Univerza
v Ljubljani

Fakulteta za
gradbeništvo in
geodezijo

Jamova *cesta* 2
1000 Ljubljana,Slovenija
telefon (01) 47 68 500 faks
(01) 42 50 681
tajnistvo@fgg.uni-lj.si

DOKTORSKI ŠTUDIJSKI PROGRAM III. STOPNJE GRAJENO OKOLJE

Doktorand/-ka:
NINA ZUPAN

MODELIRANJE METAMATERIALOV S POENOTENO VEČNIVOJSKO METODO

MODELLING OF METAMATERIALS WITH UNIFIED MULTI-SCALE METHOD

2020

Univerza
v Ljubljani

Fakulteta
za gradbeništvo
in geodezijo

Mentor/-ica: prof. dr. Jože Korelc

Somentor/-ica:

Poročevalci za oceno doktorske disertacije:

- izr. prof. dr. Marko Kegl
- prof. dr. Peter Wriggers
- prof. dr. Boštjan Brank

ERRATA

Page	Line	Error	Correction

BIBLIOGRAPHIC-DOCUMENTALISTIC INFORMATION

UDC:	531/533(043)
Author:	Nina Zupan
Supervisor:	Prof. Jože Korelc, Ph.D.
Title:	Modelling of Metamaterials with Unified Multi-scale Method
Notes:	138 p., 52 fig., 12 tab., 226 eq.
Keywords:	two-level path-following, multi-scale methods, sensitivity analysis, MIEL (mesh-in-element), FE^2, metamaterial, multi-scale optimization

Abstract:

The doctoral thesis presents a generalized essential boundary condition sensitivity analysis based implementation of FE^2 and mesh-in-element (MIEL) multi-scale methods and the application of these methods in multi-scale optimization algorithms. The implementation is derived as an alternative to standard implementations of multi-scale analysis, where the calculation of the SCHUR complement of the microscopic tangent matrix is needed to bridge different scales. The thesis presents a unified approach to the development of an arbitrary MIEL or FE^2 computational scheme for an arbitrary path-dependent material model. Implementation is based on efficient first and second order analytical sensitivity analyses, for which an automatic-differentiation-based formulation (ADB) of essential boundary condition sensitivity analysis is derived. A fully consistently linearized two-level path-following algorithm is introduced as a solution algorithm for multi-scale modeling. Sensitivity analysis allows each macro step to be followed by an arbitrary number of intermediate micro steps while retaining quadratic convergence of the overall solution algorithm.

The implementation of multi-scale optimization algorithms is described, where a gradient-based optimization algorithm was wrapped around a multi-scale solution procedure. The versatility of sensitivity analysis for connecting scales and optimization purposes was proven. Examples using a developed optimization algorithm are presented. Through optimal distribution and opening size across the domain, fascinating mechanical properties can be achieved taking into account different optimization criteria. This process was used to design metamaterials for optimal energy dissipation.

BIBILIOGRAFSKO-DOKUMENTACIJSKA STRAN IN IZVLEČEK

UDK:	531/533(043)
Avtor:	Nina Zupan
Mentor:	prof. dr. Jože Korelc
Naslov:	Modeliranje metamaterialov s poenoteno večnivojsko metodo
Obseg in oprema:	138 str., 52 sl., 12 pregl., 226 en.
Ključne besede:	dvonivojsko sledenje poti, večnivojske metode, občutljivostna analiza, MIEL (mreža v elementu), FE^2, metamaterial, večnivojska optimizacija

Izvleček:

V doktorski disertaciji je predstavljen posplošen pristop implementacije FE^2 in MIEL (mreža v elementu) večnivojskih metod preko občutljivostne analize bistvenih robnih pogojev in aplikacija teh dveh metod v večnivojskem optimizacijskem algoritmu. Implementacija na osnovi občutljivostne analize je izpeljana kot alternativa standardni implementaciji večnivojske analize, pri kateri se za povezavo med nivojema uporabi izračun Schurovega komplementa tangentne matrike mikro nivoja. V disertaciji je predstavljen poenoten pristop k razvoju poljubne MIEL ali FE^2 računske sheme za poljuben materialni model odvisen od poti. Implementacija je osnovana na učinkoviti analitični občutljivostni analizi prvega in drugega reda, za katero je za občutljivostno analizo bistvenih robnih pogojev izpeljana formulacija na osnovi uporabe avtomatskega odvajanja (ADB). Za algoritem reševanja večnivojskega modeliranja je vpeljana popolnoma konsistentna linearizacija dvonivojskga sledenja poti. Občutljivostna analiza omogoča, da lahko vsakemu makro koraku sledi poljubno število mikro podkorakov pri čemer se ohrani kvadratična konvergenca celotnega algoritma za reševanje.

Opisana je implementacija večnivojskega optimizacijskega algoritma, kjer je gradientna optimizacija izvedena v zanki okoli večnivojske procedure reševanja. Prikazana je uporabnost občutljivostne analize za različne namene, povezavo med nivojema in optimizacijo. Izpeljani optimizacijski algoritem je bil uporabljen na primerih. Pokazano je bilo, da se lahko z optimalno razporeditvijo in velikostjo odprtin po domeni doseže zanimive mehanske lastnosti z upoštevanjem optimizacijskih kriterijev. Na tak način je bil oblikovan metamaterial za optimalno disipacijo energije.

ZAHVALA

Najprej bi se rada zahvalila prof. dr. Jožetu Korelcu, ki mi je dal priložnost za raziskovalno delo in izdelavo doktorata pod njegovim mentorstvom. Hvala za vse nasvete, pomoč in ponujene možnosti za pridobivanje znanja in širjenje obzorij doma in v tujini. Iskreno se zahvaljujem tudi komisiji za oceno doktorata.

Hvala sodelavcem in prijateljem na Katedri za metalne konstrukcije in na FGG, za vse strokovne nasvete in marsikatero življenjsko lekcijo. Zaradi skupnih doživetij na fakulteti kakor tudi zunaj nje mi bo doktorski študij za vedno ostal v najlepšem spominu.

Hvala prijateljem z vseh vetrov, ki so bili najboljša družba in moja opora v najtežjih trenutkih. Vedeli so kaj potrebujem tudi takrat, ko sama nisem vedela.

Na koncu še posebna zahvala moji družini, za vso ljubezen in podporo.

TABLE OF CONTENTS

INDEX OF FIGURES

KAZALO SLIK

INDEX OF TABLES

KAZALO PREGLEDNIC

NOTATION

$\square_M$	macro level quantity
$\square_m$	micro level quantity
$\square_k, \square_{k+1}$	index of the last and the current macro step
$\square_n, \square_{n+1}$	index of the last and the current micro step
$\square_s$	index of micro step at the and of the last macro step
$\square^{(r)}$	index of micro problem
$\boldsymbol{\phi}$	set of variables calculated at the selected macro element and transferred to the selected micro problem (also sensitivity parameters)
$\mathbf{S}$	set of variables calculated at the selected micro problem and returned to the selected macro element
λ_M, λ_m	macro and micro level parameter
$\boldsymbol{\phi}_{Me}, \mathbf{S}_{Me}$	variables $\boldsymbol{\phi}$ and $\mathbf{S}$ collected for all micro problems associated with the selected macro element
$\mathbf{R}_M, \mathbf{K}_M$	macro level residual and tangent matrix
$\mathbf{R}_m, \mathbf{K}_m$	micro level residual and tangent matrix
$\mathbf{p}_M \equiv \mathbf{p}_{M\,k+1}$	macro level nodal unknowns
$\mathbf{p}_m \equiv \mathbf{p}_{m\,n+1}$	micro level nodal unknowns
$\bar{\mathbf{p}}_m = \bar{\mathbf{p}}_{m\,n+1}$	micro level nodal unknowns with prescribed essential boundary condition
$\square_e$	quantities associated with the selected macro or micro element (e.g. $\mathbf{R}_{Me}, \mathbf{R}_{me}, \mathbf{p}_{me}, \dots$)
$\mathbf{h}_g \equiv \mathbf{h}_{g\,n+1}$	set of selected micro level integration point unknowns
$\mathbf{h}_m \equiv \mathbf{h}_{m\,n+1}$	set of unknowns of all micro level integration points
$\mathbf{Q}_g, \mathbf{K}_Q$	set of integration point equations at micro level and corresponding tangent matrix
$\boldsymbol{F}$	deformation gradient
$\boldsymbol{U}$	right stretch tensor
$\boldsymbol{V}$	left stretch tensor
$\boldsymbol{C}$	right Cauchy-Green tensor
$\boldsymbol{b}$	left Cauchy-Green tensor
$\boldsymbol{E}$	Green-Lagrange strain tensor
$\boldsymbol{e}$	Euler-Almansi strain tensor
$\boldsymbol{\varepsilon}$	small strain tensor

$\boldsymbol{P}$	first Piola-Kirchhof stress tensor
W	strain energy function
$\boldsymbol{\sigma}$	Cauchy stress tensor
$\mathbf{A} : \mathbf{B}$	inner product ($\mathbf{A} : \mathbf{B} = \sum_{ij} A_{ij} B_{ij}$, $\mathbf{A} : \frac{\partial \mathbf{B}}{\partial \mathbf{p}} = \sum_{ij} A_{ij} \frac{\partial B_{ij}}{\partial \mathbf{p}}$)
$\frac{\hat{\delta} f(\mathbf{a})}{\hat{\delta} \mathbf{a}}$	computational derivative (the result of automatic differentiation)

When the selected variables has no index it always refers to the current value of the selected variable, e.g. $\mathbf{p}_M \equiv \mathbf{p}_{Mk+1}$, $\mathbf{p}_m \equiv \mathbf{p}_{mn+1}$ etc.

LIST OF ABBREVIATIONS

AD	automatic differentiation
ADB	automatic differentiation based
AM	additive manufacturing
DOF	degree of freedom
EBC	essential boundary condition
FEM	finite element method
FE^2	standard two-level finite element homogenization
MIEL	mesh-in-element multi-scale method
MKL	math kernel library
NKS	Newton-Krylov-Schur method
NR	Newton-Raphson method
RVE	representative volume element
SLA	stereolithography
SLM	selective laser melting
EA	evolutionary algorithm
GA	genetic algorithm
EP	evolutionary programming
ES	evolutionary strategies
SIMP	solid isotropic material with penalization
ESO	evolutionary structural optimization
SQP	sequential quadratic programming

1 INTRODUCTION

Demand for materials that can actively respond to the applied loads has arisen due to technological progress. Consequently, new advanced numerical methods are needed to develop and investigate new materials. Construction design in civil engineering uses limit state design, which means that geometry and material nonlinearities need to be considered.

In the field of materials, metamaterials are a new trend. The design of the inner structure of metamaterials allows them to achieve properties that cannot be found in natural materials. In civil engineering, metamaterials are interesting in the fields of energy absorption, acoustic, thermal mechanics, and smart materials and construction. Additionally, new manufacturing techniques are gaining popularity. For example, additive manufacturing (also called 3D printing) is becoming increasingly attractive in civil engineering, where demand for unique components is high. Moreover, additive manufacturing has opened new possibilities in the field of metamaterials, allowing the production of complicated inner structures that previously could only be investigated analytically and numerically.

The requirement to accurately model heterogeneous or porous materials with complicated inner structure led to the development of multi-scale methods. With the growing capabilities of computers, the use of these methods is growing for detailed analysis of mechanical response with respect to material and geometric nonlinearities. The use of different kinds of multi-scale methods is limited by the specifications of the problem to be solved. Homogenization multi-scale methods, like the standard two-level finite element homogenization approach FE^2 [32], are appropriate when structures have distinct length scales that are only weakly coupled. If the difference between two scales is finite, or in the region of high gradients where homogenization methods fail, domain decomposition methods can be used, such as the mesh-in-element (MIEL) method [39]. For products made with 3D-printing, microstructure is determined by the precision of the 3D printer, e.g. the thickness of the material layer. The properties of metamaterials are conditioned with their geometrical inner structure. For both 3D-printing and metamaterials, scale separation is sometimes inappropriate, and the MIEL method is a fitting solution.

Multi-scale problems are usually solved implicitly, and linearization is necessary to construct an iterative solution scheme. For optimal and accurate linearization, an implementation based on sensitivity analysis is proposed as an alternative to the more commonly used Schur complement based approach. For linearization of the FE^2 scheme first order essential boundary conditions sensitivity analysis is required, while second order is

required for linearization of the MIEL scheme. Within the standard implementation of multi-scale methods, only the macro scale is parametrized, and a path-following algorithm is applied only at the global level. For the sensitivity-based implementation, a consistently linearized two-level path-following algorithm can be derived. Furthermore, a unified approach to multi-scale modeling is possible, which enables the use of FE^2, MIEL and single-scale methods in one model, resulting in optimal domain discretization. Formulation based on automatic differentiation also enables unification and automation of different multi-scale modeling approaches for nonlinear, time-dependent, coupled problems, e.g., finite strain plasticity.

Because of limited material resources, environmental effects, and competition, numerical optimization is becoming an important tool in engineering. It enables the improvement of products with respect to a given criterion. For example, in civil engineering, we want to minimize weight for required load-bearing capacity or determine optimal geometry for maximal energy dissipation in case of an earthquake. The optimization of large multi-scale problems requires the use of gradient-based optimization algorithms. Gradient-based multi-scale optimization combines multi-scale algorithms and gradient-based optimization algorithms. Evaluation of the gradient of the objective function requires sensitivity analysis of both the shape and the essential boundary conditions. This algorithm could be applied in the design of metamaterials, for the analysis of shape and the dimensions of the inner structure geometry.

To be able to derive necessary equations and computer codes, sophisticated tools like *AceGen* [30] and *AceFEM* [28] for the development of algorithms and numerical analysis were used. *AceGen* is an advanced automatic code generator, where automatic differentiation technique and automatic code optimization and generation are combined with the computer algebra system Mathematica. The *AceFEM* package is a general finite element environment designed to solve multi-physics and multi-field problems.

1.1 Background of work and state of the art

1.1.1 Multi-scale methods

Nowadays, multi-scale methods are widespread in computational mechanics, and their use is increasing with the ever-increasing capabilities of computers. These methods originate from the demand to accurately model heterogeneous materials, such as fiber-reinforced

composites, particle reinforced adhesives, concrete and even metal [4, 35]. Special consideration is given also to porous materials with periodic inner structure [22, 58]. Multi-scale mechanics allows direct study of the influence of material response at the micro level to macroscopic material behavior. The biggest challenge in multi-scale mechanics is identifying the relationships that bridge various length scales. Moreover, the use of different kinds of methods is limited by the characteristics of the problem to be solved. The final goal of multi-scale modeling is to design a combined macroscopic-microscopic computational method that is much more efficient than solving the full microscopic model at the lower level and provides the required information with the desired accuracy. Detailed overviews of multi-scale methods are available elsewhere [8, 17, 45, 72].

There is no unique classification that unifies all multi-scale methods presently available [18]. From a methodological perspective, different categories of multi-scale methods can be identified related to the location and geometry of the heterogeneous scale. Multi-scale methods can also be classified from an algorithmic perspective, referring to the actual solution procedure, as parallel or serial methods, and coupled or decoupled methods. Multi-scale methods classified based on the underlying problem formulation include concurrent methods, in which both scales are simultaneously addressed; hierarchical methods, where the scales are linked hierarchically; and hybrid methods, which have properties of both previously mentioned classes. Variational multi-scale methods constitute a particular category of hierarchical techniques. This category relies on the weak form of the governing equations, which are split into fine and coarse scale contributions. We can also roughly separate multi-scale methods into two groups based on whether they use homogenization techniques or domain decomposition methods.

The computational homogenization method is the most common hierarchical method. It uses an iterative coupled algorithm to connect different scales. Homogenization is used as a method of scale bridging, which essentially relies on averaging theorems. A fundamental assumption is that the scales are far apart. In this case, boundary zones require specific treatment, because in these zones the microstructure cannot be homogenized. Interscale relations in multi-scale methods have a high impact on overall behavior. In classical homogenization theories, constant strain and stress are assumed at the micro scale, together with Hill's energy condition. Computational homogenization techniques use discretization methods, such as FEM to capture the constitutive behavior of RVE, a representative volume element. Three main types of boundary conditions can be imposed on the RVE: fully prescribed displacements, fully prescribed tractions, and periodic boundary conditions. The latter enforces a displacement constraint which is suited for periodic media,

while the others are based on uniform strain and stress assumptions. A precise overview of these methods as initially developed for the computation of linear problems is given in [70]. The methods were later extended to nonlinear problems [12, 13, 18, 42]. Nonlinear homogenization methods have wide-ranging applications to many natural and manufactured materials: asphalt, bone, ceramics, composites, concrete, geological materials and granular media, glass, metals, paper, polymers, rock, snow, ice, textile, biological tissues, and so on. At small scales, nonlinear phenomena are the rule rather than the exception. Plasticity, crack nucleation and propagation, defect mechanics (e.g., dislocations), phase transformations, inelastic creep and relaxation, and microstructure evolution in general, are the prime drivers for material nonlinearities. Homogenization of solids accounting for both geometric and material nonlinearity is more demanding. Scale transitions in damage and fracture constitute one of the most complex subjects in multi-scale mechanics, as damage is a typical phenomenon that develops across all length scales. Incorporating localization and fracture (discontinuities) in a multi-scale setting violates the classical principle of scale separation, which disables the application of most classical homogenization methods. The convergence of these homogenization methods was studied in [57]. In this dissertation, FE^2 will be studied as a representative of homogenization methods. FE^2 is a standard two-level finite element homogenization approach, where one micro FE model (also called an RVE), is at each macro mesh integration point.

In cases where the scale-separation principle does not hold, the domain decomposition strong coupling multi-scale framework is preferred over homogenization techniques[71]. Coarse and fine-scale regions are processed simultaneously and are glued together through inter-scale relations or micro-to-macro connections. These can be more expensive than homogenization techniques, but cheaper than direct numerical simulation. Methods for interscale connections are collocation methods and mortar methods, which are commonly compared to the strain and stress approaches of the Hill-Mandel theory. Domain decomposition method is a concurrent method [38] that allows the localization of fine-scale analysis in areas of interest, significantly reducing the overall computational cost. The structure is decomposed into non-overlapping subdomains. For each subdomain, the Schur complement of the tangent matrix and the condensed residual can be formed through the elimination of inner degrees of freedom (DOFs). This process can be distributed to parallel processors. For each iteration of the iterative solution algorithm, the global linear problem linking all boundary unknowns is formed and compatibility of displacements over the boundaries is obtained. A final localization step enables one to determine an inner solution in substructures. The global condensed problem is constructed using the last consistent tangent operator of each substructure. Because the number of unknowns can

be high and computing and assembling tangent operators for each substructure can be very expensive, domain decomposition is often performed with iterative parallel solvers rather than direct solvers. As shown in this dissertation the described algorithm is only consistent when the local problem is path independent or one macro step is followed by exactly one micro step.

In [7, 21], domain decomposition methods with nonlinear localization, such as the Newton-Krylov-Schur method (NKS), have been analyzed. The authors discussed possibilities to improve robustness and the convergence of parallel computation in the case of strongly nonlinear and large heterogeneous problems, as those encountered in buckling and post-buckling. The NKS method combines the Newton method, for the linearization and the incremental iterative scheme, SCHUR condensation of the tangent problem to define the interface problem, and the Krylov parallel iterative solver, for solving the linear problem condensed on the interfaces. The idea for the solution to this problem comes from the LATIN method[33, 51]. This method is based on the idea of separating the equations into two independent parts: local nonlinear equations, and global linear equations. These two groups of equations are treated iteratively until convergence is reached. At each iteration, one must solve a homogenized macro problem and a set of independent micro problems. The second possibility that was explored, was a mixed domain decomposition method with Robin-like boundary conditions . This method shares similar features with path-following methods for controlling algorithm convergence. The methods allow computational efforts to concentrate on the effectively nonlinear areas while reducing the number of global calculations and, thus, the quantity of exchanged data. As an alternative to iterative solvers, a direct solver combined with sensitivity analysis as described in the next chapters is presented. As a representative of domain decomposition methods, the mesh-in-element (MIEL) method was implemented.

1.1.2 Unified multi-scale method based on sensitivity analysis

The sensitivity analysis aims to calculate derivatives of an arbitrary response functional with respect to chosen parameters. The response functional can depend on arbitrary analysis model inputs (material constants, load intensity, and distribution, shape parameters, etc.), as well as on arbitrary intermediate or final results of the analysis (solution vectors, derived quantities such as stress tensor, integrated quantities such as damage or ductility factors, etc.). For the calculation of sensitivities, different methods can be used, such as the finite difference approximation, analytical sensitivity analysis (which could be

direct or adjoint), and semi-analytical methods. Expressions needed within the analytical sensitivity analysis can be derived manually or automatically with the use of automatic differentiation software. The complete automation of the sensitivity analysis is thus only possible if the automatic differentiation is applied to the complete simulation code, or the finite difference approximation is used. Within the commercial FE environments, semi-analytical sensitivity analysis is usually implemented. Within the semi-analytical sensitivity analysis, the global sensitivity problem is solved by the direct differentiation or adjoint method, while the derivatives at the individual finite element level are calculated by the finite difference approximation. The analytical sensitivity analysis enables efficient evaluation of sensitivities that are exact except for round off errors. Once the sensitivity of the basic problem unknowns is established, the derivatives of the response functional can also be evaluated. A comprehensive overview of the possible approaches can be found in [25].

In the literature, a lot of attention has been paid to the computation of the macroscopic tangent, which is an essential and numerically demanding part of any multi-scale simulation. The possibilities vary from applying the finite difference approximation of the macroscopic tangent (which is expensive and inaccurate but general) to deriving corresponding analytical expressions using different methods (for discussion on methods see e.g. [56]). Another alternative is standard sensitivity analysis [27, 44]. In multi-scale methods, micro and macro scales are connected, and this dependency can be described with the use of sensitivity analysis. For the implementation of multi-scale methods based on sensitivity analysis, sensitivity analysis with respect to prescribed essential boundary conditions is needed. First order essential boundary conditions sensitivity analysis is required for FE^2, and second order for MIEL.

Implementing multi-scale methods based on the sensitivity analysis of essential boundary conditions rather than on calculation of the Schur complement of the macro tangent matrix has many advantages (see e.g. [32, 45, 56] for FE^2 method and [24] for MIEL method). This is especially important for coupled path-dependent problems, such as finite-strain plasticity, where consistent linearization of the tangent matrix is of high importance. It is shown in this thesis that for MIEL-type methods, the analytical second order sensitivity analysis is numerically superior to the Schur complement implementation.

Within the standard implementation of nonlinear multi-scale methods, only the macro scale is parametrized by the load factor. Consequently, each macro step is followed by exactly one step at the micro level, and a path-following algorithm is applied only at

the global level. In this work, a nonlinear multi-scale computational scheme with two interacting path-following methods at two different levels is developed. An algorithm is derived for consistent parametrization of both macro and micro problems leading to a two-level path-following algorithm. Consistent linearization of the two-level path-following algorithm requires the introduction of a relative, rather than full, sensitivity analysis. Consistent update of sensitivity values results in quadratic convergence of the implemented methods, FE^2 and MIEL. A unified approach to multi-scale modeling is defined by an algorithm, where the multi-scale program code is automatically derived, and various types of multi-scale and single-scale approaches can be freely mixed while retaining quadratic convergence of the Newton-Raphson procedure. The derived algorithm was implemented within the *AceFEM* computational environment.

1.1.3 Metamaterials and additive manufacturing techniques

Additive manufacturing (AM), known also as 3D-printing, is becoming more popular in industry as well as with individuals. The applications of AM have significantly expanded from predominantly prototyping, tooling, and fixtures to producing many other functional components in industries such as biomedicine, aerospace, automobile, electronics, and energy. In many of these functional applications, the most sought-after capability of AM is the enablement of design for functionality, which focuses on maximizing the functions and performance of structures with minimum resource consumption (e.g., materials, production time, defect rate, etc.). There is a noticeable trend to personalization and localization of production. 3D-printing is becoming an alternative to mass industrial production in countries with a cheap labor force [15]. It should provide improvements in terms of time-to-market, ecological impact and design compared to traditional industrial processes [61]. Although additive manufacturing technologies have undergone significant development in recent years, significant challenges remain in understanding the physics of the processes. Currently, the existing knowledge about AM is still heavily empirical, especially in the areas of process development, manufacturability, and design optimization.

Different materials are used for 3D-printing, such as polymers, metals, concrete and biological materials. A comprehensive overview of techniques and materials can be found in [50]. Support technologies such as software systems, vacuum casting, investment casting, plating, and infiltration are described in [19]. Basic metallic additive manufacturing techniques are selective laser sintering, direct metal laser sintering, selective laser melting, electron beam melting and direct metal deposition.

With the advance of 3D-printing, which enables manufacturing arbitrary shapes, new possibilities for optimization of products appeared. Optimization has focused on shapes that have advantageous mechanical properties and cannot be made with classical production techniques, such as structures with inner openings, lattices, and trusses. In [16] a review of smart materials based on a classification of advanced structured materials and responsive materials is presented and current applications are described. In civil engineering products are mainly unique and thus ideally suited to using these new techniques.

A metamaterial is a heterogeneous hybrid material that can be designed and manipulated to obtain extraordinary properties beyond those that a classical composite of the same constituent materials exhibits. Originally electromagnetic metamaterials were proposed as a way of tailoring electromagnetic and optical properties [53]. The metamaterial concept was later extended to elastic, acoustic, and thermal properties, and has spread to many other fields of physics such as thermodynamics, acoustics, and mechanics. Furthermore, the field of metamaterials has been extended from the mere pursuit of various exotic properties towards the realization of practical devices, leading to the concepts of dynamically reconfigurable metadevices or functional metasurfaces [60].

Mechanical metamaterials are man-made structures with counterintuitive mechanical properties that originate in their microstructural geometry, rather than their material composition [68]. They have received increasing attention during the last few years, partially due to the advances in additive manufacturing techniques that have enabled the fabrication of materials with arbitrarily complex micro-architectures [69]. The rationally designed micro-architecture of mechanical meta-materials gives rise to unprecedented or rare mechanical properties that could be exploited to create advanced materials with novel functionalities. These unusual mechanical properties include negative values for Poisson's ratio, elasticity, stiffness, compressibility, and thermal expansion coefficient. These metamaterials can be easily compressible, yet not easily deformable; be lightweight, yet ultrastrong; and be deployable, lightweight, bistable, and reprogrammable. Some examples of materials that have been developed include auxetic, ultra-lightweight, negative mass density, negative modulus, penta-mode, dilational, anisotropic mass density, origami, nonlinear, bistable, reprogrammable, and seismic shielding mechanical metamaterials. Metamaterials could be used in diverse applications, such as shock absorbers, support structures, reflectors or concentrators, mechanical cloaks, and structures for space missions.

Auxetic materials, described in [10, 11, 34, 37, 48, 67], are materials with a negative Poisson's ratio, expanding in the direction perpendicular to the applied tensile stress,

and contracting in response to perpendicular compressive stress. This occurs due to their particular internal structure and the way this deforms when the sample is uniaxially loaded. Three well-established basic auxetic structures can be identified: reentrant structures, chiral structures, and rotating rigid structures. Almost all of these models are based on a simple mechanism that is treated as a unit cell leading to a global stiffening effect. Due to their capability to distribute stress over a larger part of the material, auxetic materials have improved resistance against damage and are important in practical applications for civil and aeronautical engineering, defense equipment, smart sensors, filter cleaning, and biomechanics. New auxetic materials have been proposed in different sources [2, 3, 14, 52, 62]. Numerous analytical and computational methods have been developed for the design of mechanical metamaterials. Many papers focus on the design of meta-materials with periodic microstructures manufacturable by additive manufacturing techniques. The typical computational method used is topology optimization, which involves optimizing a unit cell's layout subject to an objective function and boundary conditions. The determination of effective material properties is usually done by some type of multi-scale method discussed in the next section.

The advances in 3D printing and additive manufacturing techniques have enabled the experimental observation of metamaterials with mechanical behavior that had previously been proposed based on theoretical concepts and had only been studied theoretically or computationally. The vast majority of previous studies have focused on the static, quasi-static or specific types of dynamic behaviors of mechanical metamaterials. However, the actual use of mechanical metamaterials for structural applications requires a thorough study of their fatigue behavior. The study of mechanical behavior is extended to the nonlinear range.

1.1.4 Multi-scale optimization methods

Additive manufacturing permits the fabrication of components with geometrical complexity far beyond what can be achieved with conventional manufacturing technologies. Optimization provides a means for intelligently exploiting this design freedom, making these two technologies an ideal fit [6]. Optimization methods can be divided into topology optimization and parametric optimization. Parametric optimization can be subdivided into size and shape optimization, depending on what is affected by the optimization parameters. Sometimes, a combination of different methods is used. For example, topology optimization might be performed first to get an idea of the approximate optimal geome-

try followed by the parametrization of the geometry. Moreover, optimization methods can be divided into gradient-based optimization and evolution algorithms. For nonlinear problems that we are concerned with, evolution algorithms for optimization are not appropriate and we will focus only on the gradient method.

The topology optimization method is an iterative design process, which optimizes a structural or material geometry in a given design domain for a specified objective function, e.g. compliance, material weight or Poisson's ratio and a set of constraints. An overview of the developments within topology optimization can be found in [9]. An application-oriented review of topology optimization approaches is done in [43] with an attempt to illustrate the efficiency in the design of high-performance structures. The topological optimization of heterogeneous media has been an active research topic in the last decades and has become a subject of major importance with the growing development of additive manufacturing processes. Multi-scale methods for topology optimization were performed to avoid solving the full problem on a single-scale, involving all degrees of freedom within the topology optimization framework, which could be computationally very costly. In [36] a new hierarchical multi-scale formulation is developed to account for both the auxetic behavior of the microstructure and the stiffness of the macrostructure. It is used for topological design optimization of functionally graded cellular composites with auxetics using a level set method. Advances in multi-scale topology optimization and computational homogenization FE^2 are summarized in [65].

Additive manufacturing might induce limitations on the size of local details, which can lead to a violation of scale separation, as occurs for structures containing large holes, cracks, or inclusions with dimensions not much smaller than the structure dimensions or in presence of localized strain fields. In such a case, classical homogenization methods may lead to an inaccurate description of the effective behavior as nonlocal effects or strain-gradient effects may occur within the structure. One solution is to use a strain gradient homogenization method presented in [59], which is based on the filter-based homogenization. In [64], topology optimization with substructuring was presented, which considers two different yet strongly connected scales. Each substructure is condensed first into a super-element with reduced degrees of freedom and is associated with a density design variable indicating the material volume fraction. The density design variable is then linked to a geometry feature parameter of the inner structure. The approach presented in [1] uses a contrast independent coarse basis preconditioner, takes boundary conditions into account and ensures connected and macroscopically optimized microstructures regardless of the difference in micro and macroscopic length scales.

The gradient needed within the gradient-based optimization methods is evaluated by sensitivity analysis. Sensitivity based topology optimization was discussed in the literature. In [49], the Topological-Shape Sensitivity Method based on classical shape sensitivity analysis is described. The topological derivative results from classical asymptotic analysis around spherical cavities and is used as a descent direction in topology optimization problems. An exact analytical formula for the topological sensitivity of the macroscopic response of elastic microstructures to the insertion of circular inclusions is proposed in [20]. This alternative way to compute the topological derivative is based on shape sensitivity analysis concepts. The macroscopic response is assumed to be predicted by homogenization.

In [46], a topological optimization methodology that leads to additive manufacturing designs requiring significantly reduced support structures is presented. Structures achieved by topology optimization and fabricated with 3D printing are investigated in [66]. The authors considered particular features of microstructures and macro mechanical performances. They also experimentally explored stiffness and strength anisotropies existing in the 3D printed polymer material using stereolithography (SLA) and the titanium material using selective laser melting (SLM). The standard specimens and typical structures obtained by topology optimization were fabricated along different building directions. Consideration of the particular behaviors of 3D printed materials was shown to be indispensable for structural design and optimization.

Most of the work in the field of multi-scale optimization was done for topology optimization and linear problems. In civil engineering, we design buildings based on limit states, dealing with strongly nonlinear problems. Additionally, our goal is to optimize ductility for plastic deformations in the nonlinear phase. We want to optimize microstructures for the dissipation of energy in an earthquake. Instead of topology optimization, we focus on gradient-based parameter optimization, where sensitivity analysis is used for evaluation of gradients for optimization, as well as for the connection between scales.

1.2 Motivation and objectives

The development of additive manufacturing and increasing computational power together with the need for detailed computational modeling has led to the demand for sophisticated numerical tools, such as multi-scale methods and multi-scale optimization algorithms. These methods can also be used for a general nonlinear case, where scales are connected

through sensitivity analysis of essential boundary conditions. The general objective of this work is to develop a unified multi-scale algorithm that can be used for a general nonlinear case.

The key aspects to be investigated in this work are as follows:

- Automatic differentiation based notation (ADB) of first and second order essential boundary conditions sensitivity analysis derived for FE^2 and MIEL (mesh-in-element) multi-scale methods;
- Consistently linearized two-level path-following algorithm for the solution of multi-scale problems;
- Sensitivity based formulations of FE^2 and MIEL methods, that together form a unified approach to multi-scale modeling, for the nonlinear path-dependent case;
- Efficient gradient-based multi-scale optimization algorithm, based on exact sensitivity analysis and an arbitrary shape parameterization at the micro level.

1.3 Methodology

Implementation of sensitivity analysis based unified multi-scale method is performed with the use of a state of the art symbolic-numeric approach using *AceGen* and *AceFEM* environments, which run inside the tool for symbolic computation Wolfram Mathematica. *AceGen* is an automatic finite element code generator. It combines the abilities of Wolfram Mathematica, automatic differentiation, automatic code generation and simultaneous expression optimization. *AceFEM* is an environment for calculation with finite elements. Programs enable analytical sensitivity analysis of first and second order, which is used for the implementation of multi-scale finite element methods and also for the multi-scale optimization algorithm.

1.4 The outline of the thesis

The thesis is organized as follows.

- Chapter 2 summarizes basic equations of continuum mechanics and defines multi-scale problems. Symbolic description of mechanical problems, the use of advanced automatic differentiation techniques and the hybrid symbolic-numeric environment are introduced.
- Chapter 3 describes the multi-scale algorithm and the two-level path-following algorithm, together with primal and sensitivity analysis, focusing on essential boundary sensitivity analysis, which is used for the implementation of multi-scale methods.
- Chapter 4 incorporates all relevant techniques presented in previous chapters to implement FE^2 and MIEL multi-scale method and presents a unified approach to multi-scale modeling.
- Chapter 5 verifies the implementation of the multi-scale algorithm.
- Chapter 6 describes the multi-scale gradient-based optimization algorithm.
- Chapter 7 summarizes the present work and presents conclusions.

2 DEFINITION OF MULTI-SCALE PROBLEMS

In this section definition of multi-scale problems is described. Two different multi-scale methods are introduced, FE^2 and MIEL. First, some basic equations of continuum mechanics are presented followed by automatic differentiation based (ADB) problem formulation. The governing equations at the micro and macro level together with scale coupling are given.

2.1 Basic equations of continuum mechanics

The basic equations of continuum mechanics that are needed for the finite element formulation of solid mechanics and structural problems are briefly summarized from [30, 63]. The fundamental equations of kinematics, balance law, and constitutive equations, as well as variational principles, are presented.

2.1.1 Kinematics

Deformation

Kinematics describes the motion and deformation of a body in time. Here the continuum approach is applied in which a body B consists of material particles P, $P \in B$, which occupy a region within the Euclidean vector space $\mathbb{E}^3$. The configuration of a body B is a one-to-one mapping $\boldsymbol{\varphi} : B \rightarrow \mathbb{E}^3$, which places the particles of B in $\mathbb{E}^3$. The location of a particle X at time $t \in \mathbb{R}^+$ yields

$$\boldsymbol{x} = \boldsymbol{\varphi}(\boldsymbol{X}, t)\,, \tag{1}$$

where $\boldsymbol{X}$ represents particle X in the reference configuration B. Based on this the placements $\boldsymbol{x}$ and $\boldsymbol{X}$ can be formulated as position vectors in $\mathbb{E}^3$ with respect to the origin $\boldsymbol{O}$, see Figure 1. Point X is defined in the reference configuration by the position vector $\boldsymbol{X} = X_a \boldsymbol{E}_a$. $\boldsymbol{E}_a$ defines an orthogonal base system in the reference configuration with origin $\boldsymbol{O}$. Now that the placement $\boldsymbol{X}$ and $\boldsymbol{x}$ are defined, the displacement can be introduced as the difference between them

$$\boldsymbol{u} = \boldsymbol{x} - \boldsymbol{X}\,. \tag{2}$$

Being the material gradient of the placement $\boldsymbol{x}$, the tensor $\boldsymbol{F}$ maps an infinitesimal line

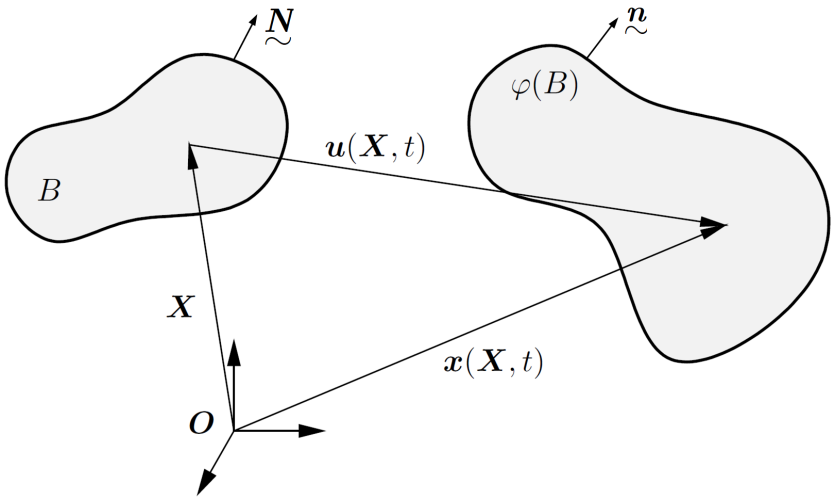


Figure 1: Initial and current configuration of body B
Slika 1: Začetna in trenutna konfiguracija telesa B

element $d\boldsymbol{X}$ from the reference configuration to the current configuration

$$d\boldsymbol{x} = \boldsymbol{F}\, d\boldsymbol{X} \tag{3}$$

$$\boldsymbol{F} = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{X}} = \mathbf{I} + \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{X}} = \mathbf{I} + \boldsymbol{H}, \tag{4}$$

where $\boldsymbol{H} = \partial \boldsymbol{u}/\partial \boldsymbol{X}$ is the displacement gradient tensor. The tensor $\boldsymbol{F}$ describes both stretching and rotation. To maintain the connection of B during the deformation process the mapping has to be one-to-one which excludes a singularity of $\boldsymbol{F}$. This is fulfilled with

$$J_F = \det \boldsymbol{F} \neq 0\,. \tag{5}$$

J_F defines a determinant named after Jacobi. Furthermore to exclude a self penetration of the body also $J_F > 0$ has to be fulfilled. Since $\boldsymbol{F}$ cannot be singular the inverse $\boldsymbol{F}^{-1}$ exists, which can be applied to invert relation

$$d\boldsymbol{X} = \boldsymbol{F}^{-1} d\boldsymbol{x}\,. \tag{6}$$

The transformation between volume elements of initial and current configuration is provided by the relation

$$dv = J_F\, dV\,. \tag{7}$$

Strain measures

Deformation gradient $\boldsymbol{F}$ can be uniquely decomposed into its stretching and rotational part by a polar decomposition

$$\boldsymbol{F} = \boldsymbol{R}\,\boldsymbol{U} = \boldsymbol{V}\,\boldsymbol{R}\,, \tag{8}$$

with the proper orthogonal rotation tensor $\boldsymbol{R}$. The symmetric, positive definite tensors $\boldsymbol{U}$ and $\boldsymbol{V}$ are called right stretch tensor and left stretch tensor. With the help of these tensors, another pair of deformation measures can be defined

$$\boldsymbol{C} = \boldsymbol{F}^T\boldsymbol{F} = \boldsymbol{U}\,\boldsymbol{U} \tag{9}$$

$$\boldsymbol{b} = \boldsymbol{F}\,\boldsymbol{F}^T = \boldsymbol{V}\,\boldsymbol{V}. \tag{10}$$

$\boldsymbol{C}$ is called right Cauchy-Green tensor and $\boldsymbol{b}$ is called left Cauchy-Green tensor. With the help of these deformation tensors, strain measures for both configurations are defined

$$\boldsymbol{E} = \frac{1}{2}(\boldsymbol{C} - \mathbf{I}) \tag{11}$$

$$\boldsymbol{e} = \frac{1}{2}(\mathbf{I} - \boldsymbol{b}^{-1})\,. \tag{12}$$

$\boldsymbol{E}$ is Green-Lagrange strain tensor, which is defined in the reference configuration and $\boldsymbol{e}$ is Euler-Almansi strain tensor defined in current configuration. The linearization of both strain tensors around $\boldsymbol{u} = 0$ leads to the definition of the small strain tensor $\boldsymbol{\varepsilon}$

$$\boldsymbol{\varepsilon} = \frac{1}{2}(\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{X}} + (\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{X}})^T)\,. \tag{13}$$

In case of incompressibility which is important in rubber materials and metal plasticity the constraint condition $\det\boldsymbol{F} = J_F = 1$ has to be fulfilled. Multiplicative decomposition of the deformation gradient is written as

$$\boldsymbol{F} = J_F^{\frac{1}{3}}\hat{\boldsymbol{F}}\,\hat{\boldsymbol{F}} \qquad \hat{\boldsymbol{F}} = J_F^{-\frac{1}{3}}\boldsymbol{F}. \tag{14}$$

It preserves a priori the volume of $\hat{\boldsymbol{F}}$ (isochoric motion), since $\det\hat{\boldsymbol{F}} \equiv 1$. From (11) a relation between the isochoric part of the right Cauchy Green deformation tensor $\hat{\boldsymbol{C}}$ and $\boldsymbol{C}$ is obtained

$$\hat{\boldsymbol{C}} = \hat{\boldsymbol{F}}^T\hat{\boldsymbol{F}} = {J_F}^{-\frac{2}{3}}\boldsymbol{F}^T\boldsymbol{F} = {J_F}^{-\frac{2}{3}}\boldsymbol{C}. \tag{15}$$

The multiplicative split of $\boldsymbol{F}$ in a volume changing part J_F and a volume preserving part

$\hat{\boldsymbol{F}}$ in the nonlinear theory, corresponds to an additive decomposition of the strain tensor in the geometrically linear theory in a deviatoric $\boldsymbol{e}_D = \boldsymbol{\varepsilon} - \frac{1}{3}\operatorname{tr}\boldsymbol{\varepsilon}\,\mathbf{I}$ and a volumetric part

$$\boldsymbol{\varepsilon} = \boldsymbol{e}_D + \frac{1}{3}\operatorname{tr}\boldsymbol{\varepsilon}\,\mathbf{I}\,. \tag{16}$$

In the theory of elasticity, hyper-elasticity is generally assumed. This means that the actual stress-strain path that is taken to achieve a certain deformation is considered irrelevant for the stress-strain relation. If, however phenomena like viscosity or plasticity are taken into account, the history of the deformation has to be followed. Material velocity gradient

$$\dot{\boldsymbol{F}} = \frac{\partial \dot{\boldsymbol{x}}}{\partial \boldsymbol{X}}\,, \tag{17}$$

is introduced in terms of the material time derivative of the deformation gradient $\boldsymbol{F}$. Time derivative of Green-Lagrange strain tensor is written as

$$\dot{\boldsymbol{E}} = \frac{1}{2}(\dot{\boldsymbol{F}}^T\boldsymbol{F} + \boldsymbol{F}^T\dot{\boldsymbol{F}}) = \boldsymbol{F}^T d\boldsymbol{F} \tag{18}$$

for the continuous case.

2.1.2 Balance Equations

This section contains the differential formulations which describe the local balance equations such as balance of mass, balance of linear and angular momentum. These equations represent the fundamental relations of continuum mechanics.

Balance of linear momentum

The linear momentum or the translational momentum is given in the current and initial configuration by

$$\boldsymbol{L} = \int\limits_{\varphi(B)} \rho\,\boldsymbol{v}\,dv = \int\limits_{B} \rho_0\,\boldsymbol{v}\,dV \tag{19}$$

for the continuous case. The change of linear momentum $\boldsymbol{L}$ in time is equal to the sum of all external forces (volume and surface forces) acting on a body B, and can be expressed with

$$\dot{\boldsymbol{L}} = \int\limits_{\varphi(B)} \rho\,\bar{\boldsymbol{b}}\,dv + \int\limits_{\varphi(\partial B)} \boldsymbol{t}\,da\,. \tag{20}$$

$\rho\,\bar{\boldsymbol{b}}$ defines the volume force, and $\boldsymbol{t}$ is the stress vector acting on the surface of the body.

With Cauchy's theorem which relates the stress vector $\boldsymbol{t}$ to the surface normal $\boldsymbol{n}$ via the linear mapping, the stress vector can be expressed with Cauchy stress tensor $\boldsymbol{\sigma}$, $\boldsymbol{t} = \boldsymbol{\sigma}\boldsymbol{n}$. Using the divergence theorem, the local balance equation of linear momentum with reference to the current configuration $\boldsymbol{\varphi}(B)$ is written

$$\operatorname{div}\boldsymbol{\sigma} + \rho\,\bar{\boldsymbol{b}} = \rho\,\dot{\boldsymbol{v}}\,. \tag{21}$$

$\rho\,\dot{\boldsymbol{v}}$ describes the inertial forces which can be neglected in case of purely static investigations.

Balance of angular momentum

The angular momentum with reference to a point $\boldsymbol{O}$, given by $\boldsymbol{x}_0$ is defined with respect to the current and initial configuration as

$$\boldsymbol{J} = \int\limits_{\boldsymbol{\varphi}(B)} (\boldsymbol{\varphi} - \boldsymbol{x}_0) \times \rho\,\boldsymbol{v}\,dv = \int\limits_{B} (\boldsymbol{\varphi} - \boldsymbol{x}_0) \times \rho_0\,\boldsymbol{v}\,dV\,. \tag{22}$$

The change in time of angular momentum $\boldsymbol{J}$ with respect to a point $\boldsymbol{O}$ is equal to the sum of all moments stemming from external volume and surface forces with respect to point $\boldsymbol{O}$.

$$\dot{\boldsymbol{J}} = \int\limits_{\boldsymbol{\varphi}(B)} (\boldsymbol{\varphi} - \boldsymbol{x}_0) \times \rho\,\bar{\boldsymbol{b}}\,dv + \int\limits_{\boldsymbol{\varphi}(\partial B)} (\boldsymbol{\varphi} - \boldsymbol{x}_0) \times \boldsymbol{t}\,da\,. \tag{23}$$

The local balance of angular momentum demands the symmetry of the Cauchy stress tensor $\boldsymbol{\sigma} = \boldsymbol{\sigma}^T$.

2.1.3 Weak from of equilibrium, variational principles

For the analysis of nonlinear initial boundary value problems in continuum mechanics a coupled system of partial differential equations has to be solved which consist of kinematic relations, local balance of momentum and the constitutive equations. Constitutive equations describe material characteristic, like strain-stress relationship. The strong form of these equations is presented here for hyper-elastic solids. For the description with respect to the initial configuration of B, the first Piola-Kirchhoff stresses $\boldsymbol{P}$ and the second Piola-Kirchhoff stresses $\boldsymbol{S}$ are used, which yield

Kinematics:	$\boldsymbol{F}$	$\boldsymbol{E} = \frac{1}{2}(\boldsymbol{F}^T\boldsymbol{F} - \mathbf{I})$
Equilibrium:	$\operatorname{div}\boldsymbol{P} + \rho_0\,\bar{\boldsymbol{b}} = \rho_0\,\dot{\boldsymbol{v}}$	$\operatorname{div}(\boldsymbol{F}\boldsymbol{S}) + \rho_0\,\bar{\boldsymbol{b}} = \rho_0\,\dot{\boldsymbol{v}}$
Constitutive equation:	$\boldsymbol{P} = \frac{\partial W}{\partial \boldsymbol{F}}$	$\boldsymbol{S} = \frac{\partial W}{\partial \boldsymbol{E}}$

Additionally the boundary conditions for the displacements have to be prescribed on ∂B_u with $\boldsymbol{u} = \bar{\boldsymbol{u}}$ and boundary conditions for the traction have to be formulated on ∂B_σ with $\boldsymbol{P}\,\boldsymbol{N} = \boldsymbol{F}\,\boldsymbol{S}\,\boldsymbol{N} = \bar{\boldsymbol{t}}$, where $\boldsymbol{N}$ is surface normal in initial configuration.

The finite elements method is applied to solve this set of equations. It is based on a variational formulation of the equations summarized above. In case of hyper-elastic material responses a functional in the strain energy can be formulated, leading to a variational principle. An approximation $\boldsymbol{u}_h$ of the exact solution $\boldsymbol{u}$ is inserted into the momentum balance equation, leading to

$$\operatorname{div}\boldsymbol{P}(\boldsymbol{u}_h) + \rho_0\,\bar{\boldsymbol{b}} - \rho_0\,\dot{\boldsymbol{v}}_h = \boldsymbol{R}_h\,. \tag{24}$$

The residual $\boldsymbol{R}_h$ denotes the error because of approximation of displacement $\boldsymbol{u}_h$. It will be reduced to zero in a weak sense by multiplying the residual by a weighting function $\boldsymbol{\eta}$ and by integrating the residual over the whole domain, resulting in

$$\int_B \boldsymbol{R}_h \cdot \boldsymbol{\eta}\,dV = 0 \Longrightarrow \int_B \operatorname{div}\boldsymbol{P}(\boldsymbol{u}_h)\cdot\boldsymbol{\eta}\,dV + \int_B \rho_0\,(\bar{\boldsymbol{b}} - \dot{\boldsymbol{v}}_h)\cdot\boldsymbol{\eta}\,dV = 0 \tag{25}$$

which holds also for exact solution $\boldsymbol{u}$

$$\int_B \operatorname{div}\boldsymbol{P}\cdot\boldsymbol{\eta}\,dV + \int_B \rho_0\,(\bar{\boldsymbol{b}} - \dot{\boldsymbol{v}})\cdot\boldsymbol{\eta}\,dV = 0\,. \tag{26}$$

By partial integration, application of the divergence theorem and introduction of the traction boundary condition, the weak form of linear momentum reads

$$\int_B \boldsymbol{P} : \operatorname{Grad}\boldsymbol{\eta}\,dV + \int_B \rho_0\,(\bar{\boldsymbol{b}} - \dot{\boldsymbol{v}})\cdot\boldsymbol{\eta}dV - \int_{\partial B_\sigma} \bar{\boldsymbol{t}}\cdot\boldsymbol{\eta}\,dA = 0\,. \tag{27}$$

Gradient of $\boldsymbol{\eta}$ can be interpreted as the directional derivative of the deformation gradient $D\boldsymbol{F}\cdot\boldsymbol{\eta}$, also known as variation $\delta\boldsymbol{F}$ of the deformation gradient. In the weak form (27), the first Piola-Kirchhoff stress tensor can be replaced through $\boldsymbol{P} = \boldsymbol{F}\,\boldsymbol{S}$ by the second

Piola-Kirchhoff stress tensor leading to

$$\boldsymbol{P} : \operatorname{Grad} \boldsymbol{\eta} = \boldsymbol{S} : \boldsymbol{F}^T \operatorname{Grad} \boldsymbol{\eta} = \boldsymbol{S} \frac{1}{2} (\boldsymbol{F}^T : \operatorname{Grad} \boldsymbol{\eta} + \operatorname{Grad}^T \boldsymbol{\eta} : \boldsymbol{F}) = \boldsymbol{S} : \delta \boldsymbol{E} . \tag{28}$$

Scalar product of a symmetrical tensor $\boldsymbol{S}$ with an anti-symmetrical part of a tensor is zero and $\delta \boldsymbol{E}$ denotes the variation of the Green-Lagrange strain tensor. Using (28), (27) is rewritten as

$$\int_B \boldsymbol{S} : \delta \boldsymbol{E} \, dV - \int_B \rho_0 \, (\bar{\boldsymbol{b}} - \dot{\boldsymbol{v}}) \cdot \boldsymbol{\eta} dV - \int_{\partial B_\sigma} \bar{\boldsymbol{t}} \cdot \boldsymbol{\eta} \, dA = 0 \, , \tag{29}$$

where the first term denotes the internal virtual work also called stress divergence term. The last two terms describe the virtual work of the applied loading and the inertia term. Another reformulation of the first term in (27) is advantageous in the automated finite element method. From

$$\boldsymbol{P} : \operatorname{Grad} \boldsymbol{\eta} = \boldsymbol{P} : \delta \boldsymbol{F} = \boldsymbol{P} : \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{\varphi}} \delta \boldsymbol{\varphi} \tag{30}$$

follows with the hyper-elastic constitutive equation $\boldsymbol{P} = \frac{\partial W}{\partial \boldsymbol{F}}$ the equivalent form for the stress divergence term

$$\boldsymbol{P} : \operatorname{Grad} \boldsymbol{\eta} = \frac{\partial W}{\partial \boldsymbol{F}} \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{\varphi}} \delta \boldsymbol{\varphi} = \frac{\partial W}{\partial \boldsymbol{\varphi}} \delta \boldsymbol{\varphi} = \frac{\partial W}{\partial \boldsymbol{\varphi}} \boldsymbol{\eta} \, . \tag{31}$$

Thus the weak form (27) can be written as

$$\int_B \frac{\partial W}{\partial \boldsymbol{\varphi}} \boldsymbol{\eta} \, dV + \int_B \rho_0 \, (\bar{\boldsymbol{b}} - \dot{\boldsymbol{v}}) \cdot \boldsymbol{\eta} dV - \int_{\partial B_\sigma} \bar{\boldsymbol{t}} \cdot \boldsymbol{\eta} \, dA = 0 \, . \tag{32}$$

This relation simplifies automatic differentiation since only one differentiation of the strain energy function with respect to all displacement variables is needed instead of two differentiations, first to obtain the stress tensor $\boldsymbol{P}$ and then to get the variation of the deformation gradient $\operatorname{Grad} \boldsymbol{\eta}$.

Variational functionals

In case of a hyper-elastic material there exist a strain energy function W, which describes the elastic energy stored in the solid. Based on this strain energy the classical principle of the minimum of potential energy can be formulated in the geometrically linear theory. Under the assumption that the applied loads are conservative, which means path

independent, the functional

$$\Pi(\boldsymbol{\varphi}) = \int_B \left[W(\boldsymbol{C}(\boldsymbol{\varphi})) - \rho_0 \, \bar{\boldsymbol{b}} \cdot \boldsymbol{\varphi} \right] dV - \int_{\partial B_\sigma} \bar{\boldsymbol{t}} \cdot \boldsymbol{\varphi} \, dA \Longrightarrow STAT \tag{33}$$

can be stated for the static problem. The tensor $\boldsymbol{C}$ denotes the right Cauchy-Green strain tensor which depends upon the deformation. Out of all possible deformations $\boldsymbol{\varphi}$, the ones which make Π stationary fulfill the equilibrium equation. The stationary value of (33) can be computed by the variation of Π with respect to the deformation. For this purpose the directional derivative

$$\delta\Pi = D\,\Pi(\boldsymbol{\varphi})\boldsymbol{\eta} = \frac{d}{d\alpha}\Pi(\boldsymbol{\varphi} + \alpha\,\boldsymbol{\eta})|_{\alpha=0}\,, \tag{34}$$

it yields

$$D\,\Pi(\boldsymbol{\varphi})\boldsymbol{\eta} = \int_B \left[\frac{\partial W}{\partial \boldsymbol{\varphi}}\boldsymbol{\eta} - \rho_0 \, \bar{\boldsymbol{b}} \cdot \boldsymbol{\eta} \right] dV - \int_{\partial B_\sigma} \bar{\boldsymbol{t}} \cdot \boldsymbol{\eta} \, dA = 0\,. \tag{35}$$

The construction of such principle has several advantages, among other it leads to the most efficient finite element formulation when the automated finite element procedure is applied.

2.2 Automatic differentiation based (ADB) notation

AceGen is an advanced automatic code generator, that combines automatic differentiation technique, automatic code optimization and generation with computer algebra system Mathematica [41]. Size of the code is reduced through the control of expression swell [26]. *AceFEM* package is a general finite element environment designed to solve multi-physics and multi-field problems.

