Rakenteiden Mekaniikka (Journal of Structural Mechanics)
Vol. 46, No 4, 2013, pp. 116 – 130

On the solution of non-linear diffusion equation

Reijo Kouhia

Summary. Solution of the diffusion equation is usually performed with the finite element discretization for the spatial elliptic part of the equation and the time dependency is integrated via some difference scheme, often the trapezoidal rule (Crank-Nicolson) or the unconditionally stable semi-implicit two-step algorithm of Lees. A common procedure is to use Picard's iteration with the trapezoidal rule. However, in highly non-linear problems the convergence of Picard's iteration is untolerably slow. A simple remedy is to use consistent linearization and Newton's method. For a certain class of non-linear constitutive models the consistent Jacobian matrix is unsymmetric. This paper discusses the use of the symmetric part of the Jacobian matrix and a combined Newton-type iteration scheme. Numerical results of highly non-linear diffusion problems are shown. Also a note concerning temporal discretization is given.

Key words: non-linear diffusion equation, finite elements, Newton's method

Introduction

Many diffusive phenomena are governed by the balance equation

$$-\nabla \cdot \mathbf{q} + s = c\dot{u}, \tag{1}$$

where $\mathbf{q}$ is the flux vector, which is related to the gradient of the quantity u by the constitutive law

$$\mathbf{q} = -\mathbf{D}(u, \mathbf{g})\mathbf{g}, \qquad \mathbf{g} = \nabla u. \tag{2}$$

Time derivatives are indicated by superimposed dots ($\partial u/\partial t = \dot{u}$). Equations (1) and (2) form a system which can be used to describe many diffusive physical phenomena, e.g. heat conduction, seepage flow, electric fields and frictionless incompressible irrotational flow. In the case of heat conduction u stands for temperature and $s, c, \mathbf{q}$ are the heat source, the heat capacity and the heat flux, respectively. In certain applications the source term s can depend on the solution u and the resulting equation is known as the diffusion-reaction equation.

In the case of heat transfer the boundary conditions can be either prescribed temperature

$$u = u_S, \quad \text{on the boundary part} \quad S_u, \tag{3}$$

specified heat flux

$$\mathbf{q} \cdot \mathbf{n} = -q_S, \quad \text{on the boundary part} \quad S_q, \tag{4}$$

a convection boundary condition

$$\mathbf{q} \cdot \mathbf{n} = h(u - u_{\text{ex}}), \quad \text{on the boundary part} \quad S_{\text{c}}, \tag{5}$$

or a radiation boundary condition

$$\mathbf{q} \cdot \mathbf{n} = \sigma \varepsilon u^4 - \beta q_{\mathrm{r}}, \quad \text{on the boundary part} \quad S_{\mathrm{r}}, \tag{6}$$

where h is the convection coefficient, u_{ex} convective exchange temperature, σ the Stefan-Bolzmann constant, ε the surface emission coefficient, β the surface absorption coefficient, and q_{r} the incident radiant heat flow per unit surface are, respectively. For transient problems also an initial temperature field should be specified.

After finite element semidiscretization, the balance equation (1) takes the form

$$\mathbf{C}\dot{\mathbf{u}} = \mathbf{f}(t, \mathbf{u}), \tag{7}$$

in which $\mathbf{C}$ is the Gram matrix − in heat transfer analysis the capacitance matrix − and $\mathbf{f}$ denotes the unbalance between the given source $\mathbf{s}$ and the internal nodal 'fluxes' $\mathbf{r}$, i.e.

