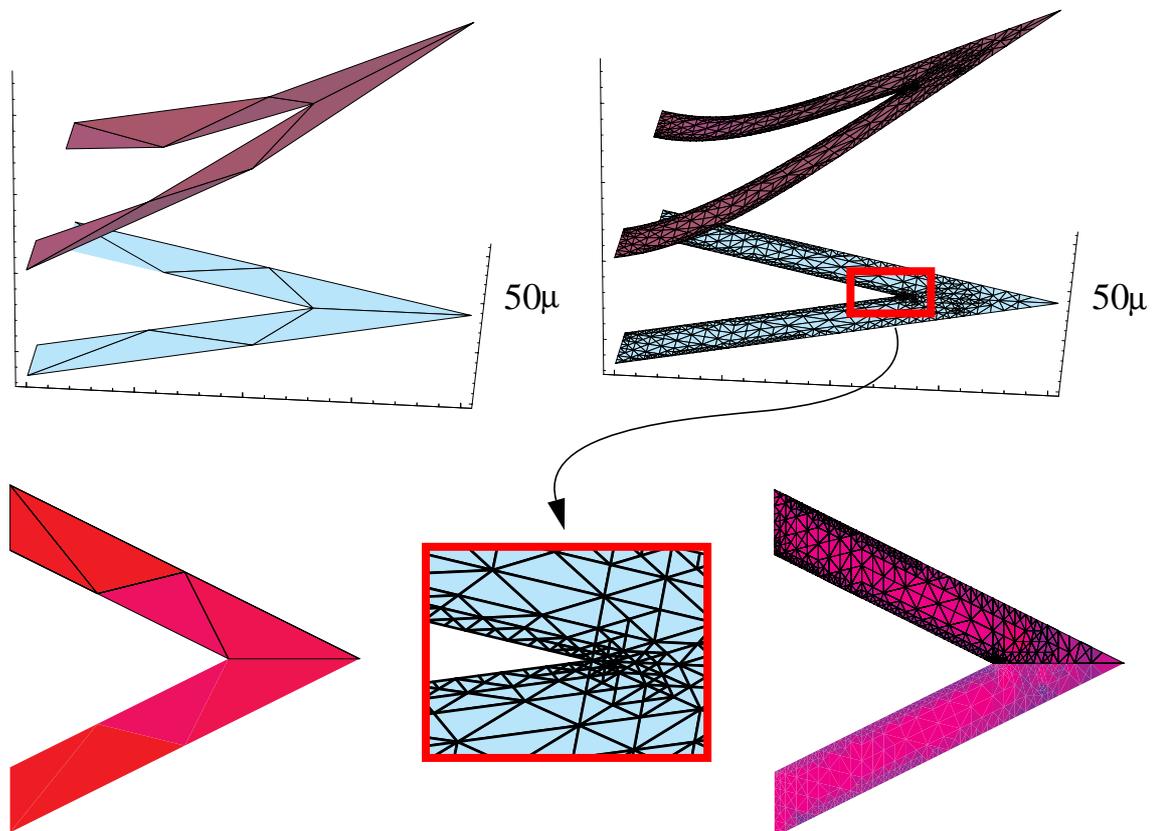


**Figure 5.27** Comparison of the error reduction for uniform and adaptive refinement for the bi-material structure under tension. Already at early stages of the refinement cycle the adaptive procedure shows its advantages.

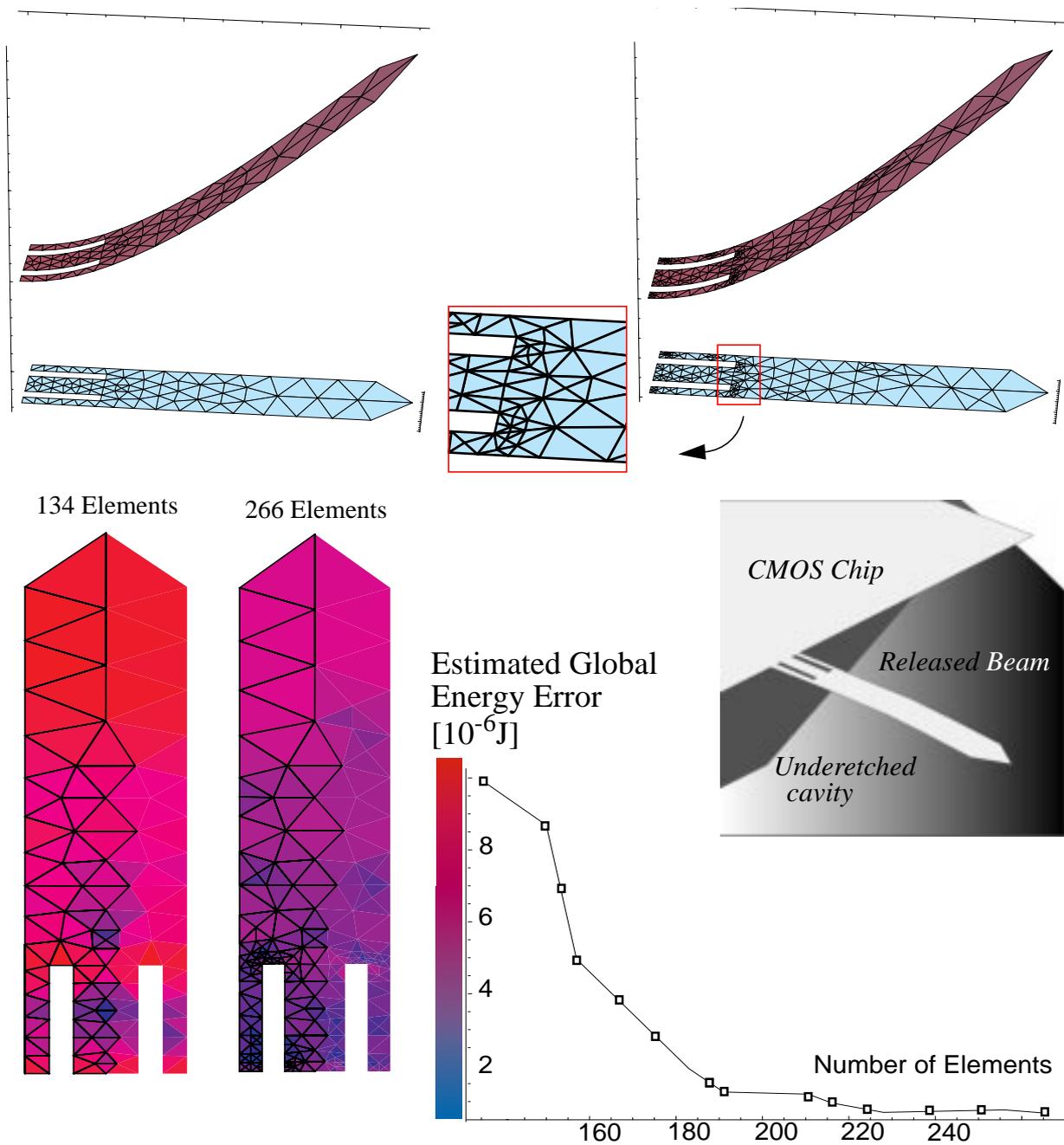
We have simulated the bending of the probe which is subject to a vertical force and clamped at the beam's ends. Fig. 5.28 shows the exaggerated deflection of the beam computed on the initial mesh as well as that computed on a mesh after 26 refinement steps.

Another AFM cantilever which we consider here is presented in [85]. It is fabricated in CMOS-technology and consists of a one micron thick silicon dioxide and silicon nitride bi-layer. The cantilever is typically ca.  $300 - 500\mu m$  long and  $80 - 100\mu m$  wide. We have again simulated the bending of the probe, which is sub-

ject to a vertical force in the tip-region and is clamped at the beam's ends. Fig. 5.29 shows the bending behaviour and the error reduction for the initial coarse mesh (134 elements) and the refined mesh consisting of 266 elements.



**Figure 5.28** Exaggerated deflection of a V-shaped AFM tip computed by the initial mesh and after a refinement of 26 cycles. Major singularities occur at the re-entrant corner and at the free boundaries. Below the error distribution of the overall error is displayed.



**Figure 5.29** Reduction of inner-element force errors (purpled) during the refinement cycle. Re-entrant corners and the clamped part of the beam are identified as regions where highest errors occur (red). The strategy equidistributes the error over the mesh. The total estimated error is plotted versus the number of elements.