Automation of primal and sensitivity analysis is done with *AceGen* . The automatic differentiation (AD) technique can be used for the evaluation of the exact derivatives of any arbitrary complex function defined by an algorithm via chain rule and represents an alternative solution to the numerical differentiation and symbolic differentiation. The result of AD procedure is called "computational derivative"and is written here as

$$\frac{\hat{\delta} f(\mathbf{a})}{\hat{\delta}\mathbf{a}} \equiv \texttt{SMSD[f[a],a]}\,. \tag{36}$$

The AD operator on the left side represents derivative of a function $f(\mathbf{a})$ with respect

to variable $\mathbf{a}$. On the right side is call to AD procedure in *AceGen* . The operator $\frac{\hat{\delta} f(\mathbf{a})}{\hat{\delta}\mathbf{a}}$ has a dual purpose to indicate both the mathematical operation of differentiation and that the AD technique is used to obtain the required quantity. Partial derivatives, total derivatives, directional derivatives, consistent derivatives etc., can all be represented by the AD procedure. Extended functionally of AD can be provided by introducing additional information used within the process of automatic differentiation. It thus defines exceptions within the AD procedure. Let $\mathbf{b}$ be a set of mutually independent intermediate variables that are part of the evaluation of a function f. $\mathbf{f}_b$ is a set of arbitrary functions of $\mathbf{a}$ such that $\mathbf{b} := \mathbf{f}_b(\mathbf{a})$, and $\mathbf{M}$ is an arbitrary matrix. AD exceptions are then introduced by the following formalism

$$\left.\frac{\hat{\delta}(\bullet)}{\hat{\delta}(\bullet)}\right|_{\frac{D\mathbf{b}}{D\mathbf{a}}=\mathbf{M}} \tag{37}$$

where $\frac{\hat{\delta}(\bullet)}{\hat{\delta}(\bullet)}$ stands for the various forms of AD operators. It indicates that during the AD procedure, the total derivatives of an arbitrary set of mutually independent intermediate variables $\mathbf{b}$ with respect to independent variables $\mathbf{a}$ are set to be equal to matrix $\mathbf{M}$. The AD exceptions can be viewed as a bridge inside the chain-rule that goes directly from $\mathbf{b}$ to $\mathbf{a}$.

$$\left.\frac{\hat{\delta} f(\mathbf{a},\mathbf{b}(\mathbf{a}))}{\hat{\delta}\mathbf{a}}\right|_{\frac{D\mathbf{b}}{D\mathbf{a}}=\mathbf{0}} \equiv \texttt{SMSD[f[a,b],a,"Constant"}\rightarrow\texttt{ b]} \tag{38}$$

Exception in (38) means, that during the AD procedure dependency $\mathbf{b}(\mathbf{a})$ on $\mathbf{a}$ is neglected. When alternative or additional dependencies for a set of intermediate variables $\mathbf{b}$ have to be considered for differentiation, then

$$\left.\frac{\hat{\delta} f(\mathbf{a},\mathbf{b})}{\hat{\delta}\mathbf{a}}\right|_{\frac{D\mathbf{b}}{D\mathbf{a}}=\mathbf{M}} = \frac{\partial f}{\partial \mathbf{a}} + \frac{\partial f}{\partial \mathbf{b}}\mathbf{M} \equiv \texttt{SMSD[f[a,b],a,"Dependancy"}\rightarrow\texttt{\{b,a,M\}]}\ . \tag{39}$$

$\mathbf{b}$ in (39) may or may not be algorithmically a function of $\mathbf{a}$. When $\mathbf{b}$ is algorithmically function of $\mathbf{a}$ then (39) defines that the true derivatives $\frac{\partial \mathbf{b}}{\partial \mathbf{a}}$ are neglected and replaced by a matrix $\mathbf{M}$. In case that $\mathbf{b}$ is not algorithmically a function of $\mathbf{a}$ then (39) introduces from the algorithmic point of view an artificial dependency between $\mathbf{a}$ and $\mathbf{b}$. The automatic differentiation exceptions are the basis for the ADB formulation of computational problem. The ADB notation can be directly translated to the *AceGen* input and is part of automatic generation of numerically efficient program codes. Details of the method and of the corresponding software *AceGen* together with numerous examples of *AceGen* inputs can be found in [26, 27, 28, 30].

2.2.1 Implicit solution of nonlinear problems

From the finite element discretization comes the system of nonlinear algebraic equations. Equation of an arbitrary nonlinear problem is written with

$$\mathbf{R}(\mathbf{p}) = \mathbf{0}\,. \tag{40}$$

Solution of the system is vector of unknowns $\mathbf{p}$. For implicit solution of an arbitrary nonlinear problem Newton-Raphson iterative method can be used. It is based on Taylor series expansion of (40) at an already known state $\mathbf{p}^{(i)}$

$$\mathbf{R}(\mathbf{p}^{(i)} + \Delta\mathbf{p}^{(i)}) = \mathbf{R}(\mathbf{p}^{(i)}) + D\mathbf{R}(\mathbf{p}^{(i)})\Delta\mathbf{p}^{(i)} + \mathbf{r}(\mathbf{p}^{(i)}) \tag{41}$$

where the upper index (i) denotes the quantities at the i-th iteration and

$$\Delta\mathbf{p}^{(i)} = \mathbf{p}^{(i+1)} - \mathbf{p}^{(i)}. \tag{42}$$

In (41) $D\mathbf{R}(\mathbf{p}^{(i)})\Delta\mathbf{p}^{(i)}$ characterizes the directional derivative of $\mathbf{R}(\mathbf{p})$ at $\mathbf{p}^{(i)}$, also referred to as linearization. The linearization of the vector $\mathbf{R}$ yields a tangent matrix $\mathbf{K}$. In further derivations all quantities without an upper index are evaluated at the already known state $\mathbf{p}^{(i)}$. From (41) the linear equation system is obtained

$$\mathbf{R}(\mathbf{p}^{(i)} + \Delta\mathbf{p}^{(i)}) \approx \mathbf{K}(\mathbf{p}^{(i)})\Delta\mathbf{p}^{(i)} + \mathbf{R}(\mathbf{p}^{(i)}) = \mathbf{0} \tag{43}$$

and a new value of the unknowns can be obtained from (42)

$$\mathbf{p}^{(i+1)} = \mathbf{p}^{(i)} + \Delta\mathbf{p}^{(i)}. \tag{44}$$

Equations (43) and (44) are the basis of the iterative solution procedure.

2.2.2 ADB form of general potential form

Lets assume that the solution of a problem is defined as the minimum of the potential $\Pi(\mathbf{p}) = \int_\Omega W(\mathbf{p})d\Omega + ...$ where $\mathbf{p} = \{p_1, p_2, ..., p_{n_{tp}}\}^T$ is a set of unknown parameters of the problem. The variation of $\Pi(\mathbf{p})$ is computed as

$$\delta\Pi(\mathbf{p}) = \frac{\partial\Pi(\mathbf{p})}{\partial\mathbf{p}}\delta\mathbf{p} = \int_\Omega \frac{\partial W(\mathbf{p})}{\partial\mathbf{p}}d\Omega\delta\mathbf{p} + ... \tag{45}$$

where $\delta\mathbf{p} = \{\delta p_1, \delta p_2, ..., \delta p_{n_{tp}}\}^T$ is a variation of unknown parameters. (45) leads to a set of nonlinear algebraic equations of the form

$$\mathbf{R} = \int_\Omega \frac{\partial W(\mathbf{p})}{\partial \mathbf{p}} d\Omega + ... = \mathbf{0}\,. \tag{46}$$

The ADB form of the general problem with potential is obtained from (46) where the partial derivative is directly replaced by the computational derivative

$$\mathbf{R} = \int_\Omega \frac{\hat{\delta} W(\mathbf{p})}{\hat{\delta} \mathbf{p}} d\Omega + ... = \mathbf{0}\,. \tag{47}$$

Furthermore, the individual element contribution $\mathbf{R}_e$ to the global residual vector $\mathbf{R}$ is in standard finite element formulations obtained by Gauss integration

$$\mathbf{R}_e \approx \sum_{g=1}^{n_g} w_{gp} \mathbf{R}_g\,, \tag{48}$$

where w_{gp} stands for the Gauss point weights, n_g is the number of Gauss points and $\mathbf{R}_g$ is the Gauss point contribution to the residual vector or Gauss point residual given by

$$\mathbf{R}_g = \frac{\hat{\delta} W(\mathbf{p})}{\hat{\delta} \mathbf{p}}\,. \tag{49}$$

The corresponding ADB form of the Gauss point contribution to the global tangent matrix $\mathbf{K}$ is

$$\mathbf{K}_g = \frac{\hat{\delta} \mathbf{R}_g}{\hat{\delta} \mathbf{p}_e}\,. \tag{50}$$

AceGen input segment for appropriate user subroutine of finite element is presented in Appendix A.2 (Step 3).

2.2.3 ADB form of general weak form

Weak form of a problem is

$$\int_\Omega \mathbf{a}(\mathbf{p}) \cdot \delta \mathbf{b}(\mathbf{p}) d\Omega + ... = \mathbf{0}\,, \tag{51}$$

where $\mathbf{a}$ and $\mathbf{b}$ are tensors of an arbitrary order and $\delta\mathbf{b}$ is a directional derivative or variation of $\mathbf{b}$. Since $\delta\mathbf{b}$ is fictitious quantity, AD cannot be applied directly. The weak form needs to be discretized first and then AD can be applied. The variation of $\mathbf{b}$ is computed as

$$\delta\mathbf{b}(\mathbf{p}) = D\mathbf{b}(\mathbf{p})\delta\mathbf{p} = \frac{\partial\mathbf{b}(\mathbf{p})}{\partial\mathbf{p}}\delta\mathbf{p}\,. \tag{52}$$

The discretized weak form is then given by

$$\int_\Omega \mathbf{a}(\mathbf{p})\cdot\delta\mathbf{b}(\mathbf{p})d\Omega + ... = \sum_{m=1}^{n_{tp}}\left(\int_\Omega \mathbf{a}(\mathbf{p})\cdot\frac{\partial\mathbf{b}(\mathbf{p})}{\partial p_m}d\Omega\right)\delta p_m + ... = \mathbf{0}\,, \tag{53}$$

and leads to a set of n_{tp} algebraic equations of the form

$$\mathbf{R} = \int_\Omega \mathbf{a}(\mathbf{p})\cdot\frac{\partial\mathbf{b}(\mathbf{p})}{\partial\mathbf{p}}d\Omega + ... = \mathbf{0}\,. \tag{54}$$

In the ADB form of the discretized weak form the partial derivative is directly replaced by the computational derivative

$$\mathbf{R} = \int_\Omega \mathbf{a}(\mathbf{p})\cdot\frac{\hat{\delta}\mathbf{b}(\mathbf{p})}{\hat{\delta}\mathbf{p}}d\Omega + ... = \mathbf{0}\,. \tag{55}$$

Pseudo-potential is introduced with

$$W^P = \mathbf{a}(\mathbf{p})\cdot\mathbf{b}(\mathbf{p})\,. \tag{56}$$

The pseudo-potential has to be formed in a way that automatic differentiation of the pseudo-potential accompanied by proper automatic differentiation exceptions leads to the correct set of discretized equations of the problem. This can be achieved by introducing the AD exception that hides the dependency $\mathbf{a}(\mathbf{p})$ from the AD procedure. The ADB form of the discretized weak form reads

$$\mathbf{R} = \int_\Omega \left.\frac{\hat{\delta}(\mathbf{a}(\mathbf{p})\cdot\mathbf{b}(\mathbf{p}))}{\hat{\delta}\mathbf{p}}\right|_{\frac{D\mathbf{a}}{D\mathbf{p}}=\mathbf{0}} d\Omega + ... = \int_\Omega \left.\frac{\hat{\delta}W^P(\mathbf{p})}{\hat{\delta}\mathbf{p}}\right|_{\text{a=const.}} d\Omega + ... = \mathbf{0}\,. \tag{57}$$

The Gauss point contribution to the residual vector is written as

$$\mathbf{R}_g = \mathbf{a}(\mathbf{p}_e)\cdot\frac{\hat{\delta}\mathbf{b}(\mathbf{p}_e)}{\hat{\delta}\mathbf{p}_e} = \left.\frac{\hat{\delta}W^P(\mathbf{p}_e)}{\hat{\delta}\mathbf{p}_e}\right|_{a=const.}\,. \tag{58}$$

The corresponding ADB form of Gauss point contribution to the global tangent matrix **K** is given by (50).

2.3 Multi-scale methods

Definition of two different multi-scale methods, MIEL and FE^2 is described. They can be applied to solve large specter of problems, varying from examples where scales are strongly coupled to those where an assumption of scale separation holds.

2.3.1 MIEL method

MIEL (mesh-in-element) method is a multi-scale finite element method that can be classified as a domain decomposition method. This method is appropriate for cases where the difference between two scales is finite and the scales remain coupled, or when in the region of high gradients the FE^2 multi-scale approach fails. The MIEL scheme was described in [39, 40, 47].

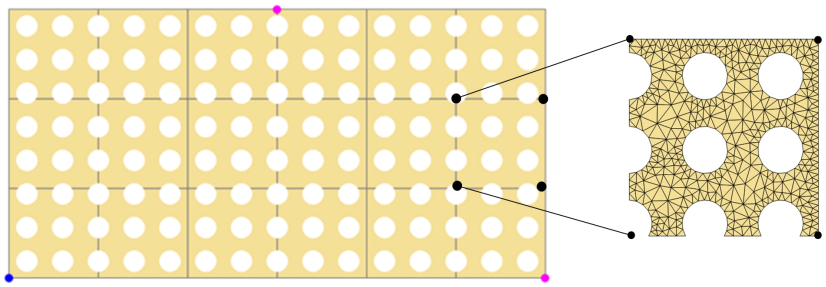


Figure 2: MIEL macro and micro model
Slika 2: MIEL makro in mikro model

Strongly coupled scales are considered, where we equally utilize the finite element method at both scales, yet the micro scale is not infinitely smaller but finitely smaller than the macro scale. By the appropriate formulation the macro mesh element quantities (residual, tangent matrix, etc.) are obtained from the micro FE calculation, replacing in this way the macroscopic constitutive law at the macro finite element level. All the displacement values at the micro mesh boundary are exactly equal to the macro displacement value on the macro mesh at the same point, see Fig. 2. Thus compatibility of the displacements over the boundary of micro mesh element is assumed.

The integration of the internal forces, part of weak form, over the micro mesh, where the micro deformation gradient $\boldsymbol{F}_m$ and micro stress tensor $\boldsymbol{P}_m$ implicitly depend on the degrees of freedom of macro element

$$\int_{\Omega_{Me}} \boldsymbol{P}_M : \delta \boldsymbol{F}_M \, dV = \int_{\Omega_m} \boldsymbol{P}_m : \delta \boldsymbol{F}_m \, dV \, . \tag{59}$$

2.3.2 FE^2 method

Standard two-level finite element homogenization approach FE^2 is appropriate for problems where scales are separated far enough and are only weakly coupled, a detailed description can be found in [32].

In this approach, we have a finite element model for the macro scale and in each Gauss point (a numerical integration point), a representative volume element (RVE) is assigned, which represents the underlying microstructure, see Fig. 3. The assumption of the existence of the RVE, which is considered both smaller enough than the macro scale characteristic volume and bigger enough than the heterogeneities on the micro scale is made. The macro mesh element quantities (residual, tangent matrix, etc.) are obtained from the micro FE calculation at the level of a Gauss point. The material response is obtained by the RVE finite element analysis, replacing in this way the macroscopic constitutive law.

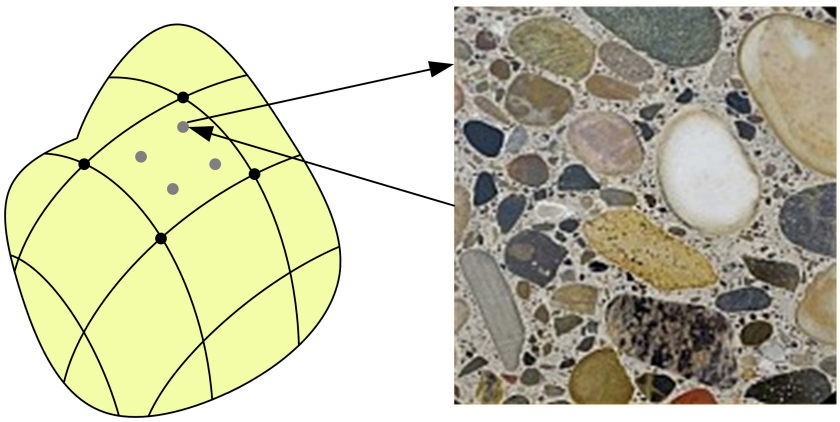


Figure 3: FE^2 macro and RVE model
Slika 3: FE^2 makro in RVE model

The coupling between macroscopic and microscopic levels is based on averaging theorems. The energy averaging theorem, known as Hill–Mandel condition requires equality between virtual work performed by variation of macroscopic deformations in associated

Gauss point at macro-scale and volume average of virtual work performed by variation of microscopic deformations on RVE

$$\boldsymbol{P}_M : \delta \boldsymbol{F}_M = \frac{1}{V_0} \int_{\Omega_m} \boldsymbol{P}_m : \delta \boldsymbol{F}_m \, dV = \{ \boldsymbol{P}_m : \delta \boldsymbol{F}_m \} \tag{60}$$

where $\boldsymbol{P}_M$ is the first Piola–Kirchhoff stress tensor associated with a variation of work conjugate deformation gradient $\delta \boldsymbol{F}_M$, and $\boldsymbol{F}_m$ is micro level deformation gradient, and $\boldsymbol{P}_m$ is the first microstructural Piola–Kirchhoff stress tensor. This criterion is fulfilled by averaging operation.

2.3.3 Micro problem material models

At the micro level we will consider finite element formulation of different material models, starting from small strain elasticity, proceeding to rate-independent nonlinear models, such as an arbitrary finite strain rate-independent elasto-plastic material model.

Small strain elastic material model

Small strain elastic material model formulation is based on small strain tensor $\boldsymbol{\varepsilon}_m$ and linear Hooke's law with strain energy function $W(\boldsymbol{\varepsilon}_{Iso})$ (62), $\boldsymbol{\varepsilon}_{Iso}$ is deviatoric part of $\boldsymbol{\varepsilon}_m$ and μ and K are material constants.

$$\boldsymbol{\varepsilon}_{Iso} = \boldsymbol{\varepsilon}_m - \frac{1}{3}(\operatorname{tr} \boldsymbol{\varepsilon}_m)\mathbf{I} \tag{61}$$

$$W = \mu \operatorname{tr}(\boldsymbol{\varepsilon}_{Iso} \, \boldsymbol{\varepsilon}_{Iso}) + \frac{1}{2} K (\operatorname{tr} \boldsymbol{\varepsilon})^2 \tag{62}$$

Standard weak form of equilibrium equations is written as

$$\int_{\Omega_m} \boldsymbol{\sigma}_m : \delta \boldsymbol{\varepsilon}_m \, dV - \int_{\partial \Omega_m} \boldsymbol{t} \cdot \delta \boldsymbol{u}_m \, dS = 0 \tag{63}$$

where micro stress tensor $\boldsymbol{\sigma}_m$ can be obtained from the elastic strain energy W by $\boldsymbol{\sigma}_m = \partial W / \partial \boldsymbol{\varepsilon}_m$.

Hyper-elastic material model

Hyper-elastic material model formulation is based on micro deformation gradient $\boldsymbol{F}_m$, the components of right Cauchy strain tensor $\boldsymbol{C}$, and Neo-Hookean strain energy function

$W(\boldsymbol{C})$ (65). *AceGen* input segment is presented in Appendix A.2 (Step 2).

$$\boldsymbol{C} = \boldsymbol{F}_m^T \boldsymbol{F}_m,\ J_F = \det \boldsymbol{F}_m \tag{64}$$

$$W = \frac{1}{2}\mu(\operatorname{tr} \boldsymbol{C} - 3 - 2\log J_F) + \frac{1}{4}\lambda(J_F^2 - 1 - 2\log J_F) \tag{65}$$

Standard weak form of equilibrium equations is then written as

$$\int_{\Omega_m} \boldsymbol{P}_m : \delta \boldsymbol{F}_m \, dV - \int_{\partial\Omega_m} \boldsymbol{t} \cdot \delta \boldsymbol{u}_m \, dS = 0 \tag{66}$$

where first Piola-Kirchhof stress tensor $\boldsymbol{P}_m$ can be obtained from the elastic strain energy W by $\boldsymbol{P}_m = \partial W / \partial \boldsymbol{F}_m$.

Small strain elasto-plastic material model

Small strain elasto-plastic model is defined by its elastic strain energy function, plastic evolution equations, and the method for time integration of evolution equations. Additive split of total deformations into elastic and plastic part is assumed.

$$\boldsymbol{\varepsilon}_m = \boldsymbol{\varepsilon}_e + \boldsymbol{\varepsilon}_p \tag{67}$$

$$\boldsymbol{\varepsilon}_{Iso} = \boldsymbol{\varepsilon}_e - \frac{1}{3}(\operatorname{tr} \boldsymbol{\varepsilon}_e)\mathbf{I} \tag{68}$$

$$W = \mu \operatorname{tr}(\boldsymbol{\varepsilon}_{Iso}\, \boldsymbol{\varepsilon}_{Iso}) + \frac{1}{2}K(\operatorname{tr} \boldsymbol{\varepsilon}_e)^2 \tag{69}$$

$$\boldsymbol{\sigma} = \frac{\partial W}{\partial \boldsymbol{\varepsilon}_e},\ \boldsymbol{\sigma}' = \boldsymbol{\sigma} - \frac{1}{3}(\operatorname{tr} \boldsymbol{\sigma})\mathbf{I} \tag{70}$$

$$\phi = (\frac{3}{2}\boldsymbol{\sigma}' \cdot \boldsymbol{\sigma}')^{1/2} - (\sigma_{\mathrm{y}0} + K_{\mathrm{h}}\gamma + R_\infty(1 - \exp(-\delta\,\gamma))) \tag{71}$$

$$\mathbf{Z} = \boldsymbol{\varepsilon}_p - \boldsymbol{\varepsilon}_{pn} - ((\gamma - \gamma_n)\frac{\partial \phi}{\partial \boldsymbol{\sigma}}) = \mathbf{0} \tag{72}$$

$$\mathbf{Q}_{mg} = \{Z_{11}, Z_{22}, Z_{33}, Z_{12}, Z_{13}, Z_{23}, \phi\} = \mathbf{0} \tag{73}$$

$$\mathbf{h}_{mg} = \{\boldsymbol{\varepsilon}_{p11}, \boldsymbol{\varepsilon}_{p22}, \boldsymbol{\varepsilon}_{p33}, \boldsymbol{\varepsilon}_{p12}, \boldsymbol{\varepsilon}_{p13}, \boldsymbol{\varepsilon}_{p23}, \gamma\} \tag{74}$$

$$\mathbf{h}_{mgn} = \{\boldsymbol{\varepsilon}_{pn11}, \boldsymbol{\varepsilon}_{pn22}, \boldsymbol{\varepsilon}_{pn33}, \boldsymbol{\varepsilon}_{pn12}, \boldsymbol{\varepsilon}_{pn13}, \boldsymbol{\varepsilon}_{pn23}, \gamma_n\} \tag{75}$$

Formulation for a small strain elasto-plastic material model is based on a small strain tensor $\boldsymbol{\varepsilon}_m$, the components of a small strain plastic tensor $\boldsymbol{\varepsilon}_p$ as plastic state variables, elastic strain tensor $\boldsymbol{\varepsilon}_e$ (67), linear Hooke's law with strain energy function $W(\boldsymbol{\varepsilon}_{Iso})$ (69), and Mises yield function with exponential hardening law (71). Discretized evolution equations (72), together with yield condition $\phi = 0$, form a set of algebraic equations

$\mathbf{Q}_{mg}$ (73) for a set $\mathbf{h}_{mg}$ (74) of unknown components of plastic strain tensor $\boldsymbol{\varepsilon}_p$ and plastic multiplier γ. $\boldsymbol{\varepsilon}_{pn}$ and γ_n are values of a plastic strain tensor and a plastic multiplier at the end of the last load step. The dependency of equation (73) on the values of variables at the end of the last load step ($\mathbf{h}_{mgn}$) makes the whole problem path-dependent. Standard weak form of equilibrium equations is written with (63).

Finite strain elasto-plastic material model

Finite strain isotropic elasto-plastic model is defined by its elastic strain energy function, plastic evolution equations and the method for time integration of evolution equations. For more details see e.g. [54]. Some of the possible variants are presented in [29].

$$\boldsymbol{b}_e = \boldsymbol{F}_m \boldsymbol{C}_p^{-1} \boldsymbol{F}_m^{-T},\ J_{\boldsymbol{b}_e} = \det \boldsymbol{b}_e \tag{76}$$

$$W = \frac{1}{2}\mu(\operatorname{tr} \boldsymbol{b}_e - 3 - \log J_{\boldsymbol{b}_e}) + \frac{1}{4}\lambda(J_{\boldsymbol{b}_e} - 1 - \log J_{\boldsymbol{b}_e}) \tag{77}$$

$$\boldsymbol{\tau} = 2\boldsymbol{b}_e \frac{\partial W}{\partial \boldsymbol{b}_e},\ \boldsymbol{\tau}' = \boldsymbol{\tau} - \frac{1}{3}(\operatorname{tr} \boldsymbol{\tau})\mathbf{I} \tag{78}$$

$$\phi = (\frac{3}{2}\boldsymbol{\tau}' \cdot \boldsymbol{\tau}')^{1/2} - (\sigma_{\mathrm{y0}} + K_{\mathrm{h}}\gamma + R_\infty(1 - \exp(-\delta\,\gamma))) \tag{79}$$

$$\mathbf{Z} = \boldsymbol{F}_m\, \boldsymbol{C}_p^{-1} - \exp(-2(\gamma - \gamma_n)\frac{\partial \phi}{\partial \boldsymbol{\tau}})\boldsymbol{F}_m\, \boldsymbol{C}_{pn}^{-1} = \mathbf{0} \tag{80}$$

$$\mathbf{Q}_{mg} = \{Z_{11}, Z_{22}, Z_{33}, Z_{12}, Z_{13}, Z_{23}, \phi\} = \mathbf{0} \tag{81}$$

$$\mathbf{h}_{mg} = \{C_{p11}^{-1}, C_{p22}^{-1}, C_{p33}^{-1}, C_{p12}^{-1}, C_{p13}^{-1}, C_{p23}^{-1}, \gamma\} \tag{82}$$

$$\mathbf{h}_{mgn} = \{C_{pn11}^{-1}, C_{pn22}^{-1}, C_{pn33}^{-1}, C_{pn12}^{-1}, C_{pn13}^{-1}, C_{pn23}^{-1}, \gamma_n\} \tag{83}$$

Formulation used in thesis is based on multiplicative split of a micro deformation gradient $\boldsymbol{F}_m$, the components of an inverse right Cauchy plastic strain tensor $\boldsymbol{C}_p^{-1}$ as plastic state variables, elastic left Cauchy strain tensor $\boldsymbol{b}_e$ (76), Neo-Hookean strain energy function $W(\boldsymbol{b}_e)$ (77) and Mises yield function with exponential hardening law (79). Backward Euler is combined with the exponential map for a stable, volume conserving integration of evolution equations [54]. Discretized evolution equations (80), together with yield condition $\phi = 0$, form a set of algebraic equations $\mathbf{Q}_{mg}$ (81) for a set $\mathbf{h}_{mg}$ (82) of unknown components of plastic strain tensor $\boldsymbol{C}_p^{-1}$ and plastic multiplier γ. $\boldsymbol{C}_{pn}^{-1}$ and γ_n are values of a plastic strain tensor and a plastic multiplier at the end of the last load step. The dependency of equation (81) on the values of variables at the end of the last load step ($\mathbf{h}_{mgn}$) makes the whole problem path-dependent. Standard weak form of equilibrium equations is written with (66).

2.3.4 Micro level FEM discretization and derivation of algebraic equilibrium equations

A general finite strain elasto-plastic material model will be treated, since this class of problems is the most general from selected problems presented in previous chapter. Application to other classes (elastic, hyper-elastic, etc.) is therefore easily derived by neglecting appropriate terms.

After finite element discretization of deformation gradient $\boldsymbol{F}_m = \boldsymbol{F}_m(\mathbf{p}_{me})$, where $\mathbf{p}_{me}$ is a set of nodal degrees of freedom of e-th micro element at the current load step, variation $\delta\boldsymbol{F}_m = \partial\boldsymbol{F}_m/\partial\mathbf{p}_{me}\,\delta\mathbf{p}_{me}$ leads from (66) together with the standard GAUSS integration of weak form and standard procedure of assembly of element contributions (denoted here with $\bigwedge$ operator) to a set of algebraic equilibrium equations (84). Equations are at each GAUSS point coupled with an additional set of equations $\mathbf{Q}_{mg}$ (81). The result is the following integration point coupled system of nonlinear algebraic equations

$$\begin{aligned}\mathbf{R}_m(\mathbf{p}_m, \mathbf{h}_m) &= \bigwedge_{e=1}^{n_{me}} \mathbf{R}_{me} + \mathbf{R}_m^{\text{ext}} \\ &= \bigwedge_{e=1}^{n_{me}} \sum_{g\in G_e} w_{gp}\,\mathbf{R}_{mg}(\mathbf{p}_{me}, \mathbf{h}_{mg}) + \mathbf{R}_m^{\text{ext}} = \mathbf{0}, \end{aligned} \tag{84}$$

$$\mathbf{Q}_{mg}(\mathbf{p}_{me}, \mathbf{h}_{mg}, \mathbf{h}_{mgn}) = \mathbf{0} \;:\; g = 1, 2, ...n_{tg} \tag{85}$$

where $\mathbf{R}_{me}$ is a contribution of e-th element to global residual $\mathbf{R}_m$ and $\mathbf{R}_m^{\text{ext}}$ is a vector of external forces. $\mathbf{p}_m$ denotes a set of micro level nodal unknowns. $\mathbf{h}_m = \bigcup_g^{n_{tg}} \mathbf{h}_{mg}$ is a set of unknowns of all GAUSS point problems. G_e is a set of GAUSS points of e-th element and w_{gp} is GAUSS point weight. The GAUSS point contribution $\mathbf{R}_{mg}$ to the element residual $\mathbf{R}_{me}$ leads from (66) to

$$\mathbf{R}_{mg} = J_\xi \boldsymbol{P}_m : \frac{\partial \boldsymbol{F}_m}{\partial \mathbf{p}_{me}} \tag{86}$$

where J_ξ stands for a standard Jacobian of the transformation from the reference coordinate system to the global coordinate system and $\boldsymbol{P}_m : \frac{\partial \boldsymbol{F}_m}{\partial \mathbf{p}_{me}} = \sum_{ij} P_{mij} \frac{\partial F_{mij}}{\partial \mathbf{p}_{me}}$.

For example, the GAUSS point residual $\mathbf{R}_{mg}$ is defined by equation (86). However, form (86) is numerically inefficient from the automatic differentiation point of view. Numerically efficient ADB form of (86) is derived as

$$\mathbf{R}_{mg} = J_\xi \, \boldsymbol{P}_m : \frac{\partial \boldsymbol{F}_m}{\partial \mathbf{p}_{me}} = J_\xi \, \frac{\hat{\delta} W}{\hat{\delta} \boldsymbol{F}_m}\bigg|_{\frac{D\mathbf{h}_{mg}}{D\boldsymbol{F}_m}=\mathbf{0}} : \frac{\hat{\delta} \boldsymbol{F}_m}{\hat{\delta} \mathbf{p}_{me}} = J_\xi \, \frac{\hat{\delta} W}{\hat{\delta} \mathbf{p}_{me}}\bigg|_{\frac{D\mathbf{h}_{mg}}{D\boldsymbol{F}_m}=\mathbf{0}} \tag{87}$$

As a side effect of the iterative solution of GAUSS point equations (81), there exist an implicit (algorithmic) dependency of $\mathbf{h}_{mg}$ on $\boldsymbol{F}_m$. The AD exception $\frac{D\mathbf{h}_{mg}}{D\boldsymbol{F}_m} = \mathbf{0}$ in (87) hides this dependency from automatic differentiation procedure and ensures correct evaluation of the weak form equations. In (87) we start with the scalar and make only one call to AD procedure, which is optimal for the backward mode implementation of automatic differentiation as shown in [27].

3 SOLUTION ALGORITHMS FOR MULTI-SCALE PROBLEMS

3.1 Generalized two-level path-following multi-scale method

For highly nonlinear problems in general, the solution cannot be achieved in one step. More efficient procedures can be derived when the resulting system of algebraic equations can be naturally parametrized. Various path-following algorithms, such as constant load-stepping, adaptive load stepping or arc-length methods, can be applied to solve the nonlinear problem. Within the standard implementation of multi-scale methods, only the macro scale is parametrized. Consequently, each macro step is followed by exactly one step at the micro level and a path-following algorithm is applied only at the global level. Here, an algorithm is derived for consistent parametrization of both macro and micro problems leading to two-level path-following algorithm. For the sake of simplicity, the two-level constant load stepping algorithm is derived. However, it can be easily extended to other path-following approaches.

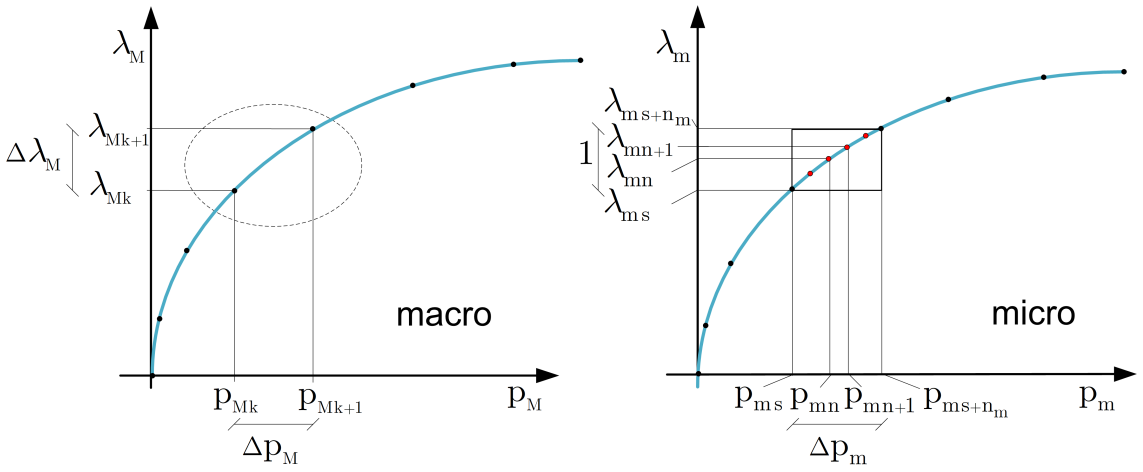


Figure 4: Generalized two-level path-following, multi-scale algorithm
Slika 4: Posplošeno dvonivojsko sledenje obtežni poti, večnivojski algoritem

Let k be the index of the last calculated macro load step and $k+1$ the current macro load step, as shown in Fig. 4. Furthermore, let n be the index of the last converged micro load step, $n+1$ the current micro load step, s the index of the micro load step at the end of the last converged macro load step, n_m the number of micro level steps within the macro level step and $s+n_m$ the index of the micro load step at the end of the macro load step as presented in Fig. 4.

As an example, problems in solid mechanics and nonlinear structural mechanics subjected to quasi-static proportional load are frequently parametrized by introducing loading parameter λ_M. In our case, λ_M will be used to parametrize macro problem. The final value of parameter λ_M is usually determined by the problem at hand, e.g. as total given load factor $\bar{\lambda}_M$. In this case, the total load is split into n_M macro steps. The finite element discretization of macro level leads to a set of nonlinear equations $\mathbf{R}_M$ at the current load level $\lambda_M = \lambda_{M\,k+1}$

$$\mathbf{R}_M(\mathbf{p}_M, \bigcup_{e=1}^{n_{Me}} \mathbf{S}_{Me}, \lambda_M) = \bigwedge_{e=1}^{n_{Me}} \mathbf{R}_{Me}(\mathbf{p}_{Me}, \mathbf{S}_{Me}) - \lambda_M \mathbf{R}_M^{\text{ref}} = \mathbf{0} \tag{88}$$

where $\mathbf{R}_{Me}$ denotes the contribution of internal forces of e-th macro element to the nodal force vector and $\mathbf{R}_M^{\text{ref}}$ is the reference load vector associated with the pattern of the applied nodal forces. $\mathbf{p}_M$ represents a set of nodal unknowns of the problem at macro level and $\mathbf{S}_{Me}$ is a set of variables transferred from the micro level problems to e-th macro element. $\mathbf{S}_{Me}$ is composed of contributions of one or several micro problems. Thus, $\mathbf{S}_{Me} = \bigcup_{r \in \mathcal{M}_e} \mathbf{S}^{(r)}$, where $\mathbf{S}^{(r)}$ is the contribution of the r-th micro problem and $\mathcal{M}_e$ is a subset of micro problems that contribute to the e-th macro element. For a general scheme it is irrelevant what the data represents physically. Transfer of data between macro and micro level is demonstrated in Fig. 5.

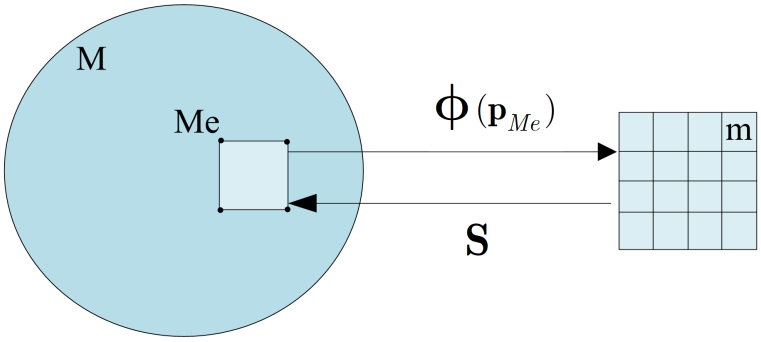


Figure 5: Transfer of data between macro and micro level
Slika 5: Prenos podatkov med makro in mikro nivojem

Within various multi-scale methods, the coupling of the scales can be done in several ways. This thesis addresses methods where micro-macro coupling is achieved by expressing the essential boundary conditions of micro level problem as a function of data calculated at macro level. Let $\bar{\mathbf{p}}_m$ be a set of micro problem nodal unknowns with imposed homogeneous essential (Dirichlet) boundary conditions, $\boldsymbol{\Phi}$ a set of variables calculated at macro level for the current macro load level λ_M on which a selected micro problem depends and

$\bar{\mathbf{p}}_m(\boldsymbol{\phi})$ a function such that at the end of the macro step $\bar{\mathbf{p}}_m = \bar{\mathbf{p}}_m(\boldsymbol{\phi})$. $\boldsymbol{\phi}$ is composed of components of macro deformation gradient in the case of FE^2 method and of components of nodal displacements of macro element in the case of MIEL method. The actual form of $\bar{\mathbf{p}}_m(\boldsymbol{\phi})$ depends on the multi-scale scheme and is presented in the following sections. Let λ_m be a current value of a micro level parameter. At the end of the last macro step, we additionally define λ_{ms} as a value of a micro level parameter and $\bar{\mathbf{p}}_{ms}$ as a value of nodal unknowns with imposed essential boundary conditions. Linear interpolation of $\bar{\mathbf{p}}_m$ within the macro step then leads to the following parametrization of micro level problem

$$\bar{\mathbf{p}}_m(\boldsymbol{\phi}, \lambda_m) = \bar{\mathbf{p}}_{ms} + (\lambda_m - \lambda_{ms})(\bar{\mathbf{p}}_m(\boldsymbol{\phi}) - \bar{\mathbf{p}}_{ms}). \tag{89}$$

The total increment $\Delta\bar{\mathbf{p}}_m$ of the micro essential boundary conditions within the macro load step is defined by

$$\Delta\bar{\mathbf{p}}_m = \bar{\mathbf{p}}_m(\boldsymbol{\phi}) - \bar{\mathbf{p}}_{ms}. \tag{90}$$

Micro level parameter introduced with (89) ensures continuous parametrization of micro problem and has the following properties for the k-th micro step: $(\lambda_m - \lambda_{ms}) \in [0, 1]$, $\lambda_{ms} = k$ and $\lambda_m = k + 1$ at the end of macro step. With the introduction of parameter λ_m, the solution of micro problem within the k-th macro step is achieved in n_m micro steps with associated solution vectors.

The finite element discretization at micro level leads from (84) to the following integration point coupled system of nonlinear algebraic equations for the chosen micro problem

$$\mathbf{R}_m(\mathbf{p}_m, \mathbf{h}_m, \bar{\mathbf{p}}_m(\boldsymbol{\phi}, \lambda_m)) = \mathop{\mathsf{A}}_{e=1}^{n_{me}} \sum_{g \in G_e} w_{gp}\, \mathbf{R}_{mg}(\mathbf{p}_{me}(\boldsymbol{\phi}, \lambda_m), \mathbf{h}_{mg}) + \mathbf{R}_m^{\text{ext}} = \mathbf{0} \tag{91}$$

$$\mathbf{Q}_{mg}(\mathbf{p}_{me}(\boldsymbol{\phi}, \lambda_m), \mathbf{h}_{mg}, \mathbf{h}_{mgn}) = \mathbf{0} \ : g = 1, 2, ...n_{tg} \tag{92}$$

where equilibrium equations $\mathbf{R}_m$ are coupled with discretized evolution equations at the Gauss point level $\mathbf{Q}_{mg}$. A standard two level Newton-Raphson method can be used to solve the resulting Gauss point coupled system of algebraic equations for the unknown $\mathbf{p}_m$ and $\mathbf{h}_m$, as described in [27].

In order to achieve quadratically convergent multi-scale solution algorithm, additionally consistently linearized macro equilibrium equations (88) is needed. The linearization of (88) leads to

$$\mathbf{K}_M = \mathop{\mathsf{A}}_{e=1}^{n_{Me}} \mathbf{K}_{Me} = \mathop{\mathsf{A}}_{e=1}^{n_{Me}} \left(\frac{\partial \mathbf{R}_{Me}}{\partial \mathbf{p}_{Me}} + \frac{\partial \mathbf{R}_{Me}}{\partial \mathbf{S}_{Me}} \frac{D\mathbf{S}_{Me}}{D\boldsymbol{\phi}_{Me}} \frac{\partial \boldsymbol{\phi}_{Me}}{\partial \mathbf{p}_{Me}} \right) \tag{93}$$

where $\boldsymbol{\phi}_{Me} = \bigcup_{r \in \mathcal{M}_e} \boldsymbol{\phi}^{(r)}$ is composed of variables calculated at the e-th macro element and transferred to a subset of micro problems $\mathcal{M}_e$ that effects the e-th macro element. Partial derivatives in (93) are explicit and can be easily derived analytically for a specific multi-scale scheme.

Calculation of the total derivative $\frac{D\mathbf{S}_{Me}}{D\boldsymbol{\phi}_{Me}} = \bigcup_{r \in \mathcal{M}_e} \left(\frac{D\mathbf{S}}{D\boldsymbol{\phi}}\right)^{(r)}$ has to be done at the micro level. As shown later on examples, in general $\mathbf{S}$ depends on $\mathbf{p}_m$ and $\mathbf{h}_m$ as well as on their first derivatives $D\mathbf{p}_m/D\boldsymbol{\phi}$ and $D\mathbf{h}_m/D\boldsymbol{\phi}$. Thus, a total derivative of $\mathbf{S}$ leads to

$$\frac{D\mathbf{S}}{D\boldsymbol{\phi}} = \frac{\partial \mathbf{S}}{\partial \mathbf{p}_m}\frac{D\mathbf{p}_m}{D\boldsymbol{\phi}} + \frac{\partial \mathbf{S}}{\partial \mathbf{h}_m}\frac{D\mathbf{h}_m}{D\boldsymbol{\phi}} + \frac{\partial \mathbf{S}}{\partial (D\mathbf{p}_m/D\boldsymbol{\phi})}\frac{D^2\mathbf{p}_m}{D\boldsymbol{\phi}^2} + \frac{\partial \mathbf{S}}{\partial (D\mathbf{h}_m/D\boldsymbol{\phi})}\frac{D^2\mathbf{h}_m}{D\boldsymbol{\phi}^2} \tag{94}$$

where first order derivatives $D\mathbf{p}_m/D\boldsymbol{\phi}$, $D\mathbf{h}_m/D\boldsymbol{\phi}$ and second order derivatives $D^2\mathbf{p}_m/D\boldsymbol{\phi}^2$, $D^2\mathbf{h}_m/D\boldsymbol{\phi}^2$ are implicit and require differentiation of a complete path-following algorithm for the solution of selected micro problem. This can be done using analytical sensitivity analysis procedures, such as described in [27]. $\boldsymbol{\phi}$ represents input data for the selected micro level simulation and is used to calculate essential boundary conditions (89). Thus, for the evaluation of implicit derivatives, a boundary condition sensitivity analysis is needed with components of $\boldsymbol{\phi}$ as sensitivity parameters.

The solution of a path-dependent micro problem in general, depends on all variables transferred from the macro level to the micro level in all macro steps. Consequently, a complete set of sensitivity parameters of the selected micro problem would be composed of all variables transferred from the selected macro element to the selected micro problem. Sensitivity analysis for a large number of parameters requires significant amount of memory as well as computation time. But actually, there is no needed for it. The variables transferred in k-th step ($\boldsymbol{\phi} = \boldsymbol{\phi}_{k+1}$) affect the selected micro problem only from the micro step at the beginning of the macro step (micro step with the index s-th) and implicit derivatives with respect to $\boldsymbol{\phi}$ are not needed any more after the completion of the macro step. Consequently,

$$\frac{D\mathbf{p}_{mi}}{D\boldsymbol{\phi}} = \mathbf{0}, \frac{D\mathbf{h}_{mi}}{D\boldsymbol{\phi}} = \mathbf{0}, \frac{D^2\mathbf{p}_{mi}}{D\boldsymbol{\phi}^2} = \mathbf{0}, \frac{D^2\mathbf{h}_{mi}}{D\boldsymbol{\phi}^2} = \mathbf{0} : \forall\, i \leqslant s, \tag{95}$$

and implicit derivatives with respect to $\boldsymbol{\phi}$ do not appear in the macro problem after the completion of the macro step. Thus, at any given time, only a set of sensitivity parameters $\boldsymbol{\phi}$, that belongs to the current macro step, has to be considered, provided that (95) holds.

Since $\mathbf{Q}_{mg}$ depends only on $\mathbf{h}_{mgn}$, it is sufficient that (95) hold to set

$$\frac{D\mathbf{h}_{ms}}{D\boldsymbol{\phi}} = \mathbf{0} \quad \text{and} \quad \frac{D^2\mathbf{h}_{ms}}{D\boldsymbol{\phi}^2} = \mathbf{0} \tag{96}$$

at the start of each micro problem increment (at the s-th micro step). *AceGen* input segment for user subroutine of finite element that resets sensitivity data is presented in Appendix A.2, Step 6.1.

3.1.1 Two-level path-following algorithm

The algorithm that summarizes the above considerations is presented in Fig. 6. First, the micro level equations (91) and (92) are solved for unknown $\mathbf{p}_m$ and $\mathbf{h}_m$ at fixed $\mathbf{p}_M$ with the use of Newton method, which is also applied to solve the macro equilibrium equation (88) in an outer loop leading to a nested iteration-subiteration solution scheme for unknown $\mathbf{p}_M$, $\mathbf{p}_m$ and $\mathbf{h}_m$. For the sake of simplicity, the algorithm is written for the constant time stepping with n_M macro steps and n_m micro steps per macro step. However, it can be easily extended to an arbitrary adaptive time stepping scheme. It is assumed here that the micro problem is path-dependent, thus the state of all micro problems has to be stored somewhere at the end of the solution of each macro step and restored at the beginning of each macro iteration.

The basic idea of this thesis is that any FE code that supports first and second order sensitivity analysis can be turned into a fully consistent, numerically efficient, quadratically convergent nonlinear multi-scale code with minimal or even without any additional coding at the level of micro finite elements. Of course, one can use finite difference approximation to evaluate macro tangent modulus $\mathbf{K}_M$. However, the resulting code is numerically efficient and inexact tangent can affect the rate of convergence of iterative procedure. In any case, one has to write additional code for data management, solution of the macro problem and for the parallelization of the multi-scale algorithm. Let us assume that the code supports primal, first and second order sensitivity analysis. That is one of the requirements for the implementation of the particular multi-scale scheme to define the following quantities and expressions:

- micro problem sensitivity parameters as a function of macro element unknowns $(\boldsymbol{\phi}(\mathbf{p}_{Me}))$,

- boundary conditions of micro problem as a function of sensitivity parameters ($\bar{\mathbf{p}}_m(\boldsymbol{\phi}, \lambda_m)$),
- derivatives of boundary conditions with respect to sensitivity parameters ($\frac{D\bar{\mathbf{p}}_m(\boldsymbol{\phi},\lambda_m)}{D\boldsymbol{\phi}}$),
- micro level variables that are passed to macro level ($\mathbf{S}$),
- total derivative of micro level variables with respect to sensitivity parameters ($D\mathbf{S}/D\boldsymbol{\phi}$),
- macro element residual vector ($\mathbf{R}_{Me}(\mathbf{p}_{Me}, \mathbf{S}_{Me})$),
- macro element tangent matrix ($\mathbf{K}_{Me}(\mathbf{p}_{Me}, \mathbf{S}_{Me}, D\mathbf{S}_{Me}/D\boldsymbol{\phi}_{Me})$).

Initialization: $\lambda_M \leftarrow 0$; $\Delta\lambda_M = \bar{\lambda}_M/n_M$; $\mathbf{p}_{Mk} \leftarrow \mathbf{0}$; $\mathbf{p}_M \leftarrow \mathbf{0}$; **foreach** *micro problem* **set** $\mathbf{p}_{ms} \leftarrow \mathbf{0}; \mathbf{h}_{ms} \leftarrow \mathbf{0}; \bar{\mathbf{p}}_{ms} \leftarrow \mathbf{0}$;

for $i = 1$ **to** n_M **do** macro steps

- $\lambda_M \leftarrow \lambda_M + \Delta\lambda_M$
- **repeat** Newton iterations at macro level
  - **foreach** *macro element* **evaluate** $\boldsymbol{\phi}_{Me}$
  - **foreach** *micro problem* **do**
    - $\mathbf{p}_{mn} \leftarrow \mathbf{p}_{ms}$;$\mathbf{h}_{mn} \leftarrow \mathbf{h}_{ms}$ `// initialize primal data`
    - $D\mathbf{h}_{mn}/D\boldsymbol{\phi} \leftarrow \mathbf{0}$;$D^2\mathbf{h}_{mn}/D\boldsymbol{\phi}^2 \leftarrow \mathbf{0}$`// delete sensitivity history to fulfill` (95)
    - $\lambda_m \leftarrow i - 1$; $\Delta\lambda_m = 1/n_m$; $\Delta\bar{\mathbf{p}}_m \leftarrow \bar{\mathbf{p}}_m(\boldsymbol{\phi}) - \bar{\mathbf{p}}_{ms}$;
    - **for** $j = 1$ **to** n_m **do** micro steps
      - $\lambda_m \leftarrow \lambda_m + \Delta\lambda_m$; $\bar{\mathbf{p}}_m \leftarrow \bar{\mathbf{p}}_m + \Delta\lambda_m\,\Delta\bar{\mathbf{p}}_m$;
      - **solve** *one micro step*
        - $\mathbf{R}_m(\mathbf{p}_m, \mathbf{h}_m) = \mathbf{0}$; $\forall g : \mathbf{Q}_{mg}(\mathbf{p}_{me}, \mathbf{h}_{mg}, \mathbf{h}_{mgn}) = \mathbf{0}$ `// solve coupled primal problem for unknown` $\mathtt{p}_m, \mathtt{h}_m$
        - $D\mathbf{p}_m/D\boldsymbol{\phi}$; $D\mathbf{h}_m/D\boldsymbol{\phi}$ `// first order sensitivity problem`
        - $D^2\mathbf{p}_m/D\boldsymbol{\phi}^2$; $D^2\mathbf{h}_m/D\boldsymbol{\phi}^2$ `// second order sensitivity problem if needed`
      - $\mathbf{p}_{mn} \leftarrow \mathbf{p}_m$; $\mathbf{h}_{mn} \leftarrow \mathbf{h}_m$ `// update primal data`
      - $D\mathbf{p}_{mn}/D\boldsymbol{\phi} \leftarrow D\mathbf{p}_m/D\boldsymbol{\phi} \ldots D^2\mathbf{h}_{mn}/D\boldsymbol{\phi}^2 \leftarrow D^2\mathbf{h}_m/D\boldsymbol{\phi}^2$ `// update all sensitivity data`
    - evaluate and store $\mathbf{S}$ and $D\mathbf{S}/D\boldsymbol{\phi}$
  - **Newton update** *at macro level*
    - $\mathbf{R}_M = \bigwedge_e^{n_{Me}} \mathbf{R}_{Me}(\mathbf{p}_{Me}, \mathbf{S}_{Me})$;
    - $\mathbf{K}_M = \bigwedge_e^{n_{Me}} \mathbf{K}_{Me}(\mathbf{p}_{Me}, \mathbf{S}_{Me}, D\mathbf{S}_{Me}/D\boldsymbol{\phi}_{Me})$;
    - $\mathbf{p}_M \leftarrow \mathbf{p}_M - \mathbf{K}_M^{-1}\mathbf{R}_M$
- **until** $\mathbf{p}_M$ *has converged*
- $\mathbf{p}_{Mk} \leftarrow \mathbf{p}_M$;
- **foreach** *micro problem* **set** $\mathbf{p}_{ms} \leftarrow \mathbf{p}_m; \mathbf{h}_{ms} \leftarrow \mathbf{h}_m; \bar{\mathbf{p}}_{ms} \leftarrow \bar{\mathbf{p}}_m$; `// update and store macro and micro data`

Figure 6: Two-level path-following, multi-scale algorithm
Slika 6: Dvonivojsko sledenje poti, večnivojski algoritem

3.2 Primal analysis of micro problem

The primal problem at the micro level represents a system of GAUSS point coupled non-linear algebraic equations that can be solved using standard nested iteration-subiteration Newton-Raphson iterative procedure (see e.g. [27]). First, the GAUSS point equation (92) is solved for $\mathbf{h}_{mg}$ at fixed $\mathbf{p}_{me}$ with the use of Newton method, which is also applied to solve the equilibrium equation (91) in an outer loop for unknown $\mathbf{p}_m$. The linearization of the dependent residual (92) yields

$$\mathbf{K}_Q(\mathbf{h}_{mg}^{(j)})\Delta\mathbf{h}_{mg}^{(j)} + \mathbf{Q}_{mg}(\mathbf{h}_{mg}^{(j)}) = \mathbf{0} \tag{97}$$

where $\mathbf{K}_Q$ stands for the dependent tangent operator defined by

$$\mathbf{K}_Q = \frac{\partial \mathbf{Q}_{mg}(\mathbf{h}_{mg}^{(j)})}{\partial \mathbf{h}_{mg}} \,. \tag{98}$$

The linear system (97) is solved for the unknown increment $\Delta\mathbf{h}_{mg}^{(j)}$ and used to update the current dependent solution vector $\mathbf{h}_{mg}$

$$\mathbf{h}_{mg}^{(j+1)} = \mathbf{h}_{mg}^{(j)} + \Delta\mathbf{h}_{mg}^{(j)} \,. \tag{99}$$

After convergence of the inner loop is achieved, the linearization of the independent residual (91) yields

$$\mathbf{K}_m(\mathbf{p}_m^{(i)}, \mathbf{h}_m)\Delta\mathbf{p}_m^{(i)} + \mathbf{R}_m(\mathbf{p}_m^{(i)}, \mathbf{h}_m) = \mathbf{0} \tag{100}$$

where $\mathbf{K}_m$ stands for the independent tangent operator obtained by a standard finite element assembly

$$\mathbf{K}_m = \mathop{\mathsf{A}}_{e=1}^{n_e} \left(\sum_{g \in G_e} w_{gp} \mathbf{K}_{mg} \right) \,. \tag{101}$$

The independent solution vector is updated after the linear system (100) has been solved for the unknown increment $\Delta\mathbf{p}_{me}$ of the independent solution vector.

$$\mathbf{p}_{me}^{(i+1)} = \mathbf{p}_{me}^{(i)} + \Delta\mathbf{p}_{me}^{(i)} \,. \tag{102}$$

In order to achieve quadratic convergence, the tangent operator has to be evaluated consistently. The linearization of the residual yields the element tangent operator

$$\mathbf{K}_{mg} = \frac{\partial \mathbf{R}_{mg}}{\partial \mathbf{p}_{me}} + \frac{\partial \mathbf{R}_{mg}}{\partial \mathbf{h}_{mg}} \frac{D\mathbf{h}_{mg}}{D\mathbf{p}_{me}} \,. \tag{103}$$

The unknown derivative $\frac{D\mathbf{h}_{mg}}{D\mathbf{p}_{me}}$ in (103) is obtained by differentiating (92) with respect to $\mathbf{p}_{me}$

$$\mathbf{K}_Q \frac{D\mathbf{h}_{mg}}{D\mathbf{p}_{me}} = -\frac{\partial \mathbf{Q}_{mg}}{\partial \mathbf{p}_{me}}. \tag{104}$$

Inserting (104) into (103) yields the final form of the independent element tangent operator

$$\mathbf{K}_{mg} = \frac{\partial \mathbf{R}_{mg}}{\partial \mathbf{p}_{me}} - \frac{\partial \mathbf{R}_{mg}}{\partial \mathbf{h}_{mg}} \mathbf{K}_Q^{-1} \frac{\partial \mathbf{Q}_{mg}}{\partial \mathbf{p}_{me}}. \tag{105}$$

The tangent operator for the inner loop $\mathbf{K}_Q$ is in ADB form is given by

$$\mathbf{K}_Q = \frac{\hat{\delta}\mathbf{Q}_{mg}}{\hat{\delta}\mathbf{h}_{mg}} \tag{106}$$

and consistent micro tangent matrix $\mathbf{K}_m$ is written as

$$\mathbf{K}_m = \mathop{\mathsf{A}}_{e=1}^{n_{me}} \sum_{g \in G_e} w_{gp} \left. \frac{\hat{\delta}\mathbf{R}_{mg}}{\hat{\delta}\mathbf{p}_{me}} \right|_{\frac{D\mathbf{h}_{mg}}{D\mathbf{p}_{me}} = -\mathbf{K}_Q^{-1} \frac{\hat{\delta}\mathbf{Q}_{mg}}{\hat{\delta}\mathbf{p}_{me}}}. \tag{107}$$

Thus the evaluation of the micro tangent matrix (107) requires proper consideration of the implicit dependency $\mathbf{h}_{mg}(\mathbf{p}_{me})$ introduced by the local iterative procedure.

