Dai Zhao

Parallelized space-time finite element method for reducing the effects of impact loads

Doctoral Thesis

Supervisor: Dr. Habil. Bartłomiej Dyniewicz, Prof. IPPT PAN

Warsaw, Poland 2024

Contents

Abstract

This manuscript presents a comprehensive exploration of the space-time finite element method (STFEM) as it pertains to solving problems governed by hyperbolic differential equations, with a particular focus on wave propagation and structural dynamics. The STFEM is distinguished by its unique capabilities that allow for continuous adaptation of the spatial mesh in real-time, which significantly enhances the accuracy and efficiency of numerical simulations. A key innovation of this method is its ability to separate the resulting equations within the algebraic system during the formulation of characteristic matrices, effectively omitting the traditional triangulation step. The primary objective of this research is to design and implement advanced software optimized for rapid, massively parallel computing. Ultimately, it will serve as a crucial tool for identifying and optimizing critical parameters influencing structural behavior, thereby contributing to the advancement of adaptive engineering solutions in the face of dynamic stresses. Through this study, we seek to significantly enhance the fields of computational mechanics and structural engineering, offering both theoretical insights and practical applications that address contemporary challenges in these domains.

Keywords: Space-time approach; Finite element method, Structural dynamics, Vibrations, Parallelisation.

Chapter 1

Introduction

Reliable simulation of dynamic systems, especially for large or deformable structures, poses significant computational challenges. Accurate simulations involve solving many interrelated equations simultaneously, requiring extensive computational power. Floating point calculations are crucial for accuracy but are more demanding compared to integer calculations used in graphics processing. Multi-core graphics cards excel in parallel processing for graphics but struggle with the heavy floating-point operations needed in precise simulations. Current computational packages use multi-core CPUs but often don't exploit full parallel processing capabilities, limiting efficiency. Fully explicit methods can speed computation through simplifications but often sacrifice accuracy and stability. Optimizing algorithms for parallel tasks can significantly enhance performance in simulations. Domain decomposition methods and parallel iterative solvers are examples of algorithms that can be efficiently executed on multi-core systems. Hybrid computing architectures, combining CPUs and GPUs, can balance the precision floating-point capabilities of CPUs with the parallel power of GPUs. Modern GPUs are improving in handling floating-point operations, allowing optimized algorithms to boost performance. Ensuring the algorithms are suited for parallel execution is essential for effective simulations. Efficient workload distribution across multiple cores or processors prevents bottlenecks and enhances performance. Precision and performance trade-offs need careful consideration to maintain essential accuracy while improving computation speed. Understanding the acceptable trade-offs for specific applications can guide the simplifications used. Efficient simulation requires advanced computational techniques and architectures. By leveraging optimized algorithms, hybrid computing solutions, and managing trade-offs, accurate simulations can be achieved more swiftly. Computational efficiency is essential for simulating large and deformable systems. The focus must be on algorithm optimization, hybrid architectures, and suitable trade-offs to enhance performance. This approach ensures that simulations are both accurate and fast, meeting the demands of dynamic system modeling.

1.1 Background

In the realm of structural dynamics, various numerical techniques are employed to solve second-order differential equations over time. One commonly utilized method is the central difference method, valued for its capacity to isolate algebraic equations, especially when a diagonal inertia matrix is implemented. To preserve the possibility of system equation diagonalization, matrices involving time derivative vectors must be also diagonal. Despite its benefits, this method is constrained by conditional stability, necessitating users to adhere to specific time step criteria, typically dictated by the most rigid finite element present. As a consequence, its application is predominantly limited to scenarios where inertial effects are predominant and concerns regarding vibrations or wave phenomena are minimal, such as in crashworthiness analyses. However, despite these restrictions, the central difference method can be harnessed efficiently on massively parallel processors for certain practical problems.

Commonly used methods from the Newmark group (the method of central differences is a particular representative of this group) are characterized by good convergence and possible unconditional stability, but they require cost solutions of systems of equations with full or band matrices. There appeared also methods that separate the system of algebraic equations, such as the Park or Trujillo method, but convergence and effectiveness, including accuracy, in general, have not been demonstrated. Let us also note that present dynamical numerical solutions have parabolic properties, ie. the infinite velocity of information propagation, instead of physical properties, that are hyperbolical.

Currently, several ways of speeding up calculations using parallel processors are used. The first one, the simplest, is the use of compilers that in the process of creation of the binary code can extract computational threads that can be executed independently of each other. They are sent to separate processors and executed in parallel. The disadvantage of this solution is a relatively small increase in performance. Separated threads are usually short and sparse.

The second way requires the programmer to be familiar with the methodology and well-thought separation of source code parts that can be implemented in parallel. In this case, the programmers most often modify classical algorithms, which can ultimately be called „parallel" in a sense. In computational methods in structural mechanics, eg. the finite element method, it is easiest to subject to parallel calculations the stage of creation of global structure matrices. Since elemental matrices are computed independently of each other, in principle one can build each of them on a separate processor. In non-linear tasks or tasks with variable coefficients, this process must be repeated many times. The second stage, which is heavily burdening the calculation process, is the solution of the system of algebraic equations. In this case, for example, methods of solving systems of equations are used in a block way.

It is also worth mentioning here the iterative methods, which constitute a separate branch of algorithms, and which are more suitable for use in multiprocessor computations. However, people are still dealing with methods known for a long time and with algorithms known for sequential calculations, which are only adapted to the needs of parallel computing.

The implementation of parallel calculations in structural mechanics applied nowadays, particularly in dynamics, can be called the adaptation of known methods rather than the use of new, innovative solutions. The obtained acceleration of calculations now reaches several to several dozen times, but in the overwhelming majority of tasks, these are realizations far from real-time calculations. According to the data provided by the manufacturer of commercial Ansys software, the machine using CUDA technology is characterized by several times higher efficiency compared with a computer using only the CPU for the calculation.

Therefore, developing effective tools for simulating the dynamics of structures on an average computer at a speed currently reserved only for extremely expensive supercomputers seems to be an interesting research topic. Calculations in many cases will be able to be performed in real time. It is important in planning and performing surgery operations. Numerical modelling in structural dynamics is important both for researchers investigating the vibrating systems, as well as for engineers working on industrial problems. To accomplish engineering goals, high-performance software is needed to allow rapid simulations of structural dynamics. The method of space-time finite elements with elements of simplex shapes is naturally suitable for massively parallel processing.

Graphic processors GPU is specialized for performing repetitive operations when rendering (imaging) three-dimensional graphics. Behind such a visualisation complicated geometrical calculations are placed. So a concept was created to transfer some of the duties of the central processor to the graphics card. This idea became the basis of the technology "General Purpose computation on GPUs", that is, the implementation of general calculations by a graphics processor, in short GPGPU. The existence of much faster tools for computer simulations also enables efficient multi-criterial optimization in structural dynamics. As a consequence, dynamic inverse problems are within the scope of computational possibilities. The use of GPU in scientific and engineering calculations is not a new idea. Currently, many commercial computing packages use this technology: Ansys, LS-DYNA, MSC Nastran, Matlab, and many more.

Modern methods of computer structure analysis require discretization of the area of the examined object. The finite element method along with other similar methods, most often used, is based on the integral approach. In the second group differential methods in which the derivatives are replaced by approximate difference quotients are placed. The boundary element method is less frequently used. In a certain period, a group of meshless methods has become popular among researchers. Even a cursory analysis of the meshless approach shows that its accuracy depends on the regularity

of the distribution of nodes in the test area. Increasing the number of nodes causes them to be more evenly distributed, and thus, the solution is smoother and seems more accurate. The convergence, however, is not satisfactory and has not been proven. Unfortunately, elementary tests of the application of this method to wave problems show a significant divergence with an accurate solution or a rapid growth of error in subsequent steps of integration over time. As a result, such solutions are useless for structural dynamics. Researchers who use this method for elliptical problems limited the demonstration of its effectiveness to relatively regular and well-developed examples. All other effective methods practically used by scientists and engineers are based on the representation of the solution in nodes, resulting from the classical discretization based on the Galerkin method.

1.2 Thesis

The space-time finite element approach, utilizing simplex-shaped elements, allows for highly efficient, massively parallelized linear and nonlinear dynamic computations that surpass the efficiency of traditional computational parallelization techniques.

The goal was achieved by designing and implementing an advanced algorithm optimized for fast, massively parallel calculations in the field of structure dynamics. The manuscript presents a highly efficient parallelized version of the continuous space-time finite element method, tailored for structural dynamics and wave propagation simulations. This innovative method enables the execution of time-dependent mathematical physics tasks across multiple processor cores without compromising the accuracy of sequential computations. Typically, in conventional approaches, the finite element method is used for spatial discretization, while a different technique is employed for the time dimension in solving differential equations of motion. Contrarily, the space-time finite element method extends this by treating time as an interpolated variable akin to a spatial variable. This unification allows for a more seamless and holistic approach to time-dependent problems. By distributing different parts of the task to individual processor cores, the proposed method harnesses parallel computation, significantly boosting efficiency. It ensures that simulations of structural dynamics, or broader physical phenomena with a temporal component, are executed more rapidly yet with the same precision as traditional sequential methods. One of the core advantages of this approach is maintaining identical accuracy in the parallelized simulation as would be achieved in a single-core, sequential computation. It reflects an extension and an improvement over conventional finite element methods, particularly in handling time as an integral part of the interpolation process. The method's novelty lies in its ability to apply space-time interpolation in a parallelized environment. It addresses the limitations of commonly used approaches where time is treated separately from space

in the approximation process. This dual interpolation not only improves computational efficiency but also simplifies the treatment of time-dependent tasks. The manuscript emphasizes the method's capability to handle structural dynamics and wave propagation with enhanced computational performance. By leveraging multiple processors, it achieves faster simulations without any sacrifice in accuracy. The space-time finite element method's parallel implementation represents a substantial advancement in computational modeling. The results underscore the method's proficiency in delivering accurate simulations akin to those obtained through sequential calculations. The approach is particularly beneficial in scenarios requiring intensive computation and high precision over extended temporal domains. Overall, the parallelized space-time finite element method marks a significant step forward in the efficient and accurate modeling of dynamic structural and physical systems.

1.3 Overview of Thesis

This dissertation presents the development and application of the space-time finite element method (STFEM) to enhance the accuracy and efficiency of solving complex differential equations. The first two chapters introduce the thesis's goals and current research, focusing on the decoupling of systems of equations to allow for parallel computations. This process involves calculating unknowns in sequential steps, facilitating parallel processing and progressive movement in computations. The simplex-shaped space-time finite element method is described as a technique for solving differential equations, utilizing simplex-shaped functions to define space-time coordinates within subelements, which is crucial for addressing wave problems and structural dynamics without necessitating triangulation.

Chapter 3 provides a detailed introduction to STFEM, discussing the flow within the mesh and deriving relevant mathematical expressions. The chapter presents essential matrices, such as the stiffness matrix and the inertia matrix, which are critical for accurately modeling dynamic behavior. The properties of simplex elements are explored, highlighting the lower triangular nature of the strain displacement matrix, which simplifies the computational process. A brief overview of the nonlinear iteration process within subdomains is also included to ensure solution accuracy.

Chapter 4 presents selected engineering problems. First, a one-dimensional rod model is constructed to perform force analysis and derive the governing partial differential equation of motion. The space-time finite element method is applied to create stiffness and inertia matrices using simplex elements. The chapter demonstrates the method's capability by comparing results from traditional finite element methods with those from STFEM. Then, large deformations in plane-stress problems, deriving key relationships between stress and strain were presented. Nonlinear terms are introduced

to maintain accuracy when the system experiences significant displacements. The chapter derives shape functions for tetrahedral elements in two dimensions within STFEM and provides numerical solutions for various scenarios, highlighting the importance of incorporating nonlinear terms. The derived characteristic matrices for the nonlinear case are presented for the first time.

Chapter 5 explores the practical implementation of STFEM in a two-dimensional model, detailing the characteristic matrices and the iterative process managed through parallel computing. The algorithm of the solution was presented. The chapter highlights enhanced computational speed and efficiency through benchmarking. Finally, the main conclusions resulting from the proposed method of discretization of the differential equation of motion are presented.

In Chapter 6, the application of STFEM is illustrated through two examples centered on developing advanced impact protection materials. The first example focuses on a smart elastic material that adapts properties in response to mechanical wave propagation, while the second models a viscoplastic material. The findings confirm the advantages of using STFEM in both parallel processing capabilities and complex material modeling.

Chapters 7 and 8 summarize the manuscript, present conclusions, and plans for further research on the presented issue.

Chapter 2

Literature survey

2.1 Parallel Computing in Structural Dynamics

Numerical engineering computations are applied in various fields, including structural analysis [86, 61, 2, 62, 81, 3], fluid dynamics [100, 57, 101, 38, 23, 60], electromagnetism [27, 88, 30, 102, 28, 92], and thermal analysis [99, 24, 73, 42, 106, 72], to solve highly complex problems that often involve non-linear equations and multi-physics interactions. These methods enable engineers to simulate and predict how systems will behave under real-world conditions, often eliminating the need for expensive physical prototypes. For example, in structural analysis, numerical methods allow for detailed assessments of stress and strain in materials under various loads, ensuring safety and reliability in bridges, buildings, and other constructions. In fluid dynamics, they help model airflow over airplane wings or water flow in hydraulic systems, providing critical insights into efficiency and performance.

Computer simulations offer unparalleled flexibility, enabling engineers to test various design scenarios and parameters before physical implementation. For instance, in the automotive industry, simulations are used to analyze crash safety, optimize fuel efficiency, and enhance the structural integrity of vehicles. Additionally, in electromagnetism, numerical methods facilitate the design of advanced electrical components, such as motors, transformers, and wireless communication devices, by predicting electromagnetic fields and power losses.

However, due to the immense data sets and complex algorithms involved, traditional computers may struggle to process such computations efficiently. As a result, high-performance computing systems, including powerful clusters and supercomputers, are often required to handle these demanding tasks. These systems leverage parallel processing across multiple cores, employing both multicore CPUs and GPUs to divide computations into smaller, manageable parts that can be executed simultaneously. This parallelization significantly speeds up the computational process, allowing engi-

neers to obtain accurate results in a fraction of the time it would take on conventional hardware.

Fast and accurate numerical computations are vital not only for reducing product development time but also for minimizing the risks associated with faulty designs or incorrect assumptions. For example, in aerospace engineering, even minor miscalculations can lead to catastrophic failures. Numerical simulations allow for the exploration of "what-if" scenarios, testing the limits of materials and systems in ways that are impractical or too dangerous to experiment with in real life. Modern computational technologies have revolutionized the engineering process, enabling rapid innovation and more sustainable, safer designs across industries. Without these tools, many cutting-edge developments in fields like renewable energy, biotechnology, and advanced manufacturing would be far more difficult, costly, and time-consuming to achieve.

Parallel computing has become a critical technique for addressing complex computational problems in structural dynamics. The ability to distribute tasks across multiple processors allows for significant reductions in computation time, enabling the analysis of larger and more complex structures. This literature review explores the development, application, and impact of parallel computing in the field of structural dynamics, drawing from a variety of scholarly sources.

The early applications of parallel computing in structural dynamics began in the 1980s, driven by the need to handle increasingly complex structural analysis tasks. Storaasli et al. (1987) were among the pioneers, documenting their experiences with parallel computing systems and demonstrating that significant improvements in computational efficiency could be achieved [96]. They highlighted the potential of parallel algorithms to reduce the time required for large-scale structural computations, marking a significant step forward in the field. Following this, Hajjar and Abel (1988) explored the use of domain decomposition techniques for transient nonlinear structural dynamics. This approach allowed the division of a large computational domain into smaller subdomains, which could be processed concurrently [48]. This work demonstrated the feasibility and efficiency of parallel processing in handling the complexities of three-dimensional framed structures. These early efforts laid the groundwork for subsequent advancements by proving that parallel computing could significantly enhance the performance of structural dynamics simulations. The focus during this period was primarily on exploiting the available parallel computing architectures, such as vector processors and early multiprocessor systems, to handle the computational load more effectively. Researchers also began to develop and refine parallel algorithms specifically designed for structural dynamics applications. The success of these early implementations encouraged further research and development in the field, leading to more sophisticated techniques and broader applications. These foundational works established the importance of parallel computing in structural dynamics and highlighted the potential for continued innovation and improvement. The early developments in parallel computing set the stage for the extensive use of these methods in modern

structural analysis, paving the way for more complex and accurate simulations.

An interesting thread related to computation parallelization is time-parallel methods. The concept of time-parallel methods, which distribute the computation across different time steps, was introduced to improve the efficiency of solving dynamic problems. Time-parallel methods, also known as parallel-in-time (PinT) methods, are advanced computational techniques designed to solve time-dependent problems more efficiently by distributing the computation of different time intervals across multiple processors simultaneously. Traditional time integration methods, such as the finite difference or finite element methods, solve the problem sequentially in time, which can be computationally expensive for large-scale structural dynamics problems. Time-parallel methods address this limitation by allowing time steps to be computed in parallel, significantly reducing the overall computation time. One of the pioneering works in this area was the development of the Parareal algorithm. This method divides the time domain into several sub-intervals and solves them iteratively. The Parareal algorithm uses a coarse solver to provide an initial approximation over the entire time domain and a fine solver to refine the solution within each sub-interval. This iterative approach ensures that the computational effort is distributed across multiple processors, achieving parallelism. Cortial and Farhat (2009) [29] proposed a time-parallel implicit method that specifically targets non-linear structural dynamics problems. Their method demonstrated substantial speed-ups by leveraging parallel computing resources to solve implicit time integration schemes, which are commonly used in structural dynamics due to their stability properties. The method efficiently handles the non-linearities and complexities associated with structural dynamics by parallelizing the time domain. Farhat et al. (2006) [35] revisited this framework in the context of linear structural dynamics and near-real-time computing. They developed parallel implicit time-integration algorithms (PITA) that aimed to deliver near-real-time predictions of structural responses. This was particularly useful for applications requiring rapid computations, such as earthquake engineering and real-time monitoring of structures. Another significant contribution to time-parallel methods is the multiple-shooting method. This approach splits the time domain into smaller intervals, solves the initial value problem independently within each interval, and then iteratively adjusts the solutions to ensure continuity and consistency across the intervals. This technique is particularly beneficial for stiff problems where traditional methods may struggle with stability and convergence issues. Implementing time-parallel methods in structural dynamics requires careful consideration of the problem's characteristics, such as non-linearity, stiffness, and boundary conditions. Domain decomposition techniques are often employed in conjunction with time-parallel methods to partition the spatial domain, further enhancing parallelism. This combined approach ensures that both spatial and temporal components of the problem are efficiently distributed across available computational resources. The performance of time-parallel methods heavily depends on the communication overhead between processors. Efficient algorithms must minimize this overhead

to fully exploit the benefits of parallelism. Techniques such as pipelining and asynchronous communication are often used to achieve this goal. Recent advancements in hardware, particularly the availability of high-performance GPUs, have further accelerated the adoption of time-parallel methods. GPUs offer massive parallelism at a relatively low cost, making them ideal for implementing PinT algorithms. Researchers have reported significant speed-ups by leveraging GPU acceleration for time-parallel computations in structural dynamics. Despite their advantages, time-parallel methods also face challenges. Ensuring stability and convergence across multiple time intervals can be difficult, particularly for highly non-linear problems. Developing robust and scalable algorithms that can handle these challenges remains an active area of research.

The field of structural dynamics has significantly benefited from advancements in domain decomposition and finite element methods (FEM). These techniques have enabled the efficient analysis of complex structural systems by partitioning computational tasks across multiple processors, reducing computation time, and enhancing accuracy. Domain decomposition has been a popular strategy for parallelizing structural dynamics computations. Sotelino (2003) [93] reviewed various parallel processing techniques and their applications in structural engineering, highlighting the importance of domain decomposition for handling large-scale structural dynamics problems. Yaghoubi et al. (2015) [104] explored the integration of domain decomposition with model reduction techniques to further enhance computational efficiency. Non-overlapping domain decomposition methods have been effectively applied in structural mechanics, as discussed by Gosselet and Rey (2006) [43]. Their work emphasizes the integration of finite and discrete element methods to improve computational efficiency. Similarly, Kwak et al. (2014) [63] presented a domain decomposition approach for simulating the dynamics of flexible multibody systems, demonstrating the effectiveness of finite element tearing and interconnecting methods. In the realm of nonlinear analysis, Jahromi et al. (2009) [55] introduced a domain decomposition method for soil-structure interaction. This method employs staggered time-marching schemes to handle complex coupling, significantly improving computational performance. Yagawa et al. (1991) [103] explored large-scale finite element analysis using domain decomposition on parallel computers, highlighting the benefits of this approach in handling extensive structural dynamics problems. Adaptive finite element methods have also seen advancements, as shown by Abas and Abdul-Rahman (2016) [1]. They integrated domain decomposition with adaptive FEM for fluid-structure interaction, achieving better computational efficiency and accuracy. Rao et al. (2003) [85] proposed a parallel overlapped domain decomposition method specifically for nonlinear dynamic finite element analysis, enhancing the efficiency of structural computations. Gravouil and Combescure (2003) [44] developed a multi-time-step and two-scale domain decomposition method for non-linear structural dynamics, providing a robust framework for solving complex dynamic problems. Stavroulakis et al. (2017) [94] leveraged GPU acceleration to enhance the perfor-

mance of the spectral stochastic finite element method using domain decomposition, demonstrating significant computational speed-ups. Rodrigues and Correa (2007) [87] applied implicit domain decomposition methods to the coupled analysis of offshore platforms, addressing the challenges posed by the marine environment on structural dynamics. Farhat and Li (2005) [36] focused on iterative domain decomposition methods for solving indefinite problems in computational structural dynamics, improving the efficiency of structural analysis through advanced iterative techniques. Overall, these studies underscore the critical role of domain decomposition and finite element methods in advancing the field of structural dynamics. By efficiently distributing computational loads and integrating advanced numerical techniques, these methods have paved the way for more accurate and faster analyses of complex structural systems. The continuous development and integration of these methods with modern computational resources, such as GPUs, promise further enhancements in the capabilities and applications of structural dynamics simulations.

The advent of more powerful computational resources, particularly high-performance computing (HPC) systems and graphical processing units (GPUs), has significantly accelerated research and applications in structural dynamics. These advancements have enabled the implementation of highly parallel algorithms, which distribute computational tasks across many processors simultaneously, drastically reducing computation times. GPUs have been particularly impactful due to their massive parallelism capabilities. Unlike traditional central processing units (CPUs), which may have a few cores optimized for sequential processing, GPUs contain thousands of smaller cores designed for handling multiple tasks concurrently. This architecture is ideal for the repetitive and parallelizable nature of finite element analysis in structural dynamics. Fazanaro et al. (2016) [37] demonstrated the role of GPUs in the numerical characterization of nonlinear dynamical systems. By leveraging GPU acceleration, they achieved significant performance gains compared to CPU-based computations. Similarly, Kang et al. (2014) [59] proposed a GPU-based parallel computation method for structural dynamic response analysis, using CUDA, a parallel computing platform and application programming interface (API) model created by NVIDIA. CUDA allows developers to harness the power of NVIDIA GPUs for general-purpose computing, enabling the efficient execution of parallel algorithms. This capability has been widely adopted in structural dynamics to accelerate finite element method (FEM) computations. The use of CUDA has demonstrated substantial reductions in computation time for dynamic response analysis of large-scale structures. In addition to GPUs, other HPC architectures, such as distributed computing clusters and supercomputers, have facilitated advancements in structural dynamics. These systems consist of numerous interconnected processors that work together to solve large-scale computational problems. The parallel nature of domain decomposition and time-parallel methods aligns well with the architecture of HPC systems, enabling significant scalability and efficiency improvements. Supercomputers, in particular, have played a crucial role in

enabling researchers to tackle complex structural dynamics problems that were previously infeasible. The use of supercomputers allows for the simulation of highly detailed models with millions of degrees of freedom, providing deeper insights into the behavior of structures under various dynamic loads. The integration of advanced computational resources with sophisticated numerical methods has also led to the development of hybrid approaches. For instance, combining domain decomposition with model reduction techniques further enhances computational efficiency. Model reduction simplifies the computational model by reducing the number of degrees of freedom while preserving essential system characteristics. Furthermore, advances in software development have complemented hardware improvements. The creation of specialized libraries and frameworks, such as PETSc (Portable, Extensible Toolkit for Scientific Computation) and Trilinos, has provided researchers with powerful tools to implement parallel algorithms more effectively. These libraries offer optimized routines for linear algebra operations, solvers, and preconditioners, which are critical components in finite element analysis. The continuous evolution of both hardware and software resources has also driven advancements in multiscale modeling. This approach involves simulating phenomena at multiple scales, from the microscopic to the macroscopic level, providing a comprehensive understanding of structural behavior. High-performance computational resources are essential for managing the enormous computational demands of multiscale models. Despite these advancements, challenges remain in fully exploiting the capabilities of modern computational resources. Efficiently partitioning computational tasks, minimizing communication overhead between processors, and developing scalable algorithms are ongoing areas of research. Additionally, the adoption of machine learning techniques in structural dynamics presents new opportunities and challenges for leveraging computational resources. Advances in computational resources, particularly GPUs and HPC systems, have revolutionized the field of structural dynamics. By enabling highly parallel computations and integrating advanced numerical methods, these resources have significantly improved the efficiency and accuracy of structural dynamics simulations. Ongoing research and development promise further enhancements, paving the way for even more complex and detailed analyses in the future.

While parallel computing has brought substantial benefits to structural dynamics, several challenges remain. The efficient partitioning of computational tasks and the management of communication overhead between processors are critical for achieving optimal performance. Additionally, the development of algorithms that can fully exploit the capabilities of modern parallel architectures continues to be an area of active research. Negrut et al. (2014) [79] discussed the potential and limitations of parallel computing in multibody system dynamics, arguing that it represents the main source of performance improvement in the near future. Parallel computing has revolutionized the field of structural dynamics by enabling the efficient analysis of complex structures that were previously infeasible to study. From early implementations using domain decomposition techniques to modern approaches leveraging GPUs, the field has seen

substantial advancements. Continued research and development in parallel algorithms and computational resources promise to further enhance the capabilities and applications of structural dynamics analysis.

2.2 Space-time finite element method

The number of works describing physical phenomena in space-time is enormous. Space-time is considered a concept that allows for a more flexible description of phenomena in various ranges. The proposed space-time way of formulation of dynamic tasks was developed by many researchers and numerous publications on the subject can be found in literature. However, the overwhelming majority of the work concerned mathematical formulations, assumptions, proofs of existence, and uniqueness of solutions, and less was devoted to algorithmic and computational aspects of solutions. Below we will present only a few publications from a vast group to outline the scientific background in which we will operate.

The evolution of space-time modeling in physical tasks has a rich history, beginning with foundational work in the 1960s and progressing through significant methodological advancements and diverse applications. The earliest attempts were made in 1964 by Gurtin [46, 47] and Herrera [51], who introduced models that elucidated the relationship between time and spatial variables using the theory of convolutions. These initial efforts established the concept of space-time finite elements, a framework that allows for the simultaneous consideration of temporal and spatial dimensions in modeling physical phenomena. In 1969, Oden [83] expanded this framework by generalizing the finite element method to include space-time considerations. Fried [41], along with Argyriss, Scharpf, and Chan [5, 6, 7], contributed by treating spatial and temporal variables with equal importance in physical problem formulations. Kączkowski [64, 65, 66, 68] applied the space-time finite element method in the dynamics of structures.

Traditionally, the space-time finite element method used a fixed spatial division of structures and assumed that each space-time element covered a rectangular region within the time-space domain, leading to algorithms similar to direct numerical integration methods. However, Kączkowski identified limitations in this approach, arguing that it restricted the method's potential. To address this, he proposed the use of non-rectangular, specifically triangular, elements. This modification offers several advantages. First, triangular elements provide greater flexibility, allowing them to adapt more effectively to complex geometries and dynamic phenomena that result in non-uniform deformation. Second, they offer improved accuracy by capturing the dynamic behavior of structures more precisely, especially in cases involving concentrated forces or abrupt geometric changes. Kączkowski demonstrated the efficacy of triangular elements through examples of dynamic problems [69, 67, 70], showing that they can

produce accurate results even when forces introduce discontinuities in the structure's shape.

The practical applications of space-time finite element methods have been diverse. These methods have been effectively employed in wave propagation studies [53, 40, 4], acoustics [98], and fluid mechanics [50]. Additionally, they have been explored for mesh adaptation in dynamic problems [12]. In the realm of moving mass problems, continuous task transitions facilitated by STFEM have been studied [14, 15, 31, 17], and dynamic simulations in train-track interactions have demonstrated their practical utility [33].

One of the critical aspects of space-time modeling is the selection of virtual functions, which significantly influence the accuracy and stability of the solutions. Different time-dependent functions result in various solution schemes, as highlighted in a review by [16]. Despite these advancements, classical stationary discretizations remain the dominant approach in engineering practice, with space-time elements being less frequently used.

Innovations such as the introduction of symplectic elements for non-stationary spatial divisions marked significant progress in the field. Initially proposed by Oden [83], these elements have been successfully applied in various contexts, including beams [10, 9, 13, 17]. Simplex-shaped elements have also been used in contact problems [11] and tested within the context of advection-diffusion equations [20].

Further developments include the study of numerical information flow in space-time meshes [32] and non-continuous formulations in time-space by Hulbert and Hughes [53, 52]. Research has also addressed Stefan problems [26, 76], parabolic evolution problems [95, 75], and Navier–Stokes equations [49], which involve lower-order time derivatives. Approaches to moving boundary problems [25, 97] and mesh evolution in vibration problems [12, 18] have further expanded the application of space-time methods.

The application of the generalized Maxwell model to a 2D slab, which refined space-time discretization, is another notable advancement [54]. Schafelner [91] provided a comprehensive formulation of space-time methods for parabolic evolution problems using unstructured decompositions. Behr [21] presented detailed formulations and algorithmic considerations for these methods, and Kacprzyk [58] proposed special interpolation functions to enhance their effectiveness.

Chapter 3

The simplex-shaped space-time finite element method

3.1 Information flow in mesh

In mainstream literature, numerical simulations of dynamic systems often rely on explicit and implicit integration methods to model the behavior of such systems over time. These methods have distinct approaches to handling the inertia matrix, which plays a critical role in wave propagation problems within dynamic systems. The inertia matrix's structure can significantly impact how information flows through the space-time mesh during calculations. When the inertia matrix is consistent or band-shaped, information transfers between nodes at a rapid pace in successive time steps, almost instantly reaching all nodes in the system. This phenomenon, depicted in Fig. 3.1, demonstrates a unique characteristic of how information spreads through the mesh, akin to a wave, despite not being directly related to the physical system being simulated. In cases where the inertia matrix is diagonal, the calculation scheme is simplified into a system of separated algebraic equations. This simplification alters the way information propagates through the mesh. An impulse or disturbance at one node now propagates gradually to neighboring nodes in subsequent time steps, rather than instantaneously reaching all nodes, as depicted in Fig. 3.2. The choice of space-time mesh type and time integration scheme influences the speed at which information propagates. The limitations or infinite speed of information flow in these meshes can have implications on the accuracy of simulation results. For instance, limited information velocity is more suitable for hyperbolic problems, while infinite speed is typical for parabolic problems. The latter may lead to issues such as wave reflections from mesh boundaries affecting results significantly. To address these drawbacks, the simplex-shaped space-time finite element method is introduced. This method enables controlled one-way information flow at a limited speed, resulting in triangular matrices

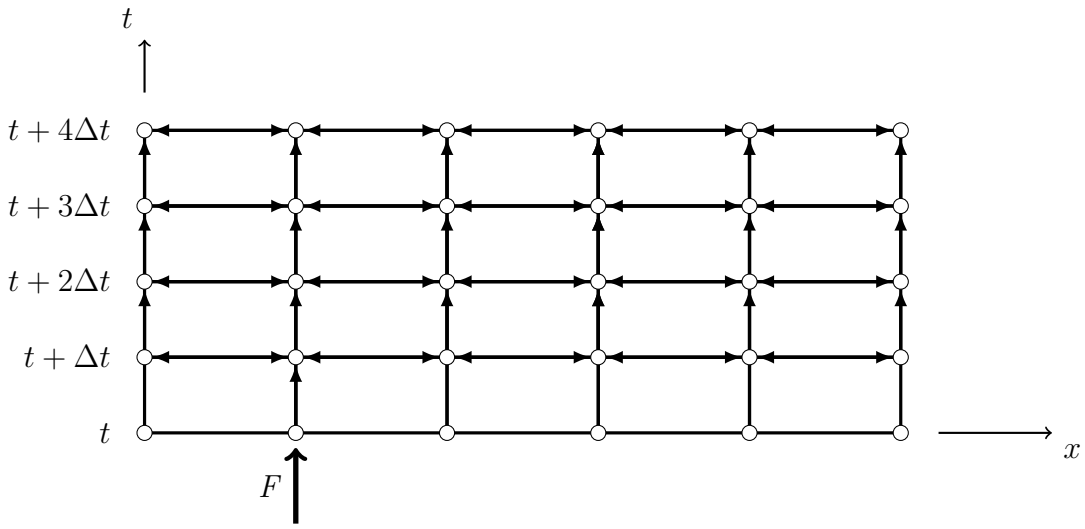


Figure 3.1: Information flow in methods with full inertia matrix

of algebraic equations. It permits the movement of numerical data in a single direction at a restricted rate (Fig. 3.3). By allowing information propagation in a specified direction at a reduced velocity, this method proves beneficial in scenarios like moving load problems [32], where maintaining a smooth flow of information is crucial to avoid non-physical disturbances in the mesh. The above properties can also be used in the parallelization of calculations, which is the subject of this dissertation.

3.2 Mathematical background

Let us consider a continuum closed in a domain $\overline{V}$, being a subdomain in Euclidean space E^3. V denotes the interior of this subspace and ∂V its boundary, being the sum of ∂V_t and ∂V_u. Stress and displacement boundary conditions are assumed on ∂V. The motion of the body in time interval $[0, T]$ is considered. Displacement vector $\mathbf{u}$, velocity vector $\mathbf{v}$, inertial forces $\rho\mathbf{f}$, symmetric tensor of stresses $\boldsymbol{\sigma}$ and strains $\boldsymbol{\varepsilon}$ are determined on Cartesian product of sets $V \times [0, T]$. Vector of surface forces $\widehat{\mathbf{t}}$ is determined on the product $\partial V \times [0, T]$. A set of kinematic and physical equations with boundary and initial conditions describes the problem. Equations of motion are as follows

$$\operatorname{div} \boldsymbol{\sigma}^T + \rho\mathbf{f} = \rho\frac{\partial \mathbf{v}}{\partial t} \ , \quad (\mathbf{x}, t) \in V \times [0, T]. \tag{3.1}$$

The respective system of equations formulates the problem locally. The transition to the global formulation is obtained by multiplication of (3.1) by the virtual displacement

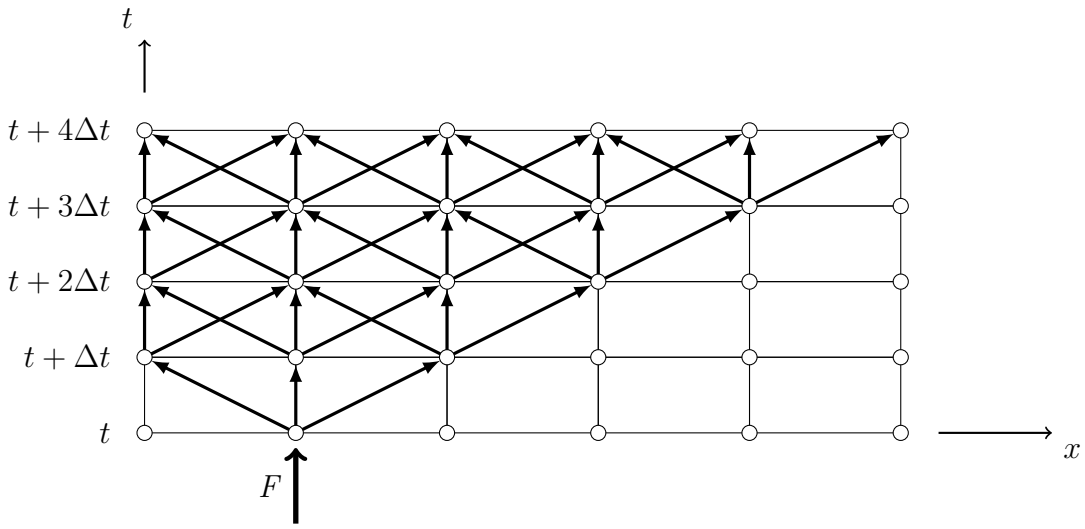


Figure 3.2: Information flow in explicit methods with diagonal inertia matrix

function $\delta\mathbf{u}(\mathbf{x},t)$. After integration we obtain

$$\int_{t_0}^{t_1}\int_V \left(\operatorname{div}\boldsymbol{\sigma}^T + \rho\mathbf{f} - \rho\dot{\mathbf{v}}\right)\delta\mathbf{u}\,\mathrm{d}V\,\mathrm{d}t + \int_{t_0}^{t_1}\int_{\partial V_t}\widehat{\mathbf{t}}\,\delta\mathbf{u}\,\mathrm{d}(\partial V)\,\mathrm{d}t = 0\ . \tag{3.2}$$

Integration by parts yields

$$\int_{t_0}^{t_1}\int_V \rho\left(\mathbf{f}\delta\mathbf{u} + \dot{\mathbf{u}}\,\delta\dot{\mathbf{u}}\right)\,\mathrm{d}V\,\mathrm{d}t + \int_{t_0}^{t_1}\int_{\partial V_t}\widehat{\mathbf{t}}\,\delta\mathbf{u}\,\mathrm{d}(\partial V)\,\mathrm{d}t = \int_{t_0}^{t_1}\int_V \boldsymbol{\sigma}\delta\boldsymbol{\varepsilon}\,\mathrm{d}V\,\mathrm{d}t\ . \tag{3.3}$$

The domain $\{\overline{V}, 0 \le t \le T\}$ must be discretised. In this initial boundary problem, the half-infinite space-time band can be split into various space-time finite elements. The straightforward partition into rectangular elements in time (generally into multiplex shape elements) makes this method similar to the classical FEM, with time integration carried out with the Newmark family method.

