



Sensitivity analysis based multi-scale methods of coupled path-dependent problems

Nina Zupan¹ · Jože Korelc¹

Received: 18 March 2019 / Accepted: 29 August 2019 / Published online: 18 September 2019
 © The Author(s) 2019

Abstract

In the paper, a generalized essential boundary condition sensitivity analysis based implementation of FE² and mesh-in-element (MIEL) multi-scale methods is derived as an alternative to standard implementations of multi-scale analysis, where the calculation of SCHUR complement of the microscopic tangent matrix is needed for bridging the scales. The paper presents a unified approach to the development of an arbitrary MIEL or FE² computational scheme for an arbitrary path-dependent material model. Implementation is based on efficient first and second order analytical sensitivity analysis, for which automatic-differentiation-based formulation of essential boundary condition sensitivity analysis is derived. A fully consistently linearized two-level path-following algorithm is introduced as a solution algorithm for the multi-scale modeling. Sensitivity analysis allows each macro step to be followed by an arbitrary number of micro substeps while retaining quadratic convergence of the overall solution algorithm.

Keywords Two-level path-following · Multi-scale methods · Sensitivity analysis · Mesh-in-element · FE²

List of symbols

\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 end of the last macro step
$\square^{(r)}$	Index of micro problem
Φ	set of variables calculated at the selected macro element and transferred to the corresponding micro problem (also sensitivity parameters)
S	Set of variables calculated at the selected micro problem and returned to the corresponding macro element
Φ_{Me}, S_{Me}	Variables Φ and S collected for all micro problems associated with the selected macro element

λ_M, λ_m	Macro and micro level parameters
$\mathbf{R}_M, \mathbf{K}_M$	Macro level residual and tangent matrix
$\mathbf{R}_m, \mathbf{K}_m$	Micro level residual and tangent matrix
$\mathbf{p}_M \equiv \mathbf{p}_{Mk+1}$	Macro level nodal unknowns (e.g. nodal displacements)
$\mathbf{p}_m \equiv \mathbf{p}_{mn+1}$	Micro level nodal unknowns (e.g. nodal displacements)
$\bar{\mathbf{p}}_m = \bar{\mathbf{p}}_{mn+1}$	Micro level nodal unknowns with prescribed essential boundary condition
\square_e	Quantities associated with the selected macro or micro element (e.g. $\mathbf{R}_{Me}, \mathbf{R}_{me}, \mathbf{p}_{me}, \dots$)
$\mathbf{h}_{mg} \equiv \mathbf{h}_{mg n+1}$	Set of integration point unknowns at micro level (e.g. plastic state variables)
\mathbf{Q}_{mg}	Set of integration point equations at micro level
$\mathbf{h}_m \equiv \mathbf{h}_{mn+1}$	Set of unknowns at all integration points of selected micro problem
$\mathbf{F}_M, \mathbf{F}_m$	Macro and micro deformation gradients
$\mathbf{P}_M, \mathbf{P}_m$	Macro and micro first Piola–Kirchhoff stress tensors
$\mathbf{A}:\mathbf{B}$	Inner product ($\mathbf{A}:\mathbf{B} = \sum_{ij} A_{ij} B_{ij}$, $\mathbf{A} : \frac{\partial \mathbf{B}}{\partial \mathbf{p}} = \sum_{ij} A_{ij} \frac{\partial B_{ij}}{\partial \mathbf{p}}$)

✉ Jože Korelc
 joze.korelc@fgg.uni-lj.si
 Nina Zupan
 nina.zupan@fgg.uni-lj.si

¹ Faculty of Civil and Geodetic Engineering, University of Ljubljana, Jamova 2, 1000 Ljubljana, Slovenia

$\frac{\delta f(\mathbf{a})}{\delta \mathbf{a}}$ Computational derivative (the result of automatic differentiation)

When the selected variable 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.

1 Introduction

Multi-scale methods are a growing trend in computational mechanics, especially with increasing capabilities of computers. Multi-scale methods originate from the demand to accurately model heterogeneous materials, like fiber reinforced composites, particle reinforced adhesives, concrete and even metal [4,8,17]. The goal of multi-scale modeling is to design a combined macroscopic–microscopic computational algorithm that is numerically more efficient than solving of the full microscopic model directly. At the same time, it gives us the information we need with the desired accuracy. For the overview of multi-scale methods reader is referred to [1,3,6,30]. The use of different kinds of multi-scale methods is limited by the characteristics of the problem to be solved. Roughly, we can separate multi-scale methods in two groups: on methods that are based on homogenization techniques and on domain decomposition methods.

A basic hypothesis of homogenization techniques is a complete separation of scales, where the size of heterogeneities is assumed to be infinitely smaller than the structural dimensions. Homogenized material behavior of representative volume elements (RVEs), which contains microstructure, is considered to be representative of the entire or part of the structure. Standard two-level finite element homogenization approach (FE²) described in [16,29] is appropriate for the problems where scales are separated far enough and are only weakly coupled, see [2,5,23,27]. If the difference between two scales is finite, in the region of high gradients or in the case of localization [25] the FE² approach fails. Then some sort of domain decomposition method can be applied. One such method is substructuring method, now more commonly named mesh-in-element (MIEL) scheme, which was described in [9].

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. The first aim of this paper was to develop a nonlinear multi-scale computational scheme with two interacting path-following methods at two levels. An algorithm will be derived for consistent parametrization of both macro and micro problems leading to a two-level path-following algorithm. For the purpose of

convergence comparison, one method from each group was implemented, FE² and MIEL.

In the literature, a lot of attention has been paid to the computation of macroscopic tangent as an essential and numerically demanding part of any multi-scale simulation. The possibilities vary from expensive and inaccurate but general finite difference approximation of macroscopic tangent, to various ways how to derive corresponding analytical expressions (for discussion on methods see e.g. [28]). One of the alternatives is also standard sensitivity analysis of coupled path-dependent problems, as introduced in [11,21]. In the primal analysis, the response of the system is evaluated, whereas in sensitivity analysis the derivatives of the response, e.g. displacements, strains, stresses or energy, with respect to arbitrary design parameter are sought. For the automation of the multi-scale methods, sensitivity analysis with respect to parameters used to define micro level boundary conditions is needed. It will be shown that the consistent linearization of the two-level path-following algorithm requires the implementation of relative sensitivity analysis instead of a full one and that a second order sensitivity analysis is also needed.

The second aim of the paper is to present advantages of analytical essential boundary condition sensitivity analysis based implementation in comparison with the classical ways of implementing multi-scale methods based on the calculation of SCHUR complement of micro tangent matrix (see e.g. [16,22,28] for FE² method and [9] for MIEL method). This is especially important for path-dependent problems such as finite strain plasticity, where consistent linearization is of high importance. It will be shown that for the MIEL type of methods the analytical second order sensitivity analysis is numerically superior with respect to SCHUR complement implementation.

Another motivation was to create a computational environment, 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. To achieve the goal, the introduced method uses an advanced feature of software tool AceGen [12]. AceGen is an automatic code generator, where automatic differentiation technique [7], automatic code optimization and code generation are combined with computer algebra system Mathematica [20]. The size of the code is reduced through control of expression swell [10]. The automatic-differentiation-based (ADB, [11]) formulation enables unification and automation of various multi-scale approaches for arbitrary nonlinear, path-dependent material models (e.g. general finite strain plasticity). A short overview of multi-scale methods based on first and second order boundary condition sensitivity analysis for the linear case was already given in [15]. Numerical simulations were performed with AceFEM numerical environment [14] that has built-in support for numerically

efficient first and second order analytical sensitivity analysis.

The paper is organized as follows. Micro problem formulation based on finite strain elasto-plastic model is briefly described in Sect. 1.1 as a basis for all subsequently derived multi-scale methods. After an introduction to automatic differentiation based notation (ADB) in Sect. 1.2, a generalized two-level path-following multi-scale algorithm is derived in Sect. 2. The basis for the multi-scale formulations is primal and sensitivity analysis of the micro problem as given in Sect. 3. MIEL and FE² multi-scale methods are then described in detail, focusing on their implementation in Sect. 4, followed by numerical examples presented in Sect. 5.

1.1 Micro problem definition

At the micro level we will consider finite element formulation of rate-independent nonlinear problems in solid and structural mechanics, such as an arbitrary finite strain rate-independent elasto-plastic material model. Here, only essential equations are presented; for more details see e.g. [26]. 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. Some of the possible variants are presented in [13]. The actual material model used to run numerical examples is summarized in Box 1. However, all the methods presented are general, independent of the specific material model.

$\mathbf{b}_e = \mathbf{F}_m \mathbf{C}_p^{-1} \mathbf{F}_m^{-T}, J_{b_e} = \det \mathbf{b}_e \quad (1)$
$W = \frac{1}{2} \mu (\text{tr} \mathbf{b}_e - 3 - \log J_{b_e}) + \frac{1}{4} \lambda (J_{b_e} - 1 - \log J_{b_e}) \quad (2)$
$\boldsymbol{\tau} = 2\mathbf{b}_e \frac{\partial W}{\partial \mathbf{b}_e}, \boldsymbol{\tau}' = \boldsymbol{\tau} - \frac{1}{3} (\text{tr} \boldsymbol{\tau}) \mathbf{I} \quad (3)$
$\phi = \left(\frac{3}{2} \boldsymbol{\tau}' \cdot \boldsymbol{\tau}' \right)^{1/2} - (\sigma_{y0} + K_h \gamma + R_\infty (1 - \exp(-\delta \gamma))) \quad (4)$
$\mathbf{Z} = \mathbf{F}_m \mathbf{C}_p^{-1} - \exp(-2(\gamma - \gamma_n) \frac{\partial \phi}{\partial \boldsymbol{\tau}}) \mathbf{F}_m \mathbf{C}_{pn}^{-1} = \mathbf{0} \quad (5)$
$\mathbf{Q}_{mg} = \{Z_{11}, Z_{22}, Z_{33}, Z_{12}, Z_{13}, Z_{23}, \phi\} = \mathbf{0} \quad (6)$
$\mathbf{h}_{mg} = \{C_{p11}^{-1}, C_{p22}^{-1}, C_{p33}^{-1}, C_{p12}^{-1}, C_{p13}^{-1}, C_{p23}^{-1}, \gamma\} \quad (7)$
$\mathbf{h}_{mgn} = \{C_{pn11}^{-1}, C_{pn22}^{-1}, C_{pn33}^{-1}, C_{pn12}^{-1}, C_{pn13}^{-1}, C_{pn23}^{-1}, \gamma_n\} \quad (8)$