3.3 Sensitivity analysis of micro problem

The aim of the sensitivity analysis is to calculate derivatives (sensitivities) of an arbitrary response functional F with respect to chosen parameters $\boldsymbol{\phi}$. Lets consider a simple example presented in Fig. 7 where an elastic bar of length L with the cross section area A and elastic modulus E is subjected to force P at the end. The goal of simulation is to minimize the response functional $F(A) = (\sigma(A) - \sigma_y)^2$ where $\sigma = E\epsilon = E\frac{u(A)}{L}$, u is an unknown tip displacement and σ_y is a yield stress. The primal problem is defined by equilibrium equation $\frac{EA}{L}u - P = 0$. Solution (also response) of the primal problem is $u = \frac{PL}{EA}$. Solution of the primal problem depends on the shape parameter L, material parameters E and A, and prescribed natural boundary parameter P. The sensitivity of the response of the primal problem with respect to L is called shape sensitivity and is defined by $\frac{\partial u}{\partial L} = \frac{L}{EA}$, sensitivity with respect to material parameters is called parameter sensitivity ($\frac{\partial u}{\partial E} = -\frac{PL}{E^2A}$, $\frac{\partial u}{\partial A} = -\frac{PL}{EA^2}$) and with respect to force P prescribed natural boundary condition sensitivity ($\frac{\partial u}{\partial P} = \frac{L}{EA}$). The goal is to minimize the response functional F with respect to cross section area A, thus only sensitivity of response with respect to A is

needed ($\boldsymbol{\phi} = \{A\}$). Gradient of response functional is $\frac{\partial F}{\partial A} = 2(\sigma - \sigma_y)\frac{E}{L}\frac{\partial u}{\partial A}$. The optimality condition $\frac{\partial F}{\partial A} = 0$ then yields the well known solution of the problem $A = P/\sigma_y$.

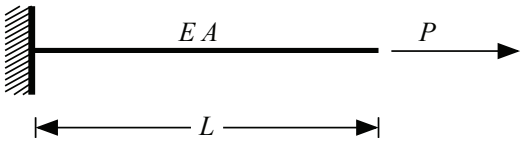


Figure 7: Stretching of elastic bar
Slika 7: Raztezanje elastične palice

The above example can be extended to arbitrary nonlinear coupled path-dependent problems, such as finite strain plasticity. In first order sensitivity analysis, the first order derivatives $\frac{\partial F}{\partial \phi_I}$ and in second order sensitivity analysis, the second order derivatives $\frac{\partial^2 F}{\partial \phi_I \partial \phi_J}$ are sought. Response functional in general depends on the solution vectors $(\mathbf{p}_M, \mathbf{p}_m)$. Therefore, the solution vectors have to be calculated first. For the sensitivity problem, we define the residuals, vectors of unknowns and response functional as a function of a vector of design parameters $\boldsymbol{\phi} = \{\phi_I : I = 1, \ldots, n_\phi\}$ where n_ϕ is the total number of sensitivity parameters. The solution algorithm of the sensitivity problem can then be obtained from the primal problem by differentiating the response functional and the residuals with respect to design parameters. As shown later, to be able to calculate second order derivatives, first order derivatives have to be calculated first.

3.3.1 Essential boundary condition sensitivity analysis

For the essential boundary condition sensitivity analysis we define the residuals and the vectors of unknowns in (91) and (92) as a function of sensitivity parameters $\boldsymbol{\phi}$ by

$$\begin{aligned}&\mathbf{R}_m(\mathbf{p}_m(\boldsymbol{\phi}), \mathbf{h}_m(\boldsymbol{\phi}), \bar{\mathbf{p}}_m(\boldsymbol{\phi}, \lambda_m))\\ &= \mathop{\mathsf{A}}_{e=1}^{n_{me}} \sum_{g \in G_e} w_{gp}\, \mathbf{R}_{mg}(\check{\mathbf{p}}_{me}(\boldsymbol{\phi}), \bar{\mathbf{p}}_{me}(\boldsymbol{\phi}, \lambda_m), \mathbf{h}_{mg}(\boldsymbol{\phi})) = \mathbf{0}, \end{aligned} \tag{108}$$

$$\mathbf{Q}_{mg}(\check{\mathbf{p}}_{me}(\boldsymbol{\phi}), \bar{\mathbf{p}}_{me}(\boldsymbol{\phi}, \lambda_m), \mathbf{h}_{mg}(\boldsymbol{\phi}), \mathbf{h}_{mgn}(\boldsymbol{\phi})) = \mathbf{0} : g = 1, 2, ...n_{tg}. \tag{109}$$

where $\bar{\mathbf{p}}_m(\boldsymbol{\phi}, \lambda_m)$ is a set of nodal DOF with prescribed essential boundary conditions defined by (89).

At the level of individual finite element, the set of nodal unknowns $\mathbf{p}_{me} = \check{\mathbf{p}}_{me} \cup \bar{\mathbf{p}}_{me}$

includes degrees of freedom with prescribed essential boundary condition $\bar{\mathbf{p}}_{me}$ and true degrees of freedom $\check{\mathbf{p}}_{me}$. At the element level they are indistinguishable. The first order sensitivity problem can be obtained from the primal problem by differentiating equations (108) and (109) with respect to sensitivity parameters, which results in

$$\frac{D\mathbf{R}_m}{D\phi_I} = \mathop{\mathsf{A}}_{e=1}^{n_{me}} \sum_{g\in G_e} w_{gp} \left(\frac{\partial \mathbf{R}_{mg}}{\partial \check{\mathbf{p}}_{me}} \frac{D\check{\mathbf{p}}_{me}}{D\phi_I} + \frac{\partial \mathbf{R}_{mg}}{\partial \bar{\mathbf{p}}_{me}} \frac{D\bar{\mathbf{p}}_{me}}{D\phi_I} + \frac{\partial \mathbf{R}_{mg}}{\partial \mathbf{h}_{mg}} \frac{D\mathbf{h}_{mg}}{D\phi_I} \right) = 0, \tag{110}$$

$$\frac{D\mathbf{Q}_{mg}}{D\phi_I} = \frac{\partial \mathbf{Q}_{mg}}{\partial \check{\mathbf{p}}_{me}} \frac{D\check{\mathbf{p}}_{me}}{D\phi_I} + \frac{\partial \mathbf{Q}_{mg}}{\partial \bar{\mathbf{p}}_{me}} \frac{D\bar{\mathbf{p}}_{me}}{D\phi_I} + \frac{\partial \mathbf{Q}_{mg}}{\partial \mathbf{h}_{mg}} \frac{D\mathbf{h}_{mg}}{D\phi_I} + \frac{\partial \mathbf{Q}_{mg}}{\partial \mathbf{h}_{mgn}} \frac{D\mathbf{h}_{mgn}}{D\phi_I} = 0. \tag{111}$$

To calculate $\frac{D\mathbf{p}_m}{D\phi_I}$, the sensitivities $\frac{D\mathbf{h}_{mg}}{D\phi_I}$ are expressed from equation (111) and inserted into equation (110). After rearrangement, in which the terms that contain the unknown sensitivity $\frac{D\mathbf{p}_m}{D\phi_I}$ are collected together, a system of linear equations is obtained

$$\mathbf{K}_m \frac{D\mathbf{p}_m}{D\phi_I} = -{}^I\tilde{\mathbf{R}}_m, \; {}^I\tilde{\mathbf{R}}_m = \mathop{\mathsf{A}}_{e=1}^{n_{me}} \sum_{g\in G_e} w_{gp} {}^I\tilde{\mathbf{R}}_{mg}. \tag{112}$$

Tangent matrix $\mathbf{K}_m$ is already evaluated and factorized from the primal problem. Therefore, only vector ${}^I\tilde{\mathbf{R}}_m$ on the right-hand side of equation (112) has to be calculated in order to obtain the resulting system of linear equations. This vector is called *independent first-order sensitivity pseudo-load vector*. After obtaining $\frac{D\mathbf{p}_m}{D\phi_I}$ as the solution of (112), the obtained values are inserted into equation (111) and $\frac{D\mathbf{h}_{mg}}{D\phi_I}$, $g = 1, 2, ...n_{tg}$ can be expressed. Corresponding expressions are

$${}^I\mathbf{Z}_g = -\mathbf{K}_Q^{-1} \left(\frac{\partial \mathbf{Q}_{mg}}{\partial \bar{\mathbf{p}}_{me}} \frac{D\bar{\mathbf{p}}_{me}}{D\phi_I} + \frac{\partial \mathbf{Q}_{mg}}{\partial \mathbf{h}_{mgn}} \frac{D\mathbf{h}_{mgn}}{D\phi_I} \right) \tag{113}$$

$${}^I\tilde{\mathbf{R}}_{mg} = \frac{\partial \mathbf{R}_{mg}}{\partial \bar{\mathbf{p}}_{me}} \frac{D\bar{\mathbf{p}}_{me}}{D\phi_I} + \frac{\partial \mathbf{R}_{mg}}{\partial \mathbf{h}_{mg}} {}^I\mathbf{Z}_g \tag{114}$$

$$\frac{D\mathbf{h}_{mg}}{D\phi_I} = {}^I\mathbf{Z}_g - \mathbf{K}_Q^{-1} \left(\frac{\partial \mathbf{Q}_{mg}}{\partial \check{\mathbf{p}}_{me}} \frac{D\check{\mathbf{p}}_{me}}{D\phi_I} \right) \tag{115}$$

where ${}^I\mathbf{Z}_g$ is an additional auxiliary variable introduced to increase numerical efficiency. It can be evaluated during the evaluation of ${}^I\tilde{\mathbf{R}}_{mg}$, stored in memory and used later for the evaluation of $D\mathbf{h}_{mg}/D\phi_I$.

Function $\bar{\mathbf{p}}_m(\boldsymbol{\phi}, \lambda_m)$ can be arbitrary complex and, in general, cannot be the input data of the finite element analysis. Anyhow, it is not the relation $\bar{\mathbf{p}}_m(\boldsymbol{\phi}, \lambda_m)$ itself that is needed within the sensitivity analysis, but its first and second derivatives. Let ϕ_I and ϕ_J be an arbitrary essential boundary condition sensitivity parameters. The rate of change

of essential boundary conditions ${}^{I}\mathbf{V} = \frac{\partial \bar{\mathbf{p}}_m(\boldsymbol{\Phi},\lambda_m)}{\partial \phi_I}$ and ${}^{IJ}\mathbf{V} = \frac{\partial^2 \bar{\mathbf{p}}_m(\boldsymbol{\Phi},\lambda_m)}{\partial \phi_I \partial \phi_J}$ are called first and second order essential boundary condition velocity fields. The values of first and second order essential boundary condition velocity fields at the nodes of e-th element are defined by

$$ {}^{I}\mathbf{V}_e = \begin{cases} \frac{\partial \bar{p}_{mei}}{\partial \phi_I} & \text{if } p_{mei} \in \bar{\mathbf{p}}_{me} \\ 0 & \text{if } p_{mei} \in \check{\mathbf{p}}_{me} \end{cases} : i = 1, \ldots, n_p, \tag{116} $$

$$ {}^{IJ}\mathbf{V}_e = \begin{cases} \frac{\partial^2 \bar{p}_{mei}}{\partial \phi_I \partial \phi_J} & \text{if } p_{mei} \in \bar{\mathbf{p}}_{me} \\ 0 & \text{if } p_{mei} \in \check{\mathbf{p}}_{me} \end{cases} : i = 1, \ldots, n_p \tag{117} $$

where n_p is the total number of element nodal DOFs. The velocity field is zero for the true degrees of freedom. We conclude that the proper definition of element velocity fields is sufficient to distinguish the difference between the degrees of freedom with prescribed essential boundary condition and true degrees of freedom at the finite element level.

The actual analytical expressions for (113), (114) and (115) are rather lengthy. To simplify the process they can be obtained automatically by using the automatic differentiation. For this purpose, an automatic differentiation based notation or ADB notation of the terms is needed. A general ADB notation of the first order terms follows from (113), (114), and (115) where all implicit derivatives are replaced by appropriate AD exceptions resulting in

$$ {}^{I}\mathbf{Z}_g = -\mathbf{K}_Q^{-1} \frac{\hat{\delta}\mathbf{Q}_{mg}}{\hat{\delta}\phi_I}\bigg|_{\frac{D\mathbf{p}_{me}}{D\phi_I} = {}^{I}\mathbf{V}_e, \frac{D\mathbf{h}_{mgn}}{D\phi_I} = {}^{I}\mathbf{H}_g^n,} \tag{118} $$

$$ {}^{I}\tilde{\mathbf{R}}_{mg} = \frac{\hat{\delta}\mathbf{R}_{mg}}{\hat{\delta}\phi_I}\bigg|_{\frac{D\mathbf{p}_{me}}{D\phi_I} = {}^{I}\mathbf{V}_e, \frac{D\mathbf{h}_{mg}}{D\phi_I} = {}^{I}\mathbf{Z}_g,} \tag{119} $$

$$ \frac{D\mathbf{h}_{mg}}{D\phi_I} = {}^{I}\mathbf{Z}_g - \mathbf{K}_Q^{-1} \frac{\hat{\delta}\mathbf{Q}_{mg}}{\hat{\delta}\phi_I}\bigg|_{\frac{D\mathbf{p}_{me}}{D\phi_I} = \begin{cases} 0 & \text{if } p_{mei} \in \bar{\mathbf{p}}_{me} \\ {}^{I}Y_i & \text{if } p_{mei} \in \check{\mathbf{p}}_{me} \end{cases}_{:i=1,\ldots,n_p}} . \tag{120} $$

${}^{I}\mathbf{Y} = \frac{D\mathbf{p}_{me}}{D\phi_I}$ are already calculated and stored sensitivities of nodal unknowns and ${}^{I}\mathbf{H}_g^n = \frac{D\mathbf{h}_{mgn}}{D\phi_I}$ are sensitivities of integration point unknowns at the end of last micro step.

For the calculation of the unknown second order sensitivities $\frac{D^2\mathbf{p}_m}{D\phi_I D\phi_J}$ and $\frac{D^2\mathbf{h}_{mg}}{D\phi_I D\phi_J}$, $\mathbf{R}_m$ and $\mathbf{Q}_{mg}$ have to be differentiated twice, with respect to ϕ_I and ϕ_J. After differentiating

(110) and (111) with respect to ϕ_J we get

$$\frac{D^2\mathbf{R}_m}{D\phi_I D\phi_J} = \mathop{\mathbf{A}}_{e=1}^{n_{me}} \sum_{g\in G_e} w_{gp}\Big(\frac{\partial^2\mathbf{R}_{mg}}{\partial\check{\mathbf{p}}_{me}^2}\frac{D\check{\mathbf{p}}_{me}}{D\phi_I}\frac{D\check{\mathbf{p}}_{me}}{D\phi_J} + \frac{\partial\mathbf{R}_{mg}}{\partial\check{\mathbf{p}}_{me}}\frac{D^2\check{\mathbf{p}}_{me}}{D\phi_I D\phi_J} + \frac{\partial^2\mathbf{R}_{mg}}{\partial\bar{\mathbf{p}}_{me}^2}\frac{D\bar{\mathbf{p}}_{me}}{D\phi_I}\frac{D\bar{\mathbf{p}}_{me}}{D\phi_J} + \frac{\partial\mathbf{R}_{mg}}{\partial\bar{\mathbf{p}}_{me}}\frac{D^2\bar{\mathbf{p}}_{me}}{D\phi_I D\phi_J} + \frac{\partial^2\mathbf{R}_{mg}}{\partial\mathbf{h}_{mg}{}^2}\frac{D\mathbf{h}_{mg}}{D\phi_I}\frac{D\mathbf{h}_{mg}}{D\phi_J} + \frac{\partial\mathbf{R}_{mg}}{\partial\mathbf{h}_{mg}}\frac{D^2\mathbf{h}_{mg}}{D\phi_I D\phi_J}\Big) = 0, \tag{121}$$

$$\begin{aligned}\frac{D^2\mathbf{Q}_{mg}}{D\phi_I D\phi_J} &= \frac{\partial^2\mathbf{Q}_{mg}}{\partial\check{\mathbf{p}}_{me}^2}\frac{D\check{\mathbf{p}}_{me}}{D\phi_I}\frac{D\check{\mathbf{p}}_{me}}{D\phi_J} + \frac{\partial\mathbf{Q}_{mg}}{\partial\check{\mathbf{p}}_{me}}\frac{D^2\check{\mathbf{p}}_{me}}{D\phi_I D\phi_J} + \frac{\partial^2\mathbf{Q}_{mg}}{\partial\bar{\mathbf{p}}_{me}^2}\frac{D\bar{\mathbf{p}}_{me}}{D\phi_I}\frac{D\bar{\mathbf{p}}_{me}}{D\phi_J} \\ &+ \frac{\partial\mathbf{Q}_{mg}}{\partial\bar{\mathbf{p}}_{me}}\frac{D^2\bar{\mathbf{p}}_{me}}{D\phi_I D\phi_J} + \frac{\partial^2\mathbf{Q}_{mg}}{\partial\mathbf{h}_{mg}{}^2}\frac{D\mathbf{h}_{mg}}{D\phi_I}\frac{D\mathbf{h}_{mg}}{D\phi_J} + \frac{\partial\mathbf{Q}_{mg}}{\partial\mathbf{h}_{mg}}\frac{D^2\mathbf{h}_{mg}}{D\phi_I D\phi_J} \\ &+ \frac{\partial^2\mathbf{Q}_{mg}}{\partial\mathbf{h}_{mgn}{}^2}\frac{D\mathbf{h}_{mgn}}{D\phi_I}\frac{D\mathbf{h}_{mgn}}{D\phi_J} + \frac{\partial\mathbf{Q}_{mg}}{\partial\mathbf{h}_{mgn}}\frac{D^2\mathbf{h}_{mgn}}{D\phi_I D\phi_J} = \mathbf{0}.\end{aligned} \tag{122}$$

To calculate $\frac{D^2\mathbf{p}_m}{D\phi_I D\phi_J}$, the sensitivities $\frac{D^2\mathbf{h}_{mg}}{D\phi_I D\phi_J}$ are expressed from equation (122) and inserted into equation (121). A procedure is equivalent to the one for the first order sensitivity analysis and after rearrangement, in which the terms that contain the unknown sensitivity $\frac{D^2\mathbf{p}_m}{D\phi_I D\phi_J}$ are collected together, a system of linear equations is obtained

$$\mathbf{K}_m\frac{D^2\mathbf{p}_m}{D\phi_I D\phi_J} = -{}^{IJ}\tilde{\mathbf{R}}_m,\ {}^{IJ}\tilde{\mathbf{R}}_m = \mathop{\mathbf{A}}_{e=1}^{n_{me}}\sum_{g\in G_e} w_{gp}\,{}^{IJ}\tilde{\mathbf{R}}_{mg}, \tag{123}$$

where again only vector ${}^{IJ}\tilde{\mathbf{R}}_m$ on the right-hand side of equation (123) has to be calculated. This vector is called *independent second-order sensitivity pseudo-load vector*. After obtaining $\frac{D^2\mathbf{p}_m}{D\phi_I D\phi_J}$, the derivatives $\frac{D^2\mathbf{h}_{mg}}{D\phi_I D\phi_J}$, $g = 1, 2, ...n_{tg}$ can be expressed. Corresponding expressions are

$$\begin{aligned}{}^{IJ}\mathbf{Z}_g = -\mathbf{K}_Q^{-1}\Big(&\frac{\partial^2\mathbf{Q}_{mg}}{\partial\check{\mathbf{p}}_{me}^2}\frac{D\check{\mathbf{p}}_{me}}{D\phi_I}\frac{D\check{\mathbf{p}}_{me}}{D\phi_J} + \frac{\partial^2\mathbf{Q}_{mg}}{\partial\bar{\mathbf{p}}_{me}^2}\frac{D\bar{\mathbf{p}}_{me}}{D\phi_I}\frac{D\bar{\mathbf{p}}_{me}}{D\phi_J} + \frac{\partial\mathbf{Q}_{mg}}{\partial\bar{\mathbf{p}}_{me}}\frac{D^2\bar{\mathbf{p}}_{me}}{D\phi_I D\phi_J} \\ &+ \frac{\partial^2\mathbf{Q}_{mg}}{\partial\mathbf{h}_{mg}{}^2}\frac{D\mathbf{h}_{mg}}{D\phi_I}\frac{D\mathbf{h}_{mg}}{D\phi_J} + \frac{\partial^2\mathbf{Q}_{mg}}{\partial\mathbf{h}_{mgn}{}^2}\frac{D\mathbf{h}_{mgn}}{D\phi_I}\frac{D\mathbf{h}_{mgn}}{D\phi_J} + \frac{\partial\mathbf{Q}_{mg}}{\partial\mathbf{h}_{mgn}}\frac{D^2\mathbf{h}_{mgn}}{D\phi_I D\phi_J}\Big)\end{aligned} \tag{124}$$

$$\begin{aligned}{}^{IJ}\tilde{\mathbf{R}}_{mg} = &\frac{\partial^2\mathbf{R}_{mg}}{\partial\check{\mathbf{p}}_{me}^2}\frac{D\check{\mathbf{p}}_{me}}{D\phi_I}\frac{D\check{\mathbf{p}}_{me}}{D\phi_J} + \frac{\partial^2\mathbf{R}_{mg}}{\partial\bar{\mathbf{p}}_{me}^2}\frac{D\bar{\mathbf{p}}_{me}}{D\phi_I}\frac{D\bar{\mathbf{p}}_{me}}{D\phi_J} + \frac{\partial\mathbf{R}_{mg}}{\partial\bar{\mathbf{p}}_{me}}\frac{D^2\bar{\mathbf{p}}_{me}}{D\phi_I D\phi_J} \\ &+ \frac{\partial^2\mathbf{R}_{mg}}{\partial\mathbf{h}_{mg}{}^2}\frac{D\mathbf{h}_{mg}}{D\phi_I}\frac{D\mathbf{h}_{mg}}{D\phi_J} + \frac{\partial\mathbf{R}_{mg}}{\partial\mathbf{h}_{mg}}{}^{IJ}\mathbf{Z}_g\end{aligned} \tag{125}$$

$$\frac{D^2\mathbf{h}_{mg}}{D\phi_I D\phi_J} = {}^{IJ}\mathbf{Z}_g - \mathbf{K}_Q^{-1}\left(\frac{\partial \mathbf{Q}_{mg}}{\partial \breve{\mathbf{p}}_{me}}\frac{D^2\breve{\mathbf{p}}_{me}}{D\phi_I D\phi_J}\right) \tag{126}$$

where ${}^{IJ}\mathbf{Z}_g$ is an additional auxiliary variable introduced to increase numerical efficiency. It can be calculated during the evaluation of ${}^{IJ}\tilde{\mathbf{R}}_{mg}$, stored in memory and used later for the evaluation of $\frac{D^2\mathbf{h}_{mg}}{D\phi_I D\phi_J}$.

Automatic differentiation based notation of ${}^{IJ}\tilde{\mathbf{R}}_{mg}$ and $\frac{D^2\mathbf{h}_{mg}}{D\phi_I D\phi_J}$ is given here by

$${}^{IJ}\mathbf{Z}_g = -\mathbf{K}_Q^{-1}\frac{\hat{\delta}}{\hat{\delta}\phi_J}\left(\left.\frac{\hat{\delta}\mathbf{Q}_{mg}}{\hat{\delta}\phi_I}\right|_{\substack{\frac{D\mathbf{p}_{me}}{D\phi_I}={}^{I}\mathbf{Y},\\ \frac{D\mathbf{h}_{mg}}{D\phi_I}={}^{I}\mathbf{H}_g,\frac{D\mathbf{h}_{mgn}}{D\phi_I}={}^{I}\mathbf{H}_g^n}}\right)\Bigg|_{\substack{\frac{D\mathbf{p}_{me}}{D\phi_J}={}^{J}\mathbf{Y},\frac{D^I\mathbf{Y}}{D\phi_J}={}^{IJ}\mathbf{V}_e,\\ \frac{D\mathbf{h}_{mgn}}{D\phi_J}={}^{J}\mathbf{H}_g^n,\frac{D^I\mathbf{H}_g^n}{D\phi_J}={}^{IJ}\mathbf{H}_g^n}} , \tag{127}$$

$${}^{IJ}\tilde{\mathbf{R}}_{mg} = \frac{\hat{\delta}}{\hat{\delta}\phi_J}\left(\left.\frac{\hat{\delta}\mathbf{R}_{mg}}{\hat{\delta}\phi_I}\right|_{\substack{\frac{D\mathbf{p}_{me}}{D\phi_I}={}^{I}\mathbf{Y},\\ \frac{D\mathbf{h}_{mg}}{D\phi_I}={}^{I}\mathbf{H}_g}}\right)\Bigg|_{\substack{\frac{D^I\mathbf{Y}}{D\phi_J}={}^{IJ}\mathbf{V}_e,\frac{D\mathbf{p}_{me}}{D\phi_J}={}^{J}\mathbf{Y},\\ \frac{D^I\mathbf{H}_g}{D\phi_J}={}^{IJ}\mathbf{Z}_g}} , \tag{128}$$

$$\frac{D^2\mathbf{h}_{mg}}{D\phi_I D\phi_J} = {}^{IJ}\mathbf{Z}_g - \mathbf{K}_Q^{-1}\left.\frac{\hat{\delta}\mathbf{Q}_{mg}}{\hat{\delta}\phi_{IJ}}\right|_{\frac{D\mathbf{p}_{me}}{D\phi_{IJ}}=\begin{cases}0 & \text{if } p_{ei}\in\bar{\mathbf{p}}_{me}\\ {}^{IJ}Y_i & \text{if } p_{ei}\in\mathbf{p}_{me}\backslash\bar{\mathbf{p}}_{me}\end{cases}:i=1,\ldots,n_p} . \tag{129}$$

Additional intermediate quantities ${}^{I}\mathbf{Z}_g$ and ${}^{IJ}\mathbf{Z}_g$ are again evaluated during the evaluation of ${}^{I}\tilde{\mathbf{R}}_{mg}$ and ${}^{IJ}\tilde{\mathbf{R}}_{mg}$, stored in memory and used later for the evaluation of $D\mathbf{h}_{mg}/D\phi_I$ and $D^2\mathbf{h}_{mg}/D\phi_I D\phi_J$. Matrices ${}^{I}\mathbf{Y} = \frac{D\mathbf{p}_{me}}{D\phi_I}$, ${}^{J}\mathbf{Y} = \frac{D\mathbf{p}_{me}}{D\phi_J}$, ${}^{IJ}\mathbf{Y} = \frac{D^2\mathbf{p}_{me}}{D\phi_I D\phi_J}$, ${}^{I}\mathbf{H}_g = \frac{D\mathbf{h}_{mg}}{D\phi_I}$, ${}^{J}\mathbf{H}_g = \frac{D\mathbf{h}_{mg}}{D\phi_J}$, ${}^{I}\mathbf{H}_g^n = \frac{D\mathbf{h}_{mgn}}{D\phi_I}$, ${}^{J}\mathbf{H}_g^n = \frac{D\mathbf{h}_{mgn}}{D\phi_J}$, ${}^{IJ}\mathbf{H}_g^n = \frac{D^2\mathbf{h}_{mgn}}{D\phi_I D\phi_J}$ are already calculated and stored as first and second order sensitivities.

All first order sensitivities have to be calculated before the algorithm can calculate the second order sensitivities. For this reason, in the algorithm in Fig. 6, the second order sensitivity analysis is performed after the first order sensitivity analysis. For the implementation of multi-scale schemes it should be noted that the only multi-scale scheme dependent expressions in equations (118) to (129) are velocity fields ${}^{I}\mathbf{V}_e$ and ${}^{IJ}\mathbf{V}_e$.

Input:

Primal analysis: material data, displacement in macro element nodes $\mathbf{p}_{Me}$

Sensitivity analysis: sensitivity parameters $\boldsymbol{\phi}$, velocity fields ${}^{I}\mathbf{V}_{e}$

$\mathbf{p}_{me} = \check{\mathbf{p}}_{me} \cup \bar{\mathbf{p}}_{me}$, $\bar{\mathbf{p}}_{m}(\boldsymbol{\phi}, \lambda_m)$ `// vector of unknowns at the boundary of micro problems`

begin Solution of independent first order sensitivity problem

- $\boldsymbol{\phi} \leftarrow \mathbf{p}_{Me}$ `// set sensitivity parameters` $\boldsymbol{\phi}$
- ${}^{I}\mathbf{V}_{e} \leftarrow \{\frac{\partial \bar{\mathbf{p}}_{mei}}{\partial \phi_I}$ if $\mathbf{p}_{mei} \in \bar{\mathbf{p}}_{me}$; 0 if $\mathbf{p}_{mei} \in \check{\mathbf{p}}_{me} : i = 1, \ldots, n_p\}$ `// set BC velocity field`
- $\mathbf{p}_{me} \leftarrow \mathbf{p}_{me} | \frac{D\mathbf{p}_{me}}{D\phi_I} = {}^{I}\mathbf{V}_{e}$ `// set AD exception`
- **foreach** *element in micro problem* **do**
  - **foreach** *Gauss point* **do**
    - $\mathbf{h}_{mgn} \leftarrow \mathbf{h}_{mgn} | \frac{D\mathbf{h}_{mgn}}{D\phi_I} = {}^{I}\mathbf{H}_{g}^{n}$; $\mathbf{K}_Q \leftarrow \frac{\hat{\delta}\mathbf{Q}_{mg}}{\delta \mathbf{h}_{mg}}$ `// LU decomposition of` K_Q
    - **foreach** *sensitivity parameter* ϕ_I **do**
      - ${}^{I}\mathbf{Z}_{g} \leftarrow -\mathbf{K}_{Q}^{-1} \frac{\hat{\delta}\mathbf{Q}_{mg}}{\hat{\delta}\phi_I}$ `// Gauss elimination procedure`
      - ${}^{I}\tilde{\mathbf{R}}_{mg} \leftarrow \frac{\delta \mathbf{R}_{mg}}{\hat{\delta}\phi_I} | \frac{D\mathbf{h}_{mg}}{D\phi_I} = {}^{I}\mathbf{Z}_{g}$
      - export ${}^{I}\mathbf{Z}_{g} \rightarrow {}^{I}\mathbf{H}_{g}$
    - **end foreach**
    - add $w_{gp}\,{}^{I}\tilde{\mathbf{R}}_{mg}$ to ${}^{I}\tilde{\mathbf{R}}_{m}$
  - **end foreach**
- **end foreach**
- solve $\mathbf{K}_m \frac{D\mathbf{p}_m}{D\phi_I} + {}^{I}\tilde{\mathbf{R}}_{m} = 0$ for unknown $\frac{D\mathbf{p}_m}{D\phi_I}$, using $\mathbf{K}_m$ from primal analysis and store it into ${}^{I}\mathbf{Y}$

begin Solution of dependent first order sensitivity problem

- Evaluation is only required for coupled case, for uncoupled ${}^{I}\mathbf{H}_{g} = {}^{I}\mathbf{Z}_{g}$
- **foreach** *element in micro problem* **do**
  - $\mathbf{p}_{me} \leftarrow \mathbf{p}_{me} | \frac{D\mathbf{p}_{me}}{D\phi_I} = \{0$ if $\mathbf{p}_{mei} \in \bar{\mathbf{p}}_{me}$; ${}^{I}\mathbf{Y}_{i}$ if $\mathbf{p}_{mei} \in \check{\mathbf{p}}_{me} : i = 1, \ldots, n_p\}$
  - **foreach** *Gauss point* **do**
    - **foreach** *sensitivity parameter* ϕ_I **do**
      - ${}^{I}\mathbf{Z}_{g} \leftarrow {}^{I}\mathbf{H}_{g}$; $\frac{D\mathbf{h}_{mg}}{D\phi_I} \leftarrow {}^{I}\mathbf{Z}_{g} - \mathbf{K}_{Q}^{-1} \frac{\hat{\delta}\mathbf{Q}_{mg}}{\hat{\delta}\phi_I}$
      - export $\frac{D\mathbf{h}_{mg}}{D\phi_I} \rightarrow {}^{I}\mathbf{H}_{g}$
    - **end foreach**
  - **end foreach**
- **end foreach**

Figure 8: Algorithm for first order sensitivity analysis for locally coupled path-dependent problems

Slika 8: Algoritem za občutljivostno analizo prvega reda za lokalno povezane probleme odvisne od poti

In Fig. 8, procedure for solving first order sensitivity is shown schematically for path-dependent Gauss point locally coupled problems. Sensitivity analysis is done after successful primal analysis. First the independent sensitivity problem is solved and then the dependent sensitivity problem. In Fig. 9, procedure for solving second order sensiti-

vity is shown schematically. *AceGen* input segment for first and second order sensitivity analysis is presented in Appendix A.2 for hyper-elastic micro finite element.

begin Solution of independent second order sensitivity problem

$\phi \leftarrow \mathbf{p}_{Me}$ `// set sensitivity parameters` ϕ

${}^{IJ}\mathbf{V}_e \leftarrow \{\frac{\partial^2 \bar{\mathbf{p}}_{mei}}{\partial\phi_I \partial\phi_J}$ if $\mathbf{p}_{mei} \in \bar{\mathbf{p}}_{me}$; 0 if $\mathbf{p}_{mei} \in \check{\mathbf{p}}_{me} : i = 1, \ldots, n_p\}$ `// set BC velocity field`

${}^{I}\mathbf{Y} \wedge {}^{J}\mathbf{Y} \leftarrow \{\frac{\partial \bar{\mathbf{p}}_{mei}}{\partial\phi_I}$ if $\mathbf{p}_{mei} \in \bar{\mathbf{p}}_{me}$; $\frac{\partial \mathbf{p}_{mei}}{\partial\phi_I}$ if $\mathbf{p}_{mei} \in \check{\mathbf{p}}_{me} : J = 1, \ldots, n_p\}$`// first order sensitivities`

${}^{I}\mathbf{Y} \leftarrow {}^{I}\mathbf{Y}\big|_{\frac{D^I \mathbf{Y}}{D\phi_J} = {}^{IJ}\mathbf{V}_e}$; $\mathbf{p}_{me} \leftarrow \mathbf{p}_{me}\big|_{\frac{D\mathbf{p}_{me}}{D\phi_I} = {}^{I}\mathbf{Y}}$ `// set AD exception`

foreach *element in micro problem* **do**

  foreach *Gauss point* **do**

    ${}^{I}\mathbf{H}_g \wedge {}^{J}\mathbf{H}_g$ `// first order sensitivities`

    ${}^{I}\mathbf{H}_g^n \leftarrow {}^{I}\mathbf{H}_g^n\big|_{\frac{D^I \mathbf{H}_g^n}{D\phi_J} = {}^{IJ}\mathbf{H}_g^n}$; $\mathbf{h}_{mgn} \leftarrow \mathbf{h}_{mgn}\big|_{\frac{D\mathbf{h}_{mgn}}{D\phi_I} = {}^{I}\mathbf{H}_g^n}$

    $\mathbf{K}_Q \leftarrow \frac{\hat{\delta}\mathbf{Q}_{mg}}{\hat{\delta}\mathbf{h}_{mg}}$ `// LU decomposition of` K_Q

    foreach *couple of sensitivity parameters* $\phi_I \phi_J$ **do**

      ${}^{IJ}\mathbf{Z}_g \leftarrow -\mathbf{K}_Q^{-1} \frac{\hat{\delta}}{\hat{\delta}\phi_J}(\frac{\hat{\delta}\mathbf{Q}_{mg}}{\hat{\delta}\phi_I})$ `// Gauss elimination procedure`

      ${}^{IJ}\tilde{\mathbf{R}}_{mg} \leftarrow \frac{\hat{\delta}}{\hat{\delta}\phi_J}(\frac{\hat{\delta}\mathbf{R}_{mg}}{\hat{\delta}\phi_I})\big|_{\frac{D^I \mathbf{H}_g}{D\phi_J} = {}^{IJ}\mathbf{Z}_g}$

    end foreach

    add $w_{gp}\,{}^{IJ}\tilde{\mathbf{R}}_{mg}$ to ${}^{IJ}\tilde{\mathbf{R}}_m$

  end foreach

end foreach

solve $\mathbf{K}_m \frac{D^2 \mathbf{p}_m}{D\phi_I D\phi_J} + {}^{IJ}\tilde{\mathbf{R}}_m = 0$ for unknown $\frac{D^2 \mathbf{p}_m}{D\phi_I D\phi_J}$ using $\mathbf{K}_m$ from primal analysis

end

begin Solution of dependent second order sensitivity problem

Evaluation is only required for coupled case, for uncoupled ${}^{IJ}\mathbf{H}_g = {}^{IJ}\mathbf{Z}_g$

foreach *element in micro problem* **do**

  $\mathbf{p}_{me} \leftarrow \mathbf{p}_{me}\big|_{\frac{D\mathbf{p}_{me}}{D\phi_{IJ}} = \{0 \text{ if } \mathbf{p}_{mei} \in \bar{\mathbf{p}}_{me};\ {}^{IJ}\mathbf{Y}_i \text{ if } \mathbf{p}_{mei} \in \check{\mathbf{p}}_{me} \,:\, i=1,\ldots,n_p\}}$

  foreach *Gauss point* **do**

    foreach *sensitivity parameter* ϕ_I **do**

      ${}^{IJ}\mathbf{Z}_g \leftarrow {}^{IJ}\mathbf{H}_g$; $\frac{D^2 \mathbf{h}_{mg}}{D\phi_I D\phi_J} \leftarrow {}^{IJ}\mathbf{Z}_g - \mathbf{K}_Q^{-1} \frac{\hat{\delta}\mathbf{Q}_{mg}}{\hat{\delta}\phi_{IJ}}$

      export $\frac{D^2 \mathbf{h}_{mg}}{D\phi_I D\phi_J} \rightarrow {}^{IJ}\mathbf{H}_g$

    end foreach

  end foreach

end foreach

Figure 9: Second order sensitivity analysis algorithm
Slika 9: Algoritem za občutljivostno analizo drugega reda

4 SENSITIVITY ANALYSIS BASED FORMULATION OF MULTI-SCALE METHODS

In this section implementation of FE^2, and MIEL methods is in-depth described. An automatized sensitivity analysis based version of methods was developed. Traditional implementation with SCHUR complement is shortly presented.

4.1 MIEL method

At the macro level, we have compatible interpolation of unknown fields at the boundary of macro elements, whereas material characteristics, inhomogeneities, inner structure, such as openings, incisions of different materials, are defined only at micro scale. In Fig. 10, the MIEL procedure is presented. Lets assume the standard interpolation of displacements $\boldsymbol{u}_M$ on the boundary of the macro element

$$\boldsymbol{u}_M(\boldsymbol{\Xi}) = \sum_i N_i(\boldsymbol{\Xi})\, \boldsymbol{u}_{Mei} \tag{130}$$

where $N_i(\boldsymbol{\Xi})$ are finite element shape functions, $\boldsymbol{\Xi} = (\xi, \eta, \zeta)$ reference coordinates and $\boldsymbol{u}_{Mei}$ are displacements in i-th macro element node. To ensure compatibility of displacements at macro and micro level, we impose the essential boundary conditions at the complete boundary of the micro mesh by

$$\bar{\boldsymbol{u}}_m(\boldsymbol{\Xi}) = (\bar{\boldsymbol{u}}_{ms}(\boldsymbol{\Xi}) + (\lambda_m - \lambda_{ms})(\boldsymbol{u}_M(\boldsymbol{\Xi}) - \bar{\boldsymbol{u}}_{ms}(\boldsymbol{\Xi}))) \tag{131}$$

where $\bar{\boldsymbol{u}}_{ms}(\boldsymbol{\Xi})$ are displacements at the boundary at the end of the last macro step. The derivatives of (131) with respect to components of macro element nodal displacements are given by

$$\frac{\partial \bar{u}_{mi}(\boldsymbol{\Xi})}{\partial u_{Mejk}} = \delta_{ik}(\lambda_m - \lambda_{ms}) N_j(\boldsymbol{\Xi}). \tag{132}$$

A set of macro element unknowns is $\mathbf{p}_{Me} = \bigcup_{j,k} u_{Mejk}$ and $\mathbf{p}_m$ is composed of the micro mesh nodal displacements. Thus, (131) defines the dependency between the degrees of freedom with prescribed essential boundary condition at the micro level $\bar{\mathbf{p}}_m = \bar{\mathbf{p}}_m(\mathbf{p}_{Me}, \lambda_m)$ and macro element unknowns $\mathbf{p}_{Me}$.

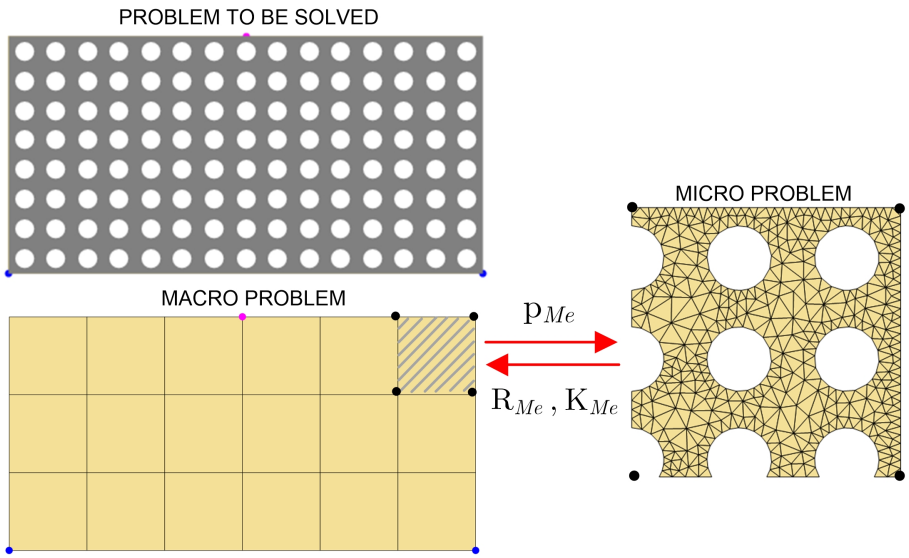


Figure 10: MIEL multi-scale scheme
Slika 10: MIEL večnivojska shema

a) Finite strain formulation

The macro element residual $\mathbf{R}_{Me}$ is in the case of MIEL obtained by the integration of the internal forces, part of weak form (66), over the micro mesh, where the micro deformation gradient $\boldsymbol{F}_m = \boldsymbol{F}_m(\mathbf{p}_{me}(\mathbf{p}_{Me}, \lambda_m))$ and micro stress tensor $\boldsymbol{P}_m = \boldsymbol{P}_m(\mathbf{p}_{me}(\mathbf{p}_{Me}, \lambda_m))$ implicitly depend on the degrees of freedom of macro element

$$\int_{\Omega_{Me}} \boldsymbol{P}_M : \delta \boldsymbol{F}_M \, dV = \int_{\Omega_m} \boldsymbol{P}_m : \delta \boldsymbol{F}_m \, dV = \sum_{e=1}^{n_{me}} \int_{\Omega_{me}} \boldsymbol{P}_m : \delta \boldsymbol{F}_m \, dV \tag{133}$$

Discretization of the micro mesh together with the variation of deformation gradient $\delta \boldsymbol{F}_m(\mathbf{p}_{me}(\mathbf{p}_{Me}, \lambda_m)) = \frac{\partial \boldsymbol{F}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}} \delta \mathbf{p}_{Me}$ and standard Gauss integration over the micro element domain Ω_{me} leads from (133) to the macro element residual $\mathbf{R}_{Me}$ in a form

$$\mathbf{R}_{Me} = \sum_{e=1}^{n_{me}} \sum_{g \in G_e} w_{gp} \, \mathbf{R}_{Mg} \tag{134}$$

$$\mathbf{R}_{Mg} = J_\xi \, \boldsymbol{P}_m : \frac{\partial \boldsymbol{F}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}} \tag{135}$$

where $\mathbf{R}_{Mg}$ is a contribution to the macro element residual evaluated at the micro element

Gauss points. Differentiation of (135) leads to the macro element tangent matrix

$$\mathbf{K}_{Me} = \sum_{e=1}^{n_{me}} \sum_{g \in G_e} w_{gp}\, \mathbf{K}_{Mg} \tag{136}$$

$$\mathbf{K}_{Mg} = \frac{\partial \mathbf{R}_{Mg}}{\partial \mathbf{p}_{Me}} = J_\xi \left(\frac{\partial \boldsymbol{P}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}} : \frac{\partial \boldsymbol{F}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}} + \boldsymbol{P}_m : \left(\frac{\partial^2 \boldsymbol{F}_m}{\partial \mathbf{p}_{me}{}^2} \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}} \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}} + \frac{\partial \boldsymbol{F}_m}{\partial \mathbf{p}_{me}} \frac{D^2\mathbf{p}_{me}}{D\mathbf{p}_{Me}^2} \right) \right) \tag{137}$$

where again $\mathbf{K}_{Mg}$ is a contribution to the macro element tangent evaluated at micro mesh Gauss points.

The residual and tangent matrix are obtained directly from the micro scale problem for each macro element and each macro element is associated with exactly one micro problem. Macro element performs only proper transfer of components of the macro element residual vector and tangent matrix from micro scale to macro scale finite element assembly procedure.

b) Small strain formulation

In the same way as for finite strain also for small strain formulation the macro element residual $\mathbf{R}_{Me}$ is obtained by the integration of the internal forces, part of weak form (63), over the micro mesh, where the micro small strain deformation tensor $\boldsymbol{\varepsilon}_m = \boldsymbol{\varepsilon}_m(\mathbf{p}_{me}(\mathbf{p}_{Me}, \lambda_m))$ and micro stress tensor $\boldsymbol{\sigma}_m = \boldsymbol{\sigma}_m(\mathbf{p}_{me}(\mathbf{p}_{Me}, \lambda_m))$ implicitly depend on the degrees of freedom of macro element

$$\int_{\Omega_{Me}} \boldsymbol{\sigma}_M : \delta\boldsymbol{\varepsilon}_M \, dV = \int_{\Omega_m} \boldsymbol{\sigma}_m : \delta\boldsymbol{\varepsilon}_m \, dV = \sum_{e=1}^{n_{me}} \int_{\Omega_{me}} \boldsymbol{\sigma}_m : \delta\boldsymbol{\varepsilon}_m \, dV \,. \tag{138}$$

Discretization of the micro mesh together with $\delta\boldsymbol{\varepsilon}_m(\mathbf{p}_{me}(\mathbf{p}_{Me}, \lambda_m)) = \frac{\partial \boldsymbol{\varepsilon}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}} \delta\mathbf{p}_{Me}$ the variation of strain tensor and standard Gauss integration over the micro element domain Ω_{me} leads from (138) to the macro element residual $\mathbf{R}_{Me}$. $\mathbf{R}_{Mg}$ is a contribution to the macro element residual evaluated at the micro element Gauss points.

$$\mathbf{R}_{Mg} = J_\xi \, \boldsymbol{\sigma}_m : \frac{\partial \boldsymbol{\varepsilon}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}} \tag{139}$$

Differentiation of (139) leads to the macro element tangent matrix

$$\mathbf{K}_{Mg} = \frac{\partial \mathbf{R}_{Mg}}{\partial \mathbf{p}_{Me}} = J_\xi \left(\frac{\partial \boldsymbol{\sigma}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}} : \frac{\partial \boldsymbol{\varepsilon}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}} + \boldsymbol{\sigma}_m : \left(\frac{\partial^2 \boldsymbol{\varepsilon}_m}{\partial \mathbf{p}_{me}{}^2} \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}} \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}} + \frac{\partial \boldsymbol{\varepsilon}_m}{\partial \mathbf{p}_{me}} \frac{D^2\mathbf{p}_{me}}{D\mathbf{p}_{Me}^2} \right) \right) \tag{140}$$

where again $\mathbf{K}_{Mg}$ is a contribution to the macro element tangent evaluated at micro mesh GAUSS points.

4.1.1 Sensitivity analysis based implementation of MIEL

Sensitivity analysis is required for evaluation of implicit dependencies $\frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}}$ and $\frac{D^2\mathbf{p}_{me}}{D\mathbf{p}_{Me}^2}$ in (135) and (137). From (130) follows a set of sensitivity parameters of the micro problem $\boldsymbol{\phi} = \boldsymbol{\phi}_{Me} = \mathbf{p}_{Me} = \bigcup_{j,k} u_{Mejk}$, and from (131) and (132) the components of velocity field ${}^I\mathbf{V} = \frac{\partial \bar{\mathbf{p}}_m(\boldsymbol{\phi}, \lambda_m)}{\partial \phi_I}$. Thus, the components of the first order boundary condition velocity field ${}^I\mathbf{V}$ are the values of the macro element shape functions at the position of the boundary nodes of the micro mesh. For boundary condition in the form of linear combination (131), the second derivatives are zero, and consequently the second order velocity fields are ${}^{IJ}\mathbf{V} = \mathbf{0}$.

Other quantities required in two-level path-following algorithm Fig. 6 are then: macro level variables that are passed to macro level $\mathbf{S} = \mathbf{S}_{Me} = \mathbf{R}_{Me}$ (135) and total derivative $\frac{D\mathbf{S}}{D\boldsymbol{\phi}} = \frac{D\mathbf{S}_{Me}}{D\boldsymbol{\phi}_{Me}} = \mathbf{K}_{Me}$. For the numerically efficient implementation of (135) and (137), we also need ADB form of (135) and (137). From (87) and ADB form of (135) and (137) follows

$$\mathbf{R}_{Mg} = J_\xi \left. \frac{\hat{\delta}\mathbf{W}}{\hat{\delta}\mathbf{p}_{Me}} \right|_{\mathbf{h}_{mg}=const., \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}}=\mathbf{Y}_{\boldsymbol{\phi}}}, \tag{141}$$

$$\mathbf{K}_{Mg} = \left. \frac{\hat{\delta}\mathbf{R}_{Mg}}{\hat{\delta}\mathbf{p}_{Me}} \right|_{\frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}}=\mathbf{Y}_{\boldsymbol{\phi}}, \frac{D\mathbf{Y}_{\boldsymbol{\phi}}}{D\mathbf{p}_{Me}}=\mathbf{Y}_{\boldsymbol{\phi}\boldsymbol{\phi}},} \tag{142}$$

where $\mathbf{Y}_{\boldsymbol{\phi}} = \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}}$ and $\mathbf{Y}_{\boldsymbol{\phi}\boldsymbol{\phi}} = \frac{D^2\mathbf{p}_{me}}{D\mathbf{p}_{Me}^2}$ are first end second order sensitivities calculated and stored during the analysis. *AceGen* input segment for MIEL task subroutine of micro finite element is presented in Step 6.2 of Appendix A.2 and *AceGen* input segment for MIEL macro finite element in Appendix A.3.

4.1.2 Schur complement based implementation of MIEL

Let us consider formulations where the solution is within one macro step path-independent, such as hyper-elastic problems solved with an arbitrary number of micro steps or elasto-plastic problems solved at the micro level in one load step. In this case, an alternative formulation of MIEL based on the calculation of SCHUR complement is possible, as originally presented in [39]. Let us form, at the converged state of the micro problem, a full set of equations that include unconstrained $\mathbf{p}_m$ and constrained $\bar{\mathbf{p}}_m$ unknowns by

$$\begin{bmatrix} \mathbf{K}_{\bar{m}} & \mathbf{K}_{\bar{m}m} \\ \mathbf{K}_{m\bar{m}} & \mathbf{K}_m \end{bmatrix} . \begin{bmatrix} \Delta\bar{\mathbf{p}}_m \\ \Delta\mathbf{p}_m \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{\bar{m}} \\ \mathbf{0} \end{bmatrix} . \tag{143}$$

SCHUR complement of (143) leads to reduced set of equations $\mathbf{K}_{cc}\,\Delta\bar{\mathbf{p}}_m = \mathbf{R}_c$, where $\mathbf{K}_{cc}$, and $\mathbf{R}_c$ are condensed tangent matrix and residual of micro problem, respectively. Since the relation $\bar{\mathbf{p}}_m = \bar{\mathbf{p}}_m(\mathbf{p}_{Me}, \lambda_m)$ is linear (see (131)), we can write

$$\bar{\mathbf{p}}_m = \mathbf{T} \,.\, \mathbf{p}_{Me} \tag{144}$$

where $\mathbf{T}$ is a transformation matrix (for details see [24]). The macro element residual and tangent matrix are then expressed by

$$\mathbf{R}_{Me} = \mathbf{T}^T \,.\, \mathbf{R}_c \tag{145}$$

$$\mathbf{K}_{Me} = \mathbf{T}^T \,.\, \mathbf{K}_{cc} \,.\, \mathbf{T} \tag{146}$$

With $\mathbf{R}_{Me}$ and $\mathbf{K}_{Me}$ known, one can apply the algorithm presented in Fig. 6, with sensitivity analysis related parts omitted. The size of $\mathbf{K}_{cc}$ is equal to the number of constrained DOFs at the boundary of the mesh and grows with the micro mesh density. For densely meshed micro structure the calculation of the SCHUR complement inflicts high memory allocation and is time consuming. On the contrary, the number of sensitivity parameters is the same as the number of nodal unknowns of the macro element, thus independent of micro mesh density. Schematic comparison can be seen for 2D case discretized with 4 nodded elements in Fig. 11. For SCHUR complement implementation, condensation is done with respect to DOFs of 20 border nodes. The dimension of the resulting matrix $\mathbf{K}_{cc}$ is 40x40. To get macro element tangent matrix $\mathbf{K}_{Me}$ with dimension 8x8, additional transformations (145), (146) need to be performed. With the growth of mesh density, also the number of micro-structure border nodes grows and with that the dimension of the matrix to be calculated. In the case of sensitivity based implementation, the second

order sensitivity analysis is needed with respect to 8 DOFs in macro element corner nodes and summation of (141), (142) over the micro mesh integration points.