### 5.3.6 A thermo-mechanical cantilever

A main operation principle for thin structure MEMS devices is that of a thermally actuated beam [93]. Usually the beam represents a multi-layer, or multimorph structure which is clamped at one end. When heated it bends into the vertical direction by virtue of the different thermal expansion coefficients of each layer. This becomes clear when inspecting equation (3.118), which can be recast, when the non-linear remainder and the piezo-electrical influences are neglected as

$$\begin{aligned}
 & (\mathbf{A}_0 : (\nabla \mathbf{u})^S - \mathbf{A}_1 : \nabla \nabla w + (\hat{\alpha} \Delta T)_0, (\nabla \mathbf{v})^S) \\
 & - (\mathbf{A}_1 : (\nabla \mathbf{u})^S - \mathbf{A}_2 : \nabla \nabla w + (\hat{\alpha} \Delta T)_1, \nabla \nabla v) \\
 & = \langle \mathbf{f}_0, \mathbf{v} \rangle - \langle \mathbf{f}_1, \nabla v \rangle + \langle g_0, v \rangle .
 \end{aligned} \tag{5.18}$$

Allowing only vertical displacement and assuming that no external force acts on the structure, we can rewrite (5.18) as

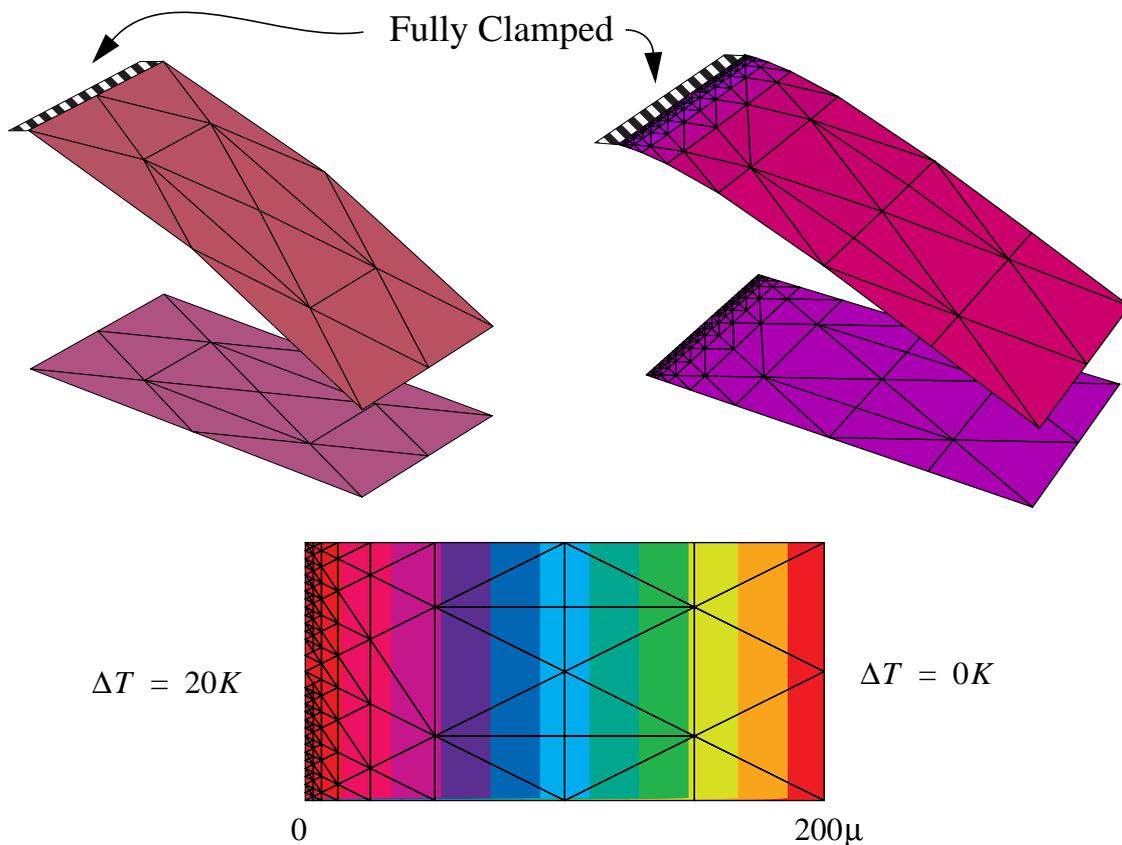
$$(\mathbf{A}_2 : \nabla \nabla w + (\hat{\alpha} \Delta T)_1, \nabla \nabla v) = 0 . \tag{5.19}$$

The temperature difference,  $\Delta T$ , together with the thermal expansion  $\hat{\alpha}$  only then do not vanish if their first order moment across the thickness exist. The smallest structure to fulfil this condition is, as expected, a bimorph. Alternatively, when the temperature difference is not due to an additional VBVP, the term  $(\hat{\alpha} \Delta T)_1$  can be viewed as some external prestress caused during the fabrication process. We demonstrate the case of a widely used bimorph structure that is clamped at one end, meaning the essential boundary condition

$$w = \mathbf{n} \cdot \nabla w = 0 , \tag{5.20}$$

and which adopts a temperature distribution decreasing linearly from the clamped end to the other as displayed in Figure 5.30. As a typical physical dimension we chose  $200\mu$  as length and an overall thickness of two microns for each layer. In this way a thermomechanical actuator can be modeled that is heated in the region of the beam's clamped end. The adaptive procedure here takes into account errors due to jumps of shear forces and bending moments as well as inner element force deviations. The errors caused by jumps of the heat fluxes across element bound-

aries are identically zero since the temperature field decreases linearly from the left to the right beam's ends.



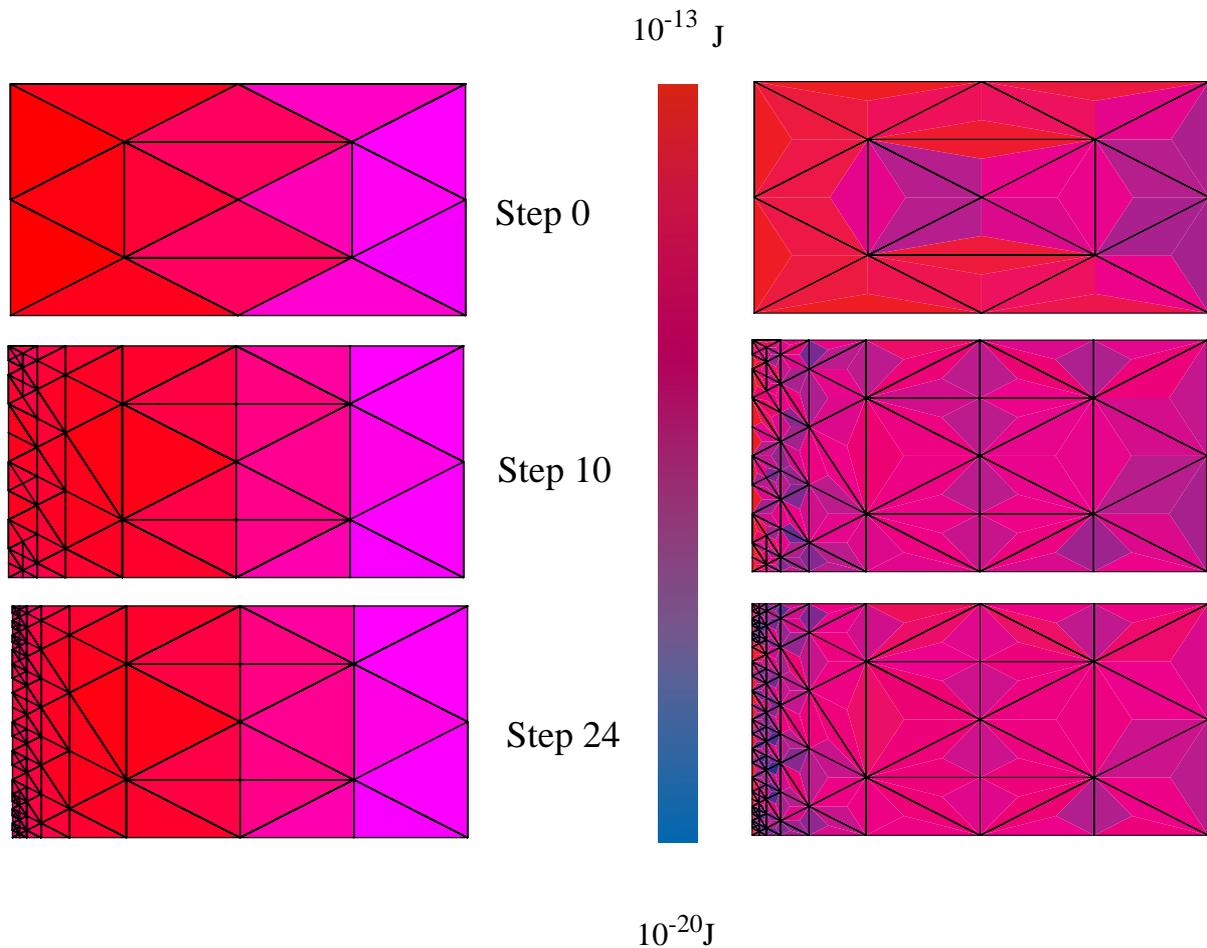
**Figure 5.30** Deflection (exaggerated) of a thermo-mechanically actuated rectangular flap and the linear temperature distribution.

We display the decrease of the shear force errors during the refinement cycle in Figure 5.31, where as well the elements' stored energy  $E_T$  in terms of

$$E_T = \frac{1}{2} (\mathbf{A}_2 : \nabla \nabla w_h, \nabla \nabla w_h)_{L^2(T)} \quad (5.21)$$

is shown, which during the refinement cycle approximately remains constant independently of the elements' sizes. As can be clearly seen, the major refinement

occurs at the clamped end of the structure whereas at the opposite end up to step 24 no new nodes are introduced.

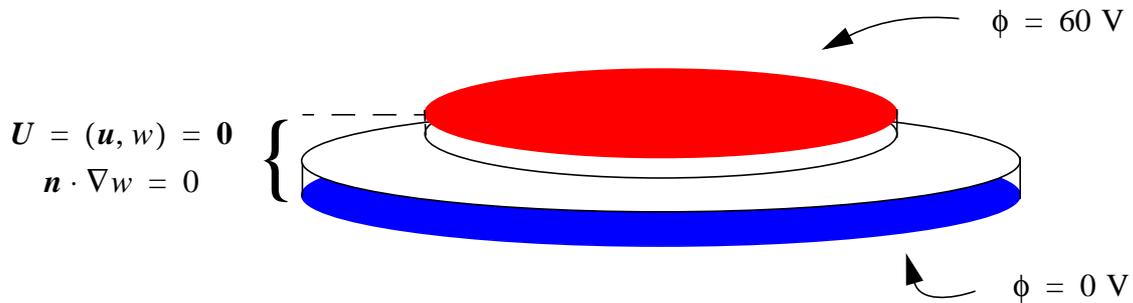


**Figure 5.31** The mechanical energy (left) does not vary remarkably during the refinement process whereas the reduction of the shear force errors can be observed in the right column.

