V International Conference on Computational Methods for Coupled Problems in Science and Engineering
COUPLED PROBLEMS 2013
S. Idelsohn, M. Papadrakakis and B. Schrefler (Eds)

AUTOMATIC DIFFERENTIATION BASED FORMULATION OF COUPLED PROBLEMS

B. HUDOBIVNIK* AND J. KORELC †

* †Faculty of Civil and Geodetic Engineering (UL FGG)
University of Ljubljana
Jamova 2, 1000 Ljubljana, Slovenia
Email: * blaz.hudobivnik@fgg.uni-lj.si
† joze.korelc@fgg.uni-lj.si , web page: http://www3.fgg.uni-lj.si/

Key words: Automatic Differentiation, Symbolic environment, Coupled Problems, Multiphysics Problems, Formulation Approach

Abstract. Our work will show that complex transient coupled problems can be formulated and solved effectively with AceGen and AceFEM using an automatic differentiation based formulation (ADB-formulation). From scalar pseudo-potential function consistent tangent matrix for strongly coupled problems can be derived, leading to quadratically convergent Newton-Raphson type procedure. Another problem considered is the implementation of finite element. Typically, all equations are written inside a single finite element and a single pseudo-potential is defined. Such implementation is efficient but rigid, therefore, a different implementation was considered. Within the second approach we wrote a separate finite element for each field, but in a way that quadratic convergent Newton-Raphson procedure is preserved. The paper presents examples where *unified* and *field-by-field* implementations are compared according to computational efficiency. The results show that with increasing ratio between the complexity of constitutive equations and discretization, generated code size and evaluation time of implementations become comparable.

1 INTRODUCTION

With the use of advanced software tools and technologies it is possible to reach automatisation of finite element method (FEM)[3]. Different approaches and tools have been developed which enable more effective problem solving. For the formulation of coupled problems one of the most important software tools is automatic differentiation [2]. The automatic differentiation procedure (AD) has two possible modes. The first is forward AD and the second backward AD. Numerical cost of the first is proportional to number of independent variables, and numerical cost of the second to the number of functions

differentiated [3]. The result of AD procedure is called "computational derivative" and is written as $\frac{\hat{\delta} f(\mathbf{a})}{\hat{\delta}\mathbf{a}}$. To establish a relation between different types of differentiations and result of automatic differentiation, definition of differentiating exceptions is introduced. For example, formulation $\left.\frac{\hat{\delta} f(\mathbf{a},\mathbf{b}(\mathbf{a}))}{\hat{\delta}\mathbf{a}}\right|_{\frac{D\mathbf{b}}{D\mathbf{a}}=\mathbf{0}}$ determines that the "real" dependence $\mathbf{b}(\mathbf{a})$ defined by algorithm is ignored and $\mathbf{b}$ is taken as constant.

2 ADB FORMULATION OF GENERAL WEAK FORM AND PSEUDO-POTENTIAL FORM

For solving physical problems with FEM we need a weak form of differential equation to describe physical problem. Let us define a general weak form in a shape of

$$\int_{\Omega} \delta\mathbf{a}(\mathbf{p}) \cdot \mathbf{b}(\mathbf{p}) d\Omega + ... = 0 \tag{1}$$

Where $\mathbf{a}$ and $\mathbf{b}$ are tensors of arbitrary order, and $\delta\mathbf{a}$ is directional derivative of variation of tensor $\mathbf{a}$. Because $\delta\mathbf{a}$ is a fictive quantity, we cannot directly apply the automatic differentiation procedure. Thus it needs to be written in discretized form. Let $\mathbf{p}$ be a vector of unknown quantities of the problem and $\delta\mathbf{a} = \frac{\partial \mathbf{a}(\mathbf{p})}{\partial \mathbf{p}} \delta\mathbf{p}$. Discretized form of general weak form is then written as

$$\int_{\Omega} \delta\mathbf{a}(\mathbf{p}) \cdot \mathbf{b}(\mathbf{p}) d\Omega + ... = \sum_{i=1}^{n_{t_p}} \left(\int_{\Omega} \frac{\delta\mathbf{a}(\mathbf{p})}{\delta p_i} \cdot \mathbf{b}(\mathbf{p}) d\Omega \right) \delta p_i + ... = 0 \tag{2}$$

From (2) follows n_{t_p} algebraic equations $\mathbf{R}$, which can be solved with standard methods, e.g. Newton-Raphson method. If the partial derivative (2) is replaced by computational derivative, the ADB formulation of weak form is obtained as follows

$$\mathbf{R} = \int_{\Omega} \frac{\delta\mathbf{a}(\mathbf{p})}{\delta\mathbf{p}} \cdot \mathbf{b}(\mathbf{p}) d\Omega + ... = \int_{\Omega} \frac{\hat{\delta}\mathbf{a}(\mathbf{p})}{\hat{\delta}\mathbf{p}} \cdot \mathbf{b}(\mathbf{p}) d\Omega + ... = \mathbf{0} \tag{3}$$

