



Jakob Harden, BSc

The Mortar Method

An advanced coupling method

MASTER'S THESIS

to achieve the university degree of

Diplom-Ingenieur

Master's degree programme: Civil Engineering and Structural Engineering

submitted to

Graz University of Technology

Supervisor

Dipl.-Ing. Dr.techn. Univ.-Doz. Christian Dünser

Institute of Structural Analysis

Graz, September 2016

AFFIDAVIT

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master's thesis.

Date

Signature

ZUSAMMENFASSUNG

Gegenstand dieser Masterarbeit ist die Betrachtung der physikalischen und mathematischen Grundlagen der *Mortar Method* (MM). Diese Kopplungsmethode ermöglicht die Kopplung von Regionen die mit der *Randelemente Methode* (BEM) oder der *Finite Elemente Methode* (FEM) diskretisiert wurden und über nicht-kohärente Netze oder inkompatible Ansatzfunktionen verfügen durch einen Ansatz in schwacher Form. Im Zuge dieser Arbeit wird das Kopplungsproblem zuerst in analytischer Form mit den Hilfsmitteln der Kontinuumsmechanik beschrieben und dann ein Ansatz in diskreter Form zur numerischen Berechnung formuliert. Ein einfaches Beispiel mit der *Finite Elemente Methode* (FEM) zeigt wie man die diskrete Berechnung der Kopplungsbedingungen durchführen kann. Am Ende dieser Arbeit erfolgt eine Parameterstudie für einen einfachen Kragträger in 2D und 3D mit der *Randelemente Methode* anhand der das Konvergenzverhalten und der auftretende Diskretisierungsfehler dargestellt werden.

ABSTRACT

Subject to this Master's Thesis are the physical and mathematical basics of the *Mortar Method* (MM). This coupling method offers a coupling approach for regions discretized with the *Boundary Element Method* (BEM) or the *Finite Element Method* (FEM) having non-coherent meshes or incompatible test functions using a discretisation approach in a weak sense. Further on this thesis the coupling problem will first be expressed analytically using the tools of the continuum mechanics and later an approach in discrete form for the numerical analysis is shown. As an introductory example the *Finite Element Method* (FEM) is used to show the process of a discrete computation of the coupling conditions. At the end is worked out a parameter study for a simple cantilever beam in 2D and 3D which will show the convergence behaviour and existing discretisation errors.

Chapter 1

Introduction

This chapter describes the motivation for the *Mortar Method* and its basic ideas. Further topics of this chapter are coupling methods and a preview to the chapters of this Master's Thesis.

1.1 Motivation for the *Mortar Method*

This method was first introduced to couple wave propagation problems with mechanical systems. Since those two problems need to have very different numerical discretisations there was a need for a coupling method that can cope with both of those approximations. This gave rise to a coupling method which uses energy functions at the interface of two different coupled problems called "*Mortar Method*" [1]. The method was then developed to couple regions of the same physical kind but with non-conforming meshes. This can happen if we think of problems which need to have different mesh densities for adjoining regions due to computational accuracy [2]. Or think of body contact problems which usually lead to non-conforming meshes. Another application are systems with meshes using different kind of element shape functions.

1.2 Motivation for this Master's Thesis

The initial idea was developed at the *Institute of Structural Analysis at Graz University of Technology*. Beside all the theory and publications there was the idea to have a practical approach and an implementation for simple but realistic elasto-static systems. The very start of this thesis was a literature research. Later the basic algorithms are developed and an implementation for the scientific software program *BEFE++* [3] was done. Finally a convergence study shall show the advantages and disadvantages of the *Mortar Method* and the chosen algorithms.

1.3 *Mortar Method* fundamentals

Basically the *Mortar Method* is a domain decomposition/coupling method. In this work it is used to couple two or more subdomains having geometrically non-conforming meshes (non-coherent mesh nodes at the sub-domain interface). The idea of this coupling method is to introduce interface fields with additional degree's of freedom in the form of *Lagrange multipliers*. The interface fields are a subspace of two adjacent sub-domains, therefore they represent a dual space. To meet that constraint the sub-domains *must* be non-overlapping. Finally this coupling method states an equilibrium of mechanical energy in each interface field. This leads to additional equations for the unknown *Lagrange multipliers*. Since this method does not control the displacements of the interface nodes directly but the equilibrium of mechanical energy at each interface field it is an approach in a weak sense. For the mathematical formulation, convergence studies and error estimates publications like [4], [5], [6] and [7] are recommended.

Basic facts:

- The *Mortar Method* is a coupling method
- Coupling of geometrically non-conforming, non-overlapping subdomain meshes
- Energy conservation at the sub-domain interface
- Introducing interface fields and *Lagrange multipliers*
- Approach in a weak sense

1.4 Other coupling methods

There are a lot of other coupling methods beside the *Mortar Method*. Two of the most important domain decomposition methods are *FETI* and *BETI* coupling because they allow for parallel computing. For that reason a short introduction to this methods is given here.

General properties of *FETI* and *BETI* coupling method:

- The stiffness matrices are assembled for each sub-domain.
- Coupling matrices are used to describe the compatibility of nodal displacements at the sub-domain interfaces.
- Dirichlet boundary conditions are incorporated by using additional constraint conditions.
- The stiffness matrices can be computed in parallel.
- The stiffness matrices do not have to be reassembled for changing coupling- and boundary conditions.
- Iterative solvers can compute the equation system piece-wise so there is no need to assemble one single system matrix for all regions.

1.4.1 *FETI* coupling

The name is a short-cut derived from the name **F**inite **E**lements, **T**earing and **I**nterconnecting.

Equation system:

$$[\mathbf{A}] \cdot \{\mathbf{x}\} = \{\mathbf{b}\} \tag{1.1}$$

$$\begin{bmatrix} [\mathbf{K}^1] & [\mathbf{0}] & \cdots & \cdots & [\mathbf{0}] & [\mathbf{C}^1]^\top & [\mathbf{D}^1]^\top \\ [\mathbf{0}] & \ddots & \ddots & & \vdots & \vdots & \vdots \\ \vdots & \ddots & [\mathbf{K}^m] & \ddots & \vdots & [\mathbf{C}^m]^\top & [\mathbf{D}^m]^\top \\ \vdots & & \ddots & \ddots & [\mathbf{0}] & \vdots & \vdots \\ [\mathbf{0}] & \cdots & \cdots & [\mathbf{0}] & [\mathbf{K}^n] & [\mathbf{C}^n]^\top & [\mathbf{D}^n]^\top \\ [\mathbf{C}^1] & \cdots & [\mathbf{C}^m] & \cdots & [\mathbf{C}^n] & [\mathbf{0}] & [\mathbf{0}] \\ [\mathbf{D}^1] & \cdots & [\mathbf{D}^m] & \cdots & [\mathbf{D}^n] & [\mathbf{0}] & [\mathbf{0}] \end{bmatrix} \cdot \begin{Bmatrix} \{\mathbf{u}^1\} \\ \vdots \\ \{\mathbf{u}^m\} \\ \vdots \\ \{\mathbf{u}^n\} \\ \{\boldsymbol{\lambda}_c\} \\ \{\boldsymbol{\lambda}_d\} \end{Bmatrix} = \begin{Bmatrix} \{\mathbf{f}_0^1\} \\ \vdots \\ \{\mathbf{f}_0^m\} \\ \vdots \\ \{\mathbf{f}_0^n\} \\ \{\mathbf{0}\} \\ \{\mathbf{u}_0\} \end{Bmatrix}$$

- $[\mathbf{K}^m]$, stiffness matrix of subdomain m
- $[\mathbf{C}^m]$, coupling matrix of subdomain m for the interface boundary
- $[\mathbf{D}^m]$, Dirichlet constraint matrix of subdomain m for the Dirichlet boundary
- $\{\mathbf{u}^m\}$, displacement degree's of freedom of subdomain m

- $\{\mathbf{f}_0^m\}$, load vector of subdomain m
- $\{\boldsymbol{\lambda}_c\}$, Lagrange multiplier for coupling degree's of freedom
- $\{\boldsymbol{\lambda}_d\}$, Lagrange multiplier for Dirichlet degree's of freedom

Equilibrium of forces: The equilibrium of forces for each subdomain is now extended by the unknown nodal forces at the interface and the Dirichlet boundary.

$$[\mathbf{K}^m] \cdot \{\mathbf{u}^m\} + [\mathbf{C}^m]^\top \cdot \{\boldsymbol{\lambda}_c\} - [\mathbf{D}^m]^\top \cdot \{\boldsymbol{\lambda}_d\} = \{\mathbf{f}_0^m\} \quad (1.2)$$

The nodal forces minus the nodal forces at the interface and/or the Dirichlet boundary have to be equal to the external forces due to the displacement.

Coupling conditions: For geometric compatibility the sum of the displacements at any node of the interface of n adjacent subdomains has to be zero.

$$\sum_{m=1}^n [\mathbf{C}^m] \cdot \{\mathbf{u}^m\} = \{\mathbf{0}\} \quad (1.3)$$

The coupling matrix entries are defined like:

- +1, for any degree of freedom u_i at the interface of the current subdomain
- -1, for all degree's of freedom collocated to u_i of all adjacent subdomains

The *RHS* vector for this part of the equation system is a zero vector.

Dirichlet boundary conditions: The displacement at any node along the Dirichlet boundary has to be equal to a given initial displacement $\{\mathbf{u}_0\}$.

$$\sum_{m=1}^n [\mathbf{D}^m] \cdot \{\mathbf{u}^m\} = \{\mathbf{u}_0\} \quad (1.4)$$

The Dirichlet matrix entries are defined like:

- 1, for all degree's of freedom at the Dirichlet boundary
- 0, else

The *RHS* vector $\{\mathbf{u}_0\}$ for this part of the equation system is:

- $u_i = 0$, for locked degree's of freedom
- $u_i = u_{i,0}$, for initially displaced degree's of freedom

1.4.2 *BETI* coupling

The name is a short-cut derived from the name **B**oundary **E**lements, **T**earing and **I**nterconnecting. The very beginning of that approach is the boundary integral equation.

$$\mathbf{C}(\mathbf{x}) \cdot \mathbf{u}(\mathbf{x}) = \oint_{\Gamma} \mathbf{U}(x, y) \cdot \mathbf{t}(\mathbf{x}) d\Gamma - \oint_{\Gamma} \mathbf{T}(x, y) \cdot \mathbf{u}(\mathbf{x}) d\Gamma \quad (1.5)$$

The parts of this equation are:

- $\mathbf{C}(\mathbf{x})$, integral free term
- $\mathbf{u}(\mathbf{x})$, displacements

- $\mathbf{t}(\mathbf{x})$, tractions
- $\mathbf{U}(x, y)$, fundamental solution of the displacements
- $\mathbf{T}(x, y)$, fundamental solution of the tractions
- Γ , boundary surface

The discrete form of equation (1.5) using the *Collocation-Method* is:

$$[\Delta \mathbf{U}^n] \cdot \{\mathbf{t}^n\} = [\Delta \mathbf{T}^n] \cdot \{\mathbf{u}^n\} \quad (1.6)$$

Based on the equation (1.6) the tractions for subdomain Ω^n can then be written like:

$$\{\mathbf{t}^n\} = [\Delta \mathbf{U}^n]^{-1} \cdot [\Delta \mathbf{T}^n] \cdot \{\mathbf{u}^n\} \quad (1.7)$$

where entries of $[\Delta \mathbf{U}^n]$ and $[\Delta \mathbf{T}^n]$ are integrated kernel-shape-function products over the elements at the boundary Γ .

The traction vector consists of two parts:

$$\{\mathbf{t}^n\} = \{\mathbf{t}_0^n\} + \{\mathbf{t}_c^n\} \quad (1.8)$$

where $\{\mathbf{t}_0^n\}$ are the tractions due to the loading and $\{\mathbf{t}_c^n\}$ are the tractions of the interface of two subdomains. Now we plug equation (1.8) into equation (1.7) and get:

$$\{\mathbf{t}_0^n\} = [\Delta \mathbf{U}^n]^{-1} \cdot [\Delta \mathbf{T}^n] \cdot \{\mathbf{u}^n\} - \{\mathbf{t}_c^n\} \quad (1.9)$$

For the following steps we define a global traction vector $\{\mathbf{t}_c\}$ that can be transformed with the coupling matrices $[\mathbf{C}^n]$ into the local (subdomain) traction vector $\{\mathbf{t}_c^n\}$.

$$\{\mathbf{t}_c^n\} = [\mathbf{C}^n]^\top \cdot \{\mathbf{t}_c\} \quad (1.10)$$

Further we define a stiffness matrix for each subdomain which is:

$$[\mathbf{K}_t^n] = [\Delta \mathbf{U}^n]^{-1} \cdot [\Delta \mathbf{T}^n] \quad (1.11)$$

Now we insert equation (1.11) and (1.10) into equation (1.9) and get:

$$[\mathbf{K}_t^n] \cdot \{\mathbf{u}^n\} - [\mathbf{C}^n]^\top \cdot \{\mathbf{t}_c\} = \{\mathbf{t}_0^n\} \quad (1.12)$$

For the reason that we also want the possibility of coupling the *Finite Element Method* and *Boundary Element Method* we have to transform the tractions of the *Boundary Element Method* subdomains into nodal forces. This can be done using the mass matrix $[\mathbf{M}^n]$ of a subdomain [8]. Now the stiffness matrix $[\mathbf{K}_t^n]$ and the load vector $\{\mathbf{t}_0^n\}$ is multiplied with the mass matrix and the global traction vector $\{\mathbf{t}_c\}$ is replaced by the vector $\{\mathbf{f}_c\}$ of unknown nodal forces.

$$\begin{aligned} [\mathbf{K}^n] &= [\mathbf{M}^n] \cdot [\Delta \mathbf{U}^n]^{-1} \cdot [\Delta \mathbf{T}^n] \\ \{\mathbf{f}_0^n\} &= [\mathbf{M}^n] \cdot \{\mathbf{t}_0^n\} \end{aligned} \quad (1.13)$$

Now equation (1.12) can be written as:

$$[\mathbf{K}^n] \cdot \{\mathbf{u}^n\} - [\mathbf{C}^n]^\top \cdot \{\mathbf{f}_c\} = \{\mathbf{f}_0^n\} \quad (1.14)$$

This is the equilibrium of forces for each subdomain Ω^n . Now the equation system can be set up in the same way as it was done for the *FETI* coupling method.

$$\begin{bmatrix} [\mathbf{K}^1] & [\mathbf{0}] & \cdots & \cdots & [\mathbf{0}] & [\mathbf{C}^1]^\top & [\mathbf{D}^1]^\top \\ [\mathbf{0}] & \ddots & \ddots & & \vdots & \vdots & \vdots \\ \vdots & \ddots & [\mathbf{K}^m] & \ddots & \vdots & [\mathbf{C}^m]^\top & [\mathbf{D}^m]^\top \\ \vdots & & \ddots & \ddots & [\mathbf{0}] & \vdots & \vdots \\ [\mathbf{0}] & \cdots & \cdots & [\mathbf{0}] & [\mathbf{K}^n] & [\mathbf{C}^n]^\top & [\mathbf{D}^n]^\top \\ [\mathbf{C}^1] & \cdots & [\mathbf{C}^m] & \cdots & [\mathbf{C}^n] & [\mathbf{0}] & [\mathbf{0}] \\ [\mathbf{D}^1] & \cdots & [\mathbf{D}^m] & \cdots & [\mathbf{D}^n] & [\mathbf{0}] & [\mathbf{0}] \end{bmatrix} \cdot \begin{Bmatrix} \{\mathbf{u}^1\} \\ \vdots \\ \{\mathbf{u}^m\} \\ \vdots \\ \{\mathbf{u}^n\} \\ \{\boldsymbol{\lambda}_c\} \\ \{\boldsymbol{\lambda}_d\} \end{Bmatrix} = \begin{Bmatrix} \{\mathbf{f}_0^1\} \\ \vdots \\ \{\mathbf{f}_0^m\} \\ \vdots \\ \{\mathbf{f}_0^n\} \\ \{\mathbf{0}\} \\ \{\mathbf{u}_0\} \end{Bmatrix}$$

- $[\mathbf{K}^m]$, stiffness matrix of subdomain m
- $[\mathbf{C}^m]$, coupling matrix of subdomain m for the interface boundary
- $[\mathbf{D}^m]$, Dirichlet constraint matrix of subdomain m for the Dirichlet boundary
- $\{\mathbf{u}^m\}$, displacement degree's of freedom of subdomain m
- $\{\mathbf{f}_0^m\}$, load vector of subdomain m
- $\{\boldsymbol{\lambda}_c\}$, Lagrange multiplier for coupling degree's of freedom
- $\{\boldsymbol{\lambda}_d\}$, Lagrange multiplier for Dirichlet degree's of freedom

1.5 Roadmap to this Master's Thesis

This document is structured into several chapters and will lead from theoretical to practical computation topics.

- **Introduction:** Fundamentals of the *Mortar Method* and other coupling methods.
- **Continuum mechanics:** Theoretical part containing the formulation for elastic continua and a numerical approximation approach.
- **A simple example:** Computation example for a simple *Finite Element Method* system showing the principle way to compute coupling coefficients.
- **Convergence study:** Convergence study for 2D and 3D cantilever problems.
- **Conclusion and outlook:** Conclusion and outlook to applications and developments of the *Mortar Method*.

Chapter 2

Continuum mechanics

This chapter contains the physical and mathematical formulation of the *Mortar Method*. First the problem is expressed in terms of elastic continua and then a numerical approximation will be derived.

2.1 Preliminaries

Since the *Mortar Method* is a very general approach we want to make some assumptions to limit the complexity and the work-load for this Master's Thesis.

2.1.1 Assumptions and Limitations

- linear solid material.
- no dependence on temperature (isothermal).
- no dependence on time.
- small displacements.
- no energy dissipation (plasticity, friction, etc.).

2.1.2 Symbol names

- \mathbb{C} , the “linear elastic tensor”, tensor of 4th order
- \mathbf{T} , internal stress, tensor of 2nd order
- \mathbf{t} , traction
- \mathbf{t}_N , traction applied to the Neumann boundary
- \mathbf{t}_I , traction applied to the interface of two sub-domains
- \mathbf{b} , applied body force (e.g. gravity)
- $\boldsymbol{\varepsilon}$, linearised strain tensor
- \mathbf{u} , displacements
- \mathbf{u}_0 , applied (initial) displacements
- $\delta\mathbf{u}$, virtual displacement test function
- $\delta\mathbf{p}$, virtual force test function
- $\Omega, \Omega^{(i)}$, domain Ω or sub-domain $\Omega^{(i)}$

- $\Gamma_D, \Gamma_D^{(i)}$, Dirichlet boundary of a domain Ω or sub-domain $\Omega^{(i)}$
- $\Gamma_N, \Gamma_N^{(i)}$, Neumann boundary of a domain Ω or sub-domain $\Omega^{(i)}$
- $\Gamma_I^{(m,n)}$, interface of sub-domain $\Omega^{(m)}$ and $\Omega^{(n)}$
- γ_j , *Lagrange multiplier field/Mortar Finite Element*, field numbers j

2.1.3 Some expressions

System domain Ω : Let us assume to have small deformations in our system domain so we define

$$\Omega = \Omega_0 = \Omega_t \quad (2.1)$$

where Ω_0 is the initial configuration and Ω_t the deformed configuration.

Sub-domain $\Omega^{(n)}$: A sub-domain is a subset of particles of the system domain

$$\Omega^{(n)} \subset \Omega, \quad \bigcup_{n=1}^N \Omega^{(n)} = \Omega \quad (2.2)$$

where N is the total number of sub-domains.

Boundary of a domain: The boundary of a domain is a subset containing all particles at the surface of the domain.

$$\Gamma \subset \Omega \quad (2.3)$$

with

$$\Gamma_D \cup \Gamma_N \cup \Gamma_I = \Gamma \quad (2.4)$$

The very same applies to all sub-domains $\Omega^{(n)}$.

Neumann boundary:

$$\Gamma_N \subset \Gamma \subset \Omega \quad (2.5)$$

Dirichlet boundary:

$$\Gamma_D \subset \Gamma \subset \Omega \quad (2.6)$$

Interface:

$$\Gamma_I^{(m,n)} = \Gamma^{(m)} \cap \Gamma^{(n)} \quad (2.7)$$

The interface is a dual space.

$$\Gamma^{(m)} \cap \Gamma^{(n)} = \Omega^{(m)} \cap \Omega^{(n)} \quad (2.8)$$

The sub-domains are non-overlapping.

Partition or field γ_j of the interface Γ_I :

$$\gamma_j \subset \Gamma_I, \quad \bigcup_{j=1}^{n_j} \gamma_j = \Gamma_I \quad (2.9)$$

where n_j is the total number of partitions/fields of finite size. In further explanations we will introduce a *Lagrange multiplier* for each partition or field, henceforth they are called *Lagrange multiplier fields*.

Relation between stress and strain:

$$\mathbf{T} = \mathbb{C} \cdot \boldsymbol{\varepsilon} \quad (2.10)$$

This relation holds only for small deformations and linear elastic material behaviour and incorporates the assumptions from above.

2.2 Mechanics of a solid body

In the following an elasto-static problem is described assuming small deformations. The problem to be decomposed is shown in the figure below.

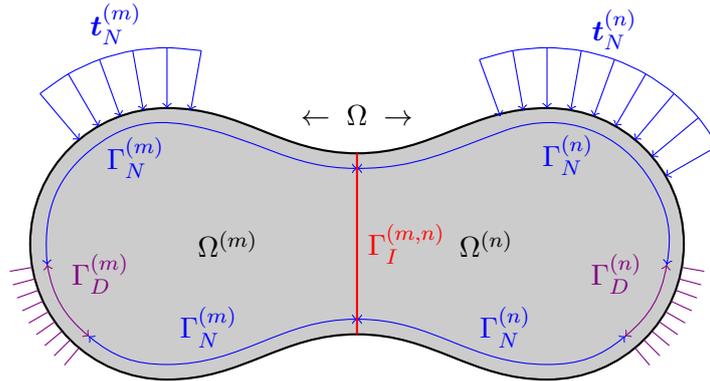


Figure 2.1: Solid body with two regions ($\Omega^{(m)}$, $\Omega^{(n)}$)

Now we cut the body along Γ_I and get additional tractions \mathbf{t}_I at the cutting surfaces Γ_I . These tractions are derived from the inner stress \mathbf{T} of the body. Their absolute value has to be equal but with opposite sign on the cutting surface of each subdomain, so their sum is zero. Figure 2.2 shows the subdomain $\Omega^{(m)}$ with the tractions $\mathbf{t}_I^{(m)}$ of the interface.

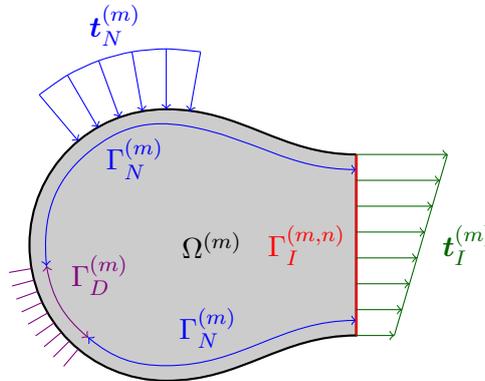


Figure 2.2: Slice of solid body, left region ($\Omega^{(m)}$)

2.2.1 Boundary value problem, strong form

To formulate the boundary value problem we use the static “Navier-Lamé equation”. First we do this for the whole domain Ω .

$$\begin{aligned} -\nabla \cdot \mathbf{T} &= \mathbf{b} && \dots \text{ on } \mathbf{x} \in \Omega, \text{ balance of linear momentum} \\ \mathbf{u} &= \mathbf{u}_0 && \dots \text{ on } \mathbf{x} \in \Gamma_D, \text{ initial displacements} \\ \mathbf{T} \cdot \mathbf{n} &= \mathbf{t}_N && \dots \text{ on } \mathbf{x} \in \Gamma_N, \text{ traction} \end{aligned} \quad (2.11)$$

Next we formulate this equations for any subdomain $\Omega^{(m)}$.

$$\begin{aligned}
-\nabla \cdot \mathbf{T} &= \mathbf{b} && \dots \text{ on } \mathbf{x} \in \Omega, \text{ balance of linear momentum} \\
\mathbf{u} &= \mathbf{u}_0 && \dots \text{ on } \mathbf{x} \in \Gamma_D^{(m)}, \text{ initial displacements} \\
\mathbf{T} \cdot \mathbf{n} &= \mathbf{t}_N^{(m)} && \dots \text{ on } \mathbf{x} \in \Gamma_N^{(m)}, \text{ traction} \\
\mathbf{T} \cdot \mathbf{n} &= \mathbf{t}_I^{(m)} && \dots \text{ on } \mathbf{x} \in \Gamma_I^{(m,n)}, \text{ traction}
\end{aligned} \tag{2.12}$$

2.2.2 Boundary value problem, weak form

To get the weak form of the equations above we will use the *Principle of Virtual Displacements*. For this we have to choose an admissible test function $\delta \mathbf{u}$, such that

$$\delta \mathbf{u} = \mathbf{u}_0 \quad \text{on } \Gamma_D \tag{2.13}$$

Balance of linear momentum:

$$-\nabla \cdot \mathbf{T} = \mathbf{b} \tag{2.14}$$

and it's residual form looks like

$$-\nabla \cdot \mathbf{T} - \mathbf{b} = 0 \tag{2.15}$$

now we apply the test functions (multiply (2.13) with (2.15))

$$(-\nabla \cdot \mathbf{T} - \mathbf{b}) \cdot \delta \mathbf{u} = 0 \tag{2.16}$$

and integrate over the domain

$$\int_{\Omega} \underbrace{(-\nabla \cdot \mathbf{T} - \mathbf{b})}_{T_1} \cdot \delta \mathbf{u} \, d\Omega = 0 \tag{2.17}$$

First we pick out the term T_1 (the scalar product is commutative)

$$T_1 = \int_{\Omega} (-\nabla \cdot \mathbf{T}) \cdot \delta \mathbf{u} \, d\Omega \tag{2.18}$$

Further we know about the following identity

$$(\nabla \cdot \mathbf{A}) \cdot \mathbf{x} \equiv \nabla \cdot (\mathbf{A}^\top \mathbf{x}) - \mathbf{A} : (\nabla \otimes \mathbf{x}) \tag{2.19}$$

where \mathbf{A} is a tensor of second order and \mathbf{x} a vector.

Now we rewrite the term T_1 according to this identity

$$T_1 = \int_{\Omega} \left(-\nabla \cdot \underbrace{(\mathbf{T}^\top \delta \mathbf{u})}_{T_2} + \mathbf{T} : (\nabla \otimes \delta \mathbf{u}) \right) d\Omega \tag{2.20}$$

then we apply the divergence theorem on T_2 , split the sum and receive for T_1

$$T_1 = - \underbrace{\int_{\Gamma} (\mathbf{T} \cdot \mathbf{n}) \delta \mathbf{u} \, d\Gamma}_{T_3} + \int_{\Omega} \mathbf{T} : \underbrace{(\nabla \otimes \delta \mathbf{u})}_{T_4} \, d\Omega \tag{2.21}$$

Next we apply the boundary conditions on T_3

$$\mathbf{T} \cdot \mathbf{n} = \mathbf{t}_N \quad \dots \text{ applied to Neumann boundary } \Gamma_N^{(m)} \tag{2.22}$$

$$\mathbf{T} \cdot \mathbf{n} = \mathbf{t}_I \quad \dots \text{ applied to the interface } \Gamma_I^{(m,n)} \quad (2.23)$$

and get

$$T_3 = - \int_{\Gamma_N^{(m)}} \mathbf{t}_N \delta \mathbf{u} d\Gamma - \int_{\Gamma_I^{(m,n)}} \mathbf{t}_I \delta \mathbf{u} d\Gamma \quad (2.24)$$

The term T_4 can be substituted by

$$T_4 = \text{sym}(\nabla \otimes \delta \mathbf{u}) = \boldsymbol{\varepsilon}(\delta \mathbf{u}) \quad (2.25)$$

Finally we get

$$\int_{\Omega^{(m)}} \mathbf{T} : \boldsymbol{\varepsilon}(\delta \mathbf{u}) d\Omega - \int_{\Gamma_I^{(m,n)}} \mathbf{t}_I \delta \mathbf{u} d\Gamma = \int_{\Omega^{(m)}} \mathbf{b} \delta \mathbf{u} d\Omega + \int_{\Gamma_N^{(m)}} \mathbf{t}_N \delta \mathbf{u} d\Gamma \quad (2.26)$$

Lagrange multiplier fields: First we pick up the term of the unknown interface tractions

$$\int_{\Gamma_I^{(m,n)}} \mathbf{t}_I \delta \mathbf{u} d\Gamma \quad (2.27)$$

and substitute the tractions by *Lagrange multipliers*

$$\mathbf{t}_I = \psi \cdot \boldsymbol{\lambda} \quad (2.28)$$

where ψ is a weight function which we define like

$$\psi := \delta_{ij} \quad \dots \text{ Kronecker-Delta} \quad (2.29)$$

$$\delta_{ij} = \begin{cases} 1 & , i = j \\ 0 & , i \neq j \end{cases} \quad (2.30)$$

where i is the index of the current point location at the interface and j the index of any other point location at the interface.

Approximation of the interface integral: As an approximation we substitute the integral over the interface $\Gamma_I^{(m,n)}$ with a sum over the *Lagrange multiplier fields* γ_j . The *point-wise* defined weight function ψ becomes now to be a *field-wise* defined function.

$$\int_{\Gamma_I^{(m,n)}} \mathbf{t}_I \delta \mathbf{u} d\Gamma = \sum_{j=1}^{n_j} \int_{\gamma_j} \mathbf{t}_I \delta \mathbf{u} d\Gamma \quad (2.31)$$

Each interface field γ_j shall now be connected to one *Lagrange multiplier* λ_j .

$$\int_{\gamma_j} \mathbf{t}_I \delta \mathbf{u} dA = \int_{\gamma_j} \boldsymbol{\lambda}_j \cdot \psi \delta \mathbf{u} dA \quad (2.32)$$

The *Lagrange multipliers* are constants and we can factor them out of the integral. Then we change the integration limits $\Gamma_I^{(m,n)}$ to γ_j , so the weight function ψ can be omitted because it is always 1 in the current γ_j .

$$\underbrace{\lambda_j \int_{\gamma_j} 1 \cdot \delta \mathbf{u} d\Gamma}_{c_j} \quad (2.33)$$

Finally we want to specify the integral to be a vector of coupling coefficients.

$$\mathbf{c}_j = \int_{\gamma_j} \delta \mathbf{u} \, d\Gamma \quad (2.34)$$

Conservation of mechanical energy at the interface fields: To follow the idea of the *Mortar Method* the mechanical energy has to be conserved at the interface. For this we use the *Principle of Virtual Work*. First we introduce a virtual force $\delta \mathbf{p}$ and then we state the equilibrium of virtual work for each interface field.

$$\partial W_j = \partial W_j^{(m)} - \partial W_j^{(n)} = 0 \quad (2.35)$$

The displacements on each side of the field are $\mathbf{u}^{(m)}$ and $\mathbf{u}^{(n)}$. The virtual work in γ_j derived from the subdomain $\Omega^{(m)}$ is

$$\partial W_j^{(m)} = \int_{\gamma_j} \delta \mathbf{p}^{(m)} \, d\Gamma \cdot \mathbf{u}^{(m)} \quad (2.36)$$

and the virtual work derived from subdomain $\Omega^{(n)}$ is

$$\partial W_j^{(n)} = \int_{\gamma_j} \delta \mathbf{p}^{(n)} \, d\Gamma \cdot \mathbf{u}^{(n)} \quad (2.37)$$

Now the equilibrium of virtual work reads like

$$\int_{\gamma_j} \delta \mathbf{p}^{(m)} \, d\Gamma \cdot \mathbf{u}^{(m)} - \int_{\gamma_j} \delta \mathbf{p}^{(n)} \, d\Gamma \cdot \mathbf{u}^{(n)} = 0 \quad (2.38)$$

Further we define the test functions for the virtual displacement $\delta \mathbf{u}$ and the virtual traction $\delta \mathbf{p}$ to be the same. Now we can identify:

$$\mathbf{c}_j^{(m)} = \int_{\gamma_j} \delta \mathbf{p}^{(m)} \, d\Gamma = \int_{\gamma_j} \delta \mathbf{u}^{(m)} \, d\Gamma \quad (2.39)$$

which is the same vector of coefficients like in equation (2.34). Finally we write the equation like

$$\mathbf{c}_j^{(m)} \cdot \mathbf{u}^{(m)} - \mathbf{c}_j^{(n)} \cdot \mathbf{u}^{(n)} = 0 \quad (2.40)$$

System equations: Now we put all things together and get the following system equations

$$\int_{\Omega^{(m)}} \mathbf{T} : \varepsilon(\delta \mathbf{u}) \, d\Omega - \boldsymbol{\lambda}_j \cdot \mathbf{c}_j^{(m)} = \int_{\Omega^{(m)}} \mathbf{b} \cdot \delta \mathbf{u} \, d\Omega + \int_{\Gamma_N^{(m)}} \mathbf{t}_N \cdot \delta \mathbf{u} \, d\Gamma \quad (2.41)$$

$$\int_{\Omega^{(n)}} \mathbf{T} : \varepsilon(\delta \mathbf{u}) \, d\Omega - \boldsymbol{\lambda}_j \cdot \mathbf{c}_j^{(n)} = \int_{\Omega^{(n)}} \mathbf{b} \cdot \delta \mathbf{u} \, d\Omega + \int_{\Gamma_N^{(n)}} \mathbf{t}_N \cdot \delta \mathbf{u} \, d\Gamma \quad (2.42)$$

$$\mathbf{c}_j^{(m)} \cdot \mathbf{u}^{(m)} - \mathbf{c}_j^{(n)} \cdot \mathbf{u}^{(n)} = 0 \quad (2.43)$$

$$(2.44)$$

- Equation (2.41): displacement field in region $\Omega^{(m)}$
- Equation (2.42): displacement field in region $\Omega^{(n)}$
- Equation (2.43): coupling condition for regions $\Omega^{(m)}$, $\Omega^{(n)}$

2.3 Numerical approximation

For the numerical approximation we can use the *Finite Element Method* or the *Boundary Element Method*. The following explanations belong to the *Finite Element Method*. For this coupling method the type of discretisation doesn't matter. We choose a parametrisation such that the nodal test functions are defined among the limits of an intrinsic coordinate space of an element. We also make use of the Gauss quadrature rules for the numerical approximation of the integral expressions. Then we can compute all the terms from the section above and we do this node-wise.

Symbol names:

- $\boldsymbol{\xi} = \{\xi_n\} \quad \forall n = 1, 2, 3$, coordinates in parameter space (intrinsic)
- $N_i(\boldsymbol{\xi})$, test function at node i
- \mathbf{D} , coefficient matrix of the linear strain tensor
- $\mathbf{B}(N_i(\boldsymbol{\xi}_g))$, differential operator matrix applied to the test function at node i
- $J(\boldsymbol{\xi})$, determinant of the Jacobi matrix
- ω_g , Gauss quadrature weight
- $\mathbf{t}_N(\boldsymbol{\xi})$, traction at the Neumann boundary
- n_g , number of Gauss points on one element

Stiffness matrix:

$$k_{ij} = \int_{\Omega} \mathbf{T} : \varepsilon(\delta \mathbf{u}) d\Omega \Rightarrow$$

$$k_{ij} = \sum_{g=1}^{n_g} \mathbf{B}(N_j(\boldsymbol{\xi}_g))^\top \cdot \mathbf{D} \cdot \mathbf{B}(N_i(\boldsymbol{\xi}_g)) \cdot \omega_g \cdot J(\boldsymbol{\xi}_g)$$

where k_{ij} is the corresponding coefficient at node i due to a unit displacement at node j .