Box 1. Micro problem material model

Formulation is based on multiplicative split of micro deformation gradient \mathbf{F}_m , the components of an inverse right Cauchy plastic strain tensor \mathbf{C}_p^{-1} as plastic state variables, elastic left Cauchy strain tensor \mathbf{b}_e (1), Neo-Hookean strain energy function $W(\mathbf{b}_e)$ (2) and Mises yield function with exponential hardening law (4). Backward Euler is combined

with the exponential map for a stable, volume conserving integration of evolution equations [26]. Discretized evolution equations (5), together with yield condition $\phi = 0$, form a set of algebraic equations \mathbf{Q}_{mg} (6) for a set \mathbf{h}_{mg} (7) of unknown components of plastic strain tensor \mathbf{C}_p^{-1} and plastic multiplier γ . \mathbf{C}_{pn}^{-1} and γ_n are values of plastic strain tensor and plastic multiplier at the end of the last load step. The dependency of Eq. (6) 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 then written as

$$\int_{\Omega_m} \mathbf{P}_m \cdot \delta \mathbf{F}_m dV - \int_{\partial \Omega_m} \mathbf{t} \cdot \delta \mathbf{u}_m dS = 0 \quad (9)$$

where first Piola–Kirchhof stress tensor \mathbf{P}_m can be obtained from the elastic strain energy W by $\mathbf{P}_m = \partial W / \partial \mathbf{F}_m$. After finite element discretization of deformation gradient $\mathbf{F}_m = \mathbf{F}_m(\mathbf{p}_{me})$, where \mathbf{p}_{me} is a set of nodal degrees of freedom of e th micro element at the current load step. The variation $\delta \mathbf{F}_m = \partial \mathbf{F}_m / \partial \mathbf{p}_{me} \delta \mathbf{p}_{me}$ leads from (9) together with the standard GAUSS integration of weak form and standard procedure of assembly of element contributions (denoted here with \mathbf{A} operator) to a set of algebraic equilibrium equations (10). Equations are at each GAUSS point coupled with an additional set of equations \mathbf{Q}_{mg} (6). The result is the following integration point coupled system of nonlinear algebraic equations

$$\begin{aligned} \mathbf{R}_m(\mathbf{p}_m, \mathbf{h}_m) &= \mathbf{A}_{e=1}^{n_{me}} \mathbf{R}_{me} + \mathbf{R}_m^{\text{ext}} \quad (10) \\ &= \mathbf{A}_{e=1}^{n_{me}} \sum_{g \in G_e} w_{gp} \mathbf{R}_{mg}(\mathbf{p}_{me}, \mathbf{h}_{mg}) + \mathbf{R}_m^{\text{ext}} = \mathbf{0}, \\ \mathbf{Q}_{mg}(\mathbf{p}_{me}, \mathbf{h}_{mg}, \mathbf{h}_{mgn}) &= \mathbf{0} : g = 1, 2, \dots, n_{tg} \quad (11) \end{aligned}$$

where \mathbf{R}_{me} is 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 (9) to

$$\mathbf{R}_{mg} = J_\xi \mathbf{P}_m : \frac{\partial \mathbf{F}_m}{\partial \mathbf{p}_{me}} \quad (12)$$

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

1.2 Automatic differentiation based (ADB) notation

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 $\frac{\hat{\delta} f(\mathbf{a})}{\hat{\delta} \mathbf{a}}$. The AD operator $\frac{\hat{\delta} f(\mathbf{a})}{\hat{\delta} \mathbf{a}}$ represents derivative of a function $f(\mathbf{a})$ with respect to variables \mathbf{a} . The operator $\frac{\hat{\delta} f(\mathbf{a})}{\hat{\delta} \mathbf{a}}$ has a dual purpose to indicate the mathematical operation of differentiation as well as to indicate that the AD technique is used to obtain the required quantity. If, for example, alternative or additional dependencies for a set of intermediate variables \mathbf{b} have to be considered for differentiation, then the AD exception is indicated by the following formalism

$$\left. \frac{\hat{\delta} f(\mathbf{a}, \mathbf{b})}{\hat{\delta} \mathbf{a}} \right|_{\frac{D\mathbf{b}}{D\mathbf{a}}=\mathbf{M}} = \frac{\partial f}{\partial \mathbf{a}} + \frac{\partial f}{\partial \mathbf{b}} \mathbf{M}, \quad (13)$$

which indicates that during the AD procedure, the total derivatives of variables \mathbf{b} with respect to variables \mathbf{a} are set to be equal to matrix \mathbf{M} . \mathbf{b} in (13) may or may not be algorithmically a function of \mathbf{a} . When \mathbf{b} is algorithmically function of \mathbf{a} then (13) defines that the true derivatives $\frac{\partial \mathbf{b}}{\partial \mathbf{a}}$ are neglected and replaced by a matrix \mathbf{M} . When \mathbf{b} is not algorithmically a function of \mathbf{a} then (13) 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.

For example, the GAUSS point residual \mathbf{R}_{mg} is defined by Eq. (12). However, form (12) is not numerically efficient from the automatic differentiation point of view. Numerically efficient ADB form of (12) is derived as

$$\begin{aligned} \mathbf{R}_{mg} &= J_{\xi} \mathbf{P}_m : \frac{\partial \mathbf{F}_m}{\partial \mathbf{p}_{me}} \\ &= J_{\xi} \left. \frac{\hat{\delta} W}{\hat{\delta} \mathbf{F}_m} \right|_{\frac{D\mathbf{h}_{mg}}{D\mathbf{F}_m}=\mathbf{0}} : \frac{\hat{\delta} \mathbf{F}_m}{\hat{\delta} \mathbf{p}_{me}} = J_{\xi} \left. \frac{\hat{\delta} W}{\hat{\delta} \mathbf{p}_{me}} \right|_{\frac{D\mathbf{h}_{mg}}{D\mathbf{F}_m}=\mathbf{0}} \end{aligned} \quad (14)$$

As a side effect of the iterative solution of Gauss-point equations (6), there exist an implicit (algorithmic) dependency of \mathbf{h}_{mg} on \mathbf{F}_m . The AD exception $\frac{D\mathbf{h}_{mg}}{D\mathbf{F}_m} = \mathbf{0}$ in (14) hides this dependency from automatic differentiation procedure and ensures correct evaluation of the weak form equations. In (14) 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 [11].

The introduced ADB notation is abstract, thus any sufficiently sophisticated software for automatic differentiation

can be used for the actual implementation. 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 [10–12, 14]. The actual AceGen and AceFEM inputs are for the complex multi-scale problems addressed in the paper too lengthy to be included in the paper. 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>.

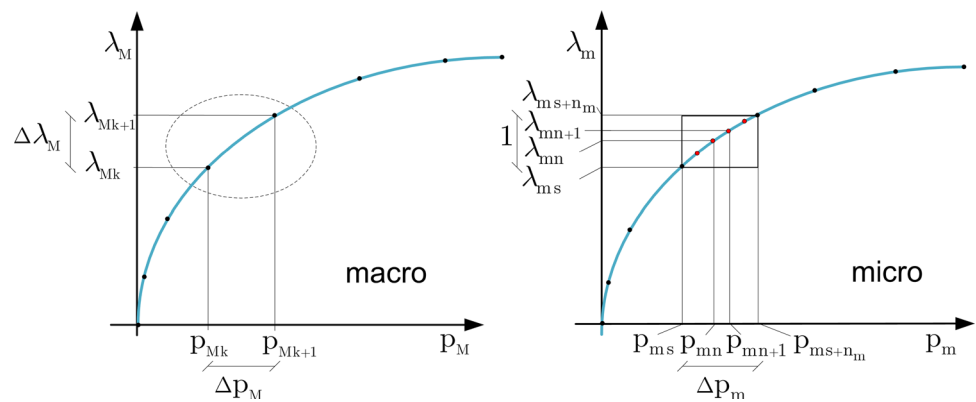
2 Generalized two-level path-following multi-scale algorithm

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 then 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.

Let k be the index of the last calculated macro load step and $k + 1$ the current macro load step, as shown in Fig. 1. 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. 1.

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 . 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 then leads to a set of nonlinear equations \mathbf{R}_M at the current load level $\lambda_M = \lambda_{Mk+1}$

Fig. 1 Generalized two-level path-following, multi-scale algorithm



$$\mathbf{R}_M \left(\mathbf{p}_M, \bigcup_{e=1}^{n_{Me}} \mathbf{S}_{Me}, \lambda_M \right) = \sum_{e=1}^{n_{Me}} \mathbf{R}_{Me}(\mathbf{p}_{Me}, \mathbf{S}_{Me}) - \lambda_M \mathbf{R}_M^{\text{ref}} = \mathbf{0} \quad (15)$$

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.

Within various multi-scale methods the coupling of the scales can be done in several ways. The paper 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 FE² 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 given 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

$$\bar{\mathbf{p}}_m(\boldsymbol{\phi}, \lambda_m) = \bar{\mathbf{p}}_{ms} + (\lambda_m - \lambda_{ms})(\bar{\mathbf{p}}_m(\boldsymbol{\phi}) - \bar{\mathbf{p}}_{ms}). \quad (16)$$

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

$$\Delta \bar{\mathbf{p}}_m = \bar{\mathbf{p}}_m(\boldsymbol{\phi}) - \bar{\mathbf{p}}_{ms}. \quad (17)$$

The micro level parameter introduced with (16) 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 (10) to the following integration point coupled system of nonlinear algebraic equations for the chosen micro problem

$$\mathbf{R}_m(\mathbf{p}_m, \mathbf{h}_m, \bar{\mathbf{p}}_m(\boldsymbol{\phi}, \lambda_m)) = \sum_{e=1}^{n_{me}} \sum_{g \in G_e} w_{gp} \mathbf{R}_{mg}(\mathbf{p}_{me}(\boldsymbol{\phi}, \lambda_m), \mathbf{h}_{mg}) + \mathbf{R}_m^{\text{ext}} = \mathbf{0} \quad (18)$$

$$\mathbf{Q}_{mg}(\mathbf{p}_{me}(\boldsymbol{\phi}, \lambda_m), \mathbf{h}_{mg}, \mathbf{h}_{mgn}) = \mathbf{0} : g = 1, 2, \dots, n_{tg} \quad (19)$$

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 [11].

In order to have quadratically convergent multi-scale solution algorithm, we also need consistently linearized macro equilibrium equations (15). The linearization of (15) leads to

$$\begin{aligned} \mathbf{K}_M &= \bigwedge_{e=1}^{n_{Me}} \mathbf{K}_{Me} \\ &= \bigwedge_{e=1}^{n_{Me}} \left(\frac{\partial \mathbf{R}_{Me}}{\partial \mathbf{p}_{Me}} + \frac{\partial \mathbf{R}_{Me}}{\partial \mathbf{S}_{Me}} \frac{D\mathbf{S}_{Me}}{D\boldsymbol{\Phi}_{Me}} \frac{\partial \boldsymbol{\Phi}_{Me}}{\partial \mathbf{p}_{Me}} \right) \end{aligned} \quad (20)$$

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 (20) are explicit and can be easily derived analytically for a specific multi-scale scheme.