Term $\frac{\hat{\delta}\mathbf{a}(\mathbf{p})}{\hat{\delta}\mathbf{p}}$ is not appropriate for backward differentiation, because cost of differentiation grows linearly with the number of components of tensor $\mathbf{a}$. The uncontrolled growth of derived expressions can also occur [3]. Formulation (3) needs to be written in appropriate shape, so that as little as possible scalar functions are differentiated. To accomplish that, scalar product of tensors $\mathbf{a}$ and $\mathbf{b}$ is introduced, so-called pseudo-potential in the shape of $W(\mathbf{p}) = \mathbf{a}(\mathbf{p}) \cdot \mathbf{b}(\mathbf{p})$. Result is so-called ADB formulation of discretized weak form:

$$\mathbf{R} = \int_{\Omega} \left.\frac{\hat{\delta}(\mathbf{a}(\mathbf{p}) \cdot \mathbf{b}(\mathbf{p}))}{\hat{\delta}\mathbf{p}}\right|_{\frac{D\mathbf{b}}{D\mathbf{p}}=\mathbf{0}} d\Omega + ... = \int_{\Omega} \left.\frac{\hat{\delta} W(\mathbf{p})}{\hat{\delta}\mathbf{p}}\right|_{\frac{D\mathbf{b}}{D\mathbf{p}}=\mathbf{0}} d\Omega + ... = \mathbf{0} \tag{4}$$

In equation (4) the introduced differentiation exception $\frac{D\mathbf{b}}{D\mathbf{p}} = \mathbf{0}$ assures that automatic differentiation returns equations of problem, which correspond to weak form (3). In the formulation of finite element method, the contribution of individual element $\mathbf{R}_e$ to the global residual $\mathbf{R}$ is obtained by numerical integration over element domain, $\mathbf{R}_e = \sum_{g=1}^{n_g} w_g \mathbf{R}_g$, where w_g is Gauss weight and $\mathbf{R}_g$ is residual calculated in each integration point and is written as

$$\mathbf{R}_g = J_g \left. \frac{\hat{\delta} W(\mathbf{p})}{\hat{\delta} \mathbf{p}} \right|_{\frac{D\mathbf{b}}{D\mathbf{p}} = \mathbf{0}} \tag{5}$$

where J_g is Jacobian determinant.

2.1 General implementation of coupled problems in environments AceGen/AceFEM

Let us take coupled problem, which is defined by n_c unknown fields $\boldsymbol{\phi} = \{\phi_1, ..., \phi_{n_c}\}$. According to FEM, ϕ_i is interpolated on element domain with the use of interpolation functions N_j: $\phi_i^h = \sum_{j=1}^{n_n} N_j p_{e,j}^i$, where $p_{e,j}^i$ is nodal the unknown of i-th field and j-th node in e-th finite element and n_n the number of element nodes. The number of all unknowns of finite element is then $n_p = \sum_{i=1}^{n_c} n_{p_i}$, where n_{p_i} is the number of unknowns used for discretisation of i-th field ϕ_i. That means the vector $\mathbf{R}_g$, which represents the contribution of g-th integration point to residual vector of e-th element $\mathbf{R}_e$, has n_p components and tangent matrix $\mathbf{K}_g$ has n_p^2 components. Consequently, the size of symbolically generated code of element grows with the square of the number of unknowns of the problem, which can become unmanageable with the large number of fields of the element. Complexity of the problem can be reduced in multiple ways.

Instead of explicit expressions for all components of $\mathbf{R}_g$ and $\mathbf{K}_g$, we can generate program code only for characteristic i-th component of residual ($R_{g,i}$) and characteristic i,j-th component of tangent matrix ($K_{g,i,j}$) (as described in [4]).

Additionally, we can generate equations that correspond to different fields or sets of fields of the problem, separately in separated finite elements, and later join them together when global residual and global tangent matrix are being assembled. With this in mind, a subset $\mathbf{G}_K \subseteq \{1, ..., n_c\}$ is defined. It determines fields $\boldsymbol{\phi}^{(K)} = \{\phi_i : i \in \mathbf{G}_K\}$, and their corresponding equations will be formulated inside K-th element source code. The unknowns of K-th source code are $\mathbf{p}_e^{(K)} = \bigcup_{i \in \mathbf{G}_K} \mathbf{p}_e^i$, where $\mathbf{p}_e^i$ is the vector of unknowns used to discretize field ϕ_i, $\mathbf{p}_e = \bigcup_{K=1}^{n_G} \mathbf{p}_e^{(K)}$ is the vector of all unknowns on finite elements level and n_G is the number of subsets of $\boldsymbol{\phi}$. Residual and stiffness matrix belonging to subset $\mathbf{G}_K$ are defined as: $\mathbf{R}_g^{(K)} = \left\{\mathbf{R}_g^i : i \in \mathbf{G}_K\right\}$ and $\mathbf{K}_g^{(K)} = \left[\mathbf{K}_g^{i,j} : i \in \mathbf{G}_K \wedge j \in \{1,...,n_c\}\right]$. Accordingly $\mathbf{R}_g^{(K)}$ has dimension $n_p^{(K)} = \sum_{i \in \mathbf{G}_K} n_{p_i}$ and $\mathbf{K}_g^{(K)}$ has dimension $n_p^{(K)} \times n_p$. Thus, $\mathbf{K}_g^{(K)}$ is not a square matrix.