The simplest space-time elements can be cut out of the time layer limited by planes $t = t_i$ and $t = t_{i+1}$ in forms of prisms. Thus final objects can be considered as finite spatial elements extended over a time interval (Figure 3.4). The unknown parameters like real and virtual displacements $\mathbf{u}$ and $\delta\mathbf{u}$, respectively, and their derivatives $\dot{\mathbf{u}}$, $\boldsymbol{\varepsilon}$, $\boldsymbol{\sigma}$ etc. are interpolated from nodal displacements $\mathbf{q}$ and $\delta\mathbf{q}$

$$\begin{aligned}
\mathbf{u}(\mathbf{x},t) &= \mathbf{N}(\mathbf{x},t)\,\mathbf{q}, & \delta\mathbf{u}(\mathbf{x},t) &= \mathbf{N}^*(\mathbf{x},t)\,\delta\mathbf{q},\\
\dot{\mathbf{u}}(\mathbf{x},t) &= \dot{\mathbf{N}}(\mathbf{x},t)\,\mathbf{q}, & \delta\dot{\mathbf{u}}(\mathbf{x},t) &= \dot{\mathbf{N}}^*(\mathbf{x},t)\,\delta\mathbf{q},\\
\boldsymbol{\varepsilon}(\mathbf{x},t) &= \mathbf{B}(\mathbf{x},t)\,\mathbf{q}, & \delta\boldsymbol{\varepsilon}(\mathbf{x},t) &= \mathbf{B}^*(\mathbf{x},t)\,\delta\mathbf{q},\\
\boldsymbol{\sigma}(\mathbf{x},t) &= \mathbf{E}\,\mathbf{B}(\mathbf{x},t)\,\mathbf{q}. &&
\end{aligned} \tag{3.4}$$

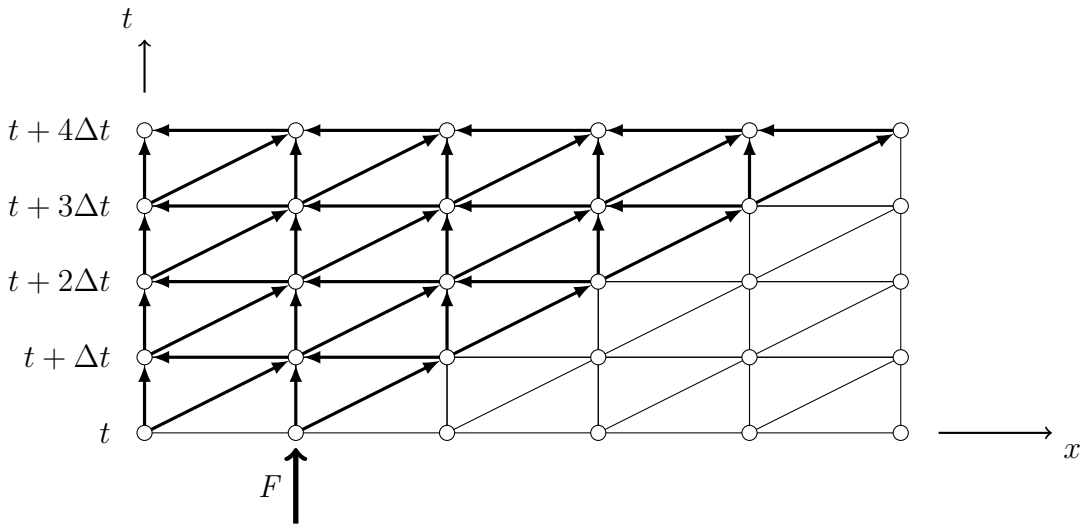


Figure 3.3: Information flow in a simplex-shaped space-time finite element mesh.

Matrix $\mathbf{B}$ can be obtained by acting with a differential operator $\mathcal{D}$ on the shape functions $\mathbf{N}$: $\mathbf{B} = \mathcal{D}\,\mathbf{N}$, where $\mathcal{D} = \frac{1}{2}\left(\mathrm{grad} + \mathrm{grad}^T\right)$. Symbol $(.)^*$ refers to the virtual state. The Kelvin-Voigt model of viscoelasticity defined by the Young modulus E and viscous damping coefficient η_w is assumed. The above interpolation is applied to each space-time subdomain. The set of local equations is then obtained.

Considering (3.4) in (3.3) the quadratic form of the equilibrium of the energy in the time interval $[t_0,\, t_1]$ can be written

$$\sum_{e=1}^{NE} \left((\mathbf{\Pi}_e^T \delta\mathbf{q}_e)^T \mathbf{\Pi}_e^T \tilde{\mathbf{K}}_e \mathbf{\Pi}_e \cdot \mathbf{\Pi}_e^T \mathbf{q}_e - (\mathbf{\Pi}_e^T \delta\mathbf{q}_e)^T \mathbf{\Pi}_e^T \mathbf{Q}_e \right) = 0. \tag{3.5}$$

NE is the number of space-time elements in the space-time layer. Matrices $\mathbf{\Pi}_e$ are zero-one tables assigning degrees of freedom of the element to the global set of degrees of freedom. These matrices determine the way of summation of local matrices into a global matrix. The same process is carried on in a classical finite element approach. The elemental space-time stiffness matrix $\tilde{\mathbf{K}}_e$ can be considered in a similar way to the equivalent stiffness matrix in the Newmark algorithm

$$\tilde{\mathbf{K}}_e = \mathbf{K}_e + \mathbf{M}_e \ . \tag{3.6}$$

$\mathbf{K}_e$ contributes the stiffness effect and is proportional to physical stiffness k multiplied by time step h, while $\mathbf{M}_e$ contributes the inertia effect and is proportional to physical mass m divided by time step h.

If the Kelvin-Voigt model is assumed and damping forces are included in (3.1), two additional terms $\mathbf{W}_e$ and $\mathbf{Z}_e$ must be added to (3.6)

$$\tilde{\mathbf{K}}_e = \mathbf{K}_e + \mathbf{M}_e + \mathbf{W}_e + \mathbf{Z}_e \ . \tag{3.7}$$

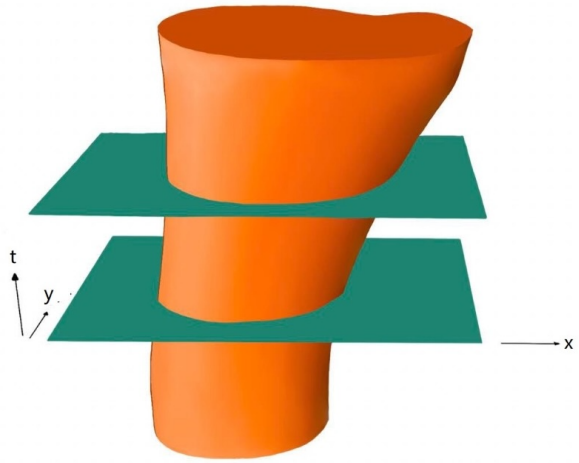


Figure 3.4: Continuous time evolution of the 2D domain.

Final forms of stiffness $\mathbf{K}_e$, inertia $\mathbf{M}_e$, internal $\mathbf{W}_e$ and external $\mathbf{Z}_e$ element damping matrices are given below

$$\begin{aligned}
\mathbf{K}_e &= \int_{t_0}^{t_1}\int_V (\mathcal{D}\mathbf{N})^T\,\mathbf{E}\,\mathcal{D}\mathbf{N}\,\mathrm{d}V\,\mathrm{d}t\ , \\
\mathbf{M}_e &= -\int_{t_0}^{t_1}\int_V \left(\frac{\partial \mathbf{N}}{\partial t}\right)^T \mathbf{R}\,\frac{\partial \mathbf{N}}{\partial t}\,\mathrm{d}V\,\mathrm{d}t\ , \\
\mathbf{W}_e &= \int_{t_0}^{t_1}\int_V (\mathcal{D}\mathbf{N})^T \eta_w \mathcal{D}\frac{\partial \mathbf{N}}{\partial t}\,\mathrm{d}V\,\mathrm{d}t\ \ , \\
\mathbf{Z}_e &= \int_{t_0}^{t_1}\int_V \mathbf{N}^T \eta_z \frac{\partial}{\partial t}\mathbf{N}\,\mathrm{d}V\,\mathrm{d}t\ ,
\end{aligned} \tag{3.8}$$

where $\mathbf{R}$ is the matrix of inertia and η_z is the external damping coefficient.

The vector of external forces acting on the space-time element is denoted by $\mathbf{Q}_e$

$$\mathbf{Q}_e = \int_{t_0}^{t_1}\int_V \mathbf{N}_e(\mathbf{x},t)\,\widehat{\mathbf{t}}(\mathbf{x},t)\,\mathrm{d}V\,\mathrm{d}t\ . \tag{3.9}$$

Let us denote the initial displacement vector by $\mathbf{q}_0$ and initial velocities by $\dot{\mathbf{q}}_0$

$$\mathbf{q}_0 = \sum_{e=1}^{NE} \mathbf{\Pi}_e^T \int_{V_e} \mathbf{N}_e(\mathbf{x},0)\ \mathbf{u}(\mathbf{x},0)\,\mathrm{d}V_e\ , \quad \dot{\mathbf{q}}_0 = \sum_{e=1}^{NE} \mathbf{\Pi}_e^T \int_{V_e} \mathbf{N}_e(\mathbf{x},0)\ \dot{\mathbf{u}}(\mathbf{x},0)\,\mathrm{d}V_e\ . \tag{3.10}$$

Finally, the equilibrium of the i-th time layer bounded by t_i and t_{i+1} is described by the equation

$$\begin{bmatrix} \tilde{\mathbf{K}}_{i(1,1)} & \tilde{\mathbf{K}}_{i(1,2)} \\ \tilde{\mathbf{K}}_{i(2,1)} & \tilde{\mathbf{K}}_{i(2,2)} \end{bmatrix} \begin{Bmatrix} \mathbf{q}_i \\ \mathbf{q}_{i+1} \end{Bmatrix} = \begin{Bmatrix} \mathbf{Q}_i \\ \mathbf{Q}_{i+1} \end{Bmatrix}\ . \tag{3.11}$$

The above submatrices of the space-time finite element $\tilde{\mathbf{K}}_{i(1,1)}$, $\tilde{\mathbf{K}}_{i(1,2)}$, $\tilde{\mathbf{K}}_{i(2,1)}$, and $\tilde{\mathbf{K}}_{i(2,2)}$, containing the influence of classical stiffness, inertia, and damping, are combined into global matrices denoted simply as $\mathbf{A}_i$, $\mathbf{B}_i$, $\mathbf{C}_i$, and $\mathbf{D}_i$, respectively, where i is the time layer, in a commonly known way dependent on the topology of the mesh. The space-time element matrices and resulting global matrices overlap in a common zone for a given time step. The same happens when forming global matrices in space in the classical finite element method.

3.3 Properties of space-time approximation with simplex elements

Properties of simplex shape space-time elements used in the parallelization of the computational process in subsequent time steps will be presented below.

The first feature is the possibility of a half-decoupling of the system of algebraic equations. It locates this method of space-time elements between methods with diagonal matrices of coefficients of the system of equations and full or band matrices. Let's look at the topology of the spatial mesh, supplemented mentally by several successive space-time layers. We must note in advance that in each case we finally get to solve a system of algebraic equations with the number of unknowns equal to the number of degrees of freedom in a spatial problem. It happens that the reader, observing the presented schematic diagrams, misreads the computational cost of the presented method.

$$\begin{bmatrix} \mathbf{A}_0 & \mathbf{B}_0 & \mathbf{0} & \mathbf{0} & ... \\ \mathbf{C}_0 & \mathbf{D}_0+\mathbf{A}_1 & \mathbf{B}_1 & \mathbf{0} & ... \\ \mathbf{0} & \mathbf{C}_1 & \mathbf{D}_1+\mathbf{A}_2 & \mathbf{B}_2 & ... \\ \mathbf{0} & \mathbf{0} & \mathbf{C}_2 & \mathbf{D}_2 & ... \\ ... & ... & ... & ... & \infty \end{bmatrix} \begin{Bmatrix} \mathbf{q}_0 \\ \mathbf{q}_1 \\ \mathbf{q}_2 \\ \mathbf{q}_3 \\ ... \end{Bmatrix} = \begin{Bmatrix} \mathbf{F}_0 \\ \mathbf{F}_1 \\ \mathbf{F}_2 \\ \mathbf{F}_3 \\ ... \end{Bmatrix}. \tag{3.12}$$

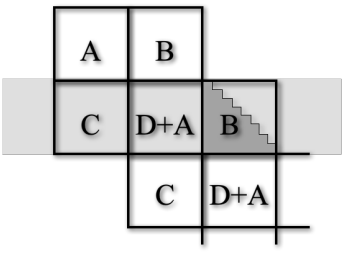


Figure 3.5: Matrix equation for one-time layer.

Each space-time layer generates a matrix equation of equilibrium, binding the state of the system in two moments limiting this layer, i.e. in moments t_i and $t_i+\Delta t$. Knowing the previous state allows you to move to the next state in a stepwise procedure.

Figure 3.5 shows the idea of building a matrix equation

$$\mathbf{C}_{i-1}\mathbf{q}_{i-1} + (\mathbf{D}_{i-1} + \mathbf{A}_i)\,\mathbf{q}_i + \mathbf{B}_i\mathbf{q}_{i+1} = \mathbf{F}_i\ . \tag{3.13}$$

The initial conditions or solutions from the previous steps reduce (3.13) to a simple equation

$$\mathbf{B}_i\mathbf{q}_{i+1} = \mathbf{F}_i - \mathbf{C}_{i-1}\mathbf{q}_{i-1} - (\mathbf{D}_{i-1} + \mathbf{A}_i)\,\mathbf{q}_i \tag{3.14}$$

where on the right we have all the known members. The matrix $\mathbf{B}_i$ with the given below node numbering strategy is a lower triangular matrix.

Decoupling of the system of equations

In Figure 3.6 the example of a 1D structure composed of 4 spatial elements modeled with triangular space-time elements is depicted. The resulting matrix $\mathbf{B}$ is a lower triangular one. Different numbering of nodes and correspondingly different schemes of

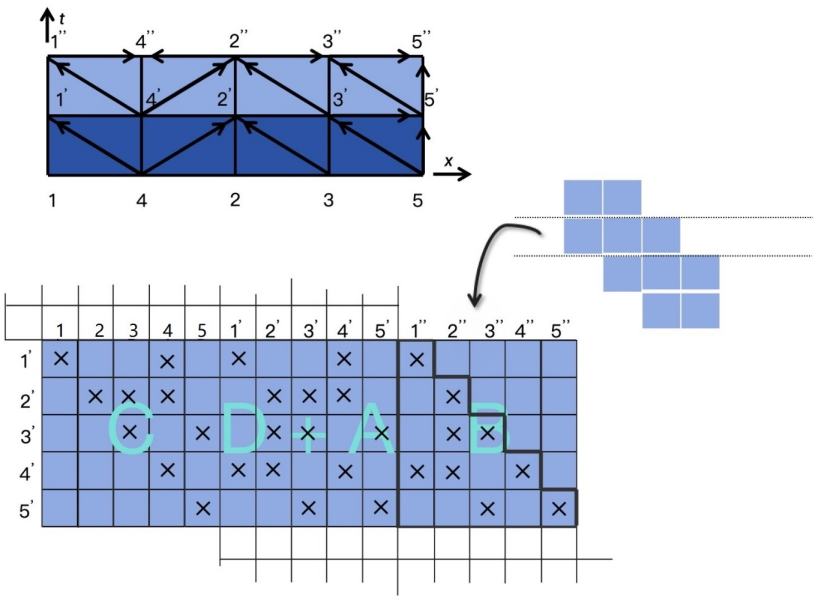


Figure 3.6: Triangular matrix and the information flow between nodes in the exemplary 1-D structure.

diagonal edges of space-time simplex elements allow us to obtain an upper triangular matrix. In yet another case, we will get the coefficients on both sides of the diagonals, but in each case it will be the same system of equations, differing only in the permutation of the unknowns and can be solved equation by equation.

Before we move on to the actual algorithm, there are some more noteworthy features of the method. The first one is the possibility of breaking down the task into classic sub-tasks described, for example, with the usual finite element method or the method of space-time elements in the shape of multiplexes, joined by a space-time layer of symplectic elements. This is shown in the Figure 3.7. The global coefficient matrix in

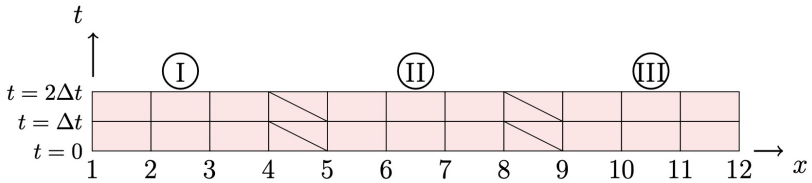


Figure 3.7: Division of the structure into sub-systems solved sequentially.

this case has the form shown in Figure 3.8. Subsystem I can be solved without the need to set the unknowns 5–12. Next, subsystem II is resolved. At the same time, subsystem I can be solved, provided that new values of the coefficients of submatrix I can be determined. In other words, the matrix in Figure 3.8 is a block-triangular matrix. Propagation of the calculation zone is shown in Figure 3.9. Real-time is the

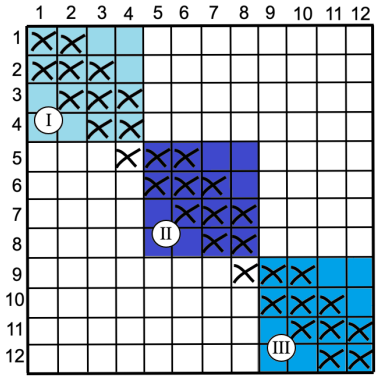


Figure 3.8: Sequential solution of subsystems I, II, and III depicted in Figure 3.7.

astronomical time in which calculations on the computer run. Model time is the time during which the simulated phenomena are observed. Physical space is the Euclidean space of the structure as seen by the observer.

It is possible to draw an identical scheme when instead of three subsystems we place many symplectic elements in the scheme. The computational front will move vertically and the time front will move diagonally. Of course, the scheme is artificially arranged. In order to obtain an image of the entire structure at a certain moment, it would be necessary to slightly hold down the calculations of lower number subsystems and wait

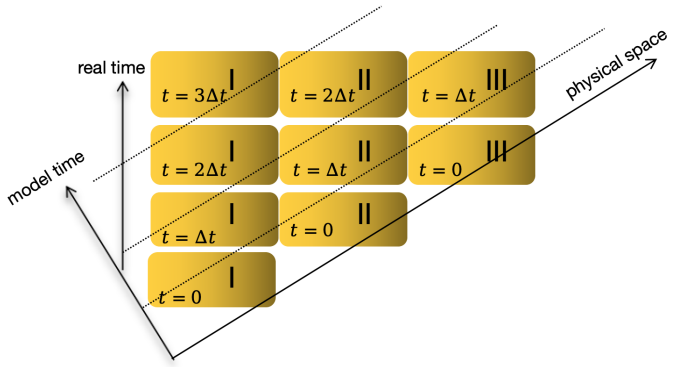


Figure 3.9: Propagation of the calculation zone.

a few or a dozen steps to catch up with the calculations in subsystems with higher numbers.

Successive systems of equations with a triangular matrix can be solved in batches. After solving the initial part of the first set of equations and completing the topology-required unknowns, one can go back to solving the first unknowns of the next set of equations without solving the remaining unknowns of the first set, unless the initial equations are still dependent on unsolved unknowns.

Let us focus on the solution performed in parts. After calculating the unknowns No. 1, 2, 3, 4 in step No. 1, we can return to the solution of the unknown No. 1 in the 2nd step, and calculate the remaining unknowns of the first step in parallel with the unknown No. 1 of step 2. In this way, the calculations can be carried out in parallel on portions of systems of equations. When the 4th batch is calculated at time t_{i-3}, the 3rd batch in parallel will be calculated at time t_{i-2}, batch 2 at time t_{i-1}, and batch 1 at the most advanced time, i.e. at time t_i. In this way, the computational process will move in a sloping front.

Numbering of nodes in symplexes

Once we know what solving a task divided into sub-areas is all about, it is necessary to show how to automate the generation of space-time elements with any spatial dimensionality. One of the simplest ways of assigning the node numbers of the first generated space-time element is simply adding to the already existing node numbers of the spatial element one node at a time of the next time layer. Nodes of consecutive elements are numbered by shifting the sequence of numbers to the left and adding in the last position the number of the first node referred to the moment $t + \Delta t$, i.e. increased by NP. For example, we show a spatial bar element with the numbers i, j, and a one-time node with the number i+NP, where NP is the number of spatial grid nodes. Space-time

prism based on space-time element i, j will be filled with space-time elements (Figure 3.10). The final pair of space-time triangles are depicted in Figure 3.11. The addition of NP to numbers in this figure indicates that the nodes numbered in space i and j are related to the next time layer.

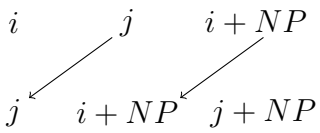


Figure 3.10: Scheme of node numbering in the case of 1D element.

In the case of a three-dimensional element in the shape of a tetrahedron, these will

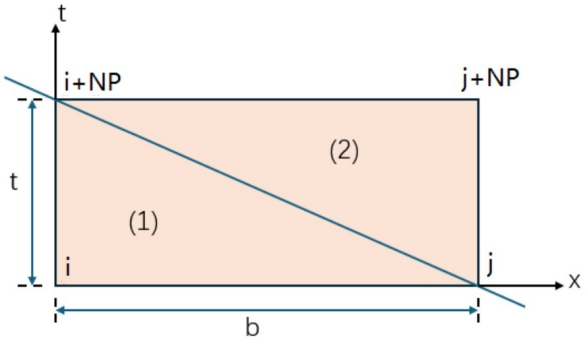


Figure 3.11: Example of node numbering in space-time triangles.

be four space-time elements (Figure 3.12). For some practical reasons, it is good to arrange the nodes i,.., l in ascending order.

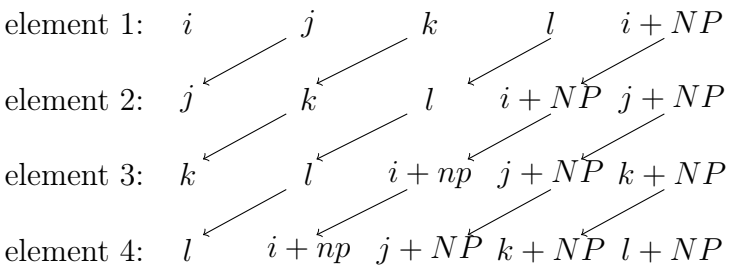


Figure 3.12: Scheme of node numbering in the case of 3D tetrahedral element.

So, starting from a spatial, classic discrete mesh, we can generate geometry sequences of space-time elements, determine the matrices of these elements, and we can

compose global matrices of coefficients. Of course, in that case, we would not yet get the desired benefits. We need to fill the fragments of the coefficient matrix according to the appropriate scheme, and then, also according to the appropriate algorithm, solve the subsystems of equations, free the memory, and fill it again with the coefficients determined from the variable non-linear terms in advancing time.

Nonlinear iteration in a subdomain

Figure 3.13 illustrates another feature resulting from the possibility of separating the system of equations. This can be effectively applied in solutions to nonlinear problems, where nonlinearity affects only a limited spatial region. Proper separation of the structure into subdomains causes a part of the unknowns located in the central part of the vector to not affect the solution of the unknowns in the initial and final parts of the vector. This is achieved by introducing a layer of symplectic elements with appropriately directed oblique edges. This can be easily obtained by appropriate numbering of nodes in the spatial grid and then applying the method of numbering symplectic elements presented in Figures 3.10, 3.12. Thanks to this approach, the iterative solution in a single time step is limited to iteratively solving a small part of the system, which is denoted on the structure scheme in Figure 3.13 with a shadow and in the matrix with a darker background. The arrows in the figure show the flow of information between the

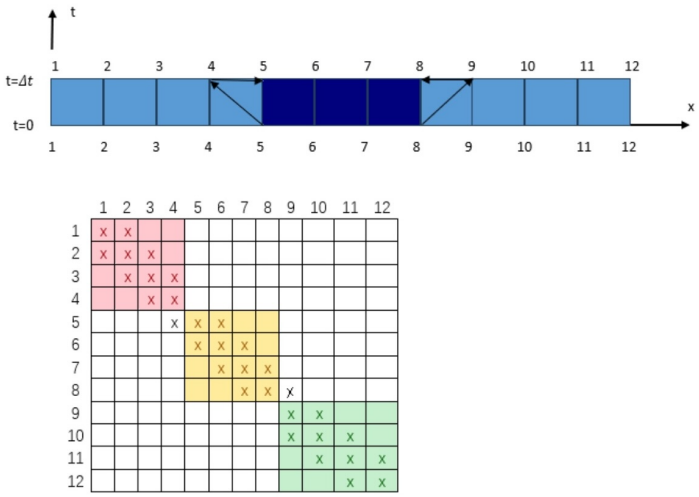


Figure 3.13: The range of influence of non-linear factors on the solution of one-time step.

nodes at the interface between the regions and between time steps. It can be seen that the unknown values in the outer zones, calculated in the first iteration, influence the solutions of the inner zone in the subsequent iterations (nodes 5–8 in $t{=}\Delta t$), not the other way around. Specifically, the results of the inner zone do not affect the results

already obtained in the outer zones.

The influence of nonlinearity explicitly affects the solution in the next step and subsequent steps. The same effect is achieved by using simplex elements throughout the spatial domain.

Chapter 4

Selected engineering problems

In this chapter, we focus on selected examples of one- and two-dimensional structures to demonstrate the application of numerical methods in structural analysis. The first example explores the small deformations of a rod, a classic one-dimensional problem, while the second example examines a two-dimensional plane stress state. These two cases provide a comprehensive framework for understanding the effectiveness of the numerical approach, particularly in deriving characteristic matrices like stiffness and inertia matrices, which are essential for structural computations.

For the one-dimensional case, we begin with the problem of small deformations in a rod under axial loading. The governing equations are derived based on the assumptions of linear elasticity, allowing us to define the problem in terms of displacements and external forces. Using these formulations, simplex shape functions were constructed to approximate the displacement field within the rod. These shape functions form the foundation for developing the stiffness and inertia matrices, which capture the rod's resistance to deformation and its dynamic behavior under external loads.

The two-dimensional example deals with a plane stress state, a common scenario in structural mechanics, particularly for thin plates and shells. The problem is defined by considering equilibrium equations in two dimensions, assuming that out-of-plane stresses are negligible. Similar to the one-dimensional case, simplex shape functions were derived to represent the displacement fields within the two-dimensional domain. These shape functions were then used to derive the stiffness and inertia matrices for the structure, providing a complete representation of its mechanical behavior.

In both cases, the derivation of stiffness and inertia matrices follows from the principle of virtual work, ensuring consistency with the underlying physical laws. The use of simplex shape functions allows for a straightforward and computationally efficient formulation, making these methods particularly suitable for numerical implementations such as the finite element method (FEM). The effectiveness of the derived characteristic matrices is demonstrated through simple examples, where the theoretical predictions are compared with numerical solutions. These examples serve as a foundation for more

complex structural problems and highlight the versatility of the numerical approach in handling both static and dynamic cases. By analyzing the small deformations of a rod and the plane stress state in two dimensions, we aim to illustrate the broader applicability of this methodology in structural engineering and its potential for solving real-world problems with precision and efficiency.

4.1 Small deformation of rod

4.1.1 Problem definition

The study of small deformations in rods is a fundamental problem in structural mechanics. The objective is to determine the displacement field within a rod subjected to external forces and constraints, assuming that the deformations are sufficiently small such that the linear theory of elasticity applies.

Let us consider a rod of length L, uniform cross-sectional area A, Young's modulus E, and mass density ρ, subjected to an external body force $f(x,t)$ per unit length. The rod undergoes small deformations, which is $w(x,t)$, allowing the use of linear elasticity theory, where x $(0 \leq x \leq L)$ and t represent the location of rod and time. Fig. 4.1 shows the general case for a one-dimensional problem. An infinitesimal segment of the

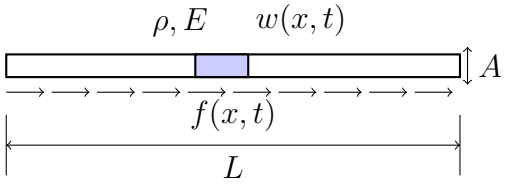


Figure 4.1: General one-dimensional case.

rod between x and $x + \Delta x$ was presented in Fig. 4.2. According to Newton's law, the

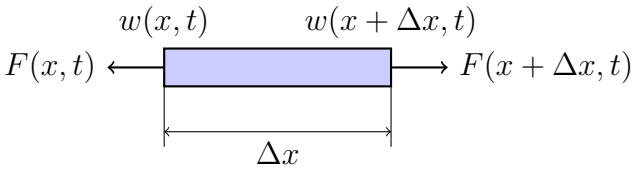


Figure 4.2: The force analysis of an infinitesimal segment in the rod.

forces analysis of this segment can be written in the following form

$$F(x+\Delta x,t) - F(x,t) + f(x,t)\Delta x = \rho A \Delta x \frac{\partial^2 w}{\partial t^2}\,. \tag{4.1}$$

The internal force F within the rod is as follows

$$F(x,t) = \sigma(x,t)A\,, \tag{4.2}$$

where σ is the stress.

Next, let us consider the relationship between displacement u and strain ϵ. In our case, the definition of this infinitesimal segment is given by the formula

$$\epsilon(x,t) = \frac{w(x+\Delta x,t) - w(x,t)}{\Delta x}\,. \tag{4.3}$$

According to Taylor series, (4.3) can be written as follows

$$\epsilon(x,t) = \frac{w(x+\Delta x,t) - w(x,t)}{\Delta x} = \frac{w(x,t) + \frac{\partial w(x,t)}{\partial x}\Delta x - w(x,t)}{\Delta x}\,. \tag{4.4}$$

Therefore, the strain ϵ is related to the displacement w by

$$\epsilon(x,t) = \frac{\partial w(x,t)}{\partial x}\,. \tag{4.5}$$

Moreover, Hooke's Law gives

$$\sigma(x,t) = E\epsilon(x,t) = E\frac{\partial w(x,t)}{\partial x}\,. \tag{4.6}$$

Similarly, using Taylor series expansion and neglecting higher-order terms the stress can be presented in the following form

$$\sigma(x+\Delta x,t) = \sigma(x,t) + \frac{\partial \sigma(x,t)}{\partial x}\Delta x\,. \tag{4.7}$$

Substituting (4.2) to (4.7) into the equilibrium equation (4.1) and simplifying

$$\frac{\partial}{\partial x}\left(EA\frac{\partial w(x,t)}{\partial x}\right) + f(x,t) = \rho A\frac{\partial^2 w(x,t)}{\partial t^2}\,. \tag{4.8}$$

In most instances, assuming E is constant, the one-dimensional solid mechanical differential equation is

$$EA\frac{\partial^2 w(x,t)}{\partial x^2} + f(x,t) = \rho A\frac{\partial^2 w(x,t)}{\partial t^2}\,. \tag{4.9}$$

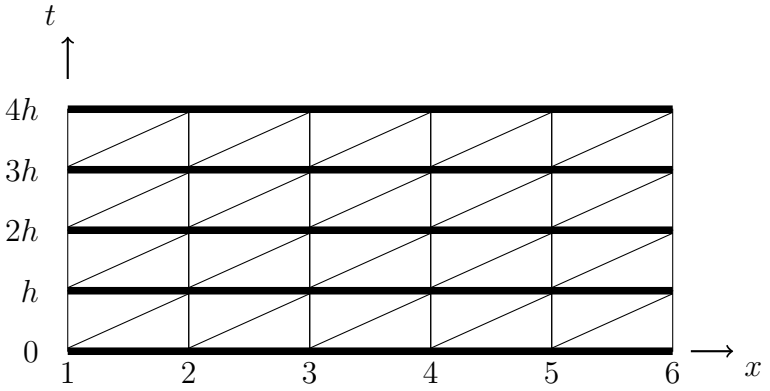


Figure 4.3: The first type of simplex-shaped discretization.

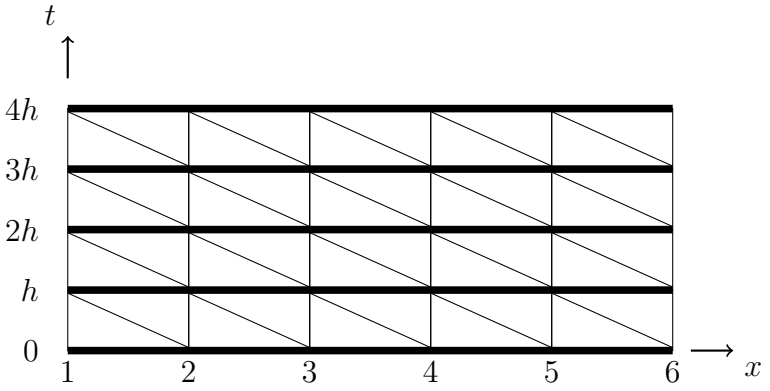


Figure 4.4: The second type of simplex-shaped discretization.

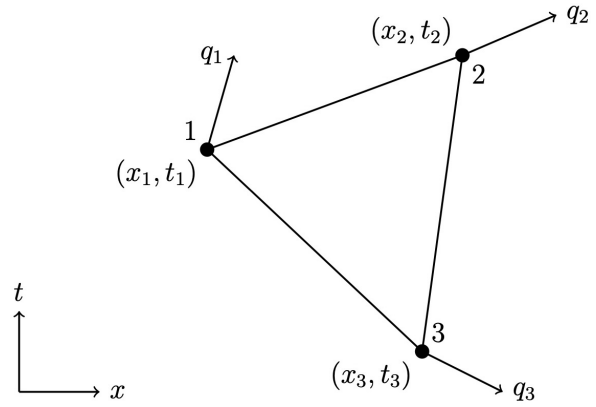


Figure 4.5: An arbitrary space-time triangle sub-element.

4.1.2 Simplex-shaped functions

In the case of one-dimensional structures such as a rod, simplex-shaped space-time discretization is possible in two variants. The first type of discretization is shown in

Fig. 4.3, while the second type of discretization is shown in Fig. 4.4. Due to the time variable t, the space-time discretization is of the second dimension.