Derivation of 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 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

$$\begin{aligned} \frac{D\mathbf{S}}{D\boldsymbol{\Phi}} &= \frac{\partial \mathbf{S}}{\partial \mathbf{p}_m} \frac{D\mathbf{p}_m}{D\boldsymbol{\Phi}} + \frac{\partial \mathbf{S}}{\partial \mathbf{h}_m} \frac{D\mathbf{h}_m}{D\boldsymbol{\Phi}} + \frac{\partial \mathbf{S}}{\partial (D\mathbf{p}_m/D\boldsymbol{\Phi})} \frac{D^2\mathbf{p}_m}{D\boldsymbol{\Phi}^2} \\ &\quad + \frac{\partial \mathbf{S}}{\partial (D\mathbf{h}_m/D\boldsymbol{\Phi})} \frac{D^2\mathbf{h}_m}{D\boldsymbol{\Phi}^2} \end{aligned} \quad (21)$$

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 complete path-following algorithm for the solution of selected micro problem. This can be done using analytical sensitivity analysis procedures, such as described in [11]. $\boldsymbol{\Phi}$ represents input data for the selected micro level simulation and is used to calculate essential boundary conditions (16). 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. However, it is not actually needed. 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,

$$\begin{aligned} \frac{D\mathbf{p}_{mi}}{D\boldsymbol{\Phi}} &= \mathbf{0}, \quad \frac{D\mathbf{h}_{mi}}{D\boldsymbol{\Phi}} = \mathbf{0}, \\ \frac{D^2\mathbf{p}_{mi}}{D\boldsymbol{\Phi}^2} &= \mathbf{0}, \quad \frac{D^2\mathbf{h}_{mi}}{D\boldsymbol{\Phi}^2} = \mathbf{0} : \forall i \leq s, \end{aligned} \quad (22)$$

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 (22) holds. Since \mathbf{Q}_{mg} depends only on \mathbf{h}_{mg} , it is sufficient for (22) to hold to set

$$\frac{D\mathbf{h}_{ms}}{D\boldsymbol{\Phi}} = \mathbf{0} \quad \text{and} \quad \frac{D^2\mathbf{h}_{ms}}{D\boldsymbol{\Phi}^2} = \mathbf{0} \quad (23)$$

at the start of each micro problem increment (at the s th micro step).

2.1 Two-level path-following algorithm

The algorithm that summarizes the above considerations is presented in Box 2. First, the micro level equations (18) (19) 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 (15) 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 paper 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 parallelization of the multi-scale algorithm. Let us assume that the code supports primal, first and second order sensitivity analysis. It is then needed 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 ($DS/D\Phi$),
- macro element residual vector ($\mathbf{R}_{Me}(\mathbf{p}_{Me}, \mathbf{S}_{Me})$),
- macro element tangent matrix ($\mathbf{K}_{Me}(\mathbf{p}_{Me}, \mathbf{S}_{Me}, DS_{Me}/D\Phi_{Me})$).

```

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  $\mathbf{p}_{ms} \leftarrow \mathbf{0}$ ;  $\mathbf{h}_{ms} \leftarrow \mathbf{0}$ ;  $\bar{\mathbf{p}}_m \leftarrow \mathbf{0}$ ;
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  $\Phi_{Me}$ 
      foreach micro problem do
         $\mathbf{p}_{mn} \leftarrow \mathbf{p}_{ms}$ ;  $\mathbf{h}_{mn} \leftarrow \mathbf{h}_{ms}$  // initialize
        primal data
         $D\mathbf{h}_{mn}/D\Phi \leftarrow \mathbf{0}$ ;  $D^2\mathbf{h}_{mn}/D\Phi^2 \leftarrow \mathbf{0}$  // delete
        sensitivity history to fulfill
        (22)
         $\lambda_m \leftarrow i - 1$ ;  $\Delta\lambda_m = 1/n_m$ ;  $\Delta\bar{\mathbf{p}}_m \leftarrow \bar{\mathbf{p}}_m(\Phi) - \bar{\mathbf{p}}_{ms}$ ;
        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  $\mathbf{p}_m, \mathbf{h}_m$ 
           $D\mathbf{p}_m/D\Phi$ ;  $D\mathbf{h}_m/D\Phi$  // first order
          sensitivity problem
           $D^2\mathbf{p}_m/D\Phi^2$ ;  $D^2\mathbf{h}_m/D\Phi^2$  // second
          order sensitivity problem if
          needed
           $\mathbf{p}_{mn} \leftarrow \mathbf{p}_m$ ;  $\mathbf{h}_{mn} \leftarrow \mathbf{h}_m$  // update primal
          data
           $D\mathbf{p}_{mn}/D\Phi \leftarrow D\mathbf{p}_m/D\Phi \dots D^2\mathbf{h}_{mn}/D\Phi^2 \leftarrow$ 
           $D^2\mathbf{h}_m/D\Phi^2$  // update all
          sensitivity data
        evaluate and store S and DS/D $\Phi$ 
      Newton update at macro level
       $\mathbf{R}_M = \mathbf{A}_e^{n_{Me}} \mathbf{R}_{Me}(\mathbf{p}_{Me}, \mathbf{S}_{Me})$ ;
       $\mathbf{K}_M = \mathbf{A}_e^{n_{Me}} \mathbf{K}_{Me}(\mathbf{p}_{Me}, \mathbf{S}_{Me}, DS_{Me}/D\Phi_{Me})$ ;
       $\mathbf{p}_M \leftarrow \mathbf{p}_M - \mathbf{K}_M^{-1} \mathbf{R}_M$ 
    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

```

Box 2. Two-level path-following multi-scale algorithm

3 Primal and sensitivity analysis of micro problem

The general procedures for primal and first order sensitivity analysis are for an arbitrary problem presented in detail

in [11]. Here, we focus on essential boundary condition sensitivity analysis that is needed for the implementation of multi-scale methods. For this purpose, first order essential boundary condition sensitivity analysis is extended to second order essential boundary condition sensitivity analysis.

3.1 Solution and automation of primal problem

The primal problem at micro level (18), (19) represents a system of GAUSS point coupled nonlinear algebraic equations that can be solved using standard nested iteration–subiteration Newton–Raphson iterative procedure (see e.g. [11]). First, the Gauss point equation (19) 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 (18) in an outer loop for unknown \mathbf{p}_m . The tangent operator for the inner loop \mathbf{K}_Q is given by

$$\mathbf{K}_Q = \frac{\hat{\delta}\mathbf{Q}_{mg}}{\hat{\delta}\mathbf{h}_{mg}} \quad (24)$$

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

$$\mathbf{K}_m = \mathbf{A} \sum_{e=1}^{n_{me}} w_{gp} \frac{\hat{\delta}\mathbf{R}_{mg}}{\hat{\delta}\mathbf{p}_{me}} \bigg|_{\frac{D\mathbf{h}_{mg}}{D\mathbf{p}_{me}} = -\mathbf{K}_Q^{-1} \frac{\hat{\delta}\mathbf{Q}_{mg}}{\hat{\delta}\mathbf{p}_{me}}} \quad (25)$$

Evaluation of the micro tangent matrix (25) requires proper consideration of the implicit dependency $\mathbf{h}_{mg}(\mathbf{p}_{me})$ introduced by the local iterative procedure. The missing implicit derivative $\frac{\partial \mathbf{h}_{mg}}{\partial \mathbf{p}_{me}} = -\mathbf{K}_Q^{-1} \frac{\partial \mathbf{Q}_{mg}}{\partial \mathbf{p}_{me}}$ can be easily obtained from (19) and introduced as an AD exception in (25) (for more details see [11]).

3.2 Solution and automation of sensitivity problem

For the essential boundary condition sensitivity analysis we define the residuals and the vectors of unknowns in (18) and (19) as a function of sensitivity parameters Φ by

$$\mathbf{R}_m(\mathbf{p}_m(\Phi), \mathbf{h}_m(\Phi), \bar{\mathbf{p}}_m(\Phi, \lambda_m)) \quad (26)$$

$$= \mathbf{A} \sum_{e=1}^{n_{me}} w_{gp} \mathbf{R}_{mg}(\check{\mathbf{p}}_{me}(\Phi), \bar{\mathbf{p}}_{me}(\Phi, \lambda_m), \mathbf{h}_{mg}(\Phi)) = \mathbf{0},$$

$$\mathbf{Q}_{mg}(\check{\mathbf{p}}_{me}(\Phi), \bar{\mathbf{p}}_{me}(\Phi, \lambda_m), \mathbf{h}_{mg}(\Phi), \mathbf{h}_{mgn}(\Phi)) = \mathbf{0} : \quad (27)$$

$$g = 1, 2, \dots, n_{tg}.$$

where $\bar{\mathbf{p}}_m(\Phi, \lambda_m)$ is a set of nodal DOF with prescribed essential boundary conditions defined by (16).

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 $\check{\mathbf{p}}_{me}$, because they are at the element level indistinguishable. The first order sensitivity problem can be obtained from the primal problem by differentiating equations (26) and (27) with respect to sensitivity parameters, which results in

$$\frac{D\mathbf{R}_m}{D\phi_I} = \mathbf{A} \sum_{e=1}^{n_{me}} \sum_{g \in G_e} w_{gp} \left(\frac{\partial \mathbf{R}_{mg}}{\partial \check{\mathbf{p}}_{me}} \frac{D\check{\mathbf{p}}_{me}}{D\phi_I} + \frac{\partial \mathbf{R}_{mg}}{\partial \bar{\mathbf{p}}_{me}} \frac{D\bar{\mathbf{p}}_{me}}{D\phi_I} + \frac{\partial \mathbf{R}_{mg}}{\partial \mathbf{h}_{mg}} \frac{D\mathbf{h}_{mg}}{D\phi_I} \right) = 0, \quad (28)$$

$$\frac{D\mathbf{Q}_{mg}}{D\phi_I} = \frac{\partial \mathbf{Q}_{mg}}{\partial \check{\mathbf{p}}_{me}} \frac{D\check{\mathbf{p}}_{me}}{D\phi_I} + \frac{\partial \mathbf{Q}_{mg}}{\partial \bar{\mathbf{p}}_{me}} \frac{D\bar{\mathbf{p}}_{me}}{D\phi_I} + \frac{\partial \mathbf{Q}_{mg}}{\partial \mathbf{h}_{mg}} \frac{D\mathbf{h}_{mg}}{D\phi_I} + \frac{\partial \mathbf{Q}_{mg}}{\partial \mathbf{h}_{mgn}} \frac{D\mathbf{h}_{mgn}}{D\phi_I} = \mathbf{0}. \quad (29)$$

To calculate $\frac{D\mathbf{p}_m}{D\phi_I}$, the sensitivities $\frac{D\mathbf{h}_{mg}}{D\phi_I}$ are expressed from Eq. (29) and inserted into Eq. (28). After rearrangement, in which the terms that contain the unknown sensitivity $\frac{D\check{\mathbf{p}}_{me}}{D\phi_I}$ are collected together, a system of linear equations is obtained

$$\mathbf{K}_m \frac{D\mathbf{p}_m}{D\phi_I} = -{}^I\tilde{\mathbf{R}}_m, \quad {}^I\tilde{\mathbf{R}}_m = \mathbf{A} \sum_{e=1}^{n_{me}} \sum_{g \in G_e} w_{gp} {}^I\tilde{\mathbf{R}}_{mg}. \quad (30)$$