$$\mathbf{f} = \mathbf{s} - \mathbf{r}. \tag{8}$$

The Gram matrix, the internal nodal flux and the source vectors are computed from the element contributions

$$\mathbf{C}^{\mathrm{e}} = \int_{V^{\mathrm{e}}} c\mathbf{N}^T\mathbf{N}\,\mathrm{d}V, \quad \mathbf{r}^{(\mathrm{e})} = \int_{V^{(\mathrm{e})}} \mathbf{B}^T\mathbf{q}\,\mathrm{d}V, \quad \mathbf{s}^{(\mathrm{e})} = \int_{V^{(\mathrm{e})}} \mathbf{N}^T s\,\mathrm{d}V, \tag{9}$$

where $\mathbf{B}$ is the matrix of discretized gradient operator and $\mathbf{N}$ is the row matrix containing the finite element interpolation functions. In rectangular Cartesian coordinate system the discrete gradient matrix has the form

$$\mathbf{B} = \begin{bmatrix} \mathbf{N}_{,x} \\ \mathbf{N}_{,y} \\ \mathbf{N}_{,z} \end{bmatrix}. \tag{10}$$

Further details in the finite element solution of diffusion equation can be found in the excellent textbooks [17, 26].

If the temporal discretization is performed by the one-step one-parameter method

$$\mathbf{u}_{n+\alpha} = (1-\alpha)\mathbf{u}_n + \alpha\mathbf{u}_{n+1}, \tag{11}$$

and the time derivative $\dot{\mathbf{u}}_{n+\alpha}$ is approximated as

$$\dot{\mathbf{u}}_{n+\alpha} \approx \frac{\mathbf{u}_{n+1} - \mathbf{u}_n}{\Delta t}, \tag{12}$$

then from (7) the fully discretized equation system is obtained

$$\frac{1}{\alpha\Delta t}\mathbf{C}\left(\mathbf{u}_{n+\alpha} - \mathbf{u}_n\right) - \mathbf{f}(t_{n+\alpha}, \mathbf{u}_{n+\alpha}) = \mathbf{0}. \tag{13}$$

This one-parameter family of methods comprises both the common implicit backward Euler ($\alpha = 1$) and Crank-Nicolson (trapezoidal rule) ($\alpha = \frac{1}{2}$) methods. It is well known that the trapezoidal rule is unconditionally stable only for linear autonomous systems. However, the midpoint version of the trapezoidal rule is unconditionally stable, as shown by Gourlay [11] and Hughes [12]. It is also the only one of this family which is second

order accurate. In linear autonomous cases the midpoint scheme is identical with the standard trapezoidal rule.

The trapezoidal rule has no algorithmic damping. This produces spurious oscillations if the data is not smooth. A simple remedy for suppressing these oscillations is to use for the few first steps the implicit backward Euler scheme and then switch to the second order accurate midpoint rule [19, 25]. This has no effect on the long term accuracy.

In order to solve the non-linear equation system (13) a Newton-type linearization step is utilized and the resulting equation, at a certain step $n+1$ and iteration i, is the following:

$$\left(\frac{1}{\alpha\Delta t}\mathbf{C}^i + \mathbf{K}^i\right)\boldsymbol{\delta}\mathbf{u}^{i+1} = \mathbf{f}(t_{n+\alpha}, \mathbf{u}^i_{n+\alpha}) - \frac{1}{\alpha\Delta t}\mathbf{C}^i\boldsymbol{\Delta}\mathbf{u}^i, \tag{14}$$

where

$$\mathbf{K}^i = -\left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{u}^i_{n+\alpha}} \tag{15}$$

is the Jacobian matrix[1], and the iterative and incremental steps are defined by

$$\boldsymbol{\delta}\mathbf{u}^{i+1} = \mathbf{u}^{i+1}_{n+\alpha} - \mathbf{u}^i_{n+\alpha}, \quad \boldsymbol{\Delta}\mathbf{u}^i = \mathbf{u}^i_{n+\alpha} - \mathbf{u}_n, \quad \text{thus} \quad \mathbf{u}^{i+1}_{n+\alpha} = \mathbf{u}_n + \boldsymbol{\Delta}\mathbf{u}^i + \boldsymbol{\delta}\mathbf{u}^{i+1}. \tag{16}$$