Applied loading: We derive one part from the body forces applied on the element volumes

$$\mathbf{f}_i = \int_{\Omega} \mathbf{b} \cdot \delta \mathbf{u} d\Omega \Rightarrow$$

$$\mathbf{f}_i = \sum_{g=1}^{n_g} N_i(\boldsymbol{\xi}_g) \cdot \mathbf{b}(\boldsymbol{\xi}_g) \cdot J(\boldsymbol{\xi}_g)$$

and another part from the traction applied on the element surfaces

$$\mathbf{f}_i = \int_{\Gamma^\sigma} \mathbf{t} \cdot \delta \mathbf{u} d\Gamma \Rightarrow$$

$$\mathbf{f}_i = \sum_{g=1}^{n_g} N_i(\boldsymbol{\xi}_g) \cdot \mathbf{f}_N(\boldsymbol{\xi}_g) \cdot J(\boldsymbol{\xi}_g)$$

Both have to be summed up for each node i .

Coupling coefficients: Since this is the main proposal of this work, the derivations of the coupling coefficients will be explained in more detail in the next section.

2.4 Coupling coefficients

In this section we will derive a numerical approximation of the coupling condition.

2.4.1 Coefficients of the coupling condition

The displacement approach $\delta \mathbf{u}$ is now replaced by the element shape functions $N_i(\boldsymbol{\xi})$. The integral formulation for the coupling coefficients for each *Lagrange multiplier field* γ_j and element shape function i is

$$\mathbf{c}_{i,j} = \int_{\gamma_j} N_i(\boldsymbol{\xi}) d\Gamma \quad (2.45)$$

To compute the coefficients $\mathbf{c}_{i,j}$ the following tasks have to be performed:

- Set up the *side-elements* of the adjacent sub-domains.
- Choose an admissible test function $N_i(\boldsymbol{\xi})$ for all nodes of the *side-elements*.
- Choose shape and size of the *Lagrange multiplier fields*.
- Carry out the integral for every *side-element's* shape functions among the limits of the *Lagrange multiplier fields*.

Side-elements: A side-element is a subspace of the boundary of an element E that is situated on the interface Γ_I .

$$S_i \subset \partial E \quad \text{and} \quad S_i \subset \Gamma_I$$

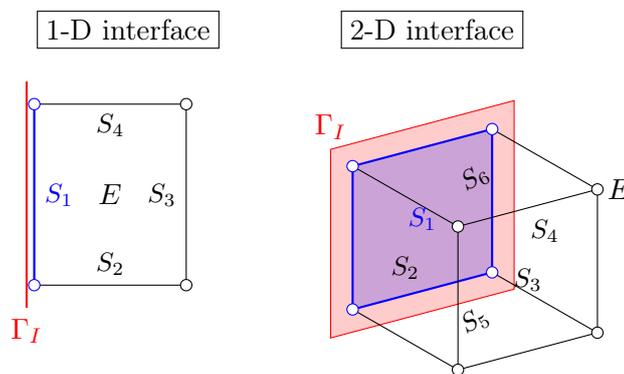


Figure 2.3: Element boundary, side elements

The very same applies to triangles and tetrahedrons.

Test functions of the side-elements: We need to choose test functions for each node of the side-elements. This can be an arbitrary function. But it would make sense to have them dependent on the test functions of their parent elements. For instance if an element test function is a linear polynomial then we should choose a linear polynomial for the side-element test function as well. If we restrict the test functions to polynomials we can state:

- A side-element test function is of same order like the element test function.

$$\mathcal{O}(N_i) = \mathcal{O}(\hat{N}_i)$$

- A side-element test function lives in a parameter space that is one dimension lower than the parameter space of the test function of the element.

$$N_i(\xi_1, \xi_2, \xi_3) \mapsto \hat{N}_i(\xi_1, \xi_2) \quad \dots \text{2-D interface}$$

$$N_i(\xi_1, \xi_2) \mapsto \hat{N}_i(\xi) \quad \dots \text{1-D interface}$$

Here an example for linear test functions $\hat{N}_{1,3}$, $\hat{N}_{3,1}$ of a 1-dimensional side-element S_1 .

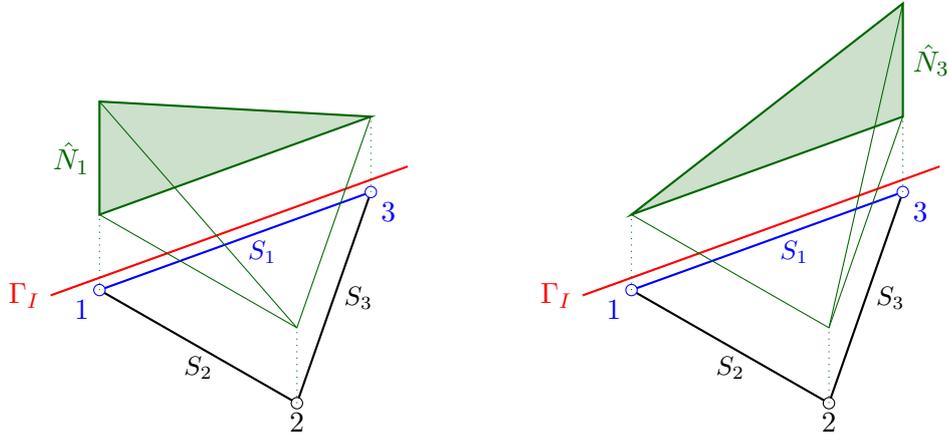


Figure 2.4: Linear test functions, 1-D side element

The choice of *Lagrange multiplier fields* and the integration needs more detailed explanation. Therefore they got their own sections. See next both sections.

2.5 Lagrange multiplier fields

In general there is no constraint on the choice of size and shape of the *Lagrange multiplier fields* γ_j at the interface Γ_I . But for convenience it would make sense to have some restrictions.

Field size: If we choose fields γ_j with a characteristic size h_{max} and then let $h_{max} \rightarrow 0$ we can see that the energy function in it's weak form will derive the strong form (point-wise). So it is a good choice to use small fields. The other point is that every field has a number of degree's of freedom in it and this will lead to a higher number of unknowns and to a larger equation system at the end.

Field shape: Since there are no constraints, the shape can be of arbitrary choice. But let us bear in mind that the shape boundaries are the integration limits. For this the fields should have a shape geometry that is as simple as possible (triangles, quadrilaterals). There are some possible choices we can find in the literature.

- Arbitrary choice of size and shape (rare)
- *Master-Slave* concept (most simple, but powerful)
- Zero moment rule (yet 1D interfaces only)

Arbitrary choice of size and shape: This approach is most independent of the mesh geometry of the sub-domains. But we have to set up a number of fields that are a subspace of both sub-domains and this could be a very difficult task if we think of curved surfaces. The main advantage is that we can completely control the number of degree's of freedom by the number of fields we choose. The disadvantage is that the integration over that fields is then for both sub-domains no more easy as we will see later.

Master-Slave concept: This is the most powerful and easy concept for a choice of field size and shape. The main idea is that we choose the *Lagrange multiplier fields* as the side-elements of one of both sub-domains. This subdomain is then called the *Master* and the opposite subdomain the *Slave*. The limitation of this concept is that we can not control the number of degree's of freedom any more. Their number is now limited to the number of side-elements of the *Master* subdomain. We also cannot choose the size of the fields freely. But we can choose between the side-elements of two sub-domains. If we choose the subdomain with the coarser mesh to be the *Master* we will get less fields of bigger size and less degree's of freedom. If we choose the subdomain with the finer mesh to be the *Master* we will get more fields of smaller size and more degree's of freedom. An additional advantage of that choice is that the integration of the *Master* side is very easy since we have to integrate the side-element test functions over the side-element parameter space itself. For this Master's Thesis this is the concept of choice.

Zero moment rule: This concept was published by *K.C. Park, C.A. Felippa, G. Rebel*[9]. It was defined for 1D interfaces, thus this applies to plane problems only. In all the concepts above the integral of a test function is mapped to a constant function at the interface field. They do not care about the distribution of the test function values along the interface field. This concept helps to make a choice of field size and location such that the integral of the momenta of the test function values becomes zero. This concept is mentioned here only for completeness since it is not subject to this Master's Thesis.

2.6 Integration

The integration limits are the boundaries of the *Lagrange multiplier fields* γ_j . The integral function (2.45) is the test function $N_i(\boldsymbol{\xi})$ of the node i defined over the parameter space $\boldsymbol{\xi}$ of a side-element S . The integration limits do not live in the same space like the test functions do. This and only this is the crucial point of the *Mortar Method*! We will have to transform the integration limits into the parameter space of the side-element first. And this is no easy task at all! The next two subsections will try to explain the geometrical problem for 1D and 2D interfaces and how the computation of the integrals can be carried out. For simplicity linear test functions are used for the the examples. This also applies to test functions of arbitrary order.

2.6.1 Integration for 1D interfaces

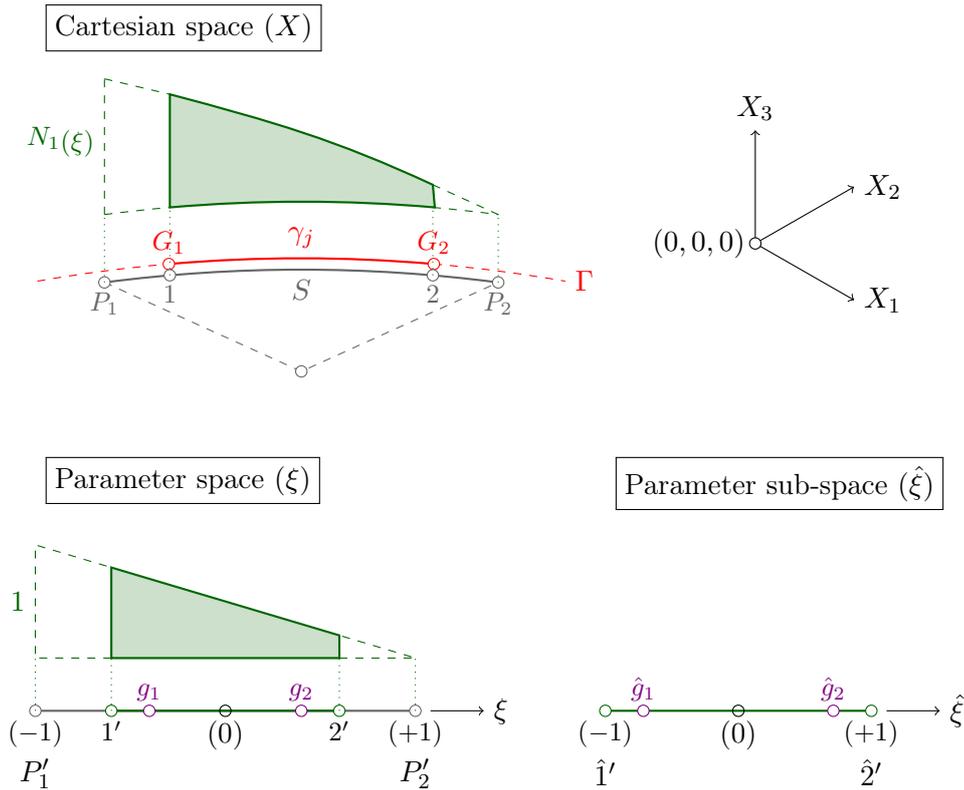


Figure 2.5: Integration, 1-D interfaces

Given:

- Side-element $S := (P_1, P_2)$
- *Lagrange multiplier field* $\gamma_j := (G_1, G_2)$
- Parameter space ξ of the side-element
- Parameter subspace $\hat{\xi}$ for line elements
- Shape functions $N_i(\xi)$ for side-elements

Integral computation, step by step:

1. Project the *Lagrange multiplier field* boundary points $[G_1, G_2]$ onto the element and get the points $[1, 2]$.
2. Transform the projected points $[1, 2]$ into the parameter space ξ of the side-element. The result is $[1', 2']$.
3. Set up a line element for the segment $[1', 2']$ using a parameter subspace $\hat{\xi}$.
4. Compute the Gauss quadrature node coordinates $\hat{\xi}_g$ and weights $\hat{\omega}_g$ in parameter subspace $\hat{\xi}$. Compute the Jacobian $\hat{J}(\hat{\xi}_g)$.
5. Transform the Gauss nodes $\hat{\xi}_g$ into parameter space ξ (result: ξ_g).
6. Compute the test function values $N_i(\xi_g)$ and the Jacobian $J(\xi_g)$.
7. Sum up over all Gauss nodes ξ_g .

Now we get the following integration formula

$$\int_{\gamma_j} N_i(\boldsymbol{\xi}) dA = \sum_{g=1}^{n_g} \hat{\omega}_g \cdot \hat{J}(\hat{\boldsymbol{\xi}}_g) \cdot N_i(\boldsymbol{\xi}_g) \cdot J(\boldsymbol{\xi}_g) \quad (2.46)$$

where n_g is the number of Gauss points.

Other geometry configurations: The picture above shows case 1) of a possible interface configuration. Here are the other two possible cases and how they have to be handled.

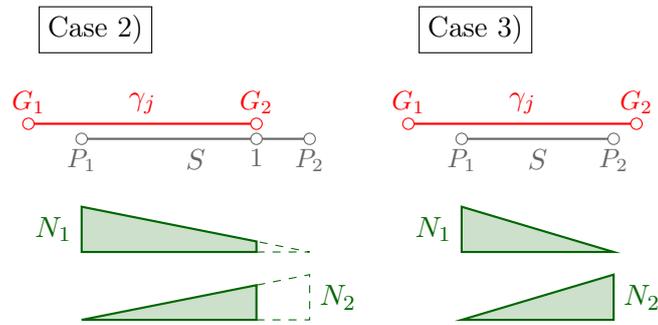


Figure 2.6: Geometry configurations, 1-D interfaces

2.6.2 Integration for 2D interfaces

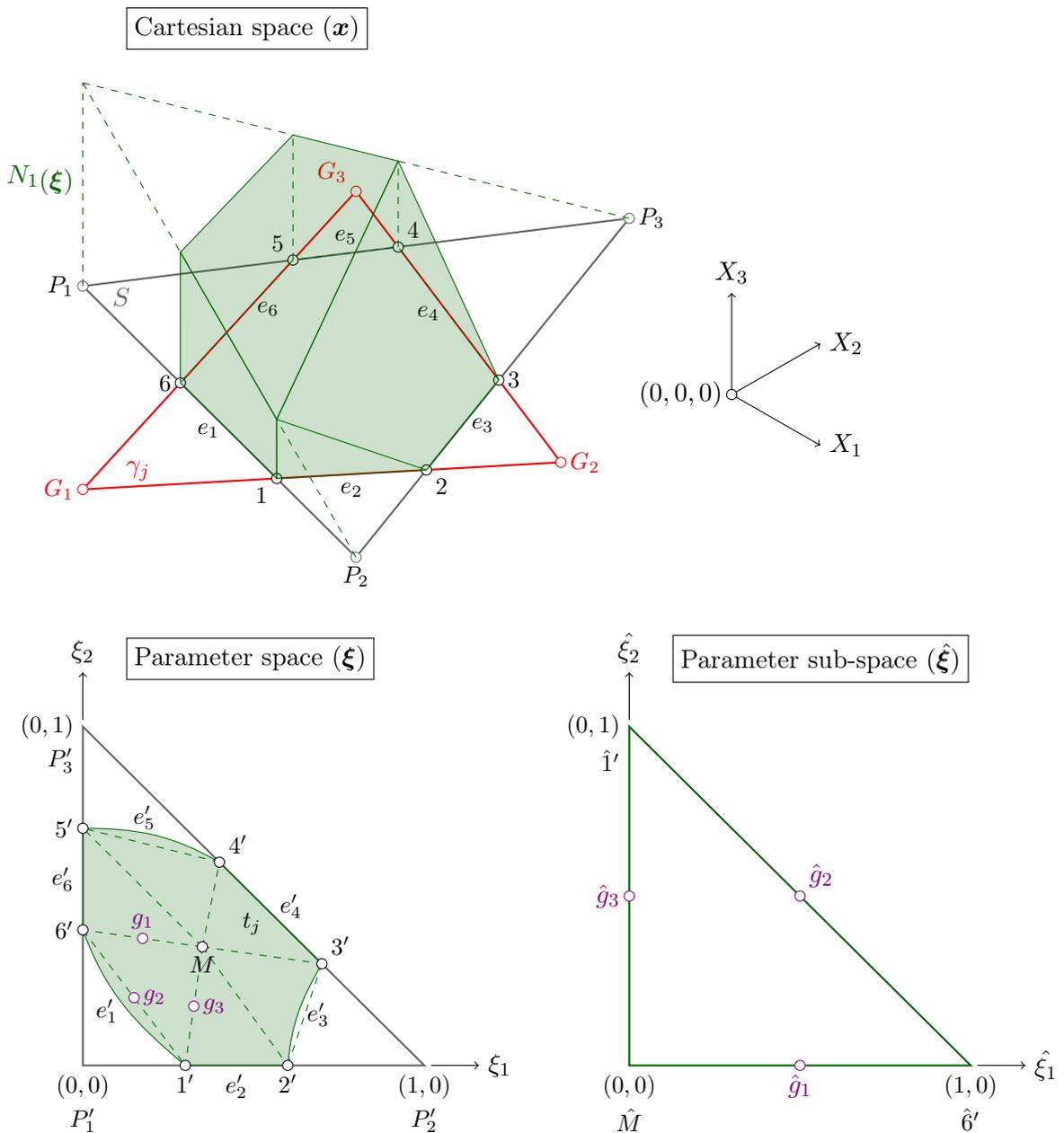


Figure 2.7: Integration, 2-D interfaces

Given:

- Side-element $S := (P_1, P_2, P_3)$
- Lagrange multiplier field $\gamma_j := (G_1, G_2, G_3)$
- Parameter space ξ of the side-element
- Parameter subspace $\hat{\xi}$ for triangles
- Shape functions $N_i(\xi)$ for side-elements

Integral computation, step by step:

1. Compute the intersection points $[1, \dots, 6]$ of the side-element and the Lagrange multiplier field γ_j .

2. Transform the intersection points $[1, \dots, 6]$ into the parameter space $\boldsymbol{\xi}$ of the side-element (result: $[1', \dots, 6']$). Also transform corner points of side-element S who are laying inside the field γ_j into this space.
3. Set up a triangulation T consisting of the intersection points $[1', \dots, 6']$ and their midpoint M . If there are transformed corner points they are also part of the triangulation.
4. Set up triangular elements t_l using a parameter subspace $\hat{\boldsymbol{\xi}}$ for all triangles in the triangulation T .
5. Compute the Gauss quadrature node coordinates $\hat{\boldsymbol{\xi}}_g$ and weights $\hat{\omega}_g$ in parameter subspace $\hat{\boldsymbol{\xi}}$. Compute the Jacobian $\hat{J}(\hat{\boldsymbol{\xi}}_g)$.
6. Transform the Gauss nodes $\hat{\boldsymbol{\xi}}_g$ into parameter space (result: $\boldsymbol{\xi}_g$).
7. Compute the test function values $N_i(\boldsymbol{\xi}_g)$ and the Jacobian $J(\boldsymbol{\xi}_g)$.
8. Sum up over all Gauss nodes $\boldsymbol{\xi}_g$ of one triangle.
9. Sum up over all triangles t_l of the triangulation T .

Now we get the following integration formula

$$\int_{\gamma_j} N_i(\boldsymbol{\xi}) dA = \sum_{l=1}^{n_l} \left(\sum_{g=1}^{n_g} \hat{\omega}_g \cdot \hat{J}(\hat{\boldsymbol{\xi}}_g) \cdot N_i(\boldsymbol{\xi}_g) \cdot J(\boldsymbol{\xi}_g) \right)_l \quad (2.47)$$

where n_l is the total number of triangles and n_g is the total number of Gauss nodes.

Note: For arbitrary *Lagrange multiplier field* and side-element geometries we can state:

- In general the edges e'_1 , e'_3 and e'_5 are no more straight lines. It would be good to compute some interpolation points and generate a finer triangulation to get more accurate results for the integral. For high-order elements this would be necessary. For linear elements there will be no big difference, thus the computation of that interpolation points can be omitted.
- The computation of the intersection points $[1, \dots, 6]$ is no easy task since we can compute them analytically only for certain cases. Just think of the rounding errors of node coordinates. This will lead to non-intersecting geometries in 3-dimensional space. So we will have to find a numerical solution that can handle such errors and detects intersection points within a given tolerance.
- The transformation of the intersection points from Cartesian coordinate space into intrinsic parameter space $[1', \dots, 6']$ is also no easy task since the inverse process will lead to a manifold of results. For this we also need to provide a numerical solution.

Other geometry configurations: The picture above shows for case 1) of intersecting geometries only. Let us have a look at other possible configurations and how they have to be treated.

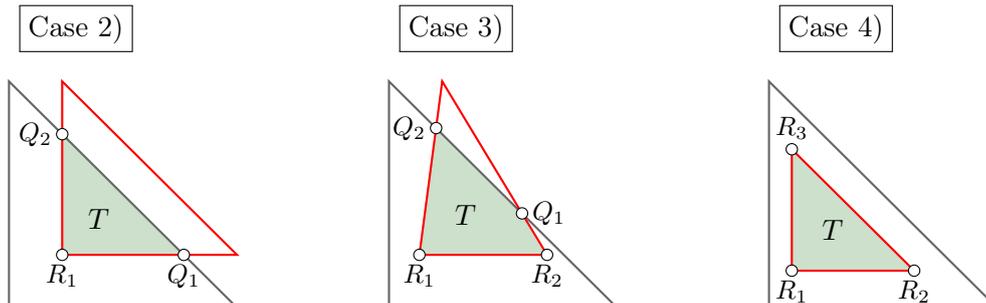


Figure 2.8: Geometry configurations, 2-D interfaces

Q_i are the intersection points and R_i are the *inside-points*. We can see that the triangulation polygon T consists of two classes of points, the intersection points Q_i and the *inside-points* R_i . All configurations share one property: If the sum of intersection points and inside-points is at least three and they are not aligned to a straight line, there exists a triangulation T . Therefore the integral exists. There are much more possible configurations for all kind of elements but the computation of the integral works for all of them like the same.

2.7 Global equation system

Finally we take a look at the global equation system of the coupled problem. The general form of this equation system is:

$$[A] \cdot \{x\} = \{b\} \tag{2.48}$$

$$\begin{bmatrix} [\mathbf{K}^1] & [\mathbf{0}] & \dots & \dots & \dots & [\mathbf{0}] & [\mathbf{D}^1]^\top & [\mathbf{C}^1]^\top & [\mathbf{0}] \\ [\mathbf{0}] & [\mathbf{K}^2] & \ddots & & & \vdots & [\mathbf{D}^2]^\top & [\mathbf{C}^2]^\top & [\mathbf{0}] \\ \vdots & \ddots & \ddots & \ddots & & \vdots & \vdots & [\mathbf{0}] & [\mathbf{0}] \\ \vdots & & \ddots & [\mathbf{K}^m] & \ddots & \vdots & [\mathbf{D}^m]^\top & [\mathbf{0}] & [\mathbf{C}^m]^\top \\ \vdots & & & \ddots & \ddots & [\mathbf{0}] & \vdots & [\mathbf{0}] & [\mathbf{0}] \\ [\mathbf{0}] & \dots & \dots & \dots & [\mathbf{0}] & [\mathbf{K}^n] & [\mathbf{D}^n]^\top & [\mathbf{0}] & [\mathbf{C}^n]^\top \\ [\mathbf{D}^1] & [\mathbf{D}^2] & \dots & [\mathbf{D}^m] & \dots & [\mathbf{D}^n] & [\mathbf{0}] & \dots & [\mathbf{0}] \\ [\mathbf{C}^1] & [\mathbf{C}^2] & [\mathbf{0}] & [\mathbf{0}] & [\mathbf{0}] & [\mathbf{0}] & \vdots & \ddots & \vdots \\ [\mathbf{0}] & [\mathbf{0}] & [\mathbf{0}] & [\mathbf{C}^m] & [\mathbf{0}] & [\mathbf{C}^n] & [\mathbf{0}] & \dots & [\mathbf{0}] \end{bmatrix} \cdot \begin{Bmatrix} \{\mathbf{u}^1\} \\ \{\mathbf{u}^2\} \\ \vdots \\ \{\mathbf{u}^m\} \\ \vdots \\ \{\mathbf{u}^n\} \\ \{\boldsymbol{\lambda}_D\} \\ \{\boldsymbol{\lambda}^{1,2}\} \\ \{\boldsymbol{\lambda}^{m,n}\} \end{Bmatrix} = \begin{Bmatrix} \{\mathbf{f}_0^1\} \\ \{\mathbf{f}_0^2\} \\ \vdots \\ \{\mathbf{f}_0^m\} \\ \vdots \\ \{\mathbf{f}_0^n\} \\ \{\mathbf{u}_0\} \\ \{\mathbf{0}\} \\ \{\mathbf{0}\} \end{Bmatrix}$$

Symbol names:

- $[\mathbf{K}^n]$, stiffness matrix of a subdomain
- $[\mathbf{D}^n]$, identity matrix of Dirichlet boundary conditions
- $[\mathbf{C}^n]$, *Mortar Method* coupling matrix
- $\{\mathbf{u}^n\}$, displacement vector of a subdomain
- $\{\mathbf{f}_0^n\}$, force vector of a subdomain
- $\{\boldsymbol{\lambda}_D\}$, Dirichlet degree of freedom vector
- $\{\mathbf{u}_0\}$, Dirichlet displacement vector
- $\{\boldsymbol{\lambda}^{mn}\}$, vector of *Lagrange multipliers*

Chapter 3

A simple example

In this chapter we will compute a simple example. The goal is to see how the coupling matrices are computed and the equation system has to be assembled. Further we will get the solution and some interpretation on what has happened at the interface.

3.1 Configuration

Let us assume to have a very simple system configuration consisting of three finite elements, one Dirichlet boundary, one Neumann boundary with the following properties.

- Young's modulus, $E = 10000 [kN/m^2]$
- Poisson's ration, $\nu = 0.2 []$
- Element thickness, $t = 1 [m]$
- Subdomain $\Omega^{(1)}$ element dimensions, $L = 1 [m], H = 1 [m]$
- Subdomain $\Omega^{(2)}$ element dimensions, $L = 1 [m], H = 0.5 [m]$
- Dirichlet boundary conditions: $u_x = 0 [m], u_y = 0 [m]$ at node 1 and 3 of subdomain $\Omega^{(1)}$
- Neumann boundary conditions: $p_x = 100 [kN/m], p_y = -100 [kN/m]$ at the boundary between node 6, 8 and 10 in subdomain $\Omega^{(2)}$
- Plain strain

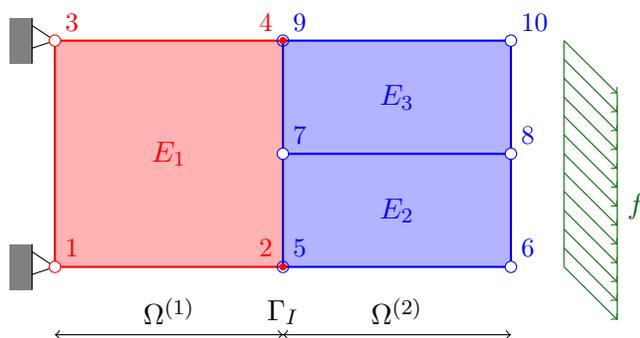


Figure 3.1: Simple cantilever, 2 regions

3.2 Stiffness matrices

The stiffness matrices of both subdomains can be computed easily. Their coefficients are not the major topic of this example, therefore they are noted here only in a general form. For convenience

the stiffness matrix is written in terms of 2×2 block matrices.

$$[k_{i,j}] = \begin{bmatrix} k_{ix,jx} & k_{ix,jy} \\ k_{iy,jx} & k_{iy,jy} \end{bmatrix}$$

where $k_{i,j}$ are the corresponding forces in x and y direction at node i due to virtual displacements in x - and y -direction at node j . For example: $k_{ix,jy}$ is the corresponding force of element node i in x -direction due to the virtual displacement in y -direction applied at node j .

Stiffness matrix, subdomain $\Omega^{(1)}$: The stiffness matrix $[K^{(1)}]$ is then a 8×8 block matrix.

$$[K^{(1)}] = \begin{bmatrix} [k_{1,1}] & \cdots & [k_{1,4}] \\ \vdots & \ddots & \vdots \\ [k_{4,1}] & \cdots & [k_{4,4}] \end{bmatrix}$$

Stiffness matrix, subdomain $\Omega^{(2)}$: Since the elements E_2 and E_3 are both using the nodes 7 and 8 we have to assemble the element stiffness matrices. The stiffness matrix $[K^{(2)}]$ is then a 12×12 block matrix.

$$[K^{(2)}] = \begin{bmatrix} [k_{5,5}] & \cdots & \cdots & [k_{5,8}] & [0] & [0] \\ \vdots & \ddots & \ddots & \vdots & [0] & [0] \\ \vdots & \ddots & [k_{7,7}] & [k_{7,8}] & \cdots & [k_{7,10}] \\ [k_{8,5}] & \cdots & [k_{8,7}] & [k_{8,8}] & \ddots & \vdots \\ [0] & [0] & \vdots & \ddots & \ddots & \vdots \\ [0] & [0] & [k_{10,7}] & \cdots & \cdots & [k_{10,10}] \end{bmatrix}$$

3.3 Dirichlet boundary condition identity matrices

For this example we want to couple-in the Dirichlet boundary conditions instead of modifying the stiffness matrix of subdomain $\Omega^{(1)}$. The boundary conditions apply to node 1 and node 3 where both degree's of freedom (u_x and u_y) are locked. The block matrices $[d_{1,1}]$ and $[d_{3,3}]$ for node 1 and 3 are 2×2 identity matrices. All other block matrices are 2×2 zero matrices.

The identity matrix for subdomain $\Omega^{(1)}$ is then

$$[D^{(1)}] = \begin{bmatrix} [d_{1,1}] & [d_{1,2}] & [d_{1,3}] & [d_{1,4}] \\ [d_{3,1}] & [d_{3,2}] & [d_{3,3}] & [d_{3,4}] \end{bmatrix} = \left[\begin{array}{cc|cc|cc|cc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{array} \right]$$

3.4 Mortar Method coupling matrices

For the computation of the coupling matrices $[C^{(1)}]$ and $[C^{(2)}]$ we choose the subdomain $\Omega^{(2)}$ to be the *Master* side and the subdomain $\Omega^{(1)}$ is the *Slave* side. First we define the *Lagrange multiplier fields* and then we compute the matrix coefficients.

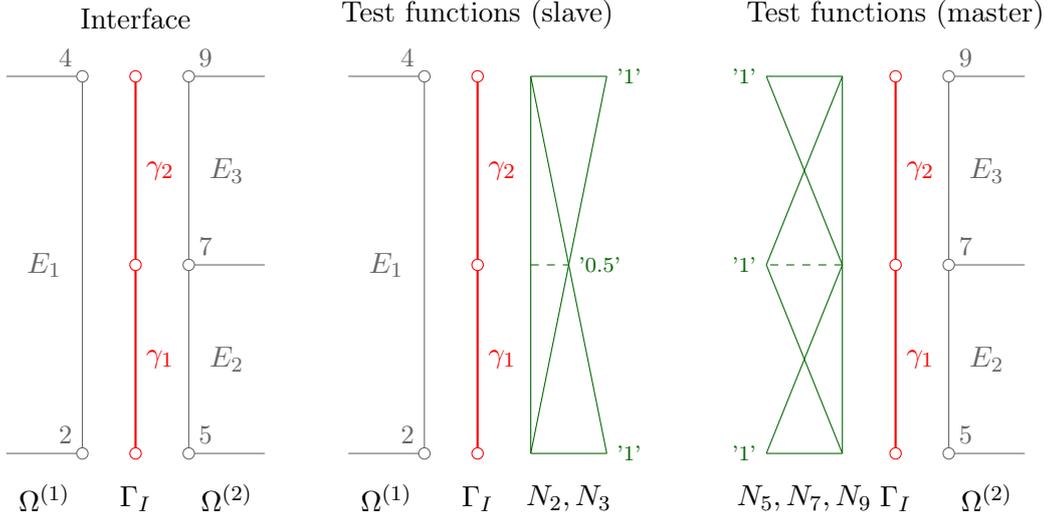


Figure 3.2: *Lagrange multiplier fields*

Here, γ_1 and γ_2 are the *Lagrange multiplier fields*. In each field are living two degree's of freedom, one for the global x direction and one for the global y direction. For the coupling coefficients we choose the following notation:

- coefficient name: c
- first subscript: node number
- second subscript: *Lagrange multiplier field* number

The value of the coupling coefficient $c_{i,j}$ is the integral of the shape function N_i over γ_j .

$$c_{i,j} = \int_{\gamma_j} N_i ds$$

where ds is a infinitesimal small part of the curve length of γ_j . The coupling matrices will be represented by 2×2 block matrices like we did for the stiffness matrices before.

$$[c_{i,j}] = c_{i,j} \cdot [I] = \begin{bmatrix} c_{i,j} & 0 \\ 0 & c_{i,j} \end{bmatrix}$$

where $[I]$ is a 2×2 identity matrix.

Coupling coefficients and matrix, *Slave* side: The coupling coefficients are used for the degree's of freedom in x and in y direction like the same. Therefore they appear twice in the coupling matrix. As a convention all coupling coefficients on the *Slave* side get a minus sign.

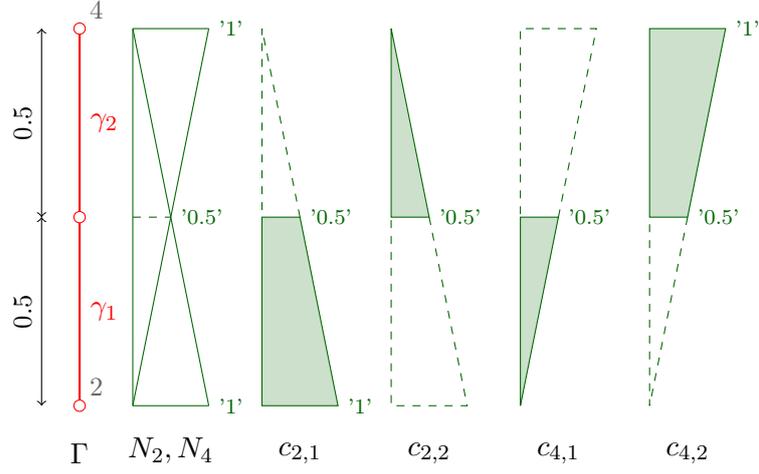


Figure 3.3: Integration of *Slave* coupling coefficients

$$c_{2,1} = -(1 + 0.5)/2 \cdot 0.5 = -0.375$$

$$c_{2,2} = -0.5/2 \cdot 0.5 = -0.125$$

$$c_{4,1} = -0.5/2 \cdot 0.5 = -0.125$$

$$c_{4,2} = -(1 + 0.5)/2 \cdot 0.5 = -0.375$$

The coupling matrices for node 1 and 3 are 2×2 zero matrices. The coupling matrix $[C^{(1)}]$ is

$$[C^{(1)}] = \begin{bmatrix} [0] & [c_{2,1}] & [0] & [c_{4,1}] \\ [0] & [c_{2,2}] & [0] & [c_{4,2}] \end{bmatrix} = \begin{bmatrix} 0 & 0 & -0.375 & 0 & 0 & 0 & -0.125 & 0 \\ 0 & 0 & 0 & -0.375 & 0 & 0 & 0 & -0.125 \\ 0 & 0 & -0.125 & 0 & 0 & 0 & -0.375 & 0 \\ 0 & 0 & 0 & -0.125 & 0 & 0 & 0 & -0.375 \end{bmatrix}$$

Coupling coefficients and matrix, *Master*: Again the coupling coefficients are used for both, the degree's of freedom in x and in y direction.