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 Eq. (30) 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 $\frac{D\check{\mathbf{p}}_{me}}{D\phi_I}$ as the solution of (30), the obtained values are inserted into Eq. (29) and $\frac{D\mathbf{h}_{mg}}{D\phi_I}$, $g = 1, 2, \dots, n_{lg}$ can be expressed. Corresponding expressions are

$${}^I\mathbf{Z}_g = -\mathbf{K}_Q^{-1} \left(\frac{\partial \mathbf{Q}_{mg}}{\partial \bar{\mathbf{p}}_{me}} \frac{D\bar{\mathbf{p}}_{me}}{D\phi_I} + \frac{\partial \mathbf{Q}_{mg}}{\partial \mathbf{h}_{mgn}} \frac{D\mathbf{h}_{mgn}}{D\phi_I} \right) \quad (31)$$

$${}^I\tilde{\mathbf{R}}_{mg} = \frac{\partial \mathbf{R}_{mg}}{\partial \bar{\mathbf{p}}_{me}} \frac{D\bar{\mathbf{p}}_{me}}{D\phi_I} + \frac{\partial \mathbf{R}_{mg}}{\partial \mathbf{h}_{mg}} {}^I\mathbf{Z}_g \quad (32)$$

$$\frac{D\mathbf{h}_{mg}}{D\phi_I} = {}^I\mathbf{Z}_g - \mathbf{K}_Q^{-1} \left(\frac{\partial \mathbf{Q}_{mg}}{\partial \check{\mathbf{p}}_{me}} \frac{D\check{\mathbf{p}}_{me}}{D\phi_I} \right) \quad (33)$$

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 input data of the finite element analysis. However, 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

$${}^I\mathbf{V}_e = \begin{cases} \frac{\partial \bar{\mathbf{p}}_{mei}}{\partial \phi_I} & \text{if } p_{mei} \in \bar{\mathbf{p}}_{me} : i = 1, \dots, n_p, \\ 0 & \text{if } p_{mei} \in \check{\mathbf{p}}_{me} \end{cases} \quad (34)$$

$${}^{IJ}\mathbf{V}_e = \begin{cases} \frac{\partial^2 \bar{\mathbf{p}}_{mei}}{\partial \phi_I \partial \phi_J} & \text{if } p_{mei} \in \bar{\mathbf{p}}_{me} : i = 1, \dots, n_p \\ 0 & \text{if } p_{mei} \in \check{\mathbf{p}}_{me} \end{cases} \quad (35)$$

where n_p is the total number of element nodal DOFs. The velocity field is zero for the true degrees of freedom. Thus, the proper definition of element velocity fields is sufficient to make 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 (32), (31) and (33) are rather lengthy, but they can be obtained automatically 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 first order terms follows from (32), (31) and (33) where all implicit derivatives are replaced by appropriate AD exceptions and leads to

$${}^I\mathbf{Z}_g = -\mathbf{K}_Q^{-1} \frac{\delta \mathbf{Q}_{mg}}{\delta \phi_I} \bigg|_{\substack{D\bar{\mathbf{p}}_{me}/D\phi_I = {}^I\mathbf{V}_e, \\ D\mathbf{h}_{mgn}/D\phi_I = {}^I\mathbf{H}_g^n}} \quad (36)$$

$${}^I\tilde{\mathbf{R}}_{mg} = \frac{\delta \mathbf{R}_{mg}}{\delta \phi_I} \bigg|_{\substack{D\bar{\mathbf{p}}_{me}/D\phi_I = {}^I\mathbf{V}_e, \\ D\mathbf{h}_{mg}/D\phi_I = {}^I\mathbf{Z}_g}} \quad (37)$$

$$\frac{D\mathbf{h}_{mg}}{D\phi_I} = {}^I\mathbf{Z}_g - \mathbf{K}_Q^{-1} \frac{\delta \mathbf{Q}_{mg}}{\delta \phi_I} \bigg|_{\substack{D\bar{\mathbf{p}}_{me}/D\phi_I = \begin{cases} 0 & \text{if } p_{mei} \in \bar{\mathbf{p}}_{me} : i=1, \dots, n_p \\ {}^I\mathbf{Y}_i & \text{if } p_{mei} \in \check{\mathbf{p}}_{me} \end{cases}}} \quad (38)$$

where ${}^I\mathbf{Y} = \frac{D\bar{\mathbf{p}}_{me}}{D\phi_I}$ 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 a pro-

cedure equivalent to the one for the first-order sensitivity analysis, one gets a system of linear equations

$$\mathbf{K}_m \frac{D^2 \mathbf{p}_m}{D\phi_I D\phi_J} = -{}^{IJ} \tilde{\mathbf{R}}_m, \quad {}^{IJ} \tilde{\mathbf{R}}_m = \sum_{e=1}^{n_{me}} w_{gp} {}^{IJ} \tilde{\mathbf{R}}_{mg}, \quad (39)$$

where again only vector ${}^{IJ} \tilde{\mathbf{R}}_m$ on the right-hand side of Eq. (39) 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 $\frac{D^2 \mathbf{h}_{mg}}{D\phi_I D\phi_J}$, $g = 1, 2, \dots, n_{tg}$ can be expressed. The actual analytical expressions are too lengthy to be presented here. Thus, only an 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

$${}^{IJ} \mathbf{Z}_g = -\mathbf{K}_Q^{-1} \frac{\hat{\delta}}{\hat{\delta}\phi_J} \times \left(\begin{array}{c} \frac{\hat{\delta} \mathbf{Q}_{mg}}{\hat{\delta}\phi_I} \left| \frac{D\mathbf{p}_{me}}{D\phi_I} = {}^I \mathbf{Y}, \right. \\ \frac{D\mathbf{h}_{mg}}{D\phi_I} = {}^I \mathbf{H}_g, \frac{D\mathbf{h}_{mg}}{D\phi_I} = {}^I \mathbf{H}_g^n \end{array} \right) \left| \begin{array}{c} \frac{D\mathbf{p}_{me}}{D\phi_J} = {}^I \mathbf{Y}, \frac{D^I \mathbf{Y}}{D\phi_J} = {}^{IJ} \mathbf{V}_e, \\ \frac{D\mathbf{h}_{mg}}{D\phi_J} = {}^J \mathbf{H}_g, \frac{D^I \mathbf{H}_g^n}{D\phi_J} = {}^{IJ} \mathbf{H}_g^n \end{array} \right. \quad (40)$$

$${}^{IJ} \tilde{\mathbf{R}}_{mg} = \frac{\hat{\delta}}{\hat{\delta}\phi_J} \left(\begin{array}{c} \frac{\hat{\delta} \mathbf{R}_{mg}}{\hat{\delta}\phi_I} \left| \frac{D\mathbf{p}_{me}}{D\phi_I} = {}^I \mathbf{Y}, \right. \\ \frac{D\mathbf{h}_{mg}}{D\phi_I} = {}^I \mathbf{H}_g \end{array} \right) \left| \begin{array}{c} \frac{D^I \mathbf{Y}}{D\phi_J} = {}^{IJ} \mathbf{V}_e, \frac{D\mathbf{p}_{me}}{D\phi_J} = {}^J \mathbf{Y}, \\ \frac{D^I \mathbf{H}_g}{D\phi_J} = {}^{IJ} \mathbf{Z}_g \end{array} \right. \quad (41)$$

$$\frac{D^2 \mathbf{h}_{mg}}{D\phi_I D\phi_J} = {}^{IJ} \mathbf{Z}_g - \mathbf{K}_Q^{-1} \frac{\hat{\delta} \mathbf{Q}_{mg}}{\hat{\delta}\phi_I} \left| \frac{D\mathbf{p}_{me}}{D\phi_I} = \begin{cases} 0 & \text{if } p_{ei} \in \bar{\mathbf{p}}_{me} \\ {}^{IJ} \mathbf{Y}_i & \text{if } p_{ei} \in \mathbf{p}_{me} \setminus \bar{\mathbf{p}}_{me} \end{cases} : i=1, \dots, n_p \right. \quad (42)$$

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 ${}^I \mathbf{Y} = \frac{D\mathbf{p}_{me}}{D\phi_I}$, ${}^J \mathbf{Y} = \frac{D\mathbf{p}_{me}}{D\phi_J}$, ${}^{IJ} \mathbf{Y} = \frac{D^2 \mathbf{p}_{me}}{D\phi_I D\phi_J}$, ${}^I \mathbf{H}_g = \frac{D\mathbf{h}_{mg}}{D\phi_I}$, ${}^J \mathbf{H}_g = \frac{D\mathbf{h}_{mg}}{D\phi_J}$, ${}^I \mathbf{H}_g^n = \frac{D\mathbf{h}_{mg}}{D\phi_I}$, ${}^J \mathbf{H}_g^n = \frac{D\mathbf{h}_{mg}}{D\phi_J}$, ${}^{IJ} \mathbf{H}_g^n = \frac{D^2 \mathbf{h}_{mg}}{D\phi_I D\phi_J}$ are already calculated and stored as first and second order sensitivities.

All first order sensitivities have to be calculated to be able to calculate the second order sensitivities. For this reason, in the algorithm in Box 2 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 Eqs. (36)–(42) are velocity fields ${}^I \mathbf{V}_e$ and ${}^{IJ} \mathbf{V}_e$.

4 Multiscale methods

4.1 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 FE² multi-scale approach fails. The MIEL scheme was described in [18, 19, 24]. Next, we developed an automatized sensitivity analysis based version of the 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. 2 the MIEL procedure is presented. Let assume the standard interpolation of displacements \mathbf{u}_M on the boundary of the macro element

$$\mathbf{u}_M(\Xi) = \sum_i N_i(\Xi) \mathbf{u}_{Mei} \quad (43)$$

where $N_i(\Xi)$ are finite element shape functions, $\Xi = (\xi, \eta, \zeta)$ reference coordinates and \mathbf{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

$$\bar{\mathbf{u}}_m(\Xi) = (\bar{\mathbf{u}}_{ms}(\Xi) + (\lambda_m - \lambda_{ms})(\mathbf{u}_M(\Xi) - \bar{\mathbf{u}}_{ms}(\Xi))) \quad (44)$$

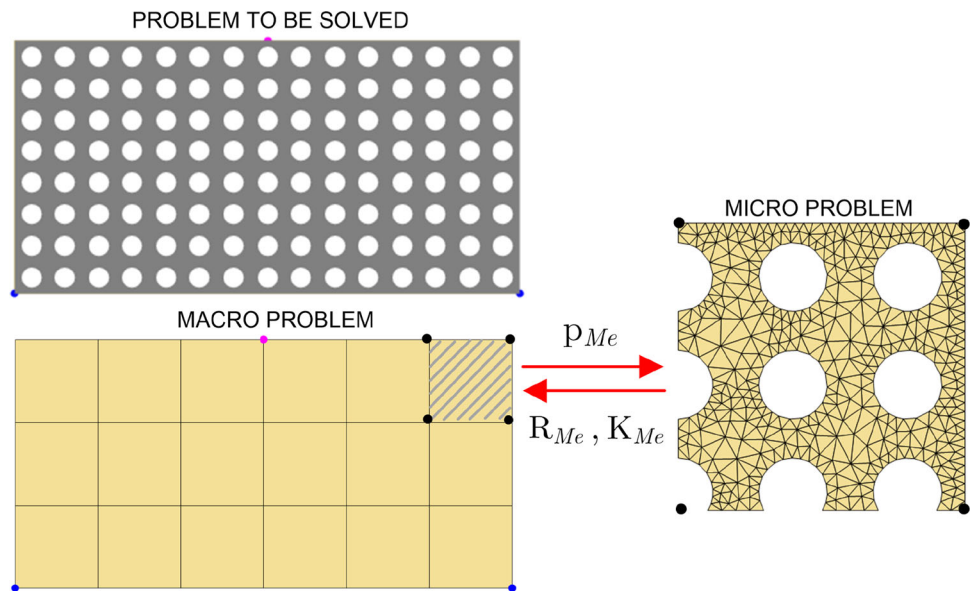
where $\bar{\mathbf{u}}_{ms}(\Xi)$ are displacements at the boundary at the end of the last macro step. The derivatives of (44) with respect to components of macro element nodal displacements are given by

$$\frac{\partial \bar{\mathbf{u}}_{mi}(\Xi)}{\partial u_{Mejk}} = \delta_{ik} (\lambda_m - \lambda_{ms}) N_j(\Xi). \quad (45)$$

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, (44) 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} .