Figure 4.5 shows an arbitrary space-time triangular sub-element, which will be the basis for deriving shape functions for single finite elements in two versions. Let us consider an arbitrary triangular sub-element with vertices at coordinates (x_1, t_1), (x_2, t_2) and (x_3, t_3). The displacement of every node is described as follows $w(x_1, t_1) = q_1$, $w(x_2, t_2) = q_2$ and $w(x_3, t_3) = q_3$. Assuming a linear distribution of displacements in the rod element, we can write

$$w(x,t) = a_1 x + a_2 t + a_3 \,. \tag{4.10}$$

Eq. (4.10) can be written in the following matrix form

$$w(x,t) = \boldsymbol{g}\,\boldsymbol{a}\,, \tag{4.11}$$

where $\boldsymbol{g} = [x, t, 1]$ and $\boldsymbol{a} = [a_1, a_2, a_3]^T$. Taking into account (4.11) and using the notation based on the nodal values in the triangle, the following matrix equation is presented

$$\begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} = \boldsymbol{G}\,\boldsymbol{a}\,, \tag{4.12}$$

where

$$\boldsymbol{G} = \begin{bmatrix} x_1 & t_1 & 1 \\ x_2 & t_2 & 1 \\ x_3 & t_3 & 1 \end{bmatrix} . \tag{4.13}$$

By performing simple mathematical operations, we obtain a recipe for calculating the polynomial coefficients (4.10) of the following form

$$\boldsymbol{a} = \boldsymbol{G}^{-1} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} . \tag{4.14}$$

According to (4.11) and (4.14) the final equation describing the displacements in the finite element using nodal values can be presented in the following form

$$w = \boldsymbol{g}\,\boldsymbol{G}^{-1} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} . \tag{4.15}$$

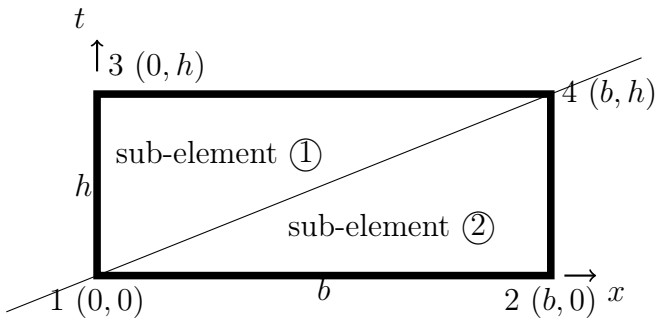


Figure 4.6: A single space-time element of the simplex shape in version 1.

4.1.2.1 1st version of simplex-shaped finite element

According to Fig. 4.3, a single space-time element of the simplex shape in version 1 is shown in Fig. 4.6. The width of elements is b and the height is h. The rod elements are divided into two triangular sub-elements. Consequently, we need to derive the shape functions for each triangular separately and then combine them to obtain the shape function for the entire space-time element.

1st space-time triangular sub-element The coordinates of the first element are shown in the Table 4.1

node 1	$x = 0$	$t = 0$
node 3	$x = 0$	$t = h$
node 4	$x = b$	$t = h$

Table 4.1. Coordinates in 1st type of space-time elements and 1st triangle sub-element (Fig. 4.6).

Taking into account the data from the table and using (4.13) and (4.15) we can describe the nodal displacements for the first triangle in the following form

$$w_1 = N_{11}q_1 + N_{13}q_3 + N_{14}q_4\,, \tag{4.16}$$

where the shape functions are given by formulas

$$\begin{aligned} N_{11} &= 1 - \frac{t}{h}\,, \\ N_{13} &= \frac{t}{h} - \frac{x}{b}\,, \\ N_{14} &= \frac{x}{b}\,. \end{aligned} \tag{4.17}$$

2nd space-time triangular sub-element The coordinates of the second element are shown in the Table 4.2

node 1	$x = 0$	$t = 0$
node 2	$x = b$	$t = 0$
node 4	$x = b$	$t = h$

Table 4.2. Coordinates in 1st type of space-time elements and 2nd triangle sub-element (Fig. 4.6).

According to (4.13) and (4.15) the nodal displacements for the second triangle can be given in the following form

$$w_2 = N_{21}q_1 + N_{22}q_2 + N_{24}q_4\,, \tag{4.18}$$

where the shape functions are given by formulas

$$\begin{aligned} N_{21} &= 1 - \frac{x}{b}\,, \\ N_{22} &= \frac{x}{b} - \frac{t}{h}\,, \\ N_{24} &= \frac{t}{h}\,. \end{aligned} \tag{4.19}$$

4.1.2.2 2nd type of simplex-shaped finite element

According to Fig. 4.4, a single space-time element of the simplex shape in version 1 is shown in Fig. 4.7.

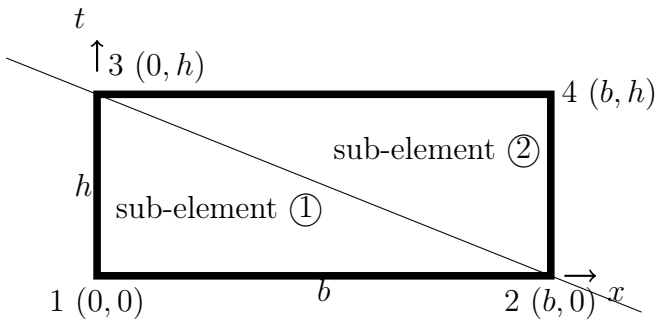


Figure 4.7: The second type of simplex-shaped discretization.

1st space-time triangular sub-element The coordinates of the first element are shown in the Table 4.3

node 1	$x = 0$	$t = 0$
node 2	$x = b$	$t = 0$
node 3	$x = 0$	$t = h$

Table 4.3. Coordinates in 2nd type of space-time elements and 1st triangle sub-element (Fig. 4.7).

According to (4.13) and (4.15) the nodal displacements for the first triangle can be written as follows

$$w_1 = N_{11}q_1 + N_{12}q_2 + N_{13}q_3\,, \tag{4.20}$$

where formulas give the shape functions

$$\begin{aligned} N_{11} &= 1 - \frac{x}{b} - \frac{t}{h}\,, \\ N_{12} &= \frac{x}{b}\,, \\ N_{13} &= \frac{t}{h}\,. \end{aligned} \tag{4.21}$$

2nd space-time triangular sub-element The coordinates of the second element are shown in the Table 4.4 According to (4.13) and (4.15) the nodal displacements for

node 2	$x = b$	$t = 0$
node 3	$x = 0$	$t = h$
node 4	$x = b$	$t = h$

Table 4.4. Coordinates in 2nd type of space-time elements and 2nd triangle sub-element (Fig. 4.7).

the second triangle can be given in the following form

$$w_2 = N_{22}q_2 + N_{23}q_3 + N_{24}q_4\,, \tag{4.22}$$

where formulas give the shape functions

$$\begin{aligned} N_{22} &= 1 - \frac{t}{h}\,, \\ N_{23} &= 1 - \frac{x}{b}\,, \\ N_{24} &= \frac{x}{b} + \frac{t}{h} - 1\,. \end{aligned} \tag{4.23}$$

4.1.3 Characteristic matrices

This subsection will develop characteristic matrices that provide a space-time representation of the rod structure. By multiplying the motion equation (4.9) by virtual displacements and integrating over the area defined by the space-time triangle, we derive the time-work equation. This process involves considering how the virtual displacements interact with the equations governing motion. Following this, we apply classical minimization techniques to optimize the time-work equation with respect to the virtual values we introduced. As a result of this minimization, we obtain the linear stiffness matrix $\boldsymbol{K}$, which reflects the structural response to deformation, and the inertia matrix $\boldsymbol{M}$, which represents the mass distribution characteristics of the system. Furthermore, each local characteristic matrix is constructed by summing two matrices that are calculated using the appropriate shape functions, as depicted in Fig. 4.5.

4.1.3.1 The stiffness matrix

Based on the standard representations of characteristic matrices (3.8), the linear stiffness matrix can be expressed in the following integral form

$$\mathbf{K}_{ij} = \int_S \mathbf{B}_i^T E \mathbf{B}_j \, \mathrm{d}S \,. \tag{4.24}$$

The $\boldsymbol{B}$ matrices characterize the linear component of the strains within the element. We establish the structure of the matrix $\boldsymbol{B}$ by utilizing differential operators on the relevant shape functions

$$\mathbf{B}_i^T = \frac{\partial N_i}{\partial x} \,, \tag{4.25}$$

$$\mathbf{B}_j = \frac{\partial N_j}{\partial x} \,. \tag{4.26}$$

If the triangle is defined by 3 points $(x_1,\, t_1)$, $(x_2,\, t_2)$ and $(x_3,\, t_3)$ and the origin of the coordinates is taken at the centroid, then integration over the area of the triangle can be calculated as the following determinant

$$S_e = \frac{1}{2} \begin{vmatrix} x_1 & t_1 & 1 \\ x_2 & t_2 & 1 \\ x_3 & t_3 & 1 \end{vmatrix} . \tag{4.27}$$

Hence, the process of integration (4.24) can be summarized using the following notation

$$\mathbf{K}_{ij} = E S_e \, \mathbf{B}_i^T \, \mathbf{B}_j \,. \tag{4.28}$$

1st type of simplex-shaped finite element Considering the shape functions of the first triangle (4.17) and the second triangle (4.19), the vectors $\boldsymbol{B}$ in this situation appears as follows

$$\mathbf{B_1} = \begin{bmatrix} 0 & -\frac{1}{b} & \frac{1}{b} \end{bmatrix}, \tag{4.29}$$

$$\mathbf{B_2} = \begin{bmatrix} -\frac{1}{b} & \frac{1}{b} & 0 \end{bmatrix}. \tag{4.30}$$

Substituting (4.29) and (4.30) into (4.28) gave the stiffness matrices of both sub-elements

$$\mathbf{K_1} = \frac{ES_eh}{2b} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{bmatrix}, \tag{4.31}$$

$$\mathbf{K_2} = \frac{ES_eh}{2b} \begin{bmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{4.32}$$

Finally, the stiffness matrix of the first type of simplex element is the sum of (4.31) and (4.32)

$$\mathbf{K} = \frac{ES_eh}{2b} \begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}. \tag{4.33}$$

2nd type of simplex-shaped finite element Similarly, taking into account the shape functions of the first triangle (4.21) and the second triangle (4.23), the vectors $\boldsymbol{B}$ in this context are represented as follows

$$\mathbf{B_1} = \begin{bmatrix} -\frac{1}{b} & \frac{1}{b} & 0 \end{bmatrix}, \tag{4.34}$$

$$\mathbf{B_2} = \begin{bmatrix} 0 & -\frac{1}{b} & \frac{1}{b} \end{bmatrix}. \tag{4.35}$$

Substituting (4.34) and (4.35) into (4.28) we obtained the stiffness matrices for both sub-elements

$$\mathbf{K_1} = \frac{ES_eh}{2b} \begin{bmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \tag{4.36}$$

$$\mathbf{K_2} = \frac{ES_eh}{2b} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{bmatrix}. \tag{4.37}$$

Ultimately, the stiffness matrix for the second type of simplex element results from the combination of (4.36) and (4.37)

$$\mathbf{K} = \frac{ES_e h}{2b} \begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix} . \tag{4.38}$$

The stiffness matrix (4.33) and (4.38) have the same form.

4.1.3.2 The inertia matrix

Considering the standard representations of the characteristic matrices (3.8), the inertia matrix can be expressed in the following manner

$$\mathbf{M}_{ij} = \int_S \mathbf{C}_i^T \rho \, \mathbf{C}_j \, \mathrm{d}S \, , \tag{4.39}$$

where the matrix $\boldsymbol{C}$ describes the velocities in the finite element and is as follows

$$\mathbf{C}_i^T = \frac{\partial N_i}{\partial t} \, , \tag{4.40}$$

$$\mathbf{C}_j = \frac{\partial N_j}{\partial t} \, . \tag{4.41}$$

Since the derivative of the shape function (4.10) with respect to time is constant, the integral (4.39) reduces to a matrix product

$$\mathbf{M}_{ij} = \rho S_e \, \mathbf{C}_i^T \, \mathbf{C}_j \, , \tag{4.42}$$

where S_e is the space-time area of the triangle and we can calculate it using the determinant (4.27).

1st type of simplex-shaped finite element Taking into account the shape functions of the first triangle (4.17) and the second triangle (4.19), the vectors $\boldsymbol{C}$ in this context are represented as follows

$$\mathbf{C_1} = \begin{bmatrix} -\frac{1}{h} & \frac{1}{h} & 0 \end{bmatrix} , \tag{4.43}$$

$$\mathbf{C_2} = \begin{bmatrix} 0 & -\frac{1}{h} & \frac{1}{h} \end{bmatrix} . \tag{4.44}$$

Replacing (4.43) and (4.44) in (4.42) resulted in the inertia matrices for both sub-elements

$$\mathbf{M_1} = \frac{\rho S_e b}{2h} \begin{bmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \tag{4.45}$$

$$\mathbf{M_2} = \frac{\rho S_e b}{2h} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{bmatrix}. \tag{4.46}$$

Finally, the inertia matrix for the first type of simplex element is derived from the combination of (4.45) and (4.46)

$$\mathbf{M} = \frac{\rho S_e b}{2h} \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix}. \tag{4.47}$$

2nd type of simplex-shaped finite element Considering the shape functions of the first triangle (4.21) and the second triangle (4.23), the vectors $\boldsymbol{C}$ in this context are depicted as follows

$$\mathbf{C_1} = \begin{bmatrix} -\frac{1}{h} & 0 & \frac{1}{h} \end{bmatrix}, \tag{4.48}$$

$$\mathbf{C_2} = \begin{bmatrix} -\frac{1}{h} & 0 & \frac{1}{h} \end{bmatrix}. \tag{4.49}$$

By inserting (4.48) and (4.49) into (4.42), the inertia matrices for both sub-elements can be expressed as follows

$$\mathbf{M_1} = \frac{\rho S_e b}{2h} \begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}, \tag{4.50}$$

$$\mathbf{M_2} = \frac{\rho S_e b}{2h} \begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}. \tag{4.51}$$

In conclusion, the inertia matrix for the second type of simplex element is derived from the addition of (4.50) and (4.51)

$$\mathbf{M} = \frac{\rho S_e b}{2h} \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix}. \tag{4.52}$$

4.1.4 Example

Let us consider a rod of length L and uniform cross-sectional area A. The rod is composed of a homogeneous, isotropic material with Young's modulus E and mass density ρ. The left end of the rod was fixed and the axial force P was applied to the right end. The diagram of the example is presented in Fig. 4.8. The following set of

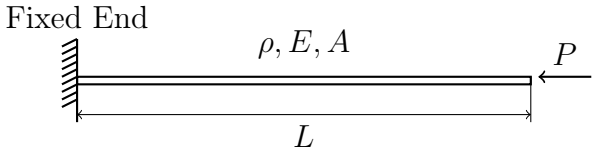


Figure 4.8: Scheme of the rod example.

dimensionless data was used:

- Length of rod $L = 5$
- Cross-Sectional area $A = 1$
- Young's modulus $E = 1$
- Mass density $\rho = 0.01$
- Axial force $P = 1$

The rod was divided into 20 finite elements and a time integration step of $h = 0.01$ was applied. Own computer programs were developed. The problem was solved using the characteristic matrices (4.33) and (4.47) describing the dynamics of the rod. The simulation results were compared with an independent solution obtained by the classical finite element method. The Newmark method was used for time integration. Fig. 4.9 and 4.10 show the results obtained in two different loading scenarios. In Fig. 4.9 we can observe the simulation results in the case of the impulse of force. The load marked as P corresponds to the force which is applied only in the first time step of the calculation. In the second case, the load P is characterized by a constant force. The results are presented in Fig.4.10. The examination of these results highlights a significant and noteworthy finding: regardless of whether the computational model utilizes the simplex-shaped space-time finite element method or the more traditional finite element technique combined with the Newmark method, the resulting data demonstrates a strong degree of correlation. This correlation suggests that both approaches yield similar outcomes, which speaks to the robustness of the analysis.

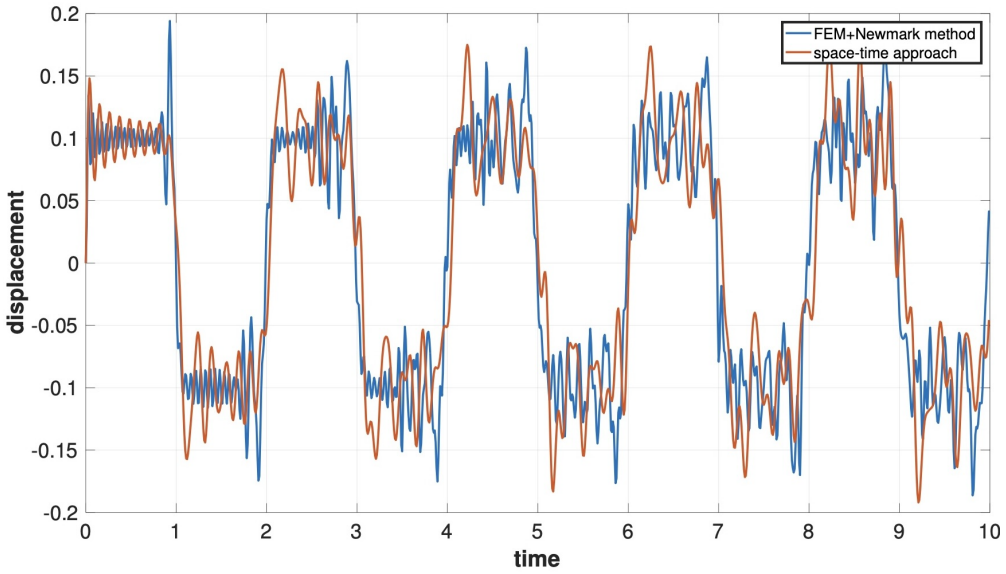


Figure 4.9: Displacement overtime at the end of a rod (the impulse of force).

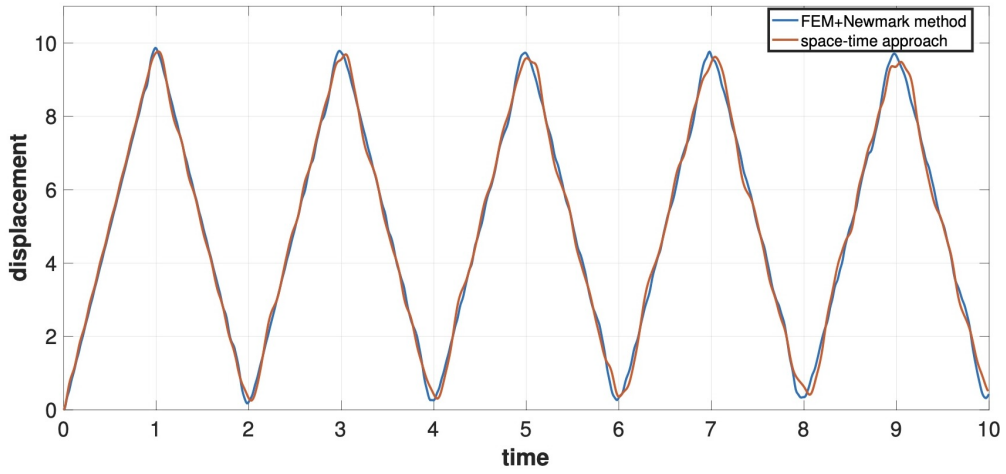


Figure 4.10: Displacement overtime at the end of a rod (a constant force).

4.2 Large deformation in plain-stress problem

4.2.1 Plane-stress problem definition

Plane stress is defined to be a state of stress within a specific plane of a material, assuming that any stress components perpendicular to that plane are effectively zero or negligible. In-plane stress analysis, we consider the body to be like a thin plate in which one dimension is much smaller than the others. Any applied loads are distributed

evenly over the entire thickness of the plate and act in the plane of the plate.

Let us consider a two-dimensional structural model considering large deformations for plane elasticity, taking into account body forces and inertia. The system of partial differential equations constituting the system of motion equations can be written in the following form

$$\begin{aligned} \frac{\partial}{\partial x}\sigma_x(x,y,t) + \frac{\partial}{\partial y}\tau_{xy}(x,y,t) + f_x &= \rho\frac{\partial^2}{\partial t^2}u(x,y,t)\ , \\ \frac{\partial}{\partial y}\sigma_y(x,y,t) + \frac{\partial}{\partial x}\tau_{xy}(x,y,t) + f_y &= \rho\,\frac{\partial^2}{\partial t^2}v(x,y,t)\ , \end{aligned} \tag{4.53}$$

where σ_x and σ_y are normal stresses and τ_{xy} is the shear stress. f_x and f_y denote the body forces per unit volume in the x and y directions, respectively, and ρ is the density of the material.

In a plane stress problem, the elastic constitutive relations refer to the mathematical equations that define how stress and strain are related in the material. These relations typically involve Hooke's Law, which states that stress is proportional to strain within the elastic limit of the material. The stress-strain relation can be given as

$$\begin{bmatrix} \sigma_x \\ \sigma_y \\ \tau_{\mathrm{xy}} \end{bmatrix} = \frac{E}{1-\nu^2}\begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1}{2}-\frac{\nu}{2} \end{bmatrix}\begin{bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{bmatrix}\ , \tag{4.54}$$

where the constitutive matrix is described by the modulus of elasticity E and Poisson's ratio ν. The ϵ_x, ϵ_y and γ_{xy} strains are the longitudinal strains along the x-axis, the longitudinal strains along the y-axis, and the shear strains, respectively.

The general two-dimensional state of strain at some point in a structure is represented by the infinitesimal element. Fig. 4.11 shows the geometric deformation of an element AOB. Point O is the original position, and after deformation, the shape arrives at $A'O'B'$. We assume the length of OA is $\mathrm{d}x$, and the length of OB is $\mathrm{d}y$. Therefore, the coordinate of points are $A(x_0+\mathrm{d}x, y_0)$ and $B(x_0, y+\mathrm{d}y)$. After deformation, points A and B move to A' and B'. Furthermore, we assume the displacement of x direction and y direction of point arbitrary are $u(x,y)$ and $v(x,y)$

$$\begin{cases} u = u(x_0,y_0)\,, \\ v = v(x_0,y_0)\,. \end{cases} \tag{4.55}$$

According to the Taylor series, the deformations of points A and B can be presented in the following forms

$$A:\begin{cases} u(x_0+\mathrm{d}x, y_0) = u(x_0,y_0) + \frac{\partial u(x_0,y_0)}{\partial x}\mathrm{d}x\,, \\ v(x_0+\mathrm{d}x, y_0) = v(x_0,y_0) + \frac{\partial v(x_0,y_0)}{\partial x}\mathrm{d}x\,, \end{cases} \tag{4.56}$$

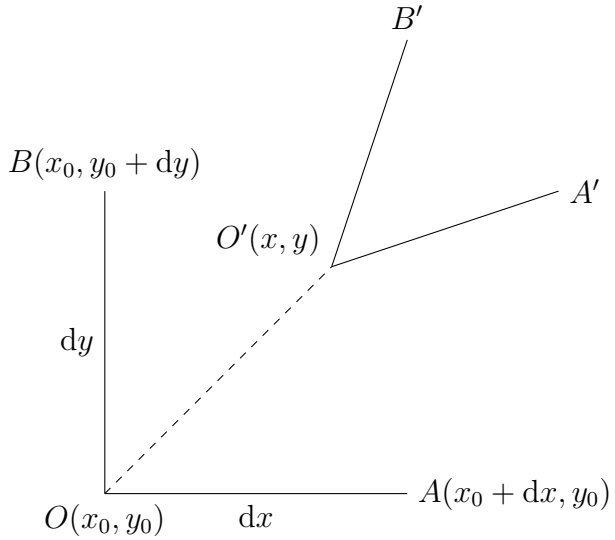


Figure 4.11: The general two-dimensional state of strain represented by infinitesimal element $\mathrm{d}x\mathrm{d}y$.

$$B : \begin{cases} u(x_0, y_0 + \mathrm{d}y) = u(x_0, y_0) + \frac{\partial u(x_0, y_0)}{\partial y}\mathrm{d}y\,, \\ v(x_0, y_0 + \mathrm{d}y) = v(x_0, y_0) + \frac{\partial v(x_0, y_0)}{\partial y}\mathrm{d}y\,. \end{cases} \tag{4.57}$$

The coordinates of points O', A', B' are

$$\begin{aligned} &O'(x_0 + u(x_0, y_0), y_0 + v(x_0, y_0)), \\ &A'\left(x_0 + \mathrm{d}x + u(x_0, y_0) + \frac{\partial u(x_0, y_0)}{\partial x}\mathrm{d}x, y_0 + v(x_0, y_0) + \frac{\partial v(x_0, y_0)}{\partial x}\mathrm{d}x\right), \\ &B'\left(x_0 + u(x_0, y_0) + \frac{\partial u(x_0, y_0)}{\partial y}\mathrm{d}y, y_0 + \mathrm{d}y + v(x_0, y_0) + \frac{\partial v(x_0, y_0)}{\partial y}\mathrm{d}y\right). \end{aligned} \tag{4.58}$$

Now, we can calculate the length of $O'A'$ and $O'B'$. It is worth noting that since $\mathrm{d}x$ and $\mathrm{d}y$ are infinitesimal, some approximate calculations will be used in the following

calculations

$$
\begin{aligned}
O'A' &= \sqrt{\left(\mathrm{d}x + \frac{\partial u(x_0,y_0)}{\partial x}\mathrm{d}x\right)^2 + \left(\frac{\partial v(x_0,y_0)}{\partial x}\mathrm{d}x\right)^2} \\
&= \mathrm{d}x\sqrt{\left(1 + \frac{\partial u(x_0,y_0)}{\partial x}\right)^2 + \left(\frac{\partial v(x_0,y_0)}{\partial x}\right)^2} \\
&= \mathrm{d}x\sqrt{1 + 2\frac{\partial u(x_0,y_0)}{\partial x} + \left(\frac{\partial v(x_0,y_0)}{\partial x}\right)^2} \\
&= \mathrm{d}x\left(1 + \frac{\partial u(x_0,y_0)}{\partial x} + \frac{1}{2}\left(\frac{\partial v(x_0,y_0)}{\partial x}\right)^2\right),
\end{aligned}
\tag{4.59}
$$

$$
\begin{aligned}
O'B' &= \sqrt{\left(\frac{\partial u(x_0,y_0)}{\partial y}\mathrm{d}y\right)^2 + \left(\mathrm{d}y + \frac{\partial v(x_0,y_0)}{\partial y}\mathrm{d}y\right)^2} \\
&= \mathrm{d}y\sqrt{\left(\frac{\partial u(x_0,y_0)}{\partial y}\right)^2 + \left(1 + \frac{\partial v(x_0,y_0)}{\partial y}\right)^2} \\
&= \mathrm{d}y\sqrt{1 + 2\frac{\partial v(x_0,y_0)}{\partial y} + \left(\frac{\partial u(x_0,y_0)}{\partial y}\right)^2} \\
&= \mathrm{d}y\left(1 + \frac{\partial v(x_0,y_0)}{\partial y} + \frac{1}{2}\left(\frac{\partial u(x_0,y_0)}{\partial y}\right)^2\right).
\end{aligned}
\tag{4.60}
$$

Taking into account (4.59) and (4.60), the longitudinal strains in the x and y directions can be written as follows

$$
\epsilon_x = \frac{O'A' - OA}{OA} = \frac{\partial u(x_0,y_0)}{\partial x} + \frac{1}{2}\left(\frac{\partial v(x_0,y_0)}{\partial x}\right)^2, \tag{4.61}
$$

$$
\epsilon_y = \frac{O'B' - OB}{OB} = \frac{\partial v(x_0,y_0)}{\partial y} + \frac{1}{2}\left(\frac{\partial u(x_0,y_0)}{\partial y}\right)^2. \tag{4.62}
$$

Finally, the relationship between displacements and strains can be written in the following form

$$
\begin{aligned}
\epsilon_x &= \frac{\partial u}{\partial x} + \frac{1}{2}\left(\frac{\partial v}{\partial x}\right)^2, \\
\epsilon_y &= \frac{\partial v}{\partial y} + \frac{1}{2}\left(\frac{\partial u}{\partial y}\right)^2, \\
\tau_{xy} &= \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}.
\end{aligned}
\tag{4.63}
$$

The system of motion equations (4.53), the stress-strain relation (4.54), and the displacements-strains relation (4.63) provide a complete description of the problem.

4.2.2 Simplex-shaped functions

According to Chapter 3, the space-time description of a considered structure is always one dimension larger than space. Therefore, a spatial two-dimensional structure is represented by a three-dimensional space-time element (Fig. 4.12). Additionally, in the

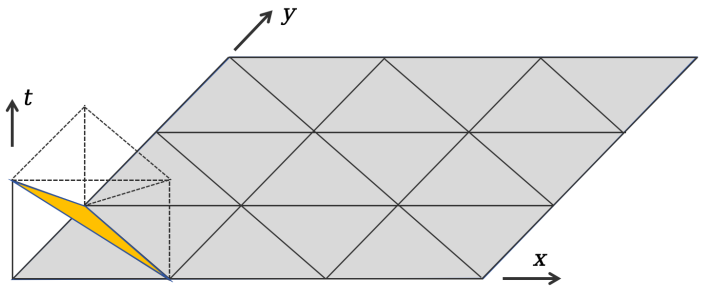


Figure 4.12: A 2-dimensional problem.

case of three-dimensional symplectic elements, space-time is divided into 3 tetrahedral-shaped elements as shown in Fig. 4.13.

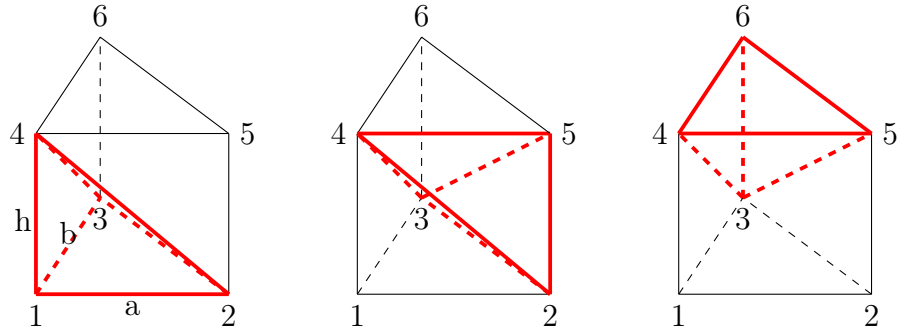


Figure 4.13: Simplex-shaped space-time sub-elements in a two-dimensional space problem.

Due to the displacement as the function of x, y and t, if we define u as the displacement along the x-axis, and v is the displacement along the y-axis. In linear

interpolations, they can be written as

$$\begin{aligned} u(x,y,t) &= a_1 x + a_2 y + a_3 t + a_4 \,, \\ v(x,y,t) &= b_1 x + b_2 y + b_3 t + b_4 \,. \end{aligned} \tag{4.64}$$

The above Eq (4.64) also can be written as matrix form

$$\begin{aligned} u(x,y,t) &= \boldsymbol{g}\,\boldsymbol{a} \,, \\ v(x,y,t) &= \boldsymbol{g}\,\boldsymbol{b} \,, \end{aligned} \tag{4.65}$$

where $\boldsymbol{g} = [x, y, t, 1]$, $\boldsymbol{a} = [a_1, a_2, a_3, a_4]^T$ and $\boldsymbol{b} = [b_1, b_2, b_3, b_4]^T$. In fact, the coefficients a_1, a_2, a_3, a_4 and b_1, b_2, b_3, b_4 are the desired shape functions.

Let denote $\boldsymbol{g}_i$ as a value of vector $\boldsymbol{g}$ at point x_i, y_i and $t_i (i = 1, 2, 3, 4)$

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} \boldsymbol{g}_1\,\boldsymbol{a} \\ \boldsymbol{g}_2\,\boldsymbol{a} \\ \boldsymbol{g}_3\,\boldsymbol{a} \\ \boldsymbol{g}_4\,\boldsymbol{a} \end{bmatrix} \,, \tag{4.66}$$

and

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} = \begin{bmatrix} \boldsymbol{g}_1\,\boldsymbol{b} \\ \boldsymbol{g}_2\,\boldsymbol{b} \\ \boldsymbol{g}_3\,\boldsymbol{b} \\ \boldsymbol{g}_4\,\boldsymbol{b} \end{bmatrix} \,. \tag{4.67}$$

New symbols of nodal displacements that simplify the algorithmization of calculations will be introduced. Fig. 4.14 shows an arbitrary tetrahedral element. Nodes 1, 2, 3 and 4 have the appropriate (x_1, y_1, t_1), (x_2, y_2, t_2), (x_3, y_3, t_3) and (x_4, y_4, t_4) coordinates. We assume that the corresponding displacements of these nodes are q_1, q_2, q_3, q_4, q_5, q_6, q_7 and q_8. The displacements q_1, q_3, q_5, q_7 are located along the x axis, and the displacements q_2, q_4, q_6, q_8 are along the y axis. For the four nodes of this tetrahedron element, the equation (4.64) is satisfied and is $u(x_1, y_1, t_1) = q_1$, $v(x_1, y_1, t_1) = q_2$, $u(x_2, y_2, t_2) = q_3$, $v(x_2, y_2, t_2) = q_4$, $u(x_3, y_3, t_3) = q_5$, $v(x_3, y_3, t_3) = q_6$, $u(x_4, y_4, t_4) = q_7$ and $v(x_4, y_4, t_4) = q_8$. In accordance with (4.66) and (4.67) and the introduced notation, we obtain the following linear distribution of nodal values in the direction of the x and y axes in the following form

$$\begin{bmatrix} q_1 \\ q_3 \\ q_5 \\ q_7 \end{bmatrix} = \boldsymbol{G}\,\boldsymbol{a} \,, \tag{4.68}$$

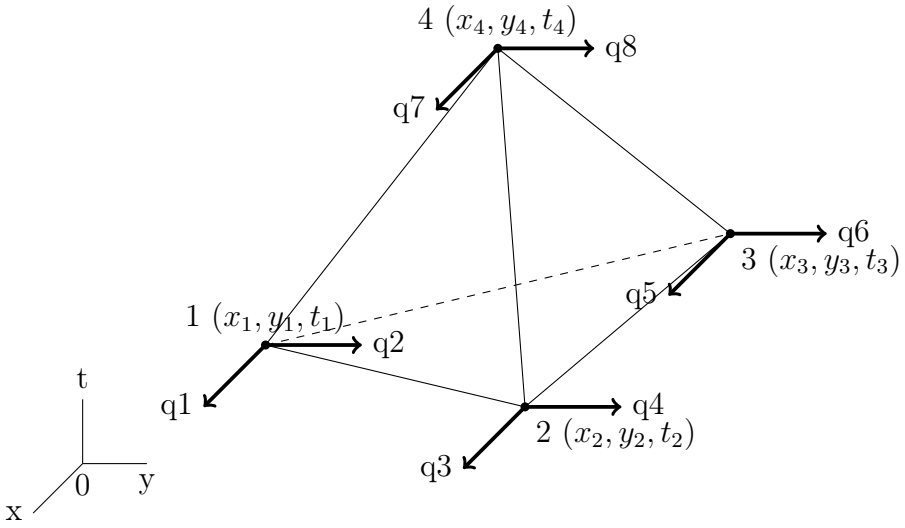


Figure 4.14: An arbitrary tetrahedral element

$$\begin{bmatrix} q_2 \\ q_4 \\ q_6 \\ q_8 \end{bmatrix} = \boldsymbol{G}\,\boldsymbol{b}\,, \tag{4.69}$$

where

$$\boldsymbol{G} = \begin{bmatrix} x_1 & y_1 & t_1 & 1 \\ x_2 & y_2 & t_2 & 1 \\ x_3 & y_3 & t_3 & 1 \\ x_4 & y_4 & t_4 & 1 \end{bmatrix}. \tag{4.70}$$

As a result of simple matrix operations, (4.68) and (4.69) can be presented in the following form

$$\boldsymbol{a} = \boldsymbol{G}^{-1} \begin{bmatrix} q_1 \\ q_3 \\ q_5 \\ q_7 \end{bmatrix}, \tag{4.71}$$

$$\boldsymbol{b} = \boldsymbol{G}^{-1} \begin{bmatrix} q_2 \\ q_4 \\ q_6 \\ q_8 \end{bmatrix}. \tag{4.72}$$

Determining the shape function requires only inverting the $\boldsymbol{G}$ matrix for a selected set of element nodal coordinates. Finally, taking into account (4.71) and (4.72), the

displacements relative to the x-axis and the y-axis of the tetrahedral element can be described by nodal values as follows

$$u = \boldsymbol{g}\,\boldsymbol{G}^{-1} \begin{bmatrix} q_1 \\ q_3 \\ q_5 \\ q_7 \end{bmatrix} , \tag{4.73}$$

$$v = \boldsymbol{g}\,\boldsymbol{G}^{-1} \begin{bmatrix} q_2 \\ q_4 \\ q_6 \\ q_8 \end{bmatrix} . \tag{4.74}$$

Now we can explicitly compute the form of the shape functions for each of the three elements in Fig. 4.13.