The Jacobian matrix consists of several parts:

$$\mathbf{K}^i = \mathbf{K}^i_0 + \mathbf{K}^i_u + \mathbf{K}^i_g + \mathbf{K}^i_s + \mathbf{K}^i_r, \tag{17}$$

where $\mathbf{K}_0$ is the linear part, $\mathbf{K}_u$ and $\mathbf{K}_g$ come from the u and $\mathbf{g}$ dependencies of the constitutive equation (2), $\mathbf{K}_r$ from the non-linear radiation boundary condition and $\mathbf{K}_s$ from the u dependent source term. They are assembled from the element matrices

$$\mathbf{K}_0^{(e)^i} = \int_{V^{(e)}} \mathbf{B}^T\mathbf{D}^i\mathbf{B}\,\mathrm{d}V, \qquad \mathbf{D}^i = \mathbf{D}(u^i, \mathbf{g}^i), \tag{18}$$

$$\mathbf{K}_u^{(e)^i} = \int_{V^{(e)}} \mathbf{B}^T\mathbf{G}^i\,\mathrm{d}V, \qquad \mathbf{G}^i = \left(\left.\frac{\partial \mathbf{D}}{\partial u}\right|_{u^i,\mathbf{g}^i}\right)\mathbf{g}^i\mathbf{N}, \tag{19}$$

$$\mathbf{K}_g^{(e)^i} = \int_{V^{(e)}} \mathbf{B}^T\mathbf{D}^i_g\mathbf{B}\,\mathrm{d}V, \qquad \mathbf{D}^i_g = \left(\left.\frac{\partial \mathbf{D}}{\partial \mathbf{g}}\right|_{u^i,\mathbf{g}^i}\right)\mathbf{g}^i, \tag{20}$$

$$\mathbf{K}_s^{(e)^i} = \int_{V^{(e)}} \frac{\partial \bar{s}}{\partial u}\mathbf{N}^T\mathbf{N}\,\mathrm{d}V, \tag{21}$$

$$\mathbf{K}_r^{(e)^i} = \int_{S^{r(e)}} C^i_r\mathbf{N}^T\mathbf{N}\,\mathrm{d}S, \tag{22}$$

where C_r is a temperature dependent term containing the radiation and emission coefficients of the radiating surface. All matrices, except $\mathbf{K}_u$, are symmetric. Dependency of the constitutive matrix $\mathbf{D}$ on u is common in many physical problems. However, the unsymmetric part is usually neglected in the solution of equation (14), which causes convergence problems in highly non-linear cases.

Consequenses of the loss of symmetry seems to be purely practical. Memory requirements will double and are thus very high for large three dimensional computations, especially if a direct linear equation solver is used. Also iterative solvers for unsymmetric linear equations are not as robust as symmetric ones, even though a lot of effort to increase their reliability has been paid during the last decades [28, 21, 29].

[1]It is also called as a tangent matrix, however, it is a true tangent only at the solution point.

Since the unsymmetric part of the Jacobian matrix resembles the convective part of the coefficient matrix in the diffusion-convection equation, one would expect numerical instabilities if standard finite-element approximation is used. However, numerical oscillations were not observed in the numerical experiments. Additional discussion can also be found in the section on numerical examples.

In this paper standard Galerkin finite element technique is used. Good introduction to the selection of optimal weighting functions can be found in Refs. [6, 7]. An interesting recent Hybrid-Trefftz finite element method for the thermal analysis of functionaly graded materials is given in Ref. [8]. Weak Galerkin finite element method is considered in Ref. [18].

In order to avoid the use of an unsymmetric iteration matrix some modified iteration schemes are studied. First, quasi-Newton techniques are shortly described. Secondly, a possible iteration scheme for the solution of the unsymmetric linear equation system is described. It utilizes the Neumann series and element by element type matrix-vector multiplication of the unsymmetric part of the Jacobian matrix. Finally, a combined Newton iteration process is described.

Quasi-Newton techniques

Basic properties

A class of inexact Newton algorithms called quasi-Newton (or variable metric, variance, secant, update or modification methods) have been developed in order to speed up the convergence of the modified Newton method[2] and which could be more efficient than the true Newton-Raphson scheme. The basic idea of these methods is to develope an update formula of the Jacobian matrix, i.e. a good approximation, in such a way which avoids the reforming and factorization of the global matrix.