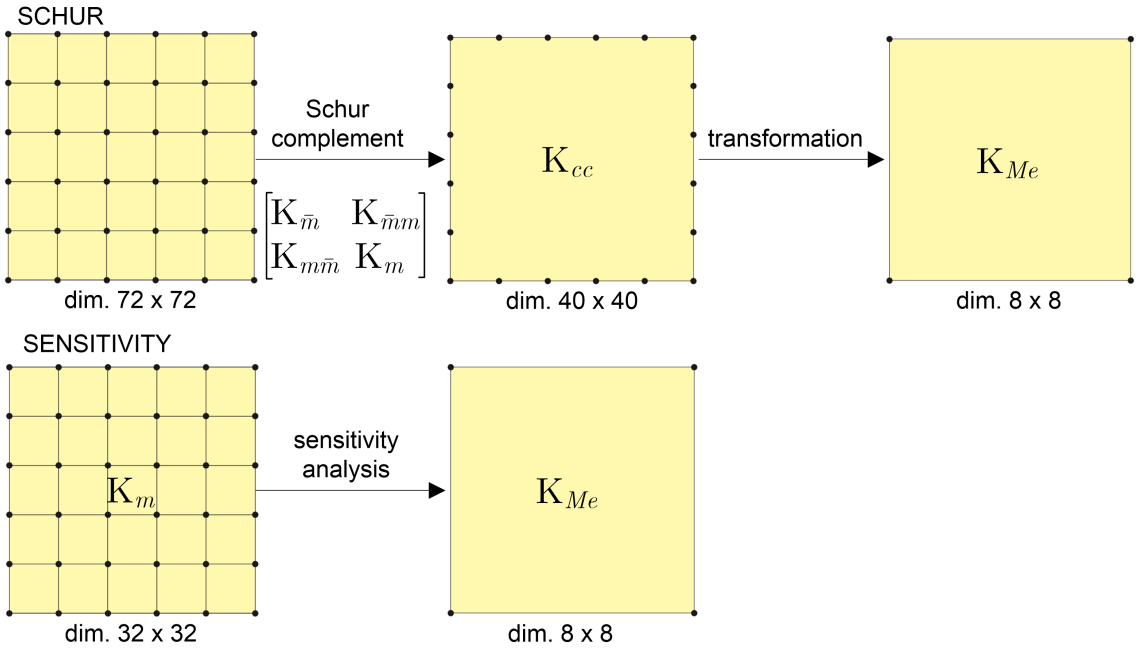


Figure 11: MIEL macro tangent matrix $\mathbf{K}_{Me}$; above - SCHUR complement implementation and below - sensitivity based implementation
Slika 11: MIEL makro tangentna matrika $\mathbf{K}_{Me}$; zgoraj - implementacija s SCHUR komplementom in spodaj - implementacija z občutljivostno analizo

The comparison of the computational cost of the two implementations is done for the 3D case, which is more computationally demanding than the 2D case. In Fig. 12, the calculation time for the SCHUR complement and for the second order sensitivity analysis are presented in relation to the density of micro mesh. The example is composed of one 3D hexahedral macro element. The macro element is uniformly subdivided into $n \times n \times n$ micro mesh. Two micro material models are considered, finite strain elasto-plastic and hyper-elastic as defined in Sec. 2.3.3, based on hyper-elastic strain energy (77). The SCHUR complement's computational time grows polynomially, whereas sensitivity calculation retains approximate linearity with the number of equations at the micro level. The timing of the sensitivity analysis increases with the complexity of the material model and the number of DOFs of the macro element. However, overall behavior remains the same.

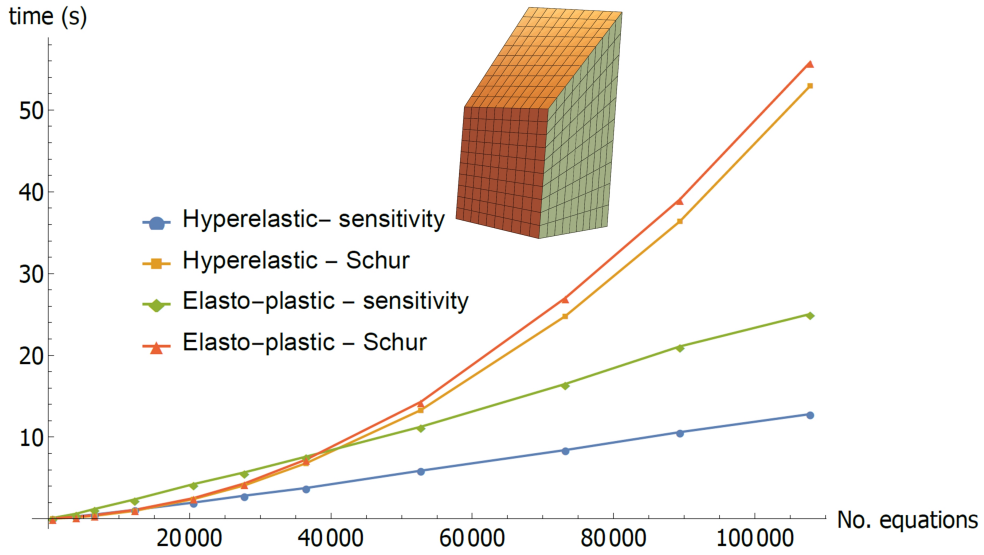


Figure 12: Comparison of the computational time with respect to micro mesh density for two implementations of MIEL method
Slika 12: Primerjava računskega časa glede na gostoto mikro mreže za različni implementaciji MIEL metode

4.2 FE^2 method

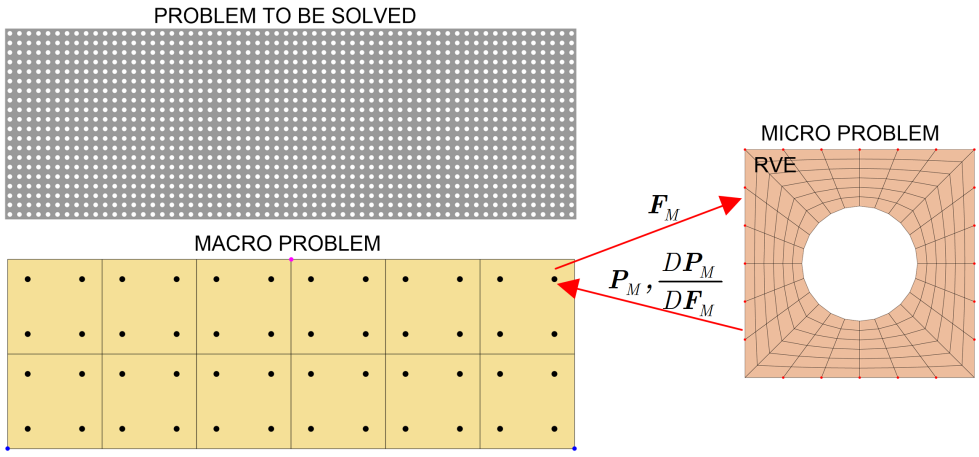


Figure 13: FE^2 multi-scale scheme
Slika 13: FE^2 večnivojska shema

a) Finite strain formulation

Within the FE^2 approach we have one micro FE model, also called a representative volume element (RVE), at each macro mesh integration point as shown in Fig. 13. All information

about micro-structure is obtained from computations at the micro level by averaging the material response characterized by an appropriate stress measure and constitutive tangent matrix over RVE. With the GAUSS point contribution to the macro level weak form ($\boldsymbol{P}_M : \delta\boldsymbol{F}_M$) and macro level discretization of deformation gradient $\delta\boldsymbol{F}_M = \frac{\partial \boldsymbol{F}_M}{\partial \mathbf{p}_{Me}}\delta\mathbf{p}_{Me}$, the macro element residual leads to

$$\mathbf{R}_{Me} = \sum_{g\in G_e} w_{gp}\, \mathbf{R}_{Mg}, \tag{147}$$

$$\mathbf{R}_{Mg} = J_\xi\, \boldsymbol{P}_M : \frac{\partial \boldsymbol{F}_M}{\partial \mathbf{p}_{Me}}, \tag{148}$$

where the macro level first Piola-Kirchoff stress tensor $\boldsymbol{P}_M$ is obtained by averaging the micro level first Piola-Kirchoff stress tensor $\boldsymbol{P}_M = \{\boldsymbol{P}_m\}$. The operation of averaging is here denoted by $\{\cdot\}$. Several types of boundary conditions can be imposed on the RVE: e.g., fully prescribed displacements and fully prescribed traction, which are based on the uniform strain and stress assumptions and periodic boundary conditions that enforce a displacement constraint, which is suited for periodic media. Here, periodic boundary conditions are achieved (see e.g. [32]) by applying first the prescribed displacements in the corners of RVE by

$$\bar{\boldsymbol{u}}_m = (\boldsymbol{F}_{Ms} + (\lambda_m - \lambda_{ms})(\boldsymbol{F}_M - \boldsymbol{F}_{Ms}) - \mathbf{I})\, \boldsymbol{X}_m \tag{149}$$

where $\boldsymbol{F}_{Ms}$ is macro deformation gradient at the end of the last macro step. The derivatives of (149) with respect to components of $\boldsymbol{F}_M$ are given by

$$\frac{\partial \bar{u}_{mi}}{\partial F_{Mjk}} = \delta_{ij}(\lambda_m - \lambda_{ms})X_{mk}. \tag{150}$$

Thus, (149) defines the dependency $\bar{\mathbf{p}}_m = \bar{\mathbf{p}}_m(\boldsymbol{F}_M, \lambda_m)$ between the set of micro nodal unknowns with prescribed essential boundary condition $\bar{\mathbf{p}}_m$ and the macro deformation gradient $\boldsymbol{F}_M$. For the unconstrained boundary nodes, the periodicity of boundary conditions is adopted with the use of Lagrange multipliers (for details see [55]). Note that the introduction of Lagrange constraints only extends the vector of micro level unknowns $\mathbf{p}_m$ with Lagrange multipliers and micro level residual $\mathbf{R}_m$ with constraint equations and it does not change the primal and sensitivity analysis procedures described in Section 3.

Differentiation of (135) then leads to the macro element tangent matrix

$$\mathbf{K}_{Me} = \sum_{g \in G_e} w_{gp}\, \mathbf{K}_{Mg} \tag{151}$$

$$\mathbf{K}_{Mg} = \frac{\partial \mathbf{R}_{Mg}}{\partial \mathbf{p}_{Me}} + \frac{\partial \mathbf{R}_{Mg}}{\partial \boldsymbol{P}_M} \frac{D\boldsymbol{P}_M}{D\boldsymbol{F}_M} \frac{\partial \boldsymbol{F}_M}{\partial \mathbf{p}_{Me}} \tag{152}$$

where $\frac{D\boldsymbol{P}_M}{D\boldsymbol{F}_M} = \{\frac{\partial \boldsymbol{P}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\boldsymbol{F}_M}\}$ is macroscopic constitutive matrix obtained by averaging the microscopic constitutive matrices.

b) Small strain formulation

For a small strain formulation, the Gauss point contribution to the macro level weak form $(\boldsymbol{\sigma}_M : \delta\boldsymbol{\varepsilon}_M)$ and macro level discretization of a small strain tensor $\delta\boldsymbol{\varepsilon}_M = \frac{\partial \boldsymbol{\varepsilon}_M}{\partial \mathbf{p}_{Me}} \delta\mathbf{p}_{Me}$, leads to the Gauss point contribution to macro element residual

$$\mathbf{R}_{Mg} = J_\xi\, \boldsymbol{\sigma}_M : \frac{\partial \boldsymbol{\varepsilon}_M}{\partial \mathbf{p}_{Me}}, \tag{153}$$

where the macro level Cauchy stress tensor $\boldsymbol{\sigma}_M$ is obtained by averaging the micro level Cauchy stress tensor $\boldsymbol{\sigma}_M = \{\boldsymbol{\sigma}_m\}$. Periodic boundary conditions are achieved by applying the prescribed displacements in the corners of RVE by

$$\bar{\boldsymbol{u}}_m = \left(\boldsymbol{\varepsilon}_{M\,s} + (\lambda_m - \lambda_{m\,s})(\boldsymbol{\varepsilon}_M - \boldsymbol{\varepsilon}_{M\,s})\right) \boldsymbol{X}_m \tag{154}$$

where $\boldsymbol{\varepsilon}_{M\,s}$ is macro small strain tensor at the end of the last macro step. The derivatives of (149) with respect to components of $\boldsymbol{\varepsilon}_M$ are given by

$$\frac{\partial \bar{u}_{mi}}{\partial \boldsymbol{\varepsilon}_{M\,jk}} = \delta_{ij}(\lambda_m - \lambda_{ms}) X_{mk}. \tag{155}$$

Thus, (154) defines the dependency $\bar{\mathbf{p}}_m = \bar{\mathbf{p}}_m(\boldsymbol{\varepsilon}_M, \lambda_m)$ between the set of micro nodal unknowns with prescribed essential boundary condition $\bar{\mathbf{p}}_m$ and the macro small strain tensor $\boldsymbol{\varepsilon}_M$. Differentiation of (139) then leads to the Gauss point contribution of macro element tangent matrix

$$\mathbf{K}_{Mg} = \frac{\partial \mathbf{R}_{Mg}}{\partial \mathbf{p}_{Me}} + \frac{\partial \mathbf{R}_{Mg}}{\partial \boldsymbol{\sigma}_M} \frac{D\boldsymbol{\sigma}_M}{D\boldsymbol{\varepsilon}_M} \frac{\partial \boldsymbol{\varepsilon}_M}{\partial \mathbf{p}_{Me}} \tag{156}$$

where $\frac{D\boldsymbol{\sigma}_M}{D\boldsymbol{\varepsilon}_M} = \{\frac{\partial \boldsymbol{\sigma}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\boldsymbol{\varepsilon}_M}\}$ is macroscopic constitutive matrix obtained by averaging the microscopic constitutive matrices.

4.2.1 Sensitivity analysis based implementation of FE^2

a) Finite strain formulation

The FE^2 method was already implemented using sensitivity analysis in [45, 55], but without two interacting path-following schemes which is introduced in this thesis. Sensitivity analysis is required for the evaluation of implicit dependency $\frac{D\boldsymbol{P}_M}{D\boldsymbol{F}_M}$ in (152). From (149) there follows a set of sensitivity parameters of micro problem

$$\boldsymbol{\phi} = \bigcup_{ij} F_{M\,ij} \tag{157}$$

and from (150) the components of velocity field ${}^{I}\mathbf{V} = \frac{\partial \bar{\mathbf{p}}_m(\boldsymbol{\phi},\lambda_m)}{\partial \phi_I}$. Thus, the components of the first order boundary condition velocity field ${}^{I}\mathbf{V}$ are appropriate nodal coordinates of the corner nodes of the micro mesh. For boundary condition in the form of linear combination (149), the second derivatives are zero, i.e., ${}^{IJ}\mathbf{V} = \mathbf{0}$. The micro level variables that are passed to macro level from a single RVE are $\mathbf{S} = \boldsymbol{P}_M = \{\boldsymbol{P}_m\}$ and the total derivative $\frac{D\mathbf{S}}{D\boldsymbol{\phi}} = \{\frac{\partial \boldsymbol{P}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\boldsymbol{F}_M}\}$. The contributions of micro problems at all Gauss points of macro element are needed for the formulation of macro element. Thus, a complete set of variables passed from macro element to micro problems is $\boldsymbol{\phi}_{Me} = \bigcup_{g\in G_e} \boldsymbol{\phi}^{(g)}$, where G_e is a set of Gauss points of the e-th macro element. A complete set of variables passed from micro to macro element is $\mathbf{S}_{Me} = \bigcup_{r\in\mathcal{M}_e} \mathbf{S}^{(r)}$ and $\frac{D\mathbf{S}_{Me}}{D\boldsymbol{\phi}_{Me}} = \bigcup_{r\in\mathcal{M}_e} \left(\frac{D\mathbf{S}}{D\boldsymbol{\phi}}\right)^{(r)}$ where $\mathcal{M}_e$ is a set of micro problems that corresponds to G_e.

For a numerically efficient implementation of (148) and (152), we also need the ADB form of (148) and (152). From $\boldsymbol{P}_M : \delta\boldsymbol{F}_M = \mathbf{S} : \delta\boldsymbol{F}_M$ the ADB form of (148) and (152) leads to

$$\mathbf{R}_{Mg} = J_\xi \left.\frac{\hat{\delta}(\mathbf{S} : \boldsymbol{F}_M)}{\hat{\delta}\mathbf{p}_{Me}}\right|_{\mathbf{S}=const.} \tag{158}$$

$$\mathbf{K}_{Mg} = \left.\frac{\hat{\delta}\mathbf{R}_{Mg}}{\hat{\delta}\mathbf{p}_{Me}}\right|_{\frac{D\mathbf{S}}{D\boldsymbol{F}_M}=\frac{D\mathbf{S}}{D\boldsymbol{\phi}}} \tag{159}$$

and

$$\frac{D\mathbf{S}}{D\boldsymbol{\phi}} = \left\{\frac{\partial \boldsymbol{P}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\boldsymbol{F}_M}\right\} = \left\{\left.\frac{\hat{\delta}\boldsymbol{P}_m}{\hat{\delta}\boldsymbol{F}_M}\right|_{\frac{D\mathbf{p}_{me}}{D\boldsymbol{F}_M}=\mathbf{Y}_{\boldsymbol{\Phi}}}\right\} \tag{160}$$

where $\mathbf{Y}_{\boldsymbol{\Phi}} = \frac{D\mathbf{p}_{me}}{D\boldsymbol{F}_M}$ are already calculated and stored first order sensitivities. *AceGen* in-

put segment for FE^2 task subroutine of micro finite element is presented in Step 6.3 of Appendix A.2 and *AceGen* input segment for FE^2 macro finite element in Appendix A.4.

In [55] a step forward was made with the introduction of symmetric stretch tensor $\boldsymbol{U}_M$ as strain measure at macro level instead of asymmetric deformation gradient $\boldsymbol{F}_M$, to determine boundary conditions on embedded micro-structure. Stretch tensor $\boldsymbol{U}_M$ can be calculated as matrix square root of Cauchy-Green tensor $\boldsymbol{C}_M$, for which efficient, automated way of evaluation together with its derivatives can be found in [23]. Use of symmetric stretch tensor $\boldsymbol{U}_M$, that has only 6 components, instead of $\boldsymbol{F}_M$ with 9, significantly reduces computational cost of boundary condition related sensitivity analysis of micro-structure and with it the evaluation of local macroscopic stress tensors and tangent matrices. The following equations summarize the alternative formulation.

$$\boldsymbol{C}_M = \boldsymbol{F}_M^T \boldsymbol{F}_M, \quad \boldsymbol{U}_M = \sqrt{\boldsymbol{C}_M} \tag{161}$$

$$\boldsymbol{\phi}_M = \mathrm{vec}(\mathbf{U}_M) = \{U_{M,11}, U_{M,12}, U_{M,13}, U_{M,22}, U_{M,23}, U_{M,33}\} \tag{162}$$

$$\bar{u}_m = (\boldsymbol{U}_M - \mathbf{I})\, \boldsymbol{X}_m, \frac{\partial \bar{u}_{mi}}{\partial U_{Mjk}} = \delta_{ij}\, X_{mk} \tag{163}$$

$$\mathbf{R}_{Mg} = J_\xi\, \boldsymbol{P}_M : \frac{\partial \boldsymbol{U}_M}{\partial \mathbf{p}_{Me}} \tag{164}$$

$$\mathbf{K}_{Mg} = \frac{\partial \mathbf{R}_{Mg}}{\partial \mathbf{p}_{Me}} + \frac{\partial \mathbf{R}_{Mg}}{\partial \boldsymbol{P}_M} \frac{D\boldsymbol{P}_M}{D\boldsymbol{U}_M} \frac{\partial \boldsymbol{U}_M}{\partial \mathbf{p}_{Me}} \tag{165}$$

The micro level data that is passed to macro level from single RVE are $\mathbf{S} = \boldsymbol{P}_M = \{\boldsymbol{P}_m\}$ and the total derivative $\frac{D\mathbf{S}}{D\boldsymbol{\phi}} = \{\frac{\partial \boldsymbol{P}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\boldsymbol{U}_M}\}$. The contributions of micro problems at all Gauss points of macro element are needed for the formulation of macro element. Thus, a complete set of data passed from macro element to micro problems is $\boldsymbol{\phi}_{Me} = \bigcup_{g \in G_e} \boldsymbol{\phi}^{(g)}$, where G_e is a set of Gauss points of the e-th element. A complete set of variables passed from micro to macro element is $\mathbf{S}_{Me} = \bigcup_{r \in \mathcal{M}_e} \mathbf{S}^{(r)}$ and $\frac{D\mathbf{S}_{Me}}{D\boldsymbol{\phi}_{Me}} = \bigcup_{r \in \mathcal{M}_e} \left(\frac{D\mathbf{S}}{D\boldsymbol{\phi}}\right)^{(r)}$ where $\mathcal{M}_e$ is a set of micro problems that corresponds to G_e. From $\boldsymbol{P}_M : \delta \boldsymbol{U}_M = \mathbf{S} : \delta \boldsymbol{U}_M$ the ADB follows as

$$\mathbf{R}_{Mg} = J_\xi \left. \frac{\hat{\delta}(\mathbf{S} : \boldsymbol{U}_M)}{\hat{\delta}\mathbf{p}_{Me}} \right|_{\mathbf{S}=const.} \tag{166}$$

$$\mathbf{K}_{Mg} = \left. \frac{\hat{\delta}\mathbf{R}_{Mg}}{\hat{\delta}\mathbf{p}_{Me}} \right|_{\frac{D\mathbf{S}}{D\boldsymbol{U}_M} = \frac{D\mathbf{S}}{D\boldsymbol{\phi}}} \tag{167}$$

and

$$\frac{D\mathbf{S}}{D\boldsymbol{\phi}} = \left\{ \frac{\partial \boldsymbol{P}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\,\boldsymbol{U}_M} \right\} = \left\{ \left. \frac{\hat{\delta}\boldsymbol{P}_m}{\hat{\delta}\,\boldsymbol{U}_M} \right|_{\frac{D\mathbf{p}_{me}}{D\boldsymbol{U}_M} = \mathbf{Y}_{\boldsymbol{\phi}}} \right\} \tag{168}$$

where $\mathbf{Y}_{\boldsymbol{\phi}} = \frac{D\mathbf{p}_{me}}{D\boldsymbol{U}_M}$ are already calculated first order sensitivities.

b) Small strain formulation

In case of small strain formulation, sensitivity analysis is required for the evaluation of implicit dependency $\frac{D\boldsymbol{\sigma}_M}{D\boldsymbol{\varepsilon}_M}$ in (152). From (154) there follows a set of sensitivity parameters of micro problem

$$\boldsymbol{\phi} = \bigcup_{ij;i\geqslant j} \boldsymbol{\varepsilon}_{M\,ij} \tag{169}$$

and from (155) the components of velocity field ${}^I\mathbf{V} = \frac{\partial \bar{\mathbf{p}}_m(\boldsymbol{\phi},\lambda_m)}{\partial \phi_I}$. Contrary to (157) only symmetric part of ε_M forms a set of sensitivity parameters. The micro level variables that are passed to macro level from a single RVE are $\mathbf{S} = \boldsymbol{\sigma}_M = \{\boldsymbol{\sigma}_m\}$ and the total derivative $\frac{D\mathbf{S}}{D\boldsymbol{\phi}} = \{\frac{\partial \boldsymbol{\sigma}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\boldsymbol{\varepsilon}_M}\}$.

For the numerically efficient implementation of (153) and (156), we also need the ADB form of (153) and (156). From $\boldsymbol{\sigma}_M : \delta\boldsymbol{\varepsilon}_M = \mathbf{S} : \delta\boldsymbol{\varepsilon}_M$ the ADB form of (153) and (156) leads to

$$\mathbf{R}_{Mg} = J_\xi \left. \frac{\hat{\delta}(\mathbf{S} : \boldsymbol{\varepsilon}_M)}{\hat{\delta}\mathbf{p}_{Me}} \right|_{\mathbf{S}=const.} \tag{170}$$

$$\mathbf{K}_{Mg} = \left. \frac{\hat{\delta}\mathbf{R}_{Mg}}{\hat{\delta}\mathbf{p}_{Me}} \right|_{\frac{D\mathbf{S}}{D\boldsymbol{\varepsilon}_M} = \frac{D\mathbf{S}}{D\boldsymbol{\phi}}} \tag{171}$$

and

$$\frac{D\mathbf{S}}{D\boldsymbol{\phi}} = \left\{ \frac{\partial \boldsymbol{P}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\boldsymbol{\varepsilon}_M} \right\} = \left\{ \left. \frac{\hat{\delta}\boldsymbol{\sigma}_m}{\hat{\delta}\boldsymbol{\varepsilon}_M} \right|_{\frac{D\mathbf{p}_{me}}{D\boldsymbol{\varepsilon}_M} = \mathbf{Y}_{\boldsymbol{\phi}}} \right\} \tag{172}$$

where $\mathbf{Y}_{\boldsymbol{\phi}} = \frac{D\mathbf{p}_{me}}{D\boldsymbol{\varepsilon}_M}$ are already calculated and stored first order sensitivities.

4.2.2 Schur complement based implementation of FE^2

a) Finite strain formulation

As in the case of the MIEL method, the SCHUR complement of constrained nodal DOF at the micro level can be used to calculate macro element residual and tangent matrix. The method leads to the traditional implementation of the FE^2 method, as introduced in [32].

For condensation of tangent matrix equations (143) can be applied, as shown in Section 4.1.2. It gives condensed tangent matrix $\mathbf{K}_{cc}$ and residual $\mathbf{R}_c$. To get Piola-Kirchoff stress tensors $\boldsymbol{P}_M$ and a constitutive matrix at the macro level some additional steps are needed. Expression (173) relates variations of macroscopic stress and deformation, where macro deformation gradient $\boldsymbol{F}_M$ comes directly from macro element and macro Piola-Kirchoff stress tensors $\boldsymbol{P}_M$ is calculated as shown below with (174). The fourth order tensor ${}^4\mathbf{S}_M$ represents the required consistent stiffness at the macroscopic integration point level and is calculated with equation (175), where summation is done over constrained nodes marked with n and where $\mathbf{X}_{0i}$ are initial coordinates of i-th node. Components of macroscopic constitutive matrix are components of ${}^4\mathbf{S}_M$ and are together with $\boldsymbol{P}_M$ sent to individual macro element GAUSS point as an input for calculations at macro element. In macro element, additional transformation is needed when residual $\mathbf{R}_g$ is calculated. For more details about equations used here see [32].

$$ {}^4\mathbf{S}_M : \delta \boldsymbol{F}_M^T = \delta \boldsymbol{P}_M \tag{173} $$

$$ \boldsymbol{P}_M = \bar{\boldsymbol{P}}_{RVE} = \frac{1}{A} \sum_{i \in n} \mathbf{R}_{ci} \otimes \mathbf{X}_{0i} \tag{174} $$

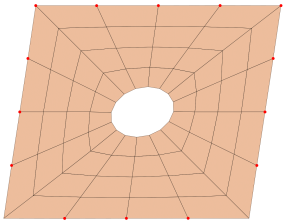
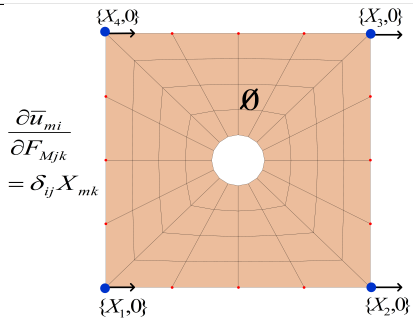
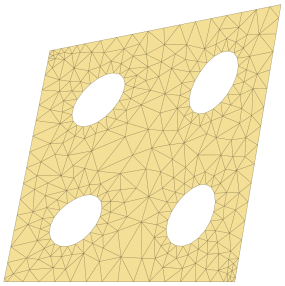
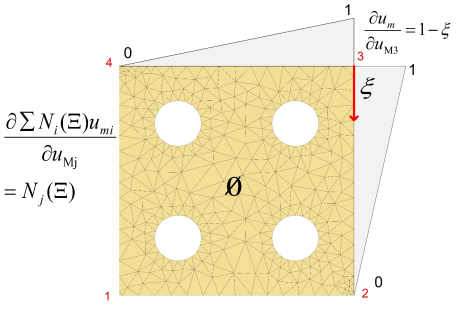
$$ {}^4\mathbf{S}_M = \frac{1}{A} \sum_{i \in n} ((\sum_{j \in n} \mathbf{K}_{ccij} \otimes \mathbf{X}_{0j}) \otimes \mathbf{X}_{0i}) \tag{175} $$

The number of RVE corner nodes is constant, which makes the cost of calculating the SCHUR complement independent of the density of the micro mesh, while the advantages of using the sensitivity analysis are less pronounced as for MIEL. Note that the standard method is only consistent for problems that are not path-dependent within a single macro step.

4.3 Unification of multi-scale models

Automatic-differentiation-based (ADB) formulation enables unification and automation of various multi-scale approaches for an arbitrary nonlinear time dependent coupled problem (e.g. general finite strain plasticity). For all methods we need individual finite element codes that support the first and second order sensitivity analysis. It is used for the evaluation of implicit derivatives, that are derivatives of unknowns of the problem. Sensitivity related codes are general, thus problem independent. Additional problem dependent user subroutines are required to evaluate homogenized constitutive matrix and, macro stress for FE^2, and residual and macro tangent matrix for MIEL.

Table 1: Comparison between FE^2 and MIEL
Preglednica 1: Primerjava med FE^2 in MIEL

method	micro-scale	characteristic velocity field
FE^2 periodic BC		
MIEL		

Differences between the methods are in essential boundary conditions at micro mesh and in essential boundary conditions velocity fields needed for the sensitivity analysis, see Tab. 1. Macro element used in FE^2 evaluates residual and tangent matrix (see Eq.(30) and (31)), whereas macro element used for MIEL is used just for the transformation of

data. The implemented FE^2 and MIEL schemes together represent unified approach to the automation of multi-scale modelling.

Implementation of presented multi-scale computational approach in *AceFEM* is fully parallelized for multi-core processors. Micro problems are distributed on kernels by evaluating each individual micro problem always on the same kernel. For FE^2 each RVE is associated with individual Gauss point and can be calculated on individual kernel. The same goes for MIEL, where each micro problem can be distributed to individual kernel. With parallelized computation, computational time for complex problems can be significantly reduced. The setup is also appropriate for implementation on clusters.

5 VALIDATION AND VERIFICATION OF ALGORITHMS

Numerical examples were calculated using program packages *AceGen* and *AceFEM* [28]. Finite element user subroutines for primal and analytical first and second order sensitivity analyses were automatically derived, optimized, and written in C with the use of the *AceGen* automatic code generator. The MIEL and FE^2 methods based on sensitivity analysis, as well as the one based on the SCHUR complement were implemented within the *AceFEM* environment according to the algorithm defined in Fig. 6. The Intel MKL sparse linear algebra numerical library was used for the linear algebra operations (calculation of the SCHUR complement and the solution of linear systems of equations). The actual *AceGen* and *AceFEM* inputs for the complex multi-scale problems addressed here are too lengthy to be included in the dissertation. However, they are freely available at `http://symech.fgg.uni-lj.si/Examples/MultiScale.pdf`, in a form of Mathematica notebook at `http://symech.fgg.uni-lj.si/Examples/MultiScale.nb` or as a part of software documentation available at `http://symech.fgg.uni-lj.si/Download.htm`. Numerical examples were presented also in [31, 73].

The abbreviations used to indicate a specific combination of method solution procedures are structured as

$$method - n_M/n_m - implementation$$

where *method* can be MIEL, FE^2 or MIX (when MIEL and FE^2 are used together in one model); n_M is set to the number of macro steps or "Adaptive"for adaptive macro time stepping; n_m is set to the number of micro steps for each macro step or "Adaptive"for adaptive micro time stepping *implementation* is set to " Sens."for the sensitivity analysis based implementation, " SchurMMA"for the SCHUR complement based formulation implemented in Mathematica, and " SchurMKL"for the SCHUR complement based formulation implemented with the Intel MKL library. The SCHUR complement based implementation is computationally identical to the sensitivity analysis based implementation for $n_m = 1$. Although Mathematica and MKL both calculate the same SCHUR complement, the algorithm implemented in MKL performs perturbation of the zeros at the main diagonal, resulting in a slightly imprecise tangent matrix as shown and explained in the examples presented in Section 5.2.

5.1 Validation of implemented multi-scale algorithm

The first numerical example is a three-dimensional cantilever with the right and left ends clamped, as shown in Fig. 14. Uniform pressure $p = 10$ in the vertical z direction was imposed in the middle of the top surface of the cantilever. The dimensions of the cantilever are $12 \times 2.4 \times 2.4$. At both levels, 3D, eight nodded, isoparametric, hexahedral elements, integrated with $2 \times 2 \times 2$ GAUSS integration are. A finite strain elasto-plastic material model as described in Section 2.3.3 is used at the micro level. Material properties are $E = 21000$, $\nu = 0.3$, $\sigma_{y0} = 24$ and $K_h = 100$. A homogeneous mesh is used at both levels; thus, for validation purposes, the micro level is uniform and no micro-structure is present. The simulations were performed with adaptive time stepping at both levels. The displacements in the z direction of nodes on the line $\overline{AB}$ are presented for all simulations in Fig. 15. The extent of the plastic zone at the end of the simulation is shown in Fig. 14, where red indicates the plastic region. Multi-scale results are compared with the single-scale results. The same finite elements are used for the single-scale mesh as for the micro level mesh of the multi-scale simulation. First, the FE^2 method is verified by comparing

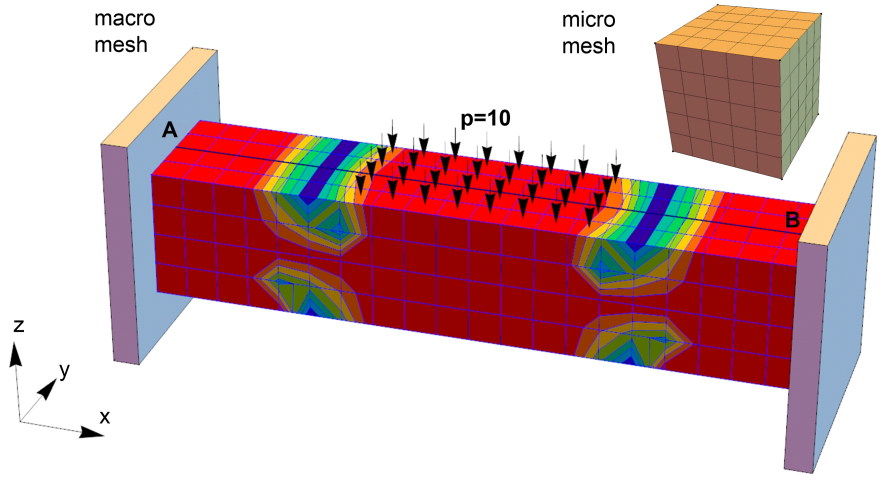


Figure 14: Clamped cantilever with macro and micro mesh and enforced natural and essential boundary conditions
Slika 14: Obojestransko vpet nosilec z mikro in makro mrežo ter predpisanimi naravnimi in bistvenimi robnimi pogoji

the results obtained from the single-scale analysis with $20 \times 4 \times 4$ mesh with those obtained from the multi-scale analysis for macro mesh grid of $20 \times 4 \times 4$ and micro mesh grids of $2 \times 2 \times 2$ and $10 \times 10 \times 10$. The results must be independent of the micro mesh density due to homogeneity and the exact enforcement of periodicity with the micro mesh boundary conditions. Multi-scale results must also be identical to the single-scale results, which is

shown in Fig. 15 (curves 1, 2 and 3). This verifies the FE^2 implementation.

For the MIEL method, the results of the single-scale and multi-scale simulations can be identical only when the micro mesh grid is $1 \times 1 \times 1$, as shown in Fig. 15 (curves 1, 4). This also verifies the MIEL implementation. With the change of micro mesh grid to $2 \times 2 \times 2$, $5 \times 5 \times 5$ and finally to $10 \times 10 \times 10$ (curves 5, 6, 7), the MIEL results approach the single-scale FEM solution obtained with the mesh $80 \times 16 \times 16$ (curve 8). This is a consequence of a better description of the deformation field over the macro element domain, which partially eliminates the locking behavior of the isoparametric hexahedral element. The effect is similar to that of enhanced-strain finite elements, where additional degrees of freedom are added inside the elements.

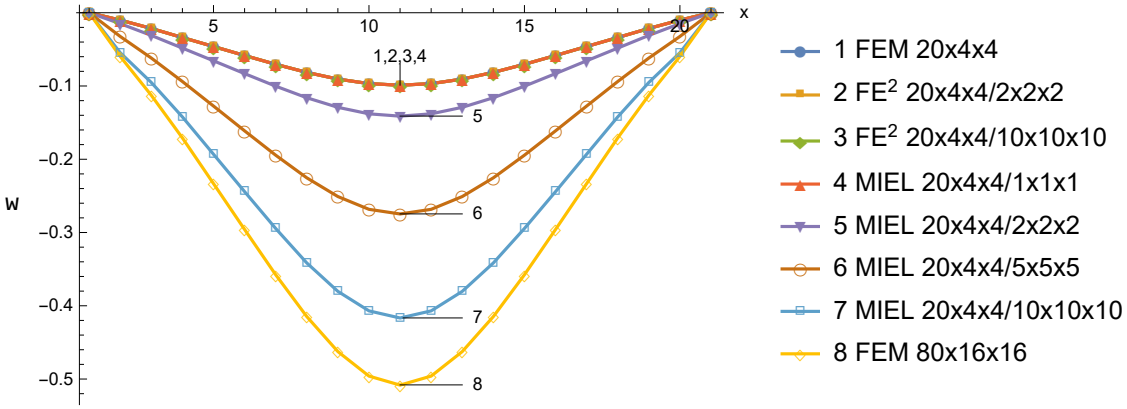


Figure 15: Displacement in z direction of line AB
Slika 15: Pomik linije AB v z smeri

5.2 Convergence rate of two-level path-following iterative procedure

The convergence rate of the two-level path-following iterative procedure defined by an algorithm in Fig. 6 is investigated using an example from the previous section. The simulation is performed in 10 steps with a constant load increment $\Delta\lambda_M = 0.1$. A homogeneous micro mesh of $5 \times 5 \times 5$ is used in all cases. Each macro step is followed by one micro step (denoted by -10/1-) or 5 micro steps (denoted by -10/5-). The convergence rate results of the two-level path-following iterative procedure are shown for the last macro load step (where most of the integration points are already in the plastic regime) in Tables 2 and 3. The effect of the number of micro steps and the type of implementation (SCHUR complement or sensitivity analysis based) is investigated.

Table 2 shows that, when one macro load step is followed by one micro step (MIEL-10/1-

SchurMKL and MIEL-10/1-Sens.), convergence is quadratic and the results are identical regardless of implementation. The sensitivity-based implementation retains quadratic convergence for $n_m = 5$, while the SchurMKL based implementation converges very slowly. The column denoted by MIEL-10/5-Sens.end contains a special case, where the sensitivity equations given in Section 3.3.1 are not resolved after each micro step but only at the end of the micro solution. This is equivalent to the MIEL-10/5-SchurMKL implementation, showing that only a fully consistent sensitivity analysis ensures quadratic convergence of the overall MIEL algorithm.

Table 2: Comparison of MIEL convergence rate for the last macro step
Preglednica 2: Primerjava konvergence MIEL za zadnji makro korak

NR it.	MIEL-10/1-Sens.	MIEL-10/5-Sens.	MIEL-10/5-Sens.end	MIEL-10/1-SchurMKL	MIEL-10/5-SchurMKL
1	1.023e-01	1.036e-01	1.036e-01	1.023e-01	1.036e-01
2	7.304e-03	4.999e-03	4.089e-03	7.304e-03	4.089e-03
3	4.779e-03	3.875e-03	4.380e-03	4.779e-03	4.380e-03
4	8.786e-05	6.749e-05	3.984e-04	8.786e-05	3.984e-04
5	6.102e-07	5.175e-07	7.115e-05	6.102e-07	7.115e-05
6	7.051e-12	5.889e-12	1.962e-05	7.066e-12	1.962e-05
7	9.325e-17	1.778e-16	6.829e-06	1.016e-14	6.829e-06
8			2.708e-06		2.708e-06
...			...		...
29			5.948e-13		5.950e-13

Secondly, the FE^2 scheme convergence rate is shown in Table 3 and the same conclusions drawn for MIEL apply. Only fully consistent sensitivity analysis ensures quadratic convergence of the overall FE^2 algorithm. The last two columns of Table 3 contain the results of the SCHUR complement based formulation implemented directly in Mathematica. This is not numerically efficient, but it is necessary to show that the FE^2-10/1-SchurMMA implementation is numerically identical to the FE^2-10/1-Sens. implementation. The imposition of periodic boundary conditions using Lagrange constraints results in the loss of positive definiteness of the tangent matrix and produces zeros along the main diagonal. Some algorithms for the evaluation of the SCHUR complement, such as the one implemented in the Intel MKL library, perform perturbation of the zeros along the main diagonal, resul-

ting in an imprecise SCHUR complement. This imprecision is sufficient to alter (although not significantly) the convergence behavior. This case is shown in the fourth column of Table 3, designated as FE^2-10/1-SchurMKL.

Table 3: Comparison of FE^2 convergence rate for last macro step
Preglednica 3: Primerjava konvergence FE^2 za zadnji makro korak

NR it.	FE^2-10/1 -Sens.	FE^2 -10/5 -Sens.	FE^2 -10/5 -Sens.end	FE^2 -10/1- SchurMKL	FE^2-10/1- SchurMMA	FE^2 -10/5- SchurMMA
1	1.310e-02	1.322e-02	1.322e-02	1.314e-02	1.310e-02	1.322e-02
2	5.014e-03	4.718e-03	4.103e-03	5.128e-03	5.014e-03	4.103e-03
3	2.648e-03	2.561e-03	2.321e-03	2.522e-03	2.648e-03	2.321e-03
4	4.127e-04	4.052e-04	7.800e-04	3.869e-04	4.127e-04	7.800e-04
5	2.557e-05	2.315e-05	2.761e-04	2.245e-05	2.557e-05	2.761e-04
6	1.428e-07	1.151e-07	1.145e-04	2.209e-08	1.428e-07	1.145e-04
7	6.368e-12	3.310e-12	5.403e-05	8.936e-11	6.367e-12	5.403e-05
8	8.720e-16	3.862e-16	2.738e-05	6.266e-13	5.230e-16	2.738e-05
9			1.451e-05			1.451e-05
...			...			...
41			9.287e-14			9.287e-14

5.3 Numerical efficiency of the two-level path-following iterative procedure

The numerical efficiency of the two-level path-following iterative procedure is investigated on an example from Section 5.1. All simulations were performed on a PC with Intel i9 2.8GHz, 16 Core processor and 128GB RAM. Micro problems were solved in parallel on 14 cores. Mathematica was used only as a steering application for parallelization and control of the iterative procedure, while all computationally intensive operations were performed with compiled C codes. A 3D finite strain elasto-plastic material model used is based on an exact exponential map (see e.g. [29]), which is, by itself, computationally intensive. Consequently, the administrative cost turns out to be negligible when compared to the actual computational cost.

The effect of our implementation on the computational time for the FE^2 formulation is

presented in Table 4. An example introduced in Section 5.1 is solved with the macro mesh grid of $20 \times 4 \times 4$ in 10 load steps, with a constant load increment $\Delta\lambda_M = 0.1$. The computational time, normalized with respect to the FE^2-10/1-Sens. Formulation, is presented along with the total number of NEWTON-RAPHSON iterations for all load steps and the total number of micro problems solved during the complete simulation. The simulation using the FE^2-10/1-Sens. formulation took 1968.5 seconds of real time. The results are presented for the $n_m = 1$ and 5, and micro mesh densities of $5 \times 5 \times 5$ and $10 \times 10 \times 10$. The first order sensitivity analysis based formulation is faster than the corresponding SCHUR complement based formulation in all cases. The loss of quadratic convergence of the FE^2-10/5-SchurMKL formulation has the largest influence on speed and results in more iterations per load step. The density of the micro mesh influences the total computational time; however, the relation between the sensitivity-based formulation and the SCHUR complement-based formulation remains the same.

Table 4: Effect of implementation on numerical efficiency of the FE^2 method
Preglednica 4: Vpliv implementacije na numerično učinkovitost FE^2 metode

implementation	micro mesh	normalized time	total NR iterations	total micro problems	total micro problems solved in all it.
FE^2-10/1-Sens.	$5 \times 5 \times 5$	1.0	60	1 000	153 600
FE^2-10/5-Sens.	$5 \times 5 \times 5$	3.6	59	1 000	151 040
FE^2-10/1-SchurMKL	$5 \times 5 \times 5$	1.5	65	1 000	166 400
FE^2-10/5-SchurMKL	$5 \times 5 \times 5$	8.8	136	1 000	348 160
FE^2-10/1-Sens.	$10 \times 10 \times 10$	6.3	60	8 000	153 600
FE^2-10/1-SchurMKL	$10 \times 10 \times 10$	8.5	67	8 000	171 520

The effect of implementation, micro mesh density and material model on computational time is presented for the MIEL formulation in Table 5. The example introduced in Section 5.1 is solved with the macro mesh grid of $10 \times 2 \times 2$ in 5 load steps with a constant load increment $\Delta\lambda_M = 0.2$. The computational time is normalized with respect to the MIEL-5/1-Sens. Formulation, which, for a $5 \times 5 \times 5$ micro mesh grid, took 37 seconds of real time. Two micro material models are considered: a finite strain elasto-plastic model and a hyper-elastic model based on hyper-elastic strain energy (77) as defined in Sec.2.3.3. For a sparse micro mesh ($5 \times 5 \times 5$), the SCHUR complement based formulation is faster than

the second order sensitivity analysis based formulation. The advantages of the sensitivity-based implementation become apparent with denser micro meshes ($30 \times 30 \times 30$). As already shown in Fig. 12, the cost of the calculation of the SCHUR complement grows much faster with the density of the micro mesh than the cost of the second order sensitivity analysis. While the cost of the SCHUR complement does not depend on the material model used, the cost of the sensitivity analysis does. Consequently, the difference between the numerical efficiency of the SCHUR- and sensitivity-based formulations is greater for the hyper-elastic material model than for the elasto-plastic material model.

Table 5: Effect of implementation and material model on normalized time for MIEL
Preglednica 5: Vpliv implementacije in materialnega modela na numerično učinkovitost MIEL

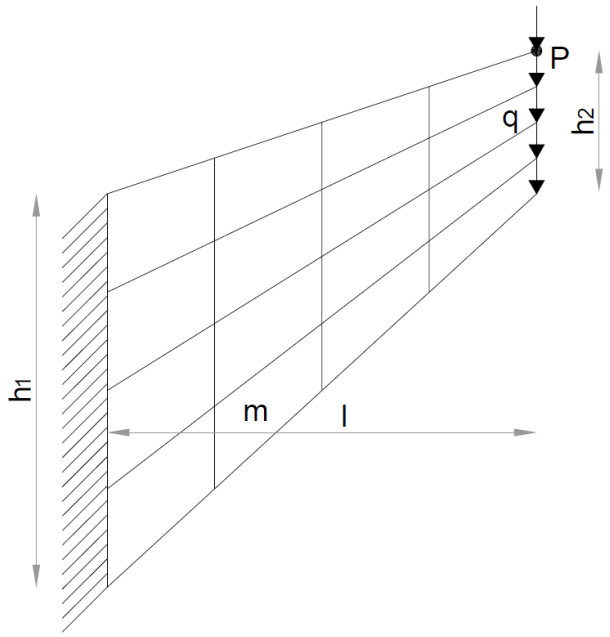
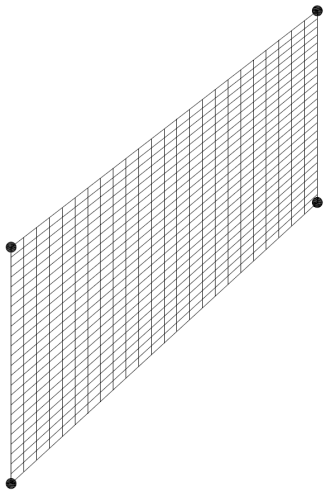
implementation	micro mesh	normalized time hyper-elastic	normalized time elasto-plastic
MIEL-5/1-Sens.	$5 \times 5 \times 5$	1.0	1.8
MIEL-5/1-SchurMKL	$5 \times 5 \times 5$	0.8	1.2
MIEL-5/1-Sens.	$30 \times 30 \times 30$	98.2	287.6
MIEL-5/1-SchurMKL	$30 \times 30 \times 30$	174.0	350.4

5.4 Convergence rates of micro-macro coupling with mesh refinement on Cooks membrane test

The multi-scale MIEL method was tested on the Cook membrane benchmark problem to verify the consistency and efficiency of micro-macro coupling. The homogeneous micro-structure is chosen intentionally for benchmark purposes. The effect of the macro mesh density and the use of different finite elements were investigated. In Table 6, characteristics of a problem at the macro and micro level are described. Geometry, constraints, and load are defined at the macro level, whereas material properties are defined at the micro level. Displacements are fixed on one side, and on the other, a distributed load is added in the vertical direction. Division of the macro mesh varied, while division on the micro level was fixed for all computations. The converged micro level mesh density was used so that results for different macro mesh densities can be compared. For a mesh at macro and micro level Q1, four nodded, isoparametric, quadrilateral, plane strain elements are used with 2×2 GAUSS integration, and for Q2S, eight nodded, isoparametric, quadrilateral,

plane strain elements, were used with 3×3 Gauss integration.

Table 6: Macro and micro problem for MIEL
Preglednica 6: Makro in mikro problem za MIEL

macro problem	characteristic micro problem
Geometry $h_1 = 44$ mm; $h_2 = 16$ mm; $l = 48$ mm; $t = 1$ mm Constraints: $X = 0$: $u = v = 0$ Load: $q = 0.1$ N/mm^2 	Material $E = 1$ N/mm^2 $\nu = 0$ *micro mesh of macro element marked with m 

The coupling of micro and macro scale convergence rates was evaluated by comparing the displacement of the upper right point P on the Cook membrane test. Vertical displacement was compared for different macro mesh densities. For the single-scale analysis, results obtained with linear and quadratic elements are shown. Three combinations were investigated for MIEL: MIEL Q1-Q1 (Q1 elements at both macro and micro levels), MIEL Q2S-Q1 (Q2S element at the macro level and Q1 element at the micro level), and MIEL Q2S-Q2S (Q2S element at the macro level and Q2S element at the micro level). The convergence of the results is faster for MIEL than for the single-scale analysis. The comparison is shown in Fig. 16. The overall convergence of Q2S elements with quadratic

interpolation is faster than with Q1. Results show that, for meshing at the micro level, use of Q2S elements is not preferable, because the small improvement in convergence does not compensate for the increased computational time. In Fig. 17, results for strain Exx in example MIEL Q2S-Q1 are also shown.

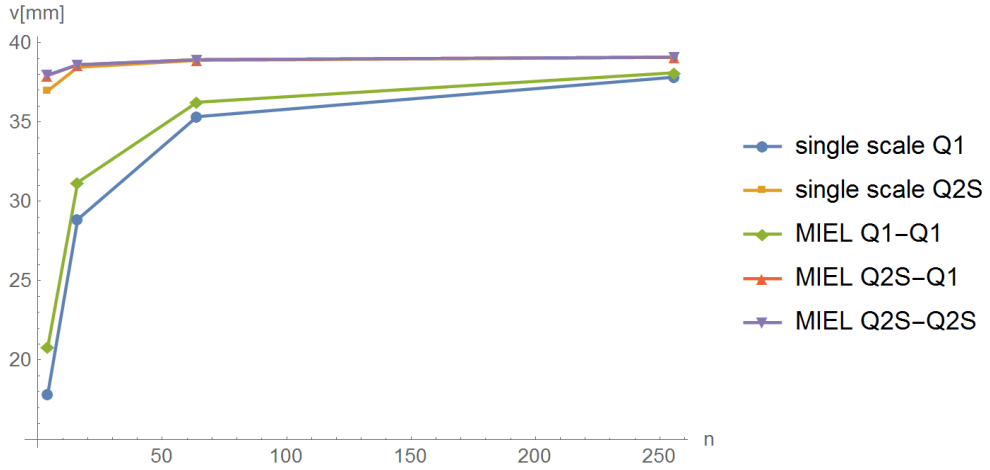


Figure 16: Convergence of result for vertical displacement
Slika 16: Konvergenca rezultata vertikalnega pomika

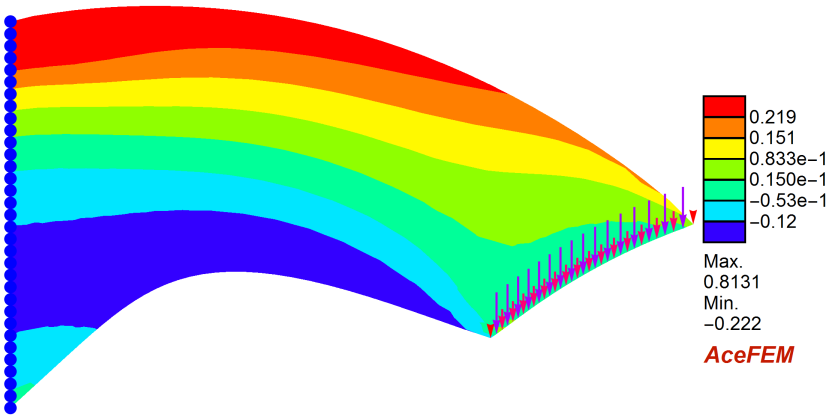


Figure 17: Results for strains Exx
Slika 17: Rezultat deformacij Exx

5.5 Effect of non-linearity of the micro-structure

This example demonstrates how the use of a two-level path-following procedure improves numerical efficiency of a multi-scale simulation in the case of highly nonlinear micro-structure response and relatively monotonic response of the macro-structure. A 2D, plane strain, uni-axial test is simulated using the FE^2 multi-scale method, based on a fully

consistent sensitivity analysis. The macro geometry and mesh together with the RVE geometry and mesh are shown in Fig. 18. The macro domain is discretized with 4×2 macro elements and displacement of $u_{max} = 0.6$ is prescribed at the end. The RVE of a periodic micro-structure is composed of a hyper-elastic rim with material properties $E = 21000$, $\nu = 0.3$ and a narrow elasto-plastic inclusion with properties $E = 21000$, $\nu = 0.3$, $\sigma_{y0} = 24$. The inclusion has an additional small imperfection in the center. At RVE level Q2, nine nodded, isoparametric, quadrilateral, plane strain elements were used, to avoid the locking effect.

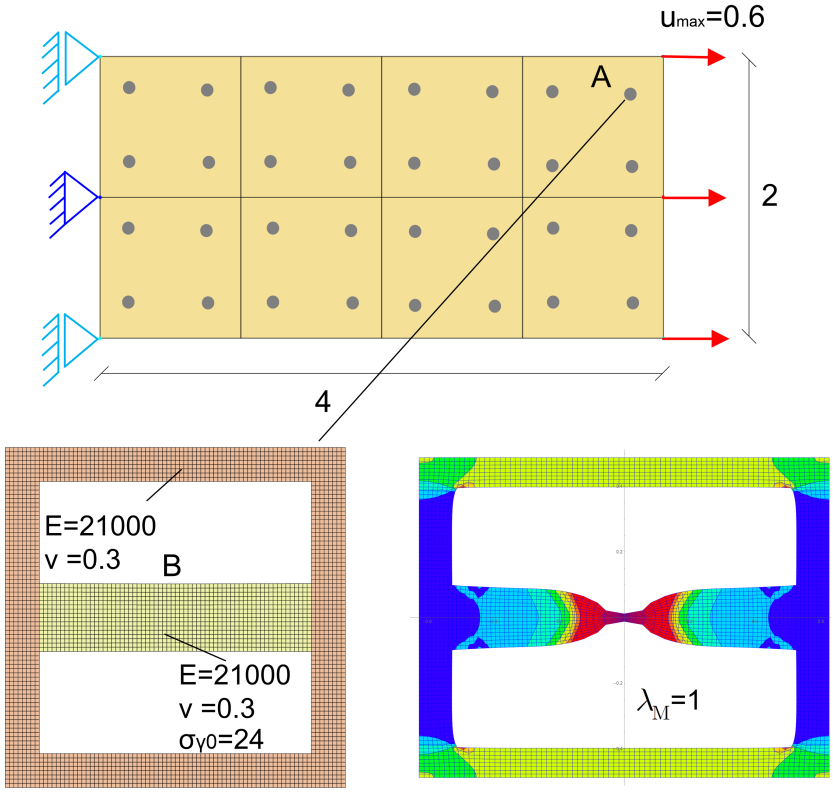


Figure 18: Uni-axial test at macro level, macro mesh and geometry, RVE mesh and geometry, and deformed RVE
Slika 18: Enoosni test na makro nivoju, makro mreža in geometrija, RVE mreža in geometrija in deformiran RVE

At a certain load level, a strongly nonlinear process of necking of the inclusion starts and requires very small load steps. If $n_m = 1$ (FE^2-Adaptive/1-Sens. approach), the maximum micro level load increment limits the macro load increment resulting in small macro steps. Due to the elastic rim, the global response remains relatively unaffected. If an

adaptive path-following procedure is applied at the micro level (FE^2-Adaptive/Adaptive-Sens. approach), significantly larger load steps can be performed at the macro level. The FE^2-Adaptive/Adaptive-Sens. approach requires 13 macro load steps, whereas the FE^2-Adaptive/1-Sens. approach requires 33 macro load steps. Fig. 19 shows the macro reaction force F and Fig. 20 shows the absolute contraction at point B on the micro level with respect to a global load factor λ_M for both cases. The response curves are almost the same for both cases. Thus, an efficient solution for strongly nonlinear problems requires two-level adaptive time-step procedures, where the maximum size of load increments at the micro level determines the overall efficiency of the simulation.