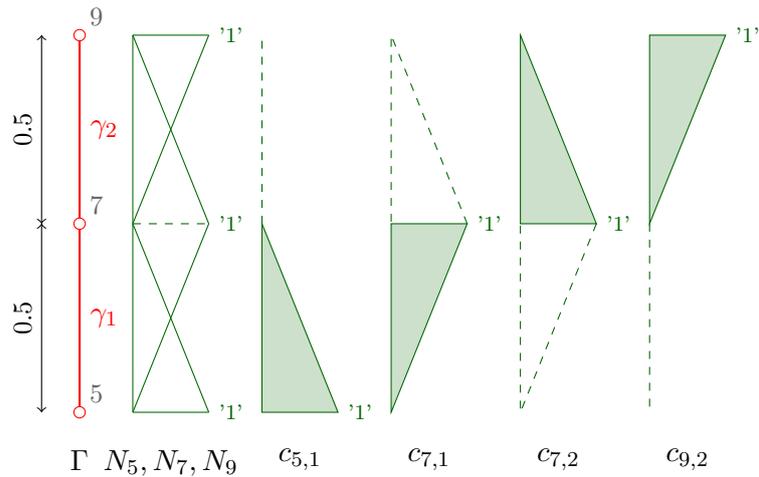


Figure 3.4: Integration of *Master* coupling coefficients

$$c_{5,1} = 1/2 \cdot 0.5 = 0.25$$

$$c_{5,2} = 0 \quad \dots \quad N_5 \text{ is zero in } \gamma_2$$

$$c_{7,1} = 1/2 \cdot 0.5 = 0.25$$

$$c_{7,2} = 1/2 \cdot 0.5 = 0.25$$

$$c_{9,1} = 0 \quad \dots \quad N_9 \text{ is zero in } \gamma_1$$

$$c_{9,2} = 1/2 \cdot 0.5 = 0.25$$

The coupling matrices for node 6, 8 and 10 are 2×2 zero matrices. The coupling matrix $[C^{(2)}]$ is

$$[C^{(2)}] = \begin{bmatrix} [c_{5,1}] & [0] & [c_{7,1}] & [0] & [c_{9,1}] & [0] \\ [c_{5,2}] & [0] & [c_{7,2}] & [0] & [c_{9,2}] & [0] \end{bmatrix}$$

$$= \left[\begin{array}{cc|cc|cc|cc|cc} 0.25 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.25 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0.25 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0.25 \end{array} \right]$$

3.5 Load vector (right hand side)

The load vector of the subdomain $\Omega^{(1)}$ is

$$\{f_0^{(1)}\}^\top = \{f_{1,x}, f_{1,y}, f_{2,x}, f_{2,y}, f_{3,x}, f_{3,y}, f_{4,x}, f_{4,y}\}$$

$$\{f_0^{(1)}\}^\top = \{0, 0, 0, 0, 0, 0, 0, 0\}$$

For sub-domain $\Omega^{(2)}$ we integrate the weight functions to get the nodal weights for all nodes at the Neumann boundary.

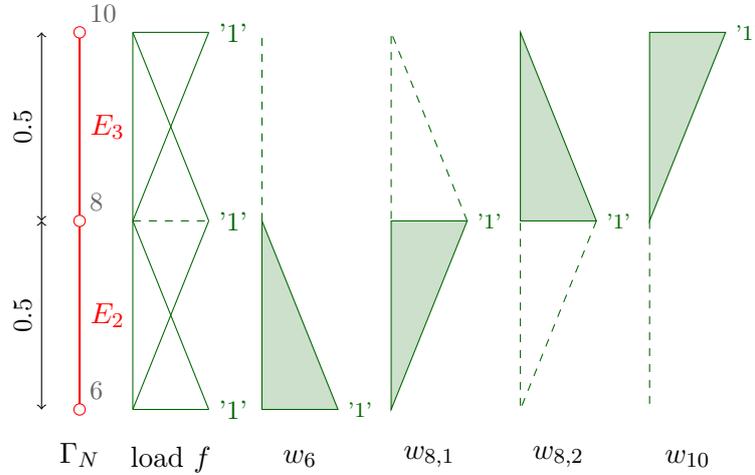


Figure 3.5: Integration of the Neumann boundary

The weight function integrals for all nodes at the Neumann boundary are

$$w_6 = 1/2 \cdot 0.5 \cdot 1 = 0.25$$

$$w_{8,1} = 1/2 \cdot 0.5 \cdot 1 = 0.25$$

$$w_{8,2} = 1/2 \cdot 0.5 \cdot 1 = 0.25$$

$$w_8 = w_{8,1} + w_{8,2} = 0.50$$

$$w_{10} = 1/2 \cdot 0.5 \cdot 1 = 0.25$$

The load distributions along the Neumann boundary are

$$f_x = 100.0 [kN/m]; f_y = -100.0 [kN/m]$$

The nodal force components can now be derived

$$f_{6,x} = w_6 \cdot 100.0 = 25.0$$

$$f_{6,y} = w_6 \cdot -100.0 = -25.0$$

$$f_{8,x} = w_8 \cdot 100.0 = 50.0$$

$$f_{8,y} = w_8 \cdot -100.0 = -50.0$$

$$f_{10,x} = w_{10} \cdot 100.0 = 25.0$$

$$f_{10,y} = w_{10} \cdot -100.0 = -25.0$$

Now the load vector of the sub-domain $\Omega^{(2)}$ is

$$\{f_0^{(2)}\}^\top = \{f_{5,x}, f_{5,y}, f_{6,x}, f_{6,y}, f_{7,x}, f_{7,y}, f_{8,x}, f_{8,y}, f_{9,x}, f_{9,y}, f_{10,x}, f_{10,y}\}$$

$$\{f_0^{(2)}\}^\top = \{0, 0, 25, -25, 0, 0, 50, -50, 0, 0, 25, -25\}$$

The displacement vector of the Dirichlet boundary conditions is

$$\{u_0\}^\top = \{u_{1x}, u_{1y}, u_{3x}, u_{3y}\} = \{0, 0, 0, 0\}$$

3.6 Degree's of freedom

The displacements for sub-domain $\Omega^{(1)}$ are

$$\{u^{(1)}\}^\top = \{u_{1,x}, u_{1,y}, \dots, u_{4,x}, u_{4,y}\}$$

and for sub-domain $\Omega^{(2)}$ they are

$$\{u^{(2)}\}^\top = \{u_{5,x}, u_{5,y}, \dots, u_{10,x}, u_{10,y}\}$$

The degree's of freedom of the Dirichlet boundary conditions are the reaction forces at the constraint nodes

$$\{\lambda_D\}^\top = \{\lambda_{d,1,x}, \lambda_{d,1,y}, \lambda_{d,3,x}, \lambda_{d,3,y}\}$$

The degree's of freedom of the coupling condition are the tractions situated at the *Lagrange multiplier fields*.

$$\{\lambda_C\}^\top = \{\lambda_{c,1,x}, \lambda_{c,1,y}, \lambda_{c,2,x}, \lambda_{c,2,y}\}$$

3.7 Global equation system

Now we are going to put all the things together and set up the global equation system.

$$\begin{bmatrix} [K^{(1)}] & [0] & [D^{(1)}]^\top & [C^{(1)}]^\top \\ [0] & [K^{(2)}] & [0] & [C^{(2)}]^\top \\ [D^{(1)}] & [0] & [0] & [0] \\ [C^{(1)}] & [C^{(2)}] & [0] & [0] \end{bmatrix} \cdot \begin{bmatrix} \{u^{(1)}\} \\ \{u^{(2)}\} \\ \{\lambda_D\} \\ \{\lambda_C\} \end{bmatrix} = \begin{bmatrix} \{f_0^{(1)}\} \\ \{f_0^{(2)}\} \\ \{u_0\} \\ \{0\} \end{bmatrix}$$

3.8 Solution and interpretation

Now we solve the equation system and get the displacements $\{u^{(1)}\}$, the displacements $\{u^{(2)}\}$, the reaction forces $\{\lambda_D\}$ and the tractions $\{\lambda_C\}$.

$\{u^{(1)}\}$	u_2	$u_{2,x}$	-0.00991006
		$u_{2,y}$	-0.0162948
	u_4	$u_{4,x}$	0.0136536
		$u_{4,y}$	-0.0168688

Table 3.1: Displacements $\{u^{(1)}\}$

$\{u^{(2)}\}$	u_5	$u_{5,x}$	-0.00991853
		$u_{5,y}$	-0.0164219
	u_6	$u_{6,x}$	-0.0120578
		$u_{6,y}$	-0.0489191
	u_7	$u_{7,x}$	0.00188022
		$u_{7,y}$	-0.0164548
	u_8	$u_{8,x}$	0.0038068
		$u_{8,y}$	-0.0488843
	u_9	$u_{9,x}$	0.0136451
		$u_{9,y}$	-0.0169959
u_{10}	$u_{10,x}$	0.0196376	
	$u_{10,y}$	-0.0493577	

Table 3.2: Displacements $\{u^{(2)}\}$

$\{\lambda_D\}$	$\lambda_{d,1}$	$\lambda_{d,1,x}$	-300.000000
		$\lambda_{d,1,y}$	-82.640545
	$\lambda_{d,3}$	$\lambda_{d,3,x}$	499.999999
		$\lambda_{d,3,y}$	-117.359455

Table 3.3: Reaction forces $\{\lambda_D\}$

$\{\lambda_C\}$	$\lambda_{c,1}$	$\lambda_{c,1,x}$	-600.000000
		$\lambda_{c,1,y}$	-210.971039
	$\lambda_{c,2}$	$\lambda_{c,2,x}$	1000.000000
		$\lambda_{c,2,y}$	-189.028961

Table 3.4: Tractions $\{\lambda_C\}$

The first thing we can see is that the displacements of the nodes 2 and 5 (and 4 and 9) are no more the same. This is for the reason that we did not use geometrical constraints for the displacements at the interface nodes. We chose an equilibrium of energy that does not control the motion of coinciding points. It only controls the tractions at the interface.

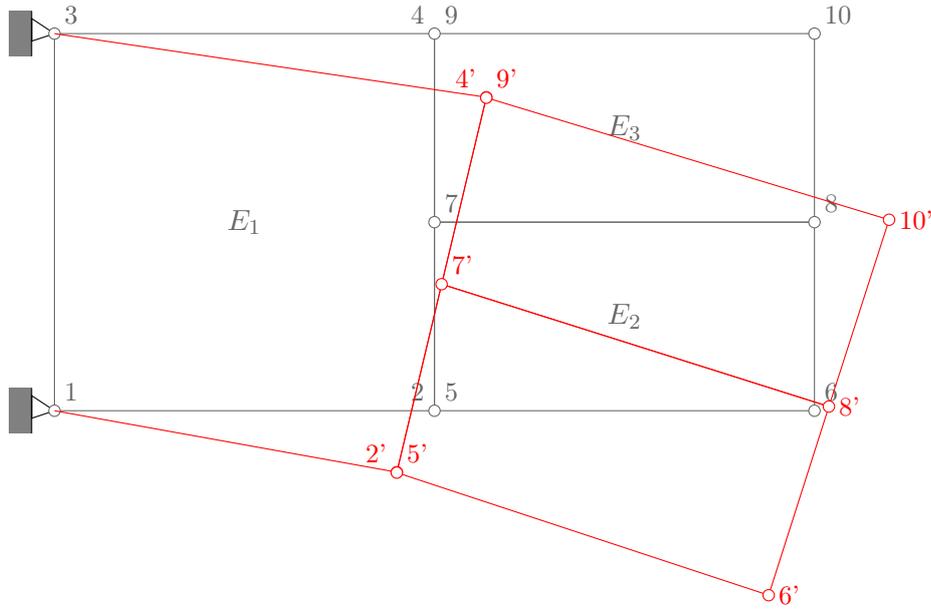


Figure 3.6: Displaced and origin system configuration

At this scale the differences of the displacements of the interface nodes seem to be very small. To show them better we zoom close to the interface nodes and look at the relative displacements Δu .

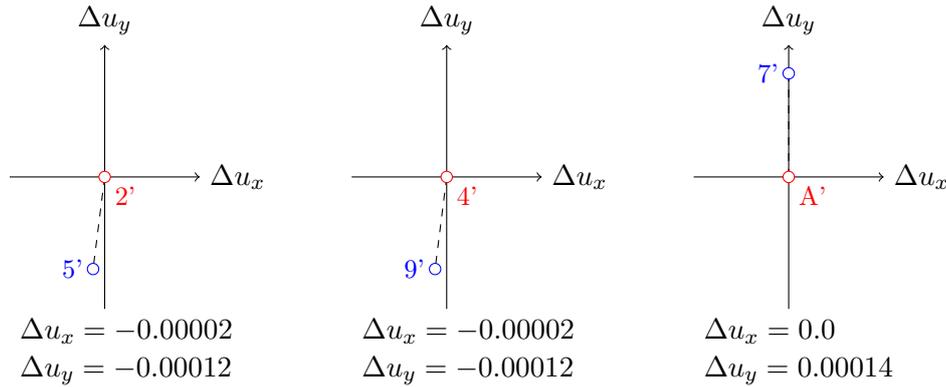


Figure 3.7: Relative displacements, interface nodes

Node A' is the displaced midpoint location of the line 2',4'. Here we can see that the node 5' and 9' have the same differences in displacements. This is due to the symmetry of the elements at the interface. Another interesting point is that the gap between the two sub-domains is rather small. The minimal displacements in x and y direction are:

$$u_{x,min} = u_{7,x} = 0.00188022$$

$$u_{y,min} = u_{2,y} = -0.0162948$$

If we compute the ratio between the disagreement of displacements (errors) and the absolute value of the minimal node displacements we get:

$$\frac{u_{x,min}}{\Delta u_{x,max}} \approx 10^2 \quad \text{and} \quad \frac{u_{y,min}}{\Delta u_{y,max}} \approx 10^2$$

For this example we can see that the displacement errors of the interface nodes are about 10^{-2} smaller than the nodal displacements itself.

3.8.1 Proofs

Finally we want to proof the equilibrium of work for all interface fields for the reason that this was the only boundary condition at the interface of the subdomains. For this we have to integrate the

product of the traction $\lambda_{c,j}$, the nodal displacement vector \mathbf{u}_i and the nodal shape functions $N_i^{(e)}(\boldsymbol{\xi})$ of the adjacent *Master* and *Slave* boundary elements among the limits of a *Lagrange multiplier field* γ_j . Since the tractions $\lambda_{c,j}$ are constant along each *Lagrange multiplier field* we can factor them out.

Integrate work for a *Master* boundary element: The *Master* boundary elements are identical to the *Lagrange multiplier fields*. So we have to integrate among the limits of the *Master* boundary element and to sum up for all element nodes n_i .

$$W_{j,Master} = \sum_{i=1}^{n_i} \left(\lambda_{c,j} \cdot \int_{\gamma_j} (N_i(\boldsymbol{\xi}) \cdot \mathbf{u}_i) d\Gamma \right) \quad (3.1)$$

Integrate work for all *Slave* boundary elements: Here we have a number of *Slave* boundary elements which share a common space with one *Lagrange multiplier field*. So we have to integrate among the limits of those common spaces (which are line segments or triangulations) and to sum up for all nodes n_i of all elements n_e .

$$W_{j,Slave} = \sum_{e=1}^{n_e} \left(\sum_{i=1}^{n_i} \left(\lambda_{c,j} \cdot \int_{\gamma_j} (N_i^{(e)}(\boldsymbol{\xi}) \cdot \mathbf{u}_i) d\Gamma \right) \right) \quad (3.2)$$

Equilibrium of work at one *Lagrange multiplier field* γ_j : The condition for the equilibrium of work to be met is

$$W_{j,Master} - W_{j,Slave} = 0 \quad (3.3)$$

***Lagrange multiplier field* γ_1 :** Integrals among the limits of the *Master* boundary element E_2 .

$$W_{5,x} = 1/2 \cdot u_{5,x} \cdot \lambda_{c,1,x} = -4.95926500$$

$$W_{5,y} = 1/2 \cdot u_{5,y} \cdot \lambda_{c,1,y} = 1.55210735$$

$$W_{7,x} = 1/2 \cdot u_{7,x} \cdot \lambda_{c,1,x} = 0.94011000$$

$$W_{7,y} = 1/2 \cdot u_{7,y} \cdot \lambda_{c,1,y} = 1.55521687$$

$$W_{1,Master} = W_{5,x} + W_{5,y} + W_{7,x} + W_{7,y} = -0.9118$$

$$W_{2,x} = (1 + 0.5)/2 \cdot u_{2,x} \cdot \lambda_{c,1,x} = -7.43254500$$

$$W_{2,y} = (1 + 0.5)/2 \cdot u_{2,y} \cdot \lambda_{c,1,y} = 2.31014184$$

$$W_{4,x} = 0.5/2 \cdot u_{4,x} \cdot \lambda_{c,1,x} = 3.41340000$$

$$W_{4,y} = 0.5/2 \cdot u_{4,y} \cdot \lambda_{c,1,y} = 0.79717293$$

$$W_{1,Slave} = W_{2,x} + W_{2,y} + W_{4,x} + W_{4,y} = -0.9118$$

Equilibrium of work for γ_1

$$W_{1,Master} - W_{1,Slave} = -0.9118 - (-0.9118) = 0.0000$$

The error ϵ_1 relative to the work of the *Master* side is 0.0%.

Lagrange multiplier field γ_2 : Integrals among the limits of the *Master* boundary element E_3 .

$$W_{9,x} = 1/2 \cdot u_{9,x} \cdot \lambda_{c,2,x} = -4.09355000$$

$$W_{9,y} = 1/2 \cdot u_{9,y} \cdot \lambda_{c,2,y} = 1.79280134$$

$$W_{7,x} = 1/2 \cdot u_{7,x} \cdot \lambda_{c,2,x} = -0.56408600$$

$$W_{7,y} = 1/2 \cdot u_{7,y} \cdot \lambda_{c,2,y} = 1.73572313$$

$$W_{2,Master} = W_{9,x} + W_{9,y} + W_{7,x} + W_{7,y} = -1.1291$$

Integrals among the limits of the common space between the *Slave* boundary element E_1 and the *Lagrange multiplier field γ_2* .

$$W_{2,x} = 0.5/2 \cdot u_{2,x} \cdot \lambda_{c,2,x} = 1.48650900$$

$$W_{2,y} = 0.5/2 \cdot u_{2,y} \cdot \lambda_{c,2,y} = 0.85943272$$

$$W_{4,x} = (1 + 0.5)/2 \cdot u_{4,x} \cdot \lambda_{c,2,x} = -6.14412000$$

$$W_{4,y} = (1 + 0.5)/2 \cdot u_{4,y} \cdot \lambda_{c,2,y} = 2.66912120$$

$$W_{2,Slave} = W_{2,x} + W_{2,y} + W_{4,x} + W_{4,y} = -1.1291$$

Equilibrium of work for γ_2

$$W_{2,Master} - W_{2,Slave} = -1.1291 - (-1.1291) = 0.0000$$

The error ϵ_2 relative to the work of the *Master* side is 0.0%.

Equilibrium of work at the complete interface Γ : Finally we sum up the quantities from above and have a look at sum over the complete interface. The difference of work at the interface is

$$\Delta W_{\Gamma} = 0.0000$$

Chapter 4

Convergence study for the 2D and 3D cantilever beam

This chapter is divided into the following sections.

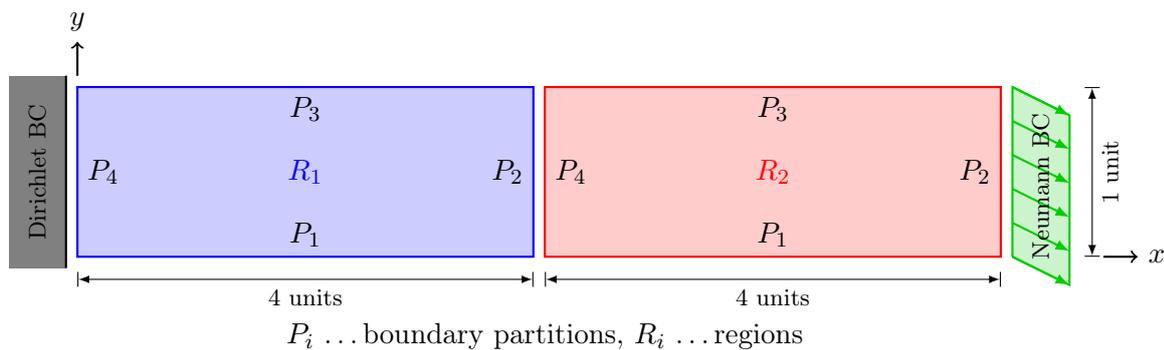
1. Preliminaries, general system settings
2. Analytic solution, solution of the Euler-Bernoulli equation
3. Observations, summary of observations made for 2D and 3D systems
4. Convergence study for the 2D cantilever beam
5. Convergence study for the 3D cantilever beam

4.1 Preliminaries

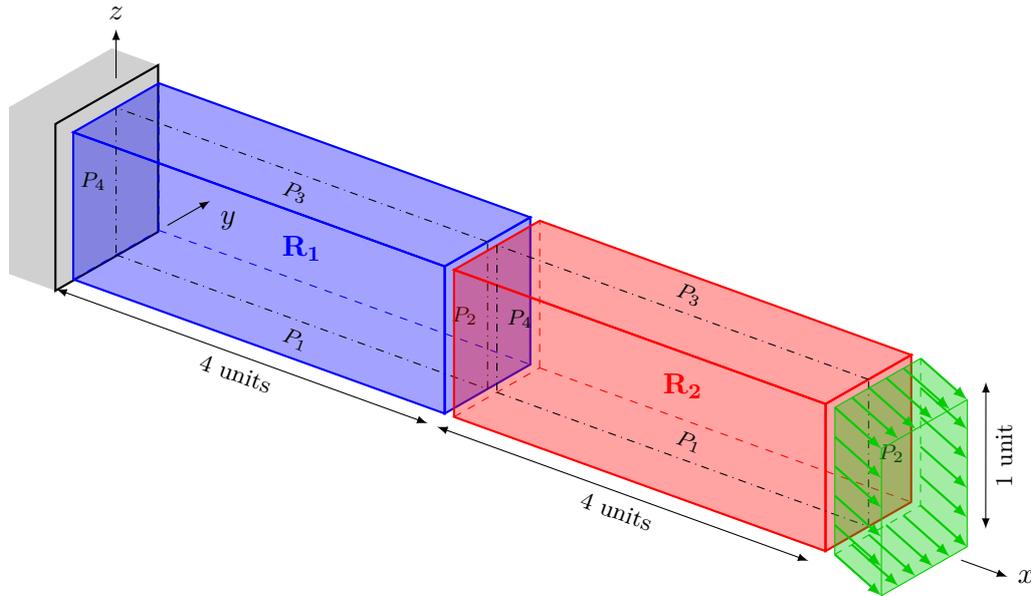
4.1.1 Discretisation

All systems shown below are discretised with the *Boundary Element Method*. And with this discretisation method a two-field problem (displacement and traction) is solved using *plain stress* conditions. This leads to a high convergence rate and less elements are needed to solve the problem.

2D system geometry



3D system geometry



4.1.2 Dirichlet boundary conditions

For all systems the Dirichlet boundary is on the left side of region R_1 in partition P_4 . The boundary condition is the same for all nodes of that partition. For 2D systems it is defined as $u_x = u_y = 0$ and for 3D systems $u_x = u_y = u_z = 0$.

4.1.3 Neumann boundary conditions

For all systems the Neumann boundary is on the right side of region R_2 in partition P_2 . The boundary condition is the same for all elements of that partition. For 2D systems it is defined as $q_x = 1$, $q_y = -1$ and for 3D systems as $q_x = 1$, $q_y = 0$, $q_z = -1$.

4.1.4 Material parameters

For all systems the Young's modulus $E = 1000.0$ and the Poisson ratio $\nu = 0.3$.

4.1.5 Mesh refinement

To be able to show the convergence behaviour of the cantilever beam according to the given boundary conditions different mesh refinement ratio's have been chosen (from coarse to dense meshes). For systems coupled with the *Mortar Method* different refinement ratio's have been chosen for both regions to show the dependency between interface configuration and convergence behaviour.

2D systems, mesh refinement					
rr ₁	rr ₂	Coupling	rr ₁	rr ₂	Coupling
2	2	Beti	3	3	Beti
5	5	Beti	7	7	Beti
9	9	Beti	10	10	Beti
2	2	Mortar	3	3	Mortar
5	5	Mortar	7	7	Mortar
9	9	Mortar	10	10	Mortar
2	3	Mortar	2	5	Mortar
2	7	Mortar	2	9	Mortar
2	10	Mortar	—	—	—
3	2	Mortar	5	2	Mortar
7	2	Mortar	9	2	Mortar
10	2	Mortar	—	—	—
3D systems, mesh refinement					
2	2	Beti	3	3	Beti
4	4	Beti	5	5	Beti
2	2	Mortar	3	3	Mortar
4	4	Mortar	5	5	Mortar
2	3	Mortar	2	4	Mortar
2	5	Mortar	—	—	—
3	2	Mortar	4	2	Mortar
5	2	Mortar	—	—	—
rr _i ... refinement ratio, number of elements per unit					

Table 4.1: Mesh refinement ratio's for 2D and 3D systems

4.1.6 Graph sets

The results are combined to graph sets in order to show the convergence of the solutions and/or the specific behaviour of the discretisation- and coupling methods. They are used to make the difference between the various solutions more visible.

2D systems - graph sets				
Set	Coupling	$rr_1 - rr_2$	Coupling	$rr_1 - rr_2$
1	Beti	2-2,3-3,5-5,7-7,9-9,10-10	—	—
2	Mortar	2-2,3-3,5-5,7-7,9-9,10-10	—	—
3	Mortar	2-3,2-5,2-7,2-9,2-10	Beti	2-2,10-10
4	Mortar	3-2,5-2,7-2,9-2,10-2	Beti	2-2,10-10
3D systems - graph sets				
1	Beti	2-2,3-3,4-4,5-5	—	—
2	Mortar	2-2,3-3,4-4,5-5	—	—
3	Mortar	2-3,2-4,2-5	Beti	2-2,5-5
4	Mortar	3-2,4-2,5-2	Beti	2-2,5-5
Set ... compilation set number				
$rr_i - rr_j$... refinement ratio's of region 1 and 2				
Coupling ... coupling method				

Table 4.2: Graph sets for 2D and 3D systems

4.1.7 Legends and captions used in figures and tables

For abbreviation legends and captions are printed in the following short forms.

Legends:

$$\boxed{B, R_1, P_2, rr_3, rr_2}$$

is to be read like: *BETI* coupling, region 1, partition 2, the refinement ratio in region 1 is 3 elements per unit and the refinement ratio in region 2 is 2 elements per unit.

$$\boxed{u_x, B, rr_2, rr_5}$$

is to be read like: displacement u_x , *BETI* coupling, the refinement ratio in region 1 is 2 elements per unit and the the refinement ratio in region 2 is 5 elements per unit.

Captions: The text below a figure or table.

$$\boxed{\text{Set 1, Mortar, } u_y, R_1, P_2, 1^{\text{st}}}$$

is to be read like: graph Set 1 (see table 4.2), *Mortar Method* coupling, displacement u_y in region 1, partition 2 (see figure 4.3 and 4.30) using element shape functions of 1st order.

4.2 Analytic solution

The analytic solution due to the *Euler-Bernoulli beam theory* is compared with the results of the numerical analysis.

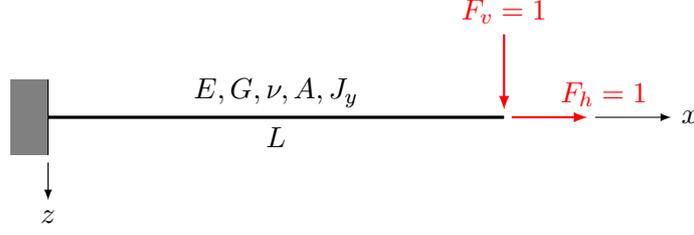


Figure 4.1: Cantilever beam configuration

4.2.1 Displacement functions:

The solution for the displacement distribution in z-direction considering shear deformation is

$$u_z(x) = \int_0^x \frac{M_y(x) \cdot \overline{M}_y(x)}{E \cdot J_y} dx + \int_0^x \frac{Q_z(x) \cdot \overline{Q}_z(x)}{G \cdot A \cdot \kappa_s} dx \quad (4.1)$$

with

$$M_y(x) = F_v(x - L); \quad \overline{M}_y(x) = 1(x - L) \quad (4.2)$$

$$Q_z(x) = F_v; \quad \overline{Q}_z(x) = 1 \quad (4.3)$$

we get the transversal displacement function

$$u_z(x) = \frac{F_v}{E \cdot J_y} \left(\frac{x^2 \cdot L}{2} - \frac{x^3}{6} \right) + \frac{F_v}{G \cdot A \cdot \kappa_s} x \quad (4.4)$$

The cross-section rotation function (without shear distortion) is

$$\varphi_y(x) = \frac{du_z(x)}{dx} = \frac{F_v}{E \cdot J_y} \left(x \cdot L - \frac{x^2}{2} \right) \quad (4.5)$$

The function for the displacements in x-direction at bottom and top edge (center line $\pm h/2$) of the beam is

$$u_{x,M}(x) = \varphi_y(x) \cdot \left(\pm \frac{h}{2} \right) = \left[\frac{F_v}{E \cdot J_y} \left(x \cdot L - \frac{x^2}{2} \right) \right] \cdot \left(\pm \frac{h}{2} \right) \quad (4.6)$$

The solution for the displacement distribution due to longitudinal forces in x-direction is:

$$u_{x,N}(x) = \int_0^x \frac{N_x(x) \cdot \overline{N}_x(x)}{E \cdot A} dx \quad (4.7)$$

with

$$N_x(x) = F_h; \quad \overline{N}_x(x) = 1 \quad (4.8)$$

we get the displacement function in x-direction

$$u_{x,N}(x) = \frac{F_h}{E \cdot A} x \quad (4.9)$$

The complete displacement distribution at bottom and top edge of the beam in x-direction is

$$u_x(x) = u_{x,M}(x) + u_{x,N}(x) = \pm \frac{h}{2} \left[\frac{F_v}{E \cdot J_y} \left(x \cdot L - \frac{x^2}{2} \right) \right] + \frac{F_h}{E \cdot A} x \quad (4.10)$$

We also need the following parameters:

b	$= 1.0$... cantilever width
h	$= 1.0$... cantilever height
E	$= 1000.0$... Young's modulus
ν	$= 0.3$... Poisson ratio
G	$= \frac{E}{2(1+\nu)} = 384.6154$... shear modulus
J_y	$= \frac{b \cdot h^3}{12} = 0.0833$... area moment of inertia about the y-axis
A	$= b \cdot h = 1.0$... cross-section area
κ_s	$= 0.85$... shear factor for rectangular shape (Timoshenko)

4.2.2 Transversal displacement maximum $u_{z,\max}$:

To be compared later we evaluate the transversal displacement (4.4) at the right side of the beam. We get the maximum displacement in z-direction $u_{z,\max}$.

$$u_{z,\max} = u_z(x = 8.0) = \underbrace{\frac{1.0}{1000.0 \cdot 0.0833} \left(\frac{8.0^2 \cdot 8.0}{2} - \frac{8.0^3}{6} \right)}_{2.0480} + \underbrace{\frac{1.0}{384.6154 \cdot 1.0 \cdot 0.85}}_{0.0245} 8.0 = \mathbf{2.0725} \quad (4.11)$$

4.2.3 Longitudinal displacement extrema $u_{x,\max}$, $u_{x,\min}$:

We also evaluate the longitudinal displacements (4.10) at the right side of the beam. We get the maximum (top of the beam) and minimum (bottom of the beam) displacements in x-direction $u_{x,\max}$ and $u_{x,\min}$. The latter is needed for comparison reasons later on this document.

$$u_{x,\max} = u_x\left(x = 8.0, +\frac{h}{2}\right) = \underbrace{\frac{1.0}{2} \left[\frac{1.0}{1000.0 \cdot 0.0833} \left(8.0 \cdot 8.0 - \frac{8.0^2}{2} \right) \right]}_{0.1920} + \underbrace{\frac{1.0}{1000.0 \cdot 1.0}}_{0.0080} 8.0 = \mathbf{0.2000} \quad (4.12)$$

$$u_{x,\min} = u_x\left(x = 8.0, -\frac{h}{2}\right) = \underbrace{-\frac{1.0}{2} \left[\frac{1.0}{1000.0 \cdot 0.0833} \left(8.0 \cdot 8.0 - \frac{8.0^2}{2} \right) \right]}_{-0.1920} + \underbrace{\frac{1.0}{1000.0 \cdot 1.0}}_{0.0080} 8.0 = \mathbf{-0.1840} \quad (4.13)$$

4.2.4 Longitudinal and transversal displacement functions:

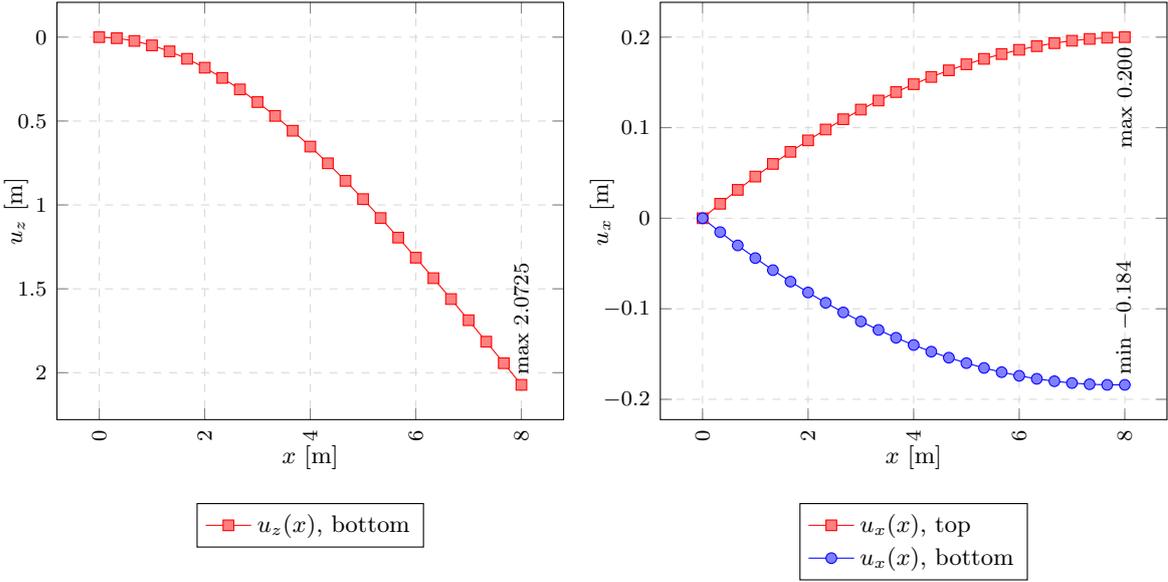


Figure 4.2: Analytic displacement functions

The picture at the left hand side shows the distribution of the transversal displacements along the cantilever beam located on the bottom line of region 1 and 2. The picture on the right hand side shows the longitudinal displacements at the bottom and the top line of region 1 and 2.

4.3 Observations

This section shows the results of the numerical analysis of the cantilever beam for 2D and 3D. The following sections contain graphical and tabulated results and more detailed explanations.

Convergence behaviour: The following observations are the same for 2D and 3D systems.

1. In general the convergence behaviour of the complete system shows major dependency on the convergence behaviour along the regions but minor on the actual interface configuration.
2. *Mortar Method* coupled systems with the same mesh refinement ratio's in both regions turn out to have the same convergence behaviour like *BETI* coupled systems. They also approach the same displacement values.
3. When using different mesh refinement ratio's in both regions we can observe that the systems still converge against the *BETI* coupled solution.
4. The disagreement of displacements at opposite points at the interface show the following behaviour:
 - They have small values, scaling with the displacement at the interface point. For 1st order element shape functions this scaling factor is about $10^{-2} \div 10^{-3}$ and for 2nd order element shape functions it is about $10^{-3} \div 10^{-4}$.
 - They become small when using high refinement ratio's. Having a dense mesh at one side of the interface (Master side) is advantageous.
 - The influence on the convergence behaviour of the system is getting lost with increasing refinement ratio's.
 - The convergence behaviour of the disagreement of displacements is *not* strictly monotonous, strict monotony starts from a certain mesh refinement ratio.
5. As expected the convergence behaviour becomes better when having more nodal degree's of freedom. The convergence behaviour of the displacements at the interface do not follow this trend directly. The following three circumstances have influence on the convergence behaviour of the displacements at the interface.
 - (a) nodal degree's of freedom of the interfacing meshes.
 - (b) *Lagrange multipliers* of the *Mortar Finite Elements* (*Master* element degree's of freedom).
 - (c) order of the element shape functions.
6. Increasing the element shape function order leads to a higher convergence rate which is as expected. The disagreement of displacements are still due to the *Lagrange multiplier fields* because the number of *Lagrange multipliers* does not increase by the order of the element shape functions.