In order to simplify the notation some abbreviations are introduced. Equation (14) can be written concisely in the following form:

$$\mathbf{H}^i \boldsymbol{\delta}\mathbf{u}^{i+1} = \tilde{\mathbf{f}}, \tag{23}$$

where

$$\mathbf{H}^i = \frac{1}{\alpha\Delta t}\mathbf{C}^i + \mathbf{K}^i, \qquad \tilde{\mathbf{f}} = \mathbf{f}(t_{n+\alpha}, \mathbf{u}^i_{n+\alpha}) - \frac{1}{\alpha\Delta t}\mathbf{C}^i \boldsymbol{\Delta}\mathbf{u}^i. \tag{24}$$

In the following the tilde over the effective unbalanced nodal flux vector is omitted. The basic requirement for the approximation $\tilde{\mathbf{H}}^{i+1}$ is to satisfy the *secant relationship* or *quasi-Newton* equation

$$\mathbf{f}(\mathbf{u}^{i+1}) = \mathbf{f}(\mathbf{u}^i) - \tilde{\mathbf{H}}^{i+1}(\mathbf{u}^{i+1} - \mathbf{u}^i), \tag{25}$$

written in a concise form

$$\tilde{\mathbf{H}}^{i+1}\boldsymbol{\delta}\mathbf{u}^{i+1} = \boldsymbol{\delta}\mathbf{f}^{i+1} \quad \text{or in the inverse form} \quad \tilde{\mathbf{A}}^{i+1}\boldsymbol{\delta}\mathbf{f}^{i+1} = \boldsymbol{\delta}\mathbf{u}^{i+1}, \tag{26}$$

where[3]

$$\boldsymbol{\delta}\mathbf{u}^{i+1} = \mathbf{u}^{i+1} - \mathbf{u}^i, \quad \boldsymbol{\delta}\mathbf{f}^{i+1} = \mathbf{f}^i - \mathbf{f}^{i+1}, \quad \tilde{\mathbf{A}}^{i+1} = (\tilde{\mathbf{H}}^{i+1})^{-1}. \tag{27}$$

In one dimensional space the secant equation defines the update - which is a scalar - uniquely. However, for multidimenisonal problems additional requirements have to be

[2] In the modified Newton method the Jacobian matrix is kept constant for the whole step.

[3] Notice the difference in definition of $\boldsymbol{\delta}\mathbf{u}$ and $\boldsymbol{\delta}\mathbf{f}$.

imposed. A reasonable requirement is, that the updated matrix $\tilde{\mathbf{H}}^i$ is close to the previous matrix $\mathbf{H}^{i-1}$. This nearness is measured by matrix norms, and usually, in connection to quasi-Newton updates, the Frobenius norm or its weighted form are often used. It is also desirable that the updated matrix should inherit some properties which are characteristic to the system. In finite element applications such properties usually are symmetry and positive definiteness of the Jacobian matrix. So, the update $\tilde{\mathbf{H}}^i$ (or $\tilde{\mathbf{A}}$) should also satisfy:

$$\text{if} \quad \mathbf{H}^i = (\mathbf{H}^i)^T \quad \text{then} \quad \tilde{\mathbf{H}}^{i+1} = (\tilde{\mathbf{H}}^{i+1})^T \tag{28}$$

$$\text{if} \quad \mathbf{x}^T\mathbf{H}^i\mathbf{x} > 0 \quad \text{then} \quad \mathbf{x}^T\tilde{\mathbf{H}}^{i+1}\mathbf{x} > 0, \quad \forall \mathbf{x} \neq \mathbf{0}. \tag{29}$$

However, it should be remembered that the new iterative change $\delta\mathbf{u}^{i+1}$ has to be easily and cost effectively computed, otherwise the benefit of this kind of update is lost since the price which is paid for omitting the full Newton step is the degradation of the convergence rate. An excellent review on quasi-Newton techniques is written by Dennis and Moré [5].