According to ADB formulation of weak form shown in previous chapter (2), pseudo-potential of K-th element is defined as

$$W_g^{(K)} = \sum_{l \in \mathbf{G}_K} W_g^{(l)}(\mathbf{p}_e) = \sum_{l \in \mathbf{G}_K} \sum_m \mathbf{a}_m^{(l)} \cdot \mathbf{b}_m^{(l)}, \tag{6}$$

where m goes through a number of terms of weak form that correspond to the l-th field. Residual and tangent matrix in g-th integration point of K-th element source code are given as

$$\mathbf{R}_g^{(K)} = J_g \left. \frac{\hat{\delta} W_g^{(K)}(\mathbf{p}_e)}{\hat{\delta} \mathbf{p}_e^{(K)}} \right|_{\frac{D\mathbf{b}_m^{(l)}}{D\mathbf{p}_e^{(K)}} = \mathbf{0},\ \forall l \in \mathbf{G}_K \wedge \forall m} \quad \text{and} \quad \mathbf{K}_g^{(K)} = \frac{\hat{\delta} \mathbf{R}_g^{(K)}}{\hat{\delta} \mathbf{p}_e} \tag{7}$$

Algorithm 1 ADB formulation of K-th element source code

for $g := 1$ **to** n_g **step** 1 **do** ▷ Loop over elements integration points

$J_g := \det\left(\frac{\delta \mathbf{X}}{\delta \boldsymbol{\Xi}}\right)$ ▷ Calculation of Jacobian determinant

$\mathbf{b}_m^{(l)}(\mathbf{p}_e) := \ldots$ ▷ Definition of auxiliary functions $\mathbf{b}_m^{(l)}$, which will be constant when pseudo-potential is differentiated

$W_g^{(K)} := \sum_{l \in \mathbf{G}_K} \sum_m \mathbf{a}_m^{(l)} \cdot \mathbf{b}_m^{(l)}$ ▷ pseudo-potential is defined as sum of scalar tensors $\mathbf{a}_m^{(l)}$ and $\mathbf{b}_m^{(l)}$.

$\mathbf{R}_g^{(K)} = J_g \left. \frac{\hat{\delta} W_g^{(K)}(\mathbf{p}_e)}{\hat{\delta} \mathbf{p}_e^{(K)}} \right|_{\frac{D\mathbf{b}_m^{(l)}}{D\mathbf{p}_e^{(K)}} = \mathbf{0},\ \forall l \in \mathbf{G}_K \wedge \forall m}$ ▷ Differentiation with respect to unknowns of k-th set with appropriate exceptions

$\mathbf{K}_g^{(K)} = \frac{\hat{\delta} \mathbf{R}_g^{(K)}}{\hat{\delta} \mathbf{p}_e}$ ▷ Differentiation with respect to all unknowns $\mathbf{p}_e$ of the problem

end for

In general we can separate formulations into two groups: formulations where all equations are written inside one element (unified approach) and formulation where equations for each field or group of fields are formulated in separated elements (field-by-field approach). In the first example it applies $n_G = 1$. $\mathbf{R}_g$ and $\mathbf{K}_g$ are evaluated in two generic formulas R_{g_i} and $K_{g_{i,j}}$, which are both written inside one element. In the second case physically separated element is generated for each set $\boldsymbol{\phi}^{(K)}$. Each element has $n_p^{(K)}$ unknowns, called primary variables of element $\mathbf{p}_e^{i \in \mathbf{G}_K}$. In addition, element has access to other variables $\mathbf{p}_e^{i \in \{1,\ldots,n_c\} \setminus \mathbf{G}_K}$ through secondary additional nodes. Because there are n_G residuals and n_G tangent matrices, we need to generate $2 \times n_G$ generic formulas in total to calculate residual and tangent matrix in all elements of sets. These generic formulas are usually simpler than the formulas in the first case, but at the same time some quantities, e.g. interpolation functions, need to be calculated several times.