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 (9), over the micro mesh, where the micro deformation

Fig. 2 MIEL multi-scale scheme



gradient $\mathbf{F}_m = \mathbf{F}_m(\mathbf{p}_{me}(\mathbf{p}_{Me}, \lambda_m))$ and micro stress tensor $\mathbf{P}_m = \mathbf{P}_m(\mathbf{p}_{me}(\mathbf{p}_{Me}, \lambda_m))$ implicitly depend on the degrees of freedom of macro element

$$\begin{aligned} \int_{\Omega_{Me}} \mathbf{P}_M \cdot \delta \mathbf{F}_M dV &= \int_{\Omega_m} \mathbf{P}_m \cdot \delta \mathbf{F}_m dV \\ &= \sum_{e=1}^{n_{me}} \int_{\Omega_{me}} \mathbf{P}_m \cdot \delta \mathbf{F}_m dV \end{aligned} \quad (46)$$

Discretization of the micro mesh together with the variation of deformation gradient $\delta \mathbf{F}_m(\mathbf{p}_{me}(\mathbf{p}_{Me}, \lambda_m)) = \frac{\partial \mathbf{F}_m}{\partial \mathbf{p}_{me}} \frac{D \mathbf{p}_{me}}{D \mathbf{p}_{Me}} \delta \mathbf{p}_{Me}$ and standard GAUSS integration over the micro element domain Ω_{me} leads from (46) to the macro element residual \mathbf{R}_{Me} in a form

$$\mathbf{R}_{Me} = \sum_{e=1}^{n_{me}} \sum_{g \in G_e} w_{gp} \mathbf{R}_{Mg} \quad (47)$$

$$\mathbf{R}_{Mg} = J_\xi \mathbf{P}_m : \frac{\partial \mathbf{F}_m}{\partial \mathbf{p}_{me}} \frac{D \mathbf{p}_{me}}{D \mathbf{p}_{Me}} \quad (48)$$

where \mathbf{R}_{Mg} is a contribution to the macro element residual evaluated at the micro element GAUSS points. Differentiation of (48) leads to the macro element tangent matrix

$$\mathbf{K}_{Me} = \sum_{e=1}^{n_{me}} \sum_{g \in G_e} w_{gp} \mathbf{K}_{Mg} \quad (49)$$

$$\begin{aligned} \mathbf{K}_{Mg} &= \frac{\partial \mathbf{R}_{Mg}}{\partial \mathbf{p}_{Me}} = J_\xi \left(\frac{\partial \mathbf{P}_m}{\partial \mathbf{p}_{me}} \frac{D \mathbf{p}_{me}}{D \mathbf{p}_{Me}} : \frac{\partial \mathbf{F}_m}{\partial \mathbf{p}_{me}} \frac{D \mathbf{p}_{me}}{D \mathbf{p}_{Me}} \right. \\ &\quad \left. + \mathbf{P}_m : \left(\frac{\partial^2 \mathbf{F}_m}{\partial \mathbf{p}_{me}^2} \frac{D \mathbf{p}_{me}}{D \mathbf{p}_{Me}} \frac{D \mathbf{p}_{me}}{D \mathbf{p}_{Me}} + \frac{\partial \mathbf{F}_m}{\partial \mathbf{p}_{me}} \frac{D^2 \mathbf{p}_{me}}{D \mathbf{p}_{Me}^2} \right) \right) \end{aligned} \quad (50)$$

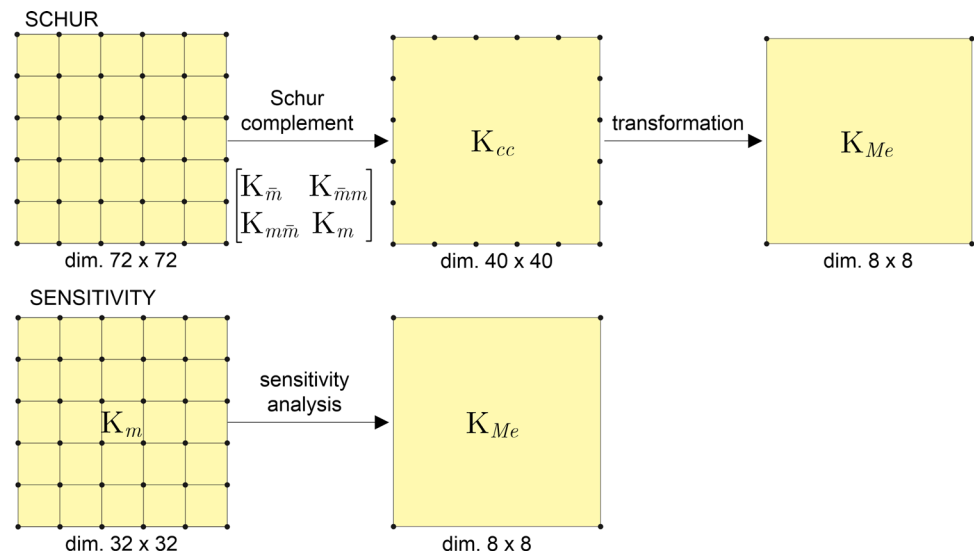
where again \mathbf{K}_{Mg} is a contribution to the macro element tangent evaluated at micro mesh GAUSS points.

The residual and tangent matrix are for each macro element obtained directly from the micro scale problem 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.

4.1.1 Sensitivity analysis based implementation of MIEL

Sensitivity analysis is required for the 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 (48) and (50). From (43) there follows a set of sensitivity parameters of the micro problem $\Phi = \Phi_{Me} = \mathbf{p}_{Me} = \bigcup_{j,k} u_{Mejk}$, and from (44) and (45) the components of velocity field ${}^I \mathbf{V} = \frac{\partial \bar{\mathbf{p}}_m(\Phi, \lambda_m)}{\partial \phi_I}$. 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 (44), the second derivatives are zero, and consequently the second order velocity fields are ${}^{IJ} \mathbf{V} = \mathbf{0}$.

Fig. 3 MIEL macro tangent matrix \mathbf{K}_{Me} ; above—Schur complement implementation and below—sensitivity based implementation



Other quantities required in two-level path-following algorithm Box 2 are then: macro level variables that are passed to macro level $\mathbf{S} = \mathbf{S}_{Me} = \mathbf{R}_{Me}$ (48) and total derivative $\frac{D\mathbf{S}}{D\Phi} = \frac{D\mathbf{S}_{Me}}{D\Phi_{Me}} = \mathbf{K}_{Me}$. For the numerically efficient implementation of (48) and (50), we also need ADB form of (48) and (50). From (14) ADB form of (48) and (50) leads to

$$\mathbf{R}_{Mg} = J_{\xi} \frac{\delta \mathbf{W}}{\delta \mathbf{p}_{Me}} \bigg|_{\mathbf{h}_{mg}=\text{const.}, \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}}=\mathbf{Y}_{\Phi}}, \quad (51)$$

$$\mathbf{K}_{Mg} = \frac{\delta \mathbf{R}_{Mg}}{\delta \mathbf{p}_{Me}} \bigg|_{\frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}}=\mathbf{Y}_{\Phi}, \frac{D^2\mathbf{p}_{me}}{D\mathbf{p}_{Me}^2}=\mathbf{Y}_{\Phi\Phi}}, \quad (52)$$

where $\mathbf{Y}_{\Phi} = \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}}$ and $\mathbf{Y}_{\Phi\Phi} = \frac{D^2\mathbf{p}_{me}}{D\mathbf{p}_{Me}^2}$ are first and second order sensitivities calculated and stored during the analysis.

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 elastoplastic 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 [18]. 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

$$\begin{bmatrix} \mathbf{K}_{\bar{m}} & \mathbf{K}_{\bar{m}m} \\ \mathbf{K}_{m\bar{m}} & \mathbf{K}_m \end{bmatrix} \cdot \begin{bmatrix} \Delta \bar{\mathbf{p}}_m \\ \Delta \mathbf{p}_m \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{\bar{m}} \\ \mathbf{0} \end{bmatrix}. \quad (53)$$

SCHUR complement of (53) 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 (44)], we can write

$$\bar{\mathbf{p}}_m = \mathbf{T} \cdot \mathbf{p}_{Me} \quad (54)$$

where \mathbf{T} is a transformation matrix (for details see [9]). The macro element residual and tangent matrix are then expressed by

$$\mathbf{R}_{Me} = \mathbf{T}^T \cdot \mathbf{R}_c \quad (55)$$

$$\mathbf{K}_{Me} = \mathbf{T}^T \cdot \mathbf{K}_{cc} \cdot \mathbf{T} \quad (56)$$

With \mathbf{R}_{Me} and \mathbf{K}_{Me} known, one can apply the algorithm presented in Box 2, 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 microstructure the calculation of the SCHUR complement inflicts high memory allocation and is time consuming. 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 noded elements in Fig. 3. 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 40×40 . To get macro element tangent matrix \mathbf{K}_{Me} with dimension 8×8 , additional transformations (55), (56) 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 (51), (52) over the micro mesh integration points.

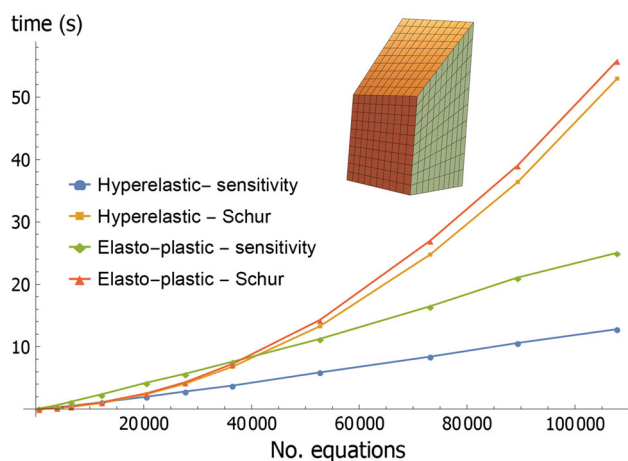


Fig. 4 Comparison of the computational time with respect to micro mesh density for two implementations of MIEL method

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. 4, the calculation time for the SCHUR complement and for the second order sensitivity analysis is 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 as defined in Sect. 1.1 and hyper-elastic based on hyper-elastic strain energy (2). 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.