The quasi-Newton techniques are closely related to the conjugate-Newton methods, see Refs. [1, 13, 22]. Applications of quasi-Newton strategies to structural and fluid flow problems can be found in Refs. [20, 9, 10, 15, 16].

Rank-one update

A single rank update to the Jacobian matrix is a correction of the form

$$\tilde{\mathbf{H}} = \mathbf{H} + \alpha\hat{\mathbf{y}}\hat{\mathbf{z}}^T \quad \text{or} \quad \tilde{\mathbf{A}} = \mathbf{A} + \beta\hat{\mathbf{q}}\hat{\mathbf{v}}^T \tag{30}$$

where the unit vectors $\hat{\mathbf{y}}, \hat{\mathbf{z}}$ (or $\hat{\mathbf{q}}, \hat{\mathbf{v}}$) and the scalar α (or β) are to be determined. Substituting this expression into the quasi-Newton equation (25) and minimizing the difference between the update and the previous matrix, gives the Broyden update formula [3]:

$$\tilde{\mathbf{H}} = \mathbf{H} + \frac{(\boldsymbol{\delta}\mathbf{f} - \mathbf{H}\boldsymbol{\delta}\mathbf{u})\boldsymbol{\delta}\mathbf{u}^T}{\boldsymbol{\delta}\mathbf{u}^T\boldsymbol{\delta}\mathbf{u}} \quad \text{or} \quad \tilde{\mathbf{A}} = \mathbf{A} + \frac{(\boldsymbol{\delta}\mathbf{u} - \mathbf{A}\boldsymbol{\delta}\mathbf{f})\boldsymbol{\delta}\mathbf{f}^T\mathbf{A}}{\boldsymbol{\delta}\mathbf{f}^T\mathbf{A}\boldsymbol{\delta}\mathbf{f}}. \tag{31}$$

Broyden's update formula does not have the property of hereditary symmetry and positive definiteness. However, it is interesting to note, that a symmetric rank one update is obtained from (30) by choosing $\mathbf{z} = \mathbf{y} = \boldsymbol{\delta}\mathbf{f} - \mathbf{H}\boldsymbol{\delta}\mathbf{u}$ (or $\mathbf{q} = \mathbf{v} = \boldsymbol{\delta}\mathbf{u} - \mathbf{A}\boldsymbol{\delta}\mathbf{f}$). Obviously in this case the closeness property is not satisfied.

The update (31) is not performing as well as symmetric rank two updates when the system possesses the symmetry property. However, it can be succesfully used in non-linear diffusion problems where the coeffcients c and $\mathbf{D}$ depend on u, thus producing unsymmetric Jacobian matrix.

Rank-two corrections

A symmetric correction of rank at most two can be written in a basic form [2]

$$\tilde{\mathbf{H}} = \mathbf{H} + \mathbf{s}\mathbf{s}^T - \mathbf{t}\mathbf{t}^T \quad \text{or} \quad \tilde{\mathbf{A}} = \mathbf{A} + \mathbf{y}\mathbf{y}^T - \mathbf{z}\mathbf{z}^T \tag{32}$$

and that particular form is also expressible in a symmetric product form

$$\tilde{\mathbf{H}} = (\mathbf{I} + \mathbf{q}\mathbf{v}^T)\mathbf{H}(\mathbf{I} + \mathbf{q}\mathbf{v}^T)^T \quad \text{or} \quad \tilde{\mathbf{A}} = (\mathbf{I} + \mathbf{w}\mathbf{p}^T)\mathbf{A}(\mathbf{I} + \mathbf{w}\mathbf{p}^T)^T \tag{33}$$

only if the determinant of $\tilde{\mathbf{H}}$ or $\tilde{\mathbf{A}}$ is positive.