3 Thermo-Hydro-Mechanical coupled problem

3.1 Weak form of THM coupled problem

At THM formulation porous medium is used, so such material model must be used that account for solid, water and air. Labels w and s relate to liquid and solid material

phases. Unlabelled material quantities refer to effective or averaged quantities of one-phase medium $\phi = \sum n^i \phi^i$, where n^i is volume fraction and ϕ^i material quantity of phases w and s. Weak form of balance equation for the case of NeoHooke type large strain material [8, 7, 5] can be given with equation:

$$\int_\Omega \delta \boldsymbol{E} : \boldsymbol{S} d\Omega - \int_\Omega \delta \boldsymbol{E} : \boldsymbol{S}^{p_w} d\Omega - \int_\Omega \rho \delta \boldsymbol{u} \cdot \boldsymbol{g} d\Omega - \int_{\Gamma_u^q} \delta \boldsymbol{u} \cdot \bar{\boldsymbol{t}} d\Gamma_u^q = 0 \tag{8}$$

Where $\boldsymbol{S}$ is total second Piola-Kirchoff stress tensor, $\boldsymbol{S}^{p_w}$ volumetric second Piola-Kirchoff stress tensor caused by water pore pressures p_w, calculated as $\boldsymbol{S}^{p_w} = -\alpha_b J \boldsymbol{C}^{-1}(p_w - p_0)$, where J is the jacobian determinant, $\boldsymbol{C}$ the right Cauchy-Green deformation tensor and α_b Biot's coefficient. $\boldsymbol{E}$ is Green-Lagrange strain tensor referring to initial configuration. Operator "**:**" denotes matrix product $\mathbf{A} : \mathbf{B} = tr(\mathbf{A}^T\mathbf{B}) = tr(\mathbf{A}\mathbf{B}^T)$, also known as Frobenius inner product or matrix contraction. $\mathbf{I}$ is identity matrix of dimension 3×3. Because plasticity is considered, hyper-elastic constitutive equations must be supplemented by additional non-linear coupled equations, forming relation between time dependent history variables $\mathbf{h}$ and constitutive equations. The constitutive model considering termo-plasticity is taken from [7] and suplemented with equatons of effective stresses of [5].

Weak form of differential equation of non-stationary heat flow is given by [6] and contribution to temperatures from mechanical work by [7], and is written:

$$\begin{aligned} &\int_\Omega (\rho c) \frac{\partial T}{\partial t} \delta T d\Omega + \int_\Omega k_T \nabla \delta T \cdot \nabla T d\Omega - \int_\Omega \left(Q + Q_w\right) \delta T d\Omega \\ &+ \int_\Omega \delta T \left(-\mathcal{D}_{mech} + \mathcal{H}\right) d\Omega - \int_{\Gamma_T^q} \delta T \left(q^T + \alpha_c \left(T - T_\infty\right)\right) d\Gamma_T^q = 0 \end{aligned} \tag{9}$$

Where T is temperature, k_T effective heat conductivity, Q internal heat source, q^T surface heat source and (ρc) averaged heat capacity, where c is specific heat capacity and ρ density. Term $\alpha_c \left(T - T_\infty\right)$ accounts for heat convection and radiation on surface. Therefore, temperature on the surface tends towards temperature T_∞, where α_c is pseudo convective coefficient. Conversion of mechanical elastic and plastic work into heat is covered by mechanical dissipation $\mathcal{D}_{mech}$ and elastic-plastic structural heating $\mathcal{H}$, given by [7]. Term $Q_w = -\rho^w c^w \frac{k_w}{\mu^w} \left(-\nabla p_w + \rho^w g\right) \cdot \nabla T$ is added to inner heat source q, to capture the influence of liquid pressures.

Weak form of water flow on porous medium is taken from [6]. It is obtained from mass balance equation for compressible liquid water. The used governing equation of one-phase liquid flow in saturated porous medium is written as:

$$\int_\Omega \delta p_w \left(\frac{\alpha_b - n}{K^s} + \frac{n}{K^w}\right) \frac{\partial p_w}{\partial t} d\Omega + \int_\Omega \frac{k_w}{\mu^w} \nabla \delta p_w \cdot \nabla p_w d\Omega + \int_\Omega \delta p_w \alpha_b tr\left(\nabla \frac{\partial u}{\partial t}\right) d\Omega$$
$$- \int_\Omega \delta p_w \alpha \frac{\partial T}{\partial t} d\Omega - \int_\Omega \frac{k_w}{\mu^w} \rho^w \nabla \delta p_w \cdot \boldsymbol{g} d\Omega + \int_{\Gamma_w^q} \delta p_w \frac{q_w}{\rho_w} d\Gamma_w^q = 0 \tag{10}$$

Where p_w is water pressure in medium, n porosity of medium, K^w bulk modulus of water, k_w is intrinsic permeability and μ^w dynamic viscosity of liquid. Term $\int_\Omega \delta p_w \alpha_b tr(\nabla \frac{\partial u}{\partial t}) d\Omega$ in equation (10) contributes to change in pressures due to speed of deformation and term $-\int_\Omega \delta p_w \alpha_t \frac{\partial T}{\partial t} d\Omega$ contributes to pressure changes due to temperature expansion of medium and liquid, where α_t is average thermal expansion coefficient of medium.

3.2 Implementation of THM finite element

Weak form of coupled thermo-hydro-mechanical problem is defined with equations (8), (9) and (10). Variables of the problem are chosen as $\boldsymbol{\phi} = \{u, v, p_w, T\}$, where u, v and w are displacements, T temperature of medium and p_w pressure of liquid in pores. For this paper we will consider three different implementations of THM finite element shown on example of 4-node isoparametric quadrilateral element (Figure 1).