### 5.3.7 A piezo-electro-mechanical membrane

A wide range of micro-electro-mechanical systems use piezo-electrically active components as their sensing or actuating elements [89]. We simulate the coupled electrical and mechanical behaviour of a circular membrane which is clamped and which has a piezo-electrically active layer on top of it. Among others, the working

principle might be that of a pressure sensor or that of a pump when used as an actuator. This could be realized by applying different electrical potentials on the top and on the bottom of the layer as sketched in Figure 5.32.



**Figure 5.32** Boundary conditions for a circular membrane actuated by a piezo-layer on top of it.

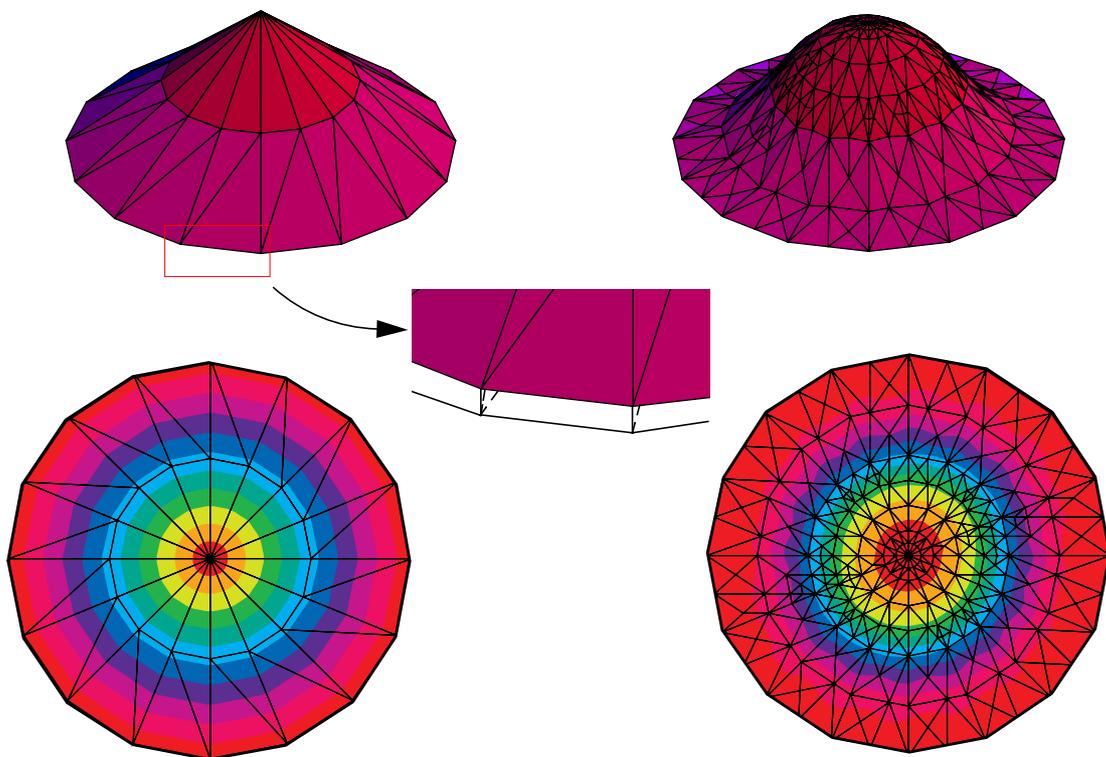
In many applications the piezo-electrically active layers are of the  $6mm$  crystal class such as, for example, ZnO or PZT-4. The unreduced, three-dimensional piezoelectric tensor  $\mathbf{d}$ , given in units of  $C/N$ , then has the form

$$\mathbf{d} = \begin{bmatrix} 0 & 0 & 0 & 0 & d_{15} & 0 \\ 0 & 0 & 0 & d_{15} & 0 & 0 \\ d_{31} & d_{31} & d_{33} & 0 & 0 & 0 \end{bmatrix} \quad (5.22)$$

where we have used the engineering notation (2.34) for the last two indices of the third rank piezo-tensor. When also taking into account the numbers of non-zero and independent elastic coefficients for the crystal class under consideration as in Figure 2.1., the tensor reduction (see A.2) yields a piezoelectric tensor that only has two independent components, which finally turn out to be identical since  $A_{11} = A_{22}$ ,

$$\hat{\pi} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ d_{31}(A_{11} + A_{21}) & d_{31}(A_{12} + A_{22}) & 0 \end{bmatrix}, \quad (5.23)$$

$A_{ij}$  denoting the components of the reduced elastic tensor. The membrane's deformation behaviour is shown in Figure 5.33.



**Figure 5.33** Exaggerated deformation and vertical displacement of a membrane which has a piezo-disc on top of it. The principle of actuation is due to a potential difference between the top and the bottom surface. The left column shows the results for the starting coarse mesh while the right column represents the results after a refinement of 12 steps. Prisms, where the electrical field is interpolated on, are shown schematically.

Errors according to (3.220) are biggest where the two different layers meet and in the membrane's center. Exemplarily we have displayed two types of the errors which occur in (3.220). The element force errors shown in the left column of Figure 5.34,

$$\|\nabla \cdot (\nabla \cdot \boldsymbol{\sigma}_{w2})\|_{L^2(T)}^2 h_T^4 \quad (5.24)$$

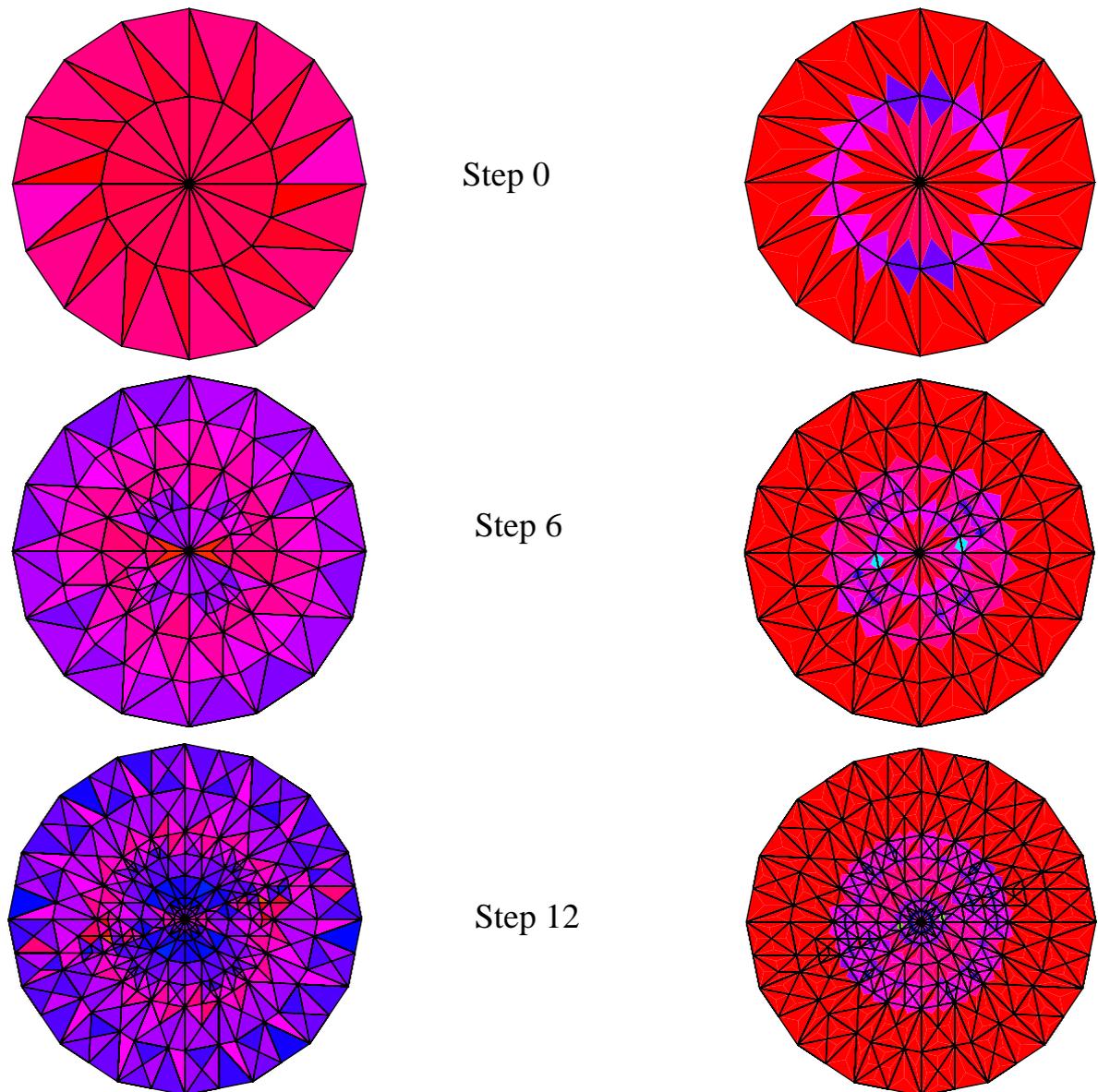
together with other errors involving the out-of-plane displacement represent the principal contribution to the total estimated error, varying with the structure parameters such as the magnitude of the BCs, material properties, or the layer-thicknesses. The contributions due to the stress caused by the inverse piezo-electric effect,