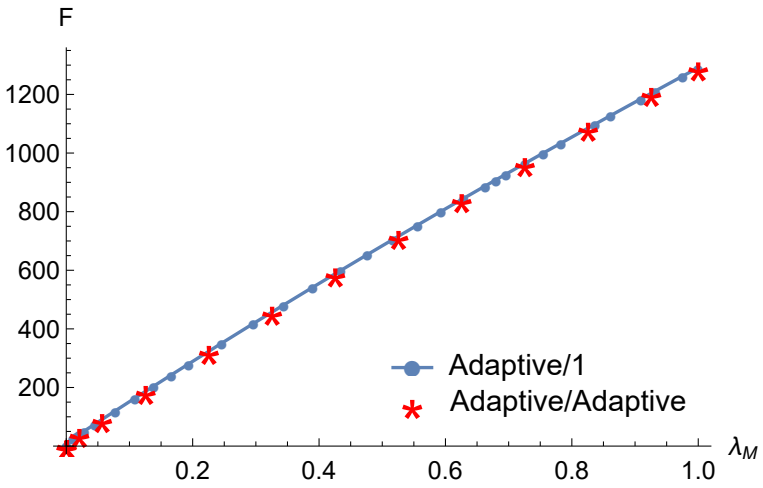


Figure 19: Horizontal residual force F
Slika 19: Horizontalna rezultanta F

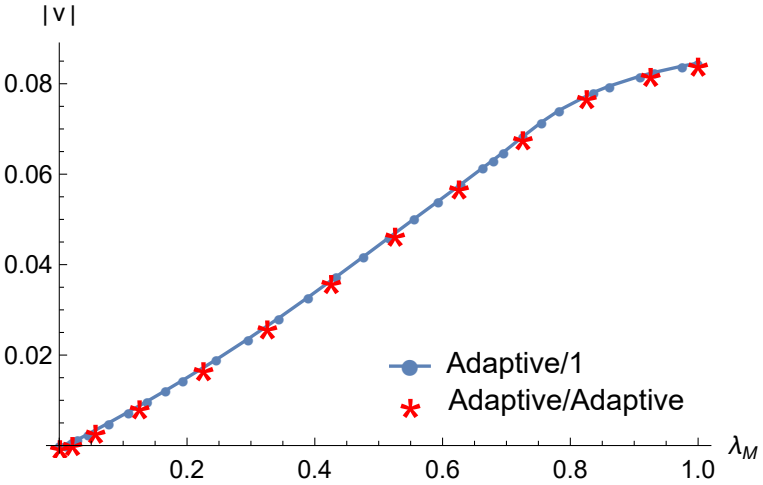


Figure 20: Vertical displacement in point B
Slika 20: Vertikalni pomik v točki B

5.6 Effect of path-dependency of micro-structure

The following example demonstrates how the use of the two-level path-following procedure improves the numerical efficiency of multi-scale simulation in the case of strongly path-dependent problems. The accuracy of the integration of the evolution equations depends on the micro step size, thus limiting the size of the micro load steps. For the $n_m = 1$ case, this also limits the macro load step size, as in the previous example. Again, the two-level adaptive path-following procedure proves to be numerically more efficient than the standard approach, where each macro step is followed by one micro step.

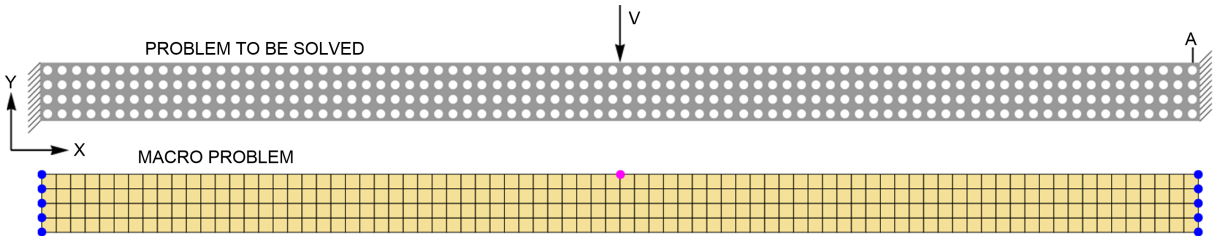


Figure 21: Macro geometry
Slika 21: Makro geometrija

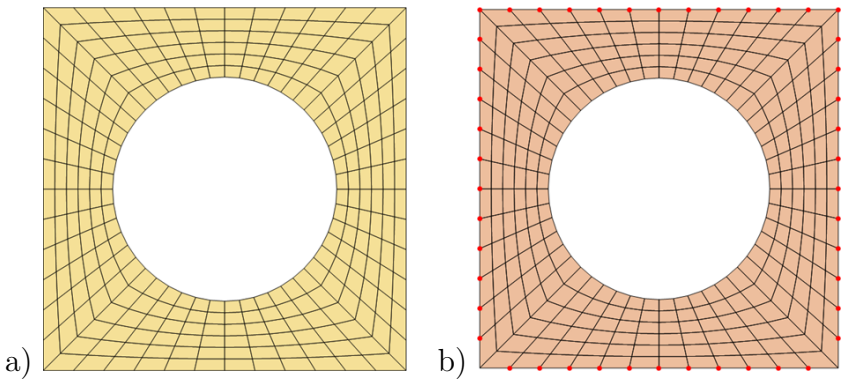


Figure 22: Micro geometry: a) MIEL b)FE^2 RVE
Slika 22: Mikro geometrija: a) MIEL b)FE^2 RVE

A long clamped beam with dimensions 20×1 and a macro mesh division of 80×4 has a prescribed vertical displacement $v_{max} = 0.25$ in the middle, as shown in Fig. 21. The beam is perforated with 320 perforations with a radius such that 30% perforation of the beam is achieved. Perforations are evenly distributed and each perforation is placed at the center of the corresponding macro element. Two cases were investigated. In the first case, a MIEL multi-scale computational scheme was employed. Due to the even distribution of perforations, all the MIEL micro meshes look the same, as shown in

Fig. 22a. In the second case, infinitely small perforations were assumed with the same 30% perforation ratio. The second case is simulated with the FE^2 scheme with RVE, as depicted in Fig. 22b. The RVE mesh is identical to the MIEL micro mesh due to the evenly distributed perforations. Eight nodded, isoparametric, quadrilateral, plane strain elements were used with 3×3 GAUSS integration. A finite strain elasto-plastic material model as described in Section 2.3.3 is used at the micro level. In both cases, material properties of the micro-structure are $E = 21000$, $\nu = 0.3$, $\sigma_{y0} = 24$, $K_h = 21$, $R_\infty = 12$ and $\delta = 30$. For various solution strategies, the value of the strain tensor component E_{xy} in point A is compared for the MIEL scheme in Figs. 23 and 24, and for the FE^2 scheme in Figs. 25 and 26.

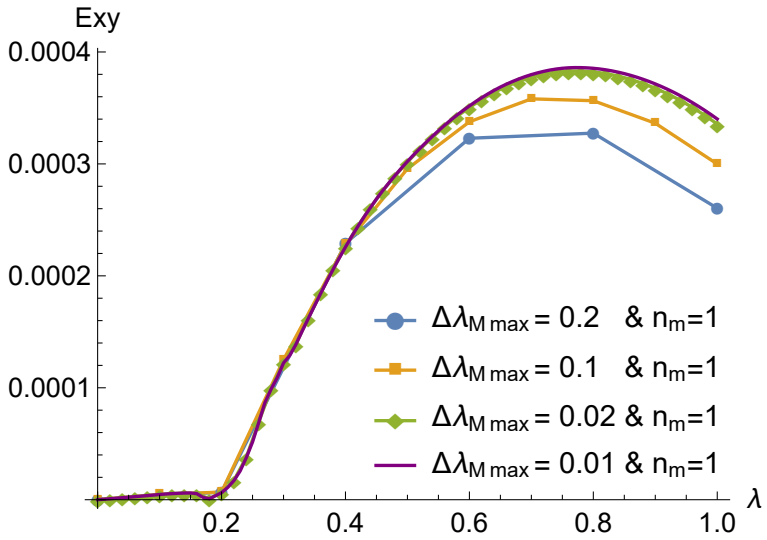


Figure 23: E_{xy} with respect to $\Delta\lambda_{M\,max}$ for the MIEL-Adaptive/1-Sens. scheme
Slika 23: E_{xy} glede na $\Delta\lambda_{M\,max}$ za MIEL-Adaptive/1-Sens. shemo

In Fig. 23, the response curve $E^A_{xy}(\lambda_M)$ is shown for the MIEL-Adaptive/1-Sens. approach with different prescribed maximal sizes of the macro load step $\Delta\lambda_{M\,max}$, using an adaptive time step at the macro level and one micro step per each macro step. A converged solution is achieved for $\Delta\lambda_{M\,max} = 0.01$. Secondly, Fig. 24 displays the results for fixed $\Delta\lambda_{M\,max} = 0.2$ and 1, 2, 5 and 10 micro steps for each macro step. The evolution equation integration error is significantly reduced with the increased number of micro steps, without the need for costly additional macro steps. There is, of course, a limit to which additional micro steps can improve the overall results, as shown in Fig. 24.

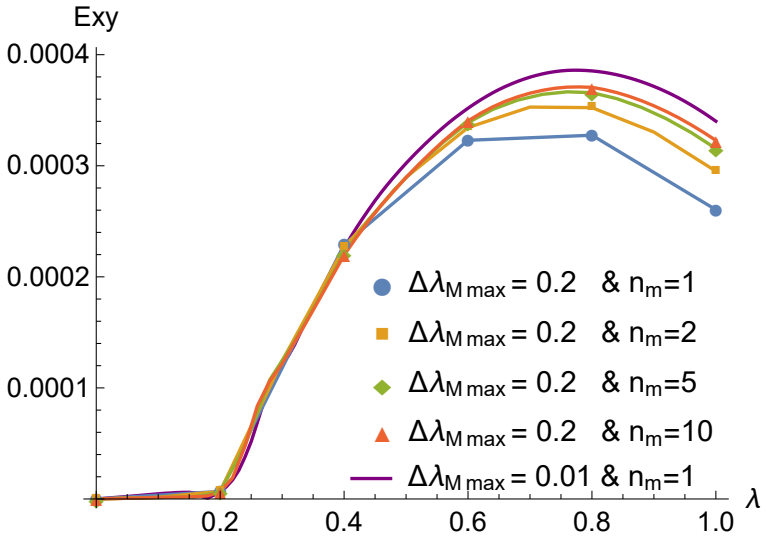


Figure 24: E_{xy} with respect to number of micro steps for the MIEL-Adaptive/n_m-Sens. scheme
Slika 24: E_{xy} glede na število mikro korakov za MIEL-Adaptive/n_m-Sens. shemo

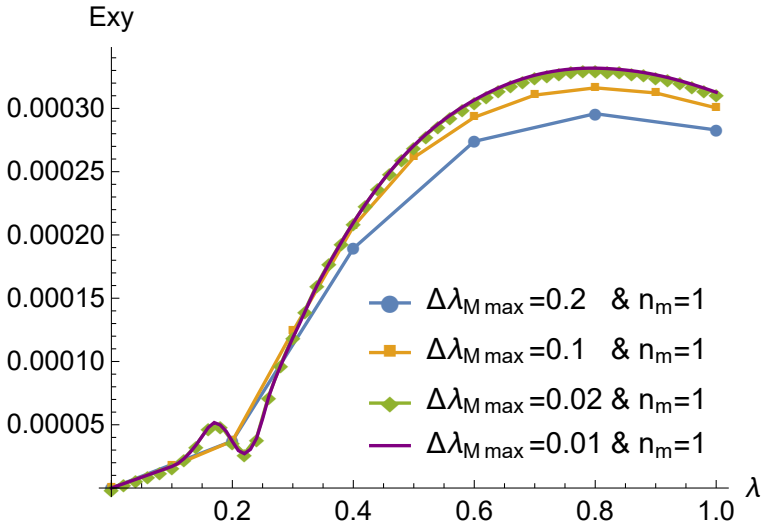


Figure 25: E_{xy} with respect to $\Delta\lambda_{M\,max}$ for the FE^2-Adaptive/1-Sens. scheme
Slika 25: FE^2-Adaptive/1-Sens., E_{xy} glede na $\Delta\lambda_{M\,max}$ za FE^2-Adaptive/1-Sens. shemo

Results for the FE^2 scheme are compared in the same way as for the MIEL scheme. The response curve $E^A_{xy}(\lambda_M)$ is shown for FE^2-Adaptive/1-Sens. with respect to the prescribed maximal size of the macro load step $\Delta\lambda_{M\,max}$ in Fig. 25 and for $\Delta\lambda_{M\,max} = 0.2$ and in Fig. 26, with different number of micro steps. An adaptive time step was used at the macro level in all cases. Conclusions are the same as for MIEL. With a two-level path-following scheme, the same accuracy is achieved with 20-times fewer macro steps. With

additional micro steps, the method was able also to capture fine details of the response curve near $\lambda_M = 0.2$. Point A is in the corner, close to the boundary where deformation gradients are high. Consequently, the converged curve $E^A_{xy}(\lambda_M)$ is different for the MIEL and the FE^2 scheme.

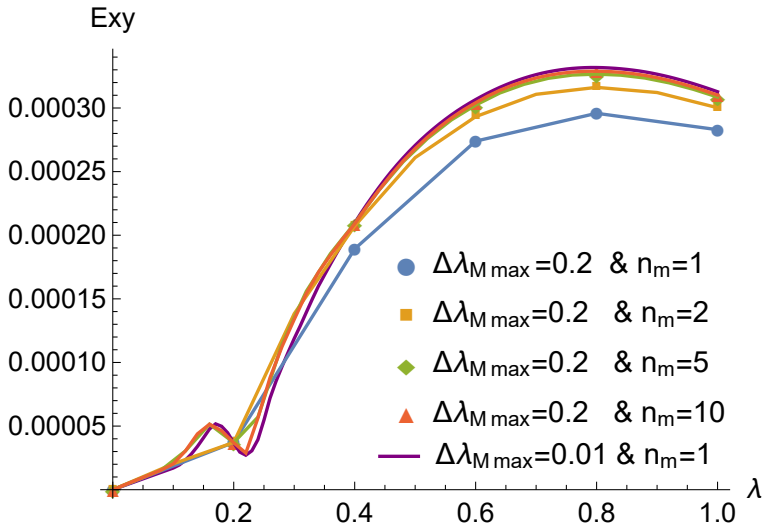


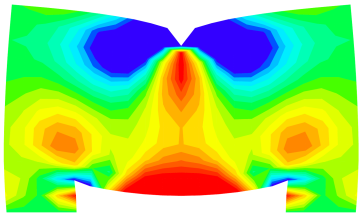
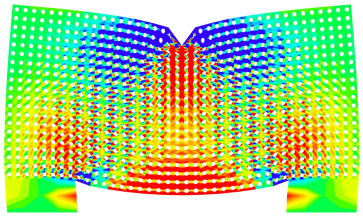
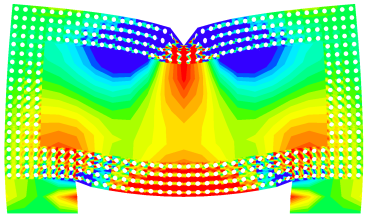
Figure 26: E_{xy} with respect to number of micro steps for the FE^2-Adaptive/n_m-Sens. scheme
Slika 26: FE^2-Adaptive/n_m-Sens., E_{xy} glede na število mikro korakov za FE^2-Adaptive/n_m-Sens. shemo

5.7 Example with mixed MIEL/FE^2/single-scale methods

As a numerical example, the bending of a beam with enforced vertical displacement was investigated. In Table 7, different combinations of multi-scale schemes are presented. In the mixed scheme (presented also in Fig. 27) all FE^2, MIEL and single-scale methods are used in one model. MIEL is used on the outer rim and FE^2 on the inside, where the periodicity of the openings is assumed. In the first two examples only one method, either FE^2 or MIEL is used. In all cases, supports are modeled with 16 macro solid elements. For these three combinations in Table 7, the numbers of micro and macro elements and the total DOF are compared. In the case of MIEL, the number of micro problems is equal to the number of macro elements, whereas for FE^2 the number of micro problems for one macro element is equal to the number of integration points, in this case, four. The total DOF represents the count of equations that need to be solved and is largest when the FE^2 method is used alone. In the example, a Neo-Hookean type hyper-elastic material model was used. The computational times for different schemes are compared in Table 7,

and we can conclude that combining the two presented multi-scale methods can reduce the level of computational demand if we assume that the accuracy of the results is still agreeable. The distribution of strain Exx for the mixed case is shown in Fig. 28.

Table 7: Comparison of computational times for selected combinations
Preglednica 7: Primerjava računskih časov za izbrane kombinacije

FE^2	MIEL	mixed
		

	FE^2	MIEL	mixed
No. micro problems	800	200	488
No. macro elements	216	216	216
total DOF	1.030.881	132.881	563.921
total time	192.4 s	50.2 s	127.3 s

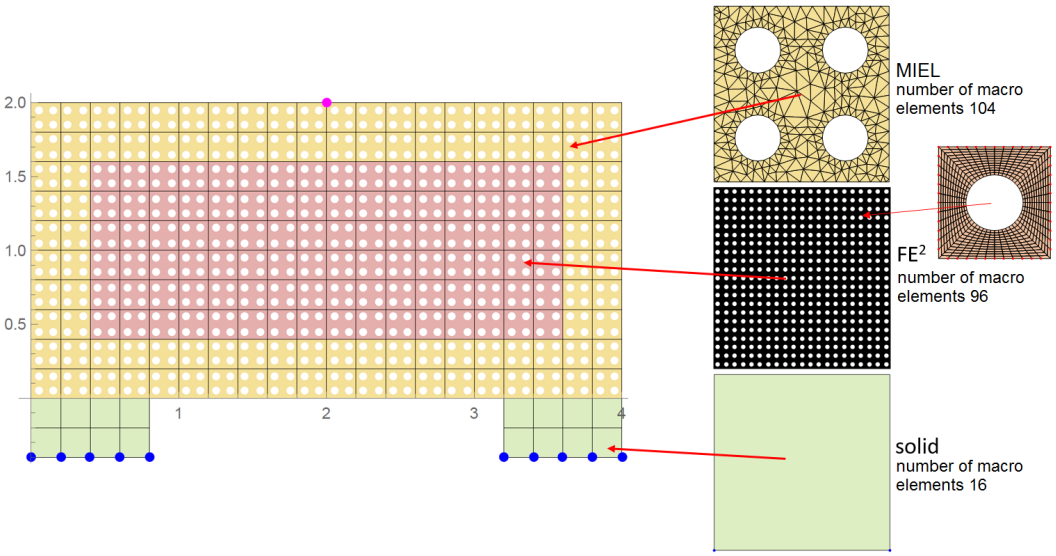


Figure 27: Mixed multi-scale model
Slika 27: Mešan večnivojski model

In the two-level path-following algorithm, the linearization must be carried out correctly. For path-dependent problems, only full sensitivity analysis at the micro level leads to the macro tangent matrix, which is algorithmically consistent and leads to a quadratically convergent scheme. When sensitivity equations are resolved at the end of the micro step (which is equivalent to SCHUR complement based implementation) the quadratic convergence of the NR method is lost.

Table 8: Comparison of convergences for FE^2 scheme
Preglednica 8: Primerjava konvergence za FE^2 shemo

it.	hyper-elastic FE^2-1/1 -Sens.	hyper-elastic FE^2-1/5 -Sens.end	elasto-plastic FE^2-1/1 -Sens.	elasto-plastic FE^2-1/5 -Sens.	elasto-plastic FE^2-1/5 -Sens.end/SCHUR
1	9.79e-04	9.79e-04	8.92e-04	8.92e-04	8.92e-04
2	2.18e-07	2.18e-07	2.30e-04	2.30e-04	2.30e-04
3	8.45e-14	8.45e-14	2.01e-05	1.76e-05	1.81e-05
4	/	/	2.21e-08	2.10e-08	2.57e-06
5	/	/	1.45e-13	4.68e-14	6.11e-07
6	/	/	/	/	1.53e-07
-	-	-	-	-	-
15	/	/	/	/	7.37e-13

Table 9: Comparison of convergences for MIEL scheme
Preglednica 9: Primerjava konvergence za MIEL shemo

it.	hyper-elastic MIEL-1/1 -Sens.	hyper-elastic MIEL-1/5 -Sens.end	elasto-plastic MIEL-1/1 -Sens.	elasto-plastic MIEL-1/5 -Sens.	elasto-plastic MIEL-1/5 -Sens.end/SCHUR
1	9.80e-04	9.80e-04	1.76e-03	1.76e-03	1.76e-03
2	2.21e-07	2.21e-07	1.48e-04	1.52e-04	2.03e-04
3	8.23e-14	8.20e-14	1.12e-04	1.12e-04	1.67e-04
4	/	/	5.78e-07	8.86e-07	6.03e-05
5	/	/	5.47e-11	9.52e-11	1.83e-05
6	/	/	/	/	1.53e-07
-	-	-	-	-	-
16	/	/	/	/	4.33e-11

Next, a finite strain plasticity material model was used. Since the elasto-plastic problems are path-dependent, the influence of the implementation of the two-level path-following procedure on the convergence rate of the Newton-Raphson iterative procedure was additionally investigated.

Table 10: Comparison of convergences for mixed FE^2 and MIEL model
Preglednica 10: Primerjava konvergence za mešan FE^2 in MIEL model

it.	hyper-elastic MIX-1/1 -Sens.	hyper-elastic MIX-1/5 -Sens.end	elasto-plastic MIX-1/1 -Sens.	elasto-plastic MIX-1/5 -Sens.	elasto-plastic MIX-1/5 -Sens.end/SCHUR
1	1.01e-03	1.01e-03	1.54e-03	1.54e-03	1.54e-03
2	2.38e-07	2.38e-07	24.10e-04	4.12e-04	5.00e-04
3	1.47e-13	1.48e-13	1.95e-04	1.92e-04	1.38e-04
4	/	/	4.26e-06	3.98e-06	1.38e-05
5	/	/	1.33e-09	1.58e-09	5.51e-07
6	/	/	1.25e-16	9.74e-17	2.99e-08
7	/	/	/	/	1.51e-09
-	-	-	-	-	-
10	/	/	/	/	2.02e-13

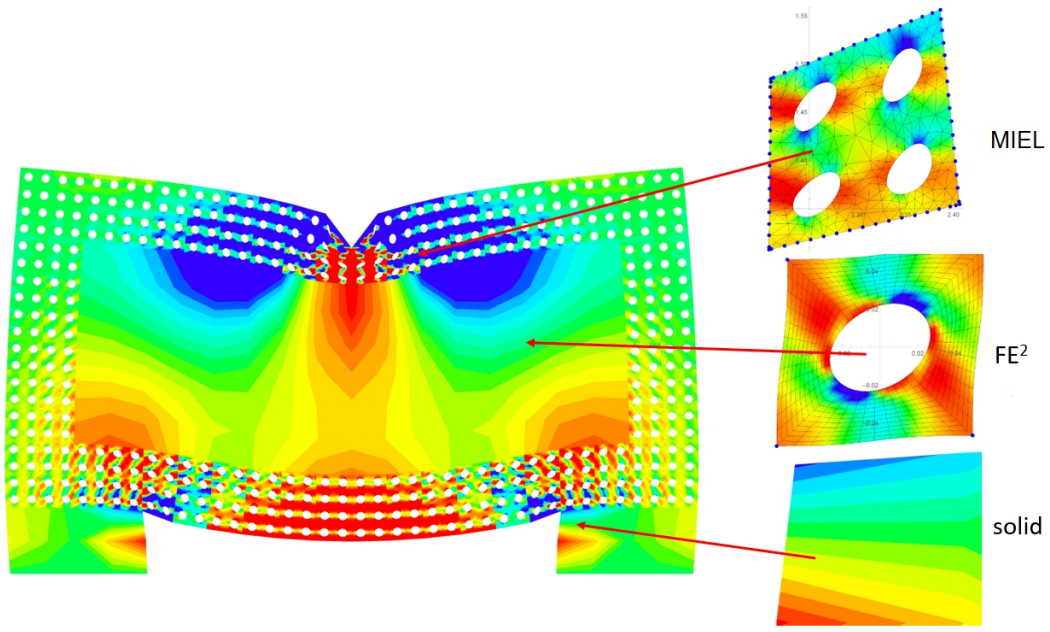


Figure 28: Results for strains Exx for mixed scheme
Slika 28: Rezultati deformacije Exx za mešan primer

For hyper-elastic materials, the convergence rate is quadratic regardless of the number of micro sub-steps, because the problem is not path-dependent. For elasto-plastic materials, the quadratic convergence is lost, unless full sensitivity is evaluated (elasto-plastic MIEL-1/5-Sens.end/SCHUR). The same result is also obtained for MIEL and mixed schemes. Results for the MIEL and mixed models are presented in Tables 9 and 10.

6 MULTI-SCALE OPTIMIZATION ALGORITHM

6.1 Structural optimization

Structural optimization is a discipline dealing with the optimal design of the load-carrying structures. Over the past decades, structural optimization has emerged as an important tool in the design process. The objective of the optimization can be to minimize stresses, weight or compliance for a given amount of material and boundary conditions. Optimization must be constrained to obtain a problem with a well-defined solution. Quantities that are usually constrained are stress, displacement, and geometry. The optimization method can be utilized to design engineering structures, as well as to tailor microstructures.

Optimization methods can be divided into discrete or continuous based on representation, and into deterministic or stochastic based on search type[5]. Moreover, based on which geometrical feature is parameterized, structural optimization can be classified into the following:

- Size optimization, where the design variable represents the size of the cross-section for discrete structural members, such as beams and columns, or thickness of the continuous material, such as panels. The optimal size or thickness typically minimizes some physical quantity such as the strain energy or the deflection, while the equilibrium constraint has to be fulfilled. Size optimization problems can easily be expressed mathematically and are traditionally solved by deterministic methods (e.g. Fully Stressed Design).

- Shape optimization, where the design variable represents the positioning of nodes or connections and definition of lines, curves and surfaces that describe structural form, varying the boundary of the considered domain so that some physical quantity is minimized.

- Topology optimization is the most general form of structural optimization. The design variable defines the connectivity of the domain, varying the configuration and connectivity of members or material. It involves features such as the number and sizes of holes in the design domain. Optimization algorithms for topology optimization can be evolutionary or gradient-based. Evolution algorithms are appropriate for small scale linear problems. For large scale nonlinear problems, gradient-based algorithms are necessary.

Different optimization approaches can be combined to achieve best results. An effective method is to combine topology optimization, for determination of the material distribution, with shape optimization for determination of the final optimal shape.

A continuum-based optimization method considers a continuous designable domain, discretized into a mesh of elements that are defined individually in a structural analysis model. The properties of the continuum elements, such as porosity or thickness, can be varied individually for size optimization, or they can be removed or considered to have vanishing thickness for shape and topology optimization. As an example, the homogenization method defines individual materials for each element in the mesh, each containing infinite microscopic voids. The porosity of the medium is optimized according to some objective function. Each element material may have its own hole-size and orientation.

Differences between various types of optimization are related to how design variables affect the analysis. In size optimization, the design variables are related to the properties of the finite elements, while in shape optimization the design variables are related to the positions of the finite element nodes and therefore directly affect the implementation within the structural analysis.

6.2 Gradient-based optimization

Structural optimization problems concerning the limit load shape optimization of real-world structures are usually nonlinear, constrained, and the relevant constraints are not known in advance. The solution of such problems requires a mathematical programming method that is capable of determining the optimum solution through an iterative approach in the minimum possible number of iterations. Firstly, a search direction within the design space is calculated, followed by the determination of the step size. A wide range of techniques exist for determining the search direction and step-size, according to the assumed characteristics of the objective function, constraints, and variables.

In general, we can separate local and global optimization algorithms, where local optimization algorithms only find the local optimum. On the other hand, global optimization algorithms attempt to find the global optimum, typically by allowing the objective function to decrease, as well as increase. A local minimum may not be a global minimum, but a global minimum is always also a local minimum. Moreover, methods can be divided into methods that solve constrained and unconstrained problems. Depending on the type of functions involved there are linear and nonlinear optimization problems. Additionally,

optimization algorithms can be divided into numeric and symbolic algorithms.

Numerical algorithms for nonlinear optimization can be broadly categorized into direct search methods and gradient-based methods. Direct methods use only the function values without derivative information, such as Nelder-Mead, genetic algorithm and differential evolution, and simulated annealing. Direct search methods tend to converge more slowly, but can be more tolerant to the presence of noise in the function and constraints. Evolutionary Algorithms (EAs) in topology optimization are versatile, stochastic, problem-solving methods, including genetic algorithms (GA), genetic programming, evolutionary programming (EP) and evolutionary strategies (ES). The most widely used numerical scheme for topology optimization is the solid isotropic material with penalization (SIMP) scheme, where the density is approximated as constant within each element. In the evolutionary structural optimization (ESO) method, under-utilized elements, as defined by some metric such as the strain energy density, are removed from a continuous finite element mesh to reduce the designable domain to an efficient optimal topology. There have been several additions to the basic ESO method. In gradient-based methods, first derivatives, and gradient information is also used. In Newton-type methods, second order derivatives (also known as Hessian) are used. Examples of Newton-type methods include the sequential quadratic programming (SQP) method, the augmented Lagrangian method, and the Interior point method.

An important subset of optimization problems is constrained nonlinear optimization, where the function is not linear and the parameter values are constrained to certain regions. In Mathematica, the only method currently available for such cases is the Interior point algorithm. The Interior point algorithm solves a constrained optimization problem by combining constraints and the objective function through the use of the barrier function.

Definition of nonlinear optimization problem

Find vector

$$\boldsymbol{\alpha} = (\alpha_1, \alpha_2, ..., \alpha_{\mathrm{n}}) \in \mathbb{R}^{\mathrm{n}} \tag{176}$$

for which, objective function $\mathrm{F}(\boldsymbol{\alpha})$ will have minimal value, while subjected to equality and inequality constraints:

$$g_j(\boldsymbol{\alpha}) = 0, j \in P = \{1, ..., p\} \tag{177}$$

$$h_k(\boldsymbol{\alpha}) \geqslant 0, k \in Q = \{1, ..., q\} \tag{178}$$

where F, g and h are smooth and continuously differentiable, and at least one of them is nonlinear. Set of acceptable solutions is noted with M:

$$M = \{\boldsymbol{\alpha} \in \mathbb{R}^n : g_j(\boldsymbol{\alpha}) = 0, j \in P, h_k(\boldsymbol{\alpha}) \geqslant 0, k \in Q\}\,. \tag{179}$$

Vector $\boldsymbol{\alpha}$ is a vector with design parameters α_i.

Constrained optimization problem can be in general transferred to unconstrained by the use of penalty function or Lagrange multiplier type of methods.

6.3 Multi-scale gradient-based optimization

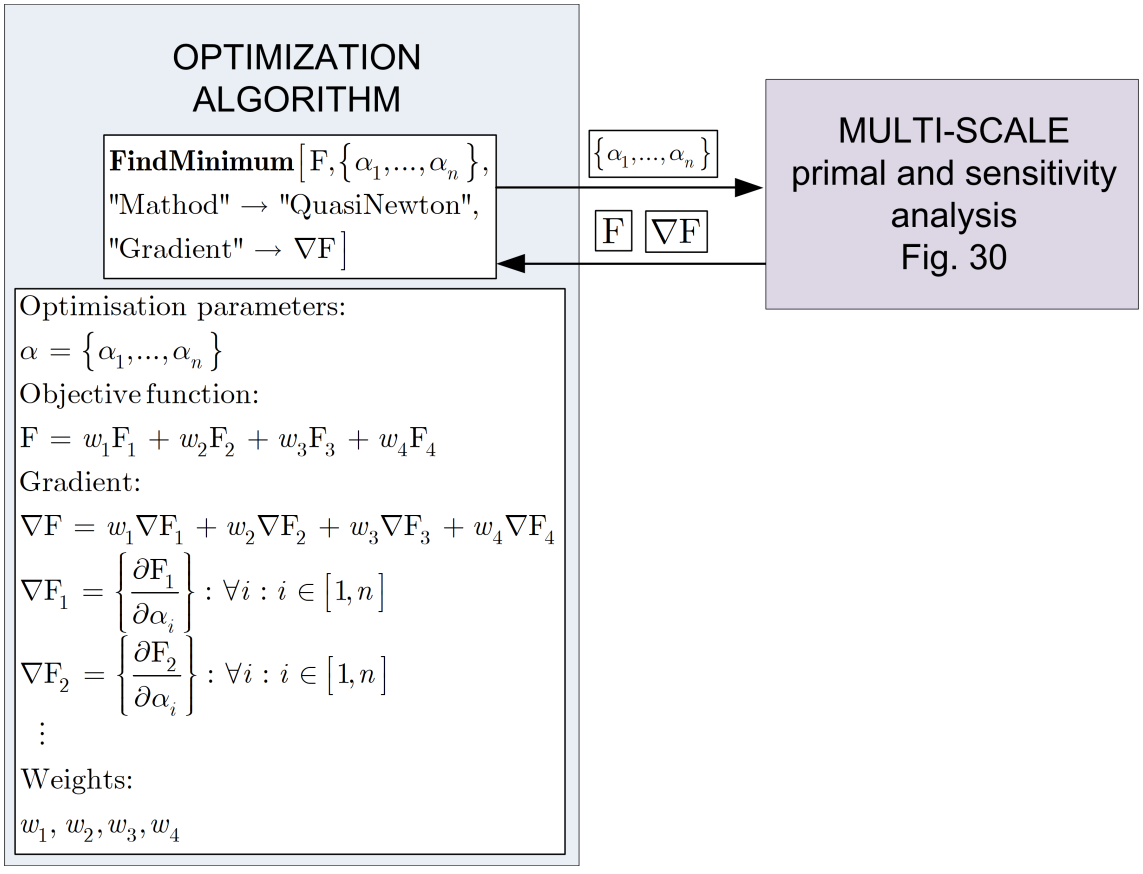


Figure 29: Optimization algorithm
Slika 29: Optimizacijski algoritem

Fig. 29 illustrates multi-scale gradient-based optimization algorithm. Optimal values of optimization parameters α_i are sought for minimal value of defined objective function F with the use of `FindMinimum` function implemented in Mathematica. Unconstrained

algorithm with Quasi-Newton method was chosen which requires for gradient information ∇F to be provided. Sensitivity analysis is required to calculate the gradient. Firstly, the objective function F is defined. Additionally, penalty functions are added to prevent optimization parameters to get out of prescribed boundaries. The objective function and its gradient are evaluated for optimization parameters after primal and sensitivity analysis are done at the macro level. Multi-scale EBC sensitivity is combined with optimization sensitivity analysis. At the micro level we have EBC sensitivity with $\boldsymbol{\phi}$ as a set of sensitivity parameters. Optimization sensitivity parameters are $\boldsymbol{\alpha}$, thus a complete set of sensitivity parameters at the micro level is

$$\boldsymbol{\phi} \cup \boldsymbol{\alpha} \,. \tag{180}$$

Since sensitivity analysis is regarding parameters concurrent, the two sets can be evaluated together within one sensitivity analysis. The only difference is that multi-scale sensitivity is relative (see Section 3.1) while optimization sensitivity is absolute. Thus the history variables for optimization sensitivity must not be set to zero at the start of each micro increment. Multi-scale sensitivity appears only at the micro level while optimization sensitivity requires both levels, macro and micro to be properly formulated. From the algorithmic point of view, the multi-scale approach corresponds to the sensitivity of locally coupled systems and the solution presented in [30] for the sensitivity analysis of locally coupled problems can be directly applied.

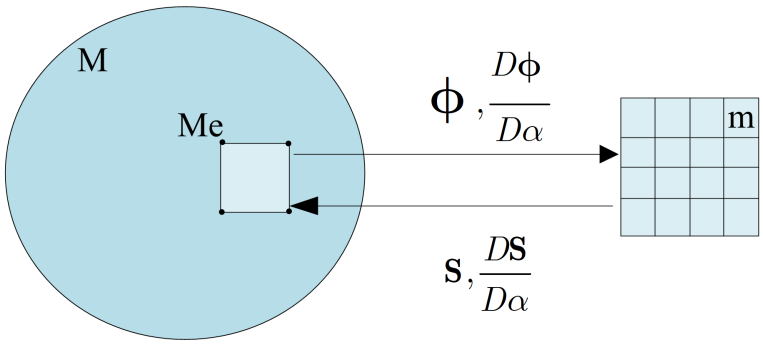


Figure 30: Transfer of data between macro and micro level for optimization
Slika 30: Prenos podatkov med makro in mikro nivojem za optimizacijo

In Fig. 30 transfer of quantities between macro and micro level is shown. From macro finite element M_e, $\boldsymbol{\phi}$ and $\frac{D\boldsymbol{\phi}}{D\boldsymbol{\alpha}}$, their derivatives with respect to optimization parameters are sent to micro problem m. From micro problem, a set of variables $\mathbf{S}$ with their derivatives with respect to optimization parameters $\frac{D\mathbf{S}}{D\boldsymbol{\alpha}}$ is sent back to macro element. Macro residual $\mathbf{R}_M$, micro residual $\mathbf{R}_m$ as defined by (88) and (91), and the vector of unknowns $\mathbf{p}_M$, and

$\mathbf{p}_m$ are expressed as a function of optimization parameters $\boldsymbol{\alpha}$

$$\mathbf{R}_M(\mathbf{p}_M(\boldsymbol{\alpha}), \bigcup_{e=1}^{n_{Me}} \mathbf{S}_{Me}(\mathbf{p}_m(\boldsymbol{\alpha})), \boldsymbol{\alpha}) = \mathbf{0} \ , \tag{181}$$

$$\mathbf{R}_m(\mathbf{p}_m(\boldsymbol{\alpha}), \bar{\mathbf{p}}_m(\boldsymbol{\phi}(\mathbf{p}_M(\boldsymbol{\alpha}))), \boldsymbol{\alpha}) = \mathbf{0} \quad m = 1, ..., n \ . \tag{182}$$

By differentiation of (181) and (182) with respect to i-th sensitivity parameter α_i we get

$$\frac{\partial \mathbf{R}_M}{\partial \mathbf{p}_M} \frac{D\mathbf{p}_M}{D\alpha_i} + \sum_e \frac{\partial \mathbf{R}_M}{\partial \mathbf{S}_{Me}} \frac{\partial \mathbf{S}_{Me}}{\partial \mathbf{p}_m} \frac{\partial \mathbf{p}_m}{\partial \alpha_i} + \frac{\partial \mathbf{R}_M}{\partial \alpha_i} = \mathbf{0} \ , \tag{183}$$

$$\frac{\partial \mathbf{R}_m}{\partial \mathbf{p}_m} \frac{D\mathbf{p}_m}{D\alpha_i} + \frac{\partial \mathbf{R}_m}{\partial \bar{\mathbf{p}}_m} \frac{\partial \bar{\mathbf{p}}_m}{\partial \boldsymbol{\phi}} \frac{\partial \boldsymbol{\phi}}{\partial \mathbf{p}_M} \frac{D\mathbf{p}_M}{D\alpha_i} + \frac{\partial \mathbf{R}_m}{\partial \alpha_i} = \mathbf{0} \quad m = 1, ..., n \tag{184}$$

where $\frac{\partial \mathbf{R}_m}{\partial \mathbf{p}_m} = \mathbf{K}_m$ and from (184) the unknown micro sensitivities $\frac{D\mathbf{p}_m}{D\alpha_i}$ can be expressed

$$\frac{D\mathbf{p}_m}{D\alpha_i} = -{\mathbf{K}_m}^{-1} \left(\frac{\partial \mathbf{R}_m}{\partial \bar{\mathbf{p}}_m} \frac{\partial \bar{\mathbf{p}}_m}{\partial \boldsymbol{\phi}} \frac{\partial \boldsymbol{\phi}}{\partial \mathbf{p}_M} \frac{D\mathbf{p}_M}{D\alpha_i} + \frac{\partial \mathbf{R}_m}{\partial \alpha_i} \right) \ . \tag{185}$$

After inserting $\frac{D\mathbf{p}_m}{D\alpha_i}$ into (183) and after rearrangement, in which the terms that contain the unknown macro sensitivities $\frac{D\mathbf{p}_M}{D\alpha_i}$ are collected together, a system of linear equations is obtained

$$\begin{aligned} &\left(\frac{\partial \mathbf{R}_M}{\partial \mathbf{p}_M} - \sum_e \frac{\partial \mathbf{R}_M}{\partial \mathbf{S}_{Me}} \frac{\partial \mathbf{S}_{Me}}{\partial \mathbf{p}_m} {\mathbf{K}_m}^{-1} \frac{\partial \mathbf{R}_m}{\partial \bar{\mathbf{p}}_m} \frac{\partial \bar{\mathbf{p}}_m}{\partial \boldsymbol{\phi}} \frac{\partial \boldsymbol{\phi}}{\partial \mathbf{p}_M} \right) \frac{D\mathbf{p}_M}{D\alpha_i} \\ &= -\left(\frac{\partial \mathbf{R}_M}{\partial \alpha_i} - \sum_e \frac{\partial \mathbf{R}_M}{\partial \mathbf{S}_{Me}} \frac{\partial \mathbf{S}_{Me}}{\partial \mathbf{p}_m} {\mathbf{K}_m}^{-1} \frac{\partial \mathbf{R}_m}{\partial \alpha_i} \right) \end{aligned} \tag{186}$$

$$\mathbf{K}_M = \left(\frac{\partial \mathbf{R}_m}{\partial \mathbf{p}_M} - \sum_e \frac{\partial \mathbf{R}_M}{\partial \mathbf{S}_{Me}} \frac{\partial \mathbf{S}_{Me}}{\partial \mathbf{p}_m} {\mathbf{K}_m}^{-1} \frac{\partial \mathbf{R}_m}{\partial \bar{\mathbf{p}}_m} \frac{\partial \bar{\mathbf{p}}_m}{\partial \boldsymbol{\phi}} \frac{\partial \boldsymbol{\phi}}{\partial \mathbf{p}_M} \right) \tag{187}$$

$${}^I\hat{\mathbf{R}}_M = \left(\frac{\partial \mathbf{R}_M}{\partial \alpha_i} - \sum_e \frac{\partial \mathbf{R}_M}{\partial \mathbf{S}_{Me}} \frac{\partial \mathbf{S}_{Me}}{\partial \mathbf{p}_m} {\mathbf{K}_m}^{-1} \frac{\partial \mathbf{R}_m}{\partial \alpha_i} \right) \tag{188}$$

$$\mathbf{K}_M \frac{D\mathbf{p}_M}{D\alpha_i} = -{}^I\hat{\mathbf{R}}_M \ , \tag{189}$$

where ${}^I\hat{\mathbf{R}}_M$ is a macro sensitivity pseudo-load vector and $\mathbf{K}_M$ is macro tangent matrix. *AceGen* input segment of finite element for sensitivity analysis at the macro level is presented in Appendix A.5. Linear system of equations for micro level sensitivities $\frac{D\mathbf{p}_m}{D\alpha_i}$ in

(185) is written as

$$\mathbf{K}_m \frac{D\mathbf{p}_m}{D\alpha_i} = -{}^{I}\hat{\mathbf{R}}_m \,. \tag{190}$$

After considering equality

$$\frac{D\boldsymbol{\phi}}{D\alpha_i} = \frac{\partial \boldsymbol{\phi}}{\partial \mathbf{p}_M} \frac{D\mathbf{p}_M}{D\alpha_i} \tag{191}$$

micro sensitivity pseudo-load vector ${}^{I}\hat{\mathbf{R}}_m$ can be expressed from (185) as

$${}^{I}\hat{\mathbf{R}}_m = \frac{\partial \mathbf{R}_m}{\partial \bar{\mathbf{p}}_m} \frac{\partial \bar{\mathbf{p}}_m}{\partial \boldsymbol{\phi}} \frac{D\boldsymbol{\phi}}{D\alpha_i} + \frac{\partial \mathbf{R}_m}{\partial \alpha_i} \,. \tag{192}$$

After taking into account equality

$$\frac{\partial \mathbf{S}_{Me}}{\partial \mathbf{p}_m} {\mathbf{K}_m}^{-1} \frac{\partial \mathbf{R}_m}{\partial \alpha_i} = \frac{\partial \mathbf{S}_{Me}}{\partial \mathbf{p}_m} (-\frac{D\overset{\circ}{\mathbf{p}}_m}{D\alpha_i}) = -\frac{D\mathbf{S}_{Me}}{D\alpha_i} \tag{193}$$

in which the tensor $\frac{D\overset{\circ}{\mathbf{p}}_m}{D\alpha_i}$ represents directional derivative of $\mathbf{p}_m$ with respect to α_i, where variation of input parameter of micro analysis with respect to α_i is neglected. A macro sensitivity pseudo-load vector ${}^{I}\hat{\mathbf{R}}_M$ can be rewritten as

$${}^{I}\hat{\mathbf{R}}_M = \frac{\partial \mathbf{R}_M}{\partial \alpha_i} + \sum_e \frac{\partial \mathbf{R}_M}{\partial \mathbf{S}_{Me}} \frac{D\mathbf{S}_{Me}}{D\alpha_i} \,, \tag{194}$$

where $\frac{D\mathbf{S}_{Me}}{D\alpha_i} = \bigcup_{e=1}^{n_{Me}} \frac{D\mathbf{S}}{D\alpha_i}$ is evaluated at the micro level and transferred to the macro level. $\mathbf{S}_{Me}$ is a set of variables transferred from the micro problems to e-th macro element and calculation of total derivative $\frac{D\mathbf{S}_{Me}}{D\alpha_i}$ has to be done at the micro level. ADB form of element contributions to micro and macro sensitivity pseudo-load vector leads to

$${}^{I}\hat{\mathbf{R}}_{me} = \frac{\partial \mathbf{R}_{me}}{\partial \bar{\mathbf{p}}_{me}} \frac{\partial \bar{\mathbf{p}}_{me}}{\partial \boldsymbol{\phi}} \frac{D\boldsymbol{\phi}}{D\alpha_i} + \frac{\partial \mathbf{R}_{me}}{\partial \alpha_i} = \frac{\hat{\delta}\mathbf{R}_{me}}{\hat{\delta}\alpha_i}\bigg|_{\frac{D\boldsymbol{\phi}}{D\alpha_i}=\boldsymbol{\Delta\phi}, \frac{D\mathbf{X}_m}{D\alpha_i}=\mathbf{V}_m,} \tag{195}$$

$${}^{I}\hat{\mathbf{R}}_{Me} = \frac{\partial \mathbf{R}_{Me}}{\partial \alpha_i} + \frac{\partial \mathbf{R}_{Me}}{\partial \mathbf{S}_{Me}} \frac{D\mathbf{S}_{Me}}{D\alpha_i} = \frac{\hat{\delta}\mathbf{R}_{Me}}{\hat{\delta}\alpha_i}\bigg|_{\frac{D\mathbf{S}_{Me}}{D\alpha_i}=\boldsymbol{\Delta}\mathbf{S}_{Me}, \frac{D\mathbf{X}_M}{D\alpha_i}=\mathbf{V}_M.} \tag{196}$$

$\boldsymbol{\Delta\phi}$ is an implicit part, that is calculated at the macro level and sent down to the micro level. $\mathbf{V}_M$ is macro level velocity field and $\mathbf{V}_m$ is micro level velocity field. $\boldsymbol{\Delta}\mathbf{S}_{Me}$ is

implicit part evaluated at the micro level and returned to the macro level.

The evaluation of the term $\frac{D\overset{\circ}{\mathbf{p}}_m}{D\alpha_i}$ in (193) requires special treatment. Since it requires differentiation of the complete micro problem, it in fact represents an additional sensitivity analysis at the micro level. This sensitivity analysis is again path-independent or path-dependent.

a) Path-independent micro level sensitivity problem for $\frac{D\overset{\circ}{\mathbf{p}}_m}{D\alpha_i}$

Micro level residual equation for path-independent sensitivity leads to

$$\mathbf{R}_m(\mathbf{p}_m(\boldsymbol{\alpha}), \boldsymbol{\alpha}) = \mathbf{0} \quad m = 1, ..., n \ . \tag{197}$$

Differentiation with respect to optimization sensitivity parameter α_i leads to

$$\frac{\partial \mathbf{R}_m}{\partial \mathbf{p}_m} \frac{D\overset{\circ}{\mathbf{p}}_m}{D\alpha_i} + \frac{\partial \mathbf{R}_m}{\partial \alpha_i} = \mathbf{0} \quad m = 1, ..., n \tag{198}$$

and a system of linear equations is obtained

$$\mathbf{K}_m \frac{D\overset{\circ}{\mathbf{p}}_m}{D\alpha_i} = -{}^{I}\overset{\circ}{\mathbf{R}}_m \tag{199}$$

$${}^{I}\overset{\circ}{\mathbf{R}}_m = \frac{\partial \mathbf{R}_m}{\partial \alpha_i} \ . \tag{200}$$

b) Path-dependent micro level sensitivity problem for $\frac{D\overset{\circ}{\mathbf{p}}_m}{D\alpha_i}$

Since all the path-dependency comes from state variables at Gauss point of micro problem no path-dependent variables appear explicitly in optimization sensitivity. Path-dependency comes implicitly through a sensitivity of micro displacements $\mathbf{p}_m$. Sensitivity problem for $\frac{D\overset{\circ}{\mathbf{p}}_m}{D\alpha_i}$ for path-dependent problems is

$$\mathbf{R}_m(\mathbf{p}_m(\boldsymbol{\alpha}), \mathbf{h}_g(\boldsymbol{\alpha}), \boldsymbol{\alpha}) = \mathbf{0} \tag{201}$$

$$\mathbf{Q}_g(\mathbf{p}_m(\boldsymbol{\alpha}), \mathbf{h}_g(\boldsymbol{\alpha}), \boldsymbol{\alpha}) = \mathbf{0} \ : g = 1, 2, ...n_g \ . \tag{202}$$

$\mathbf{h}_g$ is vector of GAUSS point coupled unknowns. The direct differentiation of (201) with respect to i-th optimization parameter α_i yields

$$\frac{\partial \mathbf{R}_m}{\partial \mathbf{p}_m}\frac{D\overset{\circ}{\mathbf{p}}_m}{D\alpha_i} + \sum_{g=1}^{n_g}\frac{\partial \mathbf{R}_g}{\partial \mathbf{h}_g}\frac{D\overset{\circ}{\mathbf{h}}_g}{D\alpha_i} + \frac{\partial \mathbf{R}_m}{\partial \alpha_i} = \mathbf{0}\,. \tag{203}$$

$\frac{D\overset{\circ}{\mathbf{p}}_m}{D\alpha_i}$ denotes the unknown sensitivity of independent solution vector and $\frac{D\overset{\circ}{\mathbf{h}}_g}{D\alpha_i}$ the unknown sensitivity of n_g dependent solution vectors. The sensitivity of the dependent solution vector $\frac{D\overset{\circ}{\mathbf{h}}_g}{D\alpha_i}$ is needed in order to solve equation (203) for the unknown $\frac{D\overset{\circ}{\mathbf{p}}_m}{D\alpha_i}$. Direct differentiation of the g-th dependent residual (202) with respect to i-th design parameter α_i yields

$$\frac{\partial \mathbf{Q}_g}{\partial \mathbf{p}_m}\frac{D\overset{\circ}{\mathbf{p}}_m}{D\alpha_i} + \frac{\partial \mathbf{Q}_g}{\partial \mathbf{h}_g}\frac{D\overset{\circ}{\mathbf{h}}_g}{D\alpha_i} + \frac{\partial \mathbf{Q}_g}{\partial \alpha_i} = \mathbf{0}. \tag{204}$$

The term $\frac{\partial \mathbf{Q}_g}{\partial \mathbf{h}_g}$ in (204) is exactly dependent tangent operator $\mathbf{K}_Q$. A linear system of equations

$$\mathbf{K}_m\frac{D\overset{\circ}{\mathbf{p}}_m}{D\alpha_i} = -{}^I\overset{\circ}{\mathbf{R}}_m \tag{205}$$

is solved for the unknown sensitivity vector $\frac{D\overset{\circ}{\mathbf{p}}_m}{D\alpha_i}$, where $\mathbf{K}_m$ is the independent micro tangent operator of the primal problem. The element contribution ${}^I\overset{\circ}{\mathbf{R}}_m$ to independent sensitivity pseudo-load vector is defined by

$${}^I\overset{\circ}{\mathbf{R}}_m = \frac{\partial \mathbf{R}_m}{\partial \alpha_i} - \frac{\partial \mathbf{R}_m}{\partial \mathbf{h}_g}{\mathbf{K}_Q}^{-1}\frac{\partial \mathbf{Q}_g}{\partial \alpha_i}\,. \tag{206}$$

With the known sensitivity of independent solution vector $\frac{D\overset{\circ}{\mathbf{p}}_m}{D\alpha_i}$, the system of linear equations

$$\mathbf{K}_Q\frac{D\overset{\circ}{\mathbf{h}}_g}{D\alpha_i} = -{}^I\overset{\circ}{\mathbf{Q}}_g \;: g = 1, \ldots, n_g \tag{207}$$

can be formed for the unknown sensitivity of dependent solution vectors $\frac{D\overset{\circ}{\mathbf{h}}_g}{D\alpha_i}$. The term ${}^I\overset{\circ}{\mathbf{Q}}_g$ is dependent sensitivity pseudo-load vector and is defined by

$${}^I\overset{\circ}{\mathbf{Q}}_g = \frac{\partial \mathbf{Q}_g}{\partial \mathbf{p}_m}\frac{D\overset{\circ}{\mathbf{p}}_m}{D\alpha_i} + \frac{\partial \mathbf{Q}_g}{\partial \alpha_i} \;: g = 1, \ldots, n_g. \tag{208}$$

In Fig. 31 procedure of multi-scale optimization sensitivity is schematically presented. Procedure starts with macro primal analysis, for which micro primal analysis and sensitivity analysis with respect to $\boldsymbol{\phi}$ is needed for each NR iteration. Sensitivities and pseudo-load vector calculated in this stage are not the total ones, for this reason they

are marked with additional circle. Micro sensitivity for $\boldsymbol{\alpha}$ is solved and used in independent optimization sensitivity, where $\frac{D\mathbf{p}_M}{D\alpha_i}$ is calculated, which represents a change of $\mathbf{p}_M$ because of total change of α_i. With this information the dependent optimization sensitivity is calculated. In response functional arbitrary quantity can be calculated with its derivatives, taking into account results of optimization sensitivity.