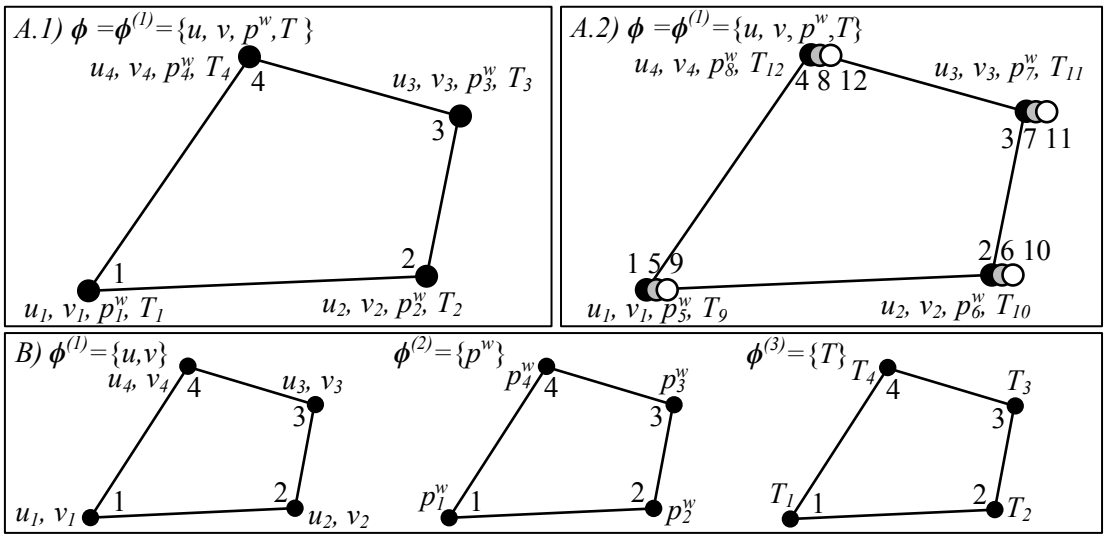

Figure 1: Three different implementations of finite elements considered (A.1, A.2 and B)

A.1) First standard finite element implementation is chosen with four nodes and all equations defined in a single element. Vector of unknown scalar fields is therefore $\boldsymbol{\phi} = \boldsymbol{\phi}^{(1)} = \{u, v, p_w, T\}$. Pseudo-potential is written as a sum of pseudo potentials

of physical fields $W = W^u + W^{p_w} + W^T$ (defined later in chapter (3.3)) ("Unified approach").

A.2) Implementation A.2 is from the aspect of the derivation of $\mathbf{R}_g$ and $\mathbf{K}_g$ equivalent to implementation A.1. Difference between A.1 and A.2 is in organisation of element nodes. Separate node is defined for each physical field, totalling to 12 nodes.

B) Three subsets of $\boldsymbol{\phi}$ are defined: $\boldsymbol{\phi}^{(1)} = \{u, v\}$, $\boldsymbol{\phi}^{(2)} = \{p_w\}$ and $\boldsymbol{\phi}^{(3)} = \{T\}$. They are discretised by sets of unknowns of variables defined in three separate elements, each with its own vector of nodal unknown degrees of freedom $\mathbf{p}_e^{(i)}$. Pseudo potential is defined for each element, i.e. $W^{(1)} = W^u$, $W^{(2)} = W^{p_w}$ in $W^{(3)} = W^T$.

For implementations A.1 and A.2, $\mathbf{K}$ in $\mathbf{R}$ are derived in one element from single pseudo potential: $\mathbf{R}_g = \begin{Bmatrix} \mathbf{R}_g^u \\ \mathbf{R}_g^{p_w} \\ \mathbf{R}_g^T \end{Bmatrix}$ and $\mathbf{K}_g = \begin{bmatrix} \mathbf{K}_g^{uu} & \mathbf{K}_g^{up_w} & \mathbf{K}_g^{uT} \\ \mathbf{K}_g^{p_w u} & \mathbf{K}_g^{p_w p_w} & \mathbf{K}_g^{p_w T} \\ \mathbf{K}_g^{Tu} & \mathbf{K}_g^{Tp_w} & \mathbf{K}_g^{TT} \end{bmatrix}$

In case B, there are three separate residuals $\mathbf{R}_g^u$, $\mathbf{R}_g^{p_w}$ and $\mathbf{R}_g^T$ and tangent-stiffness matrices: $\mathbf{K}_g^u = \begin{bmatrix} \mathbf{K}_g^{uu} & \mathbf{K}_g^{up_w} & \mathbf{K}_g^{uT} \end{bmatrix}$, $\mathbf{K}_g^{p_w} = \begin{bmatrix} \mathbf{K}_g^{p_w u} & \mathbf{K}_g^{p_w p_w} & \mathbf{K}_g^{p_w T} \end{bmatrix}$ and $\mathbf{K}_g^T = \begin{bmatrix} \mathbf{K}_g^{Tu} & \mathbf{K}_g^{Tp_w} & \mathbf{K}_g^{TT} \end{bmatrix}$.