Two most well known rank-two corrections are the Davidon-Fletcher-Powell (DFP) and the Broyden-Fletcher-Goldfarb-Shanno (BFGS) updates. These complementary formulas preserve symmetry and positive definiteness of the Jacobian matrix. These formulas are:

$$\tilde{\mathbf{A}}_{BFGS} = \mathbf{A} + \frac{\boldsymbol{\delta}\mathbf{u}(\boldsymbol{\delta}\mathbf{u} - \mathbf{A}\boldsymbol{\delta}\mathbf{f})^T + (\boldsymbol{\delta}\mathbf{u} - \mathbf{A}\boldsymbol{\delta}\mathbf{f})\boldsymbol{\delta}\mathbf{u}^T}{\boldsymbol{\delta}\mathbf{u}^T\boldsymbol{\delta}\mathbf{f}} - \frac{(\boldsymbol{\delta}\mathbf{u} - \mathbf{A}\boldsymbol{\delta}\mathbf{f})^T\boldsymbol{\delta}\mathbf{f}}{(\boldsymbol{\delta}\mathbf{u}^T\boldsymbol{\delta}\mathbf{f})^2}\boldsymbol{\delta}\mathbf{u}\boldsymbol{\delta}\mathbf{u}^T, \tag{34}$$

$$\tilde{\mathbf{H}}_{BFGS} = \mathbf{H} + \frac{\boldsymbol{\delta}\mathbf{f}\boldsymbol{\delta}\mathbf{f}^T}{\boldsymbol{\delta}\mathbf{f}^T\boldsymbol{\delta}\mathbf{u}} - \frac{\mathbf{H}\boldsymbol{\delta}\mathbf{u}\boldsymbol{\delta}\mathbf{u}^T\mathbf{H}}{\boldsymbol{\delta}\mathbf{u}^T\mathbf{H}\boldsymbol{\delta}\mathbf{u}}. \tag{35}$$

The DFP and BFGS update formulas are related to each other by the duality transformations [5]

$$\boldsymbol{\delta}\mathbf{q} \longleftrightarrow \boldsymbol{\delta}\mathbf{f}, \quad \mathbf{H} \longleftrightarrow \mathbf{A} = \mathbf{H}^{-1}, \quad \tilde{\mathbf{H}} \longleftrightarrow \tilde{\mathbf{A}} = \tilde{\mathbf{H}}^{-1}. \tag{36}$$

An alternative form of the BFGS update formula is

$$\tilde{\mathbf{A}}_{BFGS} = (\mathbf{I} - \frac{\boldsymbol{\delta}\mathbf{u}\boldsymbol{\delta}\mathbf{f}^T}{\boldsymbol{\delta}\mathbf{u}^T\boldsymbol{\delta}\mathbf{f}})\mathbf{A}(\mathbf{I} - \frac{\boldsymbol{\delta}\mathbf{f}\boldsymbol{\delta}\mathbf{u}^T}{\boldsymbol{\delta}\mathbf{u}^T\boldsymbol{\delta}\mathbf{f}}) + \frac{\boldsymbol{\delta}\mathbf{u}\boldsymbol{\delta}\mathbf{u}^T}{\boldsymbol{\delta}\mathbf{u}^T\boldsymbol{\delta}\mathbf{f}}. \tag{37}$$

For detailed derivation of these equations, see Ref. [5]. The inverse update form (37) or its product form (33) are usually used in the finite element applications. There are simple recursion formulas to compute the iterative change in both cases.

Additive decomposition

The additively decomposed Jacobian matrix (17) can be shortly denoted as

$$\mathbf{K} = \mathbf{S} + \mathbf{U} = \mathbf{S}(\mathbf{I} - \mathbf{E}), \qquad \mathbf{E} = -\mathbf{S}^{-1}\mathbf{U}, \tag{38}$$

where $\mathbf{S}$ is the symmetrix part and $\mathbf{U} = \mathbf{K}_u$ the unsymmetric part (18b). Solution of the linear equation system $\mathbf{K}\mathbf{x} = \mathbf{f}$ can be formally written as