4.2 FE² method

Standard two-level finite element homogenization approach FE² is appropriate for the problems where scales are separated far enough and are only weakly coupled, see [16]. The FE² method was already implemented using sensitivity analysis in [22,27], but without two interacting path-following schemes. Within the FE² approach we have one micro FE model, also called a representative volume element (RVE), at each macro mesh integration point as shown in Fig. 5. 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 ($\mathbf{P}_M \cdot \delta \mathbf{F}_M$) and macro level discretization of deformation gradient $\delta \mathbf{F}_M = \frac{\partial \mathbf{F}_M}{\partial \mathbf{p}_{Me}} \delta \mathbf{p}_{Me}$, the macro element residual leads to

$$\mathbf{R}_{Me} = \sum_{g \in G_e} w_{gp} \mathbf{R}_{Mg}, \quad (57)$$

$$\mathbf{R}_{Mg} = J_{\xi} \mathbf{P}_M : \frac{\partial \mathbf{F}_M}{\partial \mathbf{p}_{Me}}, \quad (58)$$

where the macro level first Piola–Kirchhoff stress tensor \mathbf{P}_M is obtained by averaging the micro level first Piola–Kirchhoff stress tensor $\mathbf{P}_M = \{\mathbf{P}_m\}$. 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. [16]) by applying first the prescribed displacements in the corners of RVE by

$$\bar{\mathbf{u}}_m = (\mathbf{F}_{Ms} + (\lambda_m - \lambda_{ms})(\mathbf{F}_M - \mathbf{F}_{Ms}) - \mathbf{I}) \mathbf{X}_m \quad (59)$$

where \mathbf{F}_{Ms} is macro deformation gradient at the end of the last macro step. The derivatives of (59) with respect to components of \mathbf{F}_M are given by

$$\frac{\partial \bar{u}_{mi}}{\partial F_{Mjk}} = \delta_{ij} (\lambda_m - \lambda_{ms}) X_{mk}. \quad (60)$$

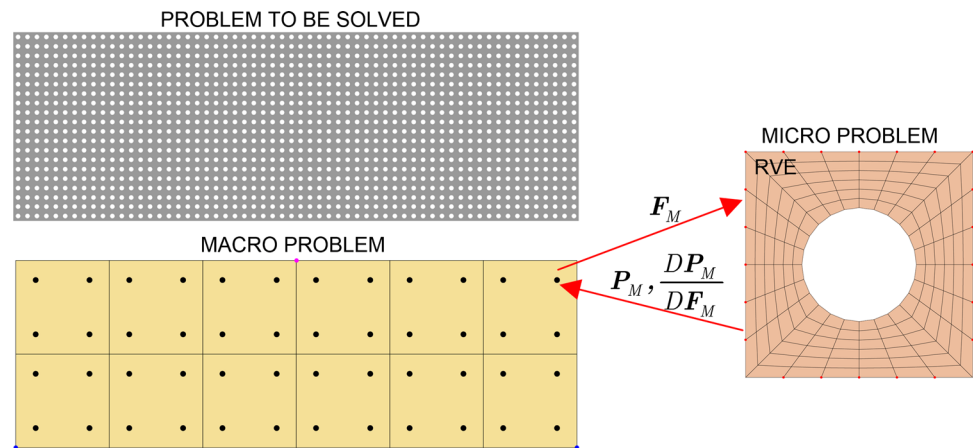
Thus, (59) defines the dependency $\bar{\mathbf{p}}_m = \bar{\mathbf{p}}_m(\mathbf{F}_M, \lambda_m)$ between the set of micro nodal unknowns with prescribed essential boundary condition $\bar{\mathbf{p}}_m$ and the macro deformation gradient \mathbf{F}_M . For the unconstrained boundary nodes, the periodicity of boundary conditions is adopted with the use of Lagrange multipliers (for details see [27]). 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 Sect. 3.

Differentiation of (48) then leads to the macro element tangent matrix

$$\mathbf{K}_{Me} = \sum_{g \in G_e} w_{gp} \mathbf{K}_{Mg} \quad (61)$$

$$\mathbf{K}_{Mg} = \frac{\partial \mathbf{R}_{Mg}}{\partial \mathbf{p}_{Me}} + \frac{\partial \mathbf{R}_{Mg}}{\partial \mathbf{P}_M} \frac{D \mathbf{P}_M}{D \mathbf{F}_M} \frac{\partial \mathbf{F}_M}{\partial \mathbf{p}_{Me}} \quad (62)$$

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

Fig. 5 FE^2 multi-scale scheme


4.2.1 Sensitivity analysis based implementation of FE^2

Sensitivity analysis is required for the evaluation of implicit dependency $\frac{DP_M}{DF_M}$ in (62). From (59) there follows a set of sensitivity parameters of micro problem

$$\Phi = \bigcup_{ij} F_{Mij} \quad (63)$$

and from (60) the components of velocity field $^I\mathbf{V} = \frac{\partial \mathbf{p}_m(\Phi, \lambda_m)}{\partial \Phi_I}$. 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 (59), 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 $\mathbf{S} = \mathbf{P}_M = \{\mathbf{P}_m\}$ and the total derivative $\frac{D\mathbf{S}}{D\Phi} = \left\{ \frac{\partial \mathbf{P}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{DF_M} \right\}$. 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 $\Phi_{Me} = \bigcup_{g \in G_e} \Phi^{(g)}$, 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 $\frac{D\mathbf{S}_{Me}}{D\Phi_{Me}} = \bigcup_{r \in \mathcal{M}_e} \left(\frac{D\mathbf{S}}{D\Phi} \right)^{(r)}$ where \mathcal{M}_e is a set of micro problems that corresponds to G_e . For the numerically efficient implementation of (58) and (62), we also need the ADB form of (58) and (62). From $\mathbf{P}_M \cdot \delta \mathbf{F}_M = \mathbf{S} \cdot \delta \mathbf{F}_M$ the ADB form of (58) and (62) leads to

$$\mathbf{R}_{Mg} = J_{\xi} \frac{\hat{\delta}(\mathbf{S} : \mathbf{F}_M)}{\hat{\delta} \mathbf{p}_{Me}} \bigg|_{\mathbf{S}=\text{const.}} \quad (64)$$

$$\mathbf{K}_{Mg} = \frac{\hat{\delta} \mathbf{R}_{Mg}}{\hat{\delta} \mathbf{p}_{Me}} \bigg|_{\frac{D\mathbf{S}}{DF_M} = \frac{D\mathbf{S}}{D\Phi}} \quad (65)$$

and

$$\frac{D\mathbf{S}}{D\Phi} = \left\{ \frac{\partial \mathbf{P}_m}{\partial \mathbf{p}_{me}} \frac{D\mathbf{p}_{me}}{DF_M} \right\} = \left\{ \frac{\hat{\delta} \mathbf{P}_m}{\hat{\delta} \mathbf{F}_M} \bigg|_{\frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}} = \mathbf{Y}_{\Phi}} \right\} \quad (66)$$

where $\mathbf{Y}_{\Phi} = \frac{D\mathbf{p}_{me}}{D\mathbf{p}_{Me}}$ are already calculated and stored first order sensitivities.

4.2.2 Schur complement based FE^2 implementation

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 [16], and it will not be repeated here. 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 thus, the advantages of using the sensitivity analysis are less pronounced than for MIEL. Note that the standard method is only consistently linearized for the problems that are path independent within a single macro step.

5 Numerical examples

Numerical examples were calculated using program packages AceGen and AceFEM [12]. Finite element user subroutines for primal and analytical first and second order sensitivity analyses were automatically derived, optimized and written in C language with the use of 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 AceFEM environment according to algorithm defined in Box 2. Intel MKL sparse linear algebra numerical library was used for the lin-

ear algebra operations (calculation of the SCHUR complement and the solution of linear systems of equations).

For 2D examples, nine noded, isoparametric, quadrilateral, plane strain elements are used integrated with 3×3 GAUSS integration, and for 3D examples, eight noded, isoparametric, hexahedral elements are used, integrated with $2 \times 2 \times 2$ GAUSS integration. For all examples, a finite strain elasto-plastic material model as described in Sect. 1.1 is used at the micro level.

The abbreviations used to indicate specific combination of methods solution procedures are structured as follows

method – n_M/n_m – *implementation*

where *method* can be MIEL or FE^2 , n_M is the number of macro steps or “Adaptive” for adaptive macro time stepping, n_m is the number of micro steps for each macro step or “Adaptive” for adaptive micro time stepping, and *implementation* is “Sens.” for sensitivity analysis based implementation, “SchurMMA” for the SCHUR complement based formulation implemented in Mathematica and “SchurMKL” for the SCHUR complement based formulation implemented with 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 theoretically the same SCHUR complement, the algorithm implemented in MKL performs perturbation of the zeros at the main diagonal resulting in slightly imprecise tangent matrix as shown and explained on examples in Sect. 5.2.

5.1 Validation of implementation of multi-scale algorithm

The first numerical example is a three-dimensional cantilever with clamped right and left end as shown in Fig. 6. Uniform pressure $p = 10$ in the vertical z direction was imposed at the top surfaces of the middle part of the cantilever. The dimensions of the cantilever are $12 \times 2.4 \times 2.4$. 3D, hexahedral elements are used at both levels. Material properties are $E = 21,000$, $\nu = 0.3$, $\sigma_{y0} = 24$ and $K_h = 100$. A homogeneous mesh is used at both levels, thus for the purpose of validation the micro level is uniform and no microstructure is present. The simulations were performed with adaptive time stepping at both levels. The displacements in the z direction of nodes on line \overline{AB} are presented for all simulations in Fig. 7. The extent of the plastic zone at the end of the simulation is shown in Fig. 6, where red color 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 the results obtained by single scale analysis with mesh $20 \times 4 \times 4$ with

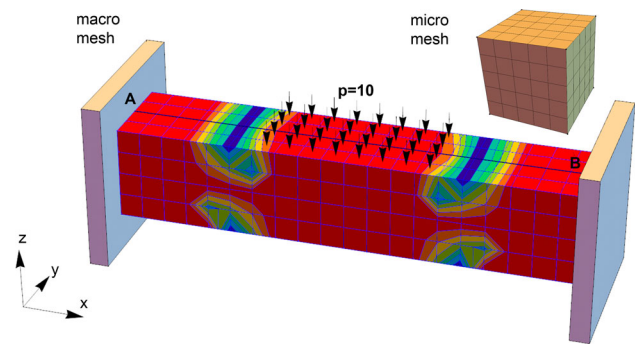


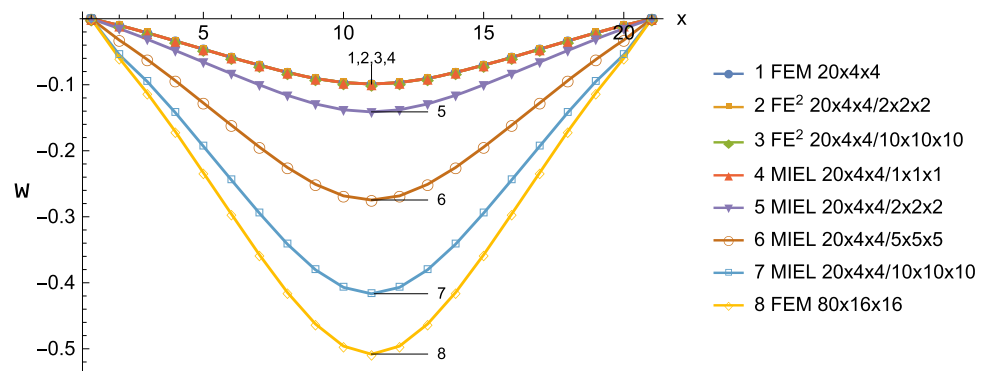
Fig. 6 Clamped cantilever with macro and micro mesh and enforced natural and essential boundary conditions

multi-scale analysis for macro mesh density $20 \times 4 \times 4$ and micro mesh densities $2 \times 2 \times 2$ and $10 \times 10 \times 10$. The results must be independent of micro mesh density due to the homogeneous micro mesh and exact enforcement of periodicity of micro mesh boundary conditions. Multi-scale results must also be exactly the same as single-scale results, which is shown in Fig. 7 (curves 1, 2 and 3). This verifies the FE^2 implementation.