3.3 Pseudo-potential of THM problem

For each of the weak forms defined in chapter 3.1, a proper pseudo potential can be written, which is derived as sum of products of auxiliary functions $\mathbf{b}_m^l$ and functions $\mathbf{a}_m^l$ (see chapter 2). Additionaly, the ADB formulation of spatial gradient is defined as $\nabla_{\mathbf{x}}(\square) = \frac{\hat{\delta}\square}{\hat{\delta}\mathbf{x}}\Big|_{\frac{D\boldsymbol{\Xi}}{D\mathbf{x}} = \mathbf{J}_g^{-1}\boldsymbol{F}^{-1}}$, where $\boldsymbol{\Xi} = \{\xi, \eta, \zeta\}$ are reference coordinates which are expressed in shape functions $N_i = N_i(\xi, \eta, \zeta)$ (e.g. $N_i^{Q_1} = 1/4\{(1-\eta)(1-\xi), ..., (\eta+1)(1-\xi)\}$). $\mathbf{J}_g = \frac{\hat{\delta}\mathbf{X}}{\hat{\delta}\boldsymbol{\Xi}}$ is Jacobian matrix of transformation from $\boldsymbol{\Xi}$ to $\mathbf{X}$ and $\mathbf{x}$ is vector of spatial coordinates. Material gradient can be derived similarly as $\nabla_{\mathbf{X}}(\square) = \frac{\hat{\delta}\square}{\hat{\delta}\mathbf{X}}\Big|_{\frac{D\boldsymbol{\Xi}}{D\mathbf{X}} = \mathbf{J}_g^{-1}}$.

3.3.1 Pseudo-potential of mechanical problem

Let us take weak form of balance equation (8). The corresponding pseudo-potential has the form of $W_g^u = W_u + \mathbf{b}_1^u : \boldsymbol{E} - \mathbf{b}_2^u \cdot \boldsymbol{u}$, where $\mathbf{b}_1^u = \boldsymbol{S}^{p_w}$ and $\mathbf{b}_2^u = \rho \boldsymbol{g}$ are auxiliary variables, which must be kept constant during automatic differentiation of $\mathbf{R}$. This will be achieved with appropriate ADB formulation shown below. W_u is elastic strain energy function, describing appropriate material model, in our case elastic and hyperelastic.

3.3.2 Pseudo potential of temperature conduction problem

Similarly as for mechanical problem, we define pseudo potential of non-stationary heat conduction from (9) in the shape of: $W_g^T = \mathbf{b}_1^T \cdot \nabla_{\mathbf{x}}(T) + b_2^T T$, where $\mathbf{b}_1^T$ and b_2^T are auxiliary variables.

$$\mathbf{b}_1^T = k_T \nabla_{\mathbf{x}}(T) \tag{11}$$

$$b_2^T = \frac{(\rho c)}{\Delta t}(T - T_p) + \rho_w c_p^w \frac{k_{ws}}{\mu_w}\left(-\nabla_{\mathbf{x}}(p_w) + \rho_w \boldsymbol{g}\right) \cdot \nabla_{\mathbf{x}}(T) - \mathcal{D}_{mech} + \mathcal{H} - Q \tag{12}$$

3.3.3 Pseudo potential of liquid flow

As for previous cases, the auxiliary variables are defined as:

$$\mathbf{b}_1^{p_w} = \frac{k_s}{\mu_w}\nabla_{\mathbf{x}}(p_w) - \frac{k_s}{\mu_w}\boldsymbol{g}\rho_w \tag{13}$$

$$b_2^{p_w} = \frac{\alpha_b}{\Delta t} tr\left(\boldsymbol{F}^{-1} - \boldsymbol{F}_n^{-1}\right) + \left(\frac{\alpha_b - n}{K_s} + \frac{n}{K_w}\right)\frac{1}{\Delta t}(p_w - p_{wp}) - \frac{\alpha}{\Delta t}(T - T_p) \tag{14}$$

Pseudo potentials then follows from (10) as $W_g^{p_w} = \mathbf{b}_1^{p_w} \cdot \nabla_{\mathbf{x}}(p_w) + b_2^{p_w} p_w$.

3.4 ADB formulation of residuals of THM problem

In case of A.1 and A.2 implementations, there is one pseudo potential per integration point $W_g = W_g^u + W_g^T + W_g^{p_w}$. Residual of subsets G_K, $\mathbf{R}_g^{(K)}$ can then be written in the ADB formulation as

$$\mathbf{R}_g = J_g \frac{\hat{\delta} W_g}{\hat{\delta}\mathbf{p}_e}\bigg|_{\frac{D\mathbf{b}_1^u}{D\mathbf{p}_e}=\mathbf{0}, \frac{D\mathbf{b}_2^u}{D\mathbf{p}_e}=\mathbf{0}, \frac{D\mathbf{h}_g}{D\mathbf{p}_e}=\mathbf{0}, \frac{DT_c}{D\mathbf{p}}=0, \frac{D\boldsymbol{F}^{-1}}{D\mathbf{p}_e}=\mathbf{0}, \frac{D\boldsymbol{F}_n^{-1}}{D\mathbf{p}_e}=\mathbf{0}, \frac{D\mathbf{b}_1^T}{D\mathbf{p}_e}=\mathbf{0}, \frac{Db_2^T}{D\mathbf{p}_e}=0, \frac{D\mathbf{b}_1^{p_w}}{D\mathbf{p}_e}=\mathbf{0}, \frac{Db_2^{p_w}}{D\mathbf{p}_e}=0} \tag{15}$$