$$\|[\mathbf{n}_{2D} \cdot \hat{\boldsymbol{\pi}}^T \cdot \nabla \phi]\|_{L^2(E_{3D})}^2 h_E^4 = \|[\mathbf{n}_{2D} \cdot \boldsymbol{\sigma}_{P\phi}]\|_{L^2(E_{3D})}^2 h_E^4, \quad (5.25)$$

generally are smaller. We should, however, mention, that these only do not vanish where the structure has more than one piezoelectrically active layer, namely in the center region. Although the potential's gradient does not vanish identically outside the center circle, no jumps of the quantities of (5.25) have to be encountered there since the bottom-layer's piezo-tensor is assumed to vanish identically. The contributions of the direct piezoelectric effect are identically zero in the case we deal with the hexagonal crystal symmetry class  $6mm$ . The error contributions due to the direct piezoelectric effect are of the general shape, see (3.220),

$$\|\boldsymbol{\Xi}_{3D} \cdot \hat{\boldsymbol{\pi}} \cdot \mathbf{v}\| = \|\boldsymbol{\Xi}_{3D} \cdot \hat{\boldsymbol{\pi}}_{i\alpha\beta} v_{\alpha\beta}\| \quad (5.26)$$

where  $\boldsymbol{\Xi}_{3D}$  either denotes the three-dimensional divergence operator or the three dimensional face normal vector. Derivatives of the in-plane or the out-of-plane displacement fields are abbreviated by  $v_{\alpha\beta}$  with the common two-index notation [20], greek indices ranging from one to two, latin indices ranging from one to three. Then, since the only non-vanishing components of  $\hat{\boldsymbol{\pi}}$  are in its last row, (5.26) is zero for two possible reasons: either the prism's horizontal face normal's z-component vanishes or the electrical displacement  $\hat{\boldsymbol{\pi}}_{i\alpha\beta} v_{\alpha\beta}$  does not depend on



**Figure 5.34** Error distribution for the clamped piezo-electrically actuated membrane. The left column shows the reduction of the inner-element force errors, a purely mechanical contribution, whereas the right column shows how the error due to the stress jumps caused by the inverse piezoelectric effect are reduced.

the  $z$ -coordinate and thus its partial derivative represented by  $\Xi_{3D}$  vanishes identically.





# 6 CONCLUSION AND OUTLOOK

Finite element (FE) models and tools enable the MEMS designer to analyze the thin structures' physical behaviour even for complicated device geometries. However, singularities arising from different natures deteriorate the quality of the approximated solution fields and thus a method to control the accuracy is highly desirable.

We have presented a method implemented in the tool ADAPTREF with which we can accurately simulate the thermo-electro-mechanical behaviour of prestressed multi-layer structures by minimizing the error at a minimum of computational cost. Based on a sound functional analytical framework, a thin plate model is used which covers thermal, electrical, mechanical and their coupling effects in thin multi-layered structures.

The aim of controlling the accuracy was to find, for a given choice of a FE-Model, i.e a geometry and a load case, together with several kinds of boundary conditions, an adapted mesh whose error is reduced to a minimum. We therefore applied residual error estimation techniques to the weak form of the coupled plate equations in order to get the several error contributions mainly consisting of inner-element force residuals and jumps of stress couples, bending moments and shear forces across element edges when only mechanical analysis is desired. In case thermo- or electro-mechanical actuated thin structures are under consideration other sources of errors have to be taken into account such as jumps of heat fluxes or jumps of the electrical displacement. We have shown how these different contributions influence the way in which the mesh is refined. The decrease of the estimated energy error proves the reliability of the theoretically derived error estimation. A maximum strategy is used for all meshes treated, by which we aim to reduce the computational error by equidistributing it over the whole structure. A recursive refinement algorithm turned out to be most efficient to split both two-

dimensional (2D) and three-dimensional (3D) simplicial elements from a geometrical viewpoint in order to preserve shape regularity and to avoid hanging nodes. The software tool ADAPTREF allows of treating mixed 2D and 3D finite elements at one time which are required to adequately simulate piezo-electrically active MEMS. The performance of the method developed and implemented is demonstrated for several thin multi-layer structures under different thermo-electro-mechanical load cases such as, for example, micro-machined AFM-beams or piezo-electrically driven micro-membranes.

The main driving force behind choosing the object-oriented approach for the architecture of the tool ADAPREF was to guarantee that it could be flexibly extended to a wider range of applications. Since its basis is formed by purely geometrical objects which can operate independently, virtually any numerical method based on domain discretization is suited to be used in conjunction with ADAPTREF in order to improve its performance. First, the variety of the polyhedral or polygonal mesh constituents could be easily increased. This in turn evokes that new splitting techniques are implemented. Another challenge is the extension to the handling of time dependent and non-linear problems where also mesh coarsening techniques are required and the hierarchical mesh data structure can be fully re-used.

Another extension would be to provide interfaces to the most common commercially used finite element tools which in general do not dispose of methods to control the accuracy. As soon as the command structure, the physical models, the element types and the solution output data structure of the program are known ADAPTREF could be accordingly extended.

The task of adding physical models probably is the most frequent request by microsystem designers. Therefore, a scheme has to be provided how to add partial differential equations to the system. Since all the physical fields are represented by functions with a finite dimension, all the desired information can be made available just by implementing the type and the order of the discrete functions, e.g. the polynomial order, and the kind of differential operators the field obeys. Hereby a great amount of re-use of existing code is assured since many differential operators acting on many kinds of fields are already available.





---

# APPENDIX

## A.1 Tensor operations

In this thesis, we confine ourselves to the common notation for quantities that in the engineering language are denoted as tensors. The operations come in a handy size when using the notation of [96]:

<i>Operation</i>	<i>Definition</i>
Contraction	$(\mathbf{A} \cdot \mathbf{B})_{i\dots j l\dots m} = \sum_k A_{i\dots j k} B_{k l\dots m}$
Double contraction	$(\mathbf{A} : \mathbf{B})_{i\dots j m\dots n} = \sum_{kl} A_{i\dots j k l} B_{k l m\dots n}$
Tensor Product	$(\mathbf{A} \mathbf{B})_{i\dots j k\dots l} = \sum_{kl} A_{i\dots j} B_{k\dots l}$
Transposition	$(\mathbf{A}^T)_{ij} = A_{ji}$
Symmetrization	$(\mathbf{A}^S)_{ij} = (A_{ij} + A_{ji})/2$
Gradient	$(\nabla \mathbf{A})_{ij\dots k} = \partial_i A_{j\dots k} = A_{j\dots k, i}$
Divergence	$(\nabla \cdot \mathbf{A})_{j\dots k} = \sum_i \partial_i A_{ij\dots k} = \sum_i A_{ij\dots k, i}$
$L^2(\Omega)$ inner product	$\int_{\Omega} \sum_{i\dots j} A_{i\dots j} B_{i\dots j} dx$

---

## A.2 Tensor Reduction for Plates

For reasons of symmetry, the stress tensor has only six independent components

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_1 & \sigma_6 & \sigma_5 \\ \sigma_6 & \sigma_2 & \sigma_4 \\ \sigma_5 & \sigma_4 & \sigma_3 \end{bmatrix} \quad (\text{A.1})$$

that can be arranged into two three-dimensional vectors

$$\hat{\boldsymbol{\sigma}}_1 = \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_6 \end{bmatrix} \quad \hat{\boldsymbol{\sigma}}_2 = \begin{bmatrix} \sigma_3 \\ \sigma_4 \\ \sigma_5 \end{bmatrix}. \quad (\text{A.2})$$

Also for the strain, its six independent components

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 & \varepsilon_6/2 & \varepsilon_5/2 \\ \varepsilon_6/2 & \varepsilon_2 & \varepsilon_4/2 \\ \varepsilon_5/2 & \varepsilon_4/2 & \varepsilon_3 \end{bmatrix} \quad (\text{A.3})$$

can be arranged into two three-dimensional vectors

$$\hat{\boldsymbol{\varepsilon}}_1 = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_6 \end{bmatrix} \quad \hat{\boldsymbol{\varepsilon}}_2 = \begin{bmatrix} \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \end{bmatrix}. \quad (\text{A.4})$$

Consequently, the 21 independent components of the elastic tensor can be arranged into four three-dimensional matrices  $\mathbf{C}_{ij}$

---


$$\begin{aligned}
\mathbf{C}_{11} &= \begin{bmatrix} C_{11} & C_{12} & C_{16} \\ C_{21} & C_{22} & C_{26} \\ C_{61} & C_{62} & C_{66} \end{bmatrix} & \mathbf{C}_{12} &= \begin{bmatrix} C_{13} & C_{14} & C_{15} \\ C_{23} & C_{24} & C_{25} \\ C_{63} & C_{64} & C_{65} \end{bmatrix} \\
\mathbf{C}_{21} &= \begin{bmatrix} C_{31} & C_{32} & C_{36} \\ C_{41} & C_{42} & C_{46} \\ C_{51} & C_{52} & C_{56} \end{bmatrix} & \mathbf{C}_{22} &= \begin{bmatrix} C_{33} & C_{34} & C_{35} \\ C_{43} & C_{44} & C_{45} \\ C_{53} & C_{54} & C_{55} \end{bmatrix}
\end{aligned} \tag{A.5}$$

relating the vectors  $\hat{\sigma}_i$  with the vectors  $\hat{\epsilon}_j$ :

$$\begin{bmatrix} \sigma(1, 2) \\ \sigma(\cdot, 3) \end{bmatrix} = \begin{bmatrix} \mathbf{C}_{11} & \mathbf{C}_{12} \\ \mathbf{C}_{21} & \mathbf{C}_{22} \end{bmatrix} \begin{bmatrix} \hat{\epsilon}_1 \\ \hat{\epsilon}_2 \end{bmatrix} \tag{A.6}$$