For the MIEL method, the results of single scale simulation and multi-scale simulation can be exactly the same only for micro mesh density $1 \times 1 \times 1$. This is shown in Fig. 7 (curves 1, 4). This verifies also the MIEL implementation. With the change of micro mesh density 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 get close to single scale FEM solution obtained with the mesh $80 \times 16 \times 16$ (curve 8). This is the consequence of a better description of the deformation field over the domain of macro element, 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.

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

Fig. 7 Displacement in z direction of line AB**Table 1** Comparison of MIEL convergence rate for the last macro step

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

Table 2 Comparison of FE² convergence rate for last macro step

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

Table 1 shows that in the case when 1 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 exactly the same regardless of implementation. Sensitivity based implementation retains quadratic convergence also 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 Sect. 3.2 are not resolved after

each micro step, but only at the end of micro solution. This is again equivalent to implementation MIEL-10/5-SchurMKL. It shows that only a fully consistent sensitivity analysis ensures quadratic convergence of the overall MIEL algorithm.

Secondly, the convergence rate of FE² scheme was investigated in Table 2. The same conclusions can be drawn as for MIEL. Only fully consistent sensitivity analysis ensures quadratic convergence of the overall FE² algorithm. The last

Table 3 Effect of implementation on numerical efficiency of the FE^2 method

Implementation	Micro mesh	Normalized time	Total NR iterations	Total micro problems
FE^2 -10/1-Sens.	$5 \times 5 \times 5$	1.0	60	153 600
FE^2 -10/5-Sens.	$5 \times 5 \times 5$	3.6	59	151 040
FE^2 -10/1-SchurMKL	$5 \times 5 \times 5$	1.5	65	166 400
FE^2 -10/5-SchurMKL	$5 \times 5 \times 5$	8.8	136	348 160
FE^2 -10/1-Sens.	$10 \times 10 \times 10$	6.3	60	153 600
FE^2 -10/1-SchurMKL	$10 \times 10 \times 10$	8.5	67	171 520

two columns 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 as well as produces zeros at 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 on the main diagonal resulting in imprecise SCHUR complement. This imprecision is sufficient to alter, although not significantly, the convergence behavior. This case is shown in fourth column in Table 2, designated as FE^2 -10/1-SchurMKL.

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

The numerical efficiency of the two-level path-following iterative procedure is investigated on an example from Sect. 5.1. All simulations were performed on PC with Intel i9 2.8GHz, 16 Core processor and 128 GB RAM. Micro problems were solved in parallel on 14 cores. Mathematica was used only as a steering application for parallelization and the control of the iterative procedure, while all computationally intensive operations were performed with compiled C codes. The material model used is a 3D finite strain elasto-plastic model based on an exact exponential map (see e.g. [13]), 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 implementation on computational time is for FE^2 formulation presented in Table 3. An example introduced in Sect. 5.1 is solved with the macro mesh density $20 \times 4 \times 4$ in 10 load steps with a constant load increment $\Delta\lambda_M = 0.1$. The computational time normalized with respect to 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 FE^2 -10/1-Sens. formulation

took 1968.5 s of real time. The results are presented for the number of micro steps $n_m = 1$ and $n_m = 5$ and the density of the micro mesh $5 \times 5 \times 5$ and $10 \times 10 \times 10$. The first order sensitivity analysis based formulation is in all cases faster than the corresponding SCHUR complement based formulation. The loss of quadratic convergence of the FE^2 -10/5-SchurMKL formulation results in more iterations per load step, which is the most influencing factor. 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.

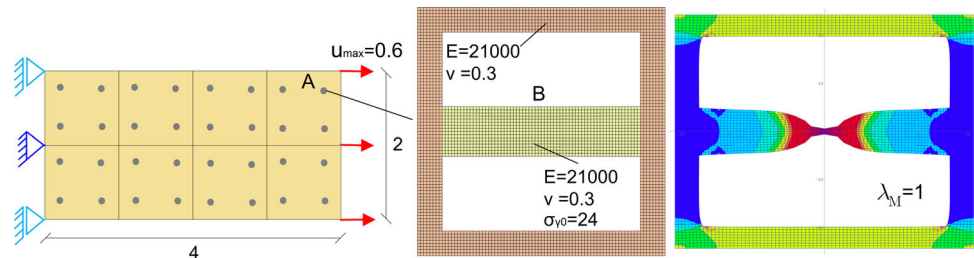
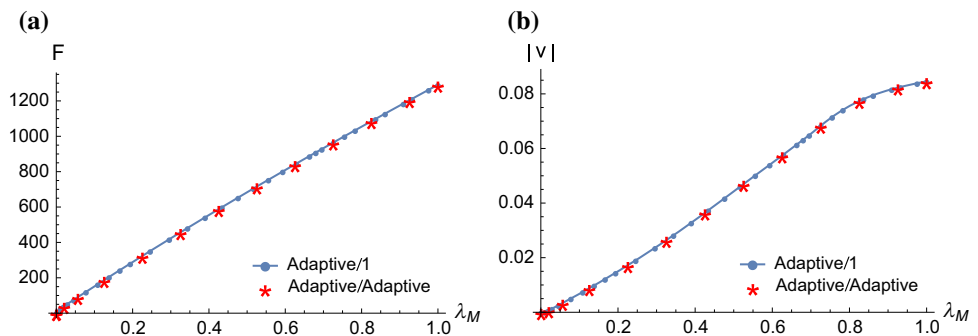
The effect of implementation, micro mesh density and material model on computational time is for the MIEL formulation presented in Table 4. The example introduced in Sect. 5.1 is in this case solved with the macro mesh density $10 \times 2 \times 2$ in 5 load steps with a constant load increment $\Delta\lambda_M = 0.2$. The computational time is normalized with respect to MIEL-5/1-Sens. formulation, which, for micro mesh $5 \times 5 \times 5$, took 37 s of real time. Two micro material models are considered: finite strain elasto-plastic model as defined in Sect. 1.1 and hyper-elastic model based on hyper-elastic strain energy (2). 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. 4, 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 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.

5.4 Effect of non-linearity of micro-structure

The example demonstrates how the use of a two-level path-following procedure improves the numerical efficiency of multi-scale simulation in the case of highly nonlinear

Table 4 Effect of implementation and material model on normalized time for the MIEL method

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

Fig. 8 Uni-axial test of macro mesh and geometry together with RVE mesh and geometry and deformed RVE**Fig. 9** Uni-axial test: **a** Horizontal residual force F and **b** vertical displacement in point B

microstructure response and relatively monotonic response of the macro structure. To demonstrate that, a 2D, plane strain, uni-axial test is simulated using 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. 8. The macro domain is discretized with 4×2 macro elements and displacement $u_{max} = 0.6$ is prescribed at the end. The RVE of periodic micro-structure is composed of hyper-elastic rim with material properties $E = 21,000$, $\nu = 0.3$ and a narrow elasto-plastic incision with properties $E = 21,000$, $\nu = 0.3$, $\sigma_{y0} = 24$. The incision has an additional small imperfection in the middle. At RVE level Q2, nine noded, isoparametric, quadrilateral, plane strain elements were used, to avoid the locking effect. At a certain load level, a strongly nonlinear process of necking of the incision 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 also 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

needs 13 macro load steps, whereas for the FE^2 -Adaptive/1-Sens. approach 33 macro load steps are done. Figure 9a shows the macro reaction force F and Fig. 9b the absolute contraction at point B at the micro level with respect to global load factor λ_M for both cases. The response curves are almost the same for both cases. Thus, an efficient solution of strongly nonlinear problems requires two level adaptive time stepping procedures where the maximum size of load increments at the micro level determines the overall efficiency of the simulation.

5.5 Effect of path dependency of microstructure

The example demonstrates how the use of 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 evolution equations depends on the micro step size, thus limiting the size of micro load steps. This, for the $n_m = 1$ case, limits also the macro load step size similarly 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.

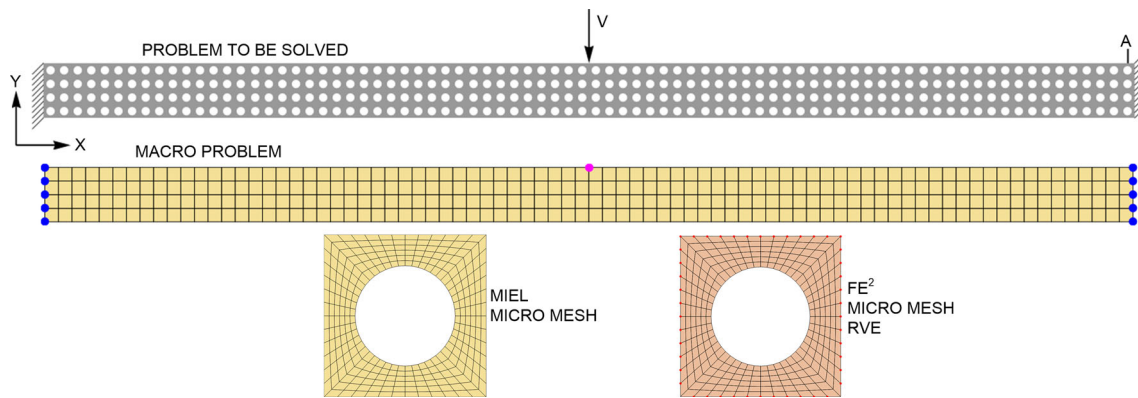


Fig. 10 Macro geometry

Fig. 11 E_{xy} with respect to $\Delta\lambda_{Mmax}$ and number of micro steps for the MIEL scheme