For case B, there are three separate pseudo potentials, one for each element and accordingly three residuals, written in the ADB formulation as:

$$\begin{aligned}
\mathbf{R}_g^u &= J_g \frac{\hat{\delta} W_g^u}{\hat{\delta}\mathbf{p}_u}\bigg|_{\frac{D\mathbf{b}_1^u}{D\mathbf{p}_e}=\mathbf{0}, \frac{D\mathbf{b}_2^u}{D\mathbf{p}_e}=\mathbf{0}, \frac{D\mathbf{h}_g}{D\mathbf{p}_e}=\mathbf{0}, \frac{DT_c}{D\mathbf{p}}=0}, \\
\mathbf{R}_g^T &= J_g \frac{\hat{\delta} W_g^T}{\hat{\delta}\mathbf{p}_T}\bigg|_{\frac{D\mathbf{b}_1^T}{D\mathbf{p}_e}=\mathbf{0}, \frac{Db_2^T}{D\mathbf{p}_e}=0, \frac{D\boldsymbol{F}^{-1}}{D\mathbf{p}_e}=\mathbf{0}}, \\
\mathbf{R}_g^{p_w} &= J_g \frac{\hat{\delta} W_g^{p_w}}{\hat{\delta}\mathbf{p}_{p_w}}\bigg|_{\frac{D\mathbf{b}_1^{p_w}}{D\mathbf{p}_e}=\mathbf{0}, \frac{Db_2^{p_w}}{D\mathbf{p}_e}=0, \frac{D\boldsymbol{F}^{-1}}{D\mathbf{p}_e}=\mathbf{0}, \frac{D\boldsymbol{F}_n^{-1}}{D\mathbf{p}_e}=\mathbf{0}}
\end{aligned} \tag{16}$$

4 EXAMPLES

4.1 Example introduction

Unified (A.1) approach and field-by-field (B) approach will be compared according to numerical efficiency on 56 simulations. Four constitutive material models will be used: linear elastic (LE), small strain elasto-plastic (LP), hyper-elastic (HY) and finite strain elasto-palstic (JC). Four combinations of physical fields "D", "DT", "DW" and "DTW" will be compared, where "D" is mechanical, "T" temperature and "W" hydrous field. Each model and example will be formulated for 2D domain using quadrilateral (Q1) and for 3D domain using hexahedral (H1) element with first order interpolation. All simulations are made using the same topology, initial and boundary conditions and material data, appropriately administered for 2D and 3D models and used fields.

The test example is a block with dimensions of $2 \times 3 \times 2$. 2D model has dimension of 2×2 with the thickness of 3 units, discretized with mesh of $80 \times 40 = 3200$ elements per subset G_K. The mesh of 3D model is discretized with the mesh of $16 \times 8 \times 16 = 2048$ elements per subset G_K.

4.2 Comparison of Computational efficiency

We compared the code size of elements defined in individual example. For A.1 approach only one element was defined per simulation, but for B approach two to three separate elements have to be defined. The total code size is therefore a sum of code sizes of element source codes of each subset G_K. Secondly, we are interested in evaluation time. We cannot compare different models directly. However, we can compare the time needed to assemble element tangent matrices $\mathbf{K}$ and residuals $\mathbf{R}$ of all subsets G_K. As above, this means that we must summarize the contribution of all elements defined for each subset G_K.

Table 1: Comparison of code size in KBytes

Approach	A.1				B		
Element	D	DT	DW	DTW	DT	DW	DTW
Q1-LE	6457	10105	10969	17721	11767	13406	25549
Q1-LP	14989	20223	20156	28340	21997	22862	36664
Q1-HY	8755	13715	15428	23068	16589	18186	33146
Q1-JC	54704	70666	63409	81667	73626	66571	92874
H1-LE	20084	27136	29588	40720	30198	35056	53598
H1-LP	34318	43459	44622	58194	47845	49896	70533
H1-HY	25876	36212	48669	62019	43142	55888	82532
H1-JC	112327	131085	140333	157384	139097	147326	181863

First we can compare the code size of elements. The code size values are listed in Table 1. Growth of code can be observed with increasing number of fields, topology dimension and constitutive model complexity. All field-by-field approach simulations

have larger total code sizes than their unified counterparts. This is expected, because some equations that are identical in all elements must be calculated for each individual element separately. These equations can represent considerable percent of code, if element constitutive equations are simple, which causes larger differences between code of A.1 and B.