1st tetrahedral element According to Fig. 4.13, the first tetrahedral element has the vertex coordinates as in Table 4.5. Based on (4.70), the matrix $\boldsymbol{G}$ is given as follows

node 1	$x = 0$	$y = 0$	$t = 0$
node 2	$x = 0$	$y = b$	$t = 0$
node 3	$x = a$	$y = b$	$t = 0$
node 4	$x = 0$	$y = 0$	$t = h$

Table 4.5. Space-time coordinates in 1st subelement (Fig. 4.13).

$$\boldsymbol{G} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & b & 0 & 1 \\ a & b & 0 & 1 \\ 0 & 0 & h & 1 \end{bmatrix} . \tag{4.75}$$

The inverse matrix takes the following form

$$\boldsymbol{G}^{-1} = \begin{bmatrix} 0 & -1/a & 1/a & 0 \\ -1/b & 1/b & 0 & 0 \\ -1/h & 0 & 0 & 1/h \\ 1 & 0 & 0 & 0 \end{bmatrix} . \tag{4.76}$$

Taking into account (4.73) and (4.74), the linear distribution of nodal displacements in the first tetrahedron element is as follows

$$\begin{aligned} u_1 &= N_{11}\,q_1 + N_{12}\,q_3 + N_{13}\,q_5 + N_{14}\,q_7\,, \\ v_1 &= N_{11}\,q_2 + N_{12}\,q_4 + N_{13}\,q_6 + N_{14}\,q_8\,, \end{aligned} \tag{4.77}$$

where, the shape functions are given by formulas

$$\begin{aligned} N_{11} &= 1 - \frac{y}{b} - \frac{t}{h}\,, \\ N_{12} &= \frac{y}{b} - \frac{x}{a}\,, \\ N_{13} &= \frac{y}{b}\,, \\ N_{14} &= \frac{y}{b} + \frac{t}{h}\,. \end{aligned} \tag{4.78}$$

2nd tetrahedral element According to Fig. 4.13, the second tetrahedral element has the vertex coordinates as in Table 4.6. Based on (4.70), the matrix $\boldsymbol{G}$ is given as

node 2	$x = 0$	$y = b$	$t = 0$
node 3	$x = a$	$y = b$	$t = 0$
node 4	$x = 0$	$y = 0$	$t = h$
node 5	$x = 0$	$y = b$	$t = h$

Table 4.6. Space-time coordinates in 2nd subelement (Fig. 4.13).

follows

$$\boldsymbol{G} = \begin{bmatrix} 0 & b & 0 & 1 \\ a & b & 0 & 1 \\ 0 & 0 & h & 1 \\ 0 & b & h & 1 \end{bmatrix}. \tag{4.79}$$

The inverse matrix takes the following form

$$\boldsymbol{G}^{-1} = \begin{bmatrix} -1/a & 1/a & 0 & 0 \\ 0 & 0 & -1/b & 1/b \\ -1/h & 0 & 0 & 1/h \\ 1 & 0 & 1 & -1 \end{bmatrix}. \tag{4.80}$$

Taking into account (4.73) and (4.74), the linear distribution of nodal displacements in the second tetrahedron element is as follows

$$\begin{aligned} u_2 &= N_{21}\, q_1 + N_{22}\, q_3 + N_{23}\, q_5 + N_{24}\, q_7\,, \\ v_2 &= N_{21}\, q_2 + N_{22}\, q_4 + N_{23}\, q_6 + N_{24}\, q_8\,, \end{aligned} \tag{4.81}$$

where, the shape functions are given by formulas

$$\begin{aligned} N_{21} &= 1 - \frac{t}{h} - \frac{x}{a}\,, \\ N_{22} &= \frac{x}{a}\,, \\ N_{23} &= 1 - \frac{y}{b}\,, \\ N_{24} &= \frac{y}{b} + \frac{t}{h} - 1\,. \end{aligned} \tag{4.82}$$

3rd tetrahedral element According to Fig. 4.13, the third tetrahedral element has the vertex coordinates as in Table 4.7. Based on (4.70), the matrix $\boldsymbol{G}$ is given as follows

node 3	$x = a$	$y = b$	$t = 0$
node 4	$x = 0$	$y = 0$	$t = h$
node 5	$x = 0$	$y = b$	$t = h$
node 6	$x = a$	$y = b$	$t = h$

Table 4.7. Space-time coordinates in 3rd subelement (Fig. 4.13).

$$\boldsymbol{G} = \begin{bmatrix} a & b & 0 & 1 \\ 0 & 0 & h & 1 \\ 0 & b & h & 1 \\ a & b & h & 1 \end{bmatrix}\,. \tag{4.83}$$

The inverse matrix takes the following form

$$\boldsymbol{G}^{-1} = \begin{bmatrix} 0 & 0 & -1/a & 1/a \\ 0 & -1/b & 1/b & 0 \\ -1/h & 0 & 0 & 1/h \\ 1 & 1 & 0 & -1 \end{bmatrix}\,. \tag{4.84}$$

Taking into account (4.73) and (4.74), the linear distribution of nodal displacements in the second tetrahedron element is as follows

$$\begin{aligned} u_{(3)} &= N_{31}\, q_1 + N_{32}\, q_3 + N_{33}\, q_5 + N_{34}\, q_7\,, \\ v_{(3)} &= N_{31}\, q_2 + N_{32}\, q_4 + N_{33}\, q_6 + N_{34}\, q_8\,, \end{aligned} \tag{4.85}$$

where, the shape functions are given by formulas

$$\begin{aligned} N_{31} &= 1 - \frac{t}{h}\,, \\ N_{32} &= 1 - \frac{y}{b}\,, \\ N_{33} &= \frac{y}{b} - \frac{x}{a}\,, \\ N_{34} &= \frac{x}{a} + \frac{t}{h} - 1\,. \end{aligned} \tag{4.86}$$

4.2.3 Characteristic matrices

In this subsection, characteristic matrices will be derived, which constitute a space-time description of the plane stress state problem. By multiplying the system of motion equations (4.53) by virtual displacements and integrating them into the volume of the space-time tetrahedron, the time-work equation was obtained. As a result of classical minimization of time-work with respect to virtual values, we obtain a linear stiffness matrix $\boldsymbol{K}$, a non-linear stiffness matrix $\boldsymbol{K}_N$ and an inertia matrix $\boldsymbol{M}$, respectively. Each local characteristic matrix is the sum of 3 matrices calculated using the appropriate shape functions according to Fig. 4.13.

4.2.3.1 The linear stiffness matrix

According to general forms of the characteristic matrices (3.8), the linear stiffness matrix takes the following integral form

$$\boldsymbol{K}_{ij} = \int_V \boldsymbol{B}_i^T \boldsymbol{D} \boldsymbol{B}_j \,\mathrm{d}V\,. \tag{4.87}$$

The $\boldsymbol{B}$ matrices represent the linear part of the strains in the element. We determine the form of the matrix $\boldsymbol{B}$ by applying differential operators to the appropriate shape functions

$$\boldsymbol{B}_i^T = \begin{bmatrix} \frac{\partial N_i}{\partial x} & 0 & \frac{\partial N_i}{\partial y} \\ 0 & \frac{\partial N_i}{\partial y} & \frac{\partial N_i}{\partial x} \end{bmatrix}, \tag{4.88}$$

$$\boldsymbol{B}_j = \begin{bmatrix} \frac{\partial N_j}{\partial x} & 0 \\ 0 & \frac{\partial N_j}{\partial y} \\ \frac{\partial N_j}{\partial y} & \frac{\partial N_j}{\partial x} \end{bmatrix}. \tag{4.89}$$

Taking into account the stress-strain relation (4.54) the constitutive matrix $\boldsymbol{D}$ can be written in the following form

$$\boldsymbol{D} = \frac{E}{1-\nu^2}\begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1}{2}-\frac{\nu}{2} \end{bmatrix}. \tag{4.90}$$

Since we are considering linear shape functions (4.64), the matrices (4.88) and (4.89) are constant.

If the tetrahedron is defined by 4 points (x_1, y_1, t_1), (x_2, y_2, t_2), (x_3, y_3, t_3) and (x_4, y_4, t_4) and the origin of the coordinates is taken at the centroid, then integration over the volume of the tetrahedron can be calculated as the following determinant

$$V_e = \frac{1}{6}\begin{vmatrix} x_1 & y_1 & t_1 & 1 \\ x_2 & y_2 & t_2 & 1 \\ x_3 & y_3 & t_3 & 1 \\ x_4 & y_4 & t_4 & 1 \end{vmatrix}. \tag{4.91}$$

Therefore, integration (4.87) comes down to the following notation

$$\boldsymbol{K}_{ij} = V_e\, \boldsymbol{B}_i^T \boldsymbol{D} \boldsymbol{B}_j\,. \tag{4.92}$$

Details on tetrahedron integration can be found in the book [108].

1st tetrahedral element Taking into account the shape functions (4.78), the matrix $\boldsymbol{B}$ in this case looks as follows

$$\boldsymbol{B}_1 = \begin{pmatrix} 0 & 0 & -\frac{1}{a} & 0 & \frac{1}{a} & 0 & 0 & 0 \\ 0 & -\frac{1}{b} & 0 & \frac{1}{b} & 0 & 0 & 0 & 0 \\ -\frac{1}{b} & 0 & \frac{1}{b} & -\frac{1}{a} & 0 & \frac{1}{a} & 0 & 0 \end{pmatrix}. \tag{4.93}$$

According to (4.92), the stiffness matrix of the linear part corresponding to the first space-time tetrahedron (Fig. 4.13) can be presented in the following form

$$\boldsymbol{K}_1 = V_1\, \boldsymbol{B}_1^T \boldsymbol{D} \boldsymbol{B}_1 =$$

$$V_1 \begin{pmatrix} \frac{E\,a\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,b\left(\nu^2-1\right)} & 0 & -\frac{E\,a\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,b\left(\nu^2-1\right)} & \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} \\ 0 & -\frac{E\,a\,h}{6\,b\left(\nu^2-1\right)} & -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & \frac{E\,a\,h}{6\,b\left(\nu^2-1\right)} \\ -\frac{E\,a\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,b\left(\nu^2-1\right)} & -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & \frac{E\,a\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,b\left(\nu^2-1\right)} - \frac{E\,b\,h}{6\,a\left(\nu^2-1\right)} & \frac{E\,h\,\nu}{6\left(\nu^2-1\right)} - \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} \\ \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & \frac{E\,a\,h}{6\,b\left(\nu^2-1\right)} & \frac{E\,h\,\nu}{6\left(\nu^2-1\right)} - \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & \frac{E\,b\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,a\left(\nu^2-1\right)} - \frac{E\,a\,h}{6\,b\left(\nu^2-1\right)} \\ 0 & \frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & \frac{E\,b\,h}{6\,a\left(\nu^2-1\right)} & -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} \\ -\frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & 0 & \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & -\frac{E\,b\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,a\left(\nu^2-1\right)} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{4.94}$$

$$\begin{pmatrix} 0 & -\frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} \\ \frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & 0 \\ \frac{E\,b\,h}{6\,a\left(\nu^2-1\right)} & \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} \\ -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & -\frac{E\,b\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,a\left(\nu^2-1\right)} \\ -\frac{E\,b\,h}{6\,a\left(\nu^2-1\right)} & 0 \\ 0 & \frac{E\,b\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,a\left(\nu^2-1\right)} \\ 0 & 0 \\ 0 & 0 \end{pmatrix}.$$

2nd tetrahedral element Taking into account the shape functions (4.82), the matrix $\boldsymbol{B}$ in this case looks as follows

$$\boldsymbol{B}_2 = \begin{pmatrix} -\frac{1}{a} & 0 & \frac{1}{a} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -\frac{1}{b} & 0 & \frac{1}{b} \\ 0 & -\frac{1}{a} & 0 & \frac{1}{a} & -\frac{1}{b} & 0 & \frac{1}{b} & 0 \end{pmatrix}. \tag{4.95}$$

According to (4.92), the stiffness matrix of the linear part corresponding to the second space-time tetrahedron (Fig. 4.13) can be presented in the following form

$$\boldsymbol{K}_2 = V_2\, \boldsymbol{B}_2^T \boldsymbol{D} \boldsymbol{B}_2 =$$

$$V_2 \begin{pmatrix} -\frac{E\,b\,h}{6\,a\left(\nu^2-1\right)} & 0 & \frac{E\,b\,h}{6\,a\left(\nu^2-1\right)} & 0 \\ 0 & \frac{E\,b\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,a\left(\nu^2-1\right)} & 0 & -\frac{E\,b\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,a\left(\nu^2-1\right)} \\ \frac{E\,b\,h}{6\,a\left(\nu^2-1\right)} & 0 & -\frac{E\,b\,h}{6\,a\left(\nu^2-1\right)} & 0 \\ 0 & -\frac{E\,b\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,a\left(\nu^2-1\right)} & 0 & \frac{E\,b\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,a\left(\nu^2-1\right)} \\ 0 & \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & 0 & -\frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} \\ -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & 0 & \frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & 0 \\ 0 & -\frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & 0 & \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} \\ \frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & 0 & -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & 0 \end{pmatrix}$$

$$\begin{pmatrix} 0 & -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} \\ \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & 0 \\ 0 & \frac{E\,h\,\nu}{6\left(\nu^2-1\right)} \\ -\frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & 0 \\ \frac{E\,a\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,b\left(\nu^2-1\right)} & 0 \\ 0 & -\frac{E\,a\,h}{6\,b\left(\nu^2-1\right)} \\ -\frac{E\,a\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,b\left(\nu^2-1\right)} & 0 \\ 0 & \frac{E\,a\,h}{6\,b\left(\nu^2-1\right)} \end{pmatrix}. \tag{4.96}$$

3rd tetrahedral element Taking into account the shape functions (4.86), the matrix $\boldsymbol{B}$ in this case looks as follows

$$\boldsymbol{B}_3 = \begin{pmatrix} 0 & 0 & 0 & 0 & -\frac{1}{a} & 0 & \frac{1}{a} & 0 \\ 0 & 0 & 0 & -\frac{1}{b} & 0 & \frac{1}{b} & 0 & 0 \\ 0 & 0 & -\frac{1}{b} & 0 & \frac{1}{b} & -\frac{1}{a} & 0 & \frac{1}{a} \end{pmatrix}. \tag{4.97}$$

According to (4.92), the stiffness matrix of the linear part corresponding to the third space-time tetrahedron (Fig. 4.13) can be presented in the following form

$$\boldsymbol{K}_3 = V_3\, \boldsymbol{B}_3^T \boldsymbol{D} \boldsymbol{B}_3 =$$

$$V_3 \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{E\,a\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,b\left(\nu^2-1\right)} & 0 \\ 0 & 0 & 0 & -\frac{E\,a\,h}{6\,b\left(\nu^2-1\right)} \\ 0 & 0 & -\frac{E\,a\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,b\left(\nu^2-1\right)} & -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} \\ 0 & 0 & \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & \frac{E\,a\,h}{6\,b\left(\nu^2-1\right)} \\ 0 & 0 & 0 & \frac{E\,h\,\nu}{6\left(\nu^2-1\right)} \\ 0 & 0 & -\frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & 0 \end{pmatrix}$$

$$\begin{pmatrix} 0 & 0 \\ 0 & 0 \\ -\frac{E\,a\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,b\left(\nu^2-1\right)} & \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} \\ -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & \frac{E\,a\,h}{6\,b\left(\nu^2-1\right)} \\ \frac{E\,a\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,b\left(\nu^2-1\right)} - \frac{E\,b\,h}{6\,a\left(\nu^2-1\right)} & \frac{E\,h\,\nu}{6\left(\nu^2-1\right)} - \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} \\ \frac{E\,h\,\nu}{6\left(\nu^2-1\right)} - \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & \frac{E\,b\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,a\left(\nu^2-1\right)} - \frac{E\,a\,h}{6\,b\left(\nu^2-1\right)} \\ \frac{E\,b\,h}{6\,a\left(\nu^2-1\right)} & -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} \\ \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & -\frac{E\,b\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,a\left(\nu^2-1\right)} \end{pmatrix}. \tag{4.98}$$

The matrices (4.94), (4.96) and (4.98) are the components of the local space-time stiffness matrix $\boldsymbol{K}$. The matrix $\boldsymbol{K}$ is the sum of these matrices. The summation is carried out in accordance with the numbering of the degrees of freedom in the nodes of the space-time finite element. Taking into account the notation of the matrix equation (3.13), matrix $\boldsymbol{K}$ can be represented using sub-matrices $\boldsymbol{A}_k$, $\boldsymbol{B}_k$, $\boldsymbol{C}_k$ and $\boldsymbol{D}_k$ in the following form

$$\boldsymbol{K} = \begin{bmatrix} \boldsymbol{A}_k & \boldsymbol{B}_k \\ \boldsymbol{C}_k & \boldsymbol{D}_k \end{bmatrix}, \tag{4.99}$$

where

$$\boldsymbol{A}_k = \begin{pmatrix} \frac{E\,a\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,b\left(\nu^2-1\right)} & 0 & -\frac{E\,a\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,b\left(\nu^2-1\right)} & \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} \\ 0 & -\frac{E\,a\,h}{6\,b\left(\nu^2-1\right)} & -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & \frac{E\,a\,h}{6\,b\left(\nu^2-1\right)} \\ -\frac{E\,a\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,b\left(\nu^2-1\right)} & -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & \frac{E\,a\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,b\left(\nu^2-1\right)}-\frac{E\,b\,h}{3\,a\left(\nu^2-1\right)} & \frac{E\,h\,\nu}{6\left(\nu^2-1\right)}-\frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} \\ \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & \frac{E\,a\,h}{6\,b\left(\nu^2-1\right)} & \frac{E\,h\,\nu}{6\left(\nu^2-1\right)}-\frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & \frac{E\,b\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{3\,a\left(\nu^2-1\right)}-\frac{E\,a\,h}{6\,b\left(\nu^2-1\right)} \\ 0 & \frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & \frac{E\,b\,h}{3\,a\left(\nu^2-1\right)} & -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} \\ -\frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & 0 & \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & -\frac{E\,b\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{3\,a\left(\nu^2-1\right)} \end{pmatrix}$$

$$\begin{pmatrix} 0 & -\frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} \\ \frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & 0 \\ \frac{E\,b\,h}{3\,a\left(\nu^2-1\right)} & \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} \\ -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & -\frac{E\,b\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{3\,a\left(\nu^2-1\right)} \\ -\frac{E\,b\,h}{3\,a\left(\nu^2-1\right)} & 0 \\ 0 & \frac{E\,b\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{3\,a\left(\nu^2-1\right)} \end{pmatrix}, \tag{4.100}$$

$$\boldsymbol{B}_k = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & 0 & \frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & 0 & 0 \\ \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & 0 & -\frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & 0 & 0 & 0 \\ 0 & \frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & 0 & -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & 0 & 0 \\ -\frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & 0 & \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & 0 & 0 & 0 \end{pmatrix}, \tag{4.101}$$

$$\boldsymbol{C}_k = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & 0 & \frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & 0 & 0 \\ \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & 0 & -\frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & 0 & 0 & 0 \\ 0 & \frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & 0 & -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & 0 & 0 \\ -\frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & 0 & \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & 0 & 0 & 0 \end{pmatrix}, \tag{4.102}$$

$$
\boldsymbol{D}_k = \left(\begin{array}{ccc}
\frac{E\,a\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{3\,b\left(\nu^2-1\right)} & 0 & -\frac{E\,a\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{3\,b\left(\nu^2-1\right)} \\
0 & -\frac{E\,a\,h}{3\,b\left(\nu^2-1\right)} & -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} \\
-\frac{E\,a\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{3\,b\left(\nu^2-1\right)} & -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & \frac{E\,a\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{3\,b\left(\nu^2-1\right)} - \frac{E\,b\,h}{6\,a\left(\nu^2-1\right)} \\
\frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & \frac{E\,a\,h}{3\,b\left(\nu^2-1\right)} & \frac{E\,h\,\nu}{6\left(\nu^2-1\right)} - \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} \\
0 & \frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & \frac{E\,b\,h}{6\,a\left(\nu^2-1\right)} \\
-\frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & 0 & \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)}
\end{array} \right.
$$

$$
\left. \begin{array}{ccc}
\frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & 0 & -\frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} \\
\frac{E\,a\,h}{3\,b\left(\nu^2-1\right)} & \frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & 0 \\
\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} - \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} & \frac{E\,b\,h}{6\,a\left(\nu^2-1\right)} & \frac{E\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\left(\nu^2-1\right)} \\
\frac{E\,b\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,a\left(\nu^2-1\right)} - \frac{E\,a\,h}{3\,b\left(\nu^2-1\right)} & -\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & -\frac{E\,b\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,a\left(\nu^2-1\right)} \\
-\frac{E\,h\,\nu}{6\left(\nu^2-1\right)} & -\frac{E\,b\,h}{6\,a\left(\nu^2-1\right)} & 0 \\
-\frac{E\,b\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,a\left(\nu^2-1\right)} & 0 & \frac{E\,b\,h\left(\frac{\nu}{2}-\frac{1}{2}\right)}{6\,a\left(\nu^2-1\right)}
\end{array} \right) . \tag{4.103}
$$

4.2.3.2 The non-linear stiffness matrix

Similarly to the linear stiffness matrix (4.87), the local stiffness matrix associated with the non-linear terms takes the following form

$$
\boldsymbol{K}_{ij}^{N} = \int_V \hat{\boldsymbol{B}}_i^T \boldsymbol{D} \hat{\boldsymbol{B}}_j \, \mathrm{d}V \, . \tag{4.104}
$$

Matrix $\hat{\boldsymbol{B}}$ describes the nonlinear components of elastic strains. According to the displacements-strains relationship (4.63) and notations (4.68) and (4.69) we obtain

$$
\hat{\boldsymbol{B}}_i^T = \frac{1}{2} \begin{bmatrix} 0 & \frac{\partial N_i}{\partial y}\frac{\partial N_i}{\partial y} q_{2i-1} & 0 \\ \frac{\partial N_i}{\partial x}\frac{\partial N_i}{\partial x} q_{2i} & 0 & 0 \end{bmatrix} , \tag{4.105}
$$

and

$$
\hat{\boldsymbol{B}}_j = \frac{1}{2} \begin{bmatrix} 0 & \frac{\partial N_j}{\partial x}\frac{\partial N_j}{\partial x} q_{2j} \\ \frac{\partial N_j}{\partial y}\frac{\partial N_j}{\partial y} q_{2j-1} & 0 \\ 0 & 0 \end{bmatrix} . \tag{4.106}
$$

Taking into account the assumptions made earlier regarding the integration of a constant matrix in a volume defined by 4 points, the integral (4.104) can be written in the following form

$$\boldsymbol{K}_{ij}^{N} = V_e \, \hat{\boldsymbol{B}}_i^T \boldsymbol{D} \hat{\boldsymbol{B}}_j \,, \tag{4.107}$$

where the space-time volume V_e can be calculated using the determinant (4.91).

1st tetrahedral element Based on the shape function (4.78), the matrix (4.106) describing the 4 nodes of the first space-time tetrahedron looks as follows

$$\hat{\boldsymbol{B}}_1 = \begin{pmatrix} 0 & 0 & 0 & \frac{q_4-q_6}{a^2} & 0 & -\frac{q_4-q_6}{a^2} & 0 & 0 \\ \frac{q_1-q_3}{b^2} & 0 & -\frac{q_1-q_3}{b^2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{4.108}$$

According to (4.108) and (4.107), the stiffness matrix for the first tetrahedron can be presented in the following form

$$\boldsymbol{K}_1^N = V_1 \begin{pmatrix} -\frac{E\,a\,h\left(\frac{q_1}{b}-\frac{q_3}{b}\right)^2}{12\,b\left(\nu^2-1\right)} & 0 & \frac{E\,a\,h\left(\frac{q_1}{b}-\frac{q_3}{b}\right)^2}{12\,b\left(\nu^2-1\right)} & -\frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_1}{b}-\frac{q_3}{b}\right)}{12\left(\nu^2-1\right)} \\ 0 & 0 & 0 & 0 \\ \frac{E\,a\,h\left(\frac{q_1}{b}-\frac{q_3}{b}\right)^2}{12\,b\left(\nu^2-1\right)} & 0 & -\frac{E\,a\,h\left(\frac{q_1}{b}-\frac{q_3}{b}\right)^2}{12\,b\left(\nu^2-1\right)} & \frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_1}{b}-\frac{q_3}{b}\right)}{12\left(\nu^2-1\right)} \\ -\frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_1}{b}-\frac{q_3}{b}\right)}{12\left(\nu^2-1\right)} & 0 & \frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_1}{b}-\frac{q_3}{b}\right)}{12\left(\nu^2-1\right)} & -\frac{E\,b\,h\left(\frac{q_4}{a}-\frac{q_6}{a}\right)^2}{12\,a\left(\nu^2-1\right)} \\ 0 & 0 & 0 & 0 \\ \frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_1}{b}-\frac{q_3}{b}\right)}{12\left(\nu^2-1\right)} & 0 & -\frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_1}{b}-\frac{q_3}{b}\right)}{12\left(\nu^2-1\right)} & \frac{E\,b\,h\left(\frac{q_4}{a}-\frac{q_6}{a}\right)^2}{12\,a\left(\nu^2-1\right)} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{4.109}$$

$$\begin{pmatrix} 0 & \frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_1}{b}-\frac{q_3}{b}\right)}{12\left(\nu^2-1\right)} \\ 0 & 0 \\ 0 & -\frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_1}{b}-\frac{q_3}{b}\right)}{12\left(\nu^2-1\right)} \\ 0 & \frac{E\,b\,h\left(\frac{q_4}{a}-\frac{q_6}{a}\right)^2}{12\,a\left(\nu^2-1\right)} \\ 0 & 0 \\ 0 & -\frac{E\,b\,h\left(\frac{q_4}{a}-\frac{q_6}{a}\right)^2}{12\,a\left(\nu^2-1\right)} \\ 0 & 0 \\ 0 & 0 \end{pmatrix}.$$

2nd tetrahedral element Taking into account the shape function of the second tetrahedron (4.82), the matrix (4.106) can be written in the following form

$$\hat{\boldsymbol{B}}_2 = \begin{pmatrix} 0 & \frac{q_4-q_6}{a^2} & 0 & -\frac{q_4-q_6}{a^2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{q_7-q_9}{b^2} & 0 & -\frac{q_7-q_9}{b^2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{4.110}$$

In the case of the second space-time tetrahedron, the stiffness matrix according to (4.110) and (4.107) is given by

$$\begin{aligned} \boldsymbol{K}_2^N = \\ V_2 \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & -\frac{E\,b\,h\left(\frac{q_4}{a}-\frac{q_6}{a}\right)^2}{12\,a\left(\nu^2-1\right)} & 0 & \frac{E\,b\,h\left(\frac{q_4}{a}-\frac{q_6}{a}\right)^2}{12\,a\left(\nu^2-1\right)} \\ 0 & 0 & 0 & 0 \\ 0 & \frac{E\,b\,h\left(\frac{q_4}{a}-\frac{q_6}{a}\right)^2}{12\,a\left(\nu^2-1\right)} & 0 & -\frac{E\,b\,h\left(\frac{q_4}{a}-\frac{q_6}{a}\right)^2}{12\,a\left(\nu^2-1\right)} \\ 0 & -\frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} & 0 & \frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} \\ 0 & 0 & 0 & 0 \\ 0 & \frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} & 0 & -\frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} \\ 0 & 0 & 0 & 0 \end{pmatrix} \\ \begin{pmatrix} 0 & 0 \\ -\frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} & 0 \\ 0 & 0 \\ \frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} & 0 \\ -\frac{E\,a\,h\left(\frac{q_7}{b}-\frac{q_9}{b}\right)^2}{12\,b\left(\nu^2-1\right)} & 0 \\ 0 & 0 \\ \frac{E\,a\,h\left(\frac{q_7}{b}-\frac{q_9}{b}\right)^2}{12\,b\left(\nu^2-1\right)} & 0 \\ 0 & 0 \end{pmatrix}. \end{aligned} \tag{4.111}$$

3rd tetrahedral element Using the shape function (4.86), the matrix (4.106) can be written as follows

$$\hat{\boldsymbol{B}}_3 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & \frac{q_{10}-q_{12}}{a^2} & 0 & -\frac{q_{10}-q_{12}}{a^2} \\ 0 & 0 & \frac{q_7-q_9}{b^2} & 0 & -\frac{q_7-q_9}{b^2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{4.112}$$

According to (4.112) and (4.107), the stiffness matrix for the third tetrahedron can be given as follows

$$\boldsymbol{K}_3^N = V_3 \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{E\,a\,h\left(\frac{q_7}{b}-\frac{q_9}{b}\right)^2}{12\,b\left(\nu^2-1\right)} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{E\,a\,h\left(\frac{q_7}{b}-\frac{q_9}{b}\right)^2}{12\,b\left(\nu^2-1\right)} & 0 \\ 0 & 0 & -\frac{E\,h\,\nu\left(\frac{q_{10}}{a}-\frac{q_{12}}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{E\,h\,\nu\left(\frac{q_{10}}{a}-\frac{q_{12}}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ \frac{E\,a\,h\left(\frac{q_7}{b}-\frac{q_9}{b}\right)^2}{12\,b\left(\nu^2-1\right)} & -\frac{E\,h\,\nu\left(\frac{q_{10}}{a}-\frac{q_{12}}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} \\ 0 & 0 \\ -\frac{E\,a\,h\left(\frac{q_7}{b}-\frac{q_9}{b}\right)^2}{12\,b\left(\nu^2-1\right)} & \frac{E\,h\,\nu\left(\frac{q_{10}}{a}-\frac{q_{12}}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} \\ \frac{E\,h\,\nu\left(\frac{q_{10}}{a}-\frac{q_{12}}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} & -\frac{E\,b\,h\left(\frac{q_{10}}{a}-\frac{q_{12}}{a}\right)^2}{12\,a\left(\nu^2-1\right)} \\ 0 & 0 \\ -\frac{E\,h\,\nu\left(\frac{q_{10}}{a}-\frac{q_{12}}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} & \frac{E\,b\,h\left(\frac{q_{10}}{a}-\frac{q_{12}}{a}\right)^2}{12\,a\left(\nu^2-1\right)} \end{pmatrix}. \tag{4.113}$$

The matrices (4.109), (4.111) and (4.113) are the components of the local space-time nonlinear stiffness matrix $\boldsymbol{K}^N$. Finally, matrix $\boldsymbol{K}^N$ can be presented using submatrices $\boldsymbol{A}_k^N$, $\boldsymbol{B}_k^N$, $\boldsymbol{C}_k^N$ and $\boldsymbol{D}_k^N$ as follows

$$\boldsymbol{K}^N = \begin{bmatrix} \boldsymbol{A}_k^N & \boldsymbol{B}_k^N \\ \boldsymbol{C}_k^N & \boldsymbol{D}_k^N \end{bmatrix}, \tag{4.114}$$

where

$$\boldsymbol{A}_K^N = \left(\begin{array}{ccccc} -\frac{E\,a\,h\left(\frac{q_1}{b}-\frac{q_3}{b}\right)^2}{12\,b\left(\nu^2-1\right)} & 0 & \frac{E\,a\,h\left(\frac{q_1}{b}-\frac{q_3}{b}\right)^2}{12\,b\left(\nu^2-1\right)} & -\frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_1}{b}-\frac{q_3}{b}\right)}{12\left(\nu^2-1\right)} & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \frac{E\,a\,h\left(\frac{q_1}{b}-\frac{q_3}{b}\right)^2}{12\,b\left(\nu^2-1\right)} & 0 & -\frac{E\,a\,h\left(\frac{q_1}{b}-\frac{q_3}{b}\right)^2}{12\,b\left(\nu^2-1\right)} & \frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_1}{b}-\frac{q_3}{b}\right)}{12\left(\nu^2-1\right)} & 0 \\ -\frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_1}{b}-\frac{q_3}{b}\right)}{12\left(\nu^2-1\right)} & 0 & \frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_1}{b}-\frac{q_3}{b}\right)}{12\left(\nu^2-1\right)} & -\frac{E\,b\,h\left(\frac{q_4}{a}-\frac{q_6}{a}\right)^2}{6\,a\left(\nu^2-1\right)} & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_1}{b}-\frac{q_3}{b}\right)}{12\left(\nu^2-1\right)} & 0 & -\frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_1}{b}-\frac{q_3}{b}\right)}{12\left(\nu^2-1\right)} & \frac{E\,b\,h\left(\frac{q_4}{a}-\frac{q_6}{a}\right)^2}{6\,a\left(\nu^2-1\right)} & 0 \end{array}\right. \tag{4.115}$$

$$\left.\begin{array}{c} \frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_1}{b}-\frac{q_3}{b}\right)}{12\left(\nu^2-1\right)} \\ 0 \\ -\frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_1}{b}-\frac{q_3}{b}\right)}{12\left(\nu^2-1\right)} \\ \frac{E\,b\,h\left(\frac{q_4}{a}-\frac{q_6}{a}\right)^2}{6\,a\left(\nu^2-1\right)} \\ 0 \\ -\frac{E\,b\,h\left(\frac{q_4}{a}-\frac{q_6}{a}\right)^2}{6\,a\left(\nu^2-1\right)} \end{array}\right),$$

$$\boldsymbol{B}_K^N = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} & 0 & \frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} & 0 & -\frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} & 0 & 0 & 0 \end{pmatrix}, \tag{4.116}$$

$$\boldsymbol{C}_K^N = \begin{pmatrix} 0 & 0 & 0 & -\frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} & 0 & \frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} & 0 & -\frac{E\,h\,\nu\left(\frac{q_4}{a}-\frac{q_6}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{4.117}$$

$$
\boldsymbol{D}_K^N =
\left(\begin{array}{cccc}
-\frac{E\,a\,h\left(\frac{q_7}{b}-\frac{q_9}{b}\right)^2}{6\,b\left(\nu^2-1\right)} & 0 & \frac{E\,a\,h\left(\frac{q_7}{b}-\frac{q_9}{b}\right)^2}{6\,b\left(\nu^2-1\right)} & -\frac{E\,h\,\nu\left(\frac{q_{10}}{a}-\frac{q_{12}}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} \\
0 & 0 & 0 & 0 \\
\frac{E\,a\,h\left(\frac{q_7}{b}-\frac{q_9}{b}\right)^2}{6\,b\left(\nu^2-1\right)} & 0 & -\frac{E\,a\,h\left(\frac{q_7}{b}-\frac{q_9}{b}\right)^2}{6\,b\left(\nu^2-1\right)} & \frac{E\,h\,\nu\left(\frac{q_{10}}{a}-\frac{q_{12}}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} \\
-\frac{E\,h\,\nu\left(\frac{q_{10}}{a}-\frac{q_{12}}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} & 0 & \frac{E\,h\,\nu\left(\frac{q_{10}}{a}-\frac{q_{12}}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} & -\frac{E\,b\,h\left(\frac{q_{10}}{a}-\frac{q_{12}}{a}\right)^2}{12\,a\left(\nu^2-1\right)} \\
0 & 0 & 0 & 0 \\
\frac{E\,h\,\nu\left(\frac{q_{10}}{a}-\frac{q_{12}}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} & 0 & -\frac{E\,h\,\nu\left(\frac{q_{10}}{a}-\frac{q_{12}}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} & \frac{E\,b\,h\left(\frac{q_{10}}{a}-\frac{q_{12}}{a}\right)^2}{12\,a\left(\nu^2-1\right)}
\end{array}\right.
$$

$$
\left.\begin{array}{cc}
0 & \frac{E\,h\,\nu\left(\frac{q_{10}}{a}-\frac{q_{12}}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} \\
0 & 0 \\
0 & -\frac{E\,h\,\nu\left(\frac{q_{10}}{a}-\frac{q_{12}}{a}\right)\left(\frac{q_7}{b}-\frac{q_9}{b}\right)}{12\left(\nu^2-1\right)} \\
0 & \frac{E\,b\,h\left(\frac{q_{10}}{a}-\frac{q_{12}}{a}\right)^2}{12\,a\left(\nu^2-1\right)} \\
0 & 0 \\
0 & -\frac{E\,b\,h\left(\frac{q_{10}}{a}-\frac{q_{12}}{a}\right)^2}{12\,a\left(\nu^2-1\right)}
\end{array}\right) . \tag{4.118}
$$

4.2.3.3 The inertia matrix

Taking into account the general forms of the characteristic matrices (3.8), the inertia matrix can be presented in the following form

$$
\boldsymbol{M}_{ij} = \int_V \boldsymbol{C}_i^T \boldsymbol{\rho}\, \boldsymbol{C}_j \, \mathrm{d}V \,, \tag{4.119}
$$

where the matrix $\boldsymbol{C}$ describes the velocities in the finite element and is as follows

$$
\boldsymbol{C}_i^T = \begin{bmatrix} \frac{\partial N_i}{\partial t} & 0 \\ 0 & \frac{\partial N_i}{\partial t} \end{bmatrix} , \tag{4.120}
$$

$$
\boldsymbol{C}_j = \begin{bmatrix} \frac{\partial N_j}{\partial t} & 0 \\ 0 & \frac{\partial N_j}{\partial t} \end{bmatrix} . \tag{4.121}
$$

However, the mass matrix $\boldsymbol{\rho}$ is presented as follows

$$
\boldsymbol{\rho} = \begin{bmatrix} \rho & 0 \\ 0 & \rho \end{bmatrix} . \tag{4.122}
$$

Since the derivative of the shape function (4.64) with respect to time is constant, the integral (4.119) reduces to a matrix product

$$\boldsymbol{M}_{ij} = V_e \, \boldsymbol{C}_i^T \boldsymbol{\rho} \, \boldsymbol{C}_j \,, \tag{4.123}$$

where V_e is the space-time volume of the tetrahedron and we can calculate it using the determinant (4.91).