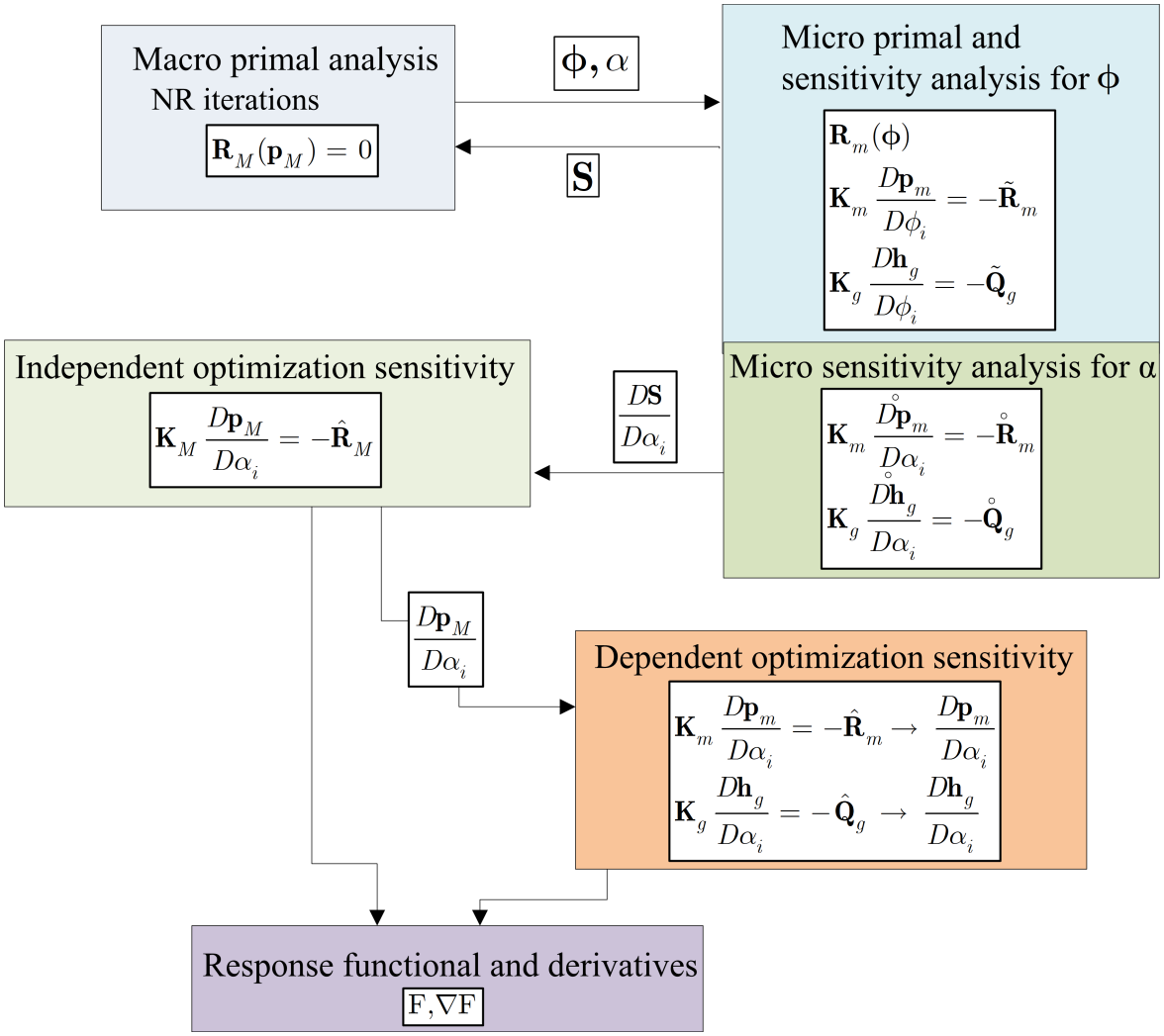


Figure 31: Diagram of multi-scale optimization sensitivity analysis
Slika 31: Diagram večnivojske optimizacijske občutljivostne analize

6.3.1 Optimization sensitivity parameters and velocity fields

Macro sensitivity parameters $\boldsymbol{\alpha}$ can be one of general sensitivity parameters, parameters related to the input data of the FE analysis at the micro level, such as geometrical characteristics and material constants, $\boldsymbol{\alpha} = \{r, E, \nu \ldots\}$. Geometrical characteristic is

denoted with r, representing an arbitrary parameter that influences inner structure at the micro level, e.g. radius of the void. It affects geometry of mesh at the micro level resulting in a need for shape sensitivity. The components of the first order shape velocity field ${}^{I}\mathbf{V}$ are derivatives of the micro mesh nodal coordinates. Micro mesh coordinates implicitly depend on r, $X_m = X_m(r)$. They need to be expressed in parameterized way, with respect to parameter r, to be able to calculate analytical velocity field

$$ {}^{I}\mathbf{V} = \frac{\partial X_m(r)}{\partial r}. \tag{209} $$

In case where coordinates are only linearly dependent on r, it is enough to prescribe only first order shape velocity field ${}^{I}\mathbf{V}$, otherwise also ${}^{II}\mathbf{V}$ is needed. Sensitivity analysis for material constants E and ν is classified as parameter sensitivity and velocity field is equal to 1, ${}^{I}\mathbf{V} = 1$.

6.3.2 Optimization with respect to plastic work

For optimization of plastic work in frame of FE^2 the evolution of plastic work is defined by $\dot{W}_p = \boldsymbol{\sigma}_{Mises} \cdot \dot{\boldsymbol{\gamma}}_g$, where $\boldsymbol{\gamma}_g$ is accumulated plastic deformation, $\boldsymbol{\sigma}_{Mises}$ is Mises stress and $\boldsymbol{\sigma}^{'}$ is deviatoric stress defined in (70). The discretized evolution equation then leads to additional algebraic equations that have to be solved for each Gauss point.

$$ \boldsymbol{\sigma}_{Mises} = (\frac{3}{2}\boldsymbol{\sigma}^{'} \cdot \boldsymbol{\sigma}^{'})^{1/2} \tag{210} $$

$$ \Delta\boldsymbol{\gamma}_g = \boldsymbol{\gamma}_g - \boldsymbol{\gamma}_{gn} \tag{211} $$

$$ \Delta W_p = \boldsymbol{\sigma}_{Mises} \cdot \Delta\boldsymbol{\gamma}_g \tag{212} $$

$$ W_p - W_{pn} - \Delta W_p = 0 \tag{213} $$

Equation (213) is added to $\mathbf{Q}_{mg}$ in (73)

$$ \mathbf{Q}_{mg} = \{Z_{11}, Z_{22}, Z_{33}, Z_{12}, Z_{13}, Z_{23}, \phi, W_p - W_{pn} - \Delta W_p\} = \mathbf{0} \tag{214} $$

and the rest follows as defined. Thus the set of Gauss points equations $\mathbf{Q}_{mg}$ and set of state variables $\mathbf{h}_{mg}$ for plasticity is extended to

$$ \mathbf{h}_{mg} = \{\boldsymbol{\varepsilon}_{p11}, \boldsymbol{\varepsilon}_{p22}, \boldsymbol{\varepsilon}_{p33}, \boldsymbol{\varepsilon}_{p12}, \boldsymbol{\varepsilon}_{p13}, \boldsymbol{\varepsilon}_{p23}, \gamma, W_p\}\,. \tag{215} $$

Since sensitivity of $\mathbf{h}_{mg}$ follows automatically by algorithm, the sensitivity of plastic work does not require additional equations only integration over the domain.

$$W_{pm} = \frac{1}{V_0} \int_{\Omega_m} W_p \, dV \tag{216}$$

$$W_{pM} = \int_{\Omega_M} W_{pm} \, dV \tag{217}$$

W_{pm} in (216) is micro level quantity calculated at RVE an is for that reason averaged over the volume. *AceGen* input segment of finite element for response functional task subroutine of FE^2 macro and micro element is presented in Appendix A.6.

6.4 Numerical examples

In the following examples multi-scale gradient-based optimization algorithm presented in Fig. 29 will be used. Optimization will be carried out with `FindMinimum` function implemented in Mathematica. For each set of optimization parameters $\boldsymbol{\alpha}$, multi-scale analysis is solved. The macro problem is defined, followed by the definition of micro problems. Sensitivities calculated at the micro level are the input of primal and sensitivity analysis of the macro problem, and solutions of the macro problem are used in the evaluation of objective function and its gradient.

6.4.1 2D functionally graded material optimization for minimum weight

Multi-scale optimization of simply supported beam with functionally graded porosity function in Fig. 32 was investigated. Function of porosity $\rho(X, \boldsymbol{\alpha})$ was defined at the macro level as a function of nodal coordinates X, and 15 optimization parameters α for value of porosity across the domain ρ_{ij}. Uniform distributed load $q = 130$ in the vertical y-direction was imposed on the cantilever with dimensions 10×2 and macro mesh density 40×8, 320 macro elements. Micro level was meshed with 384 elements. At both levels Q1, four nodded, quadrilateral elements were used. Elastic material model with material properties $E = 21000$, and $\nu = 0.3$, and $\bar{v}_A = -0.5$ was used. Optimization problem was defined as find the value of optimization parameters α_i for which beam would have the minimal volume, at given distributed load q and will have prescribed displacement $v_A = \bar{v}_A$ at the point A, the middle of the beam. Optimization parameters are sensitivity

parameters at the macro level $\boldsymbol{\alpha}$.

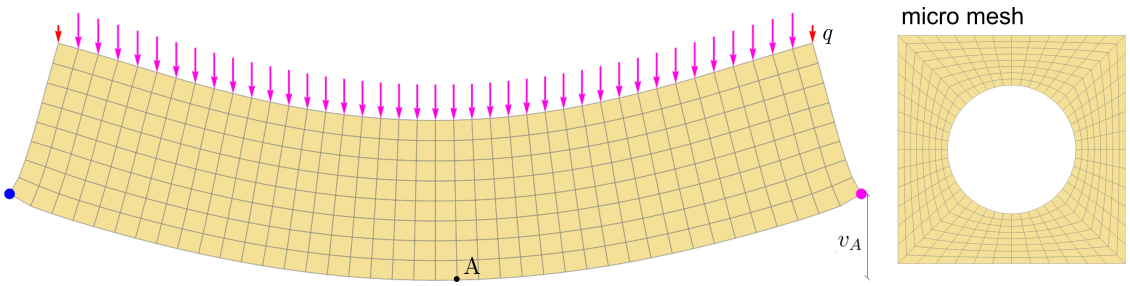


Figure 32: Simply supported beam, macro mesh and micro mesh
Slika 32: Prostoležeči nosilec, makro mreža in mikro mreža

Porosity ρ is approximated using cartesian product of B-spline shape functions in x- and y-direction in Fig. 33. Set of sensitivity parameters $\boldsymbol{\alpha}$ represents value of porosity in points across the domain ρ_{ij}.

$$\begin{aligned} \rho &= \sum_{i=1}^{5}\sum_{j=1}^{3}\mathbf{B}_i(x)\mathbf{B}_j(y)\rho_{ij} \\ \boldsymbol{\alpha} &= \bigcup_{i,j}\rho_{ij} \end{aligned} \tag{218}$$

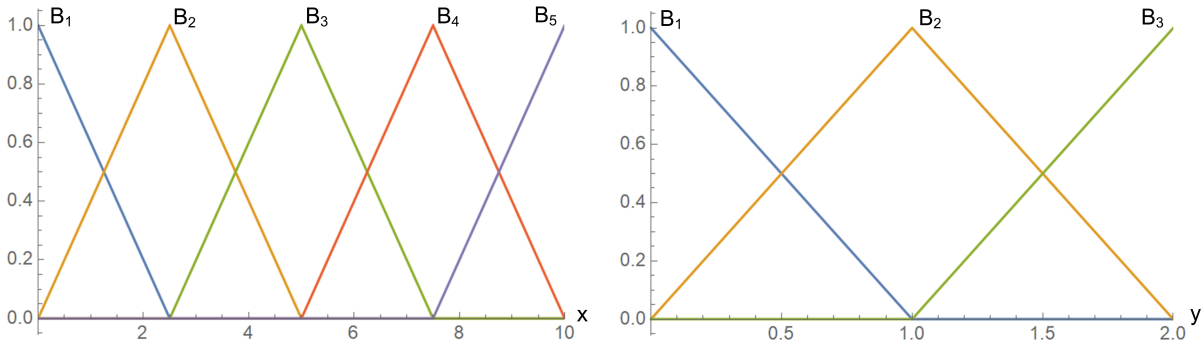


Figure 33: B-splines for shape functions interpolation of porosity
Slika 33: B-zlepki za oblikovne funkcije interpolacije poroznosti

Quasi-Newton method was used and the objective function F was defined with:

$$\begin{aligned}
&\mathrm{F} = w_1\mathrm{F}_1 + w_2\mathrm{F}_2 + w_3\mathrm{F}_3 + w_4\mathrm{F}_4\\
&\mathrm{F}_1 = V^2\\
&\mathrm{F}_2 = (v_A - \bar{v}_A)^2\\
&\mathrm{F}_3 = \sum_i b_{min\,i}(\alpha_i, \alpha_{min\,i})\\
&\mathrm{F}_4 = \sum_i b_{max\,i}(\alpha_i, \alpha_{max\,i})\\
&\nabla\mathrm{F} = w_1\nabla\mathrm{F}_1 + w_2\nabla\mathrm{F}_2 + w_3\nabla\mathrm{F}_3 + w_4\nabla\mathrm{F}_4\\
&\nabla\mathrm{F}_j = \{\frac{\partial \mathrm{F}_j}{\partial \alpha_i}\}; j = \{1,2,3,4\}; i = \{1,2,...,15\}
\end{aligned} \tag{219}$$

where F_1 is a volume part, F_2 displacement part and F_3 and F_4 are inequality constraints with w_j as weights. Volume is defined by

$$V = \int_0^H \int_0^L (1 - \rho(\boldsymbol{\alpha}))\, dx\, dy\,. \tag{220}$$

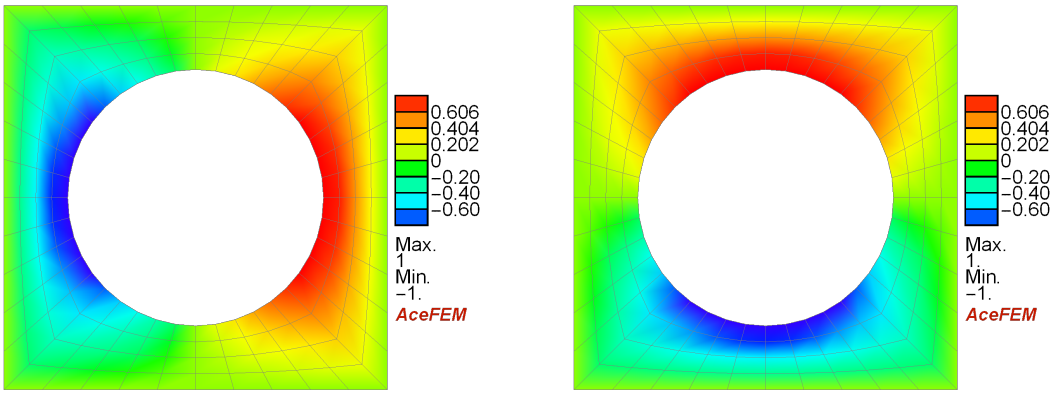
The barrier function is defined by

$$\begin{aligned}
&f_b(\nu, b, \mu, \varepsilon) = \begin{cases} -\mu \ln(\nu - b), \nu - b > \varepsilon; \\ \mu\, \frac{b^2 - 2\,b\,\nu + \nu^2 + 4\,b\,\varepsilon - 4\,\nu\,\varepsilon + 3\,\varepsilon^2 - 3\,\varepsilon^2 \ln(\varepsilon)}{2\,\varepsilon^2}, \nu - b \leqslant \varepsilon \end{cases}\\
&b_{min\,i}(\alpha_i, \alpha_{min\,i}) = f_b(\alpha_i, \alpha_{min\,i}, \mu, \varepsilon)\\
&b_{max\,i}(\alpha_i, \alpha_{max\,i}) = f_b(\alpha_{max\,i}, \alpha_i, \mu, \varepsilon);
\end{aligned} \tag{221}$$

function f_b increases steeply when optimization parameter α_i is out of permitted interval. Following values of parameters were used $\alpha_{max\,i} = 0.6$, $\alpha_{min\,i} = 0.01$, $\mu = 0.001$ and $\varepsilon = 0.00001$.

To calculate the gradient of objective function sensitivity analysis at the macro level is used. Because porosity does not affect the macro level directly but through perforations presented at the micro level, shape sensitivity analysis is needed at the micro level. For each macro element in the case of MIEL and RVE in case of FE^2 corresponding value of porosity was calculated from porosity function with respect to X. At the micro level, this parameter dictates the radius of the void r. It influences the nodal coordinates of micro mesh $X_m(r)$ and is thus micro level shape sensitivity parameter. As an input for shape sensitivity analysis, shape velocity field needs to be prescribed. It is equal to

the derivative of nodal coordinates with respect to $r = r(\rho(x, y, \boldsymbol{\alpha}))$ and is graphically presented in Fig. 34 for one micro problem. For a successful process of optimization via sensitivity analysis, the accuracy of the velocity field is important, for this reason, finite element mesh needs to be described with a parameterized function enabling calculation of analytic derivative.



(a) $\frac{Dx}{Dr}$ (b) $\frac{Dy}{Dr}$

Figure 34: Shape velocity field at the micro level
Slika 34: Oblikovno hitrostno polje na mikro nivoju

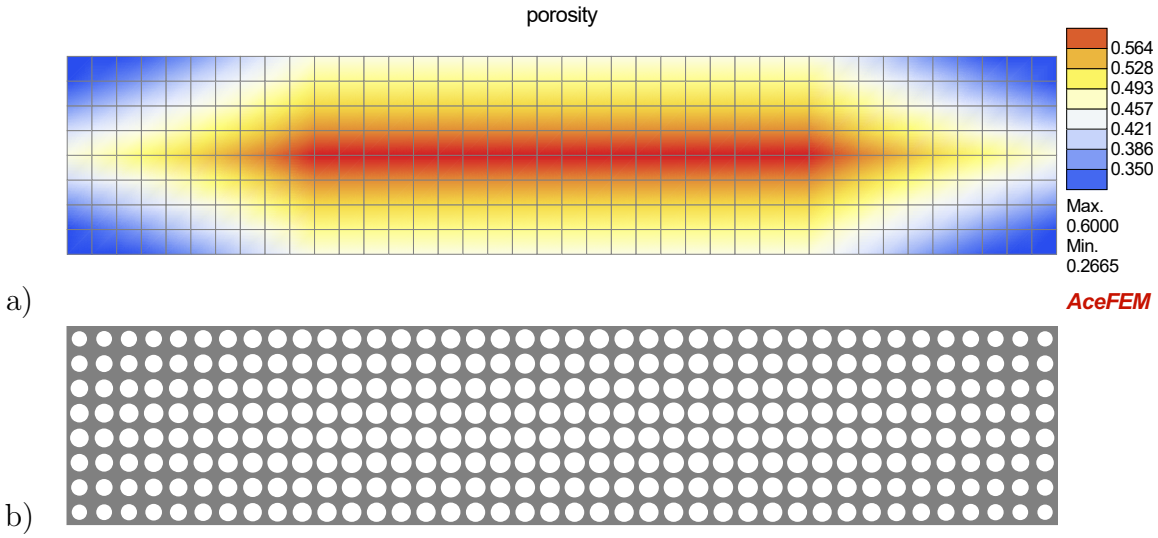


Figure 35: MIEL results: a) optimal porosity distribution b) pores represented with 40×8 raster
Slika 35: MIEL rezultati: a) optimalna razporeditev poroznosti b) predstavitev odprtin z 40×8 rastrom

Optimization started from uniform porosity 0.1. In Table 11 computational times needed for optimization and value of volume are compared for FE^2 and MIEL. Computation for MIEL required more time, but resulted in smaller volume than FE^2. For optimized porosity distribution, volume for MIEL was 51% and for FE^2 58% of volume without porosity. Final result for optimal porosity is shown for MIEL in Fig. 35 and for FE^2 in Fig. 36.

Table 11: Comparison of time needed for optimization and resulting volume
Preglednica 11: Primerjava časa potrebnega za optimizacijo in rezultirajoč volumen

	time	volume	volume/volume$_0$
MIEL	14 175 s	10.23	51 %
FE^2	10 960 s	11.50	58 %

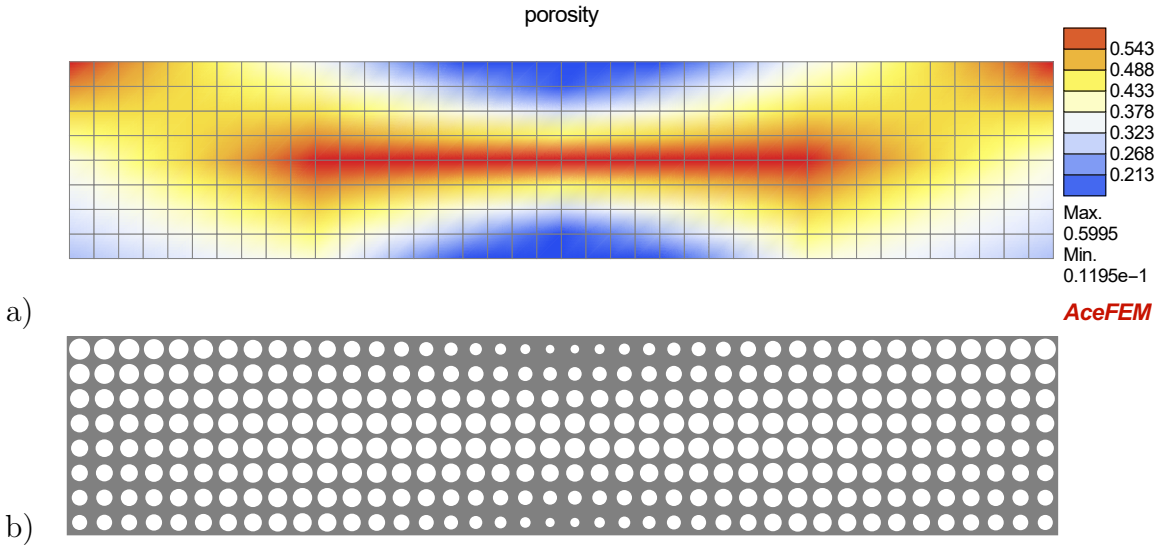


Figure 36: FE^2 results: a) optimal porosity distribution b) pores represented with 40×8 raster
Slika 36: FE^2 rezultati: a) optimalna razporeditev poroznosti b) predstavitev odprtin z 40×8 rastrom

6.4.2 3D optimization

For 3D example, console with dimension $L \times B \times H(x)$ with $L = 6$, $B = 1$ and changing height was investigated. It was loaded on the top surface with $p_z = -4$ and linear elastic material model with $\nu = 0.3$ and $E = 21000$ was used. Optimization was performed in two stages using the Interior point method. First, the geometry of console was optimized to get maximal stiffness for prescribed volume and in the second stage optimal porosity was searched for.

a) First stage

For minimal displacement in point A (w_A), optimal values of height parameters were searched for. This optimization was done at a single-scale with mesh division $16 \times 96 \times 32$. 3D, eight nodded, hexahedral elements were used. The objective function F, equality constraint for the value of volume and inequality constraints for minimal/maximal value for optimization parameters were defined:

$$\begin{aligned} &\mathrm{F} = w_A^2 \\ &(V - V_{target}) = 0; V_{target} = L \cdot B \cdot 1 \\ &\alpha_i > \alpha_{min\,i} \\ &\alpha_i < \alpha_{max\,i} \end{aligned} \tag{222}$$

Sensitivity parameters for console height were $\{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5\}$, with initial values for parameters equal to $\alpha_i = 1$. For maximal and minimal value of parameters $\alpha_{max\,i} = 4$ and $\alpha_{min\,i} = 0.1$ were used. Velocity fields were prescribed and are derivatives of nodal coordinates. Initial geometry is shown in Fig. 37.

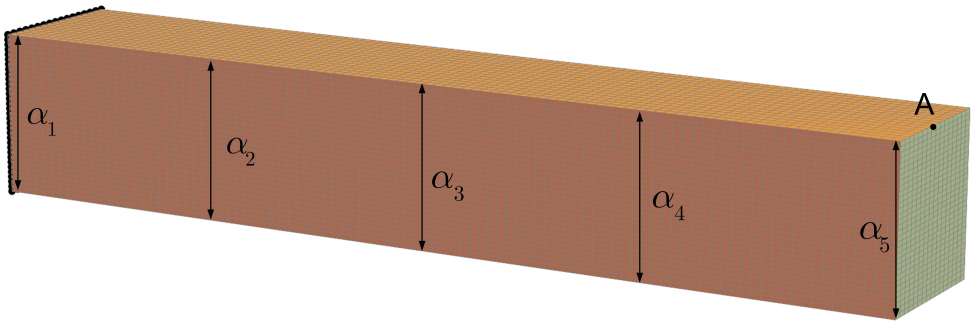


Figure 37: Geometry for initial values
Slika 37: Geometrija za začetne dimenzije

Optimized geometry is shown in Fig. 38, with values for $\boldsymbol{\alpha} = \{1.72, 1.38, 1.08, 0.59, 0.18\}$. Fig. 39 represents value of objective function F for each analysis of optimization procedure. The Interior point method requires, for the evaluation of approximate tangent matrix, separate evaluation of gradient for each optimization parameter in each step of optimization. It can be observed, that optimal solution was found in 6 optimization steps (in total 36 primal and sensitivity analyses).

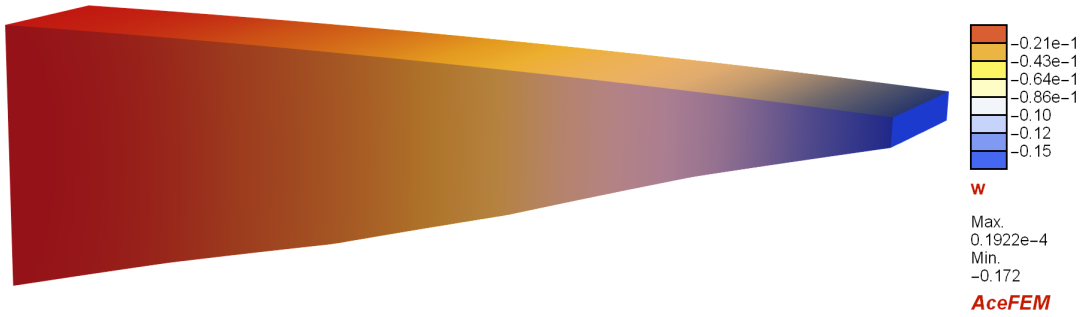


Figure 38: Optimized macro level geometry
Slika 38: Optimizirana makro geometrija

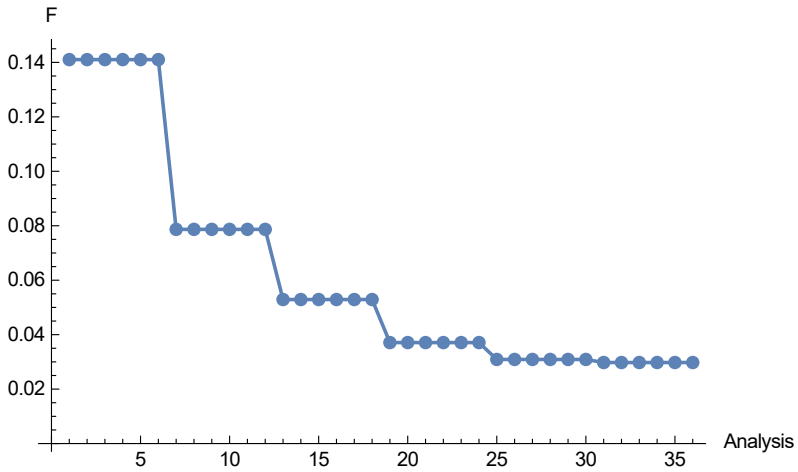


Figure 39: Value of objective function F for primal and sensitivity analysis of single-scale case
Slika 39: Vrednost namenske funkcije F za primarno in občutljivostno analizo enonivojskega primera

b) Second stage

In the second stage our goal was to determine optimal porosity for prescribed displacement $\bar{w}_A = -0.25$ and fixed volume. Multi-scale optimization algorithm for FE^2 was employed, with previously optimized geometry from Fig. 38. For multi-scale optimization, macro geometry is shown in Fig. 40. At the macro level $12 \times 2 \times 4$ division resulted in 96 macro elements and 768 RVEs (Fig. 45). This mesh is much coarser than the one used for single-scale optimization because multi-scale optimization procedure for 3D is very costly.

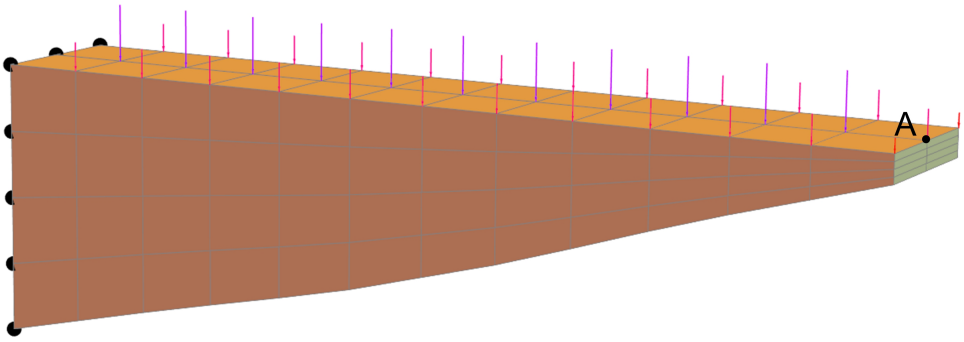


Figure 40: Macro geometry
Slika 40: Makro geometrija

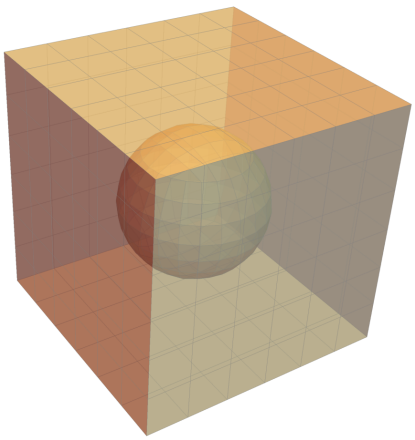


Figure 41: RVE geometry
Slika 41: RVE geometrija

Porosity ρ is approximated using product of B-spline shape functions in x-direction, Fig. 42. Set of sensitivity parameters $\boldsymbol{\alpha}$ represents value of porosity in x-direction ρ_i.

$$\begin{aligned} \rho &= \sum_{i=1}^{5} \mathbf{B}_i(x)\rho_i \\ \boldsymbol{\alpha} &= \bigcup_i \rho_i \end{aligned} \tag{223}$$

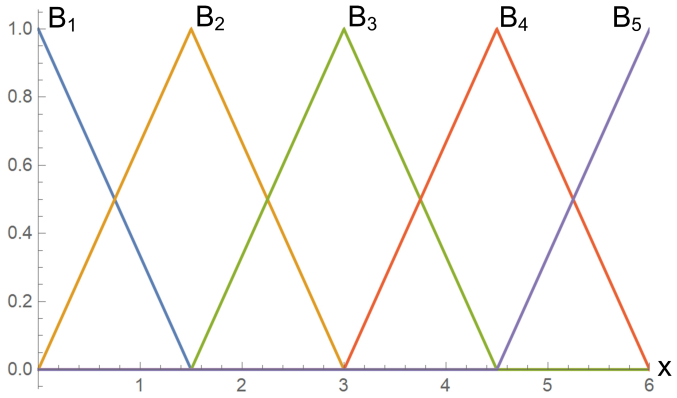


Figure 42: B-splines for shape functions interpolation of porosity
Slika 42: B-zlepki za oblikovne funkcije interpolacije poroznosti

At the micro level spherical opening was introduced inside each RVE, with radius $r = r(\rho(x, \boldsymbol{\alpha}))$. RVE geometry is shown in Fig. 41 and was mashed with 648 finite elements. Mesh coordinates were expressed as a function of its radius. Their derivatives were calculated and set as values of the velocity field. Again Interior point method was used with objective function F, and inequality constraints as follows:

$$\begin{aligned} \mathrm{F} &= w_1 \left(w_A - \bar{w}_A\right)^2 + w_2 \, V \\ \alpha_i &> \alpha_{min\,i} \\ \alpha_i &< \alpha_{max\,i} \end{aligned} \tag{224}$$

First part of objective function represents displacement part and the second represents volume. For weights following values were taken $w_1 = 100$ and $w_2 = 1$. Volume is defined by

$$V = \int_0^L (1 - \rho(\boldsymbol{\alpha}))\, h(x)\, B\, dx\,. \tag{225}$$

Optimal solution was found in 7 optimization steps (42 primal and sensitivity analyses) which are evident in Fig. 43. Optimal porosity values were $\boldsymbol{\alpha} = \{0.34, 0.36, 0.36, 0.36, 0.38\}$ and porosity distribution is shown in Fig. 44. Problem was solved on 14 parallel kernels in 43 818 s. In Fig. 45 characteristics of multi-scale problem are shown. Problem had 96

FE^2 macro elements and 768 micro problems (RVEs). Total number of micro elements was 601 344, with 761 856 micro nodes. In total 2 267 136 micro equations were solved. Individual micro mesh was generated in 17 minutes and 43 seconds and solved in 2 minutes. Thus for elastic problems, most of total time was used for micro mesh generation time and administrative time and not for actual solution of the problem, as it is the case for problems with plasticity.

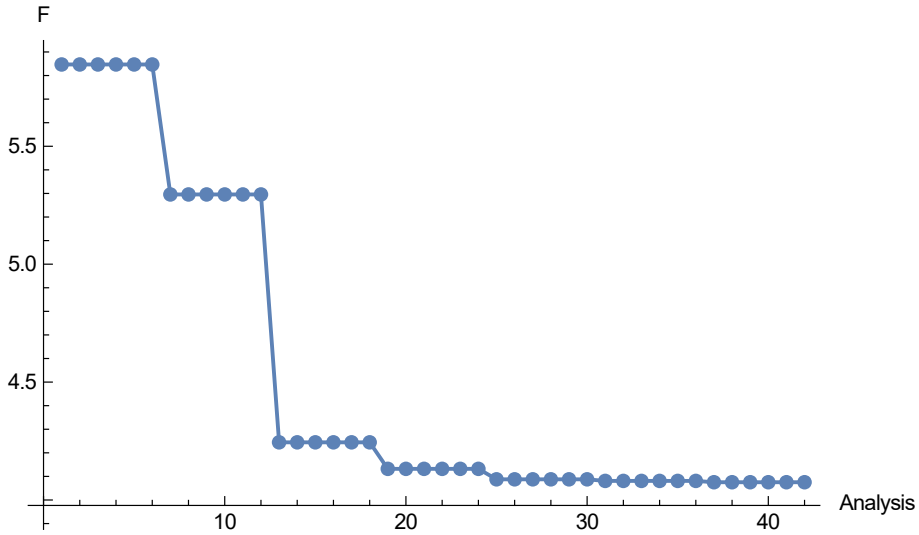


Figure 43: Value of objective function F for primal and sensitivity analysis of multi-scale case
Slika 43: Vrednost namenske funkcije F za primarno in občutljivostno analizo večnivojskega primera

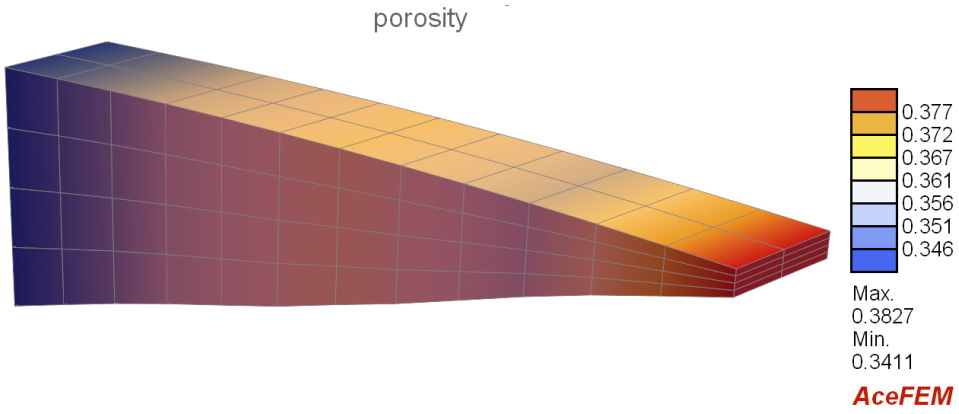


Figure 44: Optimal porosity distribution
Slika 44: Optimalna razporeditev poroznosti

Number of macro elements	96
Number of FE^2 macro elements	96
No. MIEL macro elements	0
Number of single-scale elements	0
Number of kernels	14
Total number of micro problems	768
Total number of micro elements	601 344
Total number of micro nodes	761 856
Total number of micro equations	2 267 136
Real micro mesh generation time	0:17:43.9878

Figure 45: Characteristics of multi-scale problem
Slika 45: Karakteristike večnivojskega problema

6.4.3 Optimization of metamaterial for maximal energy dissipation

The concept of protecting buildings in case of an earthquake include different ways of building design. For steel structures capacity design is established, where a structure is designed to possess adequate ductility so that it can dissipate energy by yielding and survive the shock. In this design approach, the structures are designed in such a way that plastic hinges can form only in predetermined positions. The concept of this method is to avoid the brittle mode of failure. In the design process, it is decided which objects within a structural system will be permitted to yield (ductile components) and which objects will remain elastic (brittle components). For example, we have strong-column weak-beam design, where hinges should form in the beam and not in columns. From this concept comes an idea to make some sort of damping element, made out of metamaterial. It would be able to dissipate as much energy as possible, with the use of plastic deformation. Its application would be in some sort of construction part that would be able to bear extensive damage in case of an earthquake and take over part of loading and protect main load barring parts. Dissipated energy is equal to accumulated plastic energy W_p.

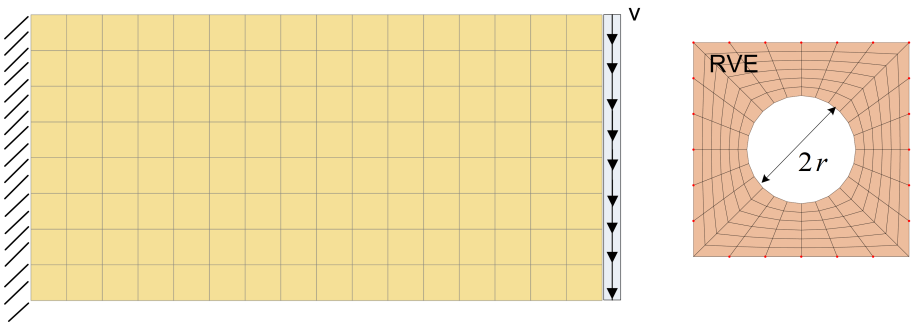


Figure 46: Macro and micro geometry
Slika 46: Makro in mikro geometrija

Proposed metamaterial would use a variation of porosity across the domain to achieve optimal plastic energy dissipation together with chosen use of material. For macro scale geometry, simple square with dimensions $L = 8$, $H = 4$, and mesh division 16×8 is chosen. It is clamped at both sides and at the end vertical displacement v is prescribed. RVE was discretized with 144 elements. Micro and RVE geometry are presented in Fig. 46. A small strain elasto-plastic material model with material properties $E = 21000$, $\nu = 0.3$, $\sigma_{y0} = 24$, $K_h = 21$, $R_\infty = 12$ and $\delta = 30$ is used at the micro level. For discretization at micro and macro level four nodded, quadrilateral elements are used. In this particular case we took plane stress continuum model, as it could reflect situation for 3D printing that is done in layers perpendicular to xy-plane.

The objective function F is defined as:

$$\begin{aligned}
&\mathrm{F} = w_1\mathrm{F}_1 + w_2\mathrm{F}_2 + w_3\mathrm{F}_3 + w_4\mathrm{F}_4 \\
&\mathrm{F}_1 = W_p \\
&\mathrm{F}_2 = (V - V_{target})^2 \, , V_{target} = 0.6 \cdot V_0 \, , V_0 = L \cdot H \\
&\mathrm{F}_3 = \sum_i \mathbf{b}_{min\,i}(\alpha_i, \alpha_{min\,i}) \\
&\mathrm{F}_4 = \sum_i \mathbf{b}_{max\,i}(\alpha_i, \alpha_{max\,i}) \\
&\nabla\mathrm{F} = w_1\nabla\mathrm{F}_1 + w_2\nabla\mathrm{F}_2 + w_3\nabla\mathrm{F}_3 + w_4\nabla\mathrm{F}_4 \\
&w_1 = -20, w_2 = 10, w_3 = w_4 = 1 \\
&\nabla\mathrm{F}_j = \{\frac{\partial \mathrm{F}_j}{\partial \alpha_i}\}; j = \{1,2,3,4\}; i = \{1,2,...,48\}
\end{aligned} \tag{226}$$

where F_1 is a plastic energy part, F_2 volume part and F_3 and F_4 include inequality constraints. w_j are weights. Target value of volume is 60% of volume without pores. The barrier function for inequality constraints is defined by (221) and parameters $\alpha_{max\,i} = 0.6$, $\alpha_{min\,i} = 0.01$, $\mu = 0.001$ and $\varepsilon = 0.00001$ are used. Volume is defined by (220).

Sensitivity parameters at the macro level are values of porosity across the domain in raster 8×6, $\{\alpha_1, \alpha_1, ..., \alpha_{48}\}$. Porosity ρ is approximated using cartesian product of B-spline shape functions in x- and y-direction in Fig. 47, using (218). Radius of pores at the micro level is function of sensitivity parameters $\boldsymbol{\alpha}$. They are added to EBC sensitivity parameters, components of the small strain tensor (see e.g. Section 4.2.1).

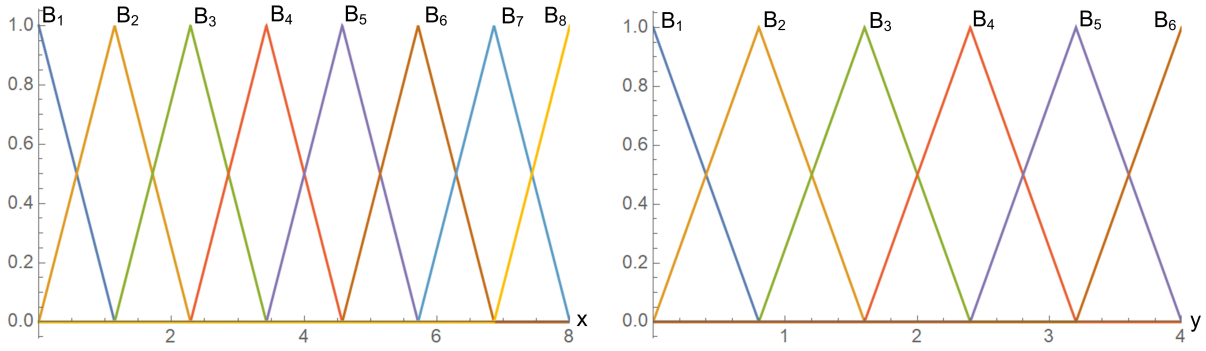


Figure 47: B-splines for shape functions interpolation of porosity
Slika 47: B-zlepki za oblikovne funkcije interpolacije poroznosti

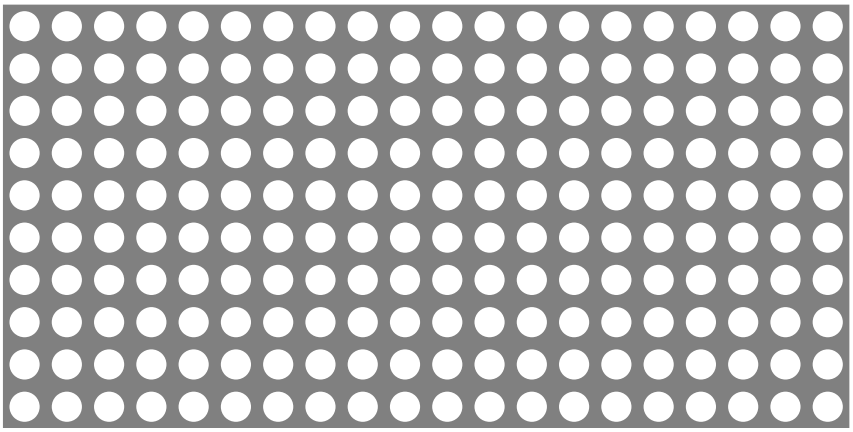


Figure 48: Pores representation with raster 20 × 10 for constant porosity 0.4
Slika 48: Predstavitev odprtin z 20 × 10 rastrom za konstantno poroznost 0.4

Two different cases were investigated. First had monotonic loading and second cyclic loading presented in Fig. 49. Initial value for optimization was constant porosity 0.4 across all the domain shown in Fig. 48. Fig. 50 displays change of accumulated plastic energy W_p for optimization iterations. We can observe that optimization algorithm finds optimal solution already before ten iterations are made. Optimal distribution of porosity, which was translated into the actual pores with raster 20 × 10, can be seen in Fig. 51 for monotonic loading and in Fig. 52 for cyclic loading.

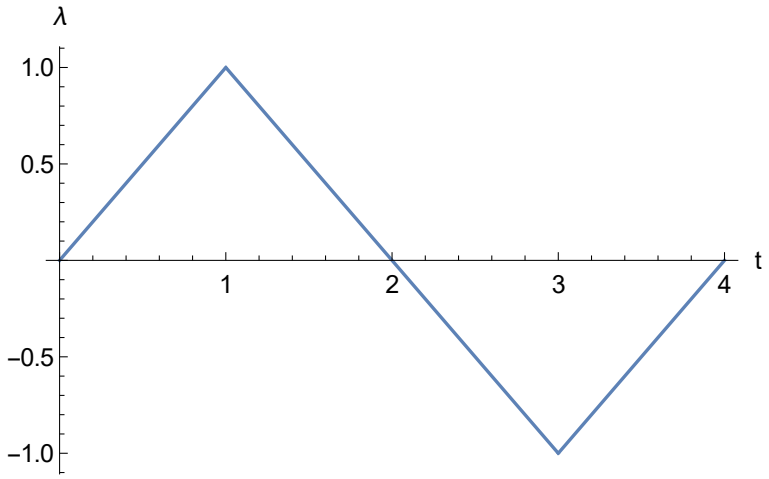


Figure 49: Cyclic loading
Slika 49: Ciklično obremenjevanje

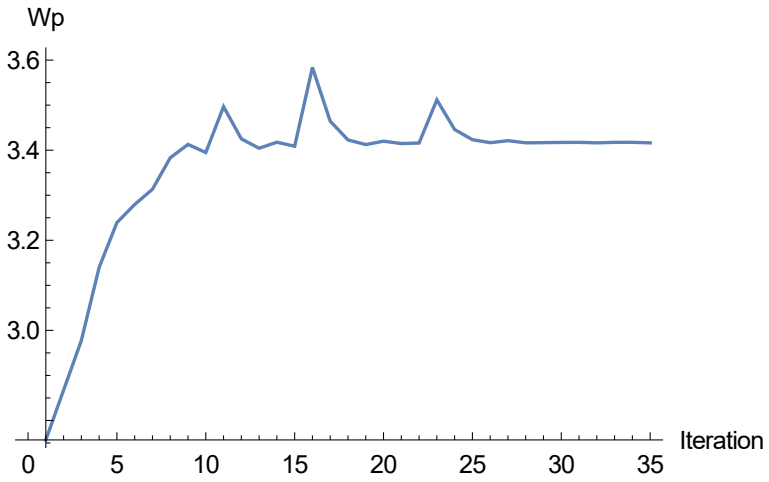


Figure 50: Change of W_p with iterations for monotonic loading (t $\in [0, 1]$)
Slika 50: Sprememba W_p z iteracijami pri monotoni obtežbi (t $\in [0, 1]$)

Table 12: Comparison of W_p for different porosity and loading type
Preglednica 12: Primerjava W_p za različno poroznost in vrsto obtežbe

	W_p (constant porosity 0.4)	W_p (optimal porosity)
monotonic loading	2.76	3.41
cyclic loading	9.73	16.68

Comparison of accumulated plastic energy for initial and optimal porosity for monotonic and cyclic loading can be seen in Tab. 12. For monotonic loading and constant porosity 0.4 accumulated plastic energy was 2.76 and for optimized porosity 3.14. For initial value of constant porosity 0.4 and cyclic loading accumulated plastic energy was 9.73 and for optimized porosity it was 16.68. Optimized porosity enables 1.71 times greater energy dissipation for the same quantity of material used for cyclic loading, and 1.23 times for monotonic loading.

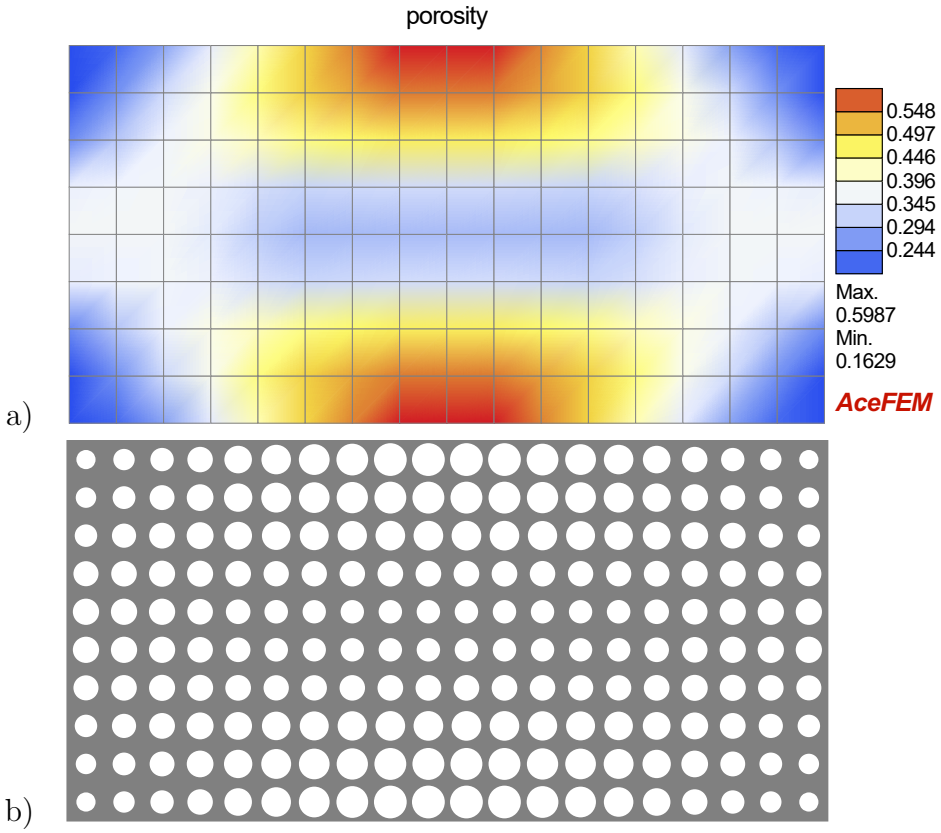


Figure 51: Results of W_p optimization for monotonic loading: a) optimal porosity distribution b) pores represented with 20×10 raster
Slika 51: Rezultati W_p optimizacije pri monotoni obtežbi: a) optimalna razporeditev poroznosti b) predstavitev odprtin z 20×10 rastrom

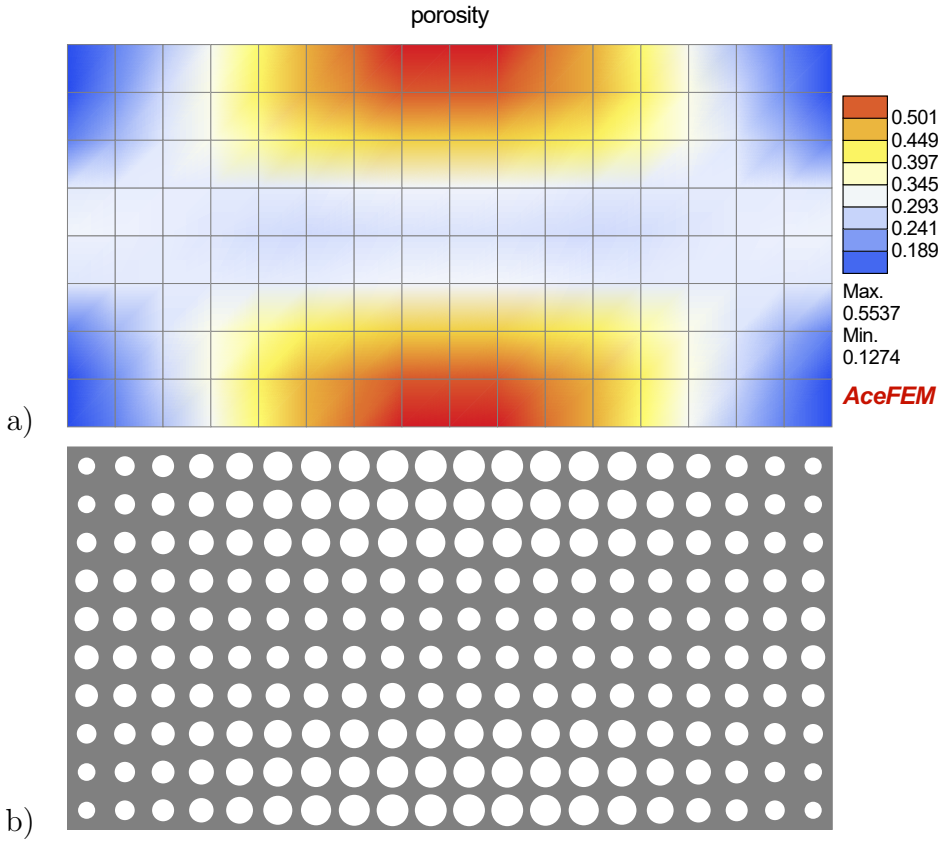


Figure 52: Results of W_p optimization for cyclic loading: a) optimal porosity distribution b) pores represented with 20×10 raster

Slika 52: Rezultati W_p optimizacije pri ciklični obtežbi: a) optimalna razporeditev poroznosti b) predstavitev odprtin z 20×10 rastrom

7 CONCLUSIONS

The objective of this thesis was to develop a generalized essential boundary condition sensitivity analysis based implementation of FE^2 and MIEL multi-scale methods. The method was derived and described in detail, as an alternative to more traditional implementations of multi-scale analysis, where the calculation of the SCHUR complement of the microscopic tangent matrix is needed for bridging scales. The thesis shows that the derivation of the algorithmic macroscopic tangent requires the first order essential boundary condition sensitivity analysis for the FE^2 multi-scale method and the second order essential boundary condition sensitivity analysis for the MIEL multi-scale method. Thus, a general automatic differentiation-based formulation (ADB) is introduced for the first and second order essential boundary condition sensitivity analysis that can be applied on an arbitrary coupled, path-dependent micro material model or element formulation.

ADB brings several advantages. The first advantage is the ability to naturally introduce a fully consistently linearized two-level path-following algorithm as a multi-scale modeling solution. Sensitivity analysis allows each macro step to be followed by an arbitrary number of micro steps, while retaining quadratic convergence of the overall solution algorithm. From the examples discussed in this work, it was shown that this cannot be achieved with the standard SCHUR complement based methods. Additionally, the method completely avoids the numerically demanding evaluation of the SCHUR complement of the micro tangent matrix which, especially for the MIEL multi-scale methods, results in large dense matrices.

The advantages of the sensitivity analysis based implementation in comparison with traditional methods were recognized and verified using numerical examples. The convergence of results with respect to the size of the macro load step and the number of micro steps was investigated. With additional micro steps, the accuracy of the global response can be improved without additional costly macro steps. This is especially evident in the case of a strongly nonlinear micro response, which, for some reason, does not reflect the global response but still limits the size of the macro load steps. Similarly, a strongly path-dependent micro material model requires small micro load steps that limit the size of the macro load step in the standard implementation. This restriction is again relaxed with the introduction of the two-level path-following algorithm.

Multi-scale methods were used for investigation of different perforated structures, some of which qualify as metamaterials, due to their unique responses. An analysis of meta-

materials with MIEL methods was able to capture the actual response that occurred due to local buckling, which classical FE^2 is not able to do.

The second part of this thesis was devoted to setting up multi-scale optimization algorithms. A gradient-based optimization algorithm was wrapped around a multi-scale analysis. The convenience of versatile use of sensitivity analysis was proven. At the micro level, shape sensitivity analysis was applied for the optimization procedure, in addition to the essential boundary condition sensitivity analysis adopted for connecting the macro and micro scales. Furthermore, a sensitivity analysis was also carried out at the macro level for the evaluation of sensitivities needed for gradient calculation as a part of the optimization algorithm.

The developed optimization algorithm was employed to obtain the optimal size and distribution of openings across the material and to fulfill different conditions. With these patterns, certain mechanical effects can be achieved that are interesting for applications in smart structures. An optimization algorithm was implemented for MIEL and FE^2 methods for both 2D and 3D cases. A combination of nonlinear multi-scale methods with an optimization procedure results in a computationally demanding operation. For FE^2, geometry optimization at the macro level was combined with optimization of RVE perforation at the micro level. With this approach, the search for optimal material characteristics and macro geometry are simultaneously addressed.

An FE^2 optimization algorithm for accumulated plastic energy was proposed. Because of the non-linearity of the plastic material model, the solution of the problem requires several load steps. Utilizing this algorithm, metamaterials which exhibit large plastic energy dissipation for a minimal amount of material were designed. This property was achieved via a variation of perforation across the domain.

8 RAZŠIRJEN POVZETEK

8.1 Uvod

V prvem delu doktorske disertacije je predstavljen posplošen pristop implementacije FE^2 in MIEL (mreža v elementu) večnivojskih metod preko občutljivostne analize bistvenih robnih pogojev. Implementacija na osnovi občutljivostne analize je izpeljana kot alternativa standardni implementaciji večnivojske analize, pri kateri se za povezavo med nivojema uporabi izračun Schurovega komplementa tangentne matrike mikro nivoja. Za večnivojski metodi FE^2 in MIEL je bila izpeljana numerično učinkovita notacija z avtomatskim odvajanjem za občutljivostno analizo prvega in drugega reda glede na bistvene robne pogoje. Za reševanje večnivojskih problemov je bila vpeljana konsistentno linearizirana dvonivojska metoda sledenja poti. Občutljivostna analiza omogoča, da lahko vsakemu makro koraku sledi poljubno število mikro podkorakov pri čemer se ohrani kvadratična konvergenca celotnega algoritma za reševanje. V drugem delu disertacije je opisana implementacija večnivojskega optimizacijskega algoritma, kjer je gradientna optimizacija izvedena v zanki okoli večnivojske procedure reševanja.