2D and 3D systems: The difference between 2D and 3D systems is pretty small. The convergence behaviour is almost the same. There is little difference in the displacement disagreements of opposite points at the interface. Further the displacements of both systems approach almost the same end value.

4.4 Convergence study for the 2D cantilever beam

This section contains the graphical and tabulated results of the numerical analysis for the 2D cantilever beam.

4.4.1 System geometry

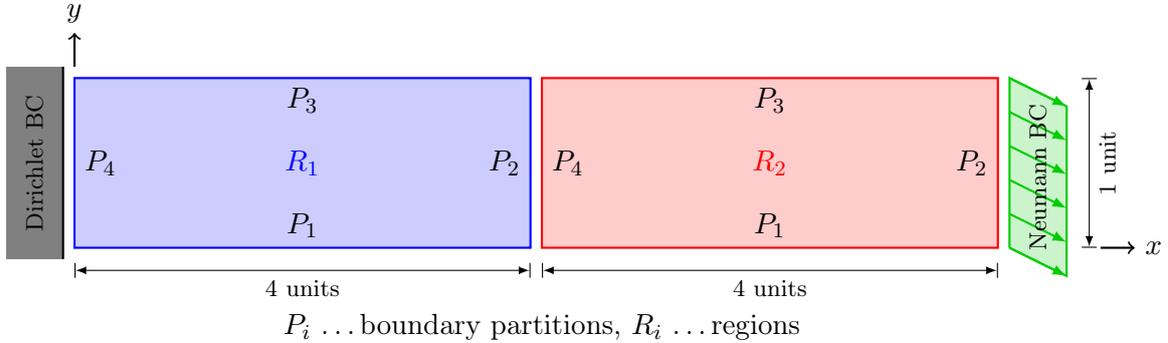


Figure 4.3: 2D cantilever with 2 regions

4.4.2 Interface displacements using first order shape functions

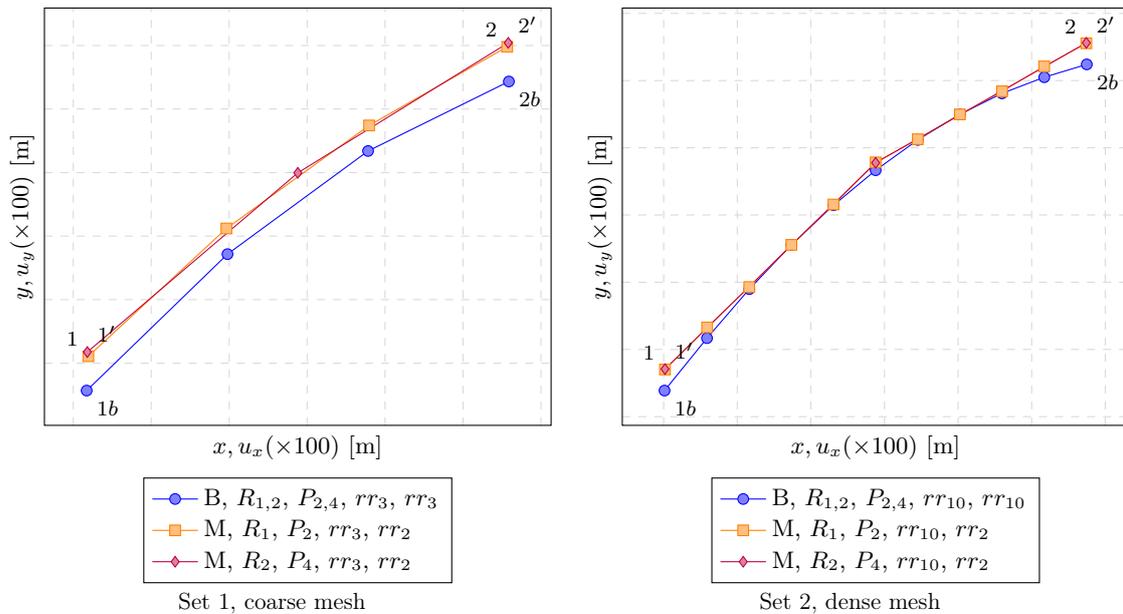


Figure 4.4: Displaced interface configurations

In Set 1 (coarse mesh) we can see the gaps and overlaps of the both regions at the interface when using the *Mortar Method*. The blue line marks the displaced figure of a *BETI* coupled system which has an offset to the other lines. If the mesh density advances (Set 2, dense mesh) the gaps and overlaps become significantly smaller. In Set 2 it is interesting to see that the nodes of the region with the coarse mesh approach the displaced figure of the region with the dense mesh.

Displacement, coarse mesh						
\mathbf{u}	\mathbf{u}_1	$\mathbf{u}_{1'}$	\mathbf{u}_{1b}	$\Delta\mathbf{u}_{1'-1}$	$\Delta\mathbf{u}_{1-1b}$	$\Delta\mathbf{u}_{1'-1b}$
u_x	-0.13033300	-0.13101200	-0.13143100	-0.00067000	0.00108000	0.00041000
u_y	-0.61578100	-0.61564800	-0.61686100	0.00014000	0.00108000	0.00122000
\mathbf{u}	\mathbf{u}_2	$\mathbf{u}_{2'}$	\mathbf{u}_{2b}	$\Delta\mathbf{u}_{2'-2}$	$\Delta\mathbf{u}_{2-2b}$	$\Delta\mathbf{u}_{2'-2b}$
u_x	0.13834800	0.13902800	0.13943200	0.00067000	-0.00108000	-0.00041000
u_y	-0.61578100	-0.61591700	-0.61713300	0.00012000	0.00108000	0.00120000

Table 4.3: Set 1, displacement values

Displacement, dense mesh						
\mathbf{u}	\mathbf{u}_1	$\mathbf{u}_{1'}$	\mathbf{u}_{1b}	$\Delta\mathbf{u}_{1'-1}$	$\Delta\mathbf{u}_{1-1b}$	$\Delta\mathbf{u}_{1'-1b}$
u_x	-0.13912400	-0.13912400	-0.13943000	0.00000000	0.00030000	0.00030000
u_y	-0.64859600	-0.64858400	-0.64922300	0.00000000	0.00063000	0.00063000
\mathbf{u}	\mathbf{u}_2	$\mathbf{u}_{2'}$	\mathbf{u}_{2b}	$\Delta\mathbf{u}_{2'-2}$	$\Delta\mathbf{u}_{2-2b}$	$\Delta\mathbf{u}_{2'-2b}$
u_x	0.14711600	0.14711600	0.14741200	0.00000000	-0.00030000	-0.00030000
u_y	-0.64859600	-0.64887800	-0.64951900	0.00002000	0.00063000	0.00064000

Table 4.4: Set 2, displacement values

Looking at this tables we can assert two things.

1. The disagreement of displacements in x- and y-direction are about $10^{-2} \div 10^{-3}$ times the absolute values at opposite points at the interface.
2. For regions with very different mesh densities (Set 2) we recognize a disagreement of displacements about $10^{-3} \div 10^{-4}$ times the absolute displacement. It seems that a higher mesh density on one side of the interface is advantageous. This can be explained with the higher number of *Lagrange multipliers* which are limited to the number of *Master* elements.

4.4.3 Interface displacements using second order shape functions

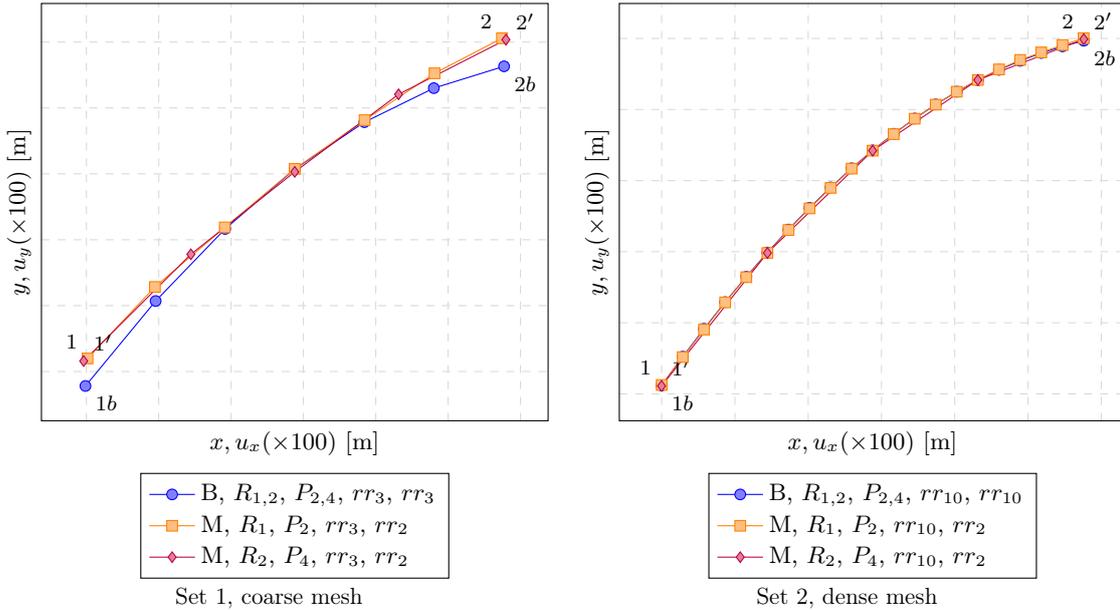


Figure 4.5: Displaced interface configurations

In Set 1 and 2 we can see that using elements with second order shape functions the results of both regions at the interface agree better then with first order shape functions. Set 2 additionally shows

that the difference to a *BETI* coupled system becomes very small. Mesh adaptation at the interface and the convergence for the whole system becomes better with increasing shape function order and increasing number of *Lagrange multiplier fields* (*Master elements*).

Displacements, coarse mesh						
\mathbf{u}	\mathbf{u}_1	$\mathbf{u}_{1'}$	\mathbf{u}_{1b}	$\Delta \mathbf{u}_{1'-1}$	$\Delta \mathbf{u}_{1-1b}$	$\Delta \mathbf{u}_{1'-1b}$
u_x	-0.13924100	-0.14190800	-0.14067700	-0.00267000	0.00143000	-0.00124000
u_y	-0.65360400	-0.65368200	-0.65443900	-0.00008000	0.00082000	0.00075000
\mathbf{u}	\mathbf{u}_2	$\mathbf{u}_{2'}$	\mathbf{u}_{2b}	$\Delta \mathbf{u}_{2'-2}$	$\Delta \mathbf{u}_{2-2b}$	$\Delta \mathbf{u}_{2'-2b}$
u_x	0.14724300	0.14993500	0.14867000	0.00269000	-0.00142000	0.00127000
u_y	-0.65360400	-0.65392600	-0.65473600	-0.00005000	0.00085000	0.00081000

Table 4.5: Set 1, displacement values

Displacements, dense mesh						
\mathbf{u}	\mathbf{u}_1	$\mathbf{u}_{1'}$	\mathbf{u}_{1b}	$\Delta \mathbf{u}_{1'-1}$	$\Delta \mathbf{u}_{1-1b}$	$\Delta \mathbf{u}_{1'-1b}$
u_x	-0.13996000	-0.14004800	-0.14007000	-0.00009000	0.00012000	0.00003000
u_y	-0.65175000	-0.65177900	-0.65176600	-0.00003000	0.00002000	-0.00002000
\mathbf{u}	\mathbf{u}_2	$\mathbf{u}_{2'}$	\mathbf{u}_{2b}	$\Delta \mathbf{u}_{2'-2}$	$\Delta \mathbf{u}_{2-2b}$	$\Delta \mathbf{u}_{2'-2b}$
u_x	0.14794000	0.14804600	0.14804800	0.00010000	-0.00010000	0.00000000
u_y	-0.65175000	-0.65202000	-0.65206300	-0.00003000	0.00008000	0.00005000

Table 4.6: Set 2, displacement values

Comparing the values in the tables the differences at the interface are much smaller than for the case of first order shape functions.

4.4.4 Discretisation with first order shape functions

- Displacements u_x at the right side of region 1, partition 2 (interface)
- Displacements u_y at the right side of region 1, partition 2 (interface)
- Displacements u_x at the left side of region 2, partition 4 (interface)
- Displacements u_y at the left side of region 2, partition 4 (interface)
- Displacements u_x at the bottom edge of region 1+2, both regions partition 1
- Displacements u_y at the bottom edge of region 1+2, both regions partition 1

Displacements u_x , interface, reg. 1, part. 2

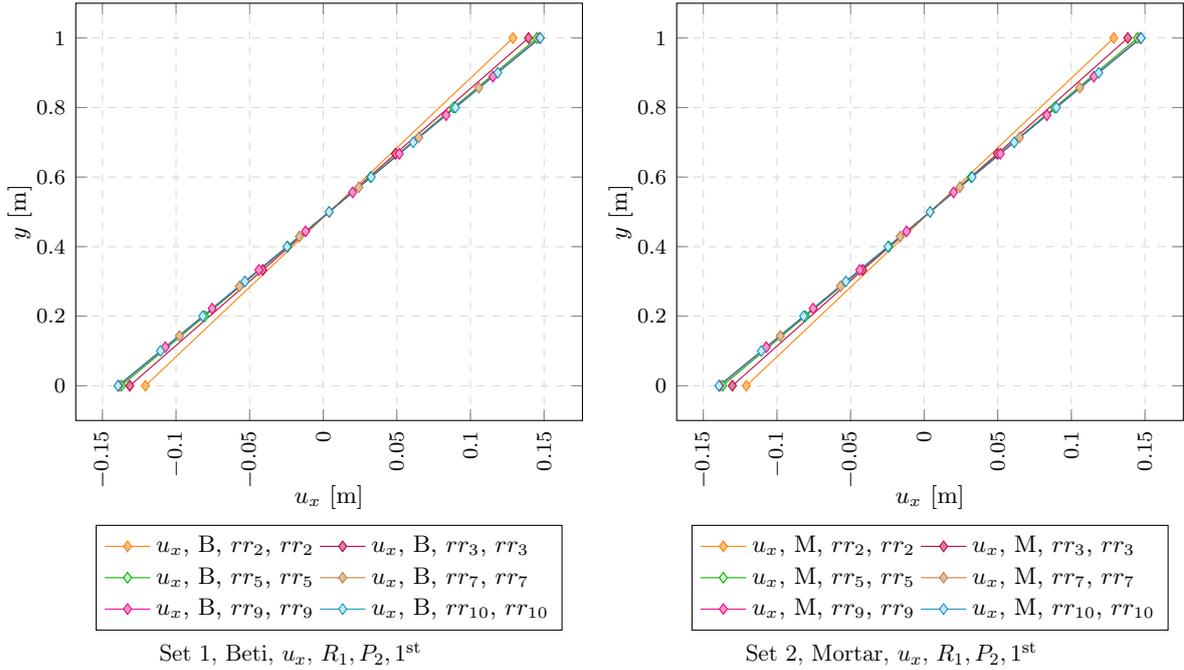


Figure 4.6: Convergence behaviour of *BETI* and *Mortar Method* coupling

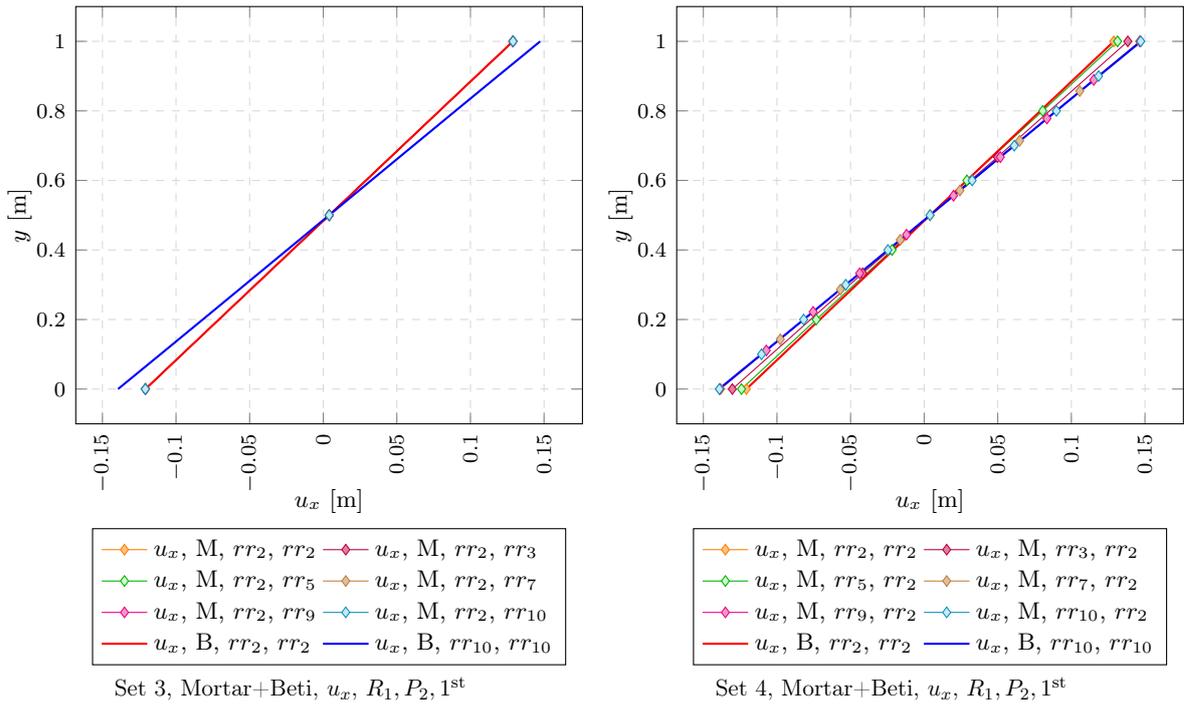


Figure 4.7: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

Set 3 has only two visible lines. All lines are part of the left side of the interface where the mesh refinement ratio is constant. Therefore all lines of the *Mortar Method* coupled systems are hiding behind the red line from the coarse meshed *BETI* coupled system. All other sets show typical convergence behaviour.

Displacements u_y , interface, reg. 1, part. 2

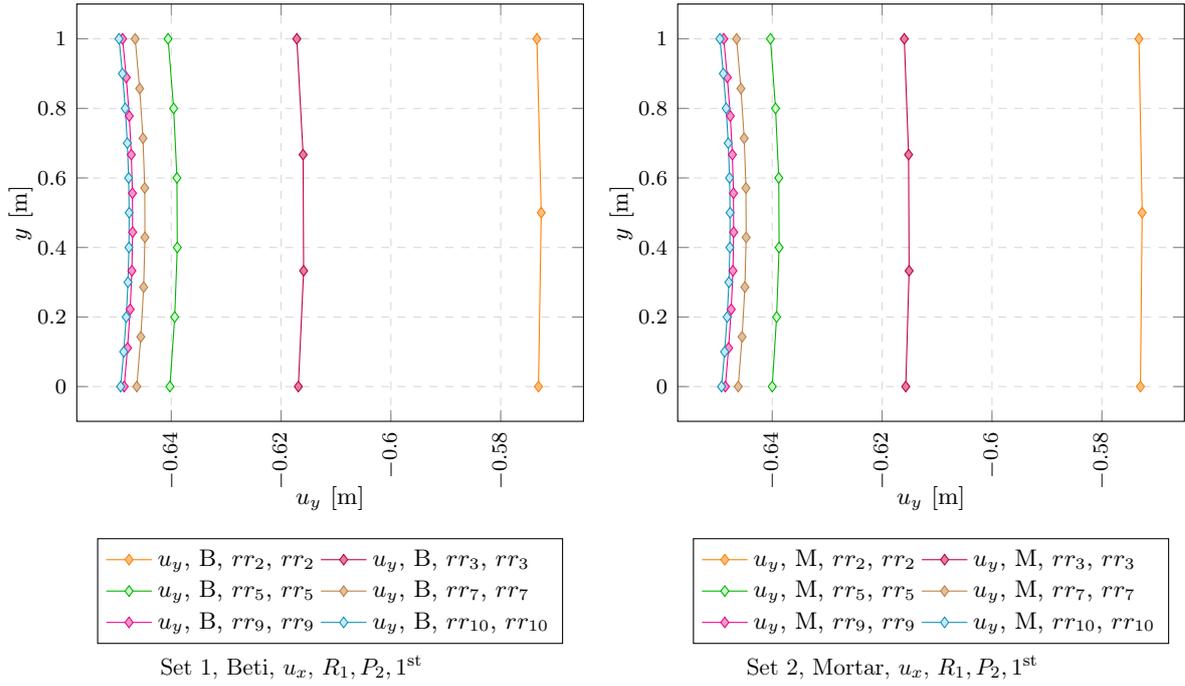


Figure 4.8: Convergence behaviour of *BETI* and *Mortar Method* coupling

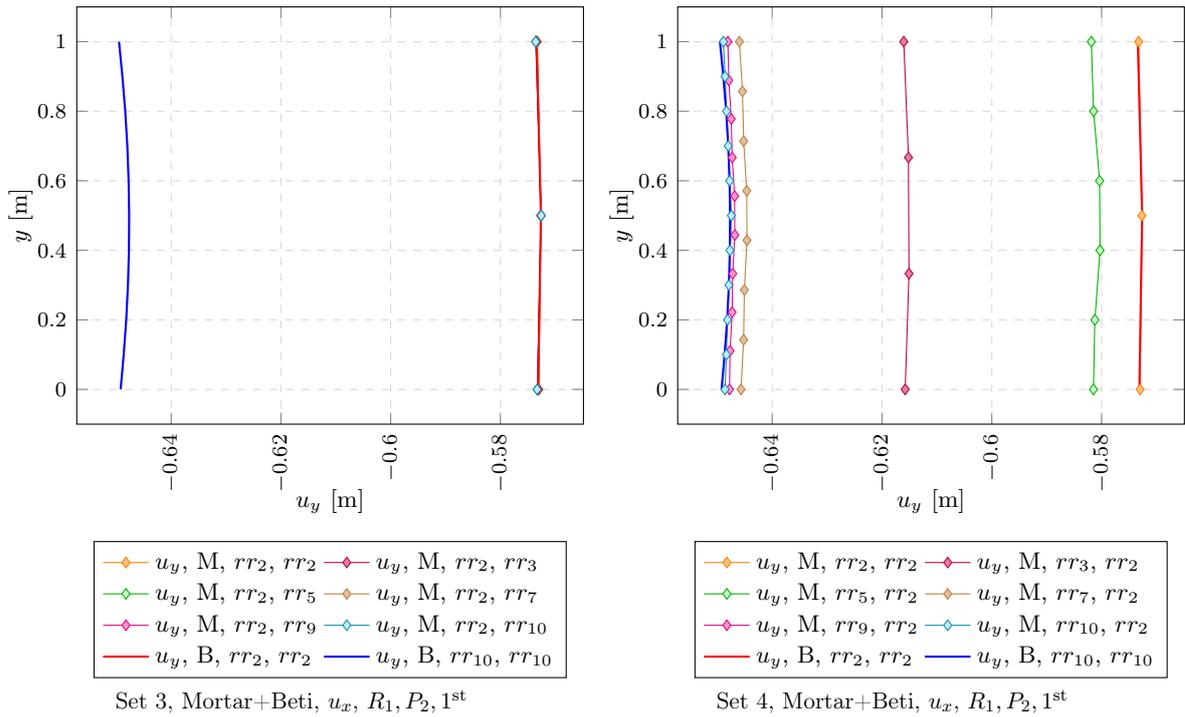


Figure 4.9: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

Set 3 has only two visible lines. All lines are part of the left side of the interface where the mesh is coarse. Therefore all lines of the *Mortar Method* coupled systems are hiding behind the red line from the coarse meshed *BETI* coupled system. All other sets show typical convergence behaviour.

Displacements u_x , interface, reg. 2, part. 4

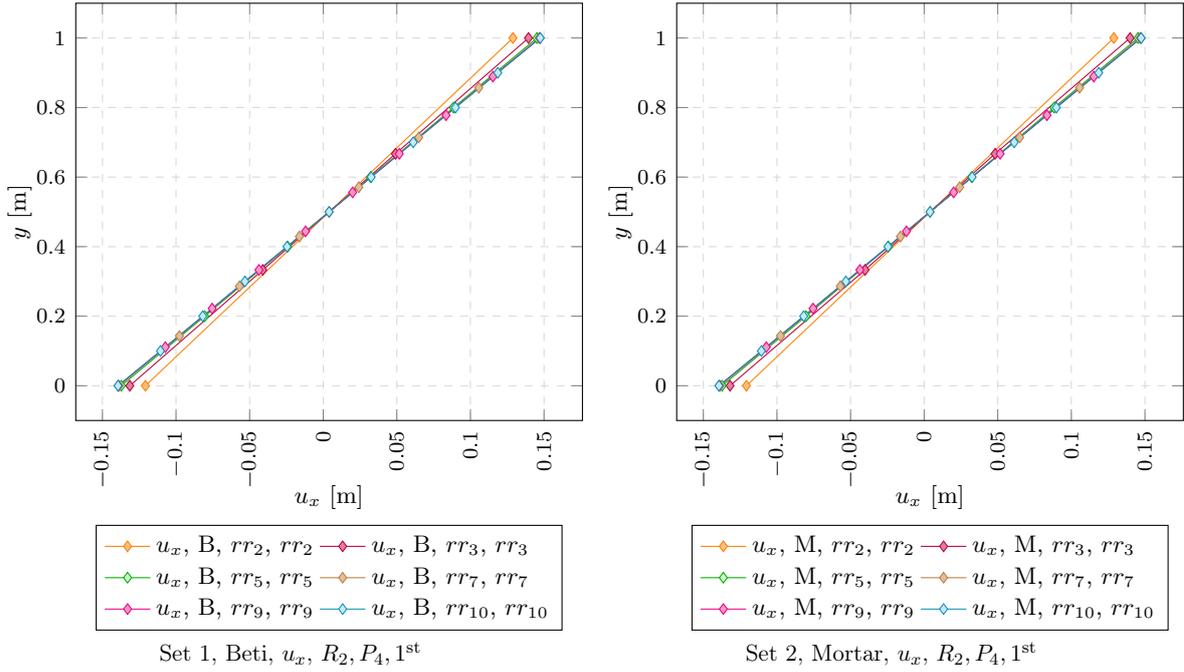


Figure 4.10: Convergence behaviour of *BETI* and *Mortar Method* coupling

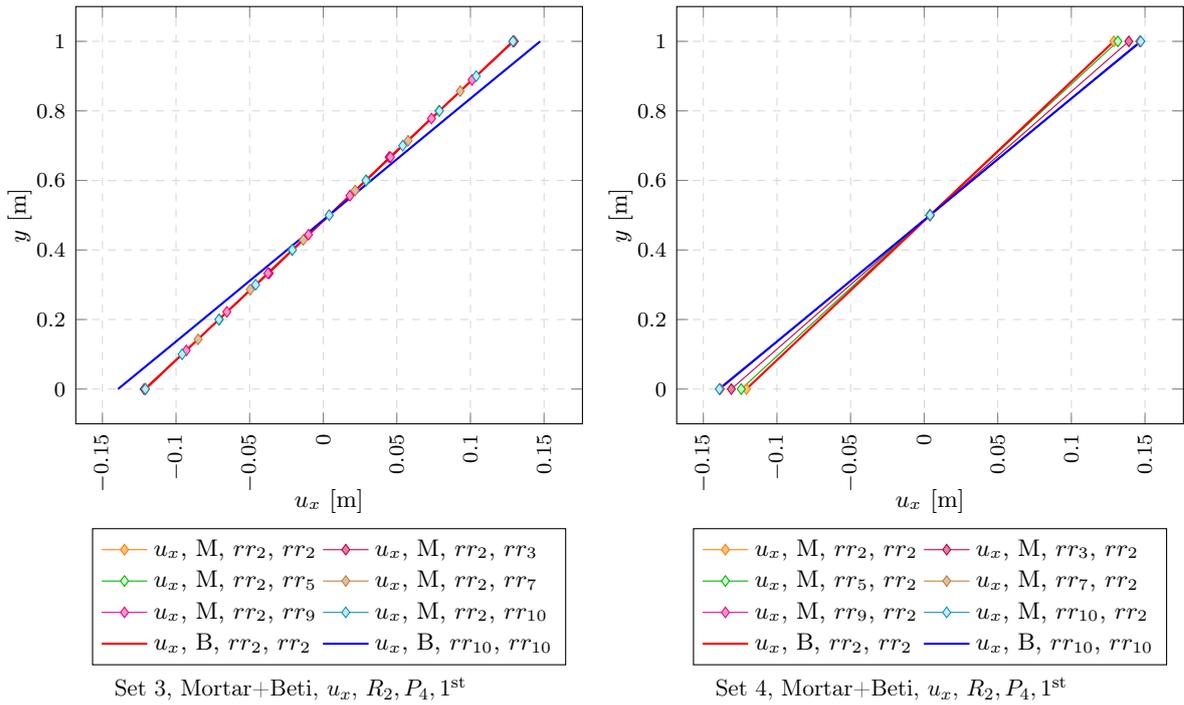


Figure 4.11: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

In Set 3 two lines are visible. In this case all lines are part of the right side of the interface. The mesh is coarse in region 1 and all *Mortar Method* coupled systems approach the displaced figure of region 1 which is the same like the red line from the coarse meshed *BETI* coupled system. All other sets show typical convergence behaviour.

Displacements u_y , interface, reg. 2, part. 4

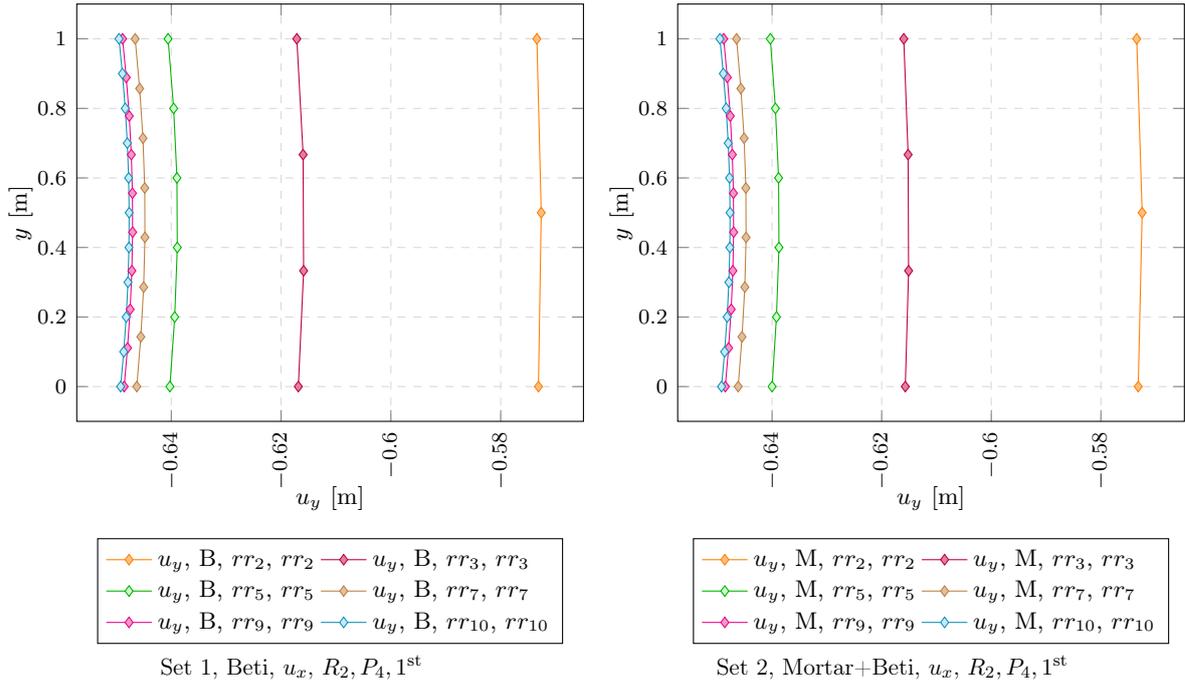


Figure 4.12: Convergence behaviour of *BETI* and *Mortar Method* coupling

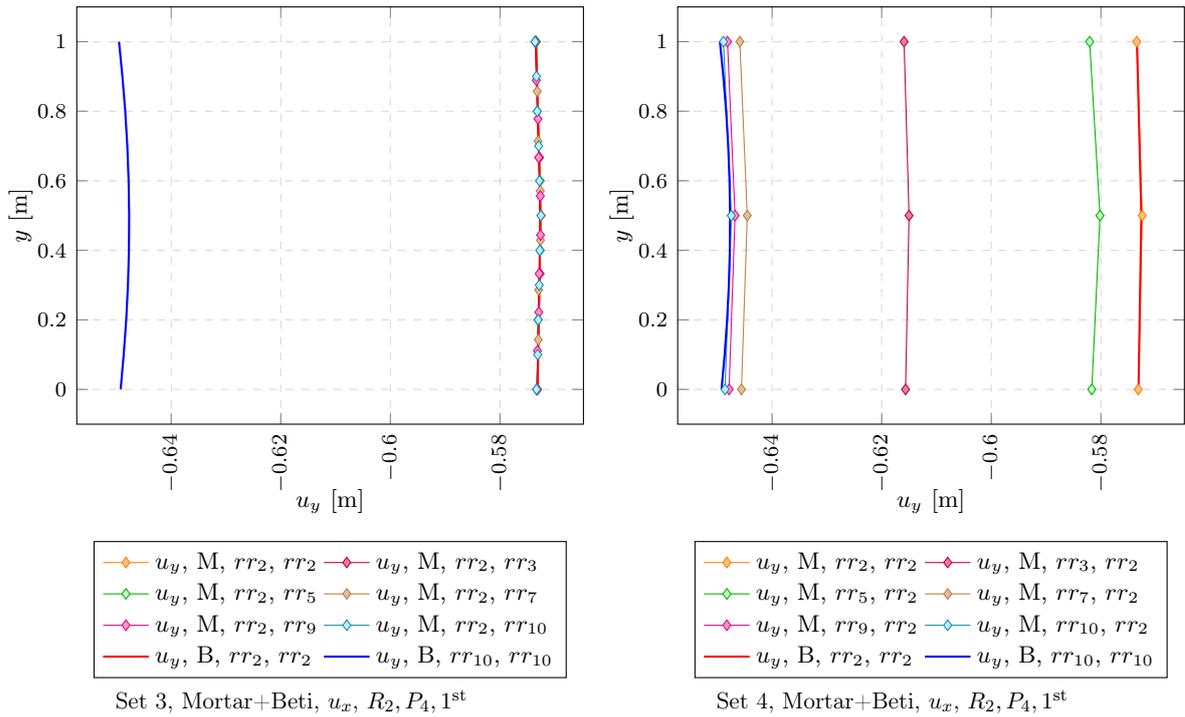


Figure 4.13: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

In Set 3 two lines are visible. In this case all lines are part of the right side of the interface. The mesh is coarse in region 1 and all *Mortar Method* coupled systems approach the displaced figure of region 1 which is the same like the red line from the coarse meshed *BETI* coupled system. All other sets show typical convergence behaviour.