1st tetrahedral element Taking into account the shape function of the first tetrahedron (4.78), the matrix (4.121) takes the following form

$$\boldsymbol{C}_1 = \begin{pmatrix} -\frac{1}{h} & 0 & 0 & 0 & 0 & 0 & \frac{1}{h} & 0 \\ 0 & -\frac{1}{h} & 0 & 0 & 0 & 0 & 0 & \frac{1}{h} \end{pmatrix} . \tag{4.124}$$

The inertia matrix calculated by (4.123) based on (4.124) and (4.122) is as follows

$$\boldsymbol{M}_1 = V_e \, \boldsymbol{C}_1^T \boldsymbol{\rho} \, \boldsymbol{C}_1 = \begin{pmatrix} \frac{a\,b\,\rho}{6\,h} & 0 & 0 & 0 & 0 & 0 & -\frac{a\,b\,\rho}{6\,h} & 0 \\ 0 & \frac{a\,b\,\rho}{6\,h} & 0 & 0 & 0 & 0 & 0 & -\frac{a\,b\,\rho}{6\,h} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{a\,b\,\rho}{6\,h} & 0 & 0 & 0 & 0 & 0 & \frac{a\,b\,\rho}{6\,h} & 0 \\ 0 & -\frac{a\,b\,\rho}{6\,h} & 0 & 0 & 0 & 0 & 0 & \frac{a\,b\,\rho}{6\,h} \end{pmatrix} . \tag{4.125}$$

2nd tetrahedral element According to the shape function of the second tetrahedron (4.82), the matrix (4.121) can be written in the following form

$$\boldsymbol{C}_2 = \begin{pmatrix} -\frac{1}{h} & 0 & 0 & 0 & 0 & 0 & \frac{1}{h} & 0 \\ 0 & -\frac{1}{h} & 0 & 0 & 0 & 0 & 0 & \frac{1}{h} \end{pmatrix} . \tag{4.126}$$

Taking into account (4.126) and (4.122) the inertia matrix (4.123) of the second tetrahedron is given by

$$\boldsymbol{M}_2 = V_e \, \boldsymbol{C}_2^T \boldsymbol{\rho} \, \boldsymbol{C}_2 = V_e \begin{pmatrix} \frac{a\,b\,\rho}{6\,h} & 0 & 0 & 0 & 0 & 0 & -\frac{a\,b\,\rho}{6\,h} & 0 \\ 0 & \frac{a\,b\,\rho}{6\,h} & 0 & 0 & 0 & 0 & 0 & -\frac{a\,b\,\rho}{6\,h} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{a\,b\,\rho}{6\,h} & 0 & 0 & 0 & 0 & 0 & \frac{a\,b\,\rho}{6\,h} & 0 \\ 0 & -\frac{a\,b\,\rho}{6\,h} & 0 & 0 & 0 & 0 & 0 & \frac{a\,b\,\rho}{6\,h} \end{pmatrix} . \tag{4.127}$$

3rd tetrahedral element Based on the shape function (4.86), the matrix (4.121) takes the following form

$$\boldsymbol{C}_3 = \begin{pmatrix} -\frac{1}{h} & 0 & 0 & 0 & 0 & 0 & \frac{1}{h} & 0 \\ 0 & -\frac{1}{h} & 0 & 0 & 0 & 0 & 0 & \frac{1}{h} \end{pmatrix}, \tag{4.128}$$

where the inertia matrix calculated by (4.123) according to (4.128) and (4.122) is as follows

$$\boldsymbol{M}_3 = V_e \, \boldsymbol{C}_3^T \boldsymbol{\rho} \, \boldsymbol{C}_3 = V_e \begin{pmatrix} \frac{a\,b\,\rho}{6\,h} & 0 & 0 & 0 & 0 & 0 & -\frac{a\,b\,\rho}{6\,h} & 0 \\ 0 & \frac{a\,b\,\rho}{6\,h} & 0 & 0 & 0 & 0 & 0 & -\frac{a\,b\,\rho}{6\,h} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{a\,b\,\rho}{6\,h} & 0 & 0 & 0 & 0 & 0 & \frac{a\,b\,\rho}{6\,h} & 0 \\ 0 & -\frac{a\,b\,\rho}{6\,h} & 0 & 0 & 0 & 0 & 0 & \frac{a\,b\,\rho}{6\,h} \end{pmatrix}. \tag{4.129}$$

The local inertia matrix $\boldsymbol{M}$ is sum of (4.125), (4.127) and (4.129). The matrix $\boldsymbol{M}$ can also be represented as 4 sub-matrices $\boldsymbol{A}_m$, $\boldsymbol{B}_m$, $\boldsymbol{C}_m$ and $\boldsymbol{D}_m$ in the following way

$$\boldsymbol{M} = \begin{bmatrix} \boldsymbol{A}_m & \boldsymbol{B}_m \\ \boldsymbol{C}_m & \boldsymbol{D}_m \end{bmatrix}, \tag{4.130}$$

where

$$\boldsymbol{A}_m = \frac{ab\rho}{6h} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{4.131}$$

$$\boldsymbol{B}_m = -\frac{ab\rho}{6h} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{4.132}$$

$$\boldsymbol{C}_m = -\frac{ab\rho}{6h}\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{4.133}$$

$$\boldsymbol{D}_m = \frac{ab\rho}{6h}\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{4.134}$$

It is noteworthy that both B_k in Eq (4.99) and B_k^N in Eq (4.114) are triangular matrices. For the space-time finite element method, when calculating the deformation of the system, the final step involves solving the linear equation $\mathbf{F} = \mathbf{K}\mathbf{x}$, where $\mathbf{F}$ is a known vector, and $\mathbf{K}$ is the aforementioned triangular matrix. This characteristic is one of the reasons why the space-time finite element method is more efficient compared to traditional finite element methods. The triangular structure of $\mathbf{K}$ simplifies the solution process, reducing computational complexity and improving efficiency. For a detailed derivation and analysis, please refer to the work by [32].

4.2.4 Example

4.2.4.1 Description of the finite element model

The finite element model depicted in the Fig. 4.15 illustrates a triangle with vertices at coordinates $A(0.1, 0.3)$, $B(0.5, 0.2)$, and $C(0.6, 0.7)$. This configuration is designed for the application of the finite space-time finite element method.

- **Vertices and Coordinates**: The triangle's vertices are defined as follows:
  - Point A is located at $(0.1, 0.3)$
  - Point B is positioned at $(0.5, 0.2)$
  - Point C is positioned at $(0.6, 0.7)$
- **Boundary Conditions**:
  - Points A and B are fixed, meaning they are constrained from movement. This is represented by filled circles at these points, along with symbolic rectangles indicating fixed supports.

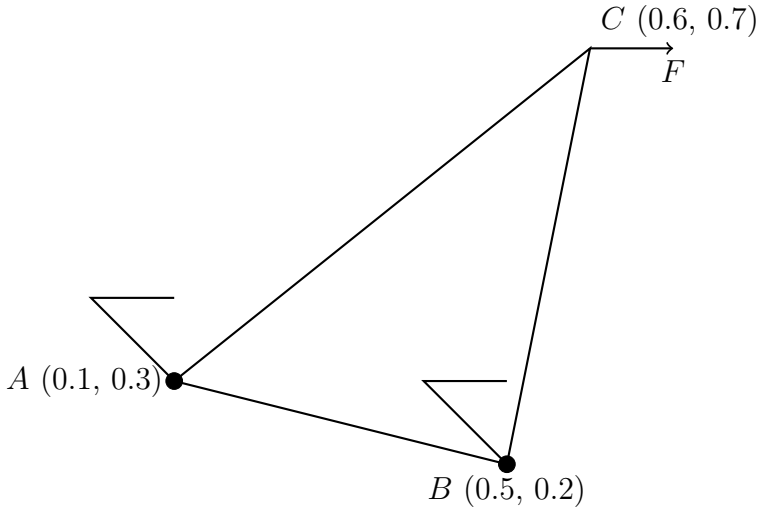


Figure 4.15: Finite element of the plane-stress problem. Points A and B are fixed, while a horizontal force F is applied at point C.

  - Point C is subjected to a horizontal force F. This force is illustrated by an arrow directed to the right, starting from point C.

- **Geometric Representation**:
  - The edges of the triangle are drawn using thick lines to clearly define the boundary of the element.
  - The nodes at each vertex are labeled with their respective coordinates for clarity and reference in the simulation.

- **Material Properties**: The material properties of the finite element are specified as follows:
  - Young's Modulus, $E = 0.01$ GPa
  - Density, $\rho = 1$ kg/m^3
  - Poisson's Ratio, $\nu = 0.2$

4.2.4.2 Results

In this section, the outcomes of the finite element simulations for different applied horizontal forces F are presented and analyzed. Three scenarios are considered, where the applied force F is set to 0.01N, 0.1N, and 1N, respectively. Each set represents a distinct loading condition that influences the deformation and stress distribution within the finite element model. The results are demonstrated in Fig. 4.16 to 4.18.

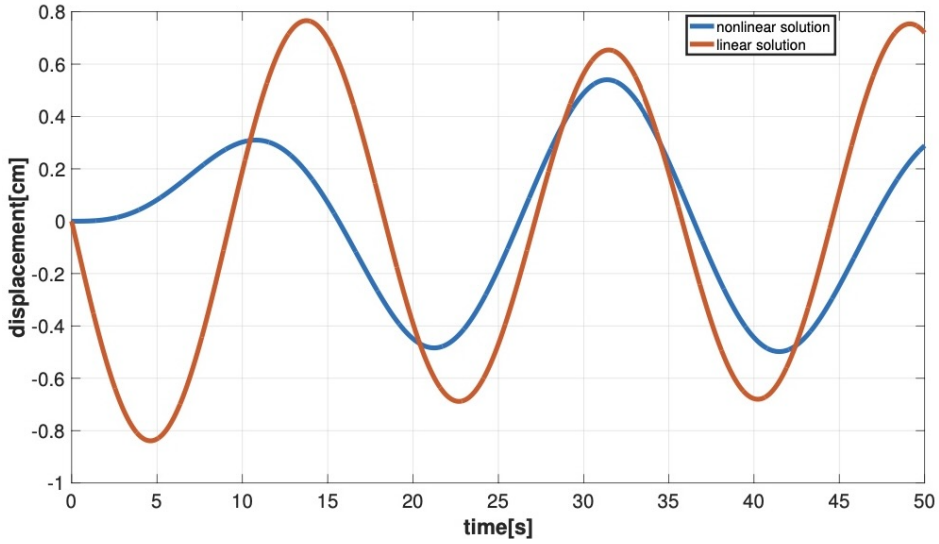


Figure 4.16: Results of finite element model for F=0.01 .

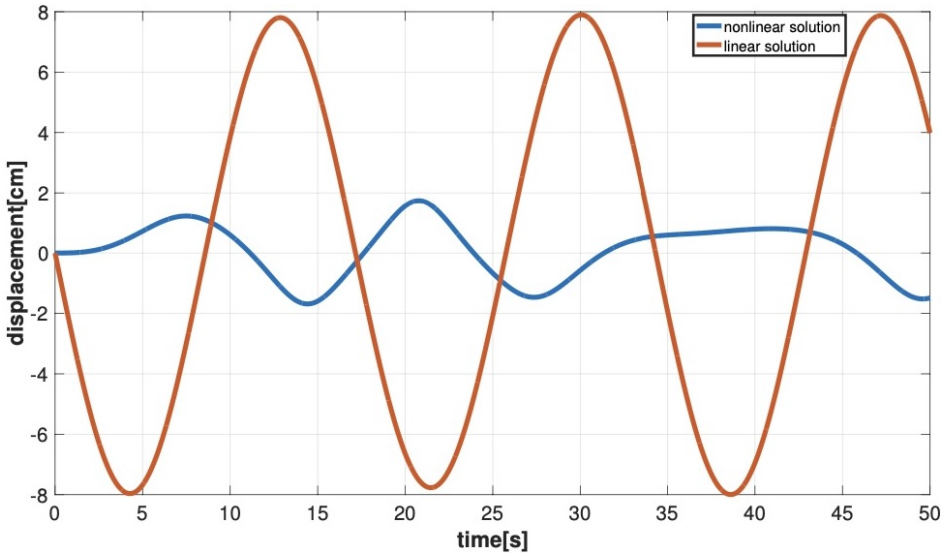


Figure 4.17: Results of finite element model for F=0.1 .

We investigate the influence of including the nonlinear term $\boldsymbol{K}^N$ in the finite element analysis by comparing the simulation results under different loading conditions.

Linear analysis (neglecting nonlinear term $\boldsymbol{K}^N$): When neglecting the nonlinear term $\boldsymbol{K}^N$, the simulation results closely align with traditional linear mechanical principles. Under small deformation conditions, such as F=0.01, the structural response of the triangular element exhibits linear behavior, and the differences between the linear and nonlinear analyses are minimal. In these cases, the linear analysis provides an effi-

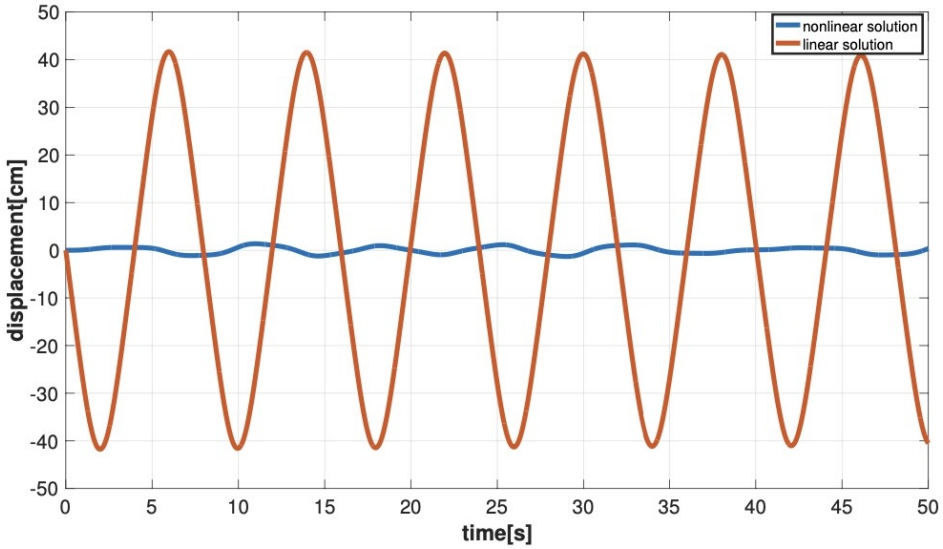


Figure 4.18: Results of finite element model for F=1 .

cient and accurate representation of the structural behavior without the computational overhead of solving nonlinear equations.

Nonlinear analysis (including nonlinear term $\boldsymbol{K}^N$): In contrast, under larger loading conditions (e.g., F=0.1 or F=1), the influence of geometric nonlinearities becomes more pronounced. Including the nonlinear term $\boldsymbol{K}^N$ in the analysis accounts for large deformations, leading to significant differences in the structural response compared to the linear analysis. In these scenarios, the nonlinear analysis provides a more accurate representation of the structural behavior, capturing the effects of nonlinear material and geometric properties.

The comparison between the linear and nonlinear analyses highlights the critical role of considering geometric nonlinearities in finite element simulations, particularly under significant loading conditions. While the linear analysis may suffice for cases with small deformations, larger deformations necessitate the inclusion of the nonlinear term $\boldsymbol{K}^N$ to capture the structural response accurately. However, it's essential to consider the computational costs associated with solving nonlinear equations, particularly for large-scale simulations.

In conclusion, the decision to include the nonlinear term $\boldsymbol{K}^N$ in finite element simulations depends on the magnitude of deformation imposed by the applied loading conditions. When deformations are small, the analysis provides an efficient and accurate solution. However, for scenarios where deformations are significant, such as those resulting from larger forces, considering the nonlinear term becomes essential to ensure the accuracy of the simulation results, albeit at the cost of increased computational complexity.

Chapter 5

Parallel simplex-shaped space-time finite element method in dynamic analysis of structures

5.1 Parallel computing algorithm

Effective design of an algorithm that optimally uses the operating memory and processor working time requires the preparation of several auxiliary tables containing various types of numbering useful for dividing the system into subsystems, depending on the number of processors at the disposal. Of course, we are interested in separating the task into at least 16 computational threads. The development of computer architecture will easily increase this number. The condition for the effectiveness of the algorithm is fast data transmission between the processor cores, cache and main memory. We are not able to assess the impact of the processor architecture on computational times, but we will show how the computational time is reduced in various multi-threaded test tasks. On this occasion, it should be noted that we do not take into account the launch of automatic threading processes carried out automatically by compilers, nor the use of pipelining or iterative procedures for solving systems of equations. We present a particular method of parallelizing construction dynamics problems, or more generally, initial-boundary problems. It is also not the purpose of the work to compare the presented method with all known or only used algorithms, because then the scope of the work would go far beyond the intended, commonly accepted one. Nor would such a comparison be credible without a thorough study of every known method. such a comparison may be made in the future.

5.2 Algorithm of the solution

The parallelization algorithm will be explained on the example of a two-dimensional problem depicted in Fig. 5.1. This will facilitate the reader to understand the next steps without losing the generality of the presentation. The element-by-element as-

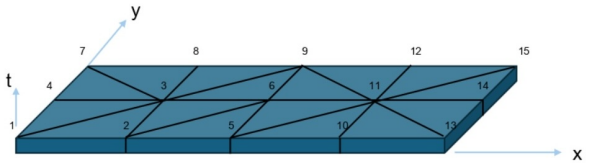


Figure 5.1: A 2-dimensional problem as an example discussed in this article.

sembly results in the coefficient matrices with the triangular matrix **B** as in Fig. 5.2. Auxiliary tables should be created that will facilitate and simplify the subsequent stages

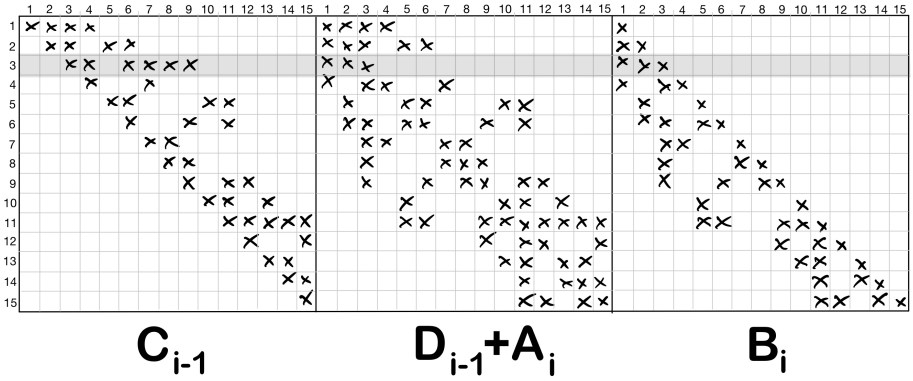


Figure 5.2: Exemplary matrices **C**, **D**+**A**, and **B** for the 2D plate structure.

of algorithmization. The first of these auxiliary tables is the IM table describing the arrangement of non-zero frames of **A**, **B**, **C**, and **D** in memory. At this point, we note that only the **B** and **D** matrices require memory reservations, and only for non-zero coefficient frames. This sequence of numbers, resulting from the topology of the spatial mesh, allows for efficient programming of operations assigned to the rows of the global matrices **A**, **B**, **C**, **D**. This sequence contains the numbers of nodes neighboring in the spatial mesh with the next node 1, 2, ..., NP, including themselves. The numbers of consecutive nodes that start the subsequences of neighboring nodes are placed in the first place (Fig. 5.3).

The second auxiliary vector is IW, which slices the IM string into parts assigned to subsequent nodes. In practice, it contains the position of the node numbers 1, 2, ...,

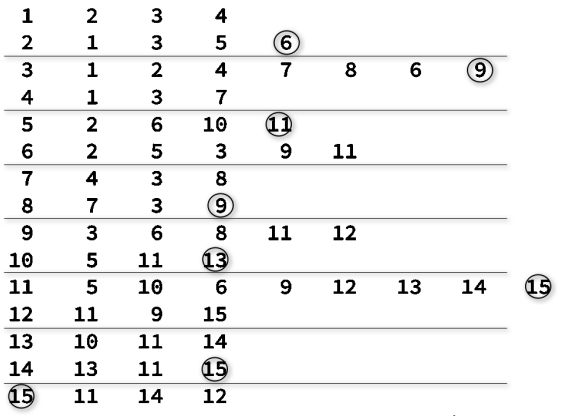


Figure 5.3: A sequence of nodes adjacent to subsequent nodes IM.

located in the first places of the strings IM:

$$\mathsf{IW} = [1, 5, 10, 18, 22, 27, ..., 76] .$$

This string ends with an element at position NP+1, containing the number of the string IM increased by 1 to allow simple counting of last row elements in an IM string.

Now one needs to separate groups of nodes that will be solved within one package on a single processor core. The number of packages should correspond to the number of available cores. With a smaller number of separated packages, the calculation performance will be lower, as there will be idle periods of unused cores. On the other hand, a larger number of packages will force unnecessary portioning and solution several packages one after another.

If we solve the sample task on a maximum of eight cores, then we can separate eight groups of nodes, in which there will be two nodes and in the last one one:

```
group 1: nodes 1, 2          group 5: nodes  9, 10
group 2: nodes 3, 4          group 5: nodes 11, 12
group 3: nodes 5, 6          group 5: nodes 13, 14
group 4: nodes 7, 8          group 5: node  15
```

Due to the limitations resulting from the mesh topology, when solving a specific node must be preceded by completing the calculations of the corresponding nodes adjacent to it, the number of node groups, packages, should be much greater than the number of cores.

According to the established scheme in the Fig. 5.4, calculations of packages 1, 4 and 8 can be carried out independently, with the exception that package 1 will have fixed unknowns in time t_{i+2}, group 4 in time t_{i+1}, and group of 8 in time t_i. The calculations can be carried out on three cores. More cores will not be physically used. Slightly

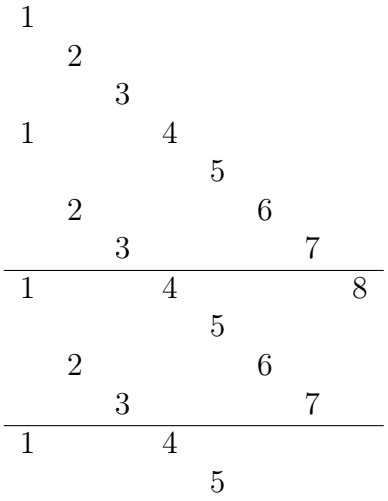


Figure 5.4: Initial stage and repeatable cycles of parallel solution.

different numbering of nodes and their neighbors would allow more even distribution of the load in steps 1 and 2. Optimization of the number of groups of nodes, their size and diversity of this number is a separate mathematical problem.

One can choose between two calculation modes. The first one consists in determining the solutions of successive groups in oblique layers, as in Fig. 3.9. The second mode, more natural, is to carry out the start-up phase of the procedure. Therefore, in the initial phase, batches 1, 2, 3 should be calculated in the first start-up step at $t = 0$, in the second step batches 1 and 2 at $t = \Delta t$, and in step 3, batch 1 at time $t = 2\Delta t$. Thanks to this, further steps can be carried out in horizontal layers as shown in the Fig. 3.9, but each package will represent a different moment.

5.2.1 Algorithm for partition of the solution processes

- Split the list of nodes coded in table IW into packages. Below the partition into 8 packages is applied:

  IMPROC = [1, 2 | 3, 4 | 5, 6 | 7, 8 | 9, 10 | 11, 12 | 13, 14 | 15] .
  Prepare the indexing table
  IWPROC = [1, 3, 5, 7, 9, 13, 15, 16] .

- A temporary table TMP will make programming easier

  1: $l \leftarrow 0$
  2: **for** $i \leftarrow$ 1, n_proc **do**
  3: **for** $j \leftarrow$ IWPROC(i), IWPROC(i+1)-1 **do**

```
4:          l ← l + 1
5:          TMP(l)=i
6:      end for
7:  end for
```

Here TMP=[1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8]

- In every package find the maximum node number (table MX_PACK indicated with circles if Fig. 5.3)
  MX_PACK=[6, 9, 11, 9, 13, 15, 15, 15] .

  Now establish what package proceeds a given package

```
1: for i ←1, n_proc do
2:     IPROCEED(i) ← TMP(MX_PACK(i))
3: end for
```

Finally IPROCEED = [3, 5, 6, 5, 7, 8, 8, 8]

- .....COUNTERS.....

```
do i=1,npack
   counters(1,i)=i
enddo

do ii=2,npack
   counters(ii,1)=counters(ii-1,iprev(1))+1
   do i=2,npack
      counters(ii,i)=max(counters(ii-1,iprev(i)),counters(ii,i-1))+1
   enddo
enddo
```

- Generate a table of processes

```
1: l ← 0
2: for i ←COUNTERS(npack,1),COUNTERS(npack-1,1)+1,-1 do
3:     l ← l + 1
4:     for j ←npack,1,-1 do
5:         for k ←1,npack do
6:             if counters(j,k) = i then
7:                 ITABPROC(l,COUNTPERC(l))=k
8:             end if
```

```
 9:         end for
10:      end for
11: end for
```

The resulting table ITABPROC now has the following form

1	4	8
3	7	
2	6	
5		

In the table below each line shows the calculations of individual packages in the same time, i.e. in parallel. After the initial 7-step stage, a repetitive sequence of four computational cycles is obtained. In this simple example, 4 computation cycles using 3 cores replace 15 cycles of a single-core processor.

One can see that they must be performed in a specific order, after the calculations of preceding packages are completed. In the case of packages 1, 4, and 8, they can be solved in parallel, but each must receive the results of packs from the previous moments, package 1 containing nodes 1 and 2 after the end of package 3 with nodes 5 and 6, package 4 containing nodes 7 and 8 after the end of package 5 with nodes 9 and 10, and package 8 with a single node 15 after the end of package 8 from a previous time step, respectively (see Fig. 5.3). Unfortunately, with the assumed topology, the calculations of package 6, 7, and 8 may only be started after the calculations of batch 8 have been completed for the previous time step. Fig. 5.5 shows the repeatable sequence of steps distributed over processor's core. The Fig. 5.5 corresponds with the part of the scheme in Fig. 5.4.

	core 1	core 2	core 3
step 1:	1	4	8
step 2:	idle	5	idle
step 3:	2	6	idle
step 4:	3	7	idle

Figure 5.5: Separating packages for calculations in the successive steps.

Characteristic matrices of the problem are collected in the form of a ribbon of square matrices with dimensions equal to the number of degrees of freedom at the node. Therefore, they often have dimensions of 2x2 or 3x3. Pairs of matrices located

in matrices **B** and **D** are collected in the order corresponding to the sequence IM. The numbers in circles correspond to the first numbers in the rows in Fig. 5.3. The subsequent numbers correspond to the consecutive numbers in those sequences. In the case where the number of the given node in the row IM, as shown in the Fig. 5.3, is greater than the first number in the row, the matrix **B** corresponding to the frame does not exist due to the triangular form of the matrix. In this case, there is no need to reserve space for these coefficients (Fig. 5.6). The location and fast access to the

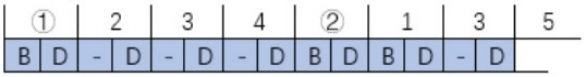


Figure 5.6: Global matrix coefficients stored as a ribbon.

appropriate coefficients can be implemented in many ways using a built-in sequence, either through an additional position array or by utilizing features and capabilities of the programming language used.

In this simple coding depicted in Fig. 5.6 only about 150-160% of the non-zero coefficients contained in the classical stiffness matrix of the finite element method are collected, without the need to reserve any memory for operations related to solving the problem. This may raise the question of what the 50-60% of memory is used for and what benefits this slightly increased demand brings.

The programming-correct approach is to split the band into two separate bands, maintaining the elements of matrix **B** and **D** separately. It is necessary to create appropriate auxiliary indexing of matrix frames in both bands. It is important to note that only the starting positions of the sequences in the frames, as shown in Fig. 5.6 and marked with numbers in circles, need to be indexed. The remaining positions of specific matrix frames are automatically reproduced by looping within the subarrays of the IM array from Fig. 5.3, taking into account whether the starting number of a given row is lower or higher than the next number in the row from Fig. 5.3.

It should be emphasized here that in the above method of writing the coefficient matrix, no operational memory is required as a workspace. Furthermore, very large tasks performed on computers with relatively small memory can be solved almost at no cost by storing the matrices in external memory and bringing them in sequences piece by piece to operational memory. This is due to the fact that the calculations are carried out node by node and the matrices are also built and collected node by node. This feature also allows to distribution of parts of very large tasks among separate computers and exchanges only solution vectors between them.

The second remark is as follows. Collecting the contents of matrices, whether directly in the form of matrix elements **B** and **D** or generated and collected as the vector product of matrices **A** and **C** by state vectors, allows for an accurate description of the model's evolution under the influence of variable coefficients or non-linear factors. This

topic is described, for example, in [13]. It should be noted here that the physical task describes a time continuous method, especially at the interface of time layers, taking into account the equilibrium of observed parameters, unlike classical approaches, which only consider equilibrium at selected time intervals.

5.3 Efficiency assessment

A simple test of parallel computation with access to common matrices shows the expected feature that properly scheduled threads do not increase parallel computation time beyond physical necessity. With 16 available cores, the computation time of N threads with similar workloads corresponds to int(N/16+0.5) the execution time of a single thread (Fig. 5.7). a similar relationship is to be expected in the final real-scale tests. The above example used for algorithm description exhibits the following execu-

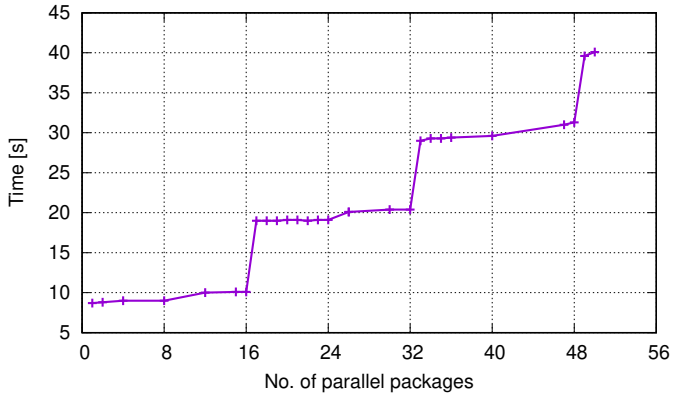


Figure 5.7: Computational time of various numbers of task parts solved in parallel.

tion time measured with 'time' command:

No. of cores	real time	user time
1 core	0m1.569s	0m1.564s
16 cores	0m0.140s	0m0.232s

5.4 Conclusions

The main conclusions from the proposed method of discretization of the differential equation of motion are as follows:

- Only non-zero characteristic matrix coefficients are stored.

- Semi-separation of the system of equations thanks to symplectic elements and the created triangular global matrix of coefficients significantly reduces the computational cost.
- The method can be placed between implicit methods with full/band matrices and explicit methods with diagonal matrices.
- Symplectic elements allow you to divide a spatial area into sub-areas and solve them in parallel.
- Separating a sub-area sensitive to non-linear influences, requiring iterative solving of a single-layer equilibrium equation, by surrounding it with simplectic elements, makes it possible to limit iterations to this area.
- Obtaining a limited speed of information propagation between grid nodes in subsequent computational steps introduces a hyperbolic character of the numerical model of wave tasks, instead of the traditionally parabolized model, in which we observe a non-physical infinite speed of information propagation. In many dynamics tasks this is of lesser importance, although it is important to be aware of it.
- The functionality of subdomain decoupling allows for distributing parts of very large tasks among separate computers and exchanging only solution vectors between them.
- The computational performance over time is linearly dependent on the number of processor cores.

Chapter 6

Examples of applications

The issue of shock impact is a significant concern as it directly endangers the health and safety of individuals involved in sports, high-risk occupations, and other hazardous activities. Athletes in sports such as football, rugby, and cycling frequently encounter collisions, falls, and other accidents, which can result in severe injuries to the head and body, or even be life-threatening. Similarly, workers in fields such as construction, machinery operation, and the military often face similar dangers. These workers are at risk of accidents, such as falling heavy objects, collisions, and explosions, especially when working at heights or operating machinery.

The development of impact protection materials is a critical and practical issue. Sudden impacts can cause significant damage to both lives and equipment. Therefore, the demand for safeguarding structures against severe impacts is steadily increasing, as the safety of structures is largely dependent on the materials used in their construction. It is highly desirable to enhance the properties of materials designed to protect the human head, body, and various engineering structures. There is a constant need for lighter, high-performance, and cost-effective materials in practical applications. Moreover, there is an increasing demand for improved models and calculation methods to accurately address practical issues, including deformation and stress analysis.

Therefore, it is essential to enhance existing protective equipment and boost its effectiveness through innovative research. Introducing new materials, developing more accurate mathematical and physical models, and creating more efficient numerical calculation methods and manufacturing processes could provide new ideas and solutions to these challenges. Continuous improvement and innovation aim to provide lighter, more comfortable, and more effective protective equipment to better safeguard the health and safety of athletes, workers, and other high-risk individuals.

Throughout history, traditional materials commonly used for impact reduction have included iron (steel) [77] and materials reinforced with natural fiber particles such as sisal and jute [90, 89]. These materials have been chosen for their specific properties such as impact strength and compression strength. Additionally, hybrid materials like

aramid fibers and kenaf fibers have been explored as alternatives to reduce costs and environmental impact while maintaining required protection levels [78]. Due to their low cost, synthetic materials such as expanded polystyrene (EPS), expanded polypropylene (EPP) and expanded polyurethane (EPU) are also popular. EPS is frequently used due to its exceptional energy absorption properties and lightweight nature [71]. However, existing materials have significant limitations, which is why structures based on intelligent materials are being intensively developed. Smart materials are advanced materials that can respond to environmental changes. They can alter one or more properties under controlled conditions by external stimuli such as stress, moisture, electric fields, magnetic fields, light, and temperature [56]. There are many types of smart materials such as piezoelectric, granular materials, shape memory alloys, electrorheological and magnetorheological fluid [8]. However, Smart materials in mechanics face limitations, excessive mass and the necessity to use additional devices to enable modification of parameters are two drawbacks of smart materials. These issues can lead to increased costs, complexity, and potential reliability concerns [34].

Mechanical metamaterials may be a potential solution to the above-mentioned problems. Metamaterials are man-made structures with counterintuitive mechanical properties that originate in the geometry of their unit cell instead of the properties of each component [105]. Engineered materials that gain their properties from structure rather than composition, have led to the study of a myriad of properties not exhibited by most (and in some cases all) natural materials (Fig.6.1). Such as pentamode materials, auxetic metamaterials, negative stiffness, negative compressibility, light fabric metamaterials, and origami material [19, 22, 80].

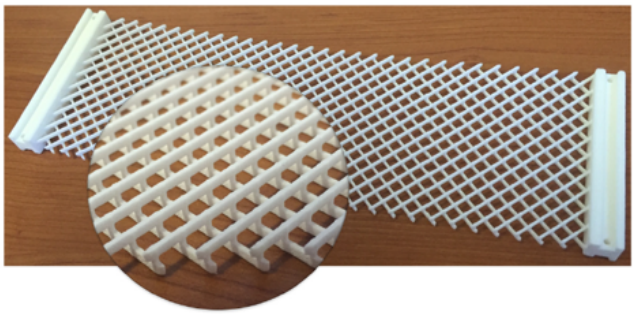


Figure 6.1: Mechanical metamaterials gain their properties from structure rather than composition [19].

The first part of the chapter presents the concept of a one-dimensional linear elastic structure in which the stiffness changes with the moving mechanical wave caused by an impact. The simplicity of the case enables a semi-analytical solution to the problem, which allows for the verification of solutions obtained numerically using the space-time

element method. The presented research was published in [107]. Next, an example of a visco-plastic two-dimensional structure is presented, the appropriately selected geometry of which allows shaping the response of the structure subjected to impact loading.

6.1 Smart elastic material with a moving local stiffness zone

The primary objective of this research endeavor is to elucidate the concept of an intelligent material that is adept at mitigating the adverse effects stemming from impact loads. It is paramount to underscore that the material under consideration exhibits the versatility of being either rigid or flexible, thereby showcasing its efficacy in attenuating the repercussions of impacts. The enhancement in operational efficiency can be attributed to the localized and transient alteration in stiffness prompted by the propagation of mechanical waves within the material. It is imperative to highlight that the envisaged material must possess a notably high usable stiffness, only deviating from this characteristic momentarily and locally during critical junctures to modify its mechanical properties. Through this approach, the functionality of an element or structure integrated with intelligent material remains consistent and unaltered across its entire operational spectrum, encompassing instances of impact occurrence.