When code size of elements is compared for unified and field-by-field approach, it can be observed that for linear elastic and hyperelastic models the code size of approach B is larger for factors from 1.11 (Q1-LE: DW) to 1.44 (Q1-LE: DTW), but when plastic models are considered, increase in code size is for smaller values between 1.04 (H1-LP: DW) and 1.28 (Q1-LP: DTW).

Here the influence of the number of fields on code size is compared. When the code size of mechanical element is comparable to the code size of temperature and pressure elements (e.g. for non plastic models), adding all physical fields increases the code by values from 2.03 to 2.74 for approach A.1 and for approach B from 2.67 to 4, but when mechanical element is more complex (e.g. plastic models), the addition of physical fields increases the code by smaller values than before, in approach A.1 from 1.4 to 1.9 and B from 1.6 to 2.4. The complexity of material model has the largest effect on code size increase, because for finite strain elasto plastic models, the code size is by factors 3.4 (when all fields are present) to 8.5 (when only one field is present) larger than for linear elastic models.

Table 2: Comparison of normalised element assembly times .

Approach	A				B		
Element	D	DT	DW	DTW	DT	DW	DTW
Q1-LE	1.	1.43	1.39	3.69	2.14	2.76	5.52
Q1-LP	1.53	2.56	2.28	4.87	2.97	3.35	6.31
Q1-HY	1.18	2.46	2.42	5.13	3.17	3.16	7.13
Q1-JC	3.23	6.02	4.87	9.07	7.03	5.2	10.3
H1-LE	5.16	10.8	10.4	22.8	13.2	24.6	26.6
H1-LP	7.92	15.1	13.7	28.	17.3	26.8	28.8
H1-HY	9.07	21.	30.2	52.3	21.5	28.1	47.3
H1-JC	21.1	38.7	45.2	67.4	40.5	43.	68.5

A.1 and B approaches return identical results in the same number of steps and iterations. The number of steps was 133 for all examples, but the average number of iterations per step increased from 1 (linear elastic with one field) to 3.3 (finite strain elasto plastic with all fields), because the problem becomes non-linear. Once assembled, global matrices have the same dimensions and values in both A.1 and B approaches, only different organisation of degrees of freedom. Therefore, the main difference comes from assembly time of matrices. Normalized element assembly times are given in Table 2. Generally, element assembly times of tangent matrices and residuals for one iteration are smaller for

approach A.1. The difference in time comes from larger total code size of B as described above. The largest difference between A.1 and B is for models Q1-LE with two physical fields. With increasing complexity and number of physical fields the difference becomes lower and approaches 1. In some cases (e.g. "H1-HY: DTW"), the factor goes below 1, which is a consequence of optimisation procedure of source code, which works better on smaller source codes.

The results lead to the conclusion that formulation with field-by-field approach is appropriate to use for solving strongly coupled problems. A major disadvantage shows only in larger code size and consequently longer assembly, times when the number of physical fields is small and constitutive equations are simple, but on the other hand, when complex model is used with more fields, the assembly times become comparable.

5 CONCLUSIONS

We have shown that with an appropriate automatic differentiation based formulation of the problem and physically separated (*field-by-field*) implementations of equations, it is possible to efficiently describe coupled problems of arbitrary complexity in a way that quadratic convergent Newton-Raphson procedure is preserved. Proposed *field-by-field* implementation was also numerically compared to the standard *unified*. The code size is on average by 17% larger and element assembly time by 30% longer than that of unified, but with increasing complexity of material model (e.g. finite strain), and number of physical fields, the difference drops to 16% for size, while the assembly times become practically identical. Since the goal is to be able to solve strongly coupled problems with arbitrary number of physical fields, the difference will decrease further with increasing complexity of problem.

REFERENCES

[1] Coussy, O. *Poromechanics*. John Wiley & Sons, Ltd, Chichester, England, (2004).

[2] Griewank, A. and Walther, A. *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*. Society for Industrial and Applied Mathematics, Philadelphia, (2008).

[3] Korelc, J. Automation of primal and sensitivity analysis of transient coupled problems. *Computational Mechanics*, (2009), 44(5):631–649.

[4] Korelc, J. AceGen and AceFEM user manual. Technical report, University of Ljubljana, (2011). URL `http://www.fgg.unilj. si/symech/`.

[5] Levenston, M. E., Eisenberg, S. R., and Grodzinsky, A. J. A variational formulation for coupled physicochemical flows during finite deformations of charged porous media. *International Journal of Solids and Structures*, (1998), 35(34-35):4999–5019.

[6] Lewis, R. W. and Schrefler, B. A. *The finite element method in the static and dynamic deformation and consolidation of porous media*. John Wiley, Chichester; New York, (1998).

[7] Simo, J. C. and Miehe, C. Associative coupled thermoplasticity at finite strains: Formulation, numerical analysis and implementation. *Computer Methods in Applied Mechanics and Engineering*, (1992), 98(1):41–104.

[8] Wriggers, P. *Nonlinear finite element methods*. Springer, Berlin, (2008).