Displacements u_x , bottom edge, reg. 1+2, part. 1

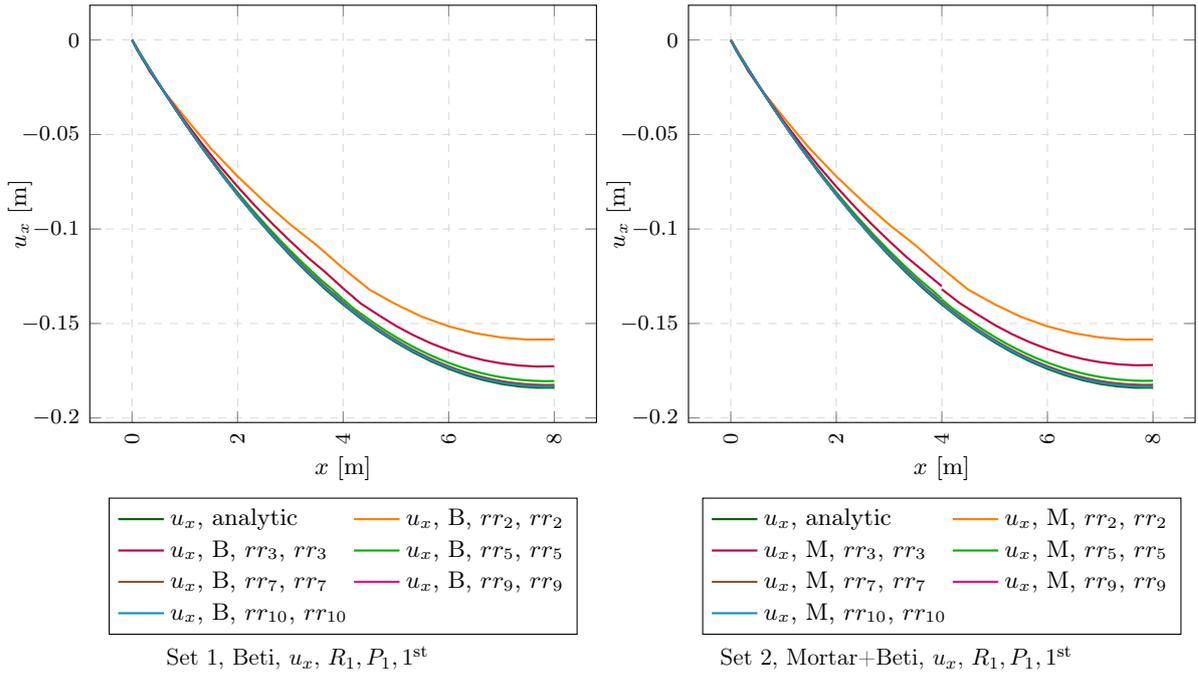


Figure 4.14: Convergence behaviour of *BETI* and *Mortar Method* coupling

Both sets show typical convergence behaviour. In the middle of the system we can see some small disagreements of displacements in x-direction at the interface.

Displacements u_x					
Coupling	rr	v_l	v_r	$\Delta v = v_r - v_l$	v_e
Mortar	2	-0.12077800	-0.12075300	0.00002000	-0.15842400
Mortar	3	-0.13031100	-0.13190000	-0.00159000	-0.17199200
Mortar	5	-0.13694400	-0.13730200	-0.00035000	-0.18030100
Mortar	7	-0.13859400	-0.13873300	-0.00014000	-0.18237900
Mortar	9	-0.13922100	-0.13929000	-0.00008000	-0.18319800
Mortar	10	-0.13943000	-0.13942900	0.00000000	-0.18344400
Beti	2	-0.12080900	-0.12080900	0.00000000	-0.15841500
Beti	10	-0.13943000	-0.13943000	0.00000000	-0.18344400

$v_l, v_r \dots$ value at the *left, right* side of the interface
 $v_e \dots$ value at beam end, $rr \dots$ refinement ratio

Table 4.7: Set 2, disagreement of displacements at the interface

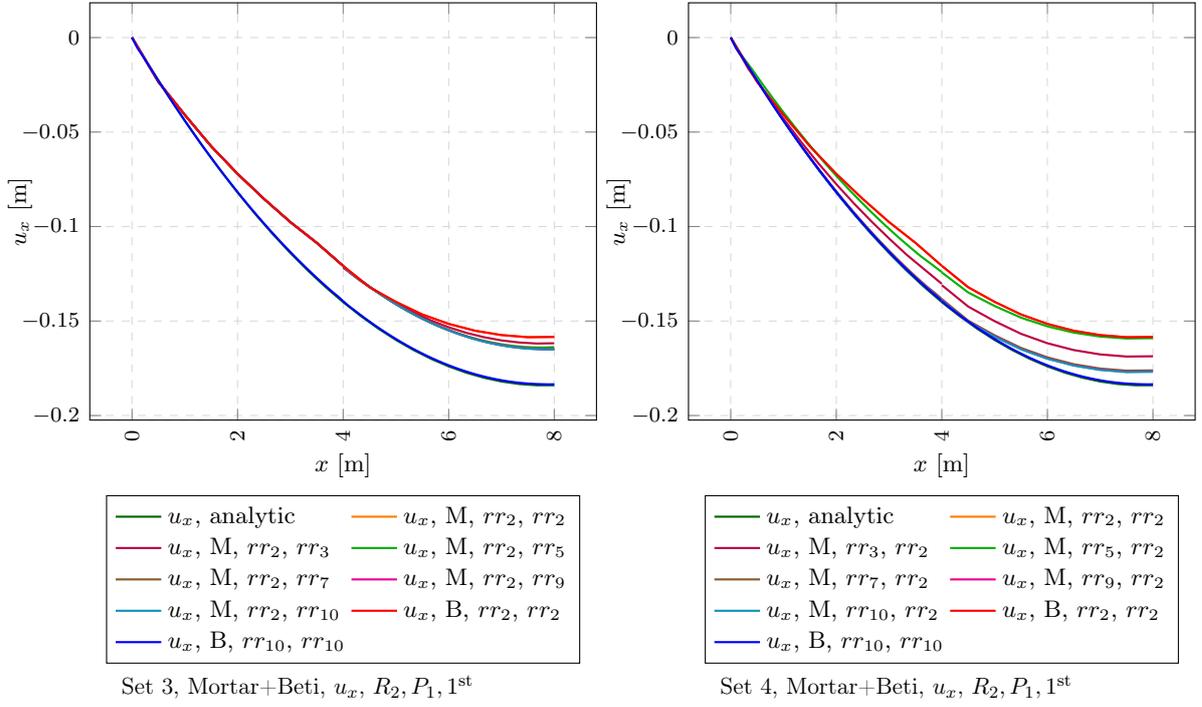


Figure 4.15: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

In Set 3 all systems converge, but not against the blue line of the *BETI* coupled system which is almost identical to the analytic solution. This is due to the coarse mesh in region 1.

Displacements u_x					
Coupling	rr	v_l	v_r	$\Delta v = v_r - v_l$	v_e
Mortar	2	-0.12077800	-0.12075300	0.00002000	-0.15842400
Mortar	3	-0.13033300	-0.13101200	-0.00067000	-0.16864800
Mortar	5	-0.12412000	-0.12429000	-0.00017000	-0.15906200
Mortar	7	-0.13830600	-0.13837400	-0.00006000	-0.17611600
Mortar	9	-0.13887400	-0.13891900	-0.00005000	-0.17666500
Mortar	10	-0.13912400	-0.13912400	0.00000000	-0.17688000
Beti	2	-0.12080900	-0.12080900	0.00000000	-0.15841500
Beti	10	-0.13943000	-0.13943000	0.00000000	-0.18344400

$v_l, v_r \dots$ value at the *left, right* side of the interface
 $v_e \dots$ value at beam end, $rr \dots$ refinement ratio

Table 4.8: Set 4, disagreement of displacements at the interface

Displacements u_y , bottom edge, reg. 1+2, part. 1

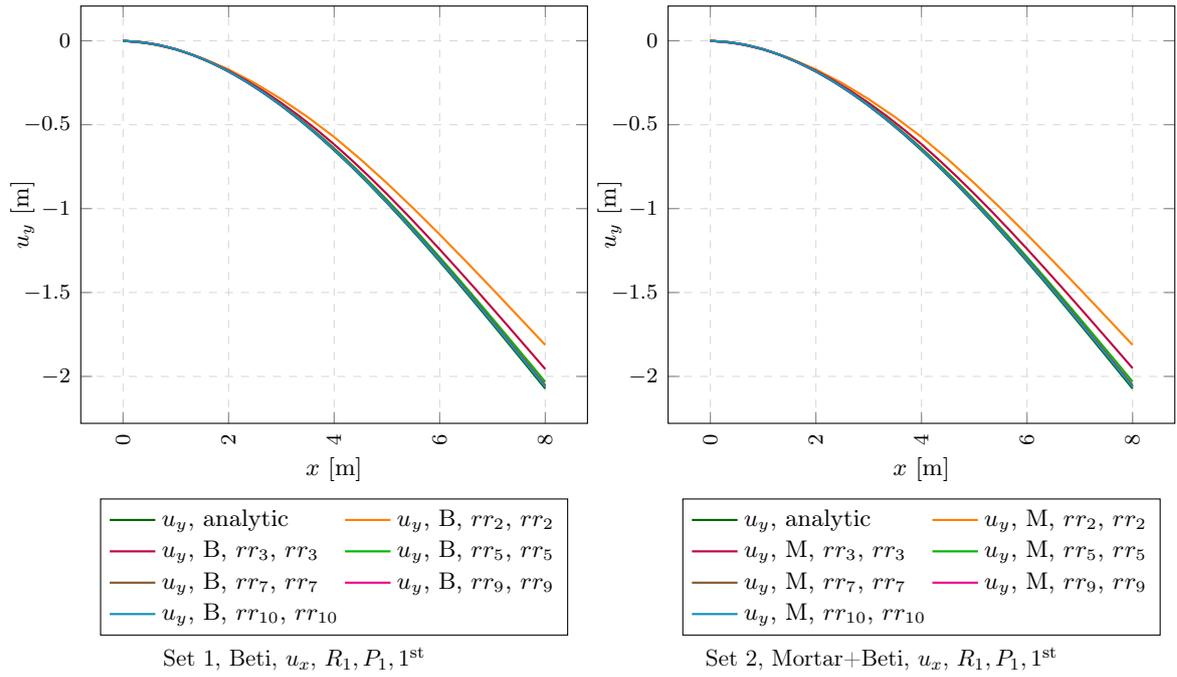


Figure 4.16: Convergence behaviour of *BETI* and *Mortar Method* coupling

Both sets show typical convergence behaviour. The disagreements of displacements at the interface do not have much influence on the displacements in y-direction.

Displacements u_y					
Coupling	rr	v_l	v_r	$\Delta v = v_r - v_l$	v_e
Mortar	2	-0.57298600	-0.57318600	-0.00020000	-1.81253000
Mortar	3	-0.61569700	-0.61569200	0.00000000	-1.95092000
Mortar	5	-0.63999900	-0.63999700	0.00000000	-2.03172000
Mortar	7	-0.64618800	-0.64618800	0.00000000	-2.05206000
Mortar	9	-0.64853600	-0.64853600	0.00000000	-2.05972000
Mortar	10	-0.64922300	-0.64922400	-0.00002000	-2.06198000
Beti	2	-0.57313200	-0.57313200	0.00000000	-1.81263000
Beti	10	-0.64922300	-0.64922300	0.00000000	-2.06199000

$v_l, v_r \dots$ value at the *left, right* side of the interface
 $v_e \dots$ value at beam end, $rr \dots$ refinement ratio

Table 4.9: Set 2, disagreement of displacements at the interface

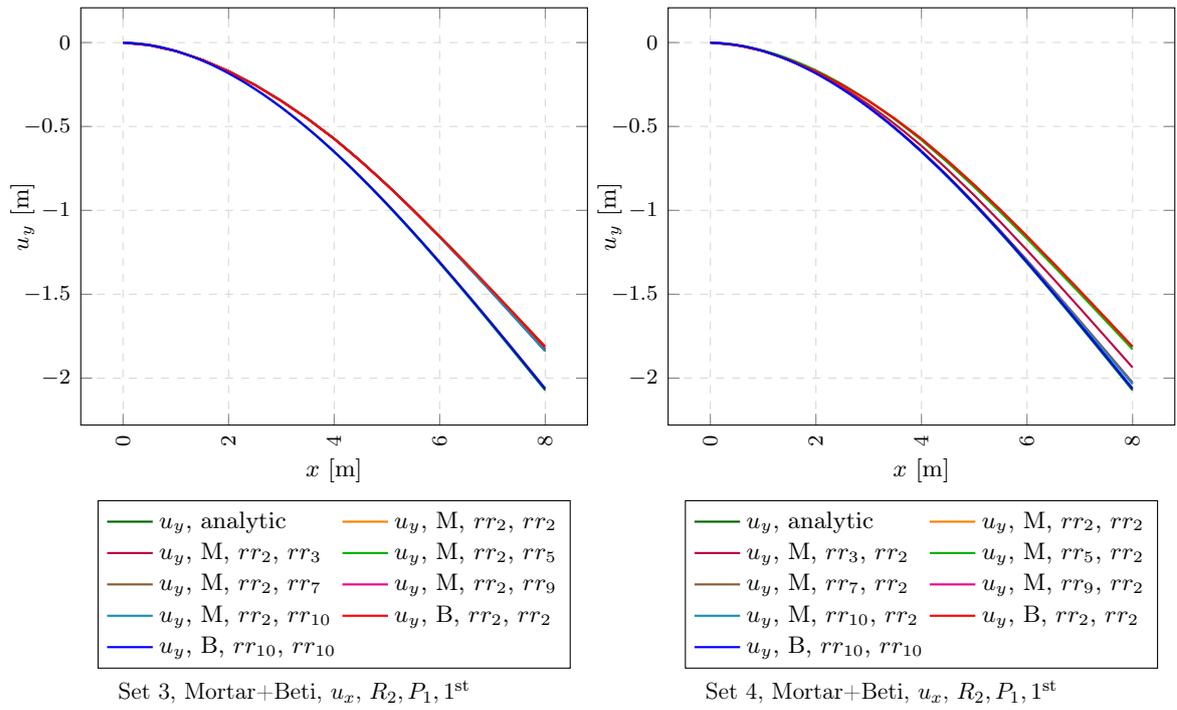


Figure 4.17: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

In Set 3 all systems converge, but not against the blue line of the *BETI* coupled system which is almost identical to the analytic solution. This is due to the coarse mesh in region 1.

Displacements u_y					
Coupling	rr	v_l	v_r	$\Delta v = v_r - v_l$	v_e
Mortar	2	-0.57298600	-0.57318600	-0.00020000	-1.81253000
Mortar	3	-0.61578100	-0.61564800	0.00014000	-1.93668000
Mortar	5	-0.58149300	-0.58166200	-0.00017000	-1.82844000
Mortar	7	-0.64568200	-0.64559600	0.00008000	-2.02592000
Mortar	9	-0.64778100	-0.64785000	-0.00006000	-2.03253000
Mortar	10	-0.64859600	-0.64858400	0.00000000	-2.03494000
Beti	2	-0.57313200	-0.57313200	0.00000000	-1.81263000
Beti	10	-0.64922300	-0.64922300	0.00000000	-2.06199000

$v_l, v_r \dots$ value at the *left, right* side of the interface
 $v_e \dots$ value at beam end, *rr* ... refinement ratio

Table 4.10: Set 4, disagreement of displacements at the interface

4.4.5 Discretisation with second order shape functions

In this section all systems from the previous section are shown again. But now the elements are discretised with 2nd order shape functions. There are no extra explanations given. This is for the reason that the convergence behaviour is same like before but with higher convergence ratio.

- Displacements u_x at the right side of region 1, partition 2 (interface)
- Displacements u_y at the right side of region 1, partition 2 (interface)
- Displacements u_x at the left side of region 2, partition 4 (interface)
- Displacements u_y at the left side of region 2, partition 4 (interface)
- Displacements u_x at the bottom edge of region 1+2, both regions partition 1
- Displacements u_y at the bottom edge of region 1+2, both regions partition 1

Displacements u_x , interface, reg. 1, part. 2

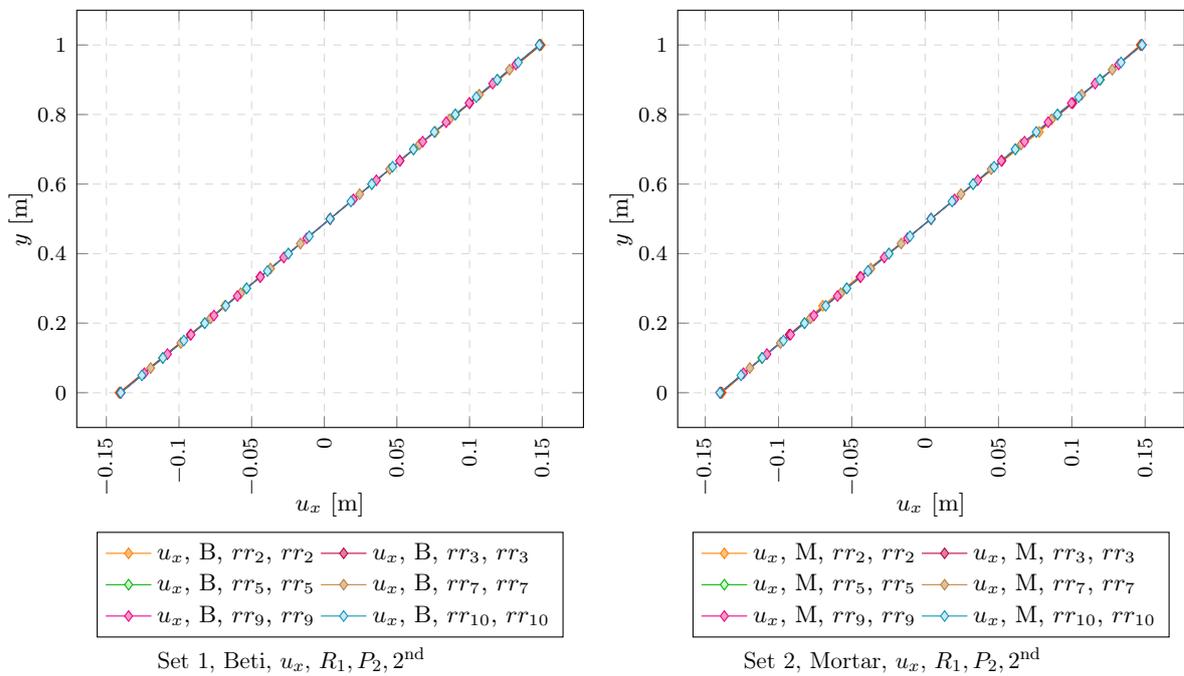


Figure 4.18: Convergence behaviour of *BETI* and *Mortar Method* coupling

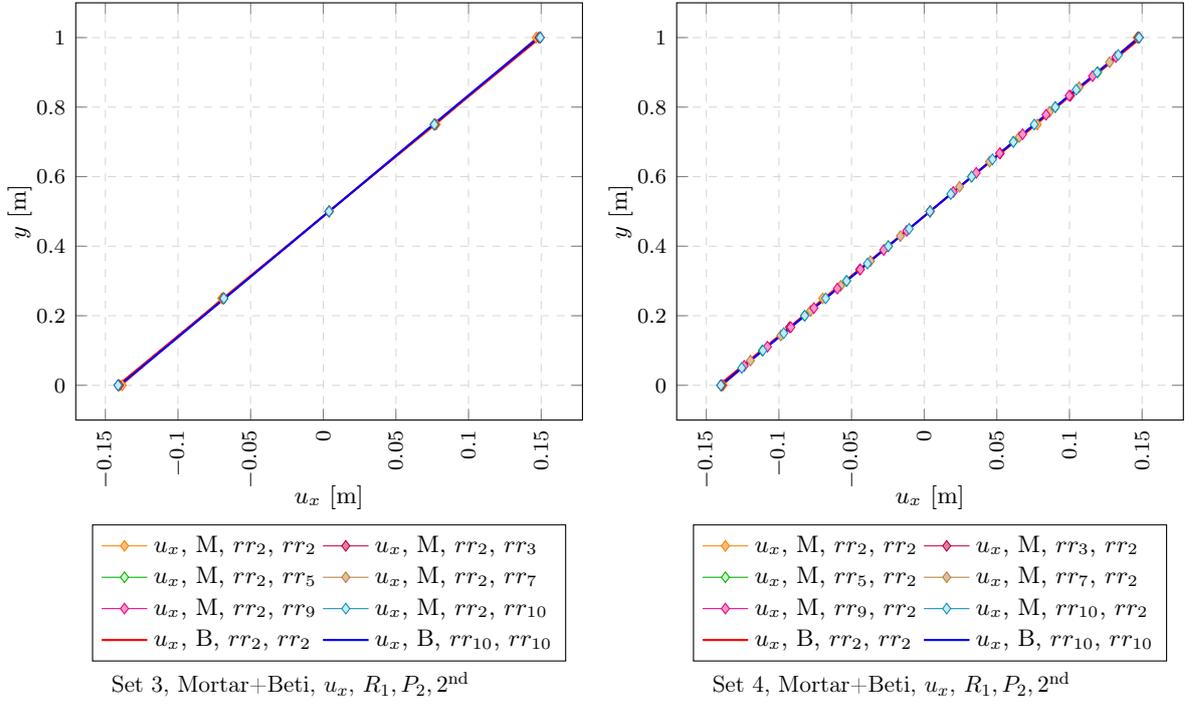


Figure 4.19: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

Displacements u_y , interface, reg. 1, part. 2

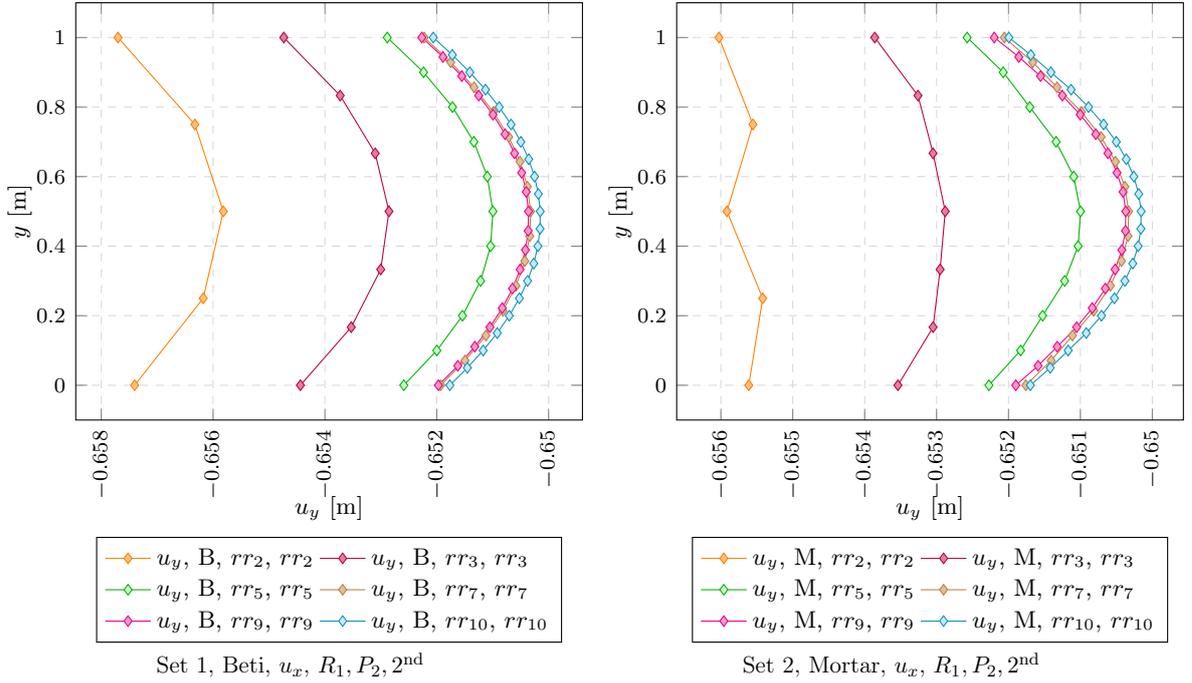


Figure 4.20: Convergence behaviour of *BETI* and *Mortar Method* coupling

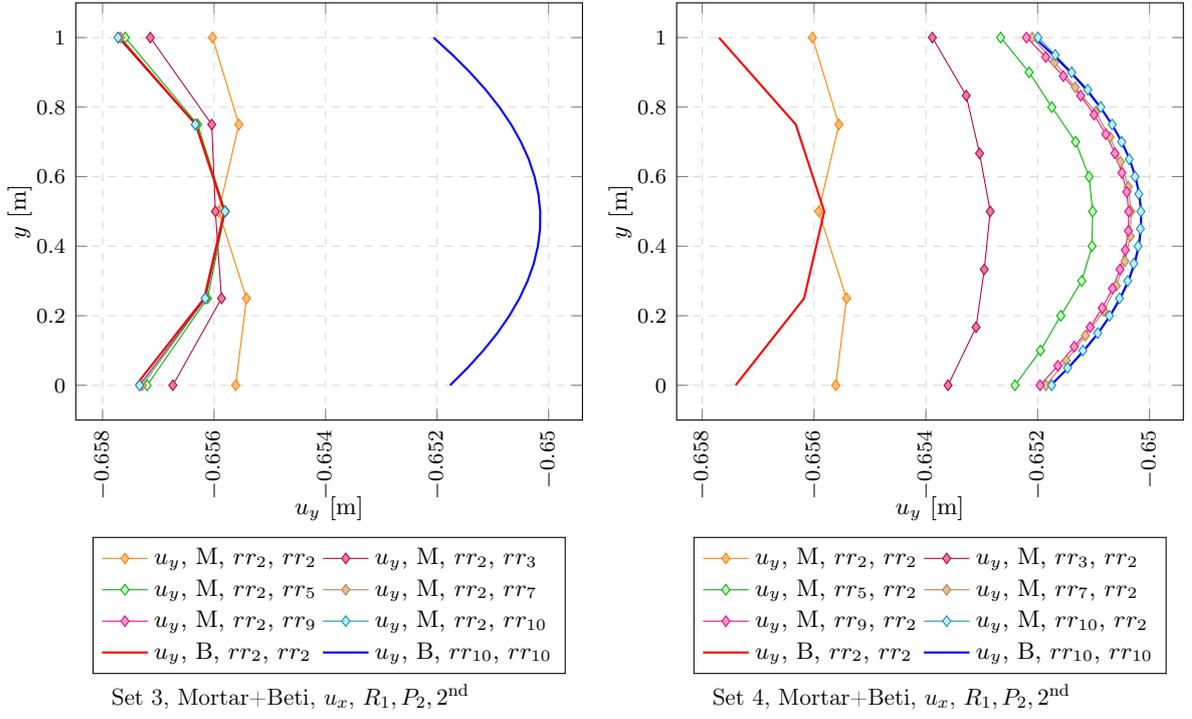


Figure 4.21: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

Displacements u_x , interface, reg. 2, part. 4

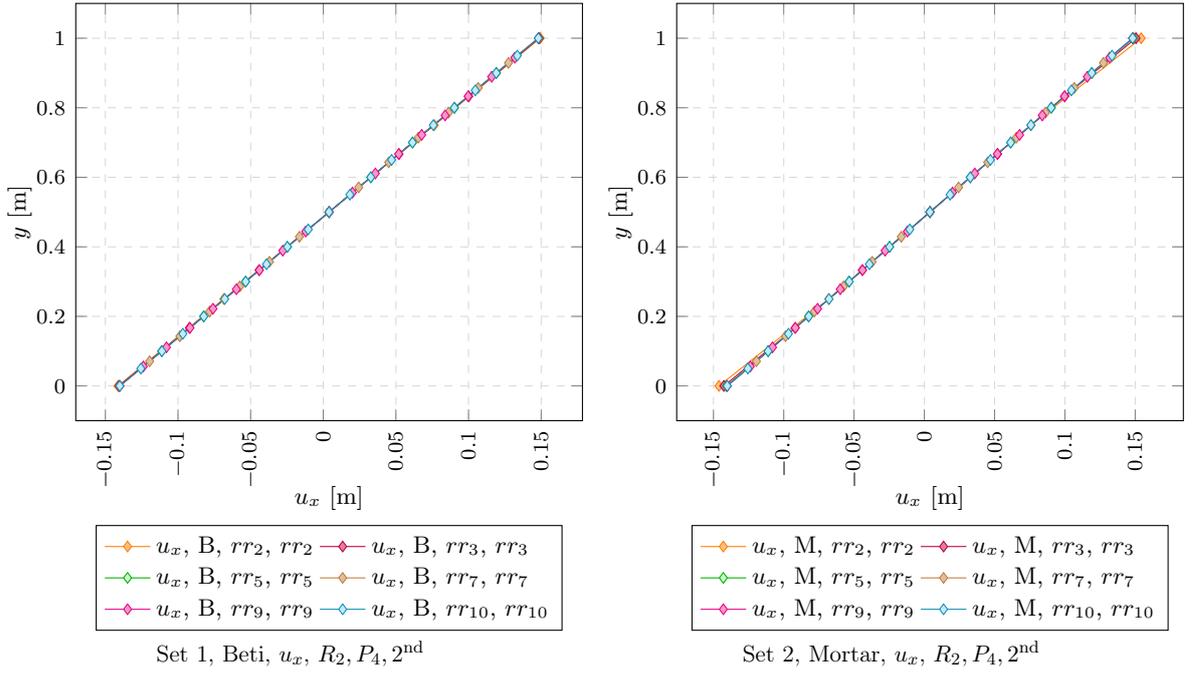


Figure 4.22: Convergence behaviour of *BETI* and *Mortar Method* coupling

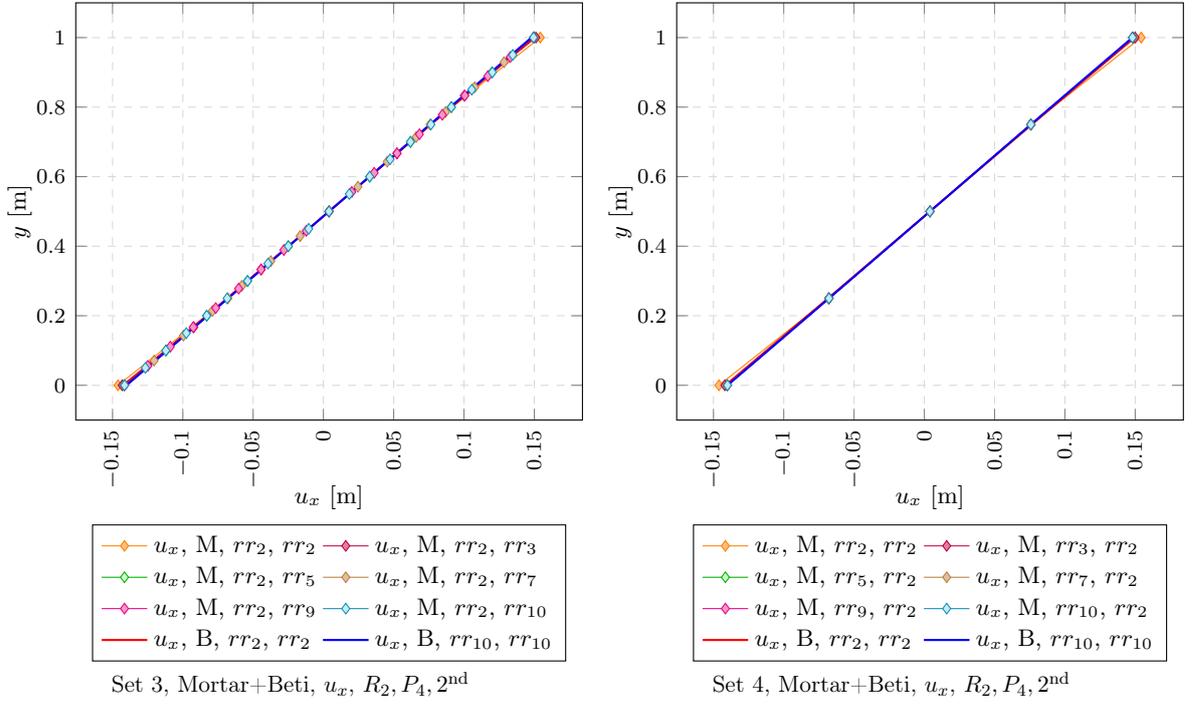


Figure 4.23: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

Displacements u_y , interface, reg. 2, part. 4

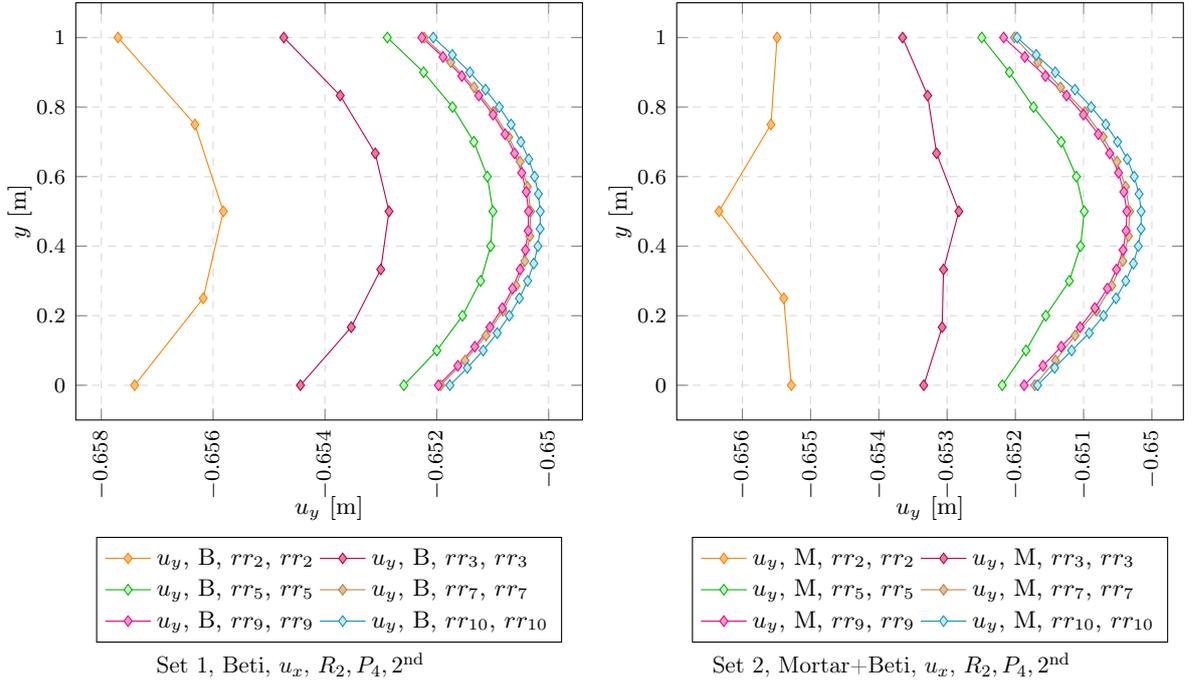


Figure 4.24: Convergence behaviour of *BETI* and *Mortar Method* coupling

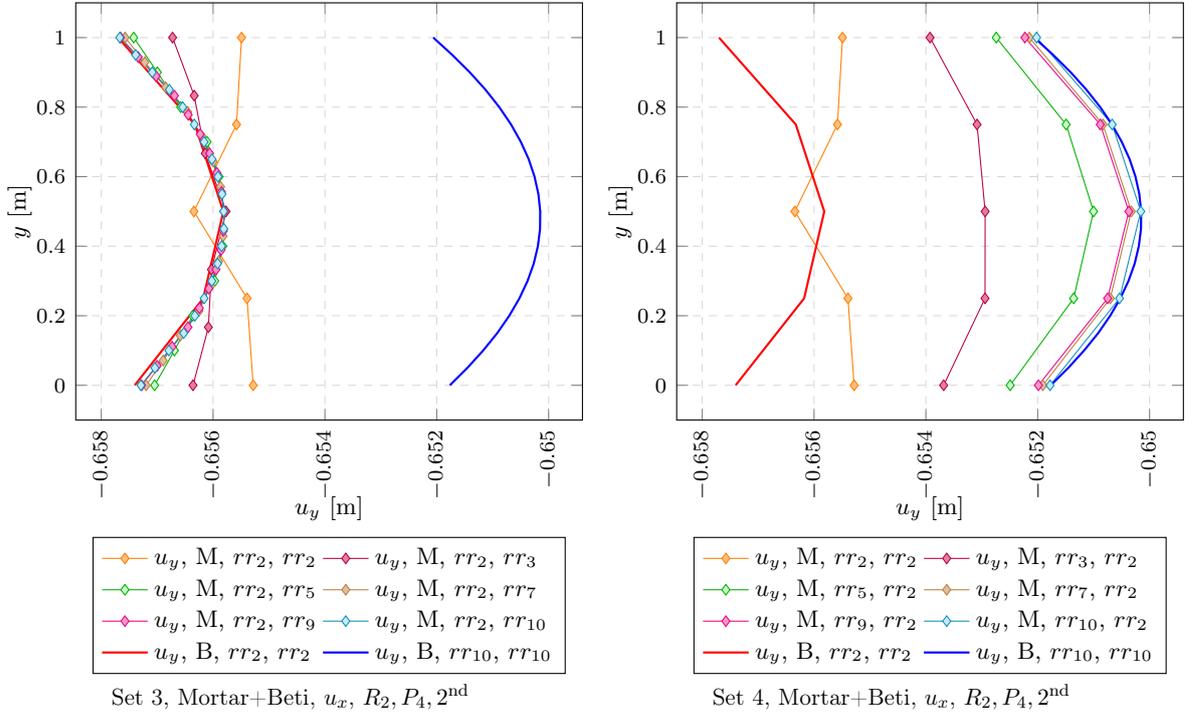


Figure 4.25: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

Displacements u_x , bottom edge, reg. 1+2, part. 1

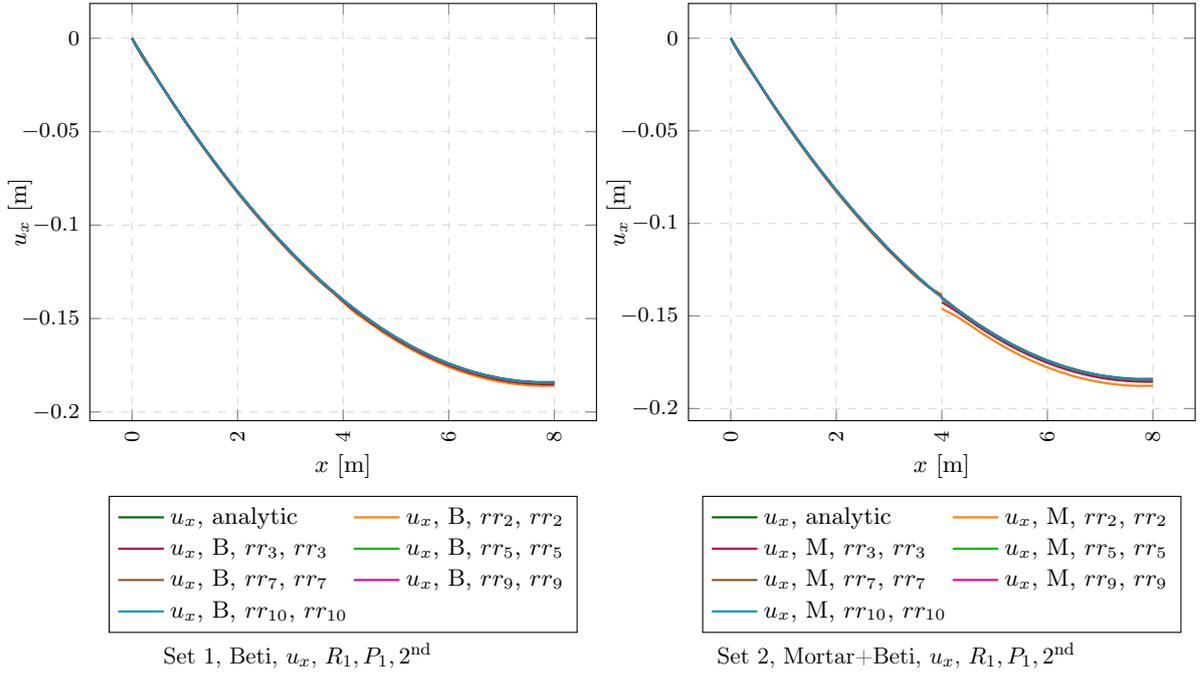


Figure 4.26: Convergence behaviour of *BETI* and *Mortar Method* coupling

Displacements u_x					
Coupling	rr	v_l	v_r	$\Delta v = v_r - v_l$	v_e
Mortar	2	-0.13846800	-0.14611700	-0.00764000	-0.18770700
Mortar	3	-0.13920500	-0.14261500	-0.00340000	-0.18554300
Mortar	5	-0.13972000	-0.14090000	-0.00117000	-0.18467000
Mortar	7	-0.13981800	-0.14044200	-0.00063000	-0.18442700
Mortar	9	-0.13998200	-0.14027700	-0.00029000	-0.18443000
Mortar	10	-0.13994500	-0.14021400	-0.00027000	-0.18436400
Beti	2	-0.14130400	-0.14130400	0.00000000	-0.18601900
Beti	10	-0.14007000	-0.14007000	0.00000000	-0.18435400