In case  $\sigma$  is composed of mechanical stress and thermal stress (thermal expansion tensor  $\mathbf{a}$  given in units of 1/K) we have

$$\begin{bmatrix} \sigma(1, 2) \\ \sigma(\cdot, 3) \end{bmatrix} = \begin{bmatrix} \mathbf{C}_{11} & \mathbf{C}_{12} \\ \mathbf{C}_{21} & \mathbf{C}_{22} \end{bmatrix} \left( \begin{bmatrix} \hat{\epsilon}_1 \\ \hat{\epsilon}_2 \end{bmatrix} - \begin{bmatrix} \hat{\mathbf{a}}_1 \\ \hat{\mathbf{a}}_2 \end{bmatrix} \Delta T \right), \tag{A.7}$$

or

$$\begin{bmatrix} \sigma(1, 2) \\ \sigma(\cdot, 3) \end{bmatrix} = \begin{bmatrix} \mathbf{C}_{11}\hat{\epsilon}_1 + \mathbf{C}_{12}\hat{\epsilon}_2 \\ \mathbf{C}_{21}\hat{\epsilon}_1 + \mathbf{C}_{22}\hat{\epsilon}_2 \end{bmatrix} - \begin{bmatrix} \mathbf{C}_{11}\mathbf{a}_1 + \mathbf{C}_{12}\mathbf{a}_2 \\ \mathbf{C}_{21}\mathbf{a}_1 + \mathbf{C}_{22}\mathbf{a}_2 \end{bmatrix} \Delta T. \tag{A.8}$$

The Kirchhoff-Love theory requires that

$$\sigma(\cdot, 3) = (\sigma_{33}, \sigma_{23}, \sigma_{13}) = (\sigma_3, \sigma_4, \sigma_5) = \mathbf{0} \tag{A.9}$$

such that (A.8) yields

$$\mathbf{C}_{21}(\hat{\epsilon}_1 - \mathbf{a}_1\Delta T) - \mathbf{C}_{22}\mathbf{a}_2\Delta T = -\mathbf{C}_{22}\hat{\epsilon}_2 \tag{A.10}$$

or,

---


$$\hat{\boldsymbol{\varepsilon}}_2 - \mathbf{a}_2 \Delta T = -\mathbf{C}_{22}^{-1} \mathbf{C}_{21} (\hat{\boldsymbol{\varepsilon}}_1 - \mathbf{a}_1 \Delta T). \quad (\text{A.11})$$

Inserting this relation into the plane stress  $\boldsymbol{\sigma}(1, 2)$  yields

$$\begin{aligned} \boldsymbol{\sigma}_1 &= \mathbf{C}_{11} (\hat{\boldsymbol{\varepsilon}}_1 - \mathbf{a}_1 \Delta T) + \mathbf{C}_{12} (\hat{\boldsymbol{\varepsilon}}_2 - \mathbf{a}_2 \Delta T) \\ &= \mathbf{C}_{11} (\hat{\boldsymbol{\varepsilon}}_1 - \mathbf{a}_1 \Delta T) - \mathbf{C}_{12} \mathbf{C}_{22}^{-1} \mathbf{C}_{21} (\hat{\boldsymbol{\varepsilon}}_1 - \mathbf{a}_1 \Delta T) \\ &= (\mathbf{C}_{11} - \mathbf{C}_{12} \mathbf{C}_{22}^{-1} \mathbf{C}_{21}) (\hat{\boldsymbol{\varepsilon}}_1 - \mathbf{a}_1 \Delta T) \\ &=: \mathbf{A} (\hat{\boldsymbol{\varepsilon}}_1 - \mathbf{a}_1 \Delta T) = : \mathbf{A} : \hat{\boldsymbol{\varepsilon}} - \hat{\boldsymbol{\alpha}} \Delta T \end{aligned} \quad (\text{A.12})$$

where  $\hat{\boldsymbol{\alpha}}$  has been defined as  $\hat{\boldsymbol{\alpha}} = \mathbf{A} \mathbf{a}_1$  and the reduced elasticity tensor  $\mathbf{A}$  as (the 3x3 matrix)

$$\mathbf{A} = \mathbf{C}_{11} - \mathbf{C}_{12} \mathbf{C}_{22}^{-1} \mathbf{C}_{21}. \quad (\text{A.13})$$

For an isotropic material, the elasticity tensor can be expressed in terms of the Lamé constants such that

$$\mathbf{A} = \begin{bmatrix} A_{1111} & A_{1122} & 0 \\ A_{2211} & A_{2222} & 0 \\ 0 & 0 & A_{1212} \end{bmatrix} = \begin{bmatrix} \frac{4\lambda\mu + 4\mu^2}{\lambda + 2\mu} & \frac{2\lambda\mu}{\lambda + 2\mu} & 0 \\ \frac{2\lambda\mu}{\lambda + 2\mu} & \frac{4\lambda\mu + 4\mu^2}{\lambda + 2\mu} & 0 \\ 0 & 0 & \mu \end{bmatrix}. \quad (\text{A.14})$$

When introducing the piezoelectric charge constants  $\mathbf{d} = [\hat{\mathbf{d}}_1 \hat{\mathbf{d}}_2]$  (unit  $C/N = m/V$ ) as

$$\hat{\mathbf{d}}_1 = \begin{bmatrix} d_{11} & d_{12} & d_{16} \\ d_{21} & d_{22} & d_{26} \\ d_{31} & d_{32} & d_{36} \end{bmatrix} \quad \hat{\mathbf{d}}_2 = \begin{bmatrix} d_{13} & d_{14} & d_{15} \\ d_{23} & d_{24} & d_{25} \\ d_{33} & d_{34} & d_{35} \end{bmatrix} \quad (\text{A.15})$$

the situation is similar since the stress then is given in terms of

---


$$\begin{bmatrix} \boldsymbol{\sigma}_1 \\ \boldsymbol{\sigma}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{C}_{11} & \mathbf{C}_{12} \\ \mathbf{C}_{21} & \mathbf{C}_{22} \end{bmatrix} \left( \begin{bmatrix} \hat{\boldsymbol{\varepsilon}}_1 \\ \hat{\boldsymbol{\varepsilon}}_2 \end{bmatrix} - \begin{bmatrix} \hat{\boldsymbol{d}}_1^T \\ \hat{\boldsymbol{d}}_2^T \end{bmatrix} \mathbf{E} \right). \quad (\text{A.16})$$

The relation for the plane stress analog to (A.12) then reads

$$\boldsymbol{\sigma}_1 = \mathbf{A}(\hat{\boldsymbol{\varepsilon}}_1 - \hat{\boldsymbol{d}}_1^T \mathbf{E}) \equiv \mathbf{A}:\hat{\boldsymbol{\varepsilon}} - \hat{\boldsymbol{\pi}}^T \cdot \mathbf{E} \quad (\text{A.17})$$

having defined the reduced piezo tensor as

$$\hat{\boldsymbol{\pi}}^T = \mathbf{A}\hat{\boldsymbol{d}}_1^T. \quad (\text{A.18})$$

The equation for the electrical displacement uses reduced quantities, too:

$$\mathbf{D} = \begin{bmatrix} \hat{\boldsymbol{d}}_1 & \hat{\boldsymbol{d}}_2 \end{bmatrix} \begin{bmatrix} \mathbf{C}_{11} & \mathbf{C}_{12} \\ \mathbf{C}_{21} & \mathbf{C}_{22} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\varepsilon}}_1 \\ \hat{\boldsymbol{\varepsilon}}_2 \end{bmatrix} + \boldsymbol{\chi} \mathbf{E} \quad (\text{A.19})$$

where the piezoelectric charge constants are related to the piezoelectric stress coefficients  $\boldsymbol{\pi}$  by

$$\boldsymbol{\pi} = \boldsymbol{d}: \mathbf{C}. \quad (\text{A.20})$$

Introducing the relation for the strain  $\hat{\boldsymbol{\varepsilon}}_2$  obtained from (A.16) by the requirement that  $\boldsymbol{\sigma}_2$  vanishes turns (A.19) into

$$\mathbf{D} = \begin{bmatrix} \hat{\boldsymbol{d}}_1 & \hat{\boldsymbol{d}}_2 \end{bmatrix} \begin{bmatrix} \mathbf{C}_{11} & \mathbf{C}_{12} \\ \mathbf{C}_{21} & \mathbf{C}_{22} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\varepsilon}}_1 \\ (\mathbf{C}_{22}^{-1} \mathbf{C}_{21} \hat{\boldsymbol{d}}_1^T + \hat{\boldsymbol{d}}_2^T) \mathbf{E} - \mathbf{C}_{22}^{-1} \mathbf{C}_{21} \hat{\boldsymbol{\varepsilon}}_1 \end{bmatrix} + \boldsymbol{\chi} \mathbf{E} \quad (\text{A.21})$$

such that the dependence is only on  $\hat{\boldsymbol{\varepsilon}}_1$ . Further calculation yields