$$\begin{aligned}
\mathbf{x} &= \mathbf{K}^{-1}\mathbf{f} = (\mathbf{I} - \mathbf{E})^{-1}\mathbf{S}^{-1}\mathbf{f} = (\mathbf{I} + \sum_{k=1}^{\infty}\mathbf{E}^k)\mathbf{S}^{-1}\mathbf{f} \\
&= \mathbf{S}^{-1}\mathbf{f} + \sum_{k=1}^{\infty}(-\mathbf{S}^{-1}\mathbf{U})^k\mathbf{S}^{-1}\mathbf{f} \\
&= \boldsymbol{\Delta}\mathbf{x}_0 + \sum_{k=1}^{\infty}\boldsymbol{\Delta}\mathbf{x}_k, \qquad \boldsymbol{\Delta}\mathbf{x}_k = (-\mathbf{S}^{-1}\mathbf{U})\boldsymbol{\Delta}\mathbf{x}_{k-1}.
\end{aligned} \tag{39}$$

Thus solving the unsymmetrix system is reduced to a sequence of solutions of the symmetric equation system. In this iteration process the unsymmetric part of the Jacobian matrix need not to be assembled, the multiplication $\mathbf{U}\boldsymbol{\Delta}\mathbf{x}$ can be performed on the element level. It is clear that this approach is only feasible if a direct linear equation solver is used.

Convergence of the iterative process (39) is assured if

$$\|\mathbf{E}\| < 1, \tag{40}$$

where $\|\cdot\|$ denotes matrix norm. This condition is, however, too restrictive in practical computations. If, for instance, an unsymmetric equation system emanating from a linear

diffusion-convection problem, is solved using the scheme (39), the convergence criterion (40) is met in problems where the Péclet number is of order unity. See also Eqs. (49),(50) in the section on numerical examples.

Nevertheless, it could be argued that few steps of the iteration (39) could be used in the non-linear case to speed up the convergence of the Newton process with an inconsistent symmetric Jacobian.

Composite Newton iteration

High-order iterative processes can be generated by the composition of two low-order processes. For instance, each Newton step can be combined with m simplified Newton steps [24]

$$\begin{aligned} \mathbf{u}^{i,k} &= \mathbf{u}^{i,k-1} + (\mathbf{H})^{-1}\,\mathbf{f}^{i,k-1}, \quad k = 1, ..., m+1, \\ \mathbf{u}^{i+1} &= \mathbf{u}^{i,m+1} \end{aligned} \tag{41}$$

giving convergence of order $m+2$. It is also called the Shamanskii method [14]. Optimal choices of m are problem dependent and affected from the computational cost ratio between forming of the Jacobian matrix and of the residual vector. If the cost of updating the tangent matrix is high, the Shamanskii method is worthwhile. Numerical experiments show that the number of simplified Newton steps should be variable and usually increasing along the iteration number, e.g. like $m = i$ where i is the number of a corrector iteration.

However, m should have some upper limit for practical purposes. In this study m is limited to three.

Numerical examples

Stationary cases

One-dimensional form of a stationary diffusion problem is the following

$$[-D(u,u')u']' = s, \qquad u(0) = u_0, \quad u(L) = u_L, \tag{42}$$

where the prime denotes the differentiation with respect to the spatial coordinate. Two different non-linear diffusive constitutive models are tested:

$$D(u') = \frac{D_0}{1 + aLu'/u_\mathrm{r}}, \tag{43}$$

$$D(u) = D_0 \exp(bu/u_\mathrm{r}), \tag{44}$$

where D_0 and u_r are reference diffusivity and reference value for u and a, b are dimensionless constants characterizing the non-linearity of the constitutive model. Boundary conditions and source terms corresponding to the constitutive models (43) and (44) are

$$u_0 = 0, \quad u_L = 0, \qquad s = \lambda u_\mathrm{r} D_0 L^{-2}, \tag{45}$$

$$u_0 = 0, \quad u_L = \lambda u_\mathrm{r}, \quad s = 0, \tag{46}$$

where the dimensionless quantity λ acts as the loading parameter.