$v_l, v_r \dots$ value at the *left, right* side of the interface
 $v_e \dots$ value at beam end, $rr \dots$ refinement ratio

Table 4.11: Set 2, disagreement of displacements at the interface

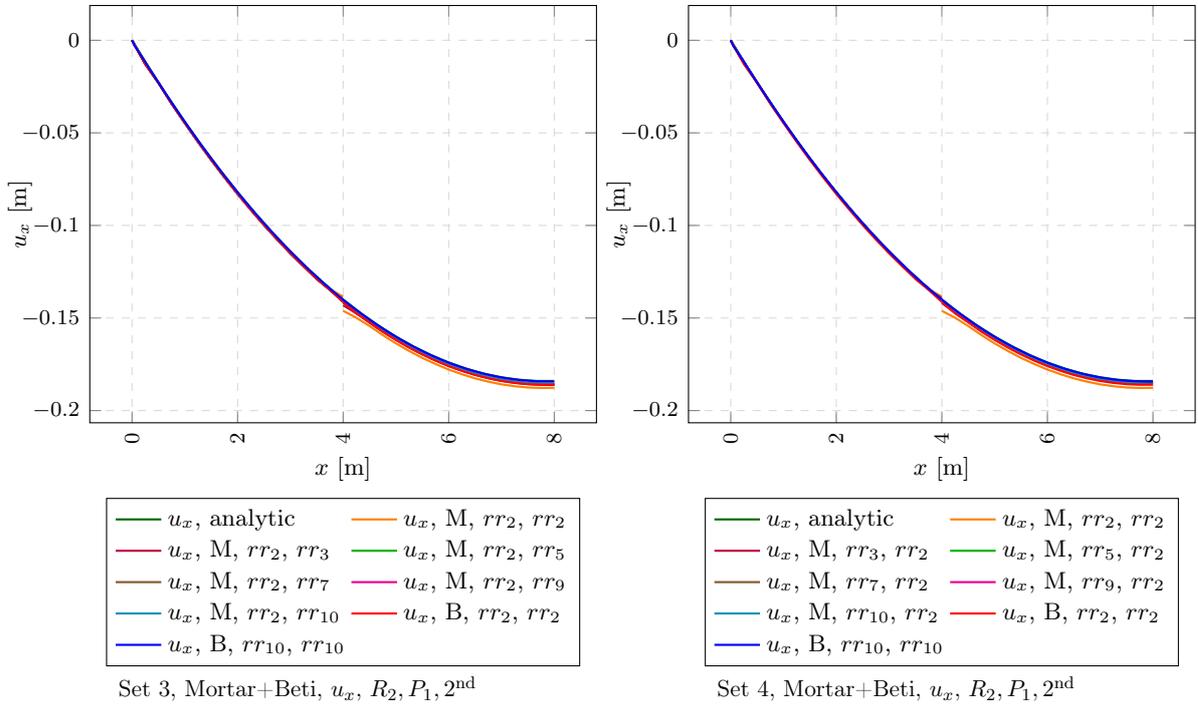


Figure 4.27: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

Displacements u_x					
Coupling	rr	v_l	v_r	$\Delta v = v_r - v_l$	v_e
Mortar	2	-0.13846800	-0.14611700	-0.00764000	-0.18770700
Mortar	3	-0.13924100	-0.14190800	-0.00267000	-0.18562800
Mortar	5	-0.13982800	-0.14038300	-0.00055000	-0.18498100
Mortar	7	-0.13987500	-0.14012300	-0.00024000	-0.18480200
Mortar	9	-0.14000500	-0.14010500	-0.00010000	-0.18480600
Mortar	10	-0.13996000	-0.14004800	-0.00009000	-0.18475700
Beti	2	-0.14130400	-0.14130400	0.00000000	-0.18601900
Beti	10	-0.14007000	-0.14007000	0.00000000	-0.18435400

$v_l, v_r \dots$ value at the *left, right* side of the interface
 $v_e \dots$ value at beam end, $rr \dots$ refinement ratio

Table 4.12: Set 4, disagreement of displacements at the interface

Displacements u_y , bottom edge, reg. 1+2, part. 1

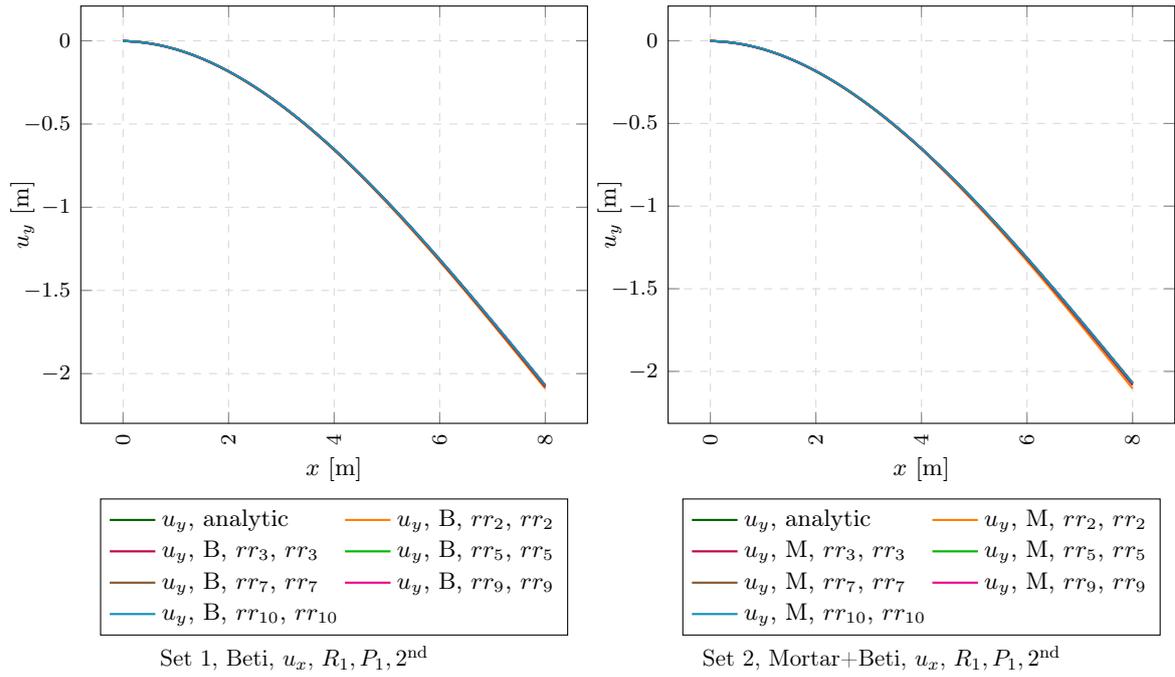


Figure 4.28: Convergence behaviour of *BETI* and *Mortar Method* coupling

Displacements u_y					
Coupling	rr	v_l	v_r	$\Delta v = v_r - v_l$	v_e
Mortar	2	-0.65561200	-0.65528300	0.00032000	-2.10400000
Mortar	3	-0.65353800	-0.65334200	0.00020000	-2.08257000
Mortar	5	-0.65227200	-0.65219200	0.00008000	-2.07361000
Mortar	7	-0.65175900	-0.65171500	0.00005000	-2.07096000
Mortar	9	-0.65189800	-0.65187100	0.00003000	-2.07098000
Mortar	10	-0.65169600	-0.65167400	0.00003000	-2.07024000
Beti	2	-0.65740300	-0.65740300	0.00000000	-2.08979000
Beti	10	-0.65176600	-0.65176600	0.00000000	-2.07016000

$v_l, v_r \dots$ value at the *left, right* side of the interface
 $v_e \dots$ value at beam end, $rr \dots$ refinement ratio

Table 4.13: Set 2, disagreement of displacements at the interface

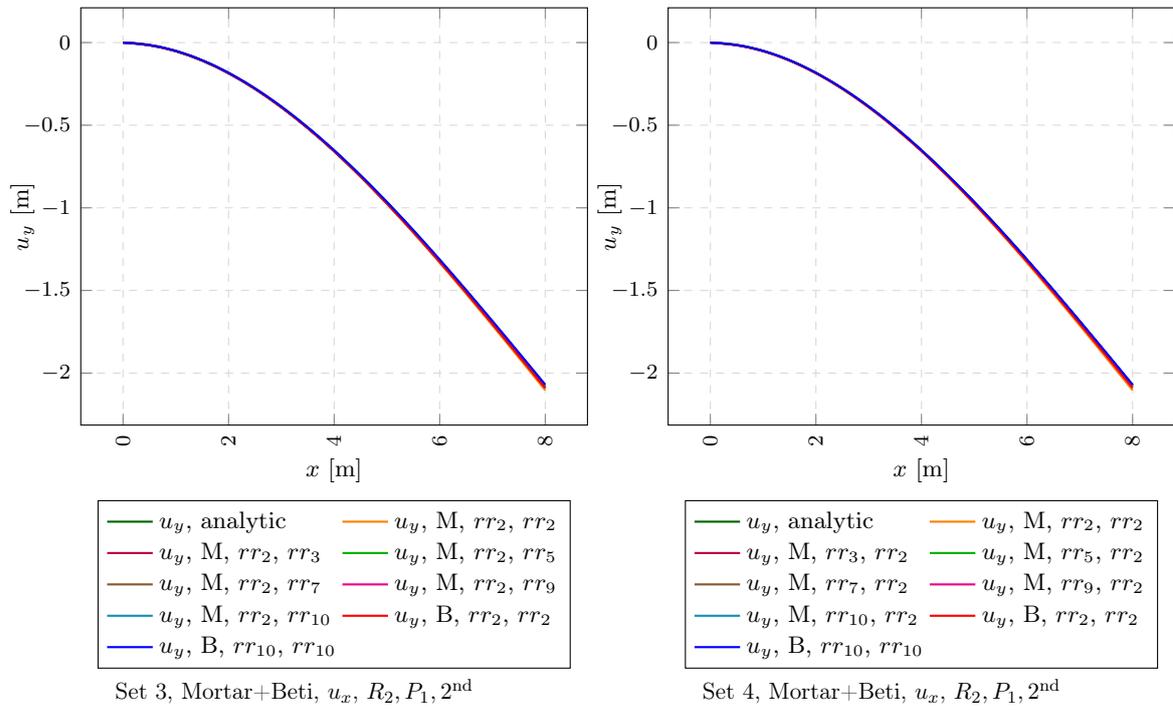


Figure 4.29: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

Displacements u_y					
Coupling	rr	v_l	v_r	$\Delta v = v_r - v_l$	v_e
Mortar	2	-0.65561200	-0.65528300	0.00032000	-2.10400000
Mortar	3	-0.65360400	-0.65368200	-0.00008000	-2.08353000
Mortar	5	-0.65240500	-0.65249200	-0.00009000	-2.07650000
Mortar	7	-0.65185300	-0.65190800	-0.00005000	-2.07446000
Mortar	9	-0.65195700	-0.65198700	-0.00003000	-2.07455000
Mortar	10	-0.65175000	-0.65177900	-0.00003000	-2.07394000
Beti	2	-0.65740300	-0.65740300	0.00000000	-2.08979000
Beti	10	-0.65176600	-0.65176600	0.00000000	-2.07016000

$v_l, v_r \dots$ value at the *left, right* side of the interface
 $v_e \dots$ value at beam end, $rr \dots$ refinement ratio

Table 4.14: Set 4, disagreement of displacements at the interface

4.5 Convergence study for the 3D cantilever beam

This section contains the graphical and tabulated results of the numerical analysis for the 3D cantilever beam.

4.5.1 System geometry

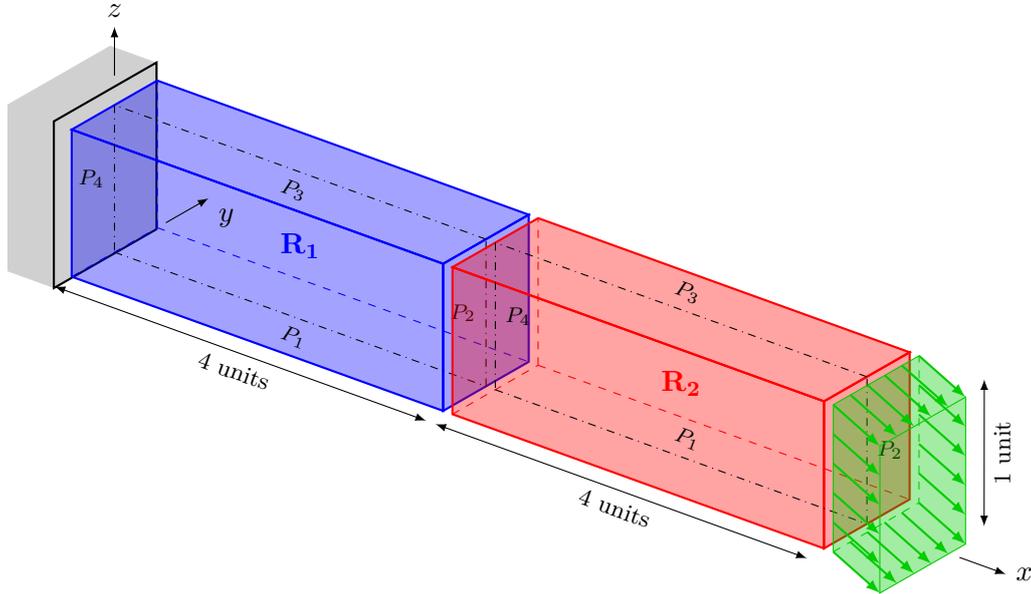


Figure 4.30: 3D cantilever with 2 regions

4.5.2 Interface displacements using first order shape functions

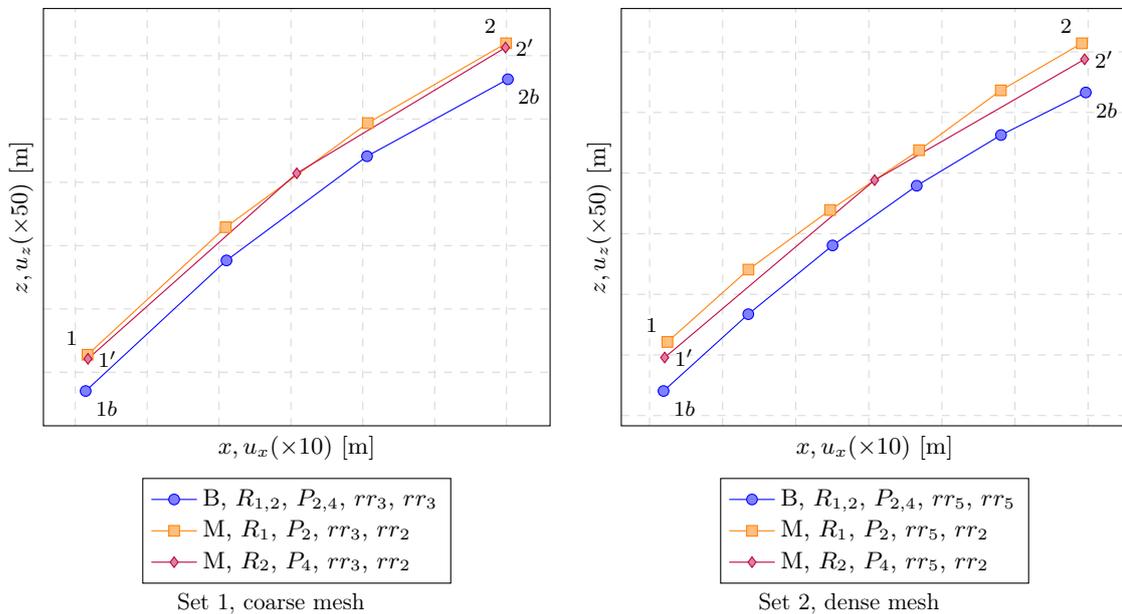


Figure 4.31: Displaced interface configurations

In Set 1 and 2 we can see the gaps and overlaps of the both regions at the interface when using the *Mortar Method*. The blue line marks the displaced figure of a *BETI* coupled system which has an offset to the other lines. In this case a higher refinement ratio in one region does not always mean that the gaps and overlaps at the interface become smaller. In some cases the convergence starts at a certain refinement ratio which is equivalent to the number of *Lagrange multipliers*.

Displacement, coarse mesh						
\mathbf{u}	\mathbf{u}_1	$\mathbf{u}_{1'}$	\mathbf{u}_{1b}	$\Delta\mathbf{u}_{1'-1}$	$\Delta\mathbf{u}_{1-1b}$	$\Delta\mathbf{u}_{1'-1b}$
u_x	-0.14152000	-0.14121700	-0.14281600	0.00030000	0.00130000	0.00160000
u_z	-0.65889300	-0.65915200	-0.66118500	-0.00026000	0.00229000	0.00203000
\mathbf{u}	\mathbf{u}_2	$\mathbf{u}_{2'}$	\mathbf{u}_{2b}	$\Delta\mathbf{u}_{2'-2}$	$\Delta\mathbf{u}_{2-2b}$	$\Delta\mathbf{u}_{2'-2b}$
u_x	0.14972000	0.14931400	0.15096800	-0.00041000	-0.00125000	-0.00166000
u_z	-0.65889300	-0.65948800	-0.66149200	-0.00026000	0.00227000	0.00201000

Table 4.15: Set 1, displacement values

Displacement, dense mesh						
\mathbf{u}	\mathbf{u}_1	$\mathbf{u}_{1'}$	\mathbf{u}_{1b}	$\Delta\mathbf{u}_{1'-1}$	$\Delta\mathbf{u}_{1-1b}$	$\Delta\mathbf{u}_{1'-1b}$
u_x	-0.13778200	-0.13966000	-0.14043900	-0.00188000	0.00266000	0.00078000
u_z	-0.64714300	-0.64817200	-0.65038000	-0.00104000	0.00323000	0.00220000
\mathbf{u}	\mathbf{u}_2	$\mathbf{u}_{2'}$	\mathbf{u}_{2b}	$\Delta\mathbf{u}_{2'-2}$	$\Delta\mathbf{u}_{2-2b}$	$\Delta\mathbf{u}_{2'-2b}$
u_x	0.14578200	0.14764800	0.14844600	0.00186000	-0.00267000	-0.00081000
u_z	-0.64714300	-0.64849100	-0.65068100	-0.00107000	0.00325000	0.00218000

Table 4.16: Set 2, displacement values

The tables show that the maximum displacement disagreement of top and bottom interface points is about 10^{-3} .

4.5.3 Interface displacements using second order shape functions

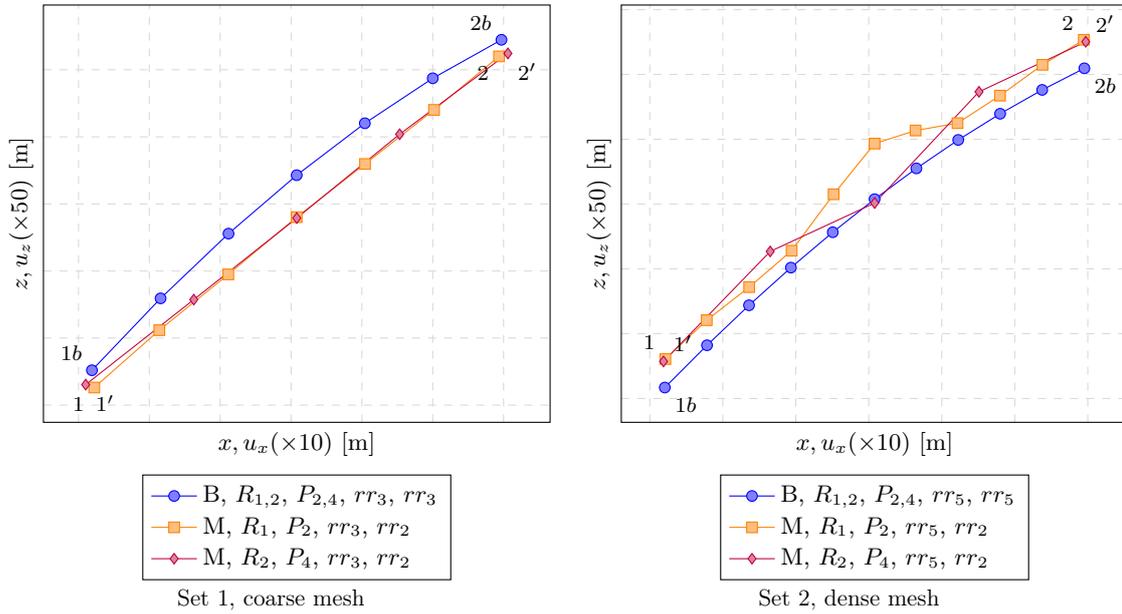


Figure 4.32: Displaced interface configurations

Set 1 shows almost perfect conforming displacement figures of the interface. There is just a small offset to the blue line of the *BETI* coupled system. In Set 2 the top and bottom points of the interface have a small offset where the displaced figures of the interface of both regions can have slightly bigger overlaps and gaps. This is due to the geometry configuration of the opposite elements at the interface.

Displacement, coarse mesh						
\mathbf{u}	\mathbf{u}_1	$\mathbf{u}_{1'}$	\mathbf{u}_{1b}	$\Delta \mathbf{u}_{1'-1}$	$\Delta \mathbf{u}_{1-1b}$	$\Delta \mathbf{u}_{1'-1b}$
u_x	-0.13888900	-0.14493500	-0.14051200	-0.00604000	0.00163000	-0.00441000
u_z	-0.65095100	-0.65077800	-0.64992000	0.00018000	-0.00104000	-0.00085000
\mathbf{u}	\mathbf{u}_2	$\mathbf{u}_{2'}$	\mathbf{u}_{2b}	$\Delta \mathbf{u}_{2'-2}$	$\Delta \mathbf{u}_{2-2b}$	$\Delta \mathbf{u}_{2'-2b}$
u_x	0.14679800	0.15303500	0.14848900	0.00623000	-0.00168000	0.00455000
u_z	-0.65095100	-0.65102500	-0.65021500	0.00017000	-0.00099000	-0.00082000

Table 4.17: Set 1, displacement values

Displacement, dense mesh						
\mathbf{u}	\mathbf{u}_1	$\mathbf{u}_{1'}$	\mathbf{u}_{1b}	$\Delta \mathbf{u}_{1'-1}$	$\Delta \mathbf{u}_{1-1b}$	$\Delta \mathbf{u}_{1'-1b}$
u_x	-0.13933100	-0.14060100	-0.13971900	-0.00127000	0.00040000	-0.00087000
u_z	-0.64556700	-0.64571200	-0.64732400	-0.00014000	0.00175000	0.00162000
\mathbf{u}	\mathbf{u}_2	$\mathbf{u}_{2'}$	\mathbf{u}_{2b}	$\Delta \mathbf{u}_{2'-2}$	$\Delta \mathbf{u}_{2-2b}$	$\Delta \mathbf{u}_{2'-2b}$
u_x	0.14729200	0.14862700	0.14767000	0.00133000	-0.00038000	0.00095000
u_z	-0.64556700	-0.64598300	-0.64762100	-0.00012000	0.00175000	0.00163000

Table 4.18: Set 2, displacement values

The tables show that the maximum displacement disagreement of top and bottom interface points is about 10^{-3} .

4.5.4 Discretisation with first order shape functions

- Displacements u_x at the right side of region 1, partition 2 (interface)
- Displacements u_z at the right side of region 1, partition 2 (interface)
- Displacements u_x at the left side of region 2, partition 4 (interface)
- Displacements u_z at the left side of region 2, partition 4 (interface)
- Displacements u_x at the bottom edge of region 1+2, both regions partition 1
- Displacements u_z at the bottom edge of region 1+2, both regions partition 1

Displacements u_x , interface, reg. 1, part. 2

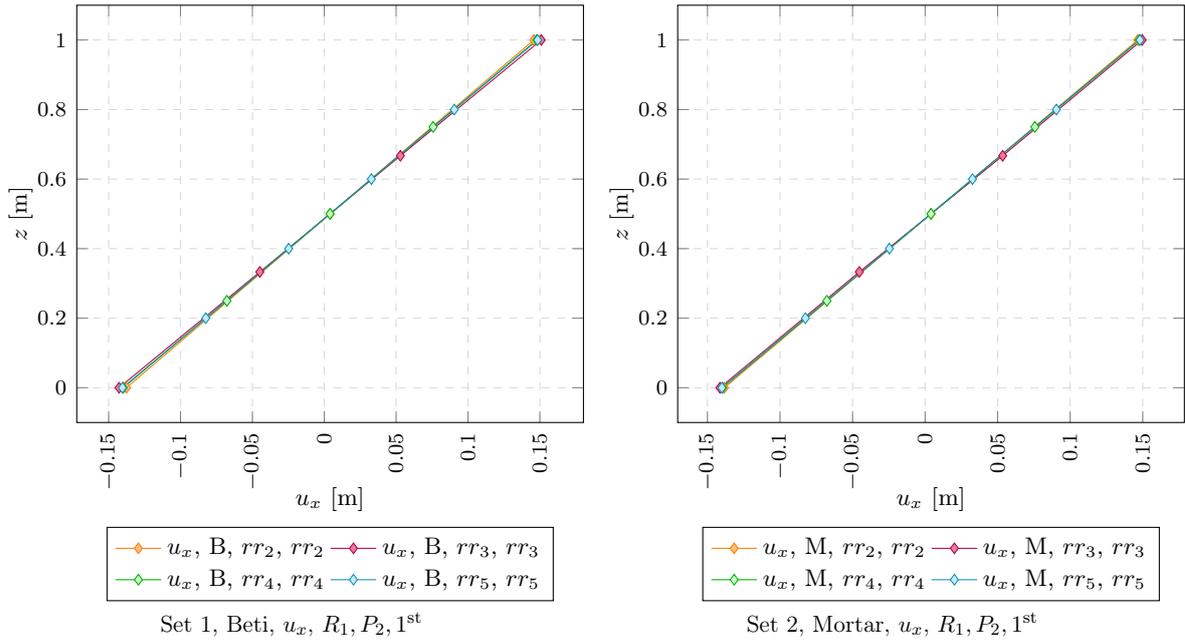


Figure 4.33: Convergence behaviour of *BETI* and *Mortar Method* coupling

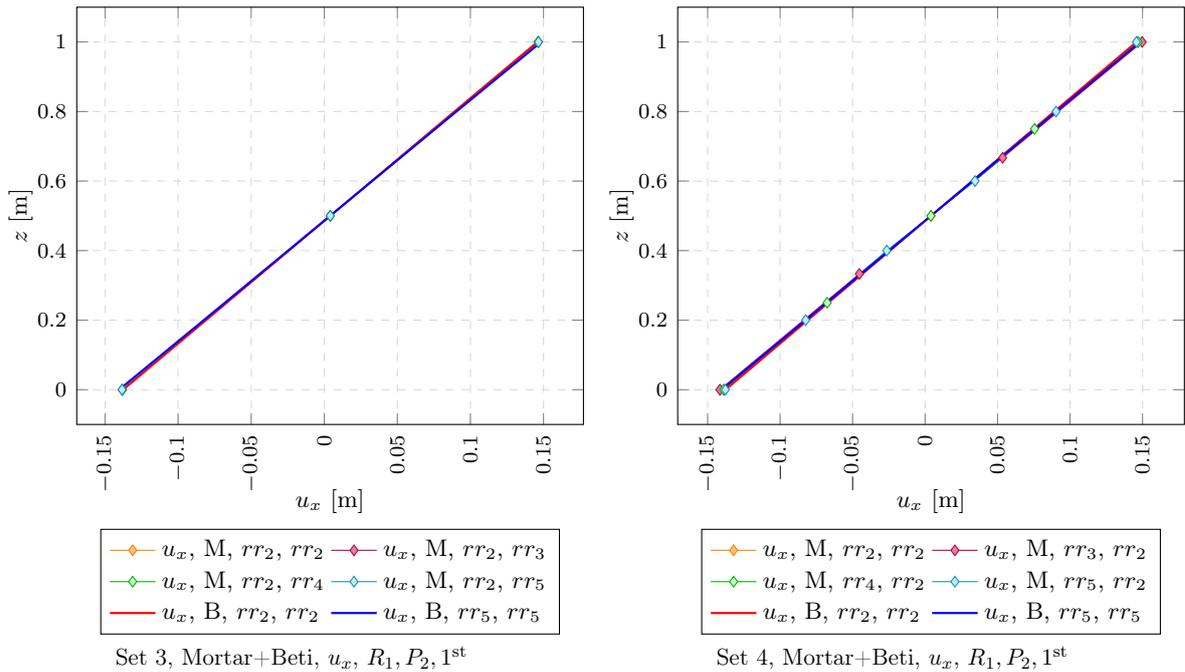


Figure 4.34: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

In all sets above the displacements in x-direction show good convergence, even for the coarse meshed region. The *Mortar Method* coupled systems approach the solution of the *BETI* coupled system.

Displacements u_z , interface, reg. 1, part. 2

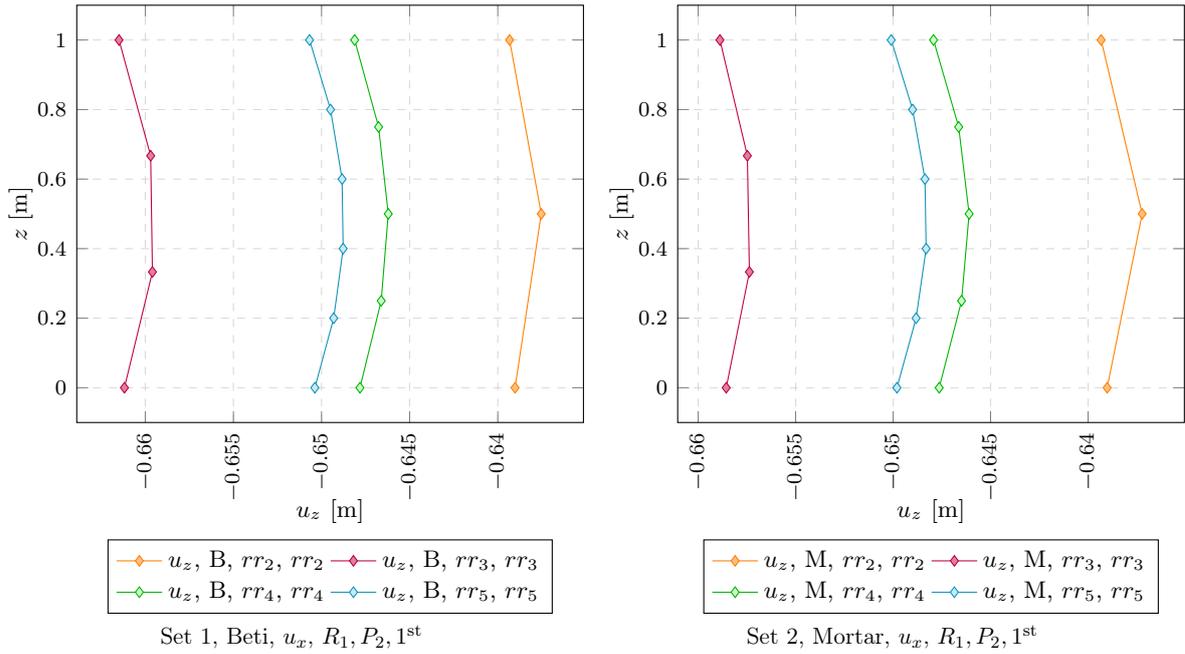


Figure 4.35: Convergence behaviour of *BETI* and *Mortar Method* coupling

The *BETI* coupled systems in Set 1 and the *Mortar Method* coupled systems in Set 2 show the same convergence behaviour. The purple line (refinement ratio of 3 elements per unit) should be situated between the orange and the green one. This non-monotonous convergence behaviour is due to the influence of the element geometry configuration along the width of the beam.