$$\mathbf{D} = \hat{\boldsymbol{d}}_1 \mathbf{A} \hat{\boldsymbol{\varepsilon}}_1 - \hat{\boldsymbol{d}}_1 \mathbf{C}_{12} \hat{\boldsymbol{d}}_2^T \mathbf{E} - \hat{\boldsymbol{d}}_1 (\mathbf{C}_{11} - \mathbf{A}) \hat{\boldsymbol{d}}_1^T \mathbf{E} + (\hat{\boldsymbol{d}}_2 \mathbf{C}_{22} \hat{\boldsymbol{d}}_2^T + \hat{\boldsymbol{d}}_2 \mathbf{C}_{21} \hat{\boldsymbol{d}}_1^T) + \boldsymbol{\chi} \mathbf{E}, \quad (\text{A.22})$$

which, when defining

---

$$\hat{\boldsymbol{\chi}} \equiv (\hat{\boldsymbol{d}}_2 \mathbf{C}_{22} \hat{\boldsymbol{d}}_2^T + \hat{\boldsymbol{d}}_2 \mathbf{C}_{21} \hat{\boldsymbol{d}}_1^T) - (\hat{\boldsymbol{d}}_1 \mathbf{C}_{12} \hat{\boldsymbol{d}}_2^T + \hat{\boldsymbol{d}}_1 (\mathbf{C}_{11} - \mathbf{A}) \hat{\boldsymbol{d}}_1^T) + \boldsymbol{\chi} \quad (\text{A.23})$$

can be recast into

$$\mathbf{D} = \hat{\boldsymbol{\pi}} \cdot \hat{\boldsymbol{\varepsilon}} + \hat{\boldsymbol{\chi}} \cdot \mathbf{E}. \quad (\text{A.24})$$

---

# REFERENCES

- [1] J. G. Korvink, *Microsystem Simulation*, Lecture Notes, Freiburg, 2000.
- [2] K. Eriksson, D. Estep, P. Hansbo, C. Johnson, *Computational differential equations*, Cambridge Univ. Press, 1996.
- [3] J.E. Marsden, T.J.R. Hughes, *Mathematical Foundations of Elasticity*, Prentice-Hall, 1983.
- [4] R. Abraham, J. E. Marsden, *Foundations of Mechanics*, Addison-Wesley, 1978.
- [5] S. S. Antman, J. E. Osborne, The principle of virtual work and integral laws of motion, *Arch. Rat. Mech. An.* **69**, (1979), pp. 231-262.
- [6] P. G. Ciarlet, *Topics in Mathematical Elasticity*, North Holland, 1983.
- [7] A. C. Eringen, *Continuum Physics, Vol II - Continuum Mechanics of Single-Substance Bodies*, Academic Press, 1975.
- [8] W. T. Koiter, ed., *Theoretical and Applied Mechanics*, North Holland, 1977.
- [9] A. Korn, “Über die Lösung des Grundproblems der Elastizitätstheorie”, *Math. Ann.* **75**, (1914), pp. 497-549.
- [10] G. A. Maugin, *Continuum mechanics of electromagnetic solids*, North-Holland, 1988.
- [11] C. Truesdell, *The Elements of Continuum Mechanics*, Springer, 1966.
- [12] S. P. Timoshenko, *Theory of elasticity*, McGraw-Hill, 1997.
- [13] P. G. Ciarlet, *Mathematical Elasticity, Vol I: Three-dimensional elasticity*, North Holland 1988.
- [14] G. Kirchhoff, “Über das Gleichgewicht und die Bewegung einer elastischen Scheibe”, *J. Reine Angew. Math.* **40**, (1850), pp. 51-58.
- [15] E. Reissner, “On the theory of bending of elastic plates”, *J. Math. Phys.* **23**, (1944), pp. 184-191.
- [16] S. P. Timoshenko, S. Woinowsky-Krieger, *Theory of Plates and Shells*, McGraw-Hill, 1996.
- [17] A. L. Goldenveizer, “Derivation of an approximate theory of shells by means of asymptotic integration of the equations of the theory of elasticity”, *Prikl. Math. Mech.* **26**, (1963), pp. 593-608.

- 
- [18] S. S. Antman, "Ordinary differential equations of one-dimensional non-linear elasticity. I. Foundations of the theories of nonlinearly elastic rods and shells", *Arch. Rat. Mech. Anal.* **61**, (1976), pp. 307-351.
- [19] D. Attou, G. A. Maugin, "Une théorie asymptotique des plaques minces piézoélectriques", *C. R. Acad. Sci. Paris, Sér. II* **304**, (1987), pp. 865-868.
- [20] G. A. Maugin, D. Attou, "An asymptotic theory of thin piezoelectric plates", *Quart. Appl. Math.* **43**, (1990), pp. 347-362.
- [21] N. N. Rogacheva, *The theory of piezoelectric shells and plates*, CRC, 1994.
- [22] I. Babuška, L. Li, "Hierarchic modelling of plates", *Computers and Structures* **40**, (1991), pp. 419-430.
- [23] I. Babuška, L. Li, "The problem of plate modelling - Theoretical and computational results", *Comp. Meth. Appl. Mech. Engrg* **100**, (1992), pp. 249-273.
- [24] I. Babuška, B. Szabo, R. Actis, "Hierarchic models for laminated composites", *Internat. J. Numer. Meth. Engrg.* **33**, (1992), pp. 503-536.
- [25] D. Blanchard, G. A. Francfort, "Asymptotic thermoelastic behavior of flat plates", *Quart. Appl. Math.* **45**, (1991), pp. 645-647.
- [26] P. G. Ciarlet, P. Destuynder, "A justification of the two-dimensional plate model", *J. Mécanique* **18**, (1979), pp. 315-344.
- [27] P.G. Ciarlet, *Plates and Junctions in Elastic Multi-Structures*, Masson-Springer, 1990.
- [28] P. G. Ciarlet, *Mathematical Elasticity, Vol II: Theory of Plates*, North Holland, 1997.
- [29] A. Sène, *Modélisation asymptotique de plaques: Controlabilité exacte frontière, piézoélectricité*, Doctoral Dissertation, Université Joseph Fourier-grenoble I, 1999.
- [30] M. Bernadou, C. Haenel, "Quelques remarques sur les coques piézoélectriques", *Rapport du Recherche*, INRIA France, 1994.
- [31] P. Bisegna, F. Maceri, "An exact three-dimensional solution for simply supported rectangular piezoelectric plates", *J. Appl. Mech.* **63**, (1996), pp. 628-638.
- [32] J. Huang, T. Wu, "Analysis of hybrid multilayered piezoelectric plates", *Int. J. Engrg. Sci* **34**, (1996), pp. 171-181.
- [33] P. Heyliger, "Static behavior of laminated elastic/piezoelectric plates", *AIAA J.* **32**, (1994), pp. 2481-2484.
- [34] S. S. Vel, R. C. Batra, "Three-dimensional analytical solution for hybrid multilayered piezoelectric plates", *J. Appl. Mech.* **67**, (2000), pp. 558-567.
-

- 
- [35] ANSYS *Program documentation*, SAS IP Inc., 1999.
- [36] SOLIDIS-ISE *Users Manual*, ISE AG (CH), 1998.
- [37] I. Babuška and W. Reinboldt, "Error estimates for adaptive finite element computations", *SIAM J. Numer. Anal.* **15**, (1978), pp. 736-754.
- [38] W.C. Rheinboldt, "On a theory of mesh-refinement processes", *SIAM J. Numer. Anal.* **17**, (1980), pp. 766-778.
- [39] J. P. Gago, J.P.;D. W. Kelly, O. C. Zienkiewicz, "A posteriori error analysis and adaptive processes in the finite element method. II. Adaptive mesh refinement", *Int. J. Numer. Methods Eng.* **19**, (1983), pp. 1621-1656.
- [40] I. Babuška, O.C. Zienkiewicz, J. P. Gago, and E.R.de A. Oliveira, Eds. *Accuracy Estimates and Adaptive Refinements in Finite Element Computations.*, Wiley, 1986.
- [41] R.E. Bank and A. Weiser, "Some a posteriori error estimators for elliptic partial differential equations", *Math. Comput.* **44**, (1985), pp. 283-301.
- [42] O.C. Zienkiewicz and J.Z. Zhu, "A simple error estimator and adaptive procedure for practical engineering analysis", *Int. J. Numer. Meth. Engrg.* **24**, (1987), pp. 337-357.
- [43] C.J. Ribbens, "A fast adaptive grid scheme for elliptic partial differential equations", *ACM Trans. Math. Software* **15**, (1989), pp.179-197.
- [44] F. Brezzi, M. Fortin, R. Stenberg, "Error analysis of mixed-interpolated elements for Reissner-Mindlin plates", *Math. Models Methods Appl. Sci.* **1**, (1991), pp. 125-151.
- [45] C. Schwab, "A-posteriori modelling error estimation for hierarchic plate models", *Numer. Math.* **74**, (1996), pp. 221-259.
- [46] R. Verfürth, *A Review of A Posteriori Error Estimation and Adaptive Mesh Refinement Techniques*, Wiley-Teubner, 1996.
- [47] D. Braess, R. Verfürth, "A posteriori error estimators for the Raviart-Thomas element", *SIAM J. Numer. Anal.* **33**, (1996), pp.2431-2444.
- [48] R.H.W. Hoppe, B. Wohlmuth, "Adaptive multilevel techniques for mixed finite element discretisation of elliptic boundary value problems", *SIAM J. Numer. Anal.* **34**, (1997), pp. 1658-1681.
- [49] R. Verfürth, *Fehlerschätzer für Finite Element Methoden in der linearen Elastizitätstheorie*, Report, Ruhr-Universität Bochum, 1994.
- [50] E.M. Abdalass, *Resolution performante du problème de Stokes par mini-éléments, maillages auto-adaptifs et méthodes multigrilles - applications*, Thèse de 3me cycle, Ecole Centrale de Lyon, 1987.
-