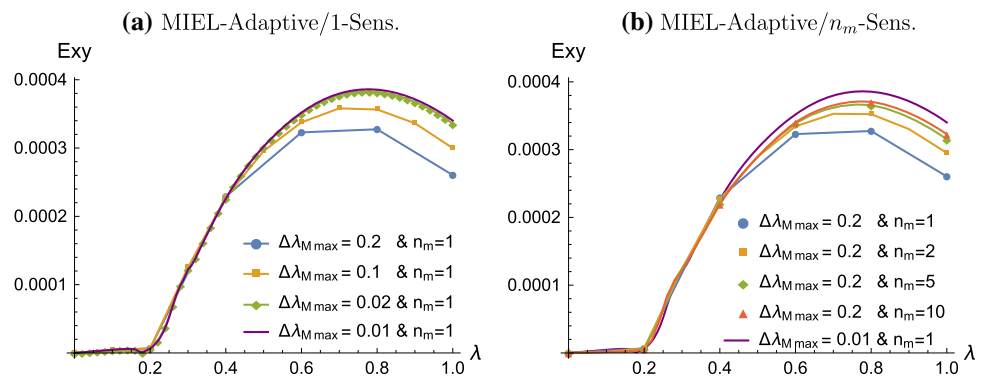
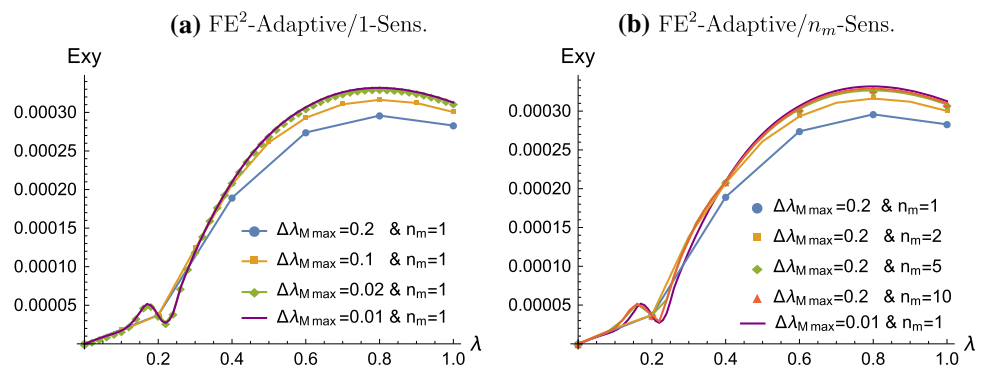


Fig. 12 E_{xy} with respect to $\Delta\lambda_{Mmax}$ and number of micro steps for the FE² scheme



Long clamped beam with dimensions 20×1 and with macro mesh division 80×4 has prescribed vertical displacement $v_{max} = 0.25$ at the middle, as shown in Fig. 10. The beam is perforated with 320 perforations with the radius that gives 30% perforation of the beam. Perforations are evenly distributed in a way that each perforation is placed at the center of the corresponding macro element. Two cases are investigated. In the first case, a MIEL multi-scale computational scheme is employed. Due to the even distribution of perforations, all the MIEL micro meshes look the same, as shown in Fig. 10a. In the second case, infinitely small perforations were assumed with the same 30% perforation ratio as in the first case. The second case is simulated by the FE² scheme with RVE, as depicted in Fig. 10b. The RVE mesh is identical

to the MIEL micro mesh due to the evenly distributed perforations. Material properties of the microstructure are in both cases $E = 21,000$, $\nu = 0.3$, $\sigma_{y0} = 24$, $K_h = 21$, $R_\infty = 12$ and $\delta = 30$. The value of strain tensor component E_{xy} in point A is for various solution strategies compared for the MIEL scheme in Fig. 11 and for the FE² scheme in Fig. 12.

In Fig. 11a the response curve $E_{xy}^A(\lambda_M)$ is shown for the MIEL-Adaptive/1-Sens. approach with different prescribed maximal size of macro load step $\Delta\lambda_{Mmax}$, adaptive time stepping at macro level and one micro step per each macro step. Converged solution is achieved for $\Delta\lambda_{Mmax} = 0.01$. Secondly, Fig. 11b displays the results for fixed $\Delta\lambda_{Mmax} = 0.2$ and 1, 2, 5 and 10 micro steps per each macro step. It can be seen that 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. 11b.

For the FE^2 scheme the results are compared in the same way as for the MIEL scheme. In Fig. 12a the response curve $E_{xy}^A(\lambda_M)$ is shown for FE^2 -Adaptive/1-Sens. with respect to the prescribed maximal size of macro load step $\Delta\lambda_{Mmax}$ and in Fig. 12b for $\Delta\lambda_{Mmax} = 0.2$ and with different number of micro steps. Adaptive time stepping at macro level is used 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 to capture also fine details of 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 $E_{xy}^A(\lambda_M)$ is different for the MIEL and the FE^2 scheme.

6 Conclusions

In the paper a generalized essential boundary condition sensitivity analysis based implementation of FE^2 and MIEL multi-scale methods 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 the scales. The paper shows that the derivation of algorithmic macroscopic tangent requires for the FE^2 multi-scale method the first order essential boundary condition sensitivity analysis and for the MIEL multi-scale method the second order essential boundary condition sensitivity analysis. 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. It has been shown in the paper that ADB brings several advantages. The first advantage is the ability to naturally introduce a fully consistently linearized two-level path-following algorithm as a solution algorithm for the multi-scale modeling. Sensitivity analysis allows that each macro step can be followed by an arbitrary number of micro substeps while retaining quadratic convergence of the overall solution algorithm. Using examples, it has been shown that this cannot be achieved with the standard SCHUR complement based methods. Additionally, the method completely avoids the evaluation of the SCHUR complement of the micro tangent matrix as numerically demanding mathematical operation which, especially for the MIEL multi-scale methods, results in large dense matrices.

The advantages of the sensitivity analysis based implementation in comparison with the traditional one were recognized and verified on numerical examples. The convergence of results with respect to the size of the macro load step and the number of substeps at the micro level was investigated. With additional micro steps, the accuracy of the global response can be improved without costly additional 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 it still limits the size of the macro load steps. Similarly, a strongly path-dependent micro material model requires small micro load steps that limit within the standard implementation also the size of the macro load step. This restriction is again relaxed with the introduction of the two-level path-following algorithm.

Acknowledgements The financial support for this work was obtained from the Slovenian Research Agency within the PhD Grant Agreement (Annex No: 630-34/2015-7).

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

- Castelletto N, Hajibeygi H, Tchelepi HA (2017) Multiscale finite-element method for linear elastic geomechanics. *J Comput Phys* 331:337–356
- Clasen H, Hirschberger CB, Korelc J, Wriggers P (2013) FE^2 -homogenization of micromorphic elasto-plastic materials. In: Computational plasticity XII: proceedings of the XII international conference on computational plasticity—fundamentals and applications
- Efendiev Y, Hou T (2008) Multiscale finite element methods: theory and applications. Springer, Dordrecht
- El Said E, Daghia F, Ivanov D, Hallett SR (2018) An iterative multi-scale modelling approach for nonlinear analysis of 3D composites. *Int J Solids Struct* 132–133:42–58
- Feyel F (1999) Multiscale FE^2 elastoviscoplastic analysis of composite structures. *Comput Mater Sci* 16:344–354
- Geers MGD, Kouznetsova V, Brekelmans WAM (2010) Multi-scale computational homogenization: trends and challenges. *J Comput Appl Math* 234:2175–2182
- Griewank A (2000) Evaluating derivatives: principles and techniques of algorithmic differentiation. SIAM, Philadelphia
- Hou TY, Wu X-H (1997) A multiscale finite element method for elliptic problems in composite materials and porous media. *J Comput Phys* 134:169–189
- Ibrahimbegović A, Marković D (2003) Strong coupling methods in multi-phase and multi-scale modeling of inelastic behavior of heterogeneous structures. *Comput Methods Appl Mech Eng* 192:3089–3107
- Korelc J (1997) Automatic generation of finite-element code by simultaneous optimization of expressions. *Theoret Comput Sci* 187(1–2):231–248

11. Korelc J (2009) Automation of primal and sensitivity analysis of transient coupled problems. *Comput Mech* 44(5):631–649
12. Korelc J (2018) AceFEM and AceGen user manuals. <http://sympch.fgg.uni-lj.si/>, 6.902 edition
13. Korelc J, Stupkiewicz S (2014) Closed-form matrix exponential and its application in finite-strain plasticity. *Int J Numer Methods Eng* 98:960–987
14. Korelc J, Wriggers P (2016) *Automation of finite element methods*. Springer, Cham
15. Korelc J, Zupan N (2018) Unified approach to sensitivity analysis based automation of multiscale modelling. In: Soric J, Wriggers P, Allix O (eds) *Multiscale modeling of heterogeneous structures* (Lecture Notes in Applied and Computational Mechanics). Springer, Berlin
16. Kouznetsova V, Brekelmans WAM, Baaijens FPT (2001) An approach to micro-macro modeling of heterogeneous materials. *Comput Mech* 27:37–48
17. Lamut M, Korelc J, Rodič T (2011) Multiscale modelling of heterogeneous materials. *Mater Tehnol* 45:421–426
18. Markovič D, Ibrahimbegović A (2004) On micro-macro interface conditions for micro scale based fem for inelastic behavior of heterogeneous materials. *Comput Methods Appl Mech Eng* 193:5503–5523
19. Markovič D, Niekamp R, Ibrahimbegović A, Matthies HG, Taylor RL (2005) Multiscale modeling of heterogeneous structures with inelastic constitutive behaviour. *Eng Comput* 22:664–683
20. Mathematica (2018). www.wolfram.com
21. Michaleris P, Tortorelli DA, Vidal CA (1994) Tangent operators and design sensitivity formulations for transient non-linear coupled problems with applications to elastoplasticity. *Int J Numer Methods Eng* 37:2471–2499
22. Miehe C, Bayreuther CG (2007) On multiscale FE analyses of heterogeneous structures: from homogenization to multigrid solvers. *Int J Numer Methods Eng* 71(10):1135–1180
23. Miehe C, Schotte J, Schröder J (1999) Computational micro-macro transitions and overall moduli in the analysis of polycrystals at large strains. *Comput Mater Sci* 16:372–382
24. Niekamp R, Markovič D, Ibrahimbegović A, Matthies HG, Taylor RL (2009) Multiscale modelling of heterogeneous structures with inelastic constitutive behavior. *Eng Comput* 26:6–28
25. Loehnert S, Mueller-Hoeppel DS (2008) Multiscale methods for fracturing solids. In: Reddy BD (ed) *IUTAM symposium on theoretical, computational and modelling aspects of inelastic media*. IUTAM BookSeries, vol 11. Springer, Dordrecht
26. Simo JC, Hughes TJR (1998) *Computational inelasticity*. Springer, New York
27. Solinc U, Korelc J (2015) A simple way to improved formulation of FE2 analysis. *Comput Mech* 56:905–915
28. Temizer I, Wriggers P (2008) On the computation of the macroscopic tangent for multiscale volumetric homogenization problems. *Comput Methods Appl Mech Eng* 198:495–510
29. Terada K, Hori M, Kyoya N, Kikuchi N (2000) Simulation of multiscale convergence in computational homogenization approaches. *Int J Solids Struct* 37:2285–2311
30. Zohdi TI, Wriggers P (2008) *An introduction to computational micromechanics*. Springer, Berlin

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.