8.2 Večnivojski problemi

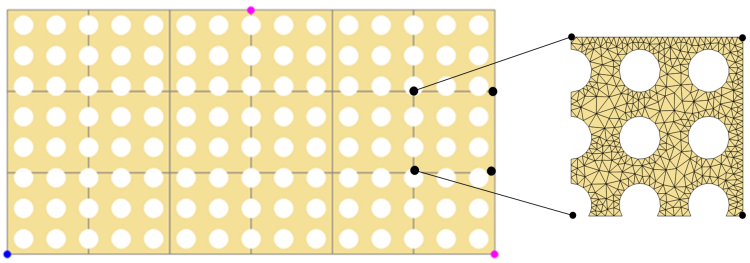
Večnivojski problemi so značilni za modeliranje heterogenih materialov, kot so na primer kompozitni materiali ojačani z vlakni, beton in celo kovine. Za njihovo reševanje se uporabljajo večnivojske metode.

8.2.1 Večnivojske metode

Večnivojske metode omogočajo direktno analizo vpliva materialnega odziva na mikro nivoju na makroskopski materialni odziv. Uporaba različnih metod je omejena s karakteristikami problema, ki ga želimo rešiti. Cilj večnivojskega modeliranja je, da se z uporabo večnivojskih algoritmov izognemo modeliranju celotnega problema na nižjem nivoju in potrebi po makro modelu, a hkrati dobimo tudi rezultate željene natančnosti. Obravnavali smo dve različni večnivojski metodi, MIEL in FE^2, ki se lahko uporabita za reševanje velikega spektra problemov. Lahko se uporabita za primere, ko sta nivoja močno povezana kot tudi za tiste za katere velja predpostavka, da sta nivoja med seboj ločena.

MIEL

Metoda MIEL (mreža v elementu) ja večnivojska metoda končnih elementov, ki jo lahko uvrstimo v skupino metod dekompozicije domene. Ta metoda je primerna za uporabo, ko je razlika med nivojema končno velika in nivoja ostaneta povezana, ali ko v območju visokih gradientov metoda FE^2 odpove. Metoda MIEL je bila opisana v [39, 40, 47].



Slika 1: MIEL makro in mikro model

Obravnavata se močno povezana nivoja, kjer se metoda končnih elementov uporabi na obeh nivojih, a je merilo na mikro nivoju končno manjše od makro merila. Z ustrezno formulacijo se količine makro končnih elementov (rezidual, tangentna matrika, itd.) dobijo iz modela s končnimi elementi na mikro nivoju in tako zamenjajo konstitutivni zakon na makro nivoju. Vse vrednosti pomikov na robu mikro mreže so enake vrednosti makro pomikov na makro mreži v isti točki Slika 1, tako da je zagotovljena kompatibilnost pomikov.

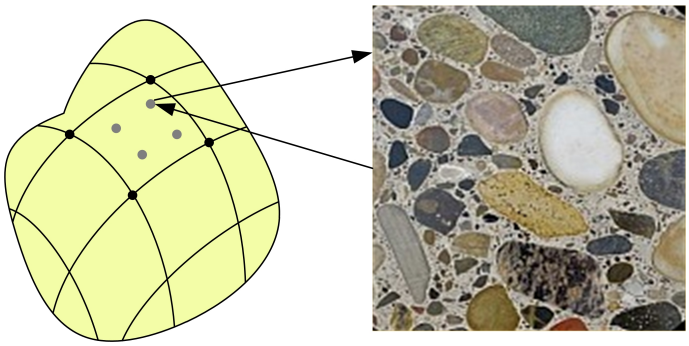
Virtualno delo notranjih sil na mikro nivoju je enako virtualnemu delu notranjih sil na makro elementu. Pri tem je $\boldsymbol{F}_m$ mikro deformacijski gradient in $\boldsymbol{P}_m$ mikro napetostni tenzor, $\boldsymbol{F}_M$ in $\boldsymbol{P}_M$ pa sta deformacijski gradient in napetostni tenzor makro elementa

$$\int_{\Omega_{Me}} \boldsymbol{P}_M : \delta \boldsymbol{F}_M \, dV = \int_{\Omega_m} \boldsymbol{P}_m : \delta \boldsymbol{F}_m \, dV \, . \tag{1}$$

FE^2

FE^2 je standardna dvonivojska homogenizacijska metoda, ki je primerna, ko se merili med seboj dovolj razlikujeta in sta nivoja le šibko povezana. Podroben opis je mogoče najti v [32]. Na makro nivoju se uporabi metoda končnih elementov in v vsaki Gaussovi točki, se predpiše reprezentativni volumen (RVE), ki predstavlja pripadajočo mikrostrukturo Slika 2. Predpostavi se, da obstaja RVE, ki je hkrati zadosti manjši od makro nivoja in

dovolj velik, da opiše heterogenosti na mikro nivoju. Materialni odziv dobimo iz analize s končnimi elementi na nivoju RVE-ja, ki zamenja konstitutivni zakon na makro nivoju.



Slika 2: FE^2 makro in RVE model

Povezava med makro in mikro nivojem temelji na principu povprečenja. Teorem povprečenja energije, znan tudi kot Hill-Mandelov kriterij, zahteva enakost med virtualnim delom variacije makroskopskih deformacij in volumskega povprečja virtualnega dela variacije mikroskopskih deformacij RVE-ja

$$\boldsymbol{P}_M : \delta \boldsymbol{F}_M = \frac{1}{V_0} \int_{\Omega_m} \boldsymbol{P}_m : \delta \boldsymbol{F}_m \, dV = \{ \boldsymbol{P}_m : \delta \boldsymbol{F}_m \} \, . \tag{2}$$

Pri tem je $\boldsymbol{P}_M$ prvi Piola–Kirchhoffov napetostni tenzor povezan z variacijo deformacijskega gradienta $\delta \boldsymbol{F}_M$ in je $\boldsymbol{F}_m$ deformacijski gradient na mikro nivoju, ter je $\boldsymbol{P}_m$ prvi Piola–Kirchhoffov napetostni tenzor na mikro nivoju.

8.2.2 ADB notacija

Bistvo notacije na osnovi avtomatskega odvajanja (ADB notacija) je avtomatsko odvajanje, v nadaljevanju AD. AD se uporablja za izvrednotenje analitičnih odvodov katerekoli poljubno kompleksne funkcije, definirane z algoritmom, preko verižnega pravila in predstavlja alternativno rešitev numeričnemu in simbolnemu odvajanju. Rezultat AD se imenuje "algoritmični"odvod in ga zapišemo z $\frac{\hat{\delta} f(\mathbf{a})}{\hat{\delta} \mathbf{a}}$. Operator AD $\frac{\hat{\delta} f(\mathbf{a})}{\hat{\delta} \mathbf{a}}$ predstavlja odvod funkcije $f(\mathbf{a})$ po spremenljivkah $\mathbf{a}$. Operator $\frac{\hat{\delta} f(\mathbf{a})}{\hat{\delta} \mathbf{a}}$ ima dvojni namen, saj nakazuje na matematično operacijo odvajanja hkrati pa izraža dejstvo, da je za izvrednotenje količin uporabljena AD metoda. Če je potrebno pri odvajanju upoštevati alternativne ali dodatne odvisnosti

posredne spremenljivke **b**, potem se AD izjema zapiše na naslednji način

$$\left.\frac{\hat{\delta} f(\mathbf{a},\mathbf{b})}{\hat{\delta}\mathbf{a}}\right|_{\frac{D\mathbf{b}}{D\mathbf{a}}=\mathbf{M}} = \frac{\partial f}{\partial \mathbf{a}} + \frac{\partial f}{\partial \mathbf{b}}\mathbf{M}\,, \tag{3}$$

ki nakazuje, da je med postopkom AD, totalni diferencial spremenljivke **b** po spremenljivki **a**, enak matriki **M**. Pri tem ni nujno, da je **b** v (3) algoritmično odvisna od **a**. Kadar je **b** algoritmično odvisen od **a**, takrat (3) definira, da so dejanski odvodi $\frac{\partial \mathbf{b}}{\partial \mathbf{a}}$ zanemarjeni in zamenjani z matriko **M**. Kadar **b** ni algoritmično odvisna od **a**, takrat (3) definira umetno odvisnost med **a** in **b**. AD izjeme so osnova za ADB notacijo mehanskega problema.

8.2.3 Implementacija

Za numerične analize in razvoj algoritmov smo uporabili programa *AceGen* in *AceFEM* [30], ki delujeta znotraj orodja za simbolno računanje Wolfram Mathematica. *AceGen* je avtomatski generator kode končnih elementov, s katerim se izognemo prekomernemu naraščanju velikosti programske kode [26]. Združuje sposobnosti Wolfram Mathematice, avtomatskega odvajanja, avtomatskega generiranja kode in simultane optimizacije izrazov. *AceFEM* je okolje za računanje s končnimi elementi. Programa omogočata analitično občutljivostno analizo prvega in drugega reda [27], ki smo jo uporabili za implementacijo večnivojskih metod končnih elementov, ter pri gradientni optimizaciji.

8.2.4 Definicija mikro problema

Na mikro nivoju obravnavamo formulacijo s končnimi elementi za poljuben nelinearen material, kot primer je izbran elasto-plastični materialni model za velike deformacije [54]. Formulacija je osnovana na razcepu deformacijskega gradienta $\boldsymbol{F}_m$.

Standardna šibka oblika ravnotežnih enačb je zapisana z

$$\int_{\Omega_m} \boldsymbol{P}_m : \delta \boldsymbol{F}_m \, dV - \int_{\partial\Omega_m} \boldsymbol{t} \cdot \delta \boldsymbol{u}_m \, dS = 0 \tag{4}$$

kjer je $\boldsymbol{P}_m$ prvi Piola-Kirchhofov tenzor napetosti, ki ga lahko dobimo iz elastične energije W z $\boldsymbol{P}_m = \partial W / \partial \boldsymbol{F}_m$. Po diskretizaciji deformacijskega gradienta s končnimi elementi dobimo $\boldsymbol{F}_m = \boldsymbol{F}_m(\mathbf{p}_{me})$, kjer je $\mathbf{p}_{me}$ set vozliščnih prostostnih stopenj e-tega mikro elementa

v trenutnem obtežnem koraku. Variacija $\delta \boldsymbol{F}_m = \partial \boldsymbol{F}_m / \partial \mathbf{p}_{me}\, \delta \mathbf{p}_{me}$ skupaj s standardno Gaussovo integracijo šibke oblike (4) in upoštevanju prispevkov posameznih elementov rezultira v setu algebrajskih enačb (5). V primeru elasto-plastičnih modelov so enačbe na vsaki Gaussovi integracijski točki povezane z dodatnim setom enačb $\mathbf{Q}_{mg}$ (6).

$$\begin{aligned}\mathbf{R}_m(\mathbf{p}_m, \mathbf{h}_m) &= \bigA_{e=1}^{n_{me}} \mathbf{R}_{me} + \mathbf{R}_m^{\text{ext}} \\ &= \bigA_{e=1}^{n_{me}} \sum_{g \in G_e} w_{gp}\, \mathbf{R}_{mg}(\mathbf{p}_{me}, \mathbf{h}_{mg}) + \mathbf{R}_m^{\text{ext}} = \mathbf{0}, \end{aligned} \tag{5}$$

$$\mathbf{Q}_{mg}(\mathbf{p}_{me}, \mathbf{h}_{mg}, \mathbf{h}_{mgn}) = \mathbf{0} \,:\, g = 1, 2, ...n_{tg} \tag{6}$$

$\mathbf{R}_{me}$ je prispevek e-tega elementa k globalnemu rezidualu $\mathbf{R}_m$ in $\mathbf{R}_m^{\text{ext}}$, je vektor zunanjih sil. $\mathbf{p}_m$ predstavlja set vozliščnih neznank mikro nivoja in $\mathbf{h}_m = \bigcup_g^{n_{tg}} \mathbf{h}_{mg}$, je set neznank vseh Gaussovih točk problema. G_e je set Gaussovih točk e-tega elementa in w_{gp} so uteži Gaussovih točk. Prispevek Gaussove točke $\mathbf{R}_{mg}$ k rezidualu elementa $\mathbf{R}_{me}$ dobimo iz šibke oblike

$$\mathbf{R}_{mg} = J_\xi \boldsymbol{P}_m : \frac{\partial \boldsymbol{F}_m}{\partial \mathbf{p}_{me}}\,, \tag{7}$$

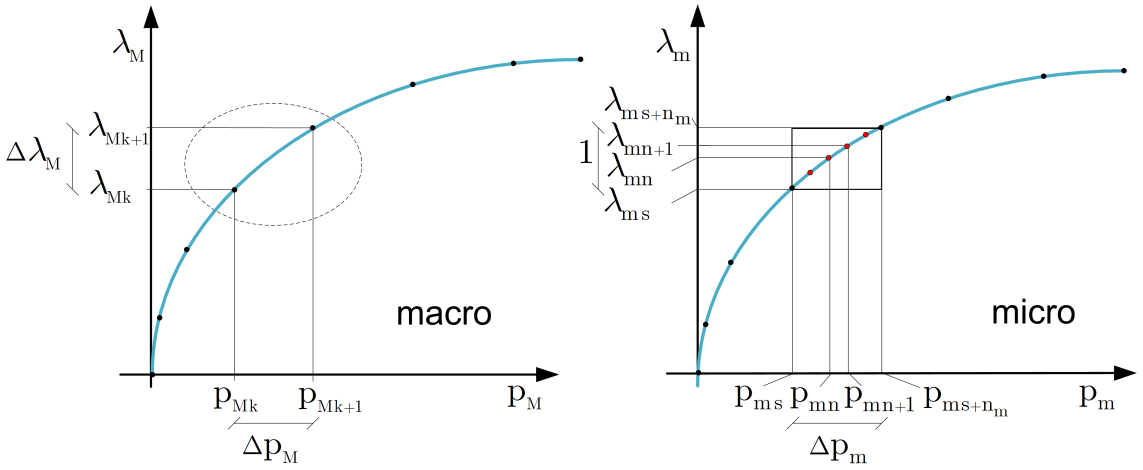
kjer je J_ξ standardni Jacobian transformacije iz referenčnega v globalni koordinatni sistem. Numerično učinkovito ADB obliko (7) zapišemo z

$$\mathbf{R}_{mg} = J_\xi\, \boldsymbol{P}_m : \frac{\partial \boldsymbol{F}_m}{\partial \mathbf{p}_{me}} = J_\xi\, \frac{\hat{\delta} W}{\hat{\delta} \boldsymbol{F}_m}\bigg|_{\frac{D\mathbf{h}_{mg}}{D\boldsymbol{F}_m}=\mathbf{0}} : \frac{\hat{\delta} \boldsymbol{F}_m}{\hat{\delta} \mathbf{p}_{me}} = J_\xi\, \frac{\hat{\delta} W}{\hat{\delta} \mathbf{p}_{me}}\bigg|_{\frac{D\mathbf{h}_{mg}}{D\boldsymbol{F}_m}=\mathbf{0}}. \tag{8}$$

Kot stranski produkt iterativnega reševanja enačb na Gaussovih točkah nastane implicitna (algoritmična) odvisnost $\mathbf{h}_{mg}$ od $\boldsymbol{F}_m$. Z AD izjemo $\frac{D\mathbf{h}_{mg}}{D\boldsymbol{F}_m} = \mathbf{0}$ v (8) to odvisnost skrijemo pred avtomatskim odvajanjem in tako dobimo pravilne ravnotežne enačbe.

8.3 Dvonivojski algoritem sledenja poti

Za nelinearne probleme v splošnem do rešitve ni mogoče priti le v enem koraku, zato uporabimo katerega od algoritmov za sledenje poti. Pri standardni implementaciji večnivojskih metod je značilno, da je parametriziran samo makro nivo. Posledično vsakemu makro koraku sledi točno en mikro korak in je algoritem za sledenje poti uporabljen samo na globalnem nivoju. V disertaciji smo izdelali algoritem za konsistentno parametrizacijo obeh nivojev to je dvonivojski algoritem sledenja poti.

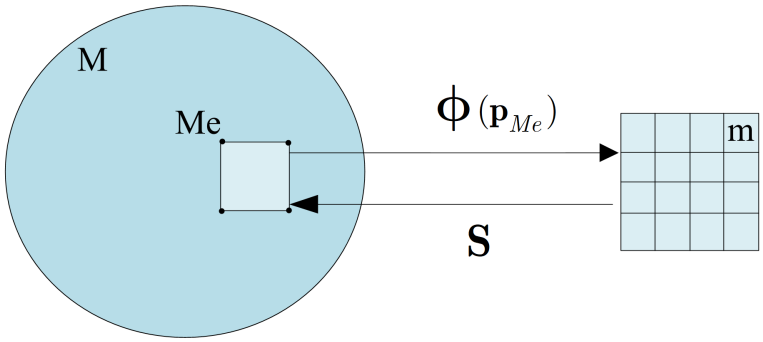


Slika 3: Posplošeno dvonivojsko sledenje obtežni poti

Rezultat diskretizacije s končnimi elementi na makro nivoju je set nelinearnih enačb $\mathbf{R}_M$ za trenutni obtežni korak $\lambda_M = \lambda_{M\,k+1}$

$$\mathbf{R}_M(\mathbf{p}_M, \bigcup_{e=1}^{n_{Me}} \mathbf{S}_{Me}, \lambda_M) = \bigwedge_{e=1}^{n_{Me}} \mathbf{R}_{Me}(\mathbf{p}_{Me}, \mathbf{S}_{Me}) - \lambda_M \, \mathbf{R}_M^{\text{ref}} = \mathbf{0}\,, \tag{9}$$

kjer je $\mathbf{R}_{Me}$ prispevek notranjih sil e-tega makro elementa k vektorju vozliščnih sil in je $\mathbf{R}_M^{\text{ref}}$ referenčni obtežni vektor. $\mathbf{p}_M$ predstavlja set vozliščnih neznank na makro nivoju in $\mathbf{S}_{Me}$ set spremenljivk, ki se prenese z mikro nivoja na e-ti makro element. $\mathbf{S}_{Me}$ je sestavljen iz prispevkov enega ali več mikro problemov. Prenos podatkov med makro in mikro nivojem je prikazan na Sliki 4.



Slika 4: Prenos podatkov med makro in mikro nivojem

Povezava med nivoji se lahko pri večnivojskih metodah izvede na različne načine. V našem primeru se mikro-makro povezava vzpostavi tako, da se robni pogoji mikro problema izrazijo kot funkcije podatkov izračunanih na makro nivoju. Naj bo $\bar{\mathbf{p}}_m$ set vozliščnih

neznank z vsiljenimi homogenimi (Dirichletovimi) robnimi pogoji, $\boldsymbol{\phi}$ je set spremenljivk izračunanih na makro nivoju za trenutni obtežni nivo λ_M od katerih je odvisen izbrani mikro problem in $\bar{\mathbf{p}}_m(\boldsymbol{\phi})$ takšna funkcija, za katero na koncu makro koraka velja $\bar{\mathbf{p}}_m = \bar{\mathbf{p}}_m(\boldsymbol{\phi})$. $\boldsymbol{\phi}$ sestavljajo komponente makro deformacijskega gradienta v primeru metode FE^2 in vozliščni pomiki makro elementa v primeru metode MIEL.

Dvonivojski algoritem sledenja poti je prikazan na Sliki 3. Najprej se rešijo enačbe na mikro nivoju, ki jih dobimo iz (5) z diskretizacijo s končnimi elementi na mikro nivoju.

$$\mathbf{R}_m(\mathbf{p}_m, \mathbf{h}_m, \bar{\mathbf{p}}_m(\boldsymbol{\phi}, \lambda_m)) = \mathop{\mathsf{A}}_{e=1}^{n_{me}} \sum_{g \in G_e} w_{gp}\, \mathbf{R}_{mg}(\mathbf{p}_{me}(\boldsymbol{\phi}, \lambda_m), \mathbf{h}_{mg}) + \mathbf{R}_m^{\text{ext}} = \mathbf{0} \tag{10}$$

$$\mathbf{Q}_{mg}(\mathbf{p}_{me}(\boldsymbol{\phi}, \lambda_m), \mathbf{h}_{mg}, \mathbf{h}_{mgn}) = \mathbf{0} \,:\, g = 1, 2, ...n_{tg} \tag{11}$$

Ravnotežne enačbe $\mathbf{R}_m$ so povezane z evolucijskimi enačbami na Gaussovih točkah $\mathbf{Q}_{mg}$. Algebrajske enačbe za neznane $\mathbf{p}_m$ in $\mathbf{h}_m$ pri fiksni vrednosti $\mathbf{p}_M$ se rešijo z uporabo vgnezdene Newtonove metode. Newtonova metoda se uporabi tudi za rešitev ravnotežnih enačb na makro nivoju (9) v zunanji zanki, kar vodi do vgnezdene sheme iteracija-poditeracija za neznane $\mathbf{p}_M$, $\mathbf{p}_m$ in $\mathbf{h}_m$.

Za konsistentno linearizacijo vgnezdene sheme morajo kode končnih elementov podpirati občutljivostno analizo prvega in drugega reda. Za izvrednotenje tangentnega modula $\mathbf{K}_M$ bi sicer lahko uporabili tudi končne diference, a bi netočna tangentna matrika lahko vplivala na konvergenco iterativnega reševanja.

Za implementacijo večnivojskega algoritma je potrebno definirati naslednje količine:

- občutljivostne parametre mikro problem kot funkcijo neznank makro elementa ($\boldsymbol{\phi}(\mathbf{p}_{Me})$),
- robne pogoje mikro problema kot funkcijo občutljivostnih parametrov ($\bar{\mathbf{p}}_m(\boldsymbol{\phi}, \lambda_m)$),
- odvode robnih pogojev po občutljivostnih parametrih ($\frac{D\bar{\mathbf{p}}_m(\boldsymbol{\phi}, \lambda_m)}{D\boldsymbol{\phi}}$),
- spremenljivke mikro nivoja, ki se prenesejo na makro nivo ($\mathbf{S}$),
- totalni odvod spremenljivk mikro nivoja po občutljivostnih parametrih ($D\mathbf{S}/D\boldsymbol{\phi}$),
- rezidual makro elementa ($\mathbf{R}_{Me}(\mathbf{p}_{Me}, \mathbf{S}_{Me})$),
- tangentno matriko makro elementa ($\mathbf{K}_{Me}(\mathbf{p}_{Me}, \mathbf{S}_{Me}, D\mathbf{S}_{Me}/D\boldsymbol{\phi}_{Me})$).

8.4 Večnivojske metode implementirane z občutljivostno analizo

Cilj občutljivostne analize je izračun odvodov poljubnega funkcionala odziva glede na izbrane parametre. Funkcional odziva je lahko odvisen od poljubnih vhodnih podatkov modela analize (materialnih konstant, velikosti in porazdelitve obtežbe, parametrov oblike itd.), pa tudi od poljubnih vmesnih ali končnih rezultatov analize. Za izračun občutljivosti se lahko uporabljajo različne metode, kot so metoda končnih diferenc, analitična občutljivostna analiza, ki je lahko direktna ali pa pridružena, ter delno analitična občutljivostna analiza. Izraze potrebne za analitično občutljivostno analizo lahko izpeljemo ročno ali pa samodejno z uporabo programske opreme za avtomatsko odvajanje.

Implementacija večnivojskih metod preko analitične občutljivostne analiza, glede na bistvene robne pogoje, ima številne prednosti v primerjavi s klasično implementacijo, ki temelji na izračunu Schurovega komplementa tangentne matrike mikro problema. To je še posebej pomembno pri povezanih problemih, odvisnih od poti, kot je plastičnost, kjer je konsistentna linearizacija problema zelo pomembna. V disertaciji je pokazano, da za metodo FE^2 potrebujemo analitično občutljivostno analizo prvega reda glede na bistvene robne pogoje (EBC) in za metodo MIEL občutljivostno analizo drugega reda glede na bistvene robne pogoje. Ugotovljeno je bilo, da je z numeričnega vidika tovrstna implementacija boljša od implementacije s Schurovim komplementom.

8.4.1 Formulacija MIEL metode

Na makro nivoju imamo kompatibilno interpolacijo neznanih polj po robu elementa, medtem ko so materialne karakteristike, nehomogenosti, notranja struktura, kot so odprtine in vključki različnih materialov definirani samo na mikro nivoju. Na Sliki 5 je predstavljena procedura MIEL. Standardna interpolacija pomikov $\boldsymbol{u}_M$ na robu makro elementa je zapisana z

$$\boldsymbol{u}_M(\boldsymbol{\Xi}) = \sum_i N_i(\boldsymbol{\Xi})\, \boldsymbol{u}_{Mei}. \tag{12}$$

$N_i(\boldsymbol{\Xi})$ so oblikovne funkcije končnega elementa, $\boldsymbol{\Xi} = (\xi, \eta, \zeta)$ referenčne koordinate in $\boldsymbol{u}_{Mei}$ pomiki v i-tem vozlišču makro elementa. Za zagotovitev kompatibilnosti med pomiki makro in mikro nivoja vsilimo robne pogoje po robu mikro mreže

$$\bar{\boldsymbol{u}}_m(\boldsymbol{\Xi}) = (\bar{\boldsymbol{u}}_{ms}(\boldsymbol{\Xi}) + (\lambda_m - \lambda_{ms})(\boldsymbol{u}_M(\boldsymbol{\Xi}) - \bar{\boldsymbol{u}}_{ms}(\boldsymbol{\Xi}))). \tag{13}$$

$\bar{\boldsymbol{u}}_{ms}(\boldsymbol{\Xi})$ so pomiki na robu v zadnjem makro koraku. Odvodi (13) po komponentah vozliščnih pomikov makro elementa so podani z

$$\frac{\partial \bar{u}_{mi}(\boldsymbol{\Xi})}{\partial u_{Mejk}} = \delta_{ik}(\lambda_m - \lambda_{ms}) N_j(\boldsymbol{\Xi}). \tag{14}$$

Set makro neznank je $\mathbf{p}_{Me} = \bigcup_{j,k} u_{Mejk}$ in $\mathbf{p}_m$ je sestavljen iz vozliščnih pomikov mikro mreže. (13) definira odvisnost med prostostnimi stopnjami s predpisanimi robnimi pogoji na mikro nivoju $\bar{\mathbf{p}}_m = \bar{\mathbf{p}}_m(\mathbf{p}_{Me}, \lambda_m)$ in neznankami makro elementa $\mathbf{p}_{Me}$.

Rezidual makro elementa $\mathbf{R}_{Me}$ je v primeru MIEL dobljen z integralom virtualnega dela notranjih sil (4) po mikro mreži, kjer je deformacijski gradient $\boldsymbol{F}_m = \boldsymbol{F}_m(\mathbf{p}_{me}(\mathbf{p}_{Me}, \lambda_m))$ in mikro napetostni tenzor $\boldsymbol{P}_m = \boldsymbol{P}_m(\mathbf{p}_{me}(\mathbf{p}_{Me}, \lambda_m))$ implicitno odvisen od prostostnih stopenj makro elementa.

$$\int_{\Omega_{Me}} \boldsymbol{P}_M : \delta \boldsymbol{F}_M \, dV = \int_{\Omega_m} \boldsymbol{P}_m : \delta \boldsymbol{F}_m \, dV = \sum_{e=1}^{n_{me}} \int_{\Omega_{me}} \boldsymbol{P}_m : \delta \boldsymbol{F}_m \, dV \tag{15}$$

Po diskretizaciji mikro domene Ω_m dobimo

$$\delta \boldsymbol{F}_m(\mathbf{p}_{me}(\mathbf{p}_{Me}, \lambda_m)) = \frac{\partial \boldsymbol{F}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}} \, \delta \mathbf{p}_{Me} \, . \tag{16}$$

Z Gaussovo integracijo po domeni mikro elementa dobimo rezidual makro elementa $\mathbf{R}_{Me}$

$$\mathbf{R}_{Me} = \sum_{e=1}^{n_{me}} \sum_{g \in G_e} w_{gp} \, \mathbf{R}_{Mg} \tag{17}$$

$$\mathbf{R}_{Mg} = J_\xi \, \boldsymbol{P}_m : \frac{\partial \boldsymbol{F}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}} \tag{18}$$

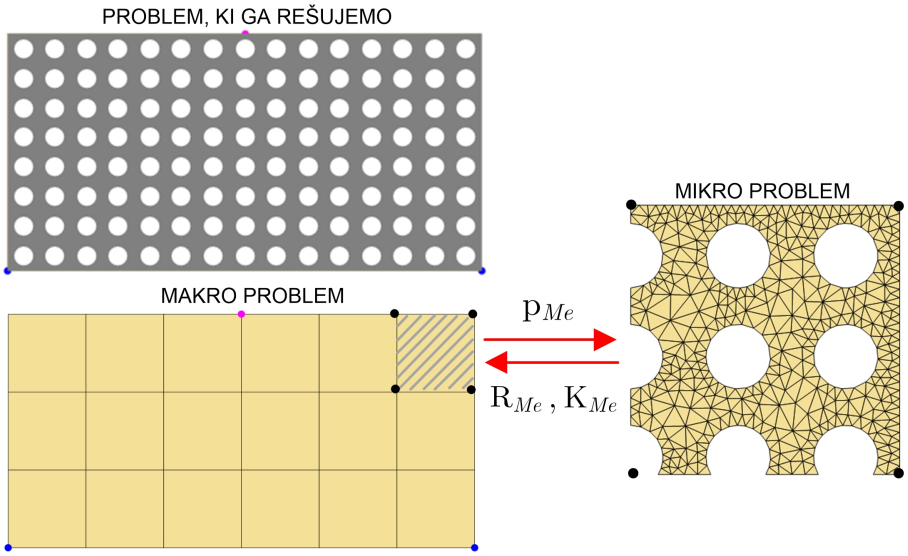
$\mathbf{R}_{Mg}$ je prispevek k rezidualu makro elementa, izračunan na Gaussovi točki mikro elementa. Z odvajanjem (18) dobimo tangentno matriko makro elementa

$$\mathbf{K}_{Me} = \sum_{e=1}^{n_{me}} \sum_{g \in G_e} w_{gp} \, \mathbf{K}_{Mg} \tag{19}$$

$$\begin{aligned} \mathbf{K}_{Mg} = \frac{\partial \mathbf{R}_{Mg}}{\partial \mathbf{p}_{Me}} = J_\xi \Bigg(& \frac{\partial \boldsymbol{P}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}} : \frac{\partial \boldsymbol{F}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}} \\ & + \boldsymbol{P}_m : \left(\frac{\partial^2 \boldsymbol{F}_m}{\partial {\mathbf{p}_{me}}^2} \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}} \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}} + \frac{\partial \boldsymbol{F}_m}{\partial \mathbf{p}_{me}} \frac{D^2\mathbf{p}_{me}}{D\mathbf{p}_{Me}^2} \right) \Bigg) \, . \end{aligned} \tag{20}$$

Kjer je $\mathbf{K}_{Mg}$ prispevek Gaussove točke mikro mreže k tangentni matriki makro elementa.

Rezidual in tangentna matrika sta za vsak makro element dobljena direktno iz problema na mikro nivoju. Makro element tako izvede le pravilen prenos količin.



Slika 5: MIEL večnivojska shema

Občutljivostno analizo potrebujemo za izračun implicitnih odvisnosti $\frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}}$ in $\frac{D^2\mathbf{p}_{me}}{D\mathbf{p}_{Me}^2}$ v (18) in (20). Iz (12) sledijo občutljivostni parametri mikro problema $\boldsymbol{\phi} = \boldsymbol{\phi}_{Me} = \mathbf{p}_{Me} = \bigcup_{j,k} u_{Mejk}$, in iz (13) ter (14) komponente hitrostnega polja ${}^{I}\mathbf{V} = \frac{\partial \bar{\mathbf{p}}_m(\boldsymbol{\phi},\lambda_m)}{\partial \phi_I}$. Komponente hitrostnega polja prvega reda za bistvene robne pogoje ${}^{I}\mathbf{V}$ so vrednosti oblikovnih funkcij makro elementa na položajih robnih vozlišč mikro mreže. Za robne pogoje izražene z linearno kombinacijo (13), so odvodi drugega reda enaki nič in je posledično tudi hitrostno polje drugega reda ${}^{IJ}\mathbf{V} = \mathbf{0}$.

Ostale količine, ki jih potrebujemo za dvonivojsko metodo sledenja poti so: spremenljivke makro nivoja, ki se pošljejo na makro nivo $\mathbf{S} = \mathbf{S}_{Me} = \mathbf{R}_{Me}$ (18) in totalni odvod $\frac{D\mathbf{S}}{D\boldsymbol{\phi}} = \frac{D\mathbf{S}_{Me}}{D\boldsymbol{\phi}_{Me}} = \mathbf{K}_{Me}$ (20). ADB oblika (18) in (20) je

$$\mathbf{R}_{Mg} = J_\xi \left.\frac{\hat{\delta}\mathbf{W}}{\hat{\delta}\mathbf{p}_{Me}}\right|_{\mathbf{h}_{mg}=const.,\,\frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}}=\mathbf{Y}_{\boldsymbol{\phi}}}, \tag{21}$$

$$\mathbf{K}_{Mg} = \left.\frac{\hat{\delta}\mathbf{R}_{Mg}}{\hat{\delta}\mathbf{p}_{Me}}\right|_{\frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}}=\mathbf{Y}_{\boldsymbol{\phi}},\,\frac{D\mathbf{Y}_{\boldsymbol{\phi}}}{D\mathbf{p}_{Me}}=\mathbf{Y}_{\boldsymbol{\phi}\boldsymbol{\phi}},} \tag{22}$$

kjer so $\mathbf{Y}_{\boldsymbol{\phi}} = \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}}$ in $\mathbf{Y}_{\boldsymbol{\phi}\boldsymbol{\phi}} = \frac{D^2\mathbf{p}_{me}}{D\mathbf{p}_{Me}^2}$ občutljivosti prvega in drugega reda.

8.4.2 Formulacija FE^2 metode

Pri metodi FE^2 imamo en reprezentativni volumen (RVE)K za vsako integracijsko točko, kot je to prikazano na Sliki 6. Vse informacije o mikro strukturi so pridobljene iz analize na mikro nivoju s povprečenjem odziva dobljenega iz ustrezne napetostne količine in konstitutivne tangentne matrike po RVE-ju. S prispevkom Gaussove točke k virtualnemu delu na makro nivoju ($\boldsymbol{P}_M : \delta\boldsymbol{F}_M$) in diskretizacijo deformacijskega gradienta na makro nivoju $\delta\boldsymbol{F}_M = \frac{\partial \boldsymbol{F}_M}{\partial \mathbf{p}_{Me}}\delta\mathbf{p}_{Me}$, dobimo

$$\mathbf{R}_{Me} = \sum_{g\in G_e} w_{gp}\,\mathbf{R}_{Mg}, \tag{23}$$

$$\mathbf{R}_{Mg} = J_\xi\,\boldsymbol{P}_M : \frac{\partial \boldsymbol{F}_M}{\partial \mathbf{p}_{Me}}, \tag{24}$$

kjer je prvi Piola-Kirchoffov napetostni tenzor makro nivoja $\boldsymbol{P}_M$ dobljen s povprečenjem tenzorja na mikro nivoju $\boldsymbol{P}_M = \{\boldsymbol{P}_m\}$. Periodični robni pogoji so doseženi tako, da se najprej v vogalne točke RVE-ja predpišejo pomiki z

$$\bar{\boldsymbol{u}}_m = \left(\boldsymbol{F}_{Ms} + (\lambda_m - \lambda_{ms})(\boldsymbol{F}_M - \boldsymbol{F}_{Ms}) - \mathbf{I}\right)\boldsymbol{X}_m \tag{25}$$

kjer je $\boldsymbol{F}_{Ms}$ makro deformacijski gradient na koncu zadnjega makro obtežnega koraka. Odvodi (25) po komponentah $\boldsymbol{F}_M$ so podani z

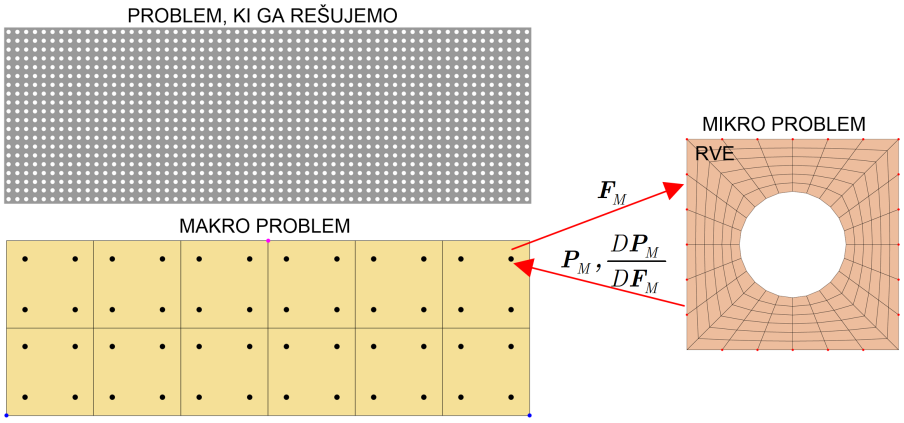
$$\frac{\partial \bar{u}_{mi}}{\partial F_{Mjk}} = \delta_{ij}(\lambda_m - \lambda_{ms})X_{mk}. \tag{26}$$

Z (25) so definirane odvisnosti $\bar{\mathbf{p}}_m = \bar{\mathbf{p}}_m(\mathbf{F}_M, \lambda_m)$ med setom mikro vozliščnih neznank s predpisanimi robnimi pogoji $\bar{\mathbf{p}}_m$ in makro deformacijskim gradientom $\mathbf{F}_M$. Za nepodprta robna vozlišča se uporabijo periodični robni pogoji z Lagrangevimi množitelji. Rezultat odvajanja (24) je tangentna matrika makro elementa

$$\mathbf{K}_{Me} = \sum_{g\in G_e} w_{gp}\,\mathbf{K}_{Mg} \tag{27}$$

$$\mathbf{K}_{Mg} = \frac{\partial \mathbf{R}_{Mg}}{\partial \mathbf{p}_{Me}} + \frac{\partial \mathbf{R}_{Mg}}{\partial \boldsymbol{P}_M}\frac{D\boldsymbol{P}_M}{D\boldsymbol{F}_M}\frac{\partial \boldsymbol{F}_M}{\partial \mathbf{p}_{Me}}\,. \tag{28}$$

$\frac{D\boldsymbol{P}_M}{D\boldsymbol{F}_M} = \{\frac{\partial \boldsymbol{P}_m}{\partial \mathbf{p}_{me}}\frac{D\mathbf{p}_{me}}{D\boldsymbol{F}_M}\}$ je makroskopska konstitutivna tangentna matrika dobljena s povprečenjem mikroskopskih konstitutivnih matrik.



Slika 6: FE^2 večnivojska shema

Občutljivostno analizo potrebujemo za izračun implicitnih odvisnosti $\frac{D\boldsymbol{P}_M}{D\boldsymbol{F}_M}$ v (28). Set občutljivostnih parametrov mikro problema je

$$\boldsymbol{\Phi} = \bigcup_{ij} F_{M\,ij} \tag{29}$$

in ${}^{I}\mathbf{V} = \frac{\partial \bar{\mathbf{p}}_m(\boldsymbol{\Phi},\lambda_m)}{\partial \phi_I}$ so komponente hitrostnega polja. Komponente hitrostnega polja prvega reda za robne pogoje ${}^{I}\mathbf{V}$ so ustrezne vozliščne koordinate vogalnih vozlišč mikro mreže. Za robne pogoje v obliki linearne kombinacije (25) so drugi odvodi enaki nič ${}^{IJ}\mathbf{V} = \mathbf{0}$. Spremenljivke z mikro nivoja, ki se prenesejo na makro nivo iz posameznega RVE-ja so $\mathbf{S} = \boldsymbol{P}_M = \{\boldsymbol{P}_m\}$ in totalni diferencial je $\frac{D\mathbf{S}}{D\boldsymbol{\Phi}} = \{\frac{\partial \boldsymbol{P}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\boldsymbol{F}_M}\}$. Za formulacijo makro elementa potrebujemo prispevke vseh mikro problemov na vseh Gaussovih točkah.

Celoten set spremenljivk, ki se pošljejo z makro elementa na mikro probleme je $\boldsymbol{\Phi}_{Me} = \bigcup_{g \in G_e} \boldsymbol{\Phi}^{(g)}$, kjer je G_e set Gaussovih točk e-tega makro elementa. Celoten set spremenljivk, ki se pošlje z mikro na makro element je $\mathbf{S}_{Me} = \bigcup_{r \in \mathcal{M}_e} \mathbf{S}^{(r)}$ in $\frac{D\mathbf{S}_{Me}}{D\boldsymbol{\Phi}_{Me}} = \bigcup_{r \in \mathcal{M}_e} \left(\frac{D\mathbf{S}}{D\boldsymbol{\Phi}}\right)^{(r)}$, kjer je $\mathcal{M}_e$ set mikro problemov povezanih z G_e.

Za numerično učinkovito implementacijo (24) in (28), potrebujemo ADB obliko teh dveh enačb, ki sledi iz $\boldsymbol{P}_M : \delta \boldsymbol{F}_M = \mathbf{S} : \delta \boldsymbol{F}_M$

$$\mathbf{R}_{Mg} = J_\xi \left. \frac{\hat{\delta}(\mathbf{S} : \boldsymbol{F}_M)}{\hat{\delta}\mathbf{p}_{Me}} \right|_{\mathbf{S}=const.} \tag{30}$$

$$\mathbf{K}_{Mg} = \left. \frac{\hat{\delta}\mathbf{R}_{Mg}}{\hat{\delta}\mathbf{p}_{Me}} \right|_{\frac{D\mathbf{S}}{D\boldsymbol{F}_M} = \frac{D\mathbf{S}}{D\boldsymbol{\Phi}}} \tag{31}$$

in

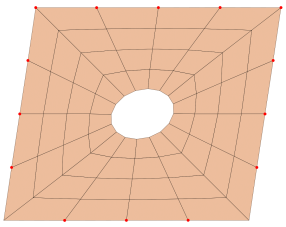
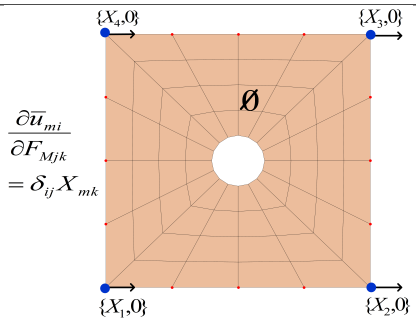
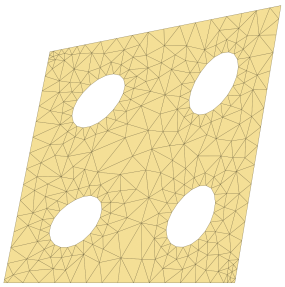
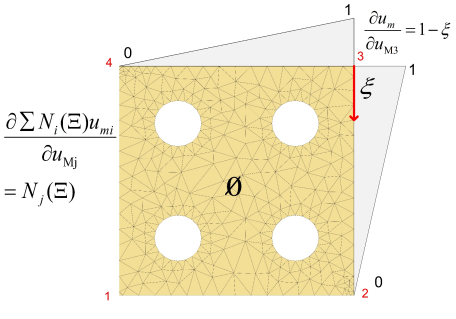
$$\frac{D\mathbf{S}}{D\boldsymbol{\phi}} = \left\{ \frac{\partial \boldsymbol{P}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{D\boldsymbol{F}_M} \right\} = \left\{ \left. \frac{\hat{\delta}\boldsymbol{P}_m}{\hat{\delta}\boldsymbol{F}_M} \right|_{\frac{D\mathbf{p}_{me}}{D\boldsymbol{F}_M} = \mathbf{Y}_{\boldsymbol{\phi}}} \right\} \tag{32}$$

kjer so $\mathbf{Y}_{\boldsymbol{\phi}} = \frac{D\mathbf{p}_{me}}{D\boldsymbol{F}_M}$ že izračunane in shranjene občutljivosti prvega reda.

8.4.3 Poenotena večnivojska metoda

Formulacija na osnovi avtomatskega odvajanja (ADB) omogoča poenotenje različnih večnivojskih metod za poljuben nelinearen, časovno odvisen povezan problem (npr. plastičnost s končnimi deformacijami). Za vse metode potrebujemo kode za končne elemente, ki podpirajo občutljivostno analizo prvega in drugega reda, ki se uporabi za izračun implicitnih odvisnosti, ki so odvodi neznank problema. Potrebujemo še dodatne podrutine, ki so odvisne od problema in se uporabijo za izvrednotenje homogenizirane konstitutivne matrike in makro napetosti za FE^2, ter reziduala in tangentne matrike za MIEL.

Tabela 1: Primerjava med FE^2 in MIEL

metoda	mikro nivo	hitrostno polje
FE^2		
MIEL		

Razlika med MIEL in FE^2 metodo je v bistvenih robnih pogojih mikro mreže in hitro-

stnem polju bistvenih robnih pogojev, ki ga potrebujemo za občutljivostno analizo, glej Preglednico 1. Makro element za FE^2 izvrednoti rezidual in tangentno matriko, medtem ko makro element za MIEL samo ustrezno prenese podatke z mikro na makro nivo. Preko občutljivostne analize implementirani metodi FE^2 in MIEL tako skupaj predstavljata poenoten pristop k avtomatizaciji večnivojskega modeliranja. Metode so v celoti paralelizirane za večjedrne procesorje in primerne tudi za uporabo na gručah. Enako velja za MIEL, kjer se vsak mikro problem lahko pošlje na drugo jedro. S paralelizacijo lahko močno skrajšamo računski čas kompleksnih problemov.

8.4.4 Numerični primeri

Numerični primeri so predstavljeni v poglavju 5. Kode končnih elementov za primarno analizo ter občutljivostno analizo prvega in drugega reda so bile avtomatsko izpeljane, optimizirane in zapisane v C jeziku z uporabo *AceGen* . Za implementacijo večnivojskih metod in izračun primerov smo uporabili *AceFEM* . Metodi MIEL in FE^2 sta bili implementirani preko občutljivostne analize bistvenih robnih pogojev (EBC), kot tudi z uporabo Schurovega komplementa. Predlagana implementacija preko EBC občutljivostne analize je bila verificirana na različnih 2D in 3D primerih. Analizirana je bila stopnja konvergence iterativnega postopka dvonivojskega sledenja poti prikazanega na Sliki 3. Primerjali smo stopnjo konvergence zadnjega makro obtežnega koraka v katerem je večina integracijskih točk že prešla v plastično stanje. Opazovali smo vpliv števila mikro korakov in vrsto implementacije ter pokazali, da je v primeru, ko enemu makro koraku sledi samo en mikro korak, konvergenca kvadratična, tako za implementacijo s Schur komplementom kot tudi za implementacijo z občutljivostno analizo. Ko enemu makro koraku sledi več mikro korakov pa se kvadratična konvergenca ohrani le v primeru implementacije preko občutljivostne analize.

Pokazano je bilo, da je implementacija preko občutljivostne analize numerično bolj učinkovita, saj je zaradi kvadratične konvergence potrebnih manj iteracij, da pridemo do končne rešitve. Dodatno pri metodi MIEL implementirani s Schur komplementom na računski čas vpliva tudi gostota mikro mreže. Ugotovljeno je bilo, da za učinkovito reševanje močno nelinearnih problemov potrebujemo dvonivojsko sledenje poti, z adaptivno določitvijo velikosti korakov na obeh nivojih, saj maksimalna velikost inkrementa obtežbe na mikro nivoju določa učinkovitost celotne simulacije.

8.5 Večnivojski optimizacijski algoritem

Optimizacija postaja vedno bolj pomembno orodje v inženirski praksi, predvsem zaradi omejenih materialnih virov, vpliva na okolje in konkurenčnosti na trgu. Optimizacija konstrukcij je disciplina, ki se ukvarja z optimalno zasnovo nosilnih konstrukcij. V gradbeništvu tako težimo k lahkim, nizko cenovnim konstrukcijam z dobrimi mehanskimi odzivi. Optimizacija nam omogoča izboljšanje izdelkov glede na dane kriterije, npr. želimo doseči minimalen volumen, ki bo zadostil pogoju zahtevane nosilnosti elementa. Optimizacijske metode se lahko uporabijo za oblikovanje konstrukcij, kot tudi za zasnovo mikro strukture. Metode lahko ločimo na diskretne in zvezne, ter glede na način iskanja rešitve na deterministične in stohastične. V splošnem optimizacijo konstrukcij ločimo glede na to, kaj se parametrizira, na optimizacijo velikosti, oblike in topologije. Za doseganje optimalnih rezultatov se lahko uporabi kombinacija različnih metod. Korak naprej v razvoju optimizacijskih algoritmov želimo narediti z uporabo gradientnega optimizacijskega algoritma za večnivojsko modeliranje, ki bo omogočal nelinearno večnivojsko optimizacijo.

Optimizacija konstrukcij na mejna stanja nosilnosti je običajno nelinearna, zato za reševanje potrebujemo metode s katerimi do optimalne rešitve pridemo na iterativni način z minimalnim možnim številom izvrednotenj funkcij. Pomembna vrsta optimizacijskih problemov je nelinearna optimizacija z omejitvami, pri kateri funkcija ni linearna in je vrednost parametrov omejena na določeno območje.

Definicija nelinearnega optimizacijskega problema

Najdi vektor

$$\boldsymbol{\alpha} = (\alpha_1, \alpha_2, ..., \alpha_\mathrm{n}) \in \mathbb{R}^\mathrm{n} \tag{33}$$

za katerega bo imela namenska funkcija $\mathrm{F}(\boldsymbol{\alpha})$ minimalno vrednost, pri dodatnih enakostih oz. neenakostih:

$$g_j(\boldsymbol{\alpha}) = 0, j \in P = \{1, ..., p\} \tag{34}$$

$$h_k(\boldsymbol{\alpha}) \geqslant 0, k \in Q = \{1, ..., q\} \tag{35}$$

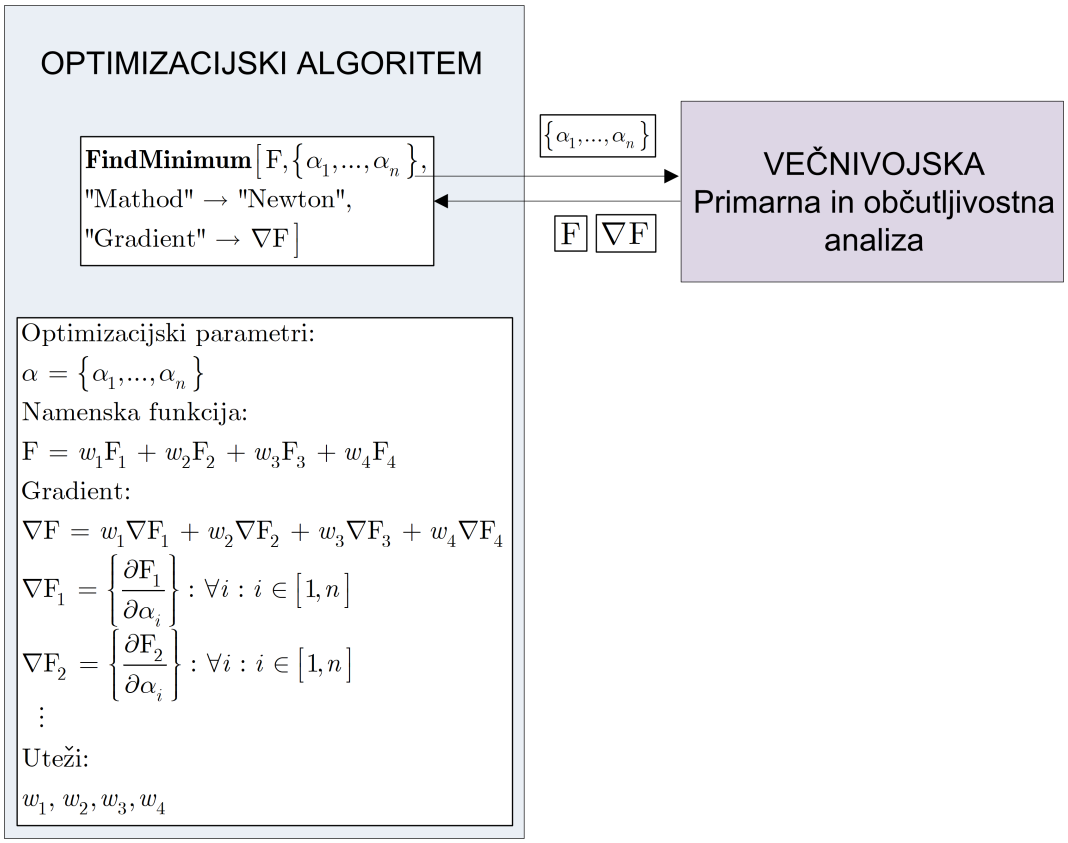
pri tem so F, g in h gladke in zvezno odvedljive, ter je vsaj ena od njih nelinearna. Set sprejemljivih rešitev označimo z M:

$$M = \{\boldsymbol{\alpha} \in \mathbb{R}^n : g_j(\boldsymbol{\alpha}) = 0, j \in P, h_k(\boldsymbol{\alpha}) \geqslant 0, k \in Q\} \tag{36}$$

Vektor $\boldsymbol{\alpha}$ je vektor optimizacijskih parametrov.

Optimizacijski problem z omejitvami (33) lahko v splošnem spremenimo v problem brez omejitev z uporabo kazenskih funkcij ali uporabo metod z Lagrangevimi množitelji. V Mathematici je za tovrstne primere implementirana metoda "interior point", ki prevede problem z omejitvami na problem brez omejitev.

8.5.1 Večnivojska gradientna optimizacija



Slika 7: Optimizacijska shema

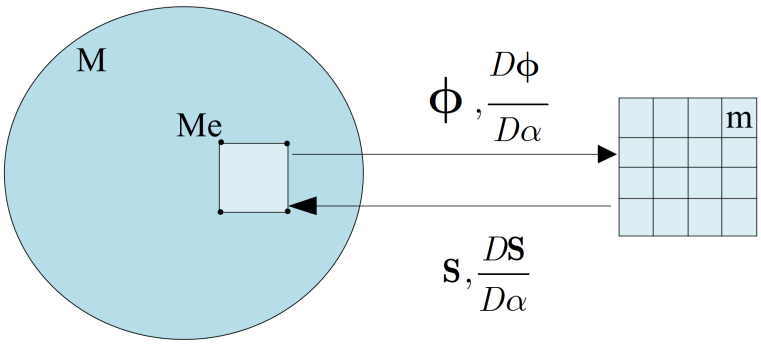
Slika 7 predstavlja večnivojski gradientni optimizacijski algoritem. Iščejo se optimalne vrednosti optimizacijskih parametrov α_i pri katerih bo imela namenska funkcija F minimalno vrednost. Uporabili smo `FindMinimum`, funkcijo implementirano v Mathematici, in optimizacijski algoritem brez omejitev s kvazi-Newtonovo metodo, ki potrebuje informacijo o gradientu namenske funkcije ∇F. Za izračun gradienta se uporabi občutljivostna analiza. Objektivna funkcija in njen gradient sta izvrednotena za optimizacijske parametre, potem ko je bila predhodno izvedena primarna in občutljivostna analiza na mikro nivoju.

Večnivojska občutljivostna analiza glede na bistvene robne pogoje (EBC) je združena z optimizacijsko občutljivostno analizo. Na mikro nivoju imamo EBC občutljivostno analizo s setom občutljivostnih parametrov $\boldsymbol{\phi}$. Optimizacijski občutljivostni parametri so $\boldsymbol{\alpha}$, tako je celoten set občutljivostnih parametrov na mikro nivoju enak

$$\boldsymbol{\phi} \cup \boldsymbol{\alpha} \,. \tag{37}$$

Občutljivostna analiza se lahko za oba seta parametrov izvede sočasno. Razlika je le v tem, da je večnivojska občutljivostna analiza relativna, optimizacijska občutljivostna analiza pa absolutna. Večnivojska občutljivostna analiza se pojavi le na mikro nivoju, medtem ko optimizacijska občutljivostna analiza zahteva občutljivostno analizo tako na mikro kot tudi na makro nivoju za ustrezno formulacijo.