- 
- [51] R.E. Bank, B.D. Welfert, “A posteriori error estimates for the Stokes equations”, *Comput. Meth. Appl. Mech. Engrg.* **82**, (1990), pp. 323-340.
- [52] J.Z. Zhu, O.C. Zienkiewicz, “Adaptive techniques in the finite element method”, *Comm. Appl. Numer. Meth.* **4**, (1988), pp. 197-204.
- [53] K. Eriksson, C. Johnson, “An adaptive finite element method for linear elliptic problems”, *Math. Comput.* **50**, (1988), pp. 361-383.
- [54] E. Baensch, A. Schmidt, “Simulation of dendritic crystal growth with thermal convection”, *Interfaces Free Bound.* **2**, (2000), pp. 95-115.
- [55] E. Baensch, K. Deckelnick, “Optimal error estimates for the Stokes and Navier-Stokes equations with slip boundary condition”, *M2AN, Math. Model. Numer. Anal.* **33**, (1999), pp. 923-938.
- [56] A. Schmidt, K. G. Siebert, “Adaptive Finite Element Methods: Numerical Aspects of Parabolic Free Boundary Problems”, *Lecture Notes - Summer School*, Jyväskylä, 1996.
- [57] O.C. Zienkiewicz, J.Z. Zhu, “Error estimates and adaptive refinement for plate bending problems”, *Int. J. Numer. Methods Eng.* **28**, (1989), pp. 2839-2853.
- [58] S. Taschini, J. Müller, A. Greiner, M. Emmenegger, H. Baltes, J.G. Korvink, “Accurate Modeling and Simulation of Thermo-mechanical Microsystem Dynamics”, *Computer Modeling and Simulation in Engineering* **1**, (2000), pp. 31–44.
- [59] S. Taschini, J. G. Korvink, H. Baltes, “Computational modeling of piezoelectric microdevices”, (*to be published*).
- [60] L. Ferrario, S. Taschini, M. Emmenegger, H. Baltes, M. Zen, J. G. Korvink, “A new computational method for piezoelectric plate modeling: Application to membrane sensors”, (*to be published*).
- [61] M. Emmenegger, S. Taschini, J. G. Korvink, H. Baltes, “Simulation of a Thermomechanically Actuated Gas Sensor” *Proc. IEEE International Workshop on Micro Electro Mechanical Systems (MEMS)*, Heidelberg, (1998), pp. 184-189.
- [62] M. Emmenegger, J. G. Korvink, M. Bächtold, M. von Arx, O. Paul, H. Baltes, “Application of Harmonic Finite Element Analysis to a CMOS Heat Capacity Measurement Structure“, *Sensors and Materials* **10**, (1998), pp. 405-412.
- [63] M. Emmenegger, *An Object Oriented Design for Efficient Microsystem Simulation*, PH. D. Thesis, ETH Zürich, 1999.
- [64] J. Altenbach, *Einführung in die Kontinuumsmechanik*, Teubner, 1994.
-

- 
- [65] K. Huang, *Statistical mechanics*, Wiley, 1987.
- [66] H. Römer, T. Filk, *Statistische Mechanik*, VCH, 1994.
- [67] J. Honerkamp, *Statistical physics : an advanced approach with applications*, Springer, 1998.
- [68] L. D. Landau, E.M. Lifshitz, *Theory of Elasticity*, Pergamon, Oxford, 1986.
- [69] H. von Helmholtz, “Chemische Annalen für die Freunde der Naturlehre” (Ed. Crell) **100**, 137 (1886) §2.
- [70] I. Fényes, “Die Anwendung der mathematischen Prinzipien der Mechanik in der Thermodynamik”, *Z. Phys.* **132**, (1952), pp. 140-145.
- [71] P. T. Landsberg, *Thermodynamics : with quantum statistical illustrations*, Interscience Publ., 1961.
- [72] H. Goldstein, *Classical mechanics*, Addison-Wesley, 1981.
- [73] O. Madelung, *Introduction to solid-state theory*, Springer, 1996.
- [74] G. Kluge, G. Neugebauer, *Grundlagen der Thermodynamik*, Spektrum Akad. Verl., 1994.
- [75] J. D. Jackson, *Classical Electrodynamics*, John Wiley, 1999.
- [76] B. D. Reddy, *Introductory Functional Analysis*, Springer, 1998.
- [77] D. Braess, *Finite Elemente*, Springer, 1997.
- [78] G. Strang, G. Fix, *An Analysis of the Finite Element Method*, Wellesley-Cambridge, 1997.
- [79] P. G. Ciarlet, *The Finite Element Method for Elliptic Problems*, North-Holland, 1987.
- [80] W. Hackbusch, *Theorie und Numerik elliptischer Differentialgleichungen*, Teubner, 1996.
- [81] R Barret, et al, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, SIAM, Philadelphia PA, 1994.
- [82] Y. Saad, *Iterative Methods for Sparse Linear Systems*”, PWS Publishing Company, 1996.
- [83] G. Booch, *Object-Oriented Analysis and Design*, Addison-Wesley, 1994.
- [84] J.G. Korvink, *An Implementation of the Adaptive Finite Element Method for Semiconductor Sensor Simulation*, Ph. D. Thesis, ETH Zürich, 1993.
- [85] D. Lange, T. Akiyama, C. Hagleitner, A. Tonin, H. R. Hidber, P. Niedermann, U. Staufer, N. F. de Rooij, O. Brand, and H. Baltes, “Parallel Scanning AFM with On-chip Circuitry in CMOS Technology”, *Proc. IEEE MEMS '99*, Orlando, (1999), pp. 447 - 453.
- [86] O. Brand, *Micromachined Resonators for Ultrasound Based Proximity Sensing*, Ph.D. Thesis, ETH Zürich, 1994.
-

- 
- [87] M. von Arx, *Thermal Properties of CMOS Thin Films*, Ph.D. Thesis, ETH Zürich, 1998.
- [88] V. Ziebart, *Mechanical Properties of CMOS Thin Films*, Ph. D. Thesis, ETH Zürich, 1999.
- [89] S. Koller-Lucaea, *Resonating Viscosity Sensors- Micro versus Macro Approach*, Ph.D. Thesis, ETH Zürich, 1999.
- [90] W. Menz, J. Mohr, O. Paul, *Microsystem technology*, Wiley-VCH, 2001.
- [91] J. E. Sader, "Parallel beam approximation for V-shaped atomic force microscope cantilevers", *Rev. Sci. Instrum.* **66**, (1995) pp. 4583-4587.
- [92] C. Serre, A. Perez-Rodriguez, J.R. Morante, P. Gorostiza, J. Esteve, "Determination of micromechanical properties of thin films by beam bending measurements with an atomic force microscope", *Sensors and Actuators A* **74**, (1999), pp. 134-138.
- [93] J. Bühler, *Deformable Micromirror Arrays by CMOS Technology*, Ph. D. Thesis, ETH Zürich, 1997.
- [94] A. Nathan, H. Baltes, *Microtransducer CAD*, Springer-Verlag, 1999.
- [95] N. W. Ashcroft, N. D. Mermin, *Solid State Physics*, Holt Rinehart and Winston, 1976.
- [96] J.W. Gibbs, E.B. Wilson, *Vector Analysis*, Yale University Press, 1901.
- [97] J. F. Nye, *Physical Properties of Crystals*, Oxford University Press, 1985.
- [98] G. Kirchhoff, *Vorlesungen über Mathematische Physik*, Mechanik, 1876.
- [99] A.E.H Love, *The Mathematical Theory of Elasticity*, Cambridge University Press, 1934.
- [100] E. Reissner, "The effect of transverse shear deformations on the bending of elastic plates", *J. Appl. Mech.* **12**, (1945), A69-A77.
- [101] E. Reissner, "Reflections on the theory of elastic plates", *Appl. Mech. Rev.* **38**, (1985), pp. 227-233.
- [102] R. D. Mindlin, "Influence of rotatory inertia and shear on flexural motions of isotropic elastic plate", *J. Appl. Mech.* **18**, (1951), pp. 31-38.
- [103] M. Bernadou, *Finite Element Methods for Thin Shell Problems*, John Wiley & Sons, 1996.
- [104] J. H. Argyris, I. Fried, D. W. Scharpf, "The TUBA family of plate elements for the matrix displacement method", *Aero. J. Royal Aeronaut. Soc.* **72**, (1968), pp. 701-709.
- [105] O. C. Zienkiewicz, R. L. Taylor, *The Finite Element Method*, McGraw-Hill, 1989.
-