The study posits a streamlined analysis concerning the behavioral dynamics of a theoretical metamaterial. This material exhibits elastic properties, with the passage of mechanical waves instigating localized adjustments in its stiffness profile. The ramifications of localized increments or decrements in Young's modulus on the structural response were meticulously scrutinized. The insights garnered from these investigations pave the way for identifying the optimal configuration of a metamaterial that aligns with the stipulated operational strategy. Leveraging a simplistic 1D model facilitates an in-depth exploration of the intricacies of the issue owing to its inherent simplicity and the expedited computational processing it affords.

In subsection 6.1.1, we elaborate on the mathematical model, where Heaviside functions are utilized to effectively represent the traveling stiffness zone within the system. This methodology leads to the emergence of a non-constant longitudinal force exerted on the rod. Following this, we clarify the problem definition and the equation of motion to enhance the understanding of the model. The next part, subsection 6.1.2, focuses on providing a semi-analytical solution to the model, achieved by applying the separation of variables method alongside integral transformation techniques. In subsection 6.1.3, we meticulously develop a detailed numerical finite element method framework to tackle the complexities involved in the problem. The results from the numerical simulations performed are instrumental in validating the accuracy and effectiveness of

the constructed numerical model. Subsection 6.1.4 explores the impact of varying the number of local material changes on the system's overall response. Finally, subsection 6.1.5 presents a comprehensive analysis and discussion, offering significant conclusions and implications drawn from the study.

6.1.1 Mathematical model

Let us consider a homogeneous straight clamped-free rod of the length L and a constant cross-section A clamped on the left side (Fig. 6.2). Assuming that the material constituting the rod exhibits linear elasticity, it can be inferred that the behavior of the material adheres to the fundamental principles delineated by Hooke's law. The function denoted as $w(x,t)$ serves to elucidate the longitudinal displacements within the rod under consideration. Consequently, the articulation of the boundary conditions can be succinctly expressed as follows

$$\begin{cases} w(0,t) = 0\,, \\ \left.\frac{\partial w(x,t)}{\partial x}\right|_{x=L} = 0\,. \end{cases} \tag{6.1}$$

Moreover, the bar has a density ρ and a global stiffness E. The following zero initial

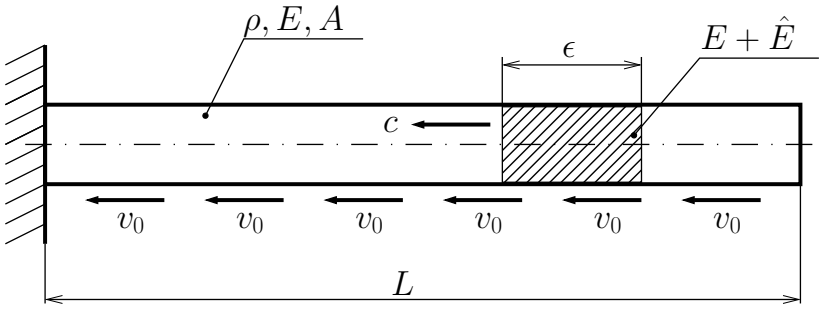


Figure 6.2: A rod with a moving stiffness zone.

displacement and initial velocity v_0

$$\begin{cases} w(x,0) = 0\,, \\ \dot{w}(x,0) = v_0\,. \end{cases} \tag{6.2}$$

Additionally, we assume that along with the wave moving in the rod, its stiffness changes locally

$$E(x) = E + \left[H(x-(L-\epsilon-ct)) - H(x-(L-ct))\right]\hat{E}\,, \tag{6.3}$$

where

$$H(x) = \begin{cases} 0, & \text{if } x < 0\,, \\ 1, & \text{if } x \geq 0\,. \end{cases} \tag{6.4}$$

The Heaviside step function, $H(x)$, plays a crucial role in the stiffness equation by modelling the localized change in stiffness as the wave propagates. The Heaviside function is defined as 0 for $x < 0$ and 1 for $x \geq 0$, and in this context, it creates a "window" function through the difference $H(x-(L-\epsilon-ct))-H(x-(L-ct))$. This difference results in a value of 1 within the interval $(L-ct) \leq x < (L-\epsilon-ct)$ and 0 elsewhere, effectively defining a region of length ϵ where the stiffness is modified by $\hat{E}$. As both $L-ct$ and $L-\epsilon-ct$ are functions of time t, this interval moves with the wave at speed c, thus ensuring that the localized change in stiffness travels along the rod as the wave propagates. Consequently, the Heaviside function allows the equation to dynamically capture the moving region of increased stiffness, reflecting the localized nature of the stiffness modification accompanying the wave. We will consider two cases: local strengthening ($\hat{E} > 0$ and local weakening ($\hat{E} < 0$ of stiffness. The difference of the Heaviside functions, $H(x-(L-\epsilon-ct))-H(x-(L-ct))$, defines the traveling stiffness zone of width ϵ. This means that within the interval $(L-ct) \leq x < (L-\epsilon-ct)$, the stiffness changes by $\hat{E}$, creating a region that either strengthens or weakens the material locally. As time progresses, this region moves along the rod with the wave at speed c, where $c = \sqrt{E/\rho}$. This speed is derived from the relationship between the stiffness E and the density ρ of the material. For the purpose of our analysis, we focus only on the stage where the wave reaches the support at $t_{\max} = L/c$, ensuring we understand the effects of this travelling stiffness zone up to the point where the wave has fully propagated through the length of the rod.

The longitudinal force N in the rod is given by

$$N = E(x)A\frac{\partial w}{\partial x}\,, \tag{6.5}$$

where $E(x)$ is the stiffness, A is the cross-sectional area, and $\frac{\partial w}{\partial x}$ is the strain in the rod. According to Newton's second law, the change in momentum of a segment of the rod is equal to the net force acting on it. For a differential segment $\mathrm{d}x$ of the rod, the equation of motion can be expressed as

$$\rho A \mathrm{d}x\,\frac{\partial^2 w(x,t)}{\partial t^2} = -N(x,t) + N(x,t) + \mathrm{d}N(x,t)\,, \tag{6.6}$$

where ρ is the density of the rod material, and $\frac{\partial^2 w(x,t)}{\partial t^2}$ represents the acceleration of the segment. This equation links the dynamic behaviour of the rod (through acceleration) to the spatial variation in the longitudinal force N, highlighting the relationship between the material properties, the cross-sectional area, and the resulting longitudinal displacements due to external forces. Thus, by combining the constitutive relation for

the longitudinal force with Newton's law, we derive the fundamental equation governing the wave propagation in the rod.

According to (6.3) the force N is not constant, so the increase in longitudinal force $\mathrm{d}N$ can be written in the following form

$$\begin{aligned}\mathrm{d}N = \frac{\partial N}{\partial x}\mathrm{d}x = \Bigg\{ EA\frac{\partial^2 w}{\partial x^2} + \big[\delta(x-(L-\epsilon-ct)) - \delta(x-(L-ct))\big]\,\hat{E}A\frac{\partial w}{\partial x} \\ + \big[H(x-(L-\epsilon-ct)) - H(x-(L-ct))\big]\,\hat{E}A\frac{\partial^2 w}{\partial x^2}\Bigg\}\,\mathrm{d}x\,.\end{aligned} \tag{6.7}$$

Finally, after rearrangements the Eqn (6.6), after taking into account (6.7) and dividing by $A\mathrm{d}x$, takes the form of the equation of motion

$$\begin{aligned}-E\frac{\partial^2 w(x,t)}{\partial x^2} - \big[\delta(x-(L-\epsilon-ct)) - \delta(x-(L-ct))\big]\,\hat{E}\frac{\partial w}{\partial x} \\ - \big[H(x-(L-\epsilon-ct)) - H(x-(L-ct))\big]\,\hat{E}\frac{\partial^2 w}{\partial x^2} + \rho\frac{\partial^2 w(x,t)}{\partial t^2} = 0\,.\end{aligned} \tag{6.8}$$

The Dirac delta function, denoted as $\delta(x)$, can be defined as follows

$$\delta(x) = \begin{cases}\infty, & \text{if } x = 0\,,\\ 0, & \text{otherwise}\,.\end{cases} \tag{6.9}$$

The relationship of these functions is

$$\delta(x) = \frac{d}{dx}H(x)\,. \tag{6.10}$$

The Eq (6.8) described is a partial differential equation with variable coefficients. It involves Dirac deltas and Heaviside step functions as variables, indicating localized changes and discontinuities in the system. Consequently, the potential solution is sought in the distributional sense, as traditional solutions may not adequately handle such singularities.

A limitation of the mathematical model is applicable only during the initial phase of the process, specifically up to the point when the wave reaches the support. This restriction stems from the nature of the function representing the moving stiffness as defined in equation (6.8). The function does not account for wave reflections from the fixed end of the rod, making any further continuation of the solution beyond this phase physically inaccurate. Incorporating wave reflections would introduce significant mathematical complexity, making the solution process far more challenging. This approach allows for a broader understanding of solutions, accommodating the irregularities presented by the Dirac deltas and Heaviside steps within the framework of distribution theory.

6.1.2 Semi-analytical solution

Semi-analytical analysis refers to a computational approach that combines analytical methods with numerical techniques to solve complex engineering problems efficiently. Various research papers highlight the utility and effectiveness of semi-analytical models in different fields. For instance, Guo et al. developed a semi-analytical model for predicting the vibration characteristics of flexible varying section disk-blades systems, showcasing high precision and computational efficiency [45]. Additionally, Qian et al. introduced a semi-analytical model for stability analysis of unsaturated slopes, incorporating spatially variable parameters and hydraulic considerations, demonstrating its potential as a benchmark for geotechnical design[74]. Furthermore, Zhou et al. utilized a semi-analytical model to analyze the dynamic characteristics of missiles during vertical cold launches, proving its effectiveness in studying the impact of wind load and other factors on missile behavior [84]. These studies collectively emphasize the significance of semi-analytical methods in enhancing computational efficiency and accuracy across diverse engineering applications. In this study, in order to solve the Eq (6.8) along with the boundary conditions (6.1) and the initial conditions (6.2) we will use the separation of variables method

$$w(x,t) = V(t)U(x)\,. \tag{6.11}$$

Substitution (6.1) into the formula (6.11) leads to boundary conditions for the function $U(x)$. To solve the boundary problem, we define eigen functions and eigenvalues. The assumed boundary conditions are satisfied by the following series of sines

$$w(x,t) = \frac{2}{L}\sum_{j=1}^{\infty} V_j(t)\sin\frac{(2j-1)\pi x}{2L}\,. \tag{6.12}$$

So we get an infinite sequence of eigenvalues. Each eigenvalue corresponds to an eigenfunction. The $V(t)$ function is determined from the following formula

$$V_j(t) = \int_0^L w(x,t)\sin\frac{(2j-1)\pi x}{2L}\mathrm{d}x\,. \tag{6.13}$$

The integral (6.13) is called the Fourier transformation. Taking into account (6.13) initial conditions (6.2) take the following form

$$V_j(0) = 0\,, \qquad \dot{V}_j(0) = \frac{2v_0 L}{(2j-1)\pi}\,. \tag{6.14}$$

According to (6.12) and (6.13) the equation of motion (6.8) after rearrangements can be written in the following form

$$\begin{aligned}
&\ddot{V}_j(t) + \frac{E}{\rho}\left(\frac{(2j-1)\pi}{2L}\right)^2 V_j(t) \\
&\quad - \frac{\hat{E}}{\rho}\frac{\pi}{L^2}\sum_{k=1}^{\infty}(2k-1)V_k(t)\cos\frac{(2k-1)\pi(L-\epsilon-ct)}{2L}\sin\frac{(2j-1)\pi(L-\epsilon-ct)}{2L} \\
&\quad + \frac{\hat{E}}{\rho}\frac{\pi}{L^2}\sum_{k=1}^{\infty}(2k-1)V_k(t)\cos\frac{(2k-1)\pi(L-ct)}{2L}\sin\frac{(2j-1)\pi(L-ct)}{2L} \\
&\quad + \frac{\hat{E}}{\rho}\frac{2}{L}\left(\frac{\pi}{2L}\right)^2\sum_{k=1}^{\infty}(2k-1)^2V_k(t)\int_{L-\epsilon-ct}^{L}\sin\frac{(2k-1)\pi x}{2L}\sin\frac{(2j-1)\pi x}{2L}\mathrm{d}x \\
&\quad - \frac{\hat{E}}{\rho}\frac{2}{L}\left(\frac{\pi}{2L}\right)^2\sum_{k=1}^{\infty}(2k-1)^2V_k(t)\int_{L-ct}^{L}\sin\frac{(2k-1)\pi x}{2L}\sin\frac{(2j-1)\pi x}{2L}\mathrm{d}x = 0\,,
\end{aligned} \tag{6.15}$$

where the integral is presented for the interval $< L-\epsilon-ct, L >$ in the following form

$$\begin{aligned}
&\int_{L-\epsilon-ct}^{L}\sin\frac{(2k-1)\pi x}{2L}\sin\frac{(2j-1)\pi x}{2L}\mathrm{d}x \\
&= \frac{2L}{\pi}\frac{1}{(2j-1)^2-(2k-1)^2}\left[(2j-1)\cos\frac{(2j-1)\pi(L-\epsilon-ct)}{2L}\sin\frac{(2k-1)\pi(L-\epsilon-ct)}{2L}\right. \\
&\left.-(2k-1)\sin\frac{(2j-1)\pi(L-\epsilon-ct)}{2L}\cos\frac{(2k-1)\pi(L-\epsilon-ct)}{2L}\right].
\end{aligned} \tag{6.16}$$

The second integral differs by the integration interval when $\epsilon = 0$. We can see that the solution (6.16) is only possible for $j \neq k$. In the case of $j = k$ we have to use l'Hôpital's rule. The formula is given in the following form

$$\int_{L-\epsilon-ct}^{L}\left(\sin\frac{(2j-1)\pi x}{2L}\right)^2\mathrm{d}x = \frac{L}{2(2j-1)\pi}\sin\frac{(2j-1)\pi(L-\epsilon-ct)}{L} + \frac{1}{2}(ct+\epsilon)\,.$$

We limit the series to a finite number of n terms. Finally, the system of equations (6.15) can be written in matrix form

$$\mathbb{M}\ddot{\mathbf{V}} + \mathbb{K}\mathbf{V} = \mathbf{0}\,. \tag{6.17}$$

The final solution to the problem is obtained by substituting the components of the vector V into the series expansion presented in equation (6.12). To solve the system of differential equations described in equation (6.17), the numerical Newmark integration

method is employed. This method, widely used in structural dynamics, allows for time-domain integration with the initial conditions provided in equation (6.14). The entire computational process has been implemented in a custom-developed Fortran program, which was specifically written for the task at hand.

One of the key advantages of this semi-analytical solution is its utility in verifying the accuracy of results obtained through the finite element method. By providing a solution based on a simplified analytical approach, it serves as a benchmark for comparison with more complex numerical models. This verification is essential to ensure the reliability and accuracy of the numerical methods used in structural analysis.

However, a limitation of the semi-analytical approach lies in the scope of the mathematical model, which is constrained to only the initial phase of the process, specifically up until the point when the wave reaches the support, $t_{max} = L/c$. This restriction arises due to the nature of the function, representing the moving stiffness. The structure of the moving stiffness is such that it does not account for wave reflections from the fixed end of the rod, thus rendering any solution beyond this initial phase physically invalid. Extending the model to include wave reflections would require significantly more complex mathematical modifications to the stiffness function, complicating the analytical framework considerably.

Despite this limitation, the semi-analytical solution is sufficient for verifying the developed numerical model, which will be presented in detail later in the manuscript. The simplicity and clarity of this solution provide a robust preliminary check against the numerical results, ensuring that the basic dynamics of the system are captured accurately before more advanced numerical methods are applied. Additionally, it is important to note that this semi-analytical solution is valid only for the specific boundary conditions outlined in equation (6.1), further emphasizing the need for caution when generalizing the results to other scenarios.

While the semi-analytical approach offers valuable insights and serves as an effective verification tool, its application is limited by the assumptions inherent in the mathematical model. These limitations do not, however, detract from its usefulness in the context of validating the numerical methods, such as the finite element method, which are capable of handling more complex boundary conditions and wave reflections.

6.1.3 Finite element approach of a moving stiffness

In this subsection, we introduce a numerical model that describes the behavior of moving stiffness as it travels along the length of the rod. The classical finite element method is employed to develop this model, offering a well-established approach for discretizing and solving the governing equations of the system. By discretizing the rod into finite elements, the dynamic interactions caused by the moving stiffness are captured in a systematic manner. While the model is initially constructed using a

traditional FEM framework, it is worth noting that the resulting matrix formulation can be seamlessly transformed into a space-time approach. Fig. 6.3 illustrates the setup of the problem, depicting a rod that is fixed at its left end. The diagram helps visualize the physical system under consideration, where the moving stiffness progresses along the rod while the fixed boundary at the left end constrains its movement. The rod is

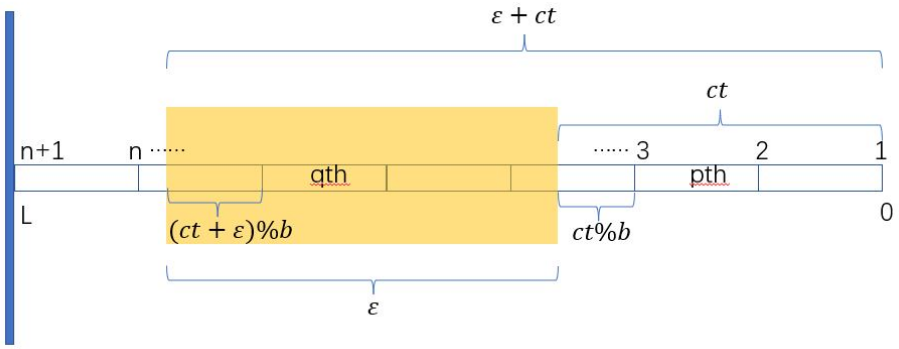


Figure 6.3: Finite element model of rod subjected to the travelling stiffness zone.

divided into n finite elements of length b. The moving element is marked in yellow and moves at c wave speed. The parameters informing about the width of the yellow zone and where it is located at a given moment of time are as follows

$$p = \frac{ct - ct\%b}{b}\,, \tag{6.18}$$

$$q = \frac{(ct+\epsilon) - (ct+\epsilon)\%b}{b}\,, \tag{6.19}$$

where % denotes modulo operation. The considered moving zone of length ϵ may include only 1 element, but also several elements. Fig.6.4 shows the case of the two involved elements. In order to solve this problem we assume the linear shape functions,

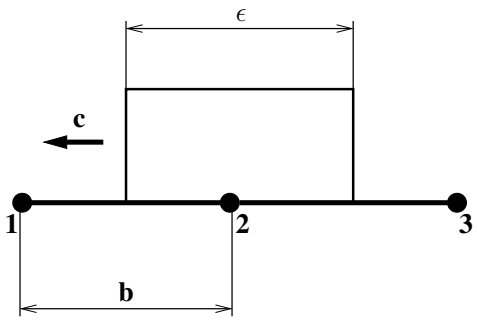


Figure 6.4: The travelling stiffness zone on the two elements involved.

then the displacement can be expressed as follows

$$w(x,t) = \left(1 - \frac{x}{b}\right) w_1(t) + \frac{x}{b} w_2(t)\,, \tag{6.20}$$

and similarly virtual displacements

$$w^*(x) = \left(1 - \frac{x}{b}\right) w_1 + \frac{x}{b} w_2\,. \tag{6.21}$$

According to (6.8) virtual energy of the travelling zone can be presented in the following form

$$\begin{aligned}\hat{\Pi} = \hat{E}A \int_0^b w^* \Big\{ &\left[H(x-(b-\epsilon-ct)) - H(x-(b-ct))\right] \frac{\partial^2 w}{\partial x^2} \\ &- \left[\delta(x-(b-\epsilon-ct)) - \delta(x-(b-ct))\right] \frac{\partial w}{\partial x} \Big\} \,\mathrm{d}x\,.\end{aligned} \tag{6.22}$$

The above integrals can be computed considering (6.20) and (6.21) as follows

$$\int_0^b w^*\, H(x-(b-\epsilon-ct)) \frac{\partial^2 w}{\partial x^2} \mathrm{d}x = \int_{b-\epsilon-ct}^b w^* \frac{\partial^2 w}{\partial x^2} \mathrm{d}x\,, \tag{6.23}$$

$$\int_0^b w^*\, \delta(x-(b-\epsilon-ct)) \frac{\partial w}{\partial x} \mathrm{d}x = \left(w^* \frac{\partial w}{\partial x}\right)_{x=b-\epsilon-ct}\,. \tag{6.24}$$

Taking into account Fig. 6.3, we can divide the moving component (covered in yellow) into 3 parts. The first part of the moving component is all elements completely covered by the moving component ($(p+2)$th to qth). In these elements, the result of the integral is given by the formula

$$\hat{E}A \int_0^b w^* \frac{\partial^2 w}{\partial x^2} dx = \hat{E}A \int_0^b \frac{\partial w^*}{\partial x} \frac{\partial w}{\partial x} dx = \frac{\hat{E}A}{b} \begin{bmatrix} w_1^* & w_2^* \end{bmatrix} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}\,. \tag{6.25}$$

As a result of the minimization of the energy, and considering all covered elements, the results are as follows

$$\begin{aligned}\mathbf{W_1} = &\frac{\hat{E}A}{b} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} w_{p+2} \\ w_{p+3} \end{bmatrix} + \frac{\hat{E}A}{b} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} w_{p+3} \\ w_{p+4} \end{bmatrix} + \dots \\ &+ \frac{\hat{E}A}{b} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} w_q \\ w_{q+1} \end{bmatrix}\,.\end{aligned} \tag{6.26}$$

The second part of the moving component refer to the elements that are partially covered by the moving component. According to $(p+1)$th and $(q+1)$th elements, integrals can be written in the following forms

$$\begin{aligned}\hat{E}A\int_{ct\%b}^{b} w^*\frac{\partial^2 w}{\partial x^2}dx &= \hat{E}A\int_{ct\%b}^{b}\frac{\partial w^*}{\partial x}\frac{\partial w}{\partial x}dx \\ &= \frac{\hat{E}A}{b^2}(b-ct\%b)\begin{bmatrix} w^*_{p+1} & w^*_{p+2}\end{bmatrix}\begin{bmatrix}1 & -1\\ -1 & 1\end{bmatrix}\begin{bmatrix}w_{p+1}\\ w_{p+2}\end{bmatrix},\end{aligned} \tag{6.27}$$

$$\begin{aligned}\hat{E}A\int_{0}^{(ct+\epsilon)\%b} w^*\frac{\partial^2 w}{\partial x^2}dx &= \hat{E}A\int_{0}^{(ct+\epsilon)\%b}\frac{\partial w^*}{\partial x}\frac{\partial w}{\partial x}dx \\ &= \frac{\hat{E}A}{b^2}((ct+\epsilon)\%b)\begin{bmatrix} w^*_{q+1} & w^*_{q+2}\end{bmatrix}\begin{bmatrix}1 & -1\\ -1 & 1\end{bmatrix}\begin{bmatrix}w_{q+1}\\ w_{q+2}\end{bmatrix}.\end{aligned} \tag{6.28}$$

Similarly, as a result of the minimization of the energy we obtain

$$\begin{aligned}\mathbf{W_2} &= \frac{\hat{E}A}{b^2}(b-ct\%b)\begin{bmatrix}1 & -1\\ -1 & 1\end{bmatrix}\begin{bmatrix}w_{p+1}\\ w_{p+2}\end{bmatrix} \\ &+ \frac{\hat{E}A}{b^2}((ct+\epsilon)\%b)\begin{bmatrix}1 & -1\\ -1 & 1\end{bmatrix}\begin{bmatrix}w_{q+1}\\ w_{q+2}\end{bmatrix}.\end{aligned} \tag{6.29}$$

For the second part calculation

$$\begin{aligned}\hat{E}A\left(w^*\frac{\partial w}{\partial x}\right) &= \frac{\hat{E}A}{b}\begin{bmatrix}w^*_1 & w^*_2\end{bmatrix}\begin{bmatrix}\frac{1}{b}x\\ 1-\frac{1}{b}x\end{bmatrix}\begin{bmatrix}1 & -1\end{bmatrix}\begin{bmatrix}w_1\\ w_2\end{bmatrix} \\ &= \frac{\hat{E}A}{b^2}\begin{bmatrix}w^*_1 & w^*_2\end{bmatrix}\begin{bmatrix}x & -x\\ b-x & x-b\end{bmatrix}\begin{bmatrix}w_1\\ w_2\end{bmatrix}.\end{aligned} \tag{6.30}$$

As a result of the minimization of the energy, and substituting $b-ct\%b$ and $(ct+\epsilon)\%b$ to x we obtain

$$\begin{aligned}\mathbf{W_3} &= \frac{\hat{E}A}{b^2}\begin{bmatrix}b-ct\%b & ct\%b-b\\ ct\%b & -ct\%b\end{bmatrix}\begin{bmatrix}w_{p+1}\\ w_{p+2}\end{bmatrix} \\ &+\frac{\hat{E}A}{b^2}\begin{bmatrix}(ct+\epsilon)\%b & -(ct+\epsilon)\%b\\ b-(ct+\epsilon)\%b & (ct+\epsilon)\%b-b\end{bmatrix}\begin{bmatrix}w_{q+1}\\ w_{q+2}\end{bmatrix}.\end{aligned} \tag{6.31}$$

Finally, vector describing the travelling zone can be written in the following form

$$
\begin{aligned}
&\mathbf{W_{move}} = \mathbf{W_1} + \mathbf{W_2} + \mathbf{W_3} \\
&= \frac{\hat{E}A}{b}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} w_{p+2} \\ w_{p+3} \end{bmatrix} + \frac{\hat{E}A}{b}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} w_{p+3} \\ w_{p+4} \end{bmatrix} + \dots \\
&+ \frac{\hat{E}A}{b}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} w_{q} \\ w_{q+1} \end{bmatrix} + \frac{\hat{E}A}{b^2}(b - ct\%b)\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} w_{p+1} \\ w_{p+2} \end{bmatrix} \\
&+ \frac{\hat{E}A}{b^2}((ct+\epsilon)\%b)\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} w_{q+1} \\ w_{q+2} \end{bmatrix} \\
&+ \frac{\hat{E}A}{b^2}\begin{bmatrix} b - ct\%b & ct\%b - b \\ ct\%b & -ct\%b \end{bmatrix}\begin{bmatrix} w_{p+1} \\ w_{p+2} \end{bmatrix} \\
&+ \frac{\hat{E}A}{b^2}\begin{bmatrix} (ct+\epsilon)\%b & -(ct+\epsilon)\%b \\ b - (ct+\epsilon)\%b & (ct+\epsilon)\%b - b \end{bmatrix}\begin{bmatrix} w_{q+1} \\ w_{q+2} \end{bmatrix}.
\end{aligned}
\tag{6.32}
$$

6.1.4 Results

A test example that illustrates the correctness of the numerical model is presented. Computer calculations were performed using the characteristic matrices of the rod derived in subsection 4.1.3.
The data set used in the simulations:

- Young's modulus - $E = 100\,[GPa]$
- Moving Young's modulus - $\hat{E} = 0.9E$
- Density of a rod - $\rho = 10\,[\frac{kg}{m^3}]$
- Length of a rod - $L = 0.05\,[m]$
- Width of a moving zone - $\epsilon = 0.01\,[L]$
- Initial velocity - $v_0 = 1\,[\frac{m}{s}]$

Figs. 6.5 and 6.6 present the obtained results, compared with the semi-analytical solution from the section 6.1.2. The previously presented analytical solution validates the results obtained using the numerical method. While the analytical solution is limited to a specific case, the numerical solutions provide a broader range of possibilities for analysis. This demonstrates the robustness and flexibility of numerical methods in solving complex dynamic problems.

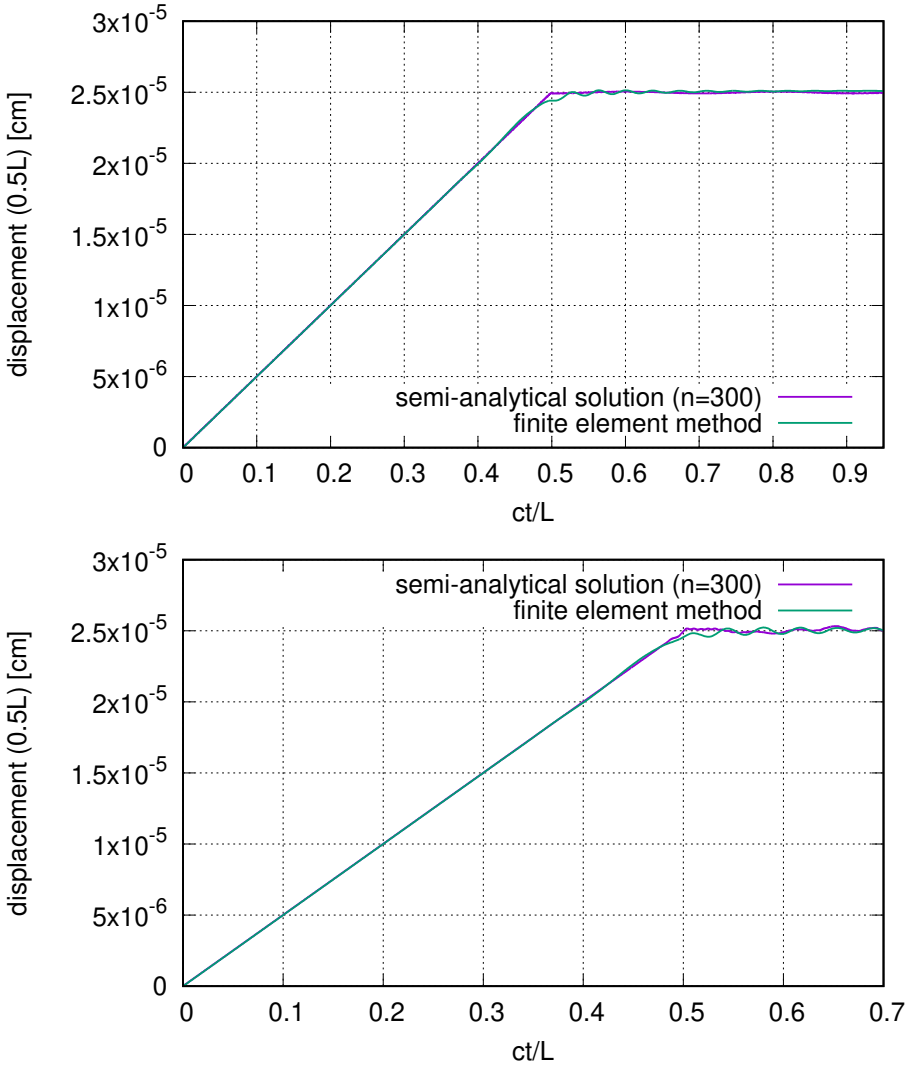


Figure 6.5: Displacements in the middle of the rod - local strengthening (above) and local softening (below).

Below, we will discuss the results obtained numerically using the matrices described in subsection 6.1.3, in the case of a rod with one end fixed and the free end loaded with a force impulse. We will consider the acceleration of the free end of the rod and the axial force appearing in the finite element at the free end. The influence of the width of the hardening/softening zone will be exhibited as well as the ratio of the contribution of $\hat{E}$ to the initial elasticity modulus E.

Fig. 6.7 presents the force and acceleration at the subjected end of the rod for both softened and stiffened travelling zones. The following parameters were assumed for the analysis: a ratio of $\hat{E}/E = 0.9$, and the width of the travelling stiffness modifi-

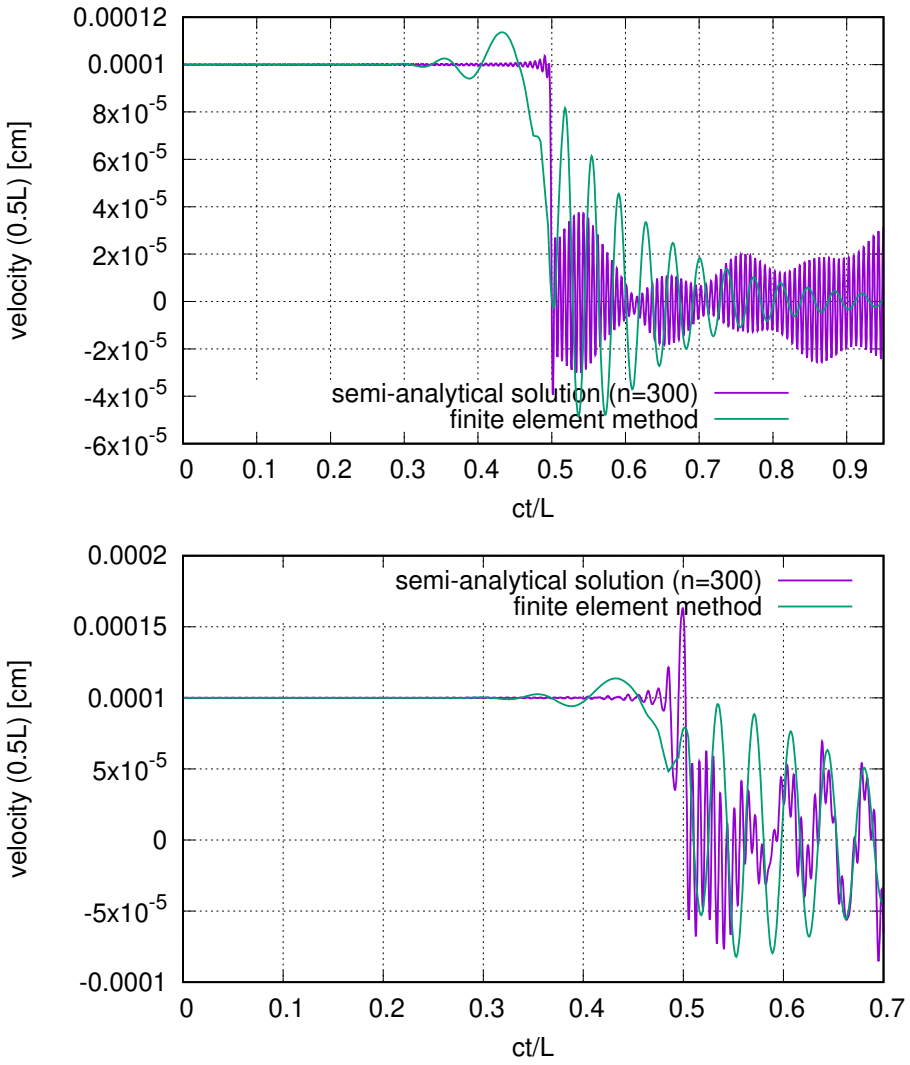


Figure 6.6: Velocities in the middle of the rod - local strengthening (above) and local softening (below).

cation zone was set to 1% of the total length of the rod. In the case of the softening material, the axial force observed in the edge element immediately after the impact is approximately 15% of the force value in the unmodified material. Conversely, in the case of the hardening material, the force is observed to be about 60% higher than that in the unmodified material. Furthermore, the acceleration recorded at the end of the rod shows significant variations depending on the stiffness modification. For the rod with a temporarily weakened material, the acceleration is 25% less than the reference acceleration measured in the unmodified rod. In contrast, the acceleration for the hardening material is 30% greater than the reference acceleration. These results highlight

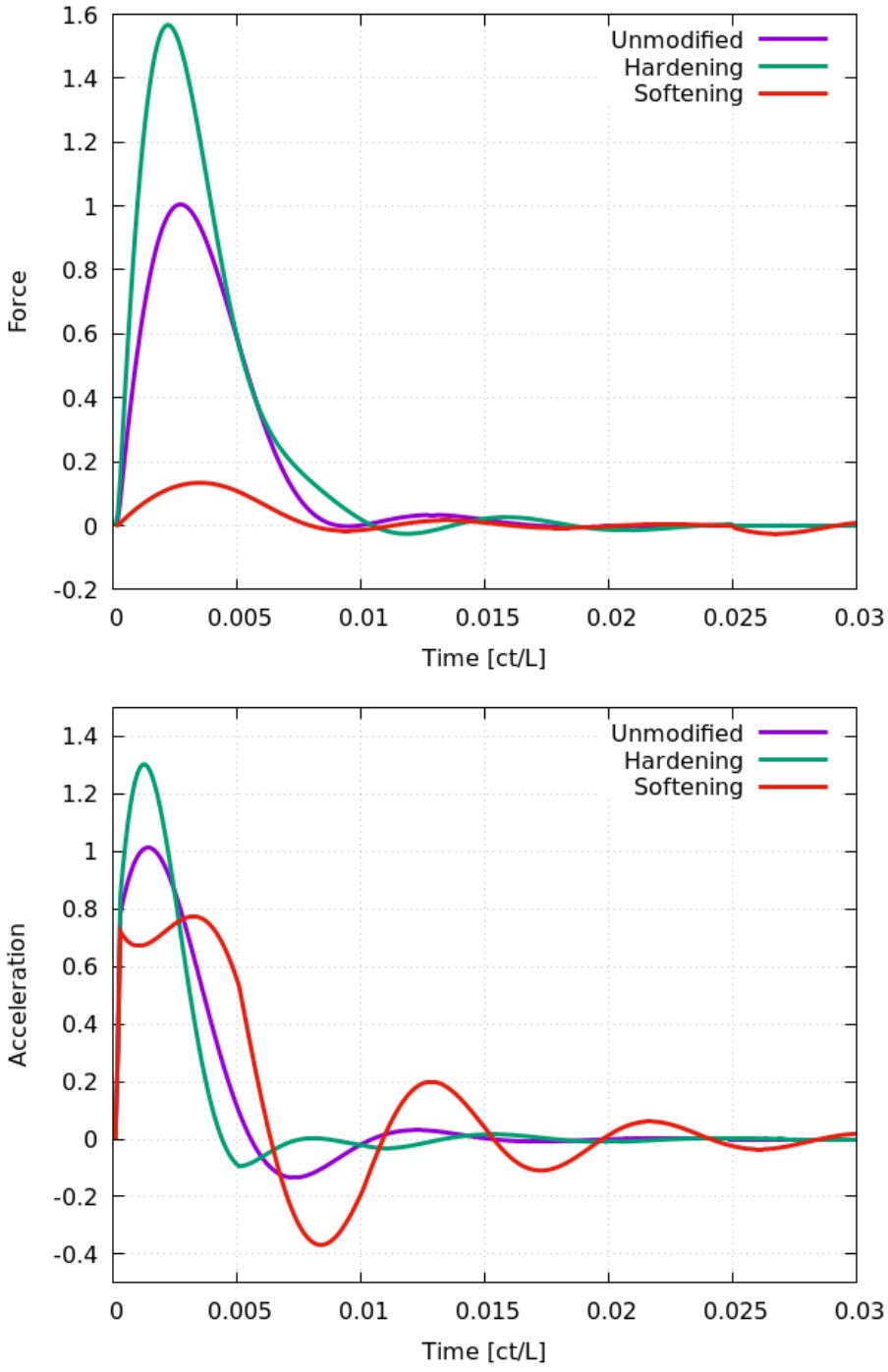


Figure 6.7: Force and acceleration at the subjected end for softened and stiffened travelling zone normalized to those obtained with unmodified material.

the substantial impact of local stiffness changes on the dynamic response of the rod, demonstrating that both softening and hardening modifications can significantly alter the force and acceleration experienced at the rod's end. The numerical data emphasize the importance of considering such local modifications in engineering applications where precise control of dynamic responses is critical.