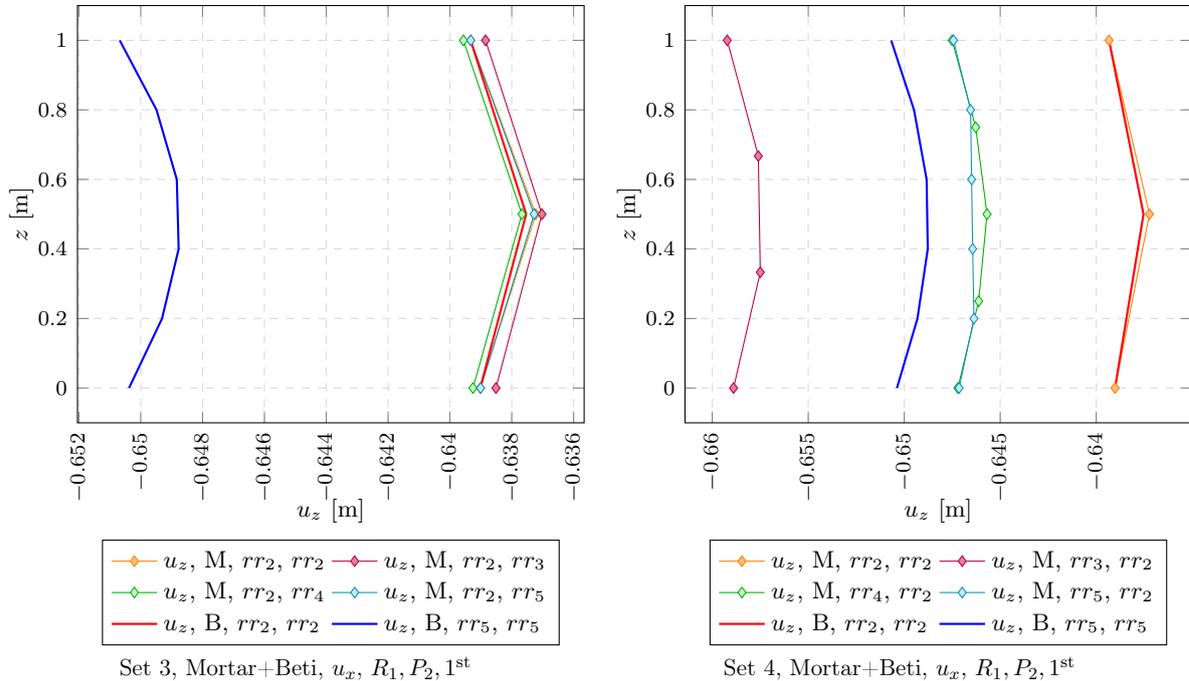


Figure 4.36: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

In Set 3 the left region has a coarse mesh. Here the displaced figures of all *Mortar Method* coupled systems approach the red line of the coarse meshed *BETI* coupled system. In Set 4 the right region has a mesh. The displaced figures of the *Mortar Method* coupled systems converge against the blue line of the dense meshed *BETI* coupled system. Again we see the non-monotonous convergence behaviour which is explained above.

Displacements u_x , interface, reg. 2, part. 4

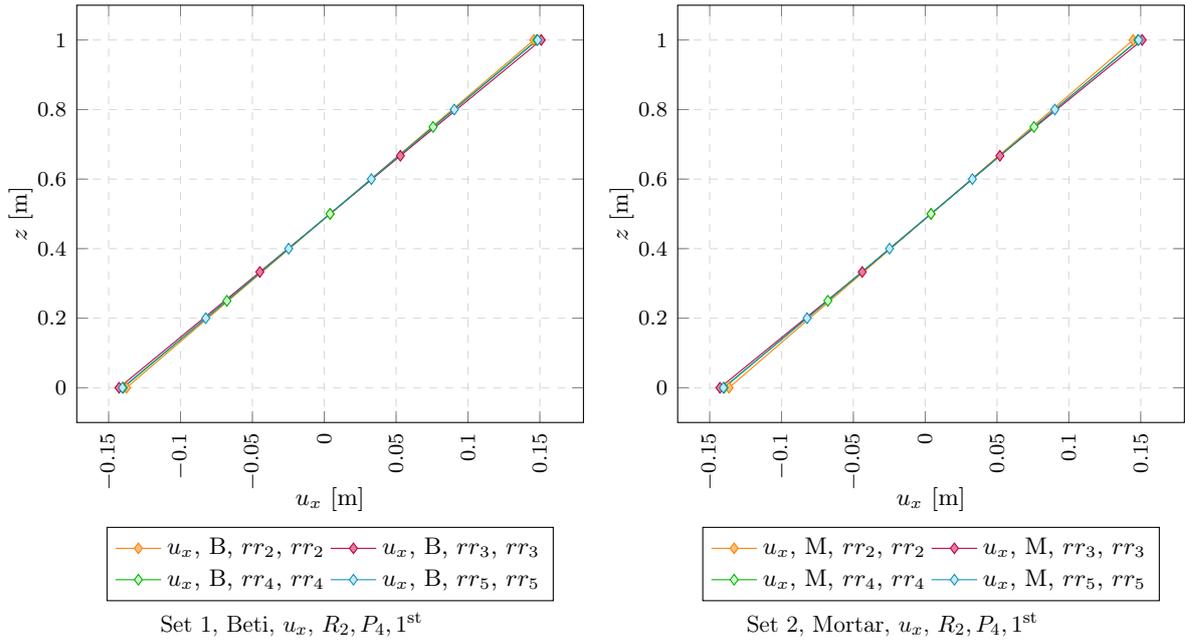


Figure 4.37: Convergence behaviour of *BETI* and *Mortar Method* coupling

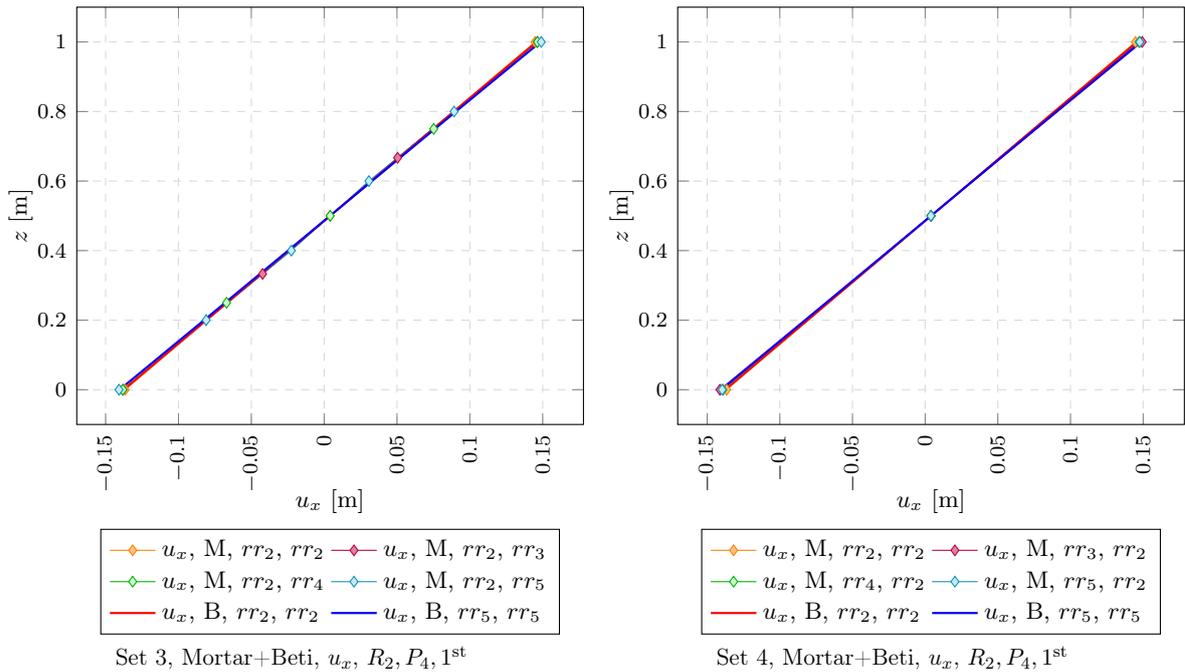


Figure 4.38: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

In all sets we can see typical convergence behaviour. All systems converge against the blue line of the dense meshed *BETI* coupled system.

Displacements u_z , interface, reg. 2, part. 4

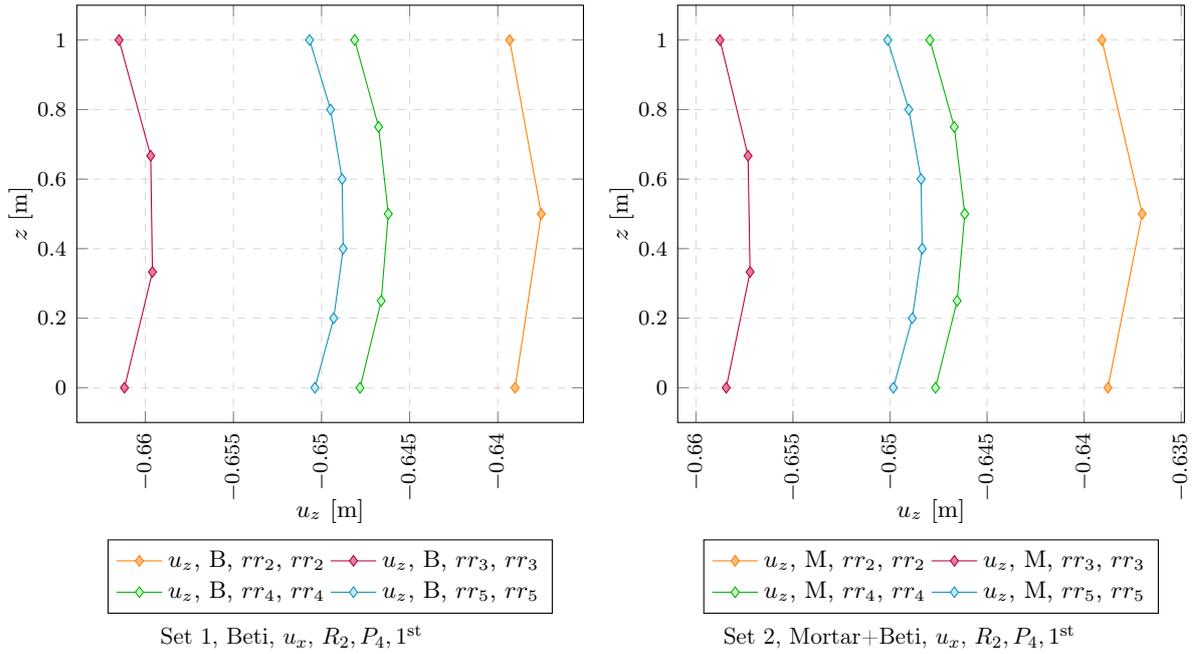


Figure 4.39: Convergence behaviour of *BETI* and *Mortar Method* coupling

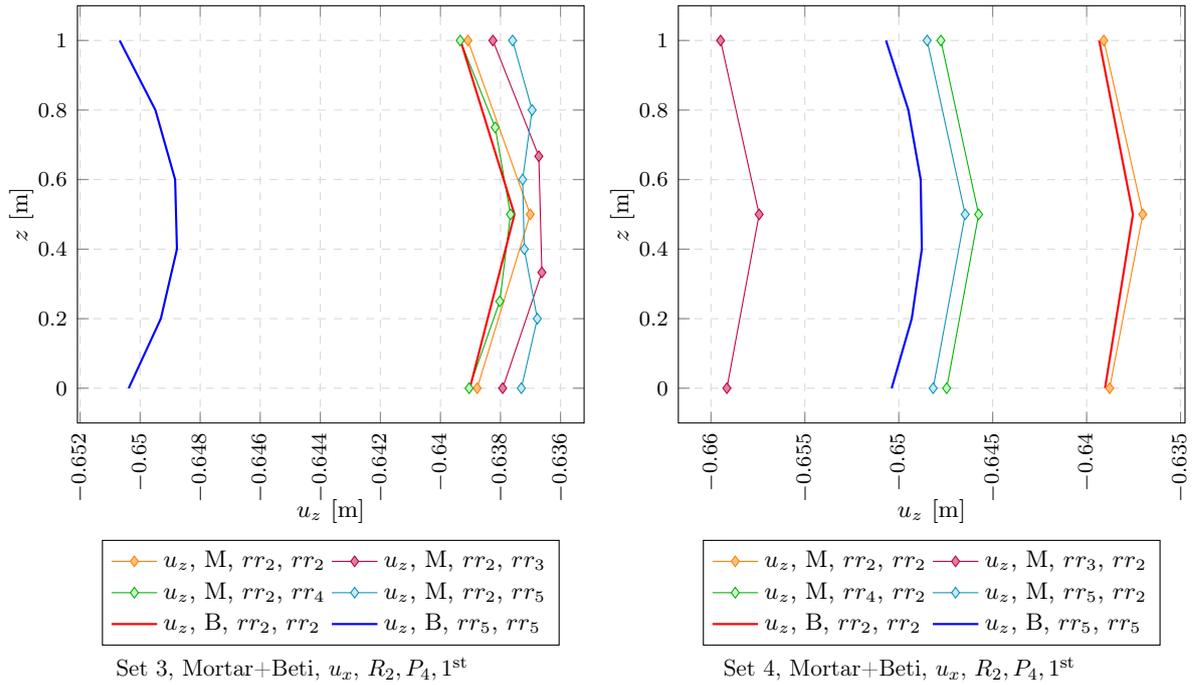


Figure 4.40: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

In Set 1, 2 and 4 we can see the non-monotonous convergence behaviour mentioned above.

Displacements u_x , bottom edge, reg. 1+2, part. 1

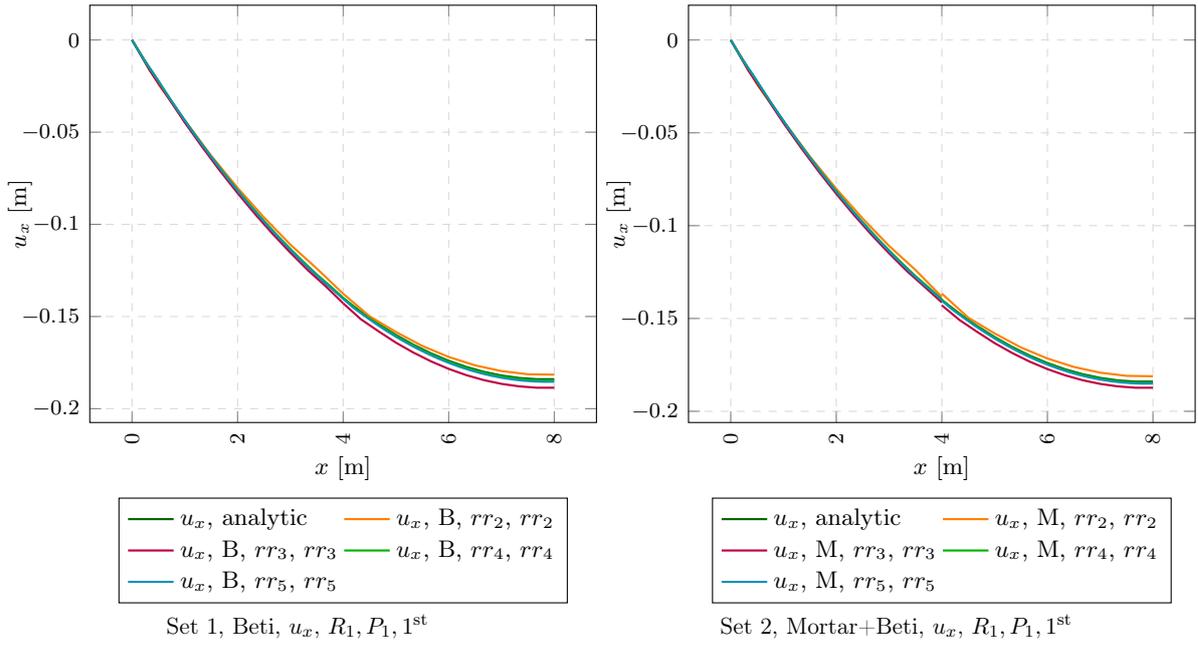


Figure 4.41: Convergence behaviour of *BETI* and *Mortar Method* coupling

Displacements u_x					
Coupling	rr	v_l	v_r	$\Delta v = v_r - v_l$	v_e
Mortar	2	-0.13836000	-0.13659200	0.00177000	-0.18114100
Mortar	3	-0.14140700	-0.14289900	-0.00150000	-0.18731200
Mortar	4	-0.13961800	-0.14002300	-0.00041000	-0.18458800
Mortar	5	-0.14005200	-0.14048200	-0.00044000	-0.18506800
Beti	2	-0.13762900	-0.13762900	0.00000000	-0.18155600
Beti	5	-0.14043900	-0.14043900	0.00000000	-0.18539000

$v_l, v_r \dots$ value at the *left, right* side of the interface
 $v_e \dots$ value at beam end, $rr \dots$ refinement ratio

Table 4.19: Set 2, disagreement of displacements at the interface

Both sets show typical convergence behaviour. At the interface of both regions we can see some small disagreement of displacements in x-direction.

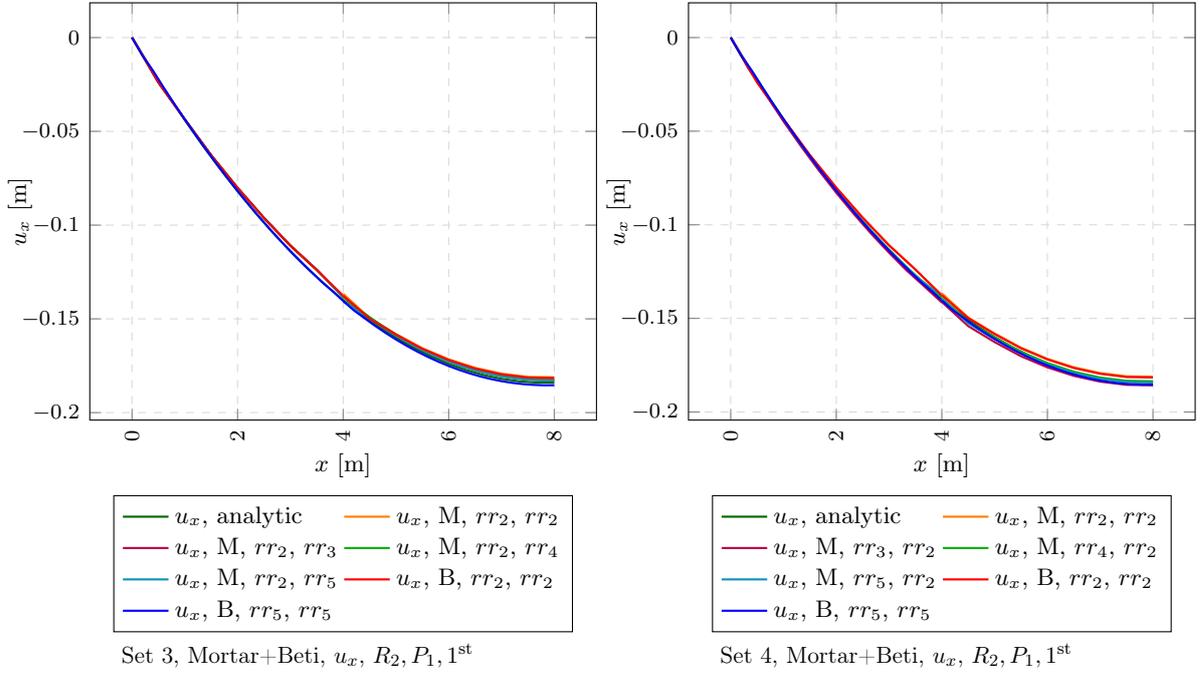


Figure 4.42: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

Displacements u_x					
Coupling	rr	v_l	v_r	$\Delta v = v_r - v_l$	v_e
Mortar	2	-0.13836000	-0.13659200	0.00177000	-0.18114100
Mortar	3	-0.14152000	-0.14121700	0.00030000	-0.18574000
Mortar	4	-0.13898300	-0.13921200	-0.00023000	-0.18356800
Mortar	5	-0.13778200	-0.13966000	-0.00188000	-0.18420900
Beti	2	-0.13762900	-0.13762900	0.00000000	-0.18155600
Beti	5	-0.14043900	-0.14043900	0.00000000	-0.18539000

$v_l, v_r \dots$ value at the *left, right* side of the interface
 $v_e \dots$ value at beam end, $rr \dots$ refinement ratio

Table 4.20: Set 4, disagreement of displacements at the interface

Even for unequal mesh refinement ratio's we get good convergence. The disagreement of displacements at the interface is still small.

Displacements u_z , bottom edge, reg. 1+2, part. 1

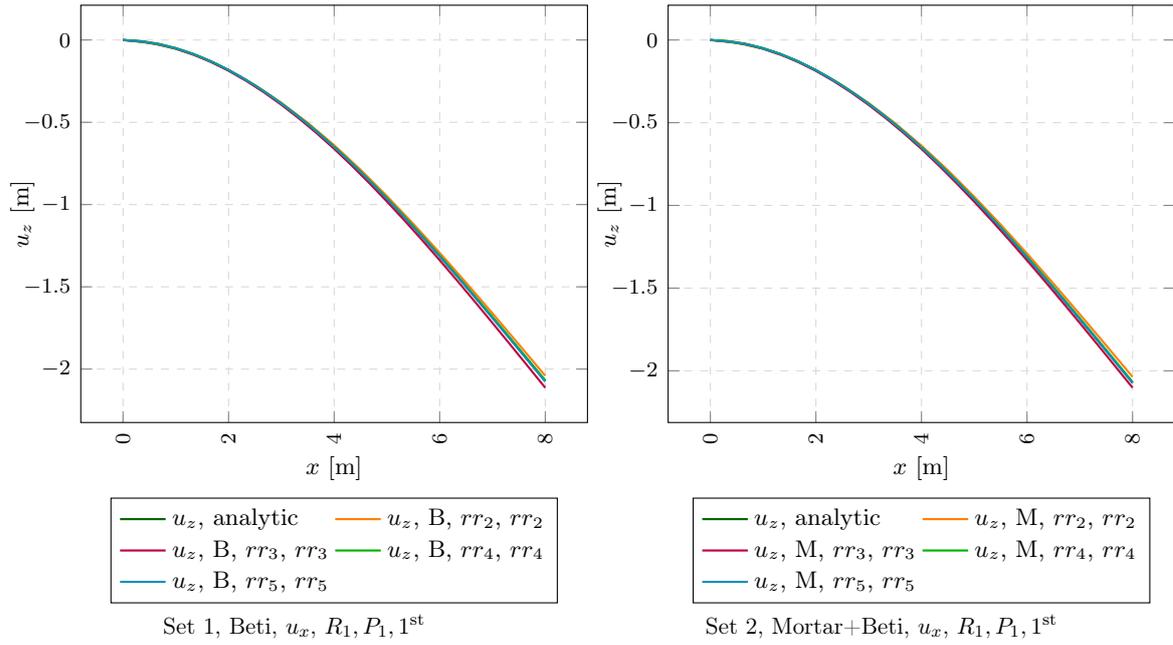


Figure 4.43: Convergence behaviour of *BETI* and *Mortar Method* coupling

Displacements u_z					
Coupling	rr	v_l	v_r	$\Delta v = v_r - v_l$	v_e
Mortar	2	-0.63902100	-0.63877600	0.00024000	-2.03687000
Mortar	3	-0.65856100	-0.65843900	0.00012000	-2.10237000
Mortar	4	-0.64763700	-0.64765800	-0.00002000	-2.06785000
Mortar	5	-0.64980400	-0.64982600	-0.00002000	-2.07348000
Beti	2	-0.63902100	-0.63902100	0.00000000	-2.04043000
Beti	5	-0.65038000	-0.65038000	0.00000000	-2.07609000
$v_l, v_r \dots$ value at the <i>left, right</i> side of the interface $v_e \dots$ value at beam end, $rr \dots$ refinement ratio					

Table 4.21: Set 2, disagreement of displacements at the interface

Set 1 and 2 show the same good convergence behaviour. The disagreement of displacements at the interface does not seem to have any impact. Even coarse meshes show good convergence.

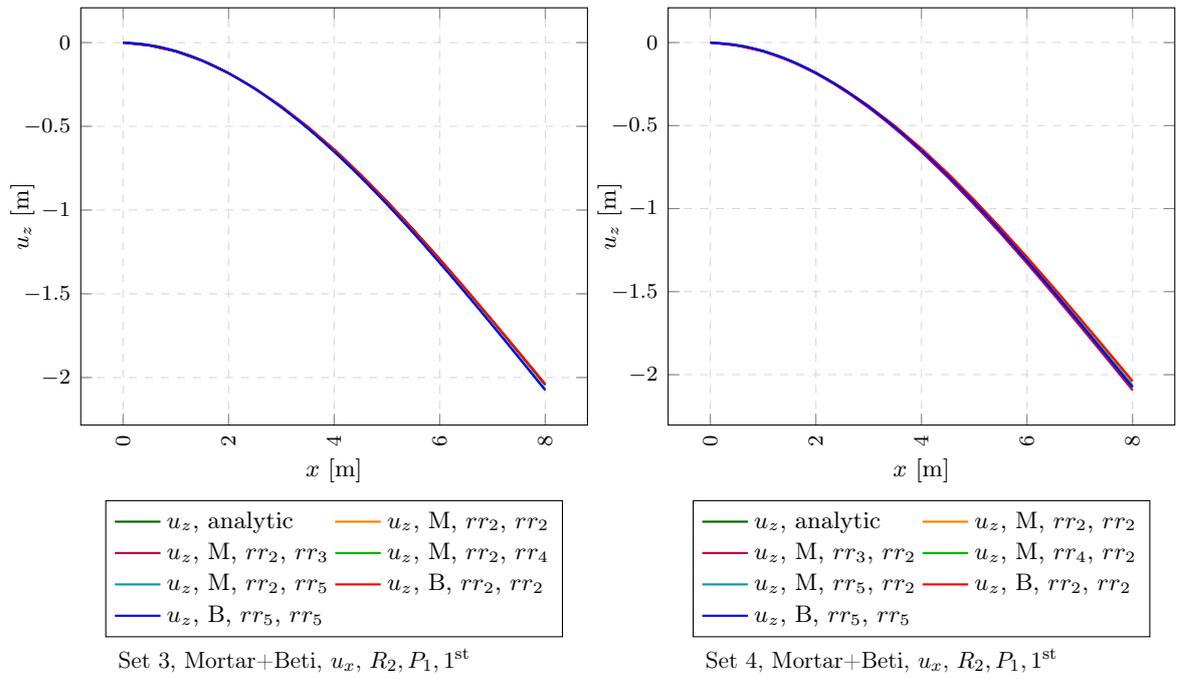


Figure 4.44: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

Displacements u_z					
Coupling	rr	v_l	v_r	$\Delta v = v_r - v_l$	v_e
Mortar	2	-0.63902100	-0.63877600	0.00024000	-2.03687000
Mortar	3	-0.65889300	-0.65915200	-0.00026000	-2.09405000
Mortar	4	-0.64720800	-0.64745200	-0.00024000	-2.06383000
Mortar	5	-0.64714300	-0.64817200	-0.00104000	-2.07021000
Beti	2	-0.63902100	-0.63902100	0.00000000	-2.04043000
Beti	5	-0.65038000	-0.65038000	0.00000000	-2.07609000

$v_l, v_r \dots$ value at the *left, right* side of the interface
 $v_e \dots$ value at beam end, $rr \dots$ refinement ratio

Table 4.22: Set 4, disagreement of displacements at the interface

Unequal mesh refinement ratio's also lead to good convergence behaviour. Again there is no influence of the interface to see.

4.5.5 Discretisation with second order shape functions

In this section all systems from the previous section are shown again. But now the elements are discretised with 2nd order shape functions. There are no extra explanations given. This is for the reason that the convergence behaviour is same like before but with higher convergence ratio.

- Displacements u_x at the right side of region 1, partition 2 (interface)
- Displacements u_z at the right side of region 1, partition 2 (interface)
- Displacements u_x at the left side of region 2, partition 4 (interface)
- Displacements u_z at the left side of region 2, partition 4 (interface)
- Displacements u_x at the bottom edge of region 1+2, both regions partition 1
- Displacements u_z at the bottom edge of region 1+2, both regions partition 1