- 
- [106] S. Taschini, *Modeling of Slender Structures in Microsystems*, Ph. D. Thesis, ETH Zürich, 2000.
- [107] Ph. Clément, “Approximation by finite element functions using local regularization”, *RAIRO Anal. Numer.* **2**, (1975), pp. 77-84.
- [108] M. Emmenegger, S. Taschini, J. G. Korvink, H. Baltes, “Simulation of a thermomechanically actuated gas sensor”, *Proc. IEEE MEMS '98*, Heidelberg, (1998), pp. 184-189.
- [109] M. Emmenegger, J. G. Korvink, M. Bächtold, M. von Arx, O. Paul, H. Baltes, “Application of Harmonic Finite Element Analysis to a CMOS Heat-Capacity Measurement Structure”, *Sensors and Materials* **10** (1998) 405-412.
- [110] F. Reif, *Fundamentals of Statistical and Thermal Physics*, MacGraw-Hill, 1965.
- [111] E. Bänsch, Local mesh refinement in 2 and 3 dimensions, *Impact Comp. Sc. Engrg* **3**, (1991), pp. 181-191.
- [112] M.C.Rivara, *Algorithms for refining triangular grids suitable for adaptive and multigrid techniques. Internat. J. Numer. Meth. Engrg.* **20**, (1984), pp. 745-756.
- [113] I.G. Rosenberg and F. Stenger, *A lower bound on the angle of triangles constructed by bisecting the longest side. Math. Comp.* **29**, (1975), pp. 390-395.
- [114] I. Kossaczky, “A recursive approach to local mesh refinement in two and three dimensions”, *J. Comput. Appl. Math.*, **55**, (1994), pp. 275-288.
- [115] O. Forster, *Integralrechnung im Reellen mit Anwendungen, Analysis Bd. 3*, Vieweg, 1999.
- [116] J. Müller, J. G. Korvink, “Automatic Mesh Adaptivity for Finite Element Simulation of Multi-Layer MEMS”, in *Materials and Device Characterisation in Micromachining III*, Y. Vladimirski, P.J. Coane, eds, Proc. SPIE 4175, Santa Clara, (2000), pp. 82-93.
- [117] J. Müller, J. G. Korvink, “Automatic Accuracy control for Large Aspect Ratio Multi-Layer MEMS Simulation”, *Proc. Euroensors XIV*, Kopenhagen, (1999), pp. 277-278.

---

---

---

Jens Müller

IMTEK at the Albert-Ludwigs-University of Freiburg im Breisgau

Fakultät für Angewandte Wissenschaften

Georges-Köhler-Allee 103

D-79085 Freiburg i. Br., Germany

[jmueller@imtek.uni-freiburg.de](mailto:jmueller@imtek.uni-freiburg.de)

---

# ACKNOWLEDGEMENTS

First of all I would like to thank Prof. Dr. Jan G. Korvink for providing the opportunity to do this work within the simulation group at the IMTEK. I would like to thank him for his great support during difficult times and his enthusiasm for the work in good times. The wonderful atmosphere that he generates in his working group has contributed a lot to the success of this work.

I would also wish to thank Prof. W. Menz for co-examining this thesis.

I owe special thanks to Dr. Markus Emmenegger for giving me an understanding of object oriented software design and Dr. Stefano Taschini for acquainting me with plate theories and for providing the Argyris triangle.

I am grateful to Lorenza Ferrario for implementing the piezo-electrical part of the finite element plate equations.

It is a pleasure to me to thank Ms. Anne Rottler and Ms. Minako Mori for their efficient administrative support.

I would like to thank all my colleagues of the simulation group for their diverse contributions to the accomplishment of this thesis: Dr. Andreas Greiner, Ricardo Osorio, Darius Koziol, Kaiping Zeng, Dr. Evgenii Rudnyi, Dr. Zhenyu Liu, Tamara Bechtold, Jan Lienemann and Sadik Hafizovic.

This work was partly funded by the DFG (Deutsche Forschungsgemeinschaft).

---

---

# CURRICULUM VITAE

Jens Müller

Born September 26, 1969

1976-1980 Grundschule Maulburg

1980-1989 Gymnasium, Schopfheim

1989 Abitur

1989-1990 Military service, Stetten a.k.M.

1990-1991 Studies in pharmaceutics, University of Freiburg

1991-1997 Studies in mathematics and physics, Albert-Ludwigs University of Freiburg

1996-1997 Working at the physics department at Freiburg University in collaboration with the ZEUS experiment at DESY, Hamburg

1997 Staatsexamen degree in mathematics and physics, University of Freiburg

since 1998 Working at the IMTEK-department at the Freiburg University pursuing the doctoral degree under the supervision of Prof. Dr. J.G. Korvink

---

---

# ABBREVIATIONS AND SYMBOLS

2D	two-dimensional
3D	three-dimensional
$A$	Reduced elastic tensor
$a$	thermal expansion tensor
AFM	atomic force microscope
BVP	boundary value problem
$\mathbb{C}$	set of complex numbers
$C^m(\Omega)$	space of functions of $m$ -fold continuously differentiable functions in domain $\Omega$ .
$C_\tau$	heat capacity at a constant value of $\tau$
$C = (C)_{ijkl} = (C)_{klij}$	4th rank elasticity tensor
C++	C++ programming language
CMOS	complementary metal-oxide semiconductor, a logic family and the related process technology
$D$	electrical displacement
$D$	flexural rigidity of a plate
$d$	piezoelectric strain coefficients, usually in units of $C/N$ .
$E$	electric field
$E, E_i,  E_i\rangle$	internal energy, energy eigenvalues, energy eigenstates
$E$	Young's modulus
$F$	Helmholtz free energy

---

$\hat{F}$	density of the Helmholtz free energy
$F_s$	prescribed shear force
$f(\mathbf{x})$	body force density vector field
$f$	heat source
FEM	finite element method
$h$	surface electrical displacement
$h_T, h_E$	simplex diameter, edge length
$H$	Hamiltonian, Hamilton operator
$H^n(\Omega)$	Sobolev space of order $n$ on domain $\Omega$ .
IC	integrated circuit
$\mathbf{K}$	stiffness matrix
$k_B$	Boltzmann constant
LHS	left hand side
$L$	Lagrangian, linear differential operator
$\hat{L}$	Lagrangian density
$L^\infty, L^2$	space of bounded measurable, space of square integrable functions
MEMS	micro-electro-mechanical systems
$\mathbf{M}_b$	prescribed bending moments
$N_i$	$i$ -th basis function or shape function
$\mathbf{n}$	surface normal vector
OOA	object oriented analysis
OOD	object oriented design
$\mathbb{P}_i$	space of polynomials of degree $i$
PDE	partial differential equation
$\mathbf{p}$	pyroelectric tensor

---

---

$\mathbb{R}$	set of real numbers
RHS	right-hand side
$S, \hat{S}$	entropy, entropy density
$T$	transformation matrix from reference element to element
$t$	time
$T$	temperature
$\hat{T}$	reference simplex
$U = (\mathbf{u}, w)$	plate displacement field
$U$	potential
$\mathbf{u}$	in-plane displacement field in 2D, also used for 3D mechanical displacement
$V$	potential energy, also used to designate several function spaces
$V$	volume
$\mathbf{V} = (\mathbf{v}, v)$	variation of plate displacement field
$\mathbf{v}$	variation of in-plane displacement field
$W_E$	(non-linear) elastic virtual work
$w$	out-of-plane displacement
$w_i$	occupation probability of i-th state
$Z$	partition function
$\mathbf{x}$	position vector in space
$\alpha$	thermal expansion coefficient tensor
$\beta$	$(k_B T)^{-1}$ , inverse of product of Boltzmann constant and temperature
$\Gamma_N$	Neumann boundary
$\Gamma_D$	Dirichlet boundary
$\Delta$	Laplacian differential operator

---

---

$\delta_{ij}$	Kronecker delta symbol
$\varepsilon$	strain tensor
$\varepsilon_0$	dielectric constant
$\eta_T$	element error estimator
$\theta$	time primitive of the temperature
$\kappa$	heat conductivity tensor
$\lambda$	Lamé constant
$\lambda_i$	barycentric coordinates
$\mu$	Lamé constant
$\nu$	Poisson number
$\pi, \hat{\pi}$	piezoelectric tensor, reduced piezoelectric tensor
$\rho$	electrical charge density mass density also used for quantum mechanical density operator
$\sigma$	stress tensor
$\phi$	electrostatic potential
$\chi$	electrical permittivity
$\psi$	variation of electric potential
$\Psi_T, \Psi_E$	triangle-, edge bubble function
$\Omega, \omega, \bar{\Omega}, \bar{\omega}$	(open) 3D , 2D domains and closures
$\partial\Omega$	boundary of the domain $\Omega$
$\emptyset$	empty set

---

---

# INDEX

<b>A</b>		<b>I</b>	
a posteriori error	74	inner product	32
Atomic Force Microscope	136	inner product space	32
<b>B</b>		<b>K</b>	
Banach spaces	34	Kirchhoff-Love theory	60
bilinear forms	35	<b>L</b>	
BinaryTree	108	Lagrangian	21
boundary value problems (BVP)	31	Lax-Milgram theorem	37
Bravais lattice	16	Legendre transformation	15
<b>C</b>		<b>M</b>	
Cauchy-Schwarz inequality	32	Maxwell relations	15, 25
conforming	52	<b>N</b>	
continuity	37	natural boundary conditions	42
crystal classes	19	norm	33
<b>D</b>		<b>O</b>	
Dirichlet boundary conditions	44	OOA	101
<b>E</b>		OOD	101
elastic tensor	18	open sets	126
equivalence of norms	34	orthogonal	32
essential boundary conditions	42	orthogonality of the error	48
<b>F</b>		<b>P</b>	
flexural rigidity	95	partition function	12
Fourier's law	22	positive definite	37
functionals	35	<b>R</b>	
<b>G</b>		reference triangle	56
Gibbs fundamental form	14	Reissner-Mindlin	60
<b>H</b>		residual error estimation	74
Hamilton's principle	21	<b>S</b>	
Hamiltonian	12	shape regular	50
harmonic potential	17	Sobolev embedding theorem	42
H-ellipticity	37	Sobolev space	40
Helmholtz free energy	15	square-integrable functions	32
hermodynamic potentials	15	susceptibilities	15, 26
Hilbert space	34		

---

---