Fig. 6.8 depicts the impact of varying amounts of local material weakening on the overall response exhibited by the system under consideration. The analysis considers

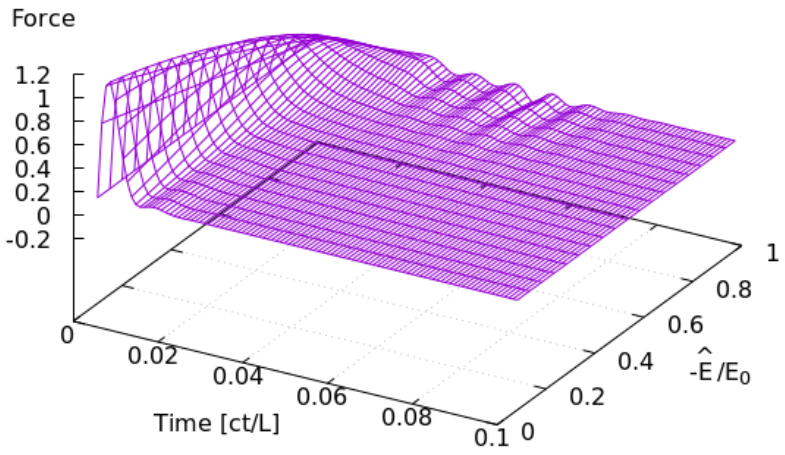


Figure 6.8: Force at the subjected free end depending on the relative additional elasticity module.

a spectrum of weakening scenarios for the modulus of elasticity denoted as $\hat{E}$, ranging from 0 to 95% of the nominal value of the base modulus of elasticity E characterizing the rod in question. In practical terms, the effective modulus $E + \hat{E}$ within the region affected by weakening experiences a transition from E to $0.05E$. Notably, a clear relationship emerges wherein the reduction in internal force experienced at the loaded extremity of the member aligns proportionally with the $\hat{E}/E$ ratio, as evidenced by the trends observed in the system response.

Fig. 6.9 illustrates the displacements in time corresponding to various values of the relative additional elasticity module $\hat{E}$. The displacements $w(x,t)$ can be directly correlated to the displacement of the free end denoted as w_0, assuming an unaltered elasticity module E. Specifically, in the particular scenario under analysis, the value

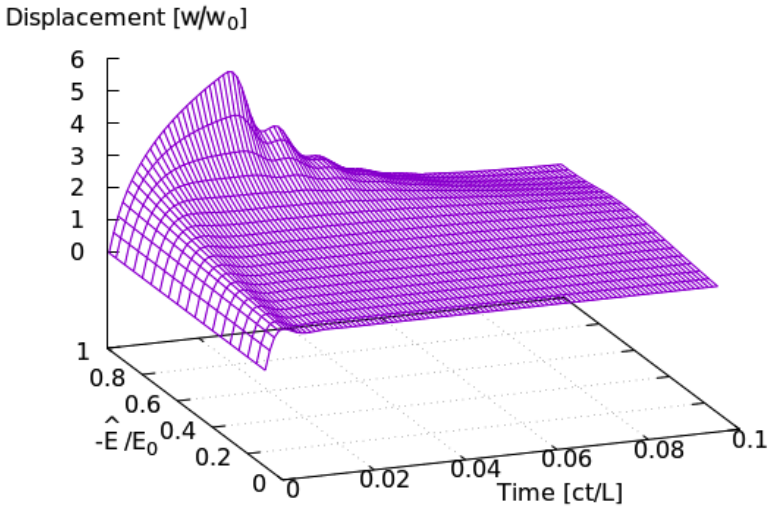


Figure 6.9: Displacement at the subjected free end depending on the relative additional elasticity module.

of w_0 is set at 1. A noteworthy observation from the data is the gradual and more pronounced advancement over time of the resultant displacement magnitude in instances where the material exhibits a softened characteristic. This particular behavior is advantageous as it enables a more gradual progression in the recorded accelerations, thereby contributing to a softer and less abrupt nature of the impacts observed.

Fig. 6.10 demonstrates that the width of the moving softened gap can be relatively narrow. While a wider gap marginally improves the results compared to a narrower gap by reducing the acting forces and accelerations, this improvement is slight. This phenomenon is theoretically explicable because, in an elastic material, the wavefront is sharply defined without parabolization of the governing equation and the blurring effects of damping. Consequently, only a narrow zone is significantly involved in the process described in this study. This observation underscores the localized nature of the stiffness modification's impact on the dynamic response of the rod, indicating that even small modifications in stiffness can have pronounced effects on the force and acceleration within the affected region.

Fig. 6.11 depicts the amplitude in terms of the width of the modified moving zone. The increasing amplitude along with the lengthening of the first period of vibration results in a reduced value of the acceleration at the free end of the rod. The extension of the oscillation period, even with increasing amplitude, leads to a reduction in the

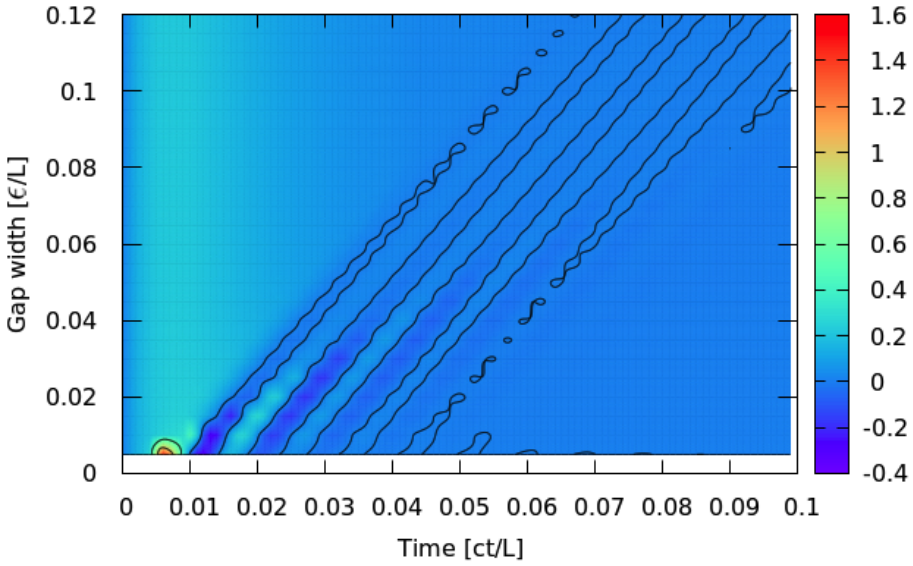


Figure 6.10: Internal force at the subjected free end depending on the modified zone width.

acceleration amplitudes. Fig. 6.11 shows a three-fold increase in displacement amplitudes, with a ten-fold increase in the time it takes to return the observed point to its equilibrium position. This is the reason for the significant decrease in accelerations and forces. It should be emphasized again that the narrow, softened moving zone somehow distributes the impulse energy over the area of the object.

6.1.5 Conclusions

This research study demonstrates that the local and temporary reduction in material strength has a significant impact on critical utility values, such as peak accelerations and peak force values experienced at the periphery of the tested object. The weakening of the material occurs specifically in areas characterized by high deformation velocities, precisely at the location where the traveling wave front emerges as a result of the impact. It is important to note that only a thin layer of the material undergoes weakening, while the remaining portion retains its original mechanical and functional properties without alteration.

The utilization of mathematical formulation and semi-analytical solution techniques in addressing the task has enabled the validation of the developed numerical model

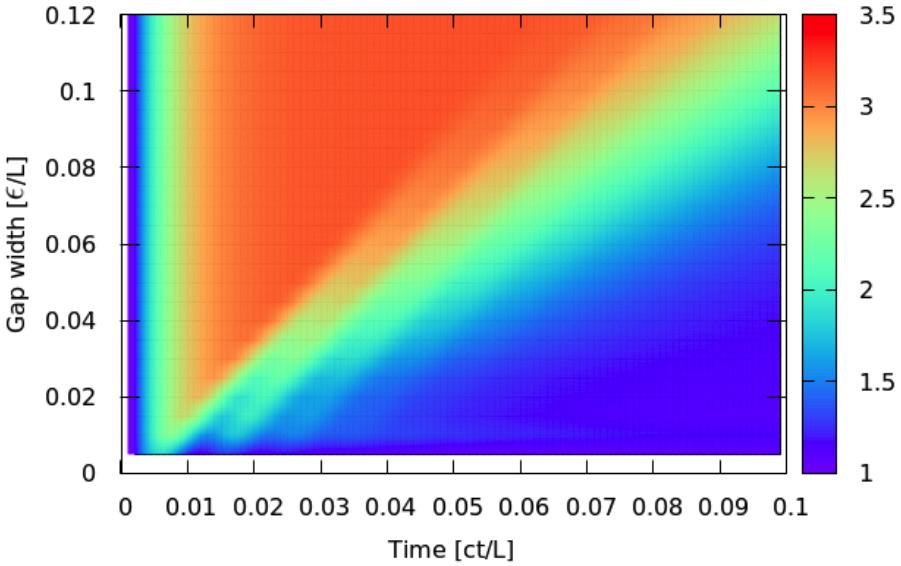


Figure 6.11: Displacements $w(x,t)/w_0$ at the subjected free end depending on the gap width.

based on finite elements. Within the realm of numerical simulation, ensuring the accuracy of depicting the propagation of the weakened zone front across the successive finite elements' domain holds paramount importance. The same level of significance is attributed to the transition at the end of the weakened zone, where a segment of the finite element domain experiences weakening while other parts remain unaffected. Moreover, the orientation of the elastic wave assumes critical importance in the numerical model. The consistency between semi-analytical and numerical findings was observed within the limits of assessable scope, accounting for the emergence of known artifacts.

The outcomes of the simulations indicate that materials exhibiting a localized reduction in stiffness offer a more efficient means of mitigating large, short-term acceleration peaks, thereby proving effective in accident prevention measures. Illustrative instances underscore that the mitigation of forces or accelerations can vary between 30 to 70% when compared to models constructed with conventional materials.

While materials with temporary reinforcement are relatively easy to manufacture and procure, the acquisition of weakening materials poses a current challenge. Hence, the ongoing efforts directed toward advancing research in this domain are deemed imperative and warrant further exploration.

6.2 Mechanical metamaterial

Researchers are actively exploring the applications of mechanical metamaterials in the development of lighter and stronger structures. By using these materials, engineers can create components that maintain structural integrity while significantly reducing weight, which is crucial in industries like aerospace and automotive. Lighter materials contribute to better fuel efficiency and enhanced performance, making them invaluable for modern engineering challenges. The ability to tailor mechanical properties also allows for greater design freedom, paving the way for innovative architectures and systems that were previously unattainable. As research progresses, the potential of mechanical metamaterials continues to expand, promising advancements in numerous fields, from construction to consumer electronics, revolutionizing how we think about and use materials in our everyday lives.

Engineering practice shows that currently available smart materials have significant drawbacks that limit their potential application in practice. This is excessive mass, but above all the need to use complex additional devices that allow, through control, modification of parameters. The solution to the problems may turn out to be mechanical metamaterials, whose dynamic properties are determined to a greater extent by their geometric structure than by the base materials that constitute the carrier. They are lightweight, and what is more, they are characterized by full autonomy of operation. Thanks to the use of effective computational tools, it seems possible to design prototype metamaterials that exhibit unusual and so far unexplored functional features.

In the literature, we can find metamaterials characterized by negative Poisson's ratio, which use elastic instability of the structure [39]. Fig. 6.12 shows an elastic material with regular geometry. The sample is compressed, and adjacent holes interact during deformation, leading to self-activating jumps in subsequent material layers. This elastic instability of the structure is a kind of switch that allows semi-active modification of the material characteristics. An appropriately designed geometric structure of such a material can be used to implement a two-state control strategy.

In this subsection, the problem of plane stress state is presented. The structure is subjected to an impulsive load. A viscoplastic material model was used in computer simulations. According to the space-time model from subsection 4.2.3, numerical calculations of the metamaterial structure were performed and the results of computer simulation were presented.

6.2.1 Space-time viscoplastic model

In the process of creating a material model intended for dynamic simulation, it is essential to address the complexities involved in identifying multiple parameters, as well as the significant computational resources necessary for running these simulations.

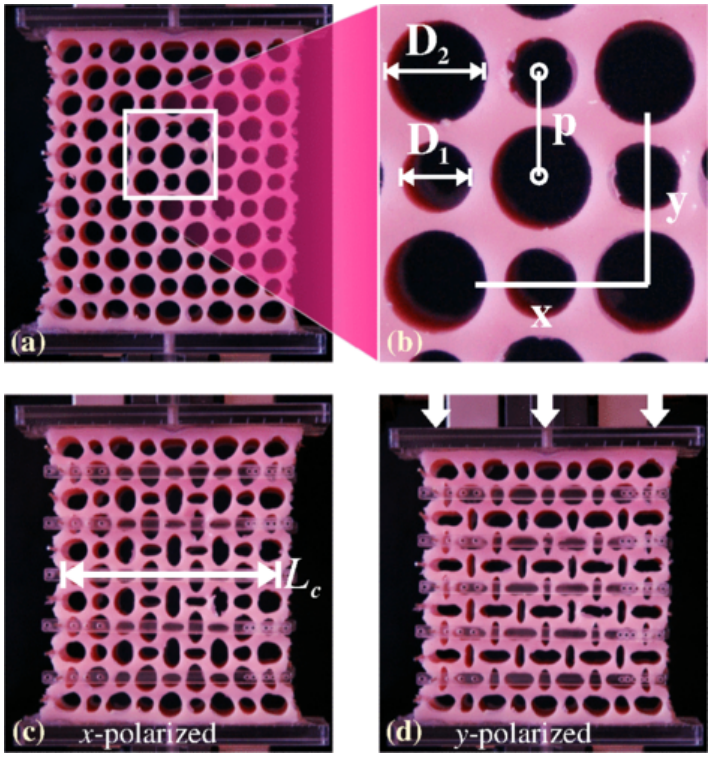


Figure 6.12: A mechanical metamaterial biholar sheet [39].

This example focuses on a specific uniform, isotropic, and non-compressive material characterized by a relatively uncomplicated Norton-Hoff formula for viscoplastic solids [82]. The model is designed to be both straightforward and user-friendly, requiring only a few parameters to be identified through experimental means. Initially utilizing the nonlinear model, modifications were later implemented by adding rate-dependent softening and an extra modifier to enhance its compatibility with the reference results.

A space-time approach was employed to describe the problem, as it allows for a more convenient representation of geometric changes during plastic flow through simultaneous interpolation in both space and time. Additionally, the space-time finite element method enables parallel computing, and when appropriately tailored, it can yield results significantly faster than traditional finite element method. In the conventional Norton-Hoff model, the deviatoric component of the Cauchy stress tensor, denoted as $\boldsymbol{\tau}'$, is defined in the following manner

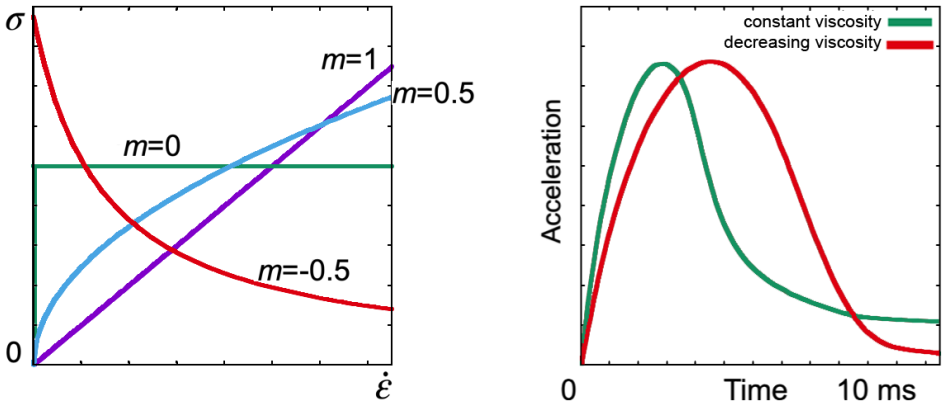
$$\boldsymbol{\tau}' = \frac{K\dot{\boldsymbol{\varepsilon}}}{\left(\sqrt{3}D\right)^{1-m}}, \tag{6.33}$$

where

$$\dot{\boldsymbol{\varepsilon}} = \frac{1}{2}(\operatorname{grad}\mathbf{v} + \operatorname{grad}^T\mathbf{v}) = \mathcal{D}\mathbf{v}, \tag{6.34}$$

$$D = \sqrt{\frac{2}{3}\dot{\boldsymbol{\varepsilon}}^T \dot{\boldsymbol{\varepsilon}}}. \tag{6.35}$$

The material viscosity is represented by K, and the deformation rate sensitivity is denoted by m, where $0 \leq m \leq 1$. Specifically, $m = 0$ indicates a perfectly plastic material, whereas $m = 1$ corresponds to a Newtonian fluid, as illustrated in Fig. 6.13a.



a) b)

Figure 6.13: Norton-Hoff material response for different rate sensitivity m (a) and the resulting acceleration over time, compared with reference literature results (b).

The constraint of having positive values for m limits the ability to achieve an accurate alignment between the numerical results and the existing literature. As a result, the model was adjusted to permit negative values in the range of $-1 \leq m \leq 0$, leading to the observed weakening illustrated in Fig. 6.13a. This modification causes viscosity to decrease as deformation rates increase, a behavior typically associated with certain biological fluids, as well as jelly and gel-like substances. Consequently, a revised version of the generalized material law (6.33)–(6.35) is proposed in the following form

$$\boldsymbol{\tau}' = s \frac{K\dot{\boldsymbol{\varepsilon}}}{\left(\sqrt{3}D\right)^{1-m}} \tag{6.36}$$

with permissible

$$m \in \langle -1, 1 \rangle \ . \tag{6.37}$$

The viscosity K and the parameter m can be modified, in addition to the introduction of the multiplier s, which facilitates various hyperbolic configurations of the dependency. This enhancement enables the acquisition of distinct viscoplastic softening characteristics.

In comparing the material response at various values of m (Fig. 6.13b), it is evident that assuming $m < 0$ (indicating softening and decreasing viscosity) results in a rapid

initial peak in acceleration, followed by a gentle decline. This behavior closely mirrors the reference results.

The virtual work within a space-time layer can be represented by the following integral

$$\int_0^h \int_V (\mathbf{v}^*)^T \rho \frac{\partial \mathbf{v}}{\partial t} \mathrm{d}\Omega + \int_0^h \int_V (\dot{\boldsymbol{\varepsilon}}^*)^T \boldsymbol{\tau}' \mathrm{d}\Omega = \int_0^h \int_{\partial V} (\mathbf{v}^*)^T \, \mathbf{f} \, \mathrm{d}(\partial V) \, \mathrm{d}t \,, \tag{6.38}$$

where h is the considered time step. Taking into account the second term in (6.38), the interpolation of the virtual velocities utilized in (6.34) along with the actual velocities in (6.36) makes it possible to express this term as follows

$$s \int_\Omega \dot{\mathbf{q}}^T (\mathcal{D}\mathbf{N}^*)^T \frac{K}{(\sqrt{3}D)^{1-m}} \mathcal{D}\mathbf{N} \, \mathrm{d}\Omega \, \cdot \dot{\mathbf{q}} \,. \tag{6.39}$$

Following several transformations, the matrix associated with this term and connected to the potential energy will be represented in the following manner

$$\mathbf{K} = \int_0^h \int_{V(t)} \left(\mathcal{D}\mathbf{N}^*(\mathbf{x}, t) \right)^T \mathbf{E} \, \mathcal{D}\mathbf{N}(\mathbf{x}, t) \mathrm{d}V \, \mathrm{d}t \,. \tag{6.40}$$

It is important to highlight the analogy with the elastic case. In viscoplastic elements, the matrix $\mathbf{E}$ now varies with the strain rate. The other matrices can be derived similarly, leading to the final equation being expressed in the following matrix form

$$(\mathbf{K} + \mathbf{M} + \mathbf{Z})\dot{\mathbf{q}} = \mathbf{F}. \tag{6.41}$$

The matrix $\mathbf{K}$ represents the stiffness, $\mathbf{M}$ accounts for the effects of inertia, $\mathbf{Z}$ reflects the influence of external damping, while $\mathbf{F}$ denotes the external load vector. The condition of incompressibility

$$\mathrm{div} \, \mathbf{v} = 0 \,, \tag{6.42}$$

which is included in the penalty function, can be readily formulated into an appropriate expression. The penalty function term that contributes to the functional is represented as follows

$$\frac{1}{2} \lambda \, \mathrm{div} \, \mathbf{v}^* \, \mathrm{div} \, \mathbf{v} = \frac{1}{2} \dot{\mathbf{q}}^T \int_\Omega (\mathcal{D}\mathbf{N}^*)^T \boldsymbol{\Lambda} \mathcal{D}\mathbf{N} \mathrm{d}\Omega \cdot \dot{\mathbf{q}} \,, \tag{6.43}$$

where λ represents the coefficient of the penalty function, and the matrix $\boldsymbol{\Lambda}$ takes the following form in three-dimensional problems

$$\boldsymbol{\Lambda} = \lambda \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} . \tag{6.44}$$

6.2.2 Numerical simulations

Let us consider the rectangular structure with three holes. The plane-stress problem is considered so we can use the numerical model from subsection 4.2.3. The example structure is divided into two parts, an upper elastic part and a lower viscoplastic part. The description of the viscoplastic material and the method of entering material parameters into the numerical model are presented in the 6.2.1 subsection. Fig. 6.14 presents a geometric diagram of the calculation example. The geometric data of the structure are presented in cm. The structure is fixed at the bottom, which means that

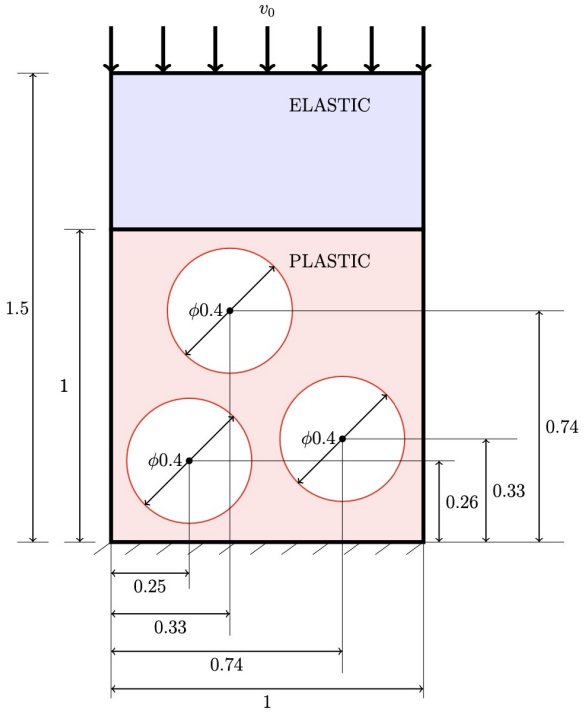


Figure 6.14: The outline of the problem.

the degrees of freedom about the x and y axes are taken away. This prevents the lower edge of the rectangle from moving in any direction. The deformation process of the structure starts from the initial velocity v_0=-0.005 cm/μs.

The computer simulations use the units cm, g, μs so that we do not deal with large numbers that could affect the inaccuracy of the solutions.
The following data set were used:

- elastic part
  - Young's modulus - E=69 GPa=0.69 g/cm/μs^2

  - Poisson ratio - ν=0.33
  - mass density - ρ=2720 kg/m^3=2.72 g/cm^3
  - damping coefficient - η_z=500 kg/s=0.5 g/μs
- viscoplastic part
  - viscosity - K=0.001
  - deformation rate sensitivity - m=0.267
  - mass density - ρ=2720 kg/m^3=2.72 g/cm^3

The freely available program "triangle" was used to generate the finite element mesh. Ultimately, due to the size of the test example, it was decided to use 967 mesh nodes, which resulted in 1750 finite elements. According to the obtained division into triangles, a simplex-shaped space-time mesh was built.

Using the calculation scheme (3.12) and characteristic matrices (4.99), (4.114) and (4.130), a computer program was developed to simulate the dynamic behavior of the example problem. Deformations of the structure lead to a rapid change of geometry. Part of the deformed material begins to be in contact with other parts of the structure, and therefore it is necessary to study dynamically changing contact zones in the material. Fig. 6.15 illustrates the progression of deformation in the tested structure over a series of successive time steps. In addition, we specifically examined the acceleration of a point with initial coordinates (0.5, 1.5), which is situated along the upper edge of the plate. The measured acceleration values for this point are depicted in Fig. 6.15. During the initial phase of the simulation, we observe that the acceleration values are notably high. This behavior indicates that the structure is responding within the elastic range, where it deforms elastically and returns to its original shape upon unloading. As the simulation progresses, the material begins to exhibit plastic behavior, marking a transition from elastic deformation to plastic deformation. At this critical juncture, we witness a sudden relative deformation of the holes in the structure, leading to a significant reduction in the acceleration of the point in question. This abrupt change illustrates the shift in material behavior as it yields and accommodates additional displacement, highlighting the complex interaction of forces at play within the structure under analysis.

Effective parallelization hinges on a well-defined flow of information during the initial stages of the computational solution process. In the first clock cycle, the calculations can only yield the unknown values for a single node, establishing a foundational reference point for subsequent computations. This first stage is crucial, as it sets the groundwork for the iterative process that follows. In the subsequent clock cycles, the methodology systematically incorporates neighboring nodes one at a time into the calculations. This incremental approach allows for a structured progression through the

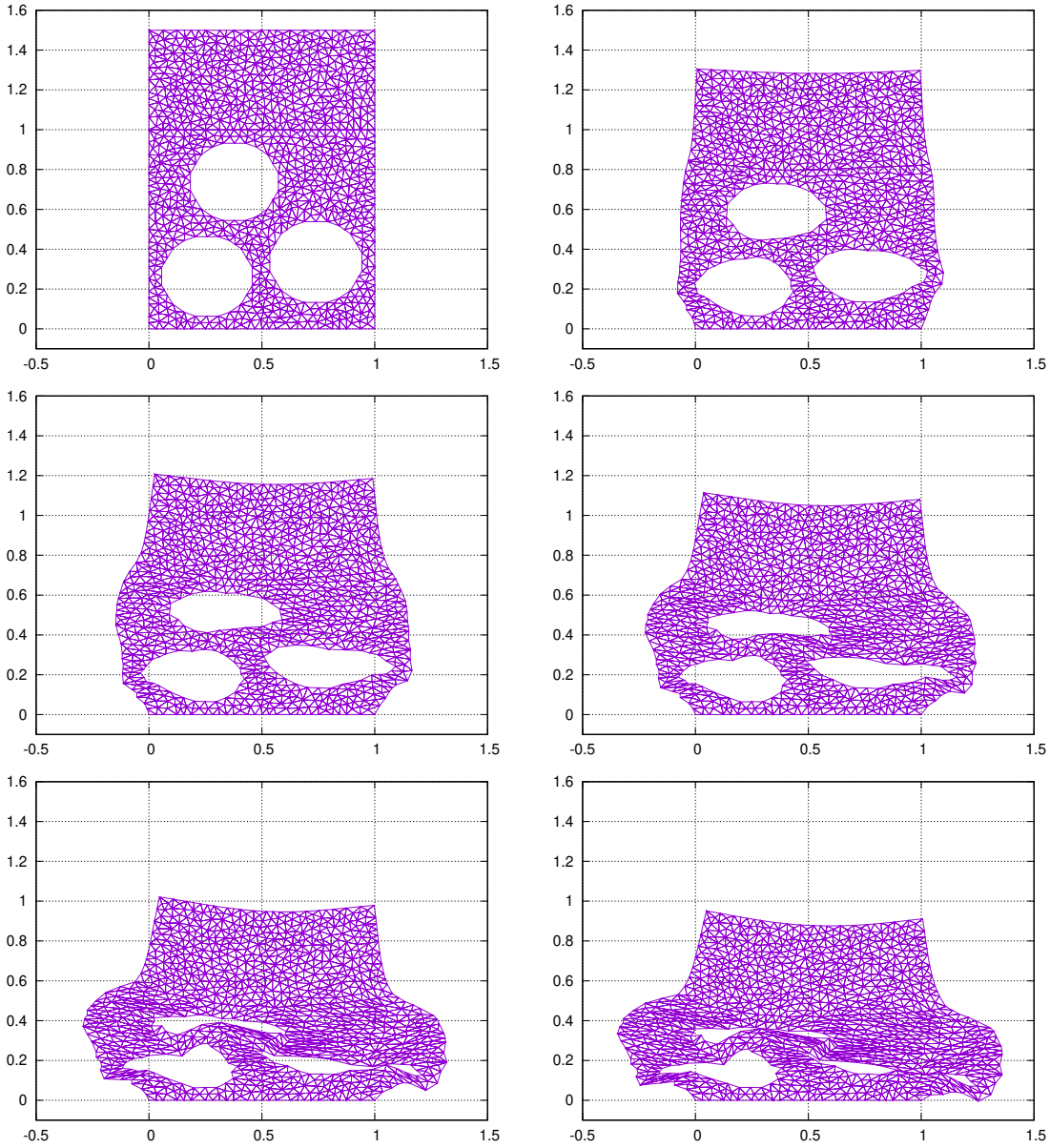


Figure 6.15: Distribution of the in-plane deformation in successive stages.

spatial domain, ensuring that each additional node is directly influenced by the results of its adjacent nodes. Consequently, the computational front line in the space-time continuum evolves as more nodes are included, effectively creating a "slope" that represents the wave of computation spreading through the system. As the process unfolds, after a determined number of initial computational steps during each cycle, every node that is integrated into the calculations corresponds to a different moment in time. This

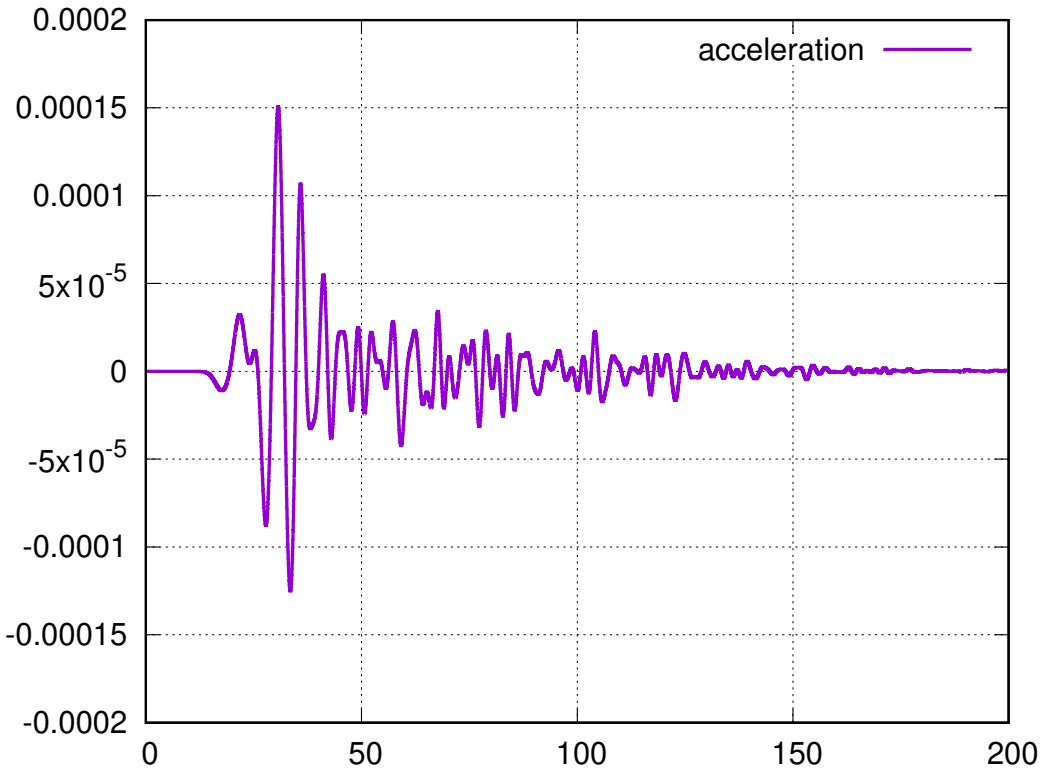


Figure 6.16: Acceleration at a selected point of the structure.

creates a dynamic and evolving computational lattice, where each position on this grid reflects the state of the system at various temporal intervals. The careful organization of information flow not only enhances computational efficiency but also ensures that the results obtained at each node are synchronized with those of its neighbors. The result is a cohesive and coherent calculation that mirrors the physical behavior of the dynamic structure being analyzed. In summary, the orderly flow of information initiated in the first clock cycle serves as the catalyst for a broader and more intricate web of calculations, which ultimately enables accurate modeling of the dynamic response of structures over time. This systematic approach to parallelization maximizes the use of computational resources, reduces processing time, and leads to more robust and reliable outcomes in the analysis of complex dynamic systems. As the methodology continues to unfold, it opens up new avenues for exploring various structural behaviors and enhancements within the realm of computational mechanics.

In the parallel space-time method, it is essential to incorporate finite element characteristic matrices into a global matrix while ensuring that concurrent access to shared memory is safeguarded against multiple processors. To achieve this, the process must be organized into sub-stages that generate element matrices from nodes that are sep-

arated by at least one spatial element. This precaution ensures that concurrently computed elements do not share common nodes, thus preventing conflicts in memory access. In the context of a two-dimensional mesh composed of regularly arranged triangles, we can conceptually identify clusters of triangles, known as "rosettes," that surround individual nodes. During consecutive computational cycles, the matrix of elements surrounding a specific node is identified, and the associated coefficients are sequentially integrated into the global matrix. This systematic approach requires that the process be carried out in a step-by-step manner for each spatial element within the designated rosette. By following this methodical procedure, we ensure that the integration of finite element matrices is not only efficient but also safe from the risks of simultaneous memory access. As each node's contributions are incorporated into the global matrix one rosette at a time, the overall coherence and integrity of the numerical analysis are maintained, facilitating accurate simulation of dynamic structures.