Displacements u_x , interface, reg. 1, part. 2

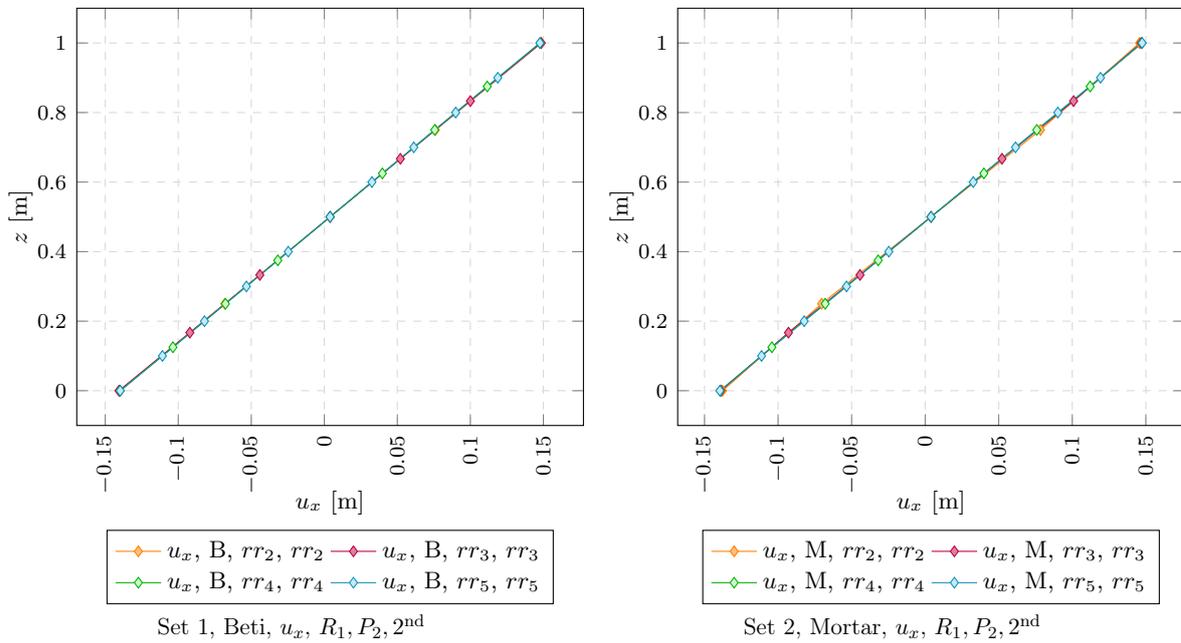


Figure 4.45: Convergence behaviour of *BETI* and *Mortar Method* coupling

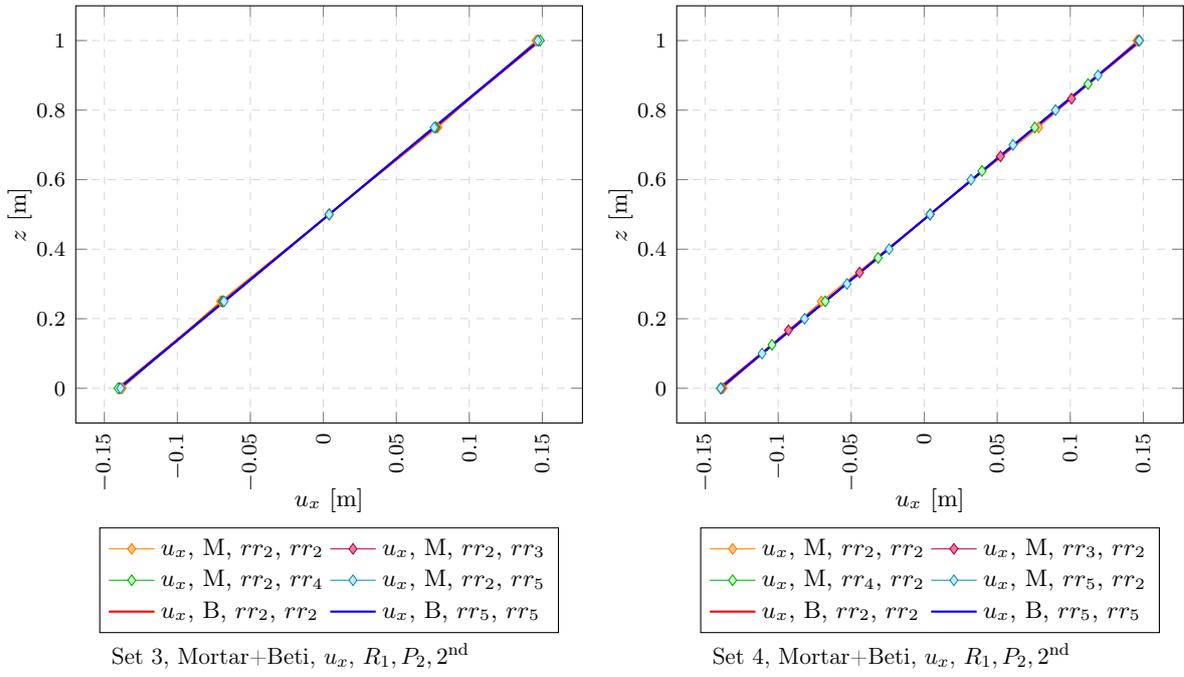


Figure 4.46: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

Displacements u_z , interface, reg. 1, part. 2

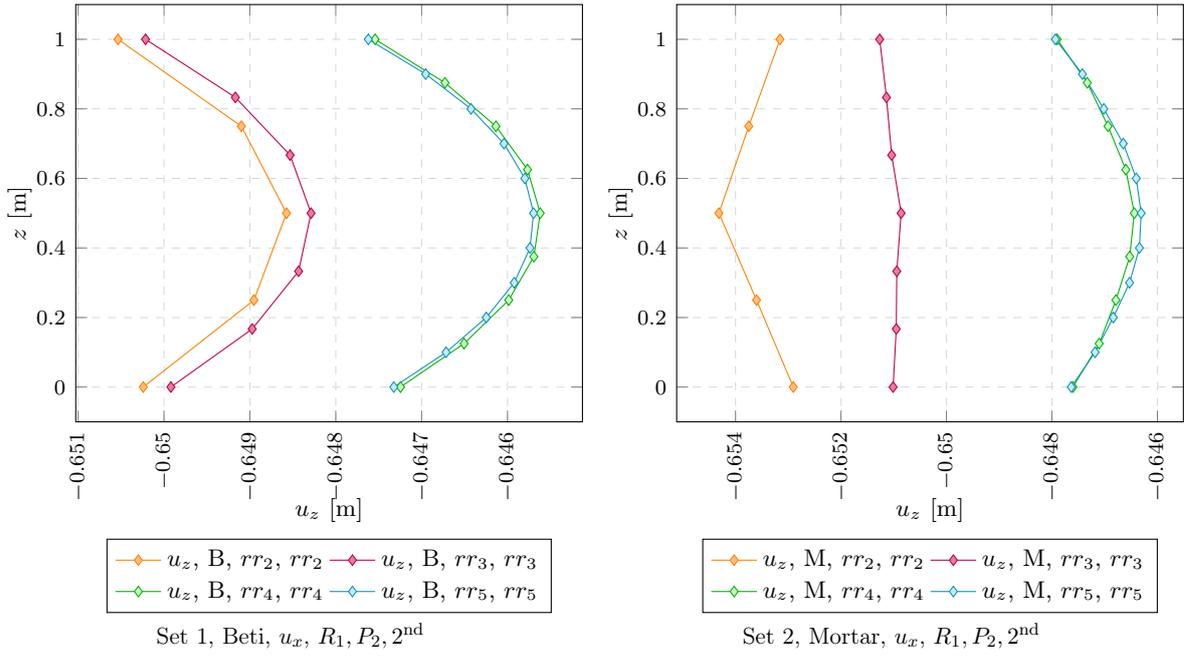


Figure 4.47: Convergence behaviour of *BETI* and *Mortar Method* coupling

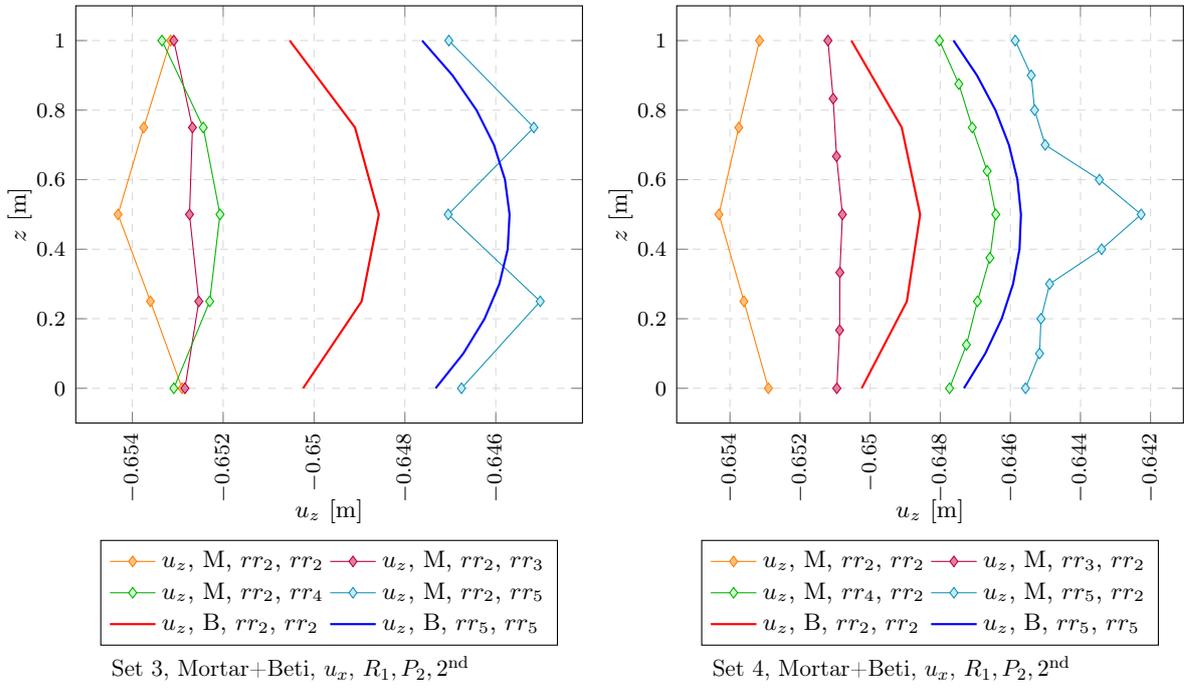


Figure 4.48: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

Displacements u_x , interface, reg. 2, part. 4

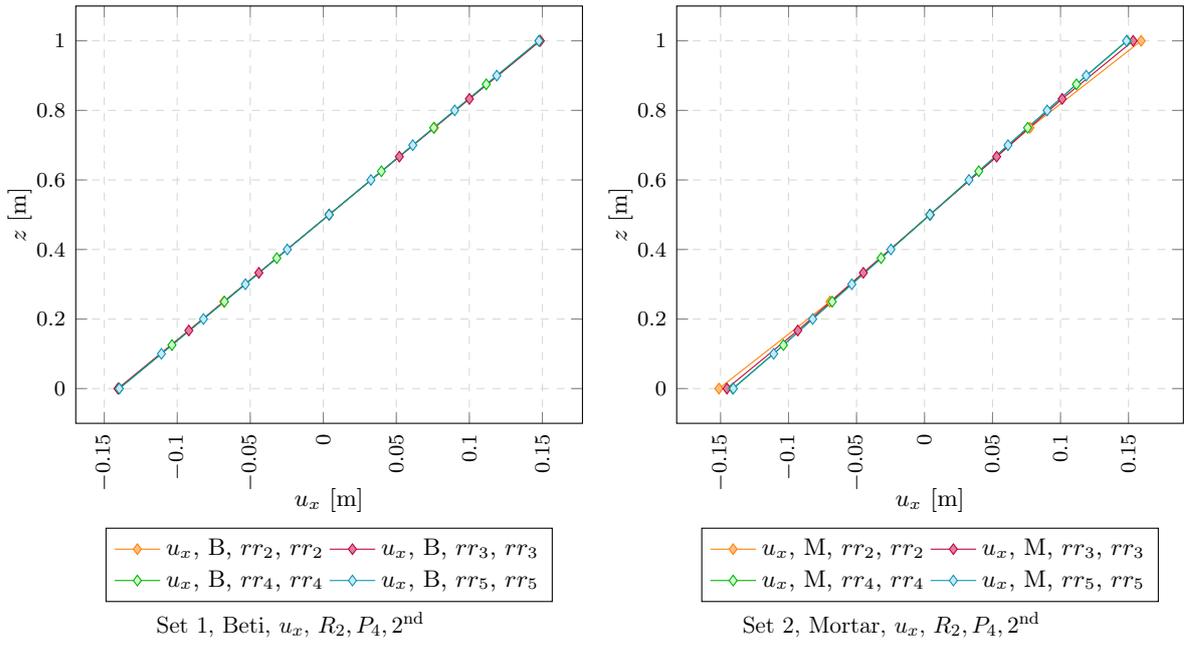


Figure 4.49: Convergence behaviour of *BETI* and *Mortar Method* coupling

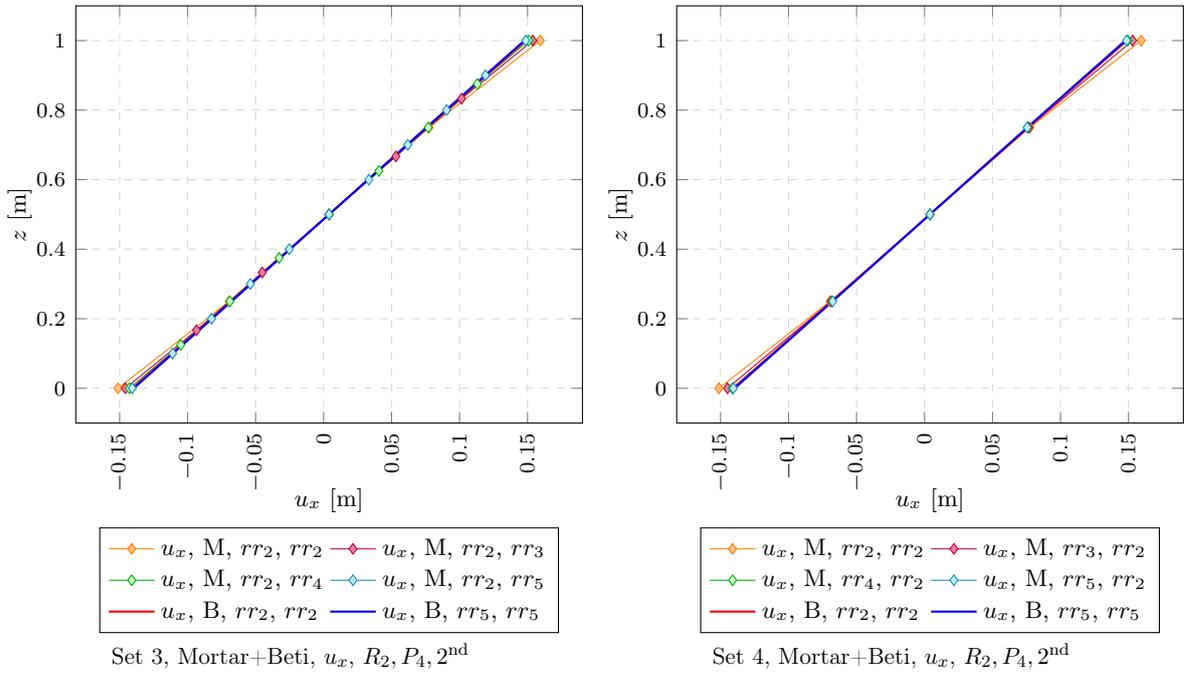


Figure 4.50: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

Displacements u_z , interface, reg. 2, part. 4

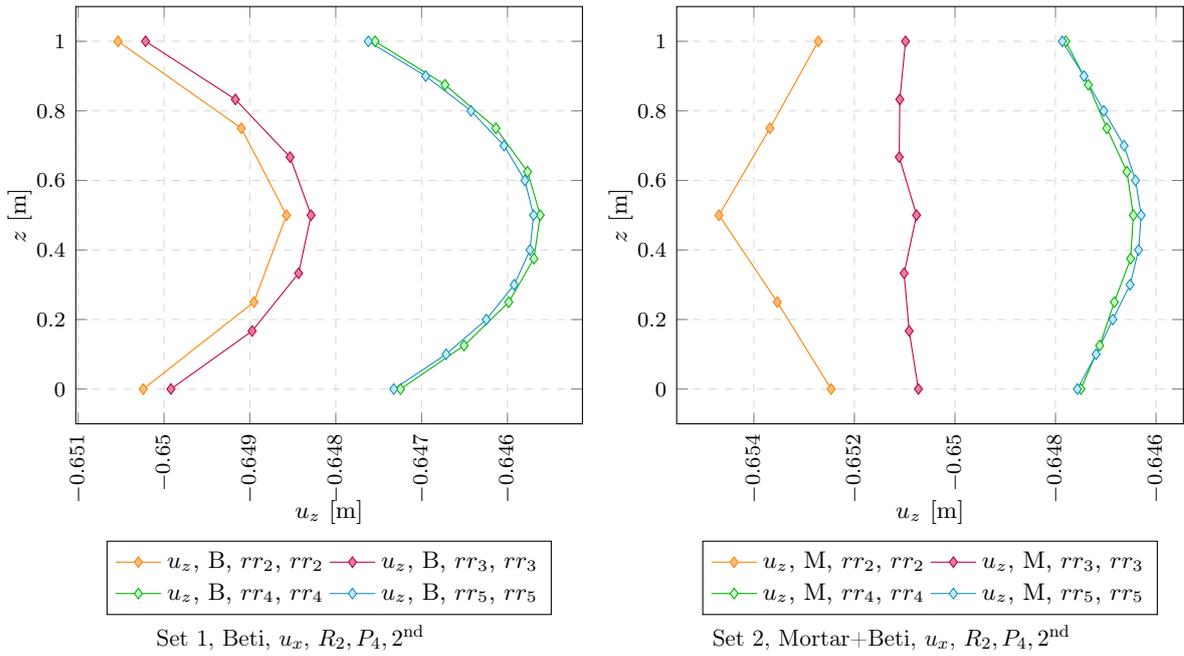


Figure 4.51: Convergence behaviour of *BETI* and *Mortar Method* coupling

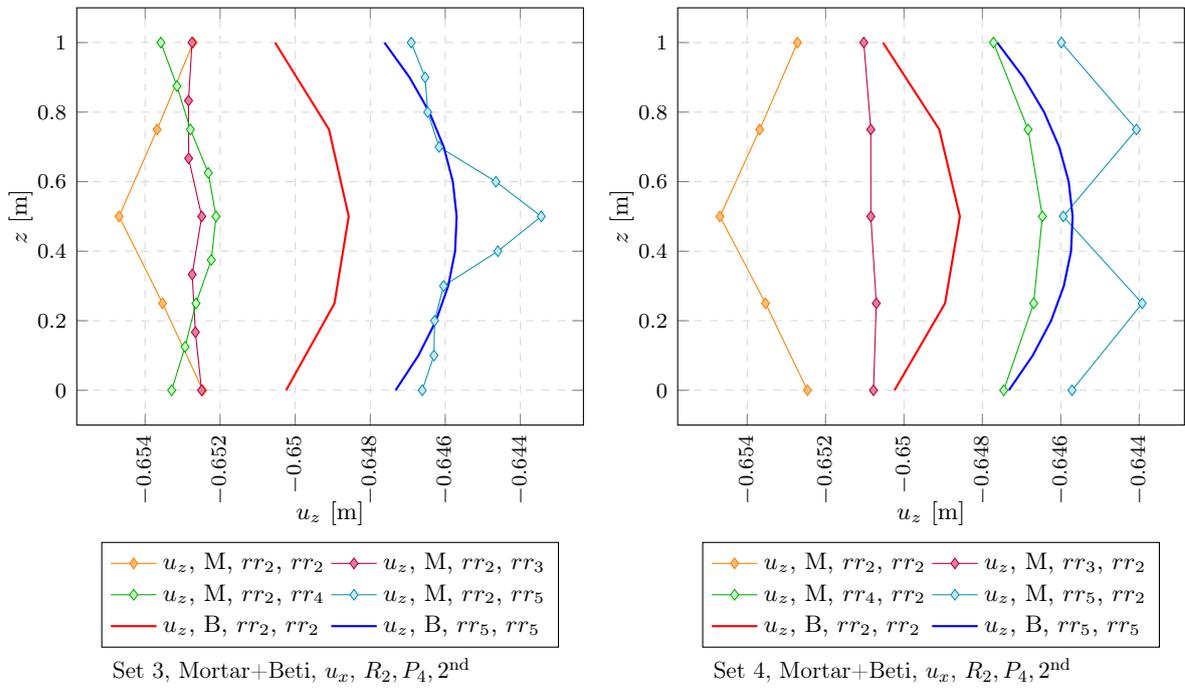


Figure 4.52: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

Displacements u_x , bottom edge, reg. 1+2, part. 1

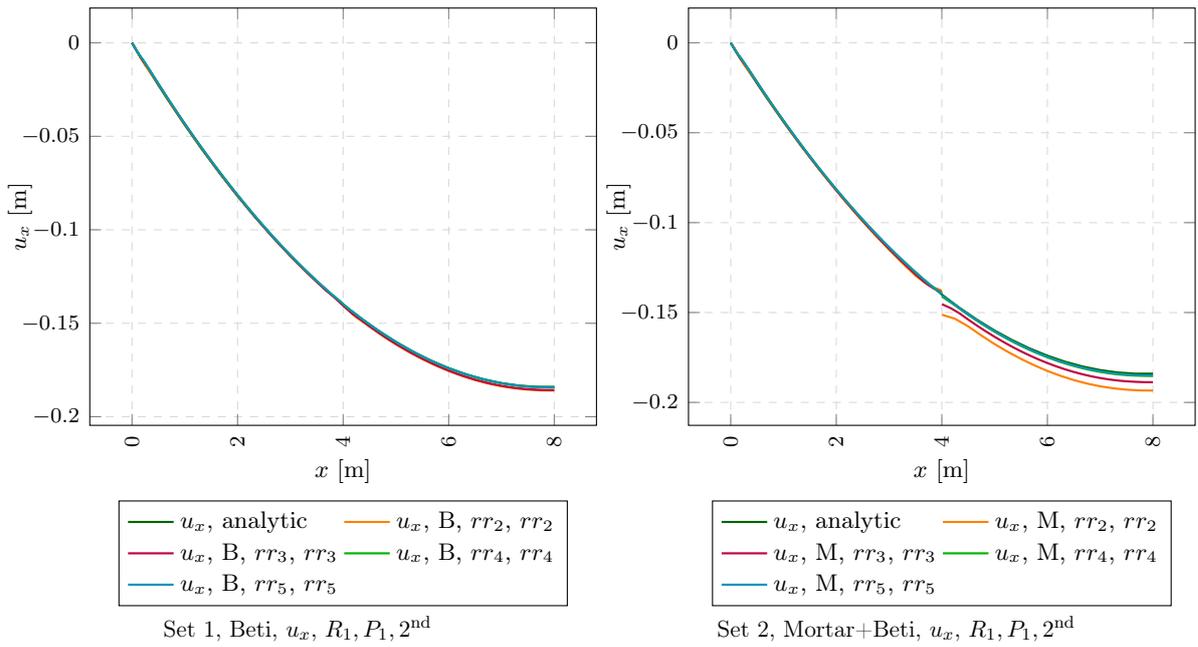


Figure 4.53: Convergence behaviour of *BETI* and *Mortar Method* coupling

Displacements u_x					
Coupling	rr	v_l	v_r	$\Delta v = v_r - v_l$	v_e
Mortar	2	-0.13779200	-0.15120300	-0.01341000	-0.19340800
Mortar	3	-0.13889700	-0.14534400	-0.00644000	-0.18875500
Mortar	4	-0.13936900	-0.14109000	-0.00171000	-0.18529400
Mortar	5	-0.13953100	-0.14059400	-0.00107000	-0.18493100
Beti	2	-0.14056000	-0.14056000	0.00000000	-0.18600500
Beti	5	-0.13971900	-0.13971900	0.00000000	-0.18429100

$v_l, v_r \dots$ value at the *left, right* side of the interface
 $v_e \dots$ value at beam end, $rr \dots$ refinement ratio

Table 4.23: Set 2, disagreement of displacements at the interface

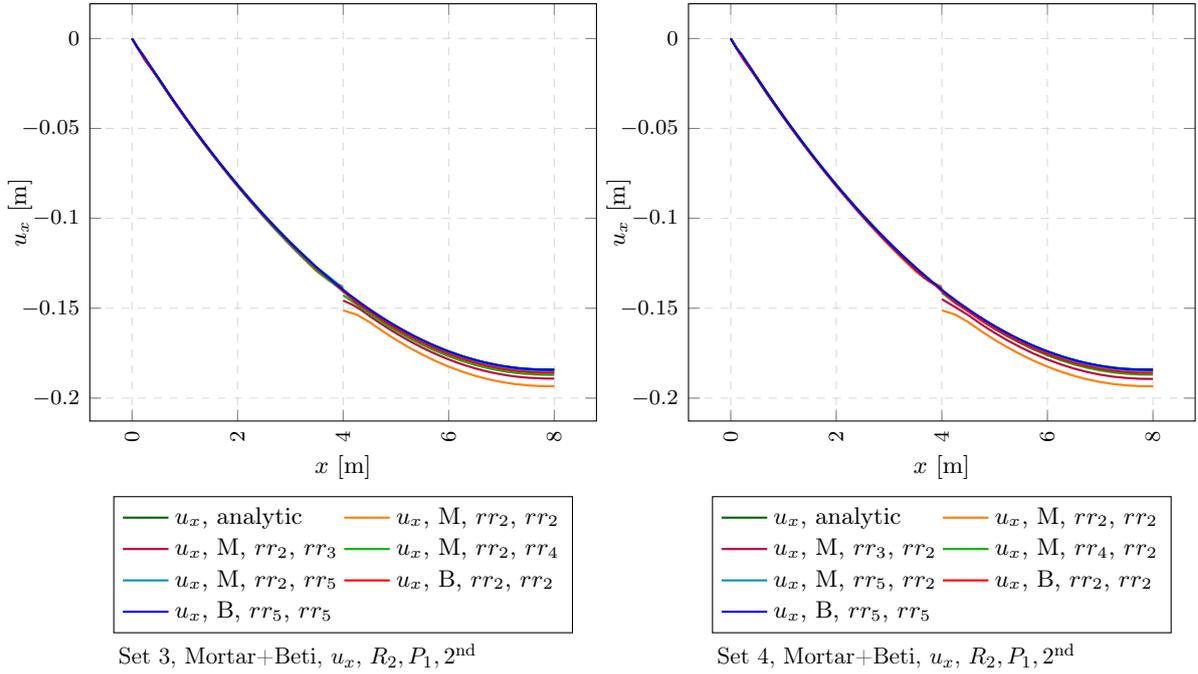


Figure 4.54: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

Displacements u_x					
Coupling	rr	v_l	v_r	$\Delta v = v_r - v_l$	v_e
Mortar	2	-0.13779200	-0.15120300	-0.01341000	-0.19340800
Mortar	3	-0.13888900	-0.14493500	-0.00604000	-0.18931800
Mortar	4	-0.13949700	-0.14141500	-0.00192000	-0.18694900
Mortar	5	-0.13933100	-0.14060100	-0.00127000	-0.18491100
Beti	2	-0.14056000	-0.14056000	0.00000000	-0.18600500
Beti	5	-0.13971900	-0.13971900	0.00000000	-0.18429100

$v_l, v_r \dots$ value at the *left, right* side of the interface
 $v_e \dots$ value at beam end, $rr \dots$ refinement ratio

Table 4.24: Set 4, disagreement of displacements at the interface

Displacements u_z , bottom edge, reg. 1+2, part. 1

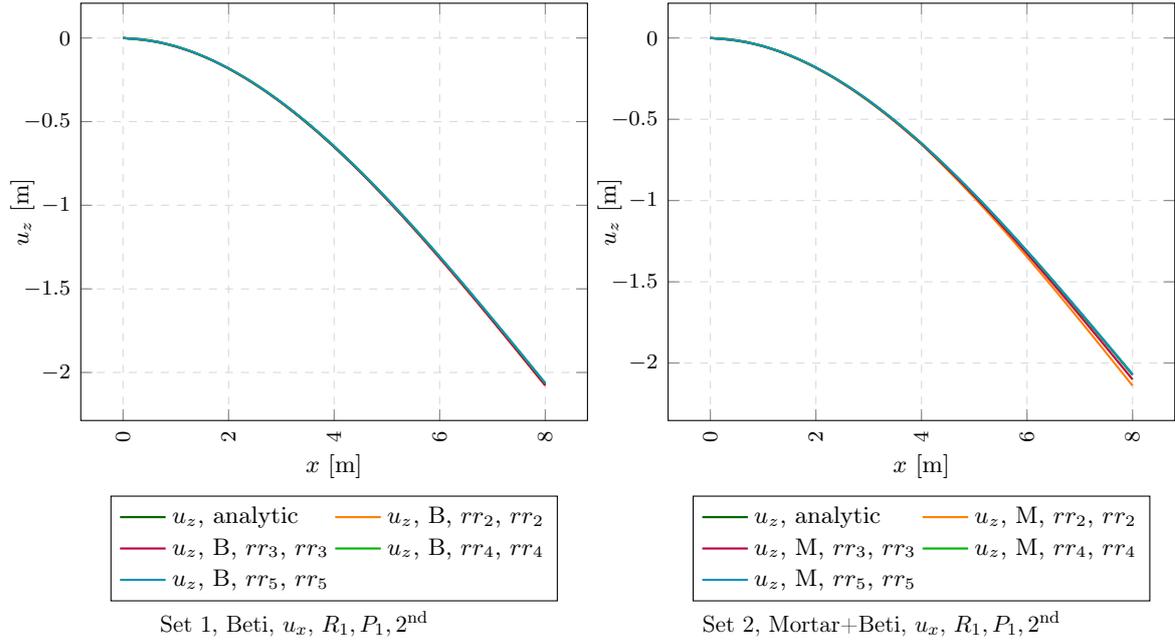


Figure 4.55: Convergence behaviour of *BETI* and *Mortar Method* coupling

Displacements u_z					
Coupling	rr	v_l	v_r	$\Delta v = v_r - v_l$	v_e
Mortar	2	-0.65290400	-0.65246200	0.00044000	-2.13942000
Mortar	3	-0.65101200	-0.65072800	0.00029000	-2.10135000
Mortar	4	-0.64760200	-0.64749600	0.00010000	-2.07181000
Mortar	5	-0.64763400	-0.64756400	0.00006000	-2.06906000
Beti	2	-0.65024200	-0.65024200	0.00000000	-2.07930000
Beti	5	-0.64732400	-0.64732400	0.00000000	-2.06384000

$v_l, v_r \dots$ value at the *left, right* side of the interface
 $v_e \dots$ value at beam end, $rr \dots$ refinement ratio

Table 4.25: Set 2, disagreement of displacements at the interface

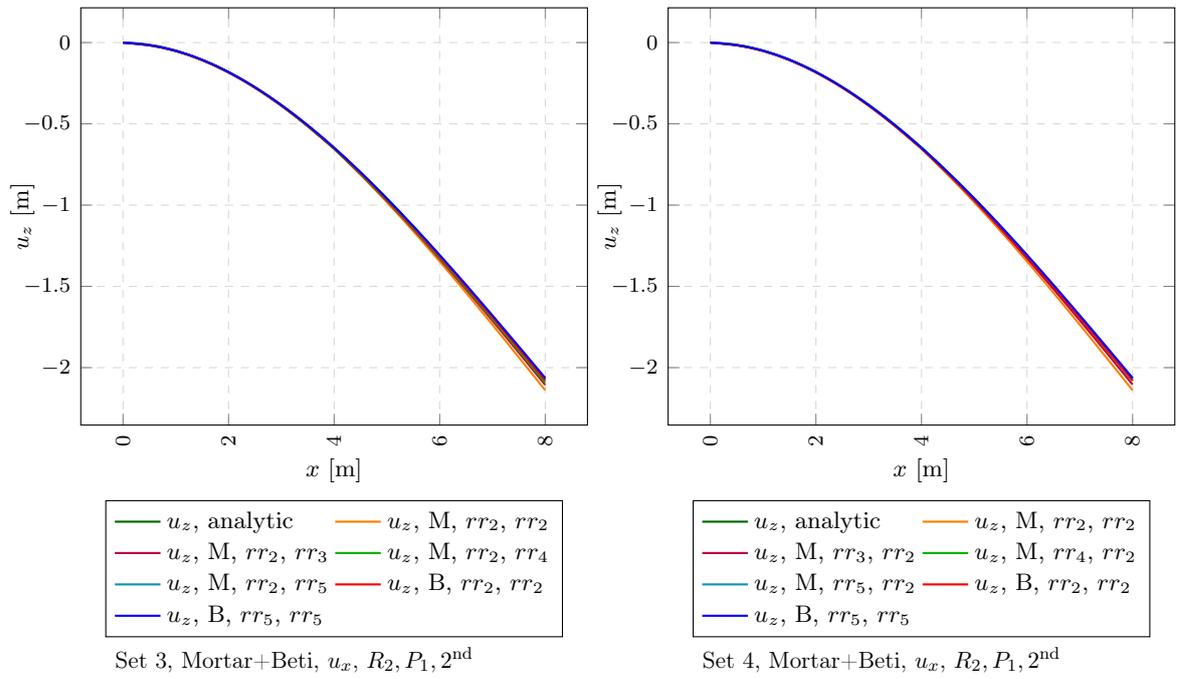


Figure 4.56: Unequal mesh densities, comparing the *Mortar Method* with *BETI* coupling

Displacements u_z					
Coupling	rr	v_l	v_r	$\Delta v = v_r - v_l$	v_e
Mortar	2	-0.65290400	-0.65246200	0.00044000	-2.13942000
Mortar	3	-0.65095100	-0.65077800	0.00018000	-2.10407000
Mortar	4	-0.64773600	-0.64745400	0.00027000	-2.08147000
Mortar	5	-0.64556700	-0.64571200	-0.00014000	-2.06743000
Beti	2	-0.65024200	-0.65024200	0.00000000	-2.07930000
Beti	5	-0.64732400	-0.64732400	0.00000000	-2.06384000

$v_l, v_r \dots$ value at the *left, right* side of the interface
 $v_e \dots$ value at beam end, $rr \dots$ refinement ratio

Table 4.26: Set 4, disagreement of displacements at the interface

Chapter 5

Conclusion and outlook

5.1 Conclusion

As a conclusion the following facts derived from the convergence study can be stated.

1. In general the convergence behaviour of the complete system shows major dependency on the convergence behaviour along the regions but minor on the actual interface configuration.
2. *Mortar Method* coupled systems with equal mesh refinement ratio's in both regions turn out to have the same convergence behaviour like *BETI* coupled systems. They also approach the same displacement values.
3. When using unequal mesh refinement ratio's in both regions we can observe that the systems still converge against the *BETI* coupled solution. This convergence behaviour is due to the actual interface geometry configuration. One fact is that having a dense mesh on one side of the interface is advantageous. Another fact is that using coarse meshes potentially breaks monotonous convergence behaviour. Monotonous convergence behaviour starts from a certain mesh refinement ratio then.
4. As expected the convergence behaviour inside the regions becomes better when having more nodal degree's of freedom. This fact has only partial influence on the convergence behaviour on the displacement at the interface. This is for the reason that a better adaptation of both meshes at the interface is due to three different circumstances.
 - (a) nodal degree's of freedom of the interfacing meshes
 - (b) *Lagrange multipliers* of the *Mortar Finite Elements* (*Master* element degree's of freedom)
 - (c) order of the element shape functions
5. Increasing the element shape function order leads to high convergence rate along the regions. This is as expected. The disagreement of displacements at the interface is still due to the *Lagrange multiplier fields* because the number of *Lagrange multipliers* do not increase by the order of the element shape functions.

5.2 Outlook

In this section the possibilities for the development and the applications of the *Mortar Method* are shown.

5.2.1 Developing the *Mortar Method*

From the current state of this coupling method there are some possibilities for further development given.

More *Lagrange multipliers*

The virtual work is computed by the integral of the displacement function times the virtual traction function ψ situated in the *Lagrange multiplier field*. ψ is constant inside the *Lagrange multiplier field*. It can be changed to a polynomial of any order like finite elements have. The benefit of this would be that the number of *Lagrange multipliers* of one *Lagrange multiplier field* is increasing by the number of degree's of freedom of this function. But this will also lead to a higher number of integrals that need to be evaluated.

Coupling condition

By now the coupling condition is always just an integral of the field function amongst the limits of the *Lagrange multiplier fields*. This formulation does not take care about the momenta this integrals have according to the *Lagrange multiplier fields*. Physically this method just satisfies the *balance of linear momentum* but not the *balance of angular momentum*. There is a publication which introduces a way to find a partitioning and the size of the *Lagrange multiplier fields* that allows for satisfying both of the momentum balances mentioned above. It is called the *zero-moment-rule*[9] which is available for 1D interfaces of 2D regions. This method does not come very handy for a numerical analysis in common *Finite Element Method* and *Boundary Element Method* applications because it is necessary to solve a 1D beam problem to find the needed points of zero moment. For this a solution can be developed that introduces *Lagrange multipliers* (additional degree's of freedom) for the angular momenta. This applies to 3D regions with 2D interfaces as well. The following mathematical formulations shall give a hint how this possibly could work.

Present coupling coefficients, balance of linear momentum:

$$c_{i,j} = \int_{\gamma_j} \varphi_i \cdot \psi_j d\gamma$$

where $c_{i,j}$ is the coupling coefficient for node i according to field j , φ_i is the nodal shape function of node i and ψ_j is the weight function inside the field γ_j .

Additional coupling coefficients, balance of angular momentum:

$$a_{i,j} = \int_{\gamma_j} (\varphi_i \cdot r) \cdot \psi_j d\gamma$$

where r is the radius of the current location of $d\gamma$ to a freely chosen reference point inside the *Lagrange multiplier field* γ_j . As an idea the balance of angular momentum can then be defined for each coordinate direction.

Generalisation

The *Mortar Method* is defined by an approach in a weak sense which is based on an integral formulation. This can be generalized by replacing the integral over the field function amongst the limits of a *Lagrange multiplier field* by a function Φ that projects the field function in a certain way into the *Lagrange multiplier field*. The only limit for the definition of this function Φ is the compatibility with the linear equation system (it needs to be a scalar function of the nodal degree's of freedom at the interface).

5.2.2 Applications of the *Mortar Method*

Here a number of applications is shown where the *Mortar Method* makes sense.

Different displacement approaches

- Unequal mesh refinement ratio's at the interface of two regions.

- Different element shape function order of the elements at the interface.
- Different discretisation types of two interfacing regions (FEM, BEM).
- Coupling a mesh (FEM or BEM) with a beam (Euler-Bernoulli theory).
- Coupling a mesh (FEM or BEM) with a potential field (e.g. Ansatz of Trefftz)

Tunnel structures

For tunnel structures it is very common to use the *Boundary Element Method* with a semi-finite domain. In modern applications also plastic zones are introduced for more accurate modelling and a better result for displacement and stress. These zones are typically discretised with finite elements since this approach fits better to the physical formulation of such plastic zones and offers more degrees of freedom for a dense solution. Since it makes sense to refine/adapt the FEM mesh for each load step we typically get non-conforming interfaces. To make the mesh adaptation completely independent from the surrounding regions the *Mortar Method* can be applied to couple those interfacing regions.

Contents

1	Introduction	1
1.1	Motivation for the <i>Mortar Method</i>	1
1.2	Motivation for this Master's Thesis	1
1.3	<i>Mortar Method</i> fundamentals	1
1.4	Other coupling methods	2
1.4.1	<i>FETI</i> coupling	2
1.4.2	<i>BETI</i> coupling	3
1.5	Roadmap to this Master's Thesis	5
2	Continuum mechanics	6
2.1	Preliminaries	6
2.1.1	Assumptions and Limitations	6
2.1.2	Symbol names	6
2.1.3	Some expressions	7
2.2	Mechanics of a solid body	8
2.2.1	Boundary value problem, strong form	8
2.2.2	Boundary value problem, weak form	9
2.3	Numerical approximation	12
2.4	Coupling coefficients	13
2.4.1	Coefficients of the coupling condition	13
2.5	<i>Lagrange multiplier fields</i>	14
2.6	Integration	15
2.6.1	Integration for 1D interfaces	16
2.6.2	Integration for 2D interfaces	18
2.7	Global equation system	20
3	A simple example	21
3.1	Configuration	21
3.2	Stiffness matrices	21
3.3	Dirichlet boundary conditionidentity matrices	22
3.4	<i>Mortar Method</i> coupling matrices	22
3.5	Load vector (right hand side)	25
3.6	Degree's of freedom	26
3.7	Global equation system	26
3.8	Solution and interpretation	27
3.8.1	Proofs	28
4	Convergence study for the 2D and 3D cantilever beam	31
4.1	Preliminaries	31
4.1.1	Discretisation	31
4.1.2	Dirichlet boundary conditions	32
4.1.3	Neumann boundary conditions	32
4.1.4	Material parameters	32

4.1.5	Mesh refinement	33
4.1.6	Graph sets	34
4.1.7	Legends and captions used in figures and tables	34
4.2	Analytic solution	34
4.2.1	Displacement functions:	35
4.2.2	Transversal displacement maximum $u_{z,\max}$:	36
4.2.3	Longitudinal displacement extrema $u_{x,\max}$, $u_{x,\min}$:	36
4.2.4	Longitudinal and transversal displacement functions:	37
4.3	Observations	38
4.4	Convergence study for the 2D cantilever beam	39
4.4.1	System geometry	39
4.4.2	Interface displacements using first order shape functions	39
4.4.3	Interface displacements using second order shape functions	40
4.4.4	Discretisation with first order shape functions	41
4.4.5	Discretisation with second order shape functions	50
4.5	Convergence study for the 3D cantilever beam	58
4.5.1	System geometry	58
4.5.2	Interface displacements using first order shape functions	58
4.5.3	Interface displacements using second order shape functions	59
4.5.4	Discretisation with first order shape functions	60
4.5.5	Discretisation with second order shape functions	69
5	Conclusion and outlook	77
5.1	Conclusion	77
5.2	Outlook	77
5.2.1	Developing the <i>Mortar Method</i>	77
5.2.2	Applications of the <i>Mortar Method</i>	78

Bibliography

- [1] Silvia Bertoluzza, Silvia Falletta, Valérie Perrier. Implementation of the mortar method in the wavelet context. *Journal of Scientific Computing*, 29(2):219–255, 2006.
- [2] Christine Bernardi, Yvon Maday, Anthony T. Patera. A new nonconforming approach to domain decomposition: The mortar element method. *Nonlinear partial differential equations and their applications: College de France Seminar*, 299:13–51, 1994.
- [3] Institute of Structural Analysis at Graz University of Technology. Befe++: a scientific computation program.
- [4] Stefanica Dan. *Domain Decomposition Methods for Mortar Finite Elements*. PhD thesis, Courant Institute of Mathematical Sciences, New York University, September 1999.
- [5] Rolf H. Krause, Barbara I. Wohlmuth. Nonconforming domain decomposition techniques for linear elasticity. *East-West Journal of Numerical Mathematics*, 8:177–206, 2000.
- [6] Bernd Flemisch, Barbara I. Wohlmuth. Nonconforming methods for nonlinear elasticity problems. In *Domain Decomposition Methods in Science and Engineering XVI*, volume 55, chapter Lecture Notes in Computational Science and Engineering, pages 65–76. Springer-Verlag, 2007.
- [7] Barbara I. Wohlmuth. A mortar finite element method using dual spaces lagrange multipliers. Report, Mathematik Institut, Universität Augsburg, Universitätsstr. 14, D-86159 Augsburg, Germany, November 2007.
- [8] Günther Of. *BETI-Gebietszerlegungsmethoden mit schnellen Randelementverfahren und Anwendungen*. PhD thesis, Universität Stuttgart, Holzgartenstr. 16, 70174 Stuttgart, 2006.
- [9] K.C. Park, C.A. Felippa, G. Rebel. Interfacing non-matching finite element discretizations: the zero moment rule. *Trends in Computational Mechanics*, (CIMNE: Barcelona, Spain):355–367, 2001.