Slika 8 prikazuje prenos podatkov med makro in mikro nivojem za optimizacijo. Z makro končnega elementa M_e se na mikro problem m prenesejo $\boldsymbol{\phi}$ in njihovi odvodi, glede na optimizacijske parametre $\frac{D\boldsymbol{\phi}}{D\boldsymbol{\alpha}}$. Z mikro problema pa se nazaj na makro element prenese set spremenljivk $\mathbf{S}$, ter njihovi odvodi glede na optimizacijske parametre $\frac{D\mathbf{S}}{D\boldsymbol{\alpha}}$. $\frac{D\mathbf{S}_{Me}}{D\alpha_i} = \bigcup_{e=1}^{n_{Me}} \frac{D\mathbf{S}}{D\alpha_i}$ je izračunan na mikro nivoju in prenesen na makro nivo. Količine $\mathbf{S}$ se uporabijo pri izračunu prispevka k rezidualu in tangentni matriki makro elementa in $\frac{D\mathbf{S}}{D\boldsymbol{\alpha}}$ pri izračunu prispevka k pseudo obtežnemu vektorju makro elementa, ki ga potrebujemo za občutljivostno analizo na makro nivoju.



Slika 8: Prenos podatkov med makro in mikro nivojem za optimizacijo

8.5.2 Numerični primeri za večnivojsko gradientno optimizacijo

Najprej smo obravnavali primer optimizacije z linearno-elastičnim materialnim modelom. Za metodi MIEL in FE^2 smo iskali optimalno razporeditev vrednosti perforacije po domeni, da se pri dani obtežbi doseže želeni pomik in je hkrati poraba materiala minimalna.

Vrednost perforacije v posameznem elementu oziroma Gaussovi točki je bila prevedena v geometrijsko karakteristiko na mikro nivoju, velikost radija odprtine r. Ta parameter vpliva na geometrijo mikro mreže, zato na mikro nivoju potrebujemo občutljivostno analizo glede na obliko domene problema. Optimizacijo smo začeli pri enakomerni poroznosti po celotni domeni. Dobljen končni rezultat optimalne razporeditve poroznosti je bil za metodi MIEL in FE^2 različen. Za MIEL je bila poroznost simetrična v obeh smereh, medtem ko je bila za FE^2 simetrična le glede na y-os.

Za projektiranje jeklenih konstrukcij na potresno obtežbo se uporablja princip projektiranja s sipanjem energije. Pri tem principu se predvidi, da na določenih mestih nastanejo plastični členki. Imamo zasnovo s šibkimi prečkami in močnim stebri, kjer se členki lahko oblikujejo le v prečkah in ne v stebrih. Iz tega koncepta izhaja ideja, da potrebujemo element, ki maksimalno disipira energijo pri minimalni porabi materiala. Metamaterial bi uporabili v konstrukcijskem elementu, ki bi se v primeru potresa lahko poškodoval in na ta način zaščitil preostali nosilni del konstrukcije. Sipana energija je pri tem enaka akumulirani plastični energiji W_p. Za predlagani metamaterial smo optimizirali variacijo poroznosti po celotni domeni, tako da smo dobili maksimalno akumulirano plastično energijo W_p za predpisani volumen. V primerjavi z začetnim primerom ko je bila poroznost konstantna po celotni domeni smo z optimizacijo povečali disipacijo energije za 20% pri monotoni obtežbi in za 70% pri ciklični obtežbi, ob isti porabi materiala.

8.6 Zaključki

Doktorska disertacija obravnava razvoj poenotenega pristopa k večnivojskemu modeliranju in večnivojskega optimizacijskega algoritma na osnovi občutljivostne analize. Implementacija večnivojskih metod MIEL in FE^2 na osnovi občutljivostne analize glede na bistvene robne pogoje je bila vpeljana kot alternativa tradicionalni implementaciji večnivojskih metod, ki zahteva izračun Schurovega komplementa tangentne matrike mikro nivoja, ki ga potrebujemo za izpeljavo povezave med nivojema. V disertaciji je pokazano, da za izračun tangente matrike pri metodi FE^2 potrebujemo občutljivostno analizo prvega reda in pri metodi MIEL občutljivostno analizo drugega reda glede na bistvene robne pogoje. Formulacija na osnovi avtomatskega odvajanja je predstavljena za občutljivostno analizo prvega in drugega reda, ki se lahko uporabi za poljuben materialni model.

Prednosti implementacije na osnovi občutljivostne analize so bile prepoznane in verifici-

rane z numeričnimi primeri. Analizirana je bila konvergenca rezultatov glede na velikost makro obtežnega koraka in glede na število podkorakov na mikro nivoju. Z dodatnimi podkoraki na mikro nivoju, je mogoče izboljšati točnost globalnega odziva, brez dodatnih računsko zahtevnih makro korakov. To še posebej pride do izraza pri primerih z močno nelinearnim odzivom na mikro nivoju, kjer je očitna prednost dvonivojskega algoritma za sledenje poti. Večnivojske metode so bile uporabljene za analizo različnih struktur z odprtinami, nekatere od teh lahko zaradi svojih neobičajnih lastnosti uvrstimo med metamateriale.

Drugi del disertacije obravnava formulacijo večnivojskega optimizacijskega algoritma. Gradientni optimizacijski algoritem je ovit okoli večnivojske analize. Pokazana je bila raznolika možnost uporabe občutljivostne analize. Na mikro nivoju se poleg občutljivostne analize glede na bistvene robne pogoje, ki jo potrebujemo za povezavo med nivojema, izvede še občutljivostna analiza oblike, za namen optimizacijskega algoritma. Zaradi optimizacije se na makro nivoju izvede dodatna občutljivostna analiza, za izračun gradienta namenske funkcije, ki je vhodni podatek pri optimizaciji. Kombinacija nelinearnih večnivojskih metod z optimizacijo ima za rezultat računsko zahtevne operacije, ki zahtevajo paralelno računanje.

Izdelani optimizacijski algoritem je bil uporabljen za izračun optimalne razporeditve odprtin po materialu na način, ki zadosti različnim zahtevam predpisanim z namensko funkcijo. Z različnimi vzorci odprtin, se lahko dosežejo mehanske lastnosti, ki so zanimive za aplikacijo v pametnih konstrukcijah. Za FE^2 je bil izdelan optimizacijski algoritem za izračun akumulirane plastične energije, ki ga lahko uporabimo za zasnovo matamateriala, ki omogoča veliko disipacijo energije ob minimalni porabi materiala.

BIBLIOGRAPHY

[1] Alexandersen, J., Lazarov, B. S., 2015. Topology optimisation of manufacturable microstructural details without length scale separation using a spectral coarse basis preconditioner. Computer Methods in Applied Mechanics and Engineering 290: 156–182. doi:10.1016/j.cma.2015.02.028.

[2] Babaee, S., Shim, J., Weaver, J. C., Chen, E. R., Patel, N., Bertoldi, K., 2013. 3D soft metamaterials with negative poissons ratio. Advanced Materials 25, 36: 5044–5049. doi:10.1002/adma.201301986.

[3] Cabras, L., Brun, M., 2014. Auxetic two-dimensional lattices with Poissons ratio arbitrarily close to -1. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences 470, 2172: 20140538–20140538. doi:10.1098/rspa.2014.0538.

[4] Castelletto, N., Hajibeygi, H., Tchelepi, H. A., 2017. Multiscale finite-element method for linear elastic geomechanics. Journal of Computational Physics 331: 337–356. doi:10.1016/j.jcp.2016.11.044.

[5] Christensen, P. W., Klarbring, A., 2008. An introduction to structural optimization. Springer Netherlands. doi:10.1007/978-1-4020-8666-3.

[6] Clausen, A., Aage, N., Sigmund, O., 2016. Exploiting additive manufacturing infill in topology optimization for improved buckling load. Engineering 2, 2: 250–257. doi:10.1016/j.eng.2016.02.006.

[7] Cresta, P., Allix, O., Rey, C., Guinard, S., 2007. Nonlinear localization strategies for domain decomposition methods: Application to post-buckling analyses. Computer Methods in Applied Mechanics and Engineering 196, 8: 1436–1446. doi:10.1016/j.cma.2006.03.013.

[8] Efendiev, Y., Hou, T., 2009. Multiscale finite element methods. Springer New York. doi:10.1007/978-0-387-09496-0.

[9] Eschenauer, H. A., Olhoff, N., 2001. Topology optimization of continuum structures: A review. Applied Mechanics Reviews 54, 4: 331. doi:10.1115/1.1388075.

[10] Evans, K. E., Alderson, A., 2000. Auxetic materials: functional materials and structures from lateral thinking! Advanced Materials 12, 9: 617–628. doi: 10.1002/(sici)1521-4095(200005)12:9⟨617::aid-adma617⟩3.0.co;2-3.

[11] Evans, K. E., Nkansah, M. A., Hutchinson, I. J., 1994. Auxetic foams: modelling negative poissons ratios. Acta Metallurgica et Materialia 42, 4: 1289–1294. doi: 10.1016/0956-7151(94)90145-7.

[12] Feyel, F., 1999. Multiscale FE2 elastoviscoplastic analysis of composite structures. Computational Materials Science 16, 1-4: 344–354. doi:10.1016/s0927-0256(99) 00077-4.

[13] Feyel, F., 2003. A multilevel finite element method (FE2) to describe the response of highly non-linear structures using generalized continua. Computer Methods in Applied Mechanics and Engineering 192: 3233–3244.

[14] Fu, M.-H., Chen, Y., Hu, L.-L., 2017. A novel auxetic honeycomb with enhanced in-plane stiffness and buckling strength. Composite Structures 160: 574–585. doi: 10.1016/j.compstruct.2016.10.090.

[15] Gao, W., Zhang, Y., Ramanujan, D., Ramani, K., Chen, Y., Williams, C. B., Wang, C. C., Shin, Y. C., Zhang, S., Zavattieri, P. D., 2015. The status, challenges, and future of additive manufacturing in engineering. Computer-Aided Design 69: 65–89. doi:10.1016/j.cad.2015.04.001.

[16] Gardan, J., 2018. Smart materials in additive manufacturing: state of the art and trends. Virtual and Physical Prototyping 14, 1: 1–18. doi:10.1080/17452759.2018. 1518016.

[17] Geers, M. G. D., Kouznetsova, V., Brekelmans, W., 2010. Multi-scale computational homogenization: trends and challenges. Journal of Computational and Applied Mathematics 234, 7: 2175–2182. doi:10.1016/j.cam.2009.08.077.

[18] Geers, M. G. D., Kouznetsova, V. G., Matouš, K., Yvonnet, J., 2017. Homogenization methods and multiscale modeling: nonlinear problems. doi:10.1002/9781119176817. ecm2107.

[19] Gibson, I., Rosen, D., Stucker, B., 2015. Additive manufacturing technologies. Springer New York. doi:10.1007/978-1-4939-2113-3.

[20] Giusti, S., Novotny, A., de Souza Neto, E., Feijóo, R., 2009. Sensitivity of the macroscopic elasticity tensor to topological microstructural changes. Journal of the Mechanics and Physics of Solids 57, 3: 555–570. doi:10.1016/j.jmps.2008.11.008.

[21] Hinojosa, J., Allix, O., Guidault, P.-A., Cresta, P., 2014. Domain decomposition methods with nonlinear localization for the buckling and post-buckling analyses of large structures. Advances in Engineering Software 70: 13–24. doi:10.1016/j.advengsoft.2013.12.010.

[22] Hou, T. Y., Wu, X.-H., 1997. A multiscale finite element method for elliptic problems in composite materials and porous media. Journal of Computational Physics 134, 1: 169–189. doi:10.1006/jcph.1997.5682.

[23] Hudobivnik, B., Korelc, J., 2016. Closed-form representation of matrix functions in the formulation of nonlinear material models. Finite Elements in Analysis and Design 111: 19–32. doi:10.1016/j.finel.2015.12.002.

[24] Ibrahimbegović, A., Markovič, D., 2003. Strong coupling methods in multi-phase and multi-scale modeling of inelastic behavior of heterogeneous structures. Computer Methods in Applied Mechanics and Engineering 192, 28-30: 3089–3107. doi:10.1016/s0045-7825(03)00342-6.

[25] Keulen, F., Haftka, R., Kim, N., 2005. Review of options for structural design sensitivity analysis. Part 1: Linear systems. Computer Methods in Applied Mechanics and Engineering 194, 30-33: 3213–3243. doi:10.1016/j.cma.2005.02.002.

[26] Korelc, J., 1997. Automatic generation of finite-element code by simultaneous optimization of expressions. Theoretical Computer Science 187, 1-2: 231–248. doi:10.1016/s0304-3975(97)00067-4.

[27] Korelc, J., 2009. Automation of primal and sensitivity analysis of transient coupled problems. Computational Mechanics 44, 5: 631–649. doi:10.1007/s00466-009-0395-2.

[28] Korelc, J., 2018. *AceGen* and *AceFEM* user manual. URL `www.fgg.uni-lj.si/symech/`.

[29] Korelc, J., Stupkiewicz, S., 2014. Closed-form matrix exponential and its application in finite-strain plasticity. International Journal for Numerical Methods in Engineering 98, 13: 960–987. doi:10.1002/nme.4653.

[30] Korelc, J., Wriggers, P., 2016. Automation of finite element methods. Springer International Publishing. doi:10.1007/978-3-319-39005-5.

[31] Korelc, J., Zupan, N., 2018. Unified approach to sensitivity analysis based automation of multiscale modelling. In Soric, J., Wriggers, P., Allix, O., Hg., Multiscale modeling of heterogeneous structures, (Lecture notes in applied and computational mechanics). Berlin: Springer International Publishing AG.

[32] Kouznetsova, V., Brekelmans, W. A. M., Baaijens, F. P. T., 2001. An approach to micro-macro modeling of heterogeneous materials. Computational Mechanics 27: 37–48. doi:10.1007/s004660000212.

[33] Ladevèze, P., Nouy, A., 2003. On a multiscale computational strategy with time and space homogenization for structural mechanics. Computer Methods in Applied Mechanics and Engineering 192, 28-30: 3061–3087. doi:10.1016/s0045-7825(03)00341-4.

[34] Lakes, R., 1993. Advances in negative poissons ratio materials. Advanced Materials 5, 4: 293–296. doi:10.1002/adma.19930050416.

[35] Lamut, M., Korelc, J., Rodič, T., 2011. Multiscale modelling of heterogeneous materials. Materiali in tehnologije 45: 421–426.

[36] Li, H., Luo, Z., Gao, L., Walker, P., 2018. Topology optimization for functionally graded cellular composites with metamaterials by level sets. Computer Methods in Applied Mechanics and Engineering 328: 340–364. doi:10.1016/j.cma.2017.09.008.

[37] Liu, H., Zhang, H., 2013. A uniform multiscale method for 3D static and dynamic analyses of heterogeneous materials. Computational Materials Science 79: 159–173. doi:10.1016/j.commatsci.2013.06.006.

[38] Lloberas-Valls, O., Rixen, D., Simone, A., Sluys, L., 2012. On micro-to-macro connections in domain decomposition multiscale methods. Computer Methods in Applied Mechanics and Engineering 225-228: 177–196. doi:10.1016/j.cma.2012.03.022.

[39] Markovič, D., Ibrahimbegović, A., 2004. On micro–macro interface conditions for micro scale based FEM for inelastic behavior of heterogeneous materials. Computer Methods in Applied Mechanics and Engineering 193, 48-51: 5503–5523. doi:10.1016/j.cma.2003.12.072.

[40] Markovič, D., Niekamp, R., Ibrahimbegović, A., Matthies, H. G., Taylor, R. L., 2005. Multi-scale modeling of heterogeneous structures with inelastic constitutive behaviour. Engineering Computations 22, 5/6: 664–683. doi:10.1108/02644400510603050.

[41] Mathematica, 2019 URL `www.wolfram.com`.

[42] Matouš, K., Geers, M. G., Kouznetsova, V. G., Gillman, A., 2017. A review of predictive nonlinear theories for multiscale modeling of heterogeneous materials. Journal of Computational Physics 330: 192–220. doi:10.1016/j.jcp.2016.10.070.

[43] Meng, L., Zhang, W., Quan, D., Shi, G., Tang, L., Hou, Y., Breitkopf, P., Zhu, J., Gao, T., 2019. From topology optimization design to additive manufacturing: today's success and tomorrow's roadmap. Archives of Computational Methods in Engineering doi:10.1007/s11831-019-09331-1.

[44] Michaleris, P., Tortorelli, D. A., Vidal, C. A., 1994. Tangent operators and design sensitivity formulations for transient non-linear coupled problems with applications to elastoplasticity. International Journal for Numerical Methods in Engineering 37, 14: 2471–2499. doi:10.1002/nme.1620371408.

[45] Miehe, C., Bayreuther, C. G., 2007. On multiscale FE analyses of heterogeneous structures: from homogenization to multigrid solvers. International Journal for Numerical Methods in Engineering 71, 10: 1135–1180. doi:10.1002/nme.1972.

[46] Mirzendehdel, A. M., Suresh, K., 2016. Support structure constrained topology optimization for additive manufacturing. Computer-Aided Design 81: 1–13. doi: 10.1016/j.cad.2016.08.006.

[47] Niekamp, R., Markovič, D., Ibrahimbegović, A., Matthies, H. G., Taylor, R. L., 2009. Multi-scale modelling of heterogeneous structures with inelastic constitutive behavior. Engineering Computations 26, 1/2: 6–28. doi:10.1108/02644400910924780.

[48] Novak, N., Vesenjak, M., Ren, Z., 2016. Auxetic cellular materials - a review. Strojniški vestnik – Journal of Mechanical Engineering 62, 9: 485–493. doi: 10.5545/sv-jme.2016.3656.

[49] Novotny, A., Feijóo, R., Taroco, E., Padra, C., 2007. Topological sensitivity analysis for three-dimensional linear elasticity problem. Computer Methods in Applied Mechanics and Engineering 196, 41-44: 4354–4364. doi:10.1016/j.cma.2007.05.006.

[50] Oropallo, W., Piegl, L. A., 2015. Ten challenges in 3D printing. Engineering with Computers 32, 1: 135–148. doi:10.1007/s00366-015-0407-0.

[51] Oumaziz, P., Gosselet, P., Boucard, P.-A., Abbas, M., 2007. A parallel noninvasive multiscale strategy for a mixed domain decomposition method with frictional contact. International Journal for Numerical Methods in Engineering 196, 8: 893–912. doi: 10.1002/nme.5830.

[52] Prall, D., Lakes, R., 1997. Properties of a chiral honeycomb with a Poissons ratio of -1. International Journal of Mechanical Sciences 39, 3: 305–314. doi:10.1016/s0020-7403(96)00025-2.

[53] Shamonina, E., Solymar, L., 2007. Metamaterials: How the subject started. Metamaterials 1, 1: 12–18. doi:10.1016/j.metmat.2007.02.001.

[54] Simo, J. C., Hughes, T. J. R., 1998. Computational Inelasticity. Springer-Verlag, New York. doi:10.1007/b98904.

[55] Šolinc, U., Korelc, J., 2015. A simple way to improved formulation of FE2 analysis. Computational Mechanics 56, 5: 905–915. doi:10.1007/s00466-015-1208-4.

[56] Temizer, İ., Wriggers, P., 2008. On the computation of the macroscopic tangent for multiscale volumetric homogenization problems. Computer Methods in Applied Mechanics and Engineering 198, 3-4: 495–510. doi:10.1016/j.cma.2008.08.018.

[57] Terada, K., Hori, M., Kyoya, T., Kikuchi, N., 2000. Simulation of multi-scale convergence in computational homogenization approaches. International Journal of Solids and Structures 37, 16: 2285–2311. doi:10.1016/s0020-7683(98)00341-2.

[58] Terada, K., Kikuchi, N., 2001. A class of general algorithms for multi-scale analyses of heterogeneous media. Computer Methods in Applied Mechanics and Engineering 190, 40-41: 5427–5464. doi:10.1016/s0045-7825(01)00179-7.

[59] Tognevi, A., Guerich, M., Yvonnet, J., 2016. A multi-scale modeling method for heterogeneous structures without scale separation using a filter-based homogenization scheme. International Journal for Numerical Methods in Engineering 108, 1: 3–25. doi:10.1002/nme.5200.

[60] Tong, X. C., 2018. Functional metamaterials and metadevices. Springer International Publishing. doi:10.1007/978-3-319-66044-8.

[61] Vayre, B., Vignat, F., Villeneuve, F., 2012. Metallic additive manufacturing: state-of-the-art review and prospects. Mechanics & Industry 13, 2: 89–96. doi:10.1051/meca/2012003.

[62] Wang, X.-T., Li, X.-W., Ma, L., 2016. Interlocking assembled 3D auxetic cellular structures. Materials & Design 99: 467–476. doi:10.1016/j.matdes.2016.03.088.

[63] Wriggers, P., 2008. Nonlinear finite element methods. Springer, Berlin. doi:10.1007/978-3-540-71001-1.

[64] Wu, Z., Xia, L., Wang, S., Shi, T., 2019. Topology optimization of hierarchical lattice structures with substructuring. Computer Methods in Applied Mechanics and Engineering 345: 602–617. doi:10.1016/j.cma.2018.11.003.

[65] Xia, L., Breitkopf, P., 2016. Recent advances on topology optimization of multiscale nonlinear structures. Archives of Computational Methods in Engineering 24, 2: 227–249. doi:10.1007/s11831-016-9170-7.

[66] Yang, K. K., Zhu, J. H., Wang, C., Jia, D. S., Song, L. L., Zhang, W. H., 2018. Experimental validation of 3D printed material behaviors and their influence on the structural topology design. Computational Mechanics 61, 5: 581–598. doi:10.1007/s00466-018-1537-1.

[67] Yang, W., Li, Z.-M., Shi, W., Xie, B.-H., Yang, M.-B., 2004. Review on auxetic materials. Journal of Materials Science 39, 10: 3269–3279. doi:10.1023/b:jmsc.0000026928.93231.e0.

[68] Yu, X., Zhou, J., Liang, H., Jiang, Z., Wu, L., 2018. Mechanical metamaterials associated with stiffness, rigidity and compressibility: A brief review. Progress in Materials Science 94: 114–173. doi:10.1016/j.pmatsci.2017.12.003.

[69] Zadpoor, A. A., 2016. Mechanical meta-materials. Materials Horizons 3, 5: 371–381. doi:10.1039/c6mh00065g.

[70] Zohdi, T. I., 2017. Homogenization methods and multiscale modeling. John Wiley & Sons, Ltd, Chichester. doi:10.1002/9781119176817.ecm2034.

[71] Zohdi, T. I., Wriggers, P., 1999. A domain decomposition method for bodies with heterogeneous microstructure basedon material regularization. International Journal of Solids and Structures 36, 17: 2507–2525. doi:10.1016/s0020-7683(98)00124-3.

[72] Zohdi, T. I., Wriggers, P., 2005. An introduction to computational micromechanics. Springer Berlin Heidelberg. doi:10.1007/978-3-540-32360-0.

[73] Zupan, N., Korelc, J., 2020. Sensitivity analysis based multi-scale methods of coupled path-dependent problems. Computational Mechanics 65, 1: 229–248. doi:10.1007/s00466-019-01762-8.

APPENDICES

APPENDIX A

A.1 Short description of AceGen syntax

The syntax of the *AceGen* script language is the same as that of the Mathematica script language with some additional functions. Basic *AceGen* functions:

- `SMSInitialize` function initializes the *AceGen* system and sets the target finite element environment to be *AceFEM* .
- `SMSTemplate` function sets the basic characteristics of the finite element that is going to be generated.
- `SMSStandardModule` function defines the input and output parameters of the *AceGen* user subroutines.
- `SMSReal` and `SMSInteger` functions connect the real and integer type input parameters of the subroutine with the Mathematica symbols. The input/output parameters of the subroutine are recognized in the input by the "$$" string at the end of the name.
- `SMSIO` updates all the data associated with the given dataType, for example `"Nodal DOFs"`.
- $\models$ is a special AceGen operator that is used during the description of the problem instead of the standard Mathematica assignment operator =. It performs simultaneous optimization of expressions and creation of the new auxiliary variables.
- `SMSD` function performs an automatic differentiation, different options were presented in Section 2.2
- `SMSExport` function assigns the results of the derivation to the output parameter of the subroutines.
- `SMSWrite` function at the end writes the generated formulae to the file in a prescribed language format.

A.2 *AceGen* input of hyper-elastic micro finite element

Presented finite element supports first and second order sensitivity analysis and includes tasks needed in multi-scale analysis.

Step 1: *AceGen* input segment for Initialization

```
SMSInitialize["ExamplesSensitivity2D", "Environment" → "AceFEM"];
SMSTemplate["SMSTopology" → "Q1", "SMSSymmetricTangent" → True,
  "SMSDomainDataNames" -> {"E -elastic modulus", "ν -poisson ratio", "t -thickness"},
  "SMSEBCSensitivity" -> True, "SMSDefaultData" -> {21 000, 0.3, 1},
  "SMSCharSwitch" →
   {"Volume and sensitivity", "Integrated stress and sensitivity (P, plane strain)",
    "Integrated strain energy and derivatives (plane strain)", "Reset sensitivity data"}];
```

Step 2: *AceGen* input segment for problem dependent definitions

```
ElementDefinitions[] := Block[{},
  Ξ = {ξ, η, ζ} ⊨ SMSIO["Integration point", Ig];
  {XIO, uIO} ⊨ SMSIO["All coordinates and DOFs"];
  Nh ⊨ 1 / 4 {(1 - ξ) (1 - η), (1 + ξ) (1 - η), (1 + ξ) (1 + η), (1 - ξ) (1 + η)};
  SMSFreeze[X, Append[Nh.XIO, ζ]];
  Je ⊨ SMSD[X, Ξ]; Jed ⊨ Det[Je];
  u ⊨ Append[Nh.uIO, 0];
  H ⊨ SMSD[u, X, "Dependency" → {Ξ, X, SMSInverse[Je]}];
  SMSFreeze[F, IdentityMatrix[3] + H, "Ignore" → PossibleZeroQ];
  JF ⊨ Det[F]; Ct ⊨ Transpose[F].F;
  {Em, ν, tζ} ⊨ SMSIO["All domain data"];
  {λ, μ} ⊨ SMSHookeToLame[Em, ν];
  W ⊨ 1 / 2 λ (JF - 1) ^ 2 + μ (1 / 2 (Tr[Ct] - 3) - Log[JF]);
  wgp ⊨ SMSIO["Integration weight", Ig];
  pe = Flatten[uIO];
  fGauss ⊨ Jed tζ;]
```

In this part the input parameters of the standard user subroutines are defined. The shape functions are defined, displacement gradient, deformation gradient and right Cauchy-Green strain tensor are calculated to formulate the potential W for hyper-elastic material model as presented in Section 2.3.3.

Step 3: *AceGen* input segment for `"Tangent and Residual"` user subroutine

```
SMSStandardModule["Tangent and residual"];
SMSDo[Ig, 1, SMSIO["No. integration points"]];
ElementDefinitions[];
SMSDo[
  Rg = fGauss SMSD[W, pe, i];
  SMSExport[wgp Rg, p$$[i], "AddIn" → True];
  SMSDo[
   Kg = SMSD[Rg, pe, j];
   SMSExport[wgp Kg, s$$[i, j], "AddIn" → True];
   , {j, i, SMSNoDOFGlobal}];
  , {i, 1, SMSNoDOFGlobal}];
SMSEndDo[];
```

Subroutine `"Tangent and residual"` is started and `ElementDefinitions` module is evaluated for each GAUSS point. The residual vector (87) and the tangent matrix (107) are computed. The results of the derivation are assigned to the output parameters `s$$` and `p$$` of the `"Tangent and residual"` standard user subroutine by the `SMSExport` function. Here the symmetry of the tangent matrix is accounted for by exporting only the upper triangle matrix.

Step 4: *AceGen* input segment for `"Sensitivity pseudo-load"` user subroutine for boundary condition sensitivity analysis

```
SMSStandardModule["Sensitivity pseudo-load"];
SMSDo[Ig, 1, SMSIO["No. integration points"]];
ElementDefinitions[];
Rg = Jedtζ SMSD[W, pe];
SensIndexStart = SMSIO["SensIndexStart"];
SMSDo[SensIndexi, SensIndexStart, SMSIO["SensIndexEnd"]];
ϕi = SMSFictive[];
"add first order essential boundary conditions velocity field ∂pe/∂ϕi to nodal DOFs";
SMSDefineDerivative[SMSIO["Nodal DOFs"], ϕi, SMSIO["Sensitivity DOFs", SensIndexi]];
Rtg = SMSD[Rg, ϕi, "Method" → "Backward"];
sensPosition = SMSInteger[SensIndexi - SensIndexStart + 1];
SMSExport[wgp Rtg, Function[{dof}, s$$[sensPosition, dof]], "AddIn" → True];
SMSEndDo[];(*SensIndexi*)
SMSEndDo[];(*Integration points*).
```

The `"Sensitivity pseudo-load"` subroutine starts with the definition of an arbitrary

sensitivity parameter ϕ_i as fictive variable with `SMSFictive` that initially does not depend on anything and thus has all the derivatives set to zero. Sensitivity analysis with respect to essential boundary conditions is shown. First order essential boundary condition velocity field is added to nodal DOFs. `SMSDefineDerivative` defines the derivative of auxiliary variable nodal DOFs, with respect to auxiliary variable ϕ_i, to be equal to sensitivities of DOFs. First order sensitivity pseudo-load vector ${}^{I}\tilde{\mathbf{R}}$ (119) is calculated with `SMSD[Rg,`ϕ_i`,"Method"→"Backward"]`, automatic differentiation of residual with respect to sensitivity parameter, using backward mode of automatic differentiation. Pseudo-load vector is then exported into external variable `s$$`, where `"AddIn"→True` adds the value of the expression to the current value of the external variable `s$$` for all GAUSS point contributions.

Step 5: *AceGen* input segment for for "`Sensitivity 2nd order`" user subroutine

```
SMSStandardModule["Sensitivity 2nd order"];
noSensParameters = SMSIO["NoSensParameters"];
SMSDo[Ig, 1, SMSIO["No. integration points"]];
ElementDefinitions[];
Rg = Jedt ζ SMSD[W, pe];
SMSDo[SensIndexi, SMSIO["SensRowStart"], SMSIO["SensRowEnd"]];
ϕi = SMSFictive[];
"add first order sensitivities ∂pe/∂ϕi to nodal DOFs";
SMSDefineDerivative[SMSIO["Nodal DOFs"], ϕi, SMSIO["Sensitivity DOFs", SensIndexi]];
SMSDo[SensIndexj, SensIndexi, noSensParameters];
SensDerivativeij =
  SMSInteger[SMSSensDerivativePosition[noSensParameters, SensIndexi, SensIndexj]];
ϕj = SMSFictive[];
"add first order sensitivities ∂pe/∂ϕj to nodal DOFs";
SMSDefineDerivative[SMSIO["Nodal DOFs"], ϕj, SMSIO["Sensitivity DOFs", SensIndexj]];
"add second order essential boundary conditions
   velocity field ∂(∂pe/∂ϕi)/∂ϕj to first order sensitivities";
SMSDefineDerivative[SMSIO["Sensitivity DOFs", SensIndexi], ϕj,
  SMSIO["Sensitivity DOFs", SensDerivativeij]];
Rtg = SMSD[SMSD[Rg, ϕi, "Method" → "Forward"], ϕj, "Method" → "Backward"];
sensPosition = SensDerivativeij - SMSIO["SensIndexStart"] + 1;
SMSExport[wgp Rtg, Function[{dof}, s$$[sensPosition, dof]], "AddIn" → True];
SMSEndDo[];(*SensIndexj*)
SMSEndDo[];(*SensIndexi*)
SMSEndDo[];(*Integration points*).
```

The "`Sensitivity 2nd order`" subroutine is started. It returns the second order sensitivity pseudo-load vector ${}^{IJ}\tilde{\mathbf{R}}$ (128) for the current subset of sensitivity parameters. It is

shown here for essential boundary conditions. Calculation is done by nested loops, outer loop over ϕ_i and inner loop over ϕ_j. First order sensitivities over ϕ_i and ϕ_j are already calculated in first order sensitivity analysis and are added to nodal DOFs. Second order essential boundary conditions velocity field is defined as a function of first order sensitivities and added to nodal DOFs. Second-order Sensitivity Pseudo-load vector is defined with `SMSD[SMSD[Rg,`ϕ_i`,"Method"→"Forward"],`ϕ_j`,"Method"→"Backward"]]`, automatic differentiation of residual is preformed for two times with respect to sensitivity parameters, first time using forward mode and second time backward mode of automatic differentiation. It is then exported into the same external variable `s$$`, as the first-order pseudo-load vector. Note that first order sensitivity has already been calculated at this point.

Step 6: *AceGen* input segment for `"Tasks"` user subroutine for additional multi-scale related tasks

User subroutine `"Tasks"` performs various user-defined tasks that require assembly of results over the whole or part of the mesh.

Step 6.1: *AceGen* input segment for `"Reset sensitivity data"` task

```
"Reset sensitivity data"
SMSStandardModule["Tasks"];
noSensDerivatives = SMSIO["NoSensDerivatives"];
SMSExport[{1, 0, 0, 0, 0}, TasksData$$];
SMSDo[
  SMSExport[Table[0, {i, SMSNoNodes}, {j, SMSDOFGlobal[[i]]}],
   Table[nd$$[i, "st", SensIndex, j], {i, SMSNoNodes}, {j, SMSDOFGlobal[[i]]}]];
  SMSExport[Table[0, {i, SMSNoNodes}, {j, SMSDOFGlobal[[i]]}], Table[nd$$[i, "sp", SensIndex, j],
    {i, SMSNoNodes}, {j, SMSDOFGlobal[[i]]}]];, {SensIndex, 1, noSensDerivatives}];
SMSEndDo[];
```

Essential boundary condition sensitivities used for connection between scales in multi-scale algorithm are relative sensitivities. If we look back at Fig. 4 in Section 3.1, sensitivities at micro level in each micro load step $\lambda_{mn}, \lambda_{mn+1}, \lambda_{ms+n_m}$, are relative values to the value in step λ_{ms}. In λ_{ms}, at the end of last converged macro load step λ_{Mk}, values are equal to zero. Meaning that after each macro load step λ_{Mk} sensitivities need to be deleted before starting next step λ_{Mk+1} , to fulfill (95). For this purpose additional task `"Reset sensitivity data"` is defined. It sets all current sensitivities `"st"` and sensitivities in previous step `"sp"` for all DOFs of all nodes to value zero.

Step 6.2: *AceGen* input segment for `"Integrated strain energy"` task

```
"Integrated strain energy and derivatives (plane strain)"
SMSStandardModule["Tasks"];
noSensParameters ⊨ SMSIO["NoSensParameters"];
noSensDerivatives ⊨ SMSIO["NoSensDerivatives"];
NoIp ⊨ SMSIO["No. integration points"];
SMSExport[{1, 0, 0, 0, 1 + noSensDerivatives}, TasksData$$];
SMSDo[Ig, 1, NoIp];
ElementDefinitions[];
"primal anaysis quantities";
ΔV ⊨ Jed t ζ;
Wg ⊨ ΔV W;
SMSExport[wgp Wg, RealOutput$$[1], "AddIn" → True];

SMSDo[SensIndexi, 1, noSensParameters];
ϕi ⊢ SMSFictive[];
SMSDefineDerivative[SMSIO["Nodal DOFs"], ϕi, SMSIO["Sensitivity DOFs", SensIndexi]];
"first order sensitivity anaysis quantities";
DWDϕi ⊨ SMSD[Wg, ϕi];
SMSExport[wgp DWDϕi, RealOutput$$[1 + SensIndexi], "AddIn" → True];

SMSDo[SensIndexj, SensIndexi, SMSIf[SMSIO["NoSecondOrderDerivatives"] == 0, 0, noSensParameters]];
(*position of the ∂^/∂ϕi∂ϕj derivative within the list of all derivatives*)
SensDerivativeij ⊨
  SMSInteger[SMSSensDerivativePosition[noSensParameters, SensIndexi, SensIndexj]];
ϕj ⊢ SMSFictive[];
SMSDefineDerivative[SMSIO["Nodal DOFs"], ϕj, SMSIO["Sensitivity DOFs", SensIndexj]];
SMSDefineDerivative[SMSIO["Sensitivity DOFs", SensIndexi],
  ϕj, SMSIO["Sensitivity DOFs", SensDerivativeij]];
"second order sensitivity anaysis quantities";
DWDϕiϕj ⊨ wgp SMSD[DWDϕi, ϕj];
SMSExport[DWDϕiϕj, RealOutput$$[1 + SensDerivativeij], "AddIn" → True];
SMSEndDo[];(*sensitivity j*)
SMSEndDo[];(*sensitivity i*)
SMSEndDo[];(*integration points*).
```

For the connection between scales in case of MIEL, user-defined task `"Integrated strain energy and derivatives"` needs to be defined in micro element. Firstly, primal analysis values are calculated, in case of MIEL integral of strain energy W, contributions of all elements over the domain of the problem are assembled with `"AddIn"→True`. Secondly, first order sensitivity analysis quantities are calculated, sensitivities of strain energy W with respect to the components of nodal degrees of freedom $\mathbf{p}_M$. `SMSDefineDerivative` defines differentiation exceptions. Derivatives of W over $\mathbf{p}_M$ represent the contribution of the micro element to the macro element residual (141). Thirdly, second order sen-

sitivity analysis quantities are calculated, second order sensitivities of strain energy W with respect to the components of nodal degrees of freedom of macro element $\mathbf{p}_M$. They represent the contribution of the micro element to the condensed tangent matrix (142) for MIEL multi-scale simulations. If second order directional derivatives are symmetric then only the upper triangular matrix is returned. All quantities are sequentially exported into `TasksData$$`.

Step 6.3: *AceGen* input segment for `"Integrated stress and sensitivity"` task

```
"Integrated stress and sensitivity (P, plane strain)"
SMSStandardModule["Tasks"];
noSensParameters = SMSIO["NoSensParameters"];
NoIp = SMSIO["No. integration points"];
StessComponents = {{1, 1}, {1, 2}, {2, 1}, {2, 2}, {3, 3}};
NoStressComponents = Length[StessComponents];
SMSExport[{1, 0, 0, 0, NoStressComponents + NoStressComponents noSensParameters }, TasksData$$];
SMSDo[Ig, 1, NoIp];
ElementDefinitions[];
"primal anaysis quantities";
ΔV = Jed tζ;
P = SMSD[W, F, "Ignore" -> NumberQ];
PI = ΔVExtract[P, StessComponents];
SMSExport[ wgp PI, Table[RealOutput$$[i], {i, NoStressComponents}], "AddIn" → True];

SMSDo[SensIndexi, 1, noSensParameters];
ϕi = SMSFictive[];
SMSDefineDerivative[SMSIO["Nodal DOFs"], ϕi, SMSIO["Sensitivity DOFs", SensIndexi]];
"first order sensitivity anaysis quantities";
DPDϕi = SMSD[PI, ϕi];
SMSExport[wgp DPDϕi,
  Table[RealOutput$$[NoStressComponents + (SensIndexi - 1) NoStressComponents + i],
   {i, NoStressComponents}], "AddIn" → True];
SMSEndDo[];(*sensitivity i*)
SMSEndDo[];(*integration points*) .
```

For the connection between scales in case of FE^2, an user-defined task is needed in micro element used for the discretization of RVE. For the finite strains material model, this would be `"Integrated stress and sensitivity"` task. In the loop over integration points module `ElementDefinitions` is evaluated. Firstly, primal analysis values are calculated, the first Piola-Kirchoff stress tensor is calculated as a derivative of strain energy W over deformation gradient $\boldsymbol{F}$ and its components are extracted. `"AddIn"→True` denotes that contributions of all elements in RVE are summarized. Secondly, first order sensitivity

analysis quantities are calculated, sensitivities of Piola-Kirchoff stress tensor components with respect to the components of deformation gradient $\boldsymbol{F}_M$ (160). Primal and sensitivity quantities are exported into `TasksData$$` and are used in the calculation of the macro residual and tangent matrix.

A.3 *AceGen* input of MIEL macro finite element

Next to the micro finite element that supports first and second order sensitivity analysis and contains additional tasks for the calculation of integrated strain energy and derivatives and reset of sensitivity data we also need a macro finite element, see Section4.1. MIEL macro element is a specific finite element created for the MIEL simulations. The macro element calculates macro tangent and residual. User defined subroutine `"Tasks"` is required. The function of this element is essentially the exchange of data between micro and macro level and calculation of macro level response.

Step 1: *AceGen* input segment for Initialization of MIEL macro element

```
SMSInitialize["MacroElementMIEL", "Environment" → "AceFEM"];
SMSTemplate["SMSTopology" → "Q1", "SMSSymmetricTangent" → True,
  "SMSCharSwitch" → {"MIEL", "Nodal displacements (2D)",
    "Integrated strain energy and derivatives (plane strain)"}];
SMSNoElementData = 1 + SMSNoDOFGlobal + SMSNoDOFGlobal (SMSNoDOFGlobal + 1) / 2;
```

Input for quadrilateral element `"Q1"` is shown, but it can be easily modified for other topologies. Names of tasks are defined in `"SMSCharSwitch"`. The first two are macro elements tasks and the third one is the name of the micro element task presented.

Step 2: *AceGen* input segment for `"Tangent and residual"` subroutine of MIEL macro element

```
SMSStandardModule["Tangent and residual"];
row = SMSInteger[SMSNoDOFGlobal + 2];
SMSDo[
  SMSExport[SMSReal[ed$$["Data", 1 + i]], p$$[i]];
  SMSDo[
   SMSExport[SMSReal[ed$$["Data", row + j - i]], s$$[i, j]];
   , {j, i, SMSNoDOFGlobal}];
  row = row + SMSNoDOFGlobal - i + 1;
  , {i, 1, SMSNoDOFGlobal, 1, row}];
```

This subroutine is required for calculation of macro level response. Macro residual and tangent matrix are not calculated within an element but are directly imported from `"Data"` field and exported to the appropriate output fields. Values were written into `"Data"` after solving the corresponding MIEL micro problem.

Step 3: *AceGen* input segment for `"Task"` subroutine of MIEL macro element

```
SMSStandardModule["Tasks"];
task = SMSInteger[Task$$];
SMSIf[task < 0
  , SMSSwitch[task
   , -1, SMSExport[{1, 0, 0, 5, 0}, TasksData$$];
   , -2, SMSExport[{1, 0, 0, 0, SMSNoDOFGlobal}, TasksData$$];];
  SMSReturn[];];
SMSSwitch[task
  , 1, SMSExport[{1, 2, SMSNoDOFGlobal, 3, SMSNoElementData}, IntegerOutput$$];
  , 2, SMSExport[SMSIO["All DOFs"] // Flatten, Table[RealOutput$$[i], {i, SMSNoDOFGlobal}]];];
```

The first task in macro element with name `"MIEL"` returns a list that contains:

- number of micro problems associated with the element which is in the case of MIEL formulation 1, as 1 macro element represents exactly 1 micro problem;
- index of the character switch constant of the macro element that defines the name of the macro task that returns multi-scale related macro data, `"Nodal displacements (2D)"`;

- length of multi-scale related macro data, that is equal to `SMSNoDOFGlobal`, number of all macro elements DOFs;
- index of the character switch constant of the macro element that defines the name of the micro problem task that returns multi-scale related micro data, `"Integrated strain energy and derivatives (plane strain)"`;
- length of multi-scale related micro data, which is equal to the length of `"Data"` field, which contains integrated strain energy and derivatives.

Second macro element task `"Nodal displacements (2D)"` exports values of all macro element nodal DOFs, which are sensitivity parameters of the micro problem.

A.4 *AceGen* input of FE^2 macro finite element

FE^2 macro element is a specific finite element that calculates macro tangent and residual and contains additional tasks for multi-scale analysis, see Section 4.2.

Step 1: *AceGen* input segment for Initialization of FE^2 macro element

```
NoStressComponents = 5; NoKinComponents = 4;
lgd = (NoStressComponents + NoStressComponents NoKinComponents);
Led = lgd es$$["id", "NoIntPoints"];
SMSInitialize["ExamplesQ1PFMacroSymm", "Environment" → "AceFEM"];
SMSTemplate["SMSTopology" → "Q1"
  , "SMSSymmetricTangent" → True, "SMSDomainDataNames" -> {"t -thickness"}
  , "SMSDefaultData" -> {1}, "SMSNoElementData" → Led
  , "SMSCharSwitch" → {"FE^2", "Material points", "Deformation gradient (plane strain)",
    "Integrated stress and sensitivity (P, plane strain)"}];
```

In `"SMSCharSwitch"`, names for tasks are defined. The first three are macro elements tasks and the fourth one is the name of the micro element task.

Step 2: *AceGen* input segment for Element definitions of FE^2 macro element

```
ElementDefinitions[] := (
  Ξ = {ξ, η, ζ} ⊨ SMSIO["Integration point", Ig];
  wgp ⊨ SMSIO["Integration weight", Ig];
  {XIO, uIO} ⊨ SMSIO["All coordinates and DOFs"];
  Nh ⊨ 1 / 4 {(1 - ξ) (1 - η), (1 + ξ) (1 - η), (1 + ξ) (1 + η), (1 - ξ) (1 + η)};
  SMSFreeze[X, Append[Nh.XIO, ζ]];
  Je ⊨ SMSD[X, Ξ]; Jed ⊨ Det[Je];
  u ⊨ Append[Nh.uIO, 0];
  H ⊨ SMSD[u, X, "Dependency" → {Ξ, X, SMSInverse[Je]}];
  SMSFreeze[F, IdentityMatrix[3] + H, "Ignore" → PossibleZeroQ];
  FI = Extract[F, {{1, 1}, {1, 2}, {2, 1}, {2, 2}}];
  Igd = SMSInteger[(Ig - 1) lgd];
  DPDF ⊨ SMSReal[Table[ed$$["Data", Igd + NoStressComponents + (j - 1) NoStressComponents + i],
      {i, NoStressComponents}, {j, NoKinComponents}]];
  PI ⊨ SMSReal[Table[ed$$["Data", Igd + i], {i, NoStressComponents}], "Dependency" → {FI, DPDF}];
  P ⊨ {{PI[[1]], PI[[2]], 0}, {PI[[3]], PI[[4]], 0}, {0, 0, PI[[5]]}};
  {tζ} ⊨ SMSIO["All domain data"];
  W ⊨ Tr[P.Transpose[F]];
  pe = Flatten[uIO];
  fGauss ⊨ Jed tζ;)
```

In `ElementDefinitions`, deformation gradient $\boldsymbol{F}_M$ is calculated and component of Piola-Kirchoff stress tensor $\boldsymbol{P}_M$ and its derivatives are read from `"Data"` field.

Step 3: *AceGen* input segment for `"Tangent and residual"` subroutine of FE^2 macro element

```
SMSStandardModule["Tangent and residual"];
SMSDo[Ig, 1, SMSIO["No. integration points"]];
ElementDefinitions[];
SMSDo[
  Rg ⊨ wgp fGauss SMSD[W, pe, i, "Constant" -> PI];
  SMSExport[Rg, p$$[i], "AddIn" → True];
  SMSDo[
   Kg ⊨ SMSD[Rg, pe, j];
   SMSExport[Kg, s$$[i, j], "AddIn" → True];
   , {j, i, SMSNoDOFGlobal}];
  , {i, 1, SMSNoDOFGlobal}];
SMSEndDo[];
```

After solution of specific FE^2 micro problem, (representative volume element RVE), results are written into `"Data"` field. `"Dependency"` of $\boldsymbol{P}_M$ on $\boldsymbol{F}_M$ is set. For calculation of

macro level response `SMSStandardModule["Tangent and residual"]` is used. Macro residual is calculated as a derivative of strain energy W over nodal unknowns $\mathbf{p}_e$ and differentiation is done under the assumption that W does not depend on $\boldsymbol{P}_M$. Tangent matrix is calculated as a derivative of residual over nodal unknowns $\mathbf{p}_e$. Residual and tangent matrix are exported to an appropriate field.

Step 4: *AceGen* input segment for `"Task"` subroutine of FE^2 macro element

```
SMSStandardModule["Tasks"];
ng = SMSIO["No. integration points"];
task = SMSInteger[Task$$];
SMSIf[task < 0
  , SMSSwitch[task
   , -1, SMSExport[{1, 0, 0, 5, 0}, TasksData$$];
   , -2, SMSExport[{1, 0, 0, 0, SMSNoDimensions * ng}, TasksData$$];
   , -3, SMSExport[{1, 0, 0, 0, NoKinComponents * ng}, TasksData$$];];
  SMSReturn[];];
SMSIf[task == 1,
  SMSExport[{ng, 3, NoKinComponents, 4, lgd}, IntegerOutput$$];];
SMSDo[
  ElementDefinitions[];
  SMSSwitch[task
   , 2,
   SMSExport[X[[{1, 2}]], Table[RealOutput$$[i + SMSNoDimensions * (Ig - 1)], {i, SMSNoDimensions}]];
   , 3, SMSExport[FI, Table[RealOutput$$[i + NoKinComponents * (Ig - 1)], {i, NoKinComponents}]];];
  , {Ig, 1, ng}];
```

The code for tasks is shown. The first task in macro element with name `"FE2"` returns a list that contains:

- number of micro problems associated with the element. In the case of FE^2 method the number of integration points of the macro element;
- index of the character switch constant of a macro element that defines the name of the macro task that returns multi-scale related macro data, `"Deformation gradient (plane strain)"`;
- length of multi-scale related macro data that is equal to `NoKinComponents` which is, number of components of deformation gradient;

- index of the character switch constant of a macro element that defines the name of the micro problem task which returns multi-scale related micro data `"Integrated stress and sensitivity (P, plane strain)"`;
- length of multi-scale related micro data, which is equal to the length of `"Data"` field. It contains integrated stress and derivatives for each macro integration point.

Second macro element task `"Material points"` returns position vector of all integration points. The third macro element task `"Deformation gradient (plane strain)"` exports components of macro deformation gradient, which are sensitivity parameters of the micro problem.

A.5 *AceGen* input for macro level sensitivity analysis needed for optimization

Sensitivity analysis at the macro level is needed for multi-scale optimization, see Section 6. It is inserted after **Step 2:** `"Tangent and Residual"` user subroutine.

Step: *AceGen* input segment for `"Sensitivity pseudo-load"` subroutine of MIEL macro element

```
SMSStandardModule["Sensitivity pseudo-load"];
SensIndexStart ⊨ SMSIO["SensIndexStart"];
SMSDo[SensIndexi, SensIndexStart, SMSIO["SensIndexEnd"]];
sensPosition ⊨ SMSInteger[SensIndexi - SensIndexStart + 1];
SMSDo[SMSExport[ SMSReal[ed$$["Data", 1 + SMSNoDOFGlobal + SMSNoDOFGlobal (SMSNoDOFGlobal + 1) / 2 + j]],
    s$$[sensPosition, j]], {j, 1, SMSNoDOFGlobal}];
SMSEndDo[];
```

In the case of MIEL, sensitivity pseudo-load vector is not calculated at the macro level but is just read from `"Data"` field, where contributions of all micro problem elements were assembled beforehand.

Step: *AceGen* input segment for "`Sensitivity pseudo-load`" subroutine of FE^2 macro element

```
SMSStandardModule["Sensitivity pseudo-load"];
SMSDo[Ig, 1, SMSIO["No. integration points"]];
ElementDefinitions[];
Rg ⊨ fGauss wgp SMSD[W, pe, "Constant" → PI];
SensIndexStart ⊨ SMSIO["SensIndexStart"];
SMSDo[SensIndexi, SensIndexStart, SMSIO["SensIndexEnd"]];
ϕi ⊢ SMSFictive[];
DPDϕ ⊢ Table[SMSReal[ed$$["Data", Igd + NoStressComponents +
      NoStressComponents (NoKinComponents + SensIndexi - 1) + i]], {i, 1, NoStressComponents}];
SMSDefineDerivative[PI, ϕi, DPDϕ];
Rtg ⊨ SMSD[ Rg, ϕi, "Method" → "Backward"];
sensPosition ⊨ SMSInteger[SensIndexi - SensIndexStart + 1];
SMSExport[Rtg, Function[{dof}, s$$[sensPosition, dof]], "AddIn" → True];
SMSEndDo[];(*sensitivity i*)
SMSEndDo[];(*integration points*).
```

Macro residual is calculated as a derivative of strain energy W over nodal unknowns $\mathbf{p}_e$ and differentiation is done under the assumption that W does not depend on Piola-Kirchoff stress tensor $\boldsymbol{P}_M$. Derivatives of $\boldsymbol{P}_M$ with respect to macro sensitivity parameters $\boldsymbol{\phi}_M$ are read from "`Data`" field and are used to define differentiation exception. The sensitivity pseudo-load vector is calculated as a derivative of residual over sensitivity parameters, considering AD exceptions and all element contributions are assembled during export to an external variable `s$$`.

A.6 *AceGen* input for Response functional task needed for plastic-work optimization

During solution of micro problems also micro element "`Response functional`" task is evaluated, see Section 6.3.2. Contributions of all elements in RVE are assembled and averaged values of W_p is calculated and stored in "`Data`" for all RVEs.

Step: *AceGen* input segment for `"Response functional"` task subroutine of micro element

```
"Response functional"
SMSStandardModule[FEMModule = "Tasks"];
noSensDerivatives ⊨ SMSIO["NoSensDerivatives"];
noSensParameters ⊨ SMSIO["NoSensParameters"];
NoIp ⊨ SMSIO["No. integration points"];

SMSDo[Ig, 1, NoIp];
Discretization[];
ConstitutiveEquations["Initialize", _, _];
ConstitutiveEquations["Post-processing", hgIO, hgnIO];
AppendTo[pseudoWConstants, hgIO];
"primal anaysis quantities";
Wpg ⊨ fGauss Wp;
SMSExport[wgp Wpg, RealOutput$$[1], "AddIn" → True];

SMSDo[SensIndexi, 1, noSensParameters];
ϕi ⊢ SMSFictive[];
"shape velocity field";
SMSDefineDerivative[Take[SMSIO["Nodal coordinates"], noTopologicalNodes],
  ϕi, SMSIO["Shape velocity field", SensIndexi, noTopologicalNodes]];
"sensitivity of global DOF";
SMSDefineDerivative[SMSIO["Nodal DOFs"], ϕi, SMSIO["Sensitivity DOFs", SensIndexi]];
"sensitivity of global DOF at time n";
SMSDefineDerivative[SMSIO["Nodal DOFs n"], ϕi, SMSIO["Sensitivity DOFs n", SensIndexi]];
sensPosi ⊢ SMSInteger[(SensIndexi - 1) NoIp + Ig];
"DhgDϕiIO sensitivity of hg";
DhgDϕiIO = SMSIO["Time dependent", "hgsens"[_] → sensPosi];
SMSDefineDerivative[hgIO, ϕi, DhgDϕiIO];
"DhgnDϕiIO - sensitivity of hgn";
DhgnDϕiIO = SMSIO["Time dependent n", "hgsens"[_] → sensPosi];
SMSDefineDerivative[hgnIO, ϕi, DhgnDϕiIO];
"first order sensitivity anaysis quantities";
DWpDϕi ⊨ SMSD[Wpg, ϕi];
SMSExport[wgp DWpDϕi, RealOutput$$[1 + SensIndexi], "AddIn" → True];
SMSEndDo[];(*sensitivity*)
SMSEndDo[];(*integration points*).
```

Step: *AceGen* input segment for `"Response functional"` task subroutine of FE^2 macro element

```
"Response functional"
SMSStandardModule["Tasks"];
NoStressComponents = 5; NoKinComponents = 4;
NoIp ⊨ SMSIO["No. integration points"];
noSensDerivatives ⊨ SMSIO["NoSensDerivatives"];
noSensParameters ⊨ SMSIO["NoSensParameters"];
lgmicro ⊨ NoKinComponents (1 + noSensDerivatives);
SMSExport[{1, 0, 0, 0, 1 + noSensDerivatives}, TasksData$$];

SMSDo[Ig, 1, NoIp];
ElementDefinitions[]
SMSExport[ fGauss wgp SMSReal[ed$$["Data", Igd + NoStressComponents + NoStressComponents
        (NoKinComponents + noSensParameters) + 1]], RealOutput$$[1], "AddIn" → True];

SMSDo[SensIndexi, 1, noSensParameters];
rf ⊨ SMSReal[ed$$["Data", Igd + NoStressComponents +
      NoStressComponents (NoKinComponents + noSensParameters) + 1 + NoKinComponents + SensIndexi]];
SMSExport[fGauss wgp rf , RealOutput$$[1 + SensIndexi], "AddIn" → True];
SMSEndDo[];(*sensitivity i*)
SMSEndDo[];(*integration points*).
```

Macro element `"Response functional"` task needs to be defined for FE^2 macro element as well. Its purpose is to read plastic energy W_p and its derivatives from `"Data"` field for each macro elements integration point and integrate it over macro element.