6.2.3 Conclusions

The proposed methodology presents a groundbreaking approach for the parallel bulk computation of dynamic structures, marking a significant advancement in computational techniques within structural engineering and dynamic analysis. This innovative framework allows for the direct decoupling of the resulting system of equations, enabling the efficient distribution of computational tasks across multiple parallel processing units. By leveraging parallel computing, the methodology can exploit the capabilities of modern hardware, significantly accelerating analysis times.

A key feature of this approach is the transformation of the resulting global matrix into a triangular form, which retains only non-zero coefficients. This simplification not only enhances memory efficiency but also improves computational speed, as operations on sparse matrices tend to require fewer resources and time compared to dense matrices. Proper spatial discretization is essential in this context, as it aids in effectively separating zones characterized by nonlinear behavior that necessitates iterative solutions. By isolating these nonlinear regions, the methodology allows for iterative recomputation only of small sections of the algebraic equation system, rather than the entire system. This targeted recomputation significantly reduces overall computational effort, permitting a frontal solution of the time-stepping scheme that contributes to enhanced efficiency.

Chapter 7

Summary of research work

In this dissertation, the initial two chapters outline the objectives of the thesis and the current research being presented. The focus is on the concept of decoupling systems of equations, which allows for parallel computations across different segments of the system. This decoupling technique involves sequentially calculating unknown variables, enabling parallel processing and facilitating a progressive movement in the computation. Additionally, the discussion includes the simplex-shaped space-time finite element method, a strategy employed to solve differential equations. Simplex-shaped functions help define the space-time coordinates within various subelements, which is instrumental in tackling wave problems and understanding structural dynamics. The space-time finite element method also permits continuous adaptation of the spatial mesh over time, as well as the separation of resulting equations from the algebraic equation system during the formation of characteristic matrices, all without necessitating triangulation.

In Chapter 3, we provide an in-depth introduction to the space-time finite element method. We begin by discussing the flow of data within the mesh before deriving the relevant mathematical expressions in detail. This method incorporates space-time shape functions that are analogous to those used in classical finite element techniques. By integrating these shape functions into the governing equations of motion, we obtain the general findings for the system. We present the formulations for several essential matrices: the stiffness matrix, the inertia matrix, the internal damping matrix, and the external damping matrix. These matrices are vital for accurately depicting the dynamic behavior of the system.

Moreover, we delve into the properties of simplex elements and their function within the method. A key characteristic to highlight is that the strain displacement matrix is lower triangular, which simplifies the computational process and facilitates the resolution of the mechanical system. Furthermore, we provide a brief overview of the nonlinear iteration procedures used within a subdomain, which are critical for managing nonlinearities and ensuring solution accuracy in complex scenarios. Overall, this

chapter thoroughly explores the fundamentals of the space-time finite element method, emphasizing its mathematical underpinnings and practical applications.

In Chapter 4, we focus on selected examples of one- and two-dimensional structures to demonstrate the application of numerical methods in structural analysis. We construct a standard one-dimensional rod model. This involves conducting a thorough force analysis and deriving the one-dimensional partial differential equation of motion. By utilizing the space-time finite element method laid out in Chapter 3, we apply simplex elements to derive the stiffness and inertia matrices specific to this one-dimensional context. The force analysis enables us to accurately represent the dynamic behavior of the rod, leading to the formulation of its governing equations. By employing the space-time finite element method, we convert these equations into a discretized format suitable for numerical examination. The role of the simplex elements is crucial in this transformation, ensuring both the efficiency and precision of the resulting matrices. Following this, we illustrate the application of the method through a classic one-dimensional example. This specific model is solved using two numerical strategies: the semi-analytical method in conjunction with the Newmark technique, and the space-time finite element method. A comparison of the results from these two simulations is conducted to evaluate the effectiveness of the proposed method.

In the second part of Chapter 4, the focus shifts to addressing the challenges posed by large deformations within the context of plane-stress problems. We commence by deriving the fundamental relationship between stress and strain, offering a comprehensive understanding of the mechanics involved. This relationship is integral for solving the two-dimensional partial differential equations that describe deformation behavior. When the system undergoes large displacements, integrating nonlinear terms becomes essential for achieving accurate outcomes, as neglecting these nonlinearities can result in considerable errors. To tackle this, we derive the shape functions for a tetrahedral element in the two-dimensional case, specifically within the framework of the space-time finite element method. By incorporating these shape functions into the equations of motion, we yield numerical solutions for the system. We offer explicit formulations for both the inertia matrix and the stiffness matrix, presented in their linear and nonlinear forms. Including nonlinear terms is particularly vital for addressing large deformations, as these terms account for the geometric and material nonlinearities that develop. To validate our methodology, we provide a detailed numerical example where we apply the space-time finite element method to resolve the plane-stress problem under conditions of both small and large displacements. A comparison of outcomes is made to emphasize the importance of accounting for nonlinear terms in situations involving significant displacement. Our results unequivocally indicate that for large deformations, incorporating nonlinear terms is essential for acquiring accurate and trustworthy outcomes. The comparative analysis shows that the linear model fails to capture the actual behavior of the system under large displacements, while the nonlinear model offers a much closer approximation.

In Chapter 5, we delve into the practical applications of the space-time finite element method. To enhance clarity, we opt for a two-dimensional model for the simulation. This chapter provides an extensive examination of the characteristic matrices relevant to this model, detailing their configurations and significance in the computational workflow. We explore how the iterative process of solving the model is facilitated using parallel computing techniques. Specifically, we outline how computations are distributed across multiple threads to improve efficiency and decrease overall processing time. An algorithm governing this parallelized operational workflow is presented, emphasizing its effectiveness in optimizing the performance of the space-time finite element method.

In Chapter 6, we apply the space-time finite element method to two distinct examples that center on the development of advanced impact protection materials. The first example focuses on creating a smart elastic material that can modify its properties in response to the propagation of mechanical waves. We detail a comprehensive mathematical model for this innovative material. To solve the governing equations, we utilize both semi-analytical solutions and space-time finite element model. A comparative analysis of these solutions demonstrates that they yield similar results, showcasing the effectiveness of the numerical methodology. We further provide results illustrating the force and acceleration at the impacted end of the material in both softened and stiffened zones, normalized against the results obtained from unmodified material. Additionally, we investigate the force at the impacted free end as a function of the relative increase in the elasticity modulus.

The second example is centered on developing mechanical materials using the space-time finite element method to model a viscoplastic material. In this scenario, the space-time description proves beneficial due to its capability to manage evolving geometry during plastic flow. The simultaneous interpolation of both space and time afforded by the space-time finite element method is particularly effective in capturing the intricate behavior of viscoplastic materials. Moreover, the use of the space-time finite element method allows for efficient parallel computing, resulting in significantly reduced computation times relative to traditional finite element methods. The results confirm that the space-time finite element method, complemented by its parallel processing capabilities and sophisticated interpolation techniques, offers considerable advantages in the development and assessment of complex mechanical materials.

In summary, this dissertation investigates the advancement and application of the space-time finite element method, aiming to enhance both the efficiency and accuracy of solving complex differential equations. It begins by tackling the challenge of decoupling systems of equations, which promotes parallel computation and improves scalability. The space-time finite element method, leveraging simplex-shaped functions, is presented as a more effective solution for hyperbolic, parabolic, and elliptic differential equations. The method is applied to a one-dimensional rod model, demonstrating its ability to solve governing equations and compare outcomes with traditional

approaches. The research further extends into large deformation issues under plane-stress conditions, where the necessity of incorporating nonlinear terms for accurate simulations of significant displacements is highlighted. The method's versatility is validated through the development of a smart elastic material and a viscoplastic model. Parallel computing is employed to optimize the performance of the space-time finite element method, with exhaustive comparisons made for both computational efficiency and numerical results across various examples. The findings underline the method's robustness in managing complex simulations and its superiority over classical methods in terms of both accuracy and computational speed.

Chapter 8

Conclusions and future recommendation

This dissertation presents a comprehensive study on the development and application of the space-time finite element method for solving complex differential equations and advancing the simulation of dynamic structures. The proposed methodology introduces a novel approach to parallel computation, enabling the direct decoupling of systems of equations and efficient distribution of computations to parallel processing units. The resulting global matrices, taking a triangular form with non-zero coefficients, highlight the efficiency of this approach. Proper spatial discretization facilitates the separation of nonlinear zones that require iterative solutions, thereby reducing computational effort by allowing only small parts of the system to be recomputed iteratively. This method significantly enhances computational efficiency, especially in large-scale problems.

A key differentiator of the space-time element method from traditional approaches is its simultaneous discretization of spatial and temporal variables. Unlike the classical two-step interpolation method, which first transforms partial differential equations into ordinary differential equations in time, the space-time FEM integrates these variables, facilitating a more straightforward transition from static to dynamic solutions. This integration allows the use of effective tools for each stage and simplifies the application of numerical procedures for both static and dynamic analysis. The innovative approach of simultaneously discretizing spatial and temporal domains offers a unified framework that can handle complex boundary conditions and varying material properties more effectively than traditional methods.

One critical aspect highlighted in this dissertation is the decision to include the nonlinear term in finite element simulations, which depends on the magnitude of deformation imposed by the applied loading conditions. For small deformations, the analysis provides an efficient and accurate solution without the nonlinear term. However, for significant deformations resulting from larger forces, including the nonlinear term is essential to ensure simulation accuracy, despite the increased computational complexity.

This highlights the importance of adaptive algorithms that can selectively incorporate nonlinear effects based on the deformation magnitude, thus optimizing both accuracy and computational resources.

In demonstrating practical applications, the dissertation presented two examples in Chapter 6. The first example focused on developing a smart elastic material with variable properties in response to mechanical wave propagation. By using a Heaviside function to simulate the wave and comparing semi-analytical solutions with finite model results, the research showed that the material's locally weakening stiffness could effectively reduce peak accelerations and forces by 30% to 70%. This finding has significant implications for the development of protective materials. The ability to dynamically alter material properties in response to external stimuli can lead to the design of advanced materials for impact mitigation in various fields, such as automotive safety, sports equipment, and aerospace engineering. Future research should correlate the mathematical model with real materials, explore multilayer materials with different mechanical functions, and verify impact energy absorption in two- and three-dimensional systems.

The second example concerned the plane stress problem in which the structure is subjected to an impulse load. A viscoplastic material model was employed for the computer simulations. The publicly accessible program "triangle" was utilized to create the finite element mesh. Based on the resulting triangular division, a simplex-shaped space-time mesh was constructed. A computer program was then developed to simulate the dynamic behavior of the problem at hand. Utilizing the space-time model outlined in subsection 4.2.3, numerical calculations of the metamaterial structure were carried out, and the results from the computer simulation were presented.

Overall, this dissertation underscores the effectiveness of the space-time finite element method in addressing complex dynamic problems. The integration of spatial and temporal discretization, combined with parallel computing, offers a powerful tool for engineers and researchers. The significant reduction in computational time and enhanced accuracy of simulations pave the way for practical applications across various fields, including material science, biomechanics, and structural engineering.

Based on the findings and limitations identified in this research, several recommendations for future work are proposed:

1. Extension to Nonlinear Problems Expanding the methodology to handle a broader range of nonlinear problems will enhance its applicability, allowing for more comprehensive solutions to practical engineering challenges involving complex material behaviors and large deformations.

2. Optimization of Parallel Algorithms Further optimization of parallel computation algorithms, including exploring advanced parallelization techniques and hardware acceleration such as GPUs, can lead to significant performance improvements.

3. Application to Multiphysics Problems Integrating the space-time finite element method with multiphysics simulations can provide a comprehensive framework

for solving coupled problems involving multiple physical phenomena, expanding its utility in fields like fluid-structure interaction and thermal analysis.

4. Development of Adaptive Refinement Techniques Implementing adaptive refinement techniques within the continuous mesh adaptation framework can further improve accuracy and computational efficiency, especially for problems with localized features or singularities.

5. Experimental Validation Conducting experimental studies to validate the computational results will strengthen the credibility of the proposed methodologies, providing practical insights and identifying areas for further improvement.

In conclusion, this dissertation presents a novel and effective approach to solving complex differential equations through the space-time finite element method and parallel computation. The research offers significant contributions to computational mechanics, providing a solid foundation for future advancements. By addressing the proposed future work, the method's potential can be further realized, leading to even greater impacts in the field of engineering and beyond.

List of symbols, abbreviations, and acronyms

FEM	Finite Element Methods
STFEM	Space-time Finite Element Methods
GPU	Graphics Processing Unit.
CPU	Central Processing Unit
CUDA	Compute Unified Device Architecture
GPGPU	General-Purpose computing on Graphics Processing Units
PinT	Parallel-in-Time
PITA	Parallel Implicit Time-Integration Algorithms
HPC	High-Performance Computing
API	Application Programming Interface
PETSc	Portable, Extensible Toolkit for Scientific Computation

General notations

x	coordinate
y	coordinate
t	time
$\cdot\,\dot{}$	derivatives with respect to time
∇	gradient operator

div	divergence operator
$\mathcal{D}$	differential operator
t_i	time in i step
h	time step
$\mathbf{N}$	shape function matrix
$\mathbf{N}^*$	virtual shape function matrix
L	length
A	cross-sectional area
E	Young's modulus
ν	Poisson's ratio
ρ	density of the material
$\boldsymbol{B}$	linear strain matrix
$\boldsymbol{C}$	velocity matrix
$\boldsymbol{K}$	linear stiffness matrix
$\boldsymbol{M}$	inertia matrix
$\boldsymbol{K}^N$	nonlinear stiffness matrix
$\boldsymbol{D}$	material properties matrix
$H(x)$	Heaviside step function
$\delta(x)$	Dirac delta function
c	wave propagation speed in rod

Chapter 3

$\overline{V}$	closure of the domain V, a subdomain in E^3
V	interior of the domain $\overline{V}$
∂V	boundary of the domain V
$\mathbf{u}$	displacement vector
$\mathbf{v}$	velocity vector
$\mathbf{f}$	body force vector (inertial forces)
$\boldsymbol{\sigma}$	symmetric Cauchy stress tensor
$\boldsymbol{\varepsilon}$	symmetric strain tensor
$\widehat{\mathbf{t}}$	surface traction vector (force on boundary)
$\mathbf{x}$	spatial coordinates in E^3
$\delta\mathbf{u}$	virtual displacement vector
$\mathbf{q}$	nodal displacement vector
$\tilde{\mathbf{K}}_e$	element space-time stiffness matrix
$\mathbf{K}_e$	element stiffness matrix
$\mathbf{M}_e$	element inertia matrix
$\mathbf{W}_e$	internal damping matrix
$\mathbf{Z}_e$	external damping matrix
$\mathbf{E}$	elasticity matrix
$\mathbf{R}$	inertia matrix
η_w	internal viscous damping coefficient
η_z	external damping coefficient
$\mathbf{Q}_e$	vector of external forces
$\mathbf{q}_0$	initial displacement vector
$\dot{\mathbf{q}}_0$	initial velocity vector

$\mathbf{A}_i$	submatrix of the space-time stiffness matrix (upper left)
$\mathbf{B}_i$	submatrix of the space-time stiffness matrix (upper right)
$\mathbf{C}_i$	submatrix of the space-time stiffness matrix (lower left)
$\mathbf{D}_i$	submatrix of the space-time stiffness matrix (lower right)

Chapter 4

u	displacement in the x-direction
v	displacement in the y-direction
f	external body force per unit length
σ	stress in the rod
ϵ	strain in the rod
F	internal force
q_i	nodal displacement of i-th node
$\boldsymbol{G}$	matrix of the coordinates
b	length of the 1d element
S_e	area of the triangle
P	axial force
σ_x	normal stress in the x-direction
σ_y	normal stress in the y-direction
τ_{xy}	shear stress in the xy-plane
f_x	body force per unit volume in the x-direction
f_y	body force per unit volume in the y-direction
ϵ_x	longitudinal strain in the x-direction
ϵ_y	longitudinal strain in the y-direction
γ_{xy}	shear strain in the xy-plane

V_e	volume of the tetrahedron
$\boldsymbol{K}_i$	stiffness matrix for the i-th tetrahedral element
$\boldsymbol{M}_i$	inertia matrix for the i-th tetrahedral element
$\boldsymbol{K}_i^N$	nonlinear stiffness matrix for the i-th tetrahedral element

Chapter 6

w	longitudinal displacement in rod
$\hat{E}$	Young's modulus of the moving part
v_0	initial velocity
ϵ	width of the traveling stiffness zone
N	longitudinal force
$V_j(t)$	Fourier sine integral transformation of function $w(x,t)$
$\mathbb{M}$	inertial matrix in semi-analytical solution
$\mathbb{K}$	stifness matrix in semi-analytical solution
$\hat{\Pi}$	virtual energy
K	material viscosity (Norton-Hoff model)
m	deformation rate sensitivity (Norton-Hoff model)
$\boldsymbol{\tau}'$	Cauchy stress tensor
D	equivalent strain rate
$\dot{\boldsymbol{\varepsilon}}$	strain rate vector
s	multiplier
λ	coefficient of the penalty function

List of Tables

List of Figures

Bibliography

[1] A. Abas and R. Abdul-Rahman. Adaptive fem with domain decomposition method for partitioned-based fluid–structure interaction. *Arabian Journal for Science and Engineering*, 41(2):611–622, 2016.

[2] H. Adeli. High-performance computing for large-scale analysis, optimization, and control. *Journal of Aerospace Engineering*, 13(1):1–10, 2000.

[3] J.M. Alonso, C. de Alfonso, G. García, and V. Hernández. Grid technology for structural analysis. *Advances in Engineering Software*, 38(11):738–749, 2007. Engineering Computational Technology.

[4] M. Anderson and J.-H. Kimn. A numerical approach to space–time finite elements for the wave equation. *J. Comput. Phys.*, 226:466–476, 2007.

[5] J. H. Argyris and A. S. L. Chan. Application of the finite elements in space and time. *Ing. Archiv*, 41:235–257, 1972.

[6] J. H. Argyris and D. W. Scharpf. Finite elements in space and time. *Nucl. Engng Design*, 10:456–469, 1969.

[7] J. H. Argyris and D. W. Scharpf. Finite elements in time and space. *Aeron. J. Roy. Aeron. Soc.*, 73:1041–1044, 1969.

[8] S. Bahl, H. Nagar, I. Singh, and S. Sehgal. Smart materials types, properties and applications: A review. *Materials Today: Proceedings*, 28:1302–1306, 2020.

[9] C. Bajer. Space–time finite element formulation for the dynamical evolutionary process. *Appl. Math. and Comp. Sci.*, 3(2):251–268, 1993.

[10] C. I. Bajer. Triangular and tetrahedral space–time finite elements in vibration analysis. *International Journal for Numerical Methods in Engineering*, 23:2031–2048, 1986.

[11] C. I. Bajer. Dynamics of contact problem by the adaptive simplex–shaped space–time approximation. *J. Theor. Appl. Mech.*, 7:235–248, 1988. Special issue, suplement No. 1.

[12] C. I. Bajer. Adaptive mesh in dynamic problem by the space–time approach. *Computers and Structures*, 33(2):319–325, 1989.

[13] C. I. Bajer and C. Bohatier. The soft way method and the velocity formulation. *Computers and Structures*, 55(6):1015–1025, 1995.

[14] C. I. Bajer and B. Dyniewicz. Space–time approach to numerical analysis of a string with a moving mass. *Int. J. Numer. Meth. Engng.*, 76(10):1528–1543, 2008.

[15] C. I. Bajer and B. Dyniewicz. Numerical modelling of structure vibrations under inertial moving load. *Arch. Appl. Mech.*, 79(6-7):499–508, 2009.

[16] C. I. Bajer and B. Dyniewicz. Virtual functions of the space-time finite element method in moving mass problems. *Computers and Structures*, 87:444–455, 2009.

[17] C.I. Bajer and B. Dyniewicz. *Numerical analysis of vibrations of structures under moving inertial load*. Springer, 2012.

[18] R.E. Bank and R.F. Santos. Analysis of some moving space-time finite element methods. *SIAM Journal on Numerical Analysis*, 30(1):1–18, 1993.

[19] E. Barchiesi, M. Spagnuolo, and L. Placidi. Mechanical metamaterials: a state of the art. *Mathematics and Mechanics of Solids*, 24(1):212–234, 2019.

[20] M. Behr. Simplex space–time meshes in finite element simulations. *Int. J. Numer. Meth. Fluids*, 57(9):1421–1434, 2008.

[21] M. Behr. Simplex space–time meshes in finite element simulations. *International Journal for Numerical Methods in Fluids*, 57(9):1421–1434, 2008.

[22] K. Bertoldi, V. Vitelli, J. Christensen, and M. Van Hecke. Flexible mechanical metamaterials. *Nature Reviews Materials*, 2(11):1–11, 2017.

[23] J. Blazek. *Computational fluid dynamics: principles and applications*. Elsevier, 2015.

[24] A. Boglietti, A. Cavagnino, D. Staton, M. Shanel, M. Mueller, and C. Mejuto. Evolution and modern approaches for thermal analysis of electrical machines. *IEEE Transactions on Industrial Electronics*, 56(3):871–882, 2009.

[25] R. Bonnerot and P. Jamet. A second order finite element method for the one–dimensional Stefan problem. *International Journal for Numerical Methods in Engineering*, 8:811–820, 1974.

[26] R. Bonnerot and P. Jamet. Numerical computation of the free boundary for the two–dimensional Stefan problem by space–time finite elements. *J. Comput. Phys.*, 25:163–181, 1977.

[27] A. Bossavit. *Computational electromagnetism: variational formulations, complementarity, edge elements.* Academic Press, San Diego, 1998.

[28] M.V.K. Chari and S.J. Salon. *Numerical Methods in Electromagnetism.* Elsevier, 2000.

[29] J. Cortial and C. Farhat. A time-parallel implicit method for accelerating the solution of non-linear structural dynamics problems. *International Journal for Numerical Methods in Engineering*, 77(4):451–470, 2009.

[30] D. B. Davidson. *Computational electromagnetics for RF and microwave engineering.* Cambridge University Press, 2005.

[31] B. Dyniewicz. Space–time finite element approach to general description of a moving inertial load. *Finite Elem. Anal. and Des.*, 62:8–17, 2012.

[32] B. Dyniewicz. Efficient numerical approach to unbounded systems subjected to a moving load. *Computational Mechanics*, 54(2):321–329, 2014.

[33] B. Dyniewicz, C. I. Bajer, and J. Matej. Mass splitting of train wheels in the numerical analysis of high speed train–track interactions. *Veh. Syst. Dyn.*, 53(1):51–67, 2015.

[34] B. Dyniewicz, J. M. Bajkowski, and C. I. Bajer. Smart control in vibrations of structures. In *Proceedings of the 6th World Conference on Structural Control and Monitoring*, pages 15–17, 2014.

[35] C. Farhat, J. Cortial, C. Dastillung, and H. Bavestrello. Time-parallel implicit integrators for the near-real-time prediction of linear structural dynamic responses. *International Journal for Numerical Methods in Engineering*, 67(5):697–724, 2006.

[36] C. Farhat and J. Li. An iterative domain decomposition method for the solution of a class of indefinite problems in computational structural dynamics. *Applied Numerical Mathematics*, 54(2):150–166, 2005. 6th IMACS.

[37] F.I. Fazanaro, D.C. Soriano, R. Suyama, M.K. Madrid, J.R. de Oliveira, I.B. Muñoz, and R. Attux. Numerical characterization of nonlinear dynamical systems using parallel computing: The role of gpus approach. *Communications in Nonlinear Science and Numerical Simulation*, 37:143–162, 2016.

[38] J. H. Ferziger and M. Peric. *Computational methods for fluid dynamics*. Springer Berlin, Heidelberg, 2002.

[39] B. Florijn, C. Coulais, and M. van Hecke. Programmable mechanical metamaterials. *Phys. Rev. Lett.*, 113:175503, Oct 2014.

[40] D. French and T. Peterson. A continuous space–time finite element method for the wave equation. *Math. Comput.*, 65:491–506, 1996.

[41] I. Fried. Finite element analysis of time–dependent phenomena. *AIAA J.*, 7:1170–1173, 1989.

[42] S. V. Garimella, A. S. Fleischer, J. Y. Murthy, A. Keshavarzi, R. Prasher, C. Patel, S. H. Bhavnani, R. Venkatasubramanian, R. Mahajan, Y. Joshi, B. Sammakia, B. A. Myers, L. Chorosinski, M. Baelmans, P. Sathyamurthy, and P. E. Raad. Thermal challenges in next-generation electronic systems. *IEEE Transactions on Components and Packaging Technologies*, 31(4):801–815, 2008.

[43] P. Gosselet and C. Rey. Non-overlapping domain decomposition methods in structural mechanics. *Archives of Computational Methods in Engineering*, 13(4):515, 2006.

[44] A. Gravouil and A. Combescure. Multi-time-step and two-scale domain decomposition method for non-linear structural dynamics. *International Journal for Numerical Methods in Engineering*, 58(10):1545–1569, 2003.

[45] X. Guo, J. Zeng, H. Ma, S. Liu, Z. Luo, C. Li, Q. Han, and B. Wen. Semi-analytical modeling and experimental verification of a flexible varying section disk-blades system with elastic supports. *Thin-Walled Struct.*, 185(110563):110563, 2023.

[46] M. E. Gurtin. Variational principles for linear elastodynamics. *Arch. Rat. Mech. Anal.*, 16:34–50, 1964.

[47] M. E. Gurtin. Variational principles for linear initial–value problems. *Quart. Appl. Math.*, 22:252–256, 1964.

[48] J.F. Hajjar and J.F. Abel. Parallel processing for transient nonlinear structural dynamics of three-dimensional framed structures using domain decomposition. *Computers and Structures*, 30(6):1237–1254, 1988.

[49] P. Hansbo and A. Szepessy. A velocity–pressure streamline diffusion finite element method for the incompressible Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 84:175–192, 1990.

[50] S. W. Hansen. Several related models for multilayer sandwich plates. *Mathematical Models and Methods in Applied Sciences*, 14(8):1103–1132, 2004.

[51] I. Herrera and J. Bielak. A simplified version of Gurtin's variational principles. *Arch. Rat. Mech. Anal.*, 53:131–149, 1974.

[52] T. J. R. Hughes and G. M. Hulbert. Space–time element methods for elastodynamics: formulations and error estimates. *Comput. Meth. Appl. Mech. Engng*, 66:339–363, 1988.

[53] G. M. Hulbert. *Space–time finite element method for second–order hyperbolic equation.* PhD thesis, Stanford University, CA, August 1989.

[54] A. Idesman, R. Niekamp, and E. Stein. Finite elements in space and time for generalized viscoelastic Maxwell model. *Computational Mechanics*, 27:49–60, 2001.

[55] H.Z. Jahromi, B.A. Izzuddin, and L. Zdravkovic. A domain decomposition approach for coupled modelling of nonlinear soil–structure interaction. *Computer Methods in Applied Mechanics and Engineering*, 198(33):2738–2749, 2009.

[56] N. Jain, S.N. Ovhal, V. Patil, and K.N. Kartik. Smart materials–a state-of-the-art-review. *Materials Today: Proceedings*, 2023.

[57] B. Jiang. *The Least-Squares Finite Element Method. Theory and Applications in Computational Fluid Dynamics and Electromagnetics.* Springer-Verlag Berlin Heidelberg, 1998.

[58] Z. Kacprzyk. General formulation of the stiffness matrix for the space–time finite elements. *MATEC Web Conf.*, 117:1–8, 2017.

[59] D.-K. Kang, C.-W. Kim, and H.-I. Yang. GPU-based parallel computation for structural dynamic response analysis with cuda. *Journal of Mechanical Science and Technology*, 28(10):4155–4162, 2014.

[60] G. Karniadakis and S. Sherwin. *Spectral/hp Element Methods for Computational Fluid Dynamics.* Oxford University Press, 2005.

[61] R. Kicinger, T. Arciszewski, and K. DeJong. Evolutionary computation and structural design: A survey of the state-of-the-art. *Comput. Struct.*, 83(23–24):1943–1978, 2005.

[62] R.F Kulak, P.A Pfeiffer, and E.J Plaskacz. Modeling of containment structures on high performance computers. *Nuclear Engineering and Design*, 174(2):143–156, 1997.

[63] J-Y Kwak, T-Y Chun, S-J Shin, and O.A. Bauchau. Domain decomposition approach to flexible multibody dynamics simulation. *Computational Mechanics*, 53(1):147–158, 2014.

[64] Z. Kączkowski. The method of finite space–time elements in dynamics of structures. *Journal of Technical Physics*, 16(1):69–84, 1975.

[65] Z. Kączkowski. The method of time-dependent finite elements. *Archives of Civil Engineering.*, 22(3):365–378, 1976.

[66] Z. Kączkowski. General formulation of the stiffness matrix for the space–time finite elements. *Archives of Civil Engineering.*, 25(3):351–357, 1979.

[67] Z. Kączkowski. On application of non-rectangular space-time elements (in Polish). *J. Theoret. and Appl. Mech.*, 21(4):531–542, 1983.

[68] Z. Kączkowski and J. Langer. Synthesis of the space-time finite element method. *Archives of Civil Engineering.*, 26(1):11–17, 1980.

[69] Z. Kączkowski and M. Witkowski. Accounting for external damping in the method of finite space–time elements (in Polish). *Archives of Civil Engineering.*, 23(3):243–254, 1977.

[70] Z. Kączkowski and M. Żyszko. Flexural vibrations of bars by the method of time–space finite elements (in Polish). *Archives of Civil Engineering.*, 24(1):67–78, 1978.

[71] G. La Fauci, M. Parisi, A. Nanni, L. Crosetta, N.M. Pugno, and M. Colonna. Design and proof-of-concept of an advanced protective system for the dissipation of tangential impact energy in helmets, based on non-newtonian fluids. *Smart Mater. Struct.*, 32(4):044004, April 2023.

[72] P. L'Eplattenier, G. Cook, C. Ashcraft, M. Burger, J. Imbert, and M. Worswick. Introduction of an electromagnetism module in ls-dyna for coupled mechanical-thermal-electromagnetic simulations. *steel research international*, 80(5):351–358, 2009.

[73] F. Marignetti, V. Delli Colli, and Y. Coia. Design of axial flux pm synchronous machines through 3-d coupled electromagnetic thermal and fluid-dynamical finite-element analysis. *IEEE Transactions on Industrial Electronics*, 55(10):3591–3601, 2008.

[74] B. R. Mishra, N. J. Varghese, and K. Sasihithlu. Semi-analytical technique for the design of disordered coatings with tailored optical properties. *Optics Express*, 2023.

[75] S.E. Moore. A stable space–time finite element method for parabolic evolution problems. *Calcolo*, 55(18), 2018.

[76] M. Morandi Cecchi and A. Scarpa. A space–time finite element method for nonlinear parabolic problems. *Applied Numerical Mathematics*, 15(2):247–258, 1994.

[77] B Murali and J Nagarani. Design and fabrication of construction helmet by using hybrid composite material. In *2013 International Conference on Energy Efficient Technologies for Sustainability*. IEEE, April 2013.

[78] B Murali and J Nagarani. Design and fabrication of construction helmet by using hybrid composite material. In *2013 International Conference on Energy Efficient Technologies for Sustainability*, pages 145–147. IEEE, 2013.

[79] D. Negrut, R. Serban, H. Mazhar, and T. Heyn. Parallel Computing in Multibody System Dynamics: Why, When, and How. *Journal of Computational and Nonlinear Dynamics*, 9(4):041007, 07 2014.

[80] Z. G Nicolaou and A. Motter. Mechanical metamaterials with negative compressibility transitions. *Nature materials*, 11(7):608–613, 2012.

[81] A.K. Noor. New computing systems and future high-performance computing environment and their impact on structural analysis and design. *Computers & Structures*, 64(1):1–30, 1997. Computational Structures Technology.

[82] F. H. Norton. *The Creep of Steel at High Temperatures*. McGraw-Hill Company, London, 1929.

[83] J. T. Oden. A generalized theory of finite elements, II. Applications. *International Journal for Numerical Methods in Engineering*, 1:247–259, 1969.

[84] Z.-H. Qian, L. Wang, D.-A. Sun, and J.-F. Zou. A semianalytical model for three-dimensional stability analysis of potentially rotational slopes in unsaturated soils. *Int. J. Geomech.*, 22(10), 2022.

[85] A Rama Mohan Rao, T.V.S.R Appa Rao, and B Dattaguru. A new parallel overlapped domain decomposition method for nonlinear dynamic finite element analysis. *Computers and Structures*, 81(26):2441–2454, 2003.

[86] V. K. Raman. An introduction to high-performance scientific computing. *Journal of Chemical Information and Computer Sciences*, 37(5):964–964, 1997.

[87] M.V. Rodrigues, F.N. Correa, and B.P. Jacob. Implicit domain decomposition methods for coupled analysis of offshore platforms. *Communications in Numerical Methods in Engineering*, 23(6):599–621, 2007.

[88] S. Russenschuck. *Field Computation for Accelerator Magnets: Analytical and Numerical Methods for Electromagnetic Design and Optimization*. Wiley-VCH Verlag GmbH & Co. KGaA, 2010.

[89] S.D. Salman. Kenaf-ramid fiber-reinforced polyvinyl butyral hybrid composites for military helmet. *Faculty of Engineering; Universiti Putra Malaysia: Putra, Malaysia*, page 136, 2017.

[90] F Salvemini, F Grazzi, A Fedrigo, A Williams, F Civita, A Scherillo, P Vontobel, S Hartmann, E Lehmann, and M Zoppi. Revealing the secrets of composite helmets of ancient japanese tradition. *The European Physical Journal Plus*, 128:1–10, 2013.

[91] A. Schafelner. *Space-time Finite Element Methods*. PhD thesis, Johannes Kepler University Linz, 2021.

[92] L. Sevgi. *Complex electromagnetic problems and numerical simulation approaches*. Wiley, 2003.

[93] E. D. Sotelino. Parallel processing techniques in structural engineering applications. *Journal of Structural Engineering*, 129(12):1698–1706, 2003.

[94] G. Stavroulakis, D.G. Giovanis, V. Papadopoulos, and M. Papadrakakis. A gpu domain decomposition solution for spectral stochastic finite element method. *Computer Methods in Applied Mechanics and Engineering*, 327:392–410, 2017. Advances in Computational Mechanics and Scientific Computation—the Cutting Edge.

[95] O. Steinbach. Space-time finite element methods for parabolic problems. *Computational Methods in Applied Mathematics*, 15(4):551–566, 2015.

[96] O. Storaasli, J. Ransom, and R. Fulton. Structural dynamic analysis on a parallel computer: The finite element machine. *Computers and Structures*, 26(4):551–559, 1987.

[97] T.E. Tezduyar, M. Behr, and J. Liou. A new strategy for finite element computations involving moving boundaries and interfaces—the deforming-spatial-domain/space-time procedure: I. the concept and the preliminary numerical tests. *Computer Methods in Applied Mechanics and Engineering*, 94(3):339–351, 1992.

[98] L. Thompson and P. Pinsky. A space–time finite element method for the exterior structural acoustics problem: time–dependent radiation boundary conditions in two space dimensions. *Int. J. Numer. Meth. Engng.*, 39:1635–1657, 1996.

[99] E. A. Thornton. *Thermal Structures for Aerospace Applications*. American Institute of Aeronautics and Astronautics, Inc., 1996.

[100] E. F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Springer Berlin, Heidelberg, 2009.

[101] J. Tu, G.-H. Yeoh, and C. Liu. *Computational fluid dynamics: a practical approach*. Elsevier, 2019.

[102] Q. Wang. *Practical design of magnetostatic structure using numerical simulation*. Wiley, 2013.

[103] G. Yagawa, N. Soneda, and S. Yoshimura. A large scale finite element analysis using domain decomposition method on a parallel computer. *Computers and Structures*, 38(5):615–625, 1991.

[104] V. Yaghoubi, M.K. Vakilzadeh, and T. Abrahamsson. A parallel solution method for structural dynamic response analysis. In Matt Allen, Randall L. Mayes, and Daniel J. Rixen, editors, *Dynamics of Coupled Structures, Volume 4*, pages 149–161, Cham, 2015. Springer International Publishing.

[105] X. Yu, J. Zhou, H. Liang, Z. Jiang, and L. Wu. Mechanical metamaterials associated with stiffness, rigidity and compressibility: A brief review. *Progress in Materials Science*, 94:114–173, 2018.

[106] Y. Zhang, J. Ruan, T. Huang, X. Yang, H. Zhu, and G. Yang. Calculation of temperature rise in air-cooled induction motors through 3-d coupled electromagnetic fluid-dynamical and thermal finite-element analysis. *IEEE Transactions on Magnetics*, 48(2):1047–1050, 2012.

[107] D. Zhao, B. Dyniewicz, and C.I. Bajer. Smart elastic material with a moving local stiffness zone for reducing the effects of impact loading. *Shock and Vibration*, 2023:1–10, 2023.

[108] O. C. Zienkiewicz. *The finite element method*. McGraw-Hill, London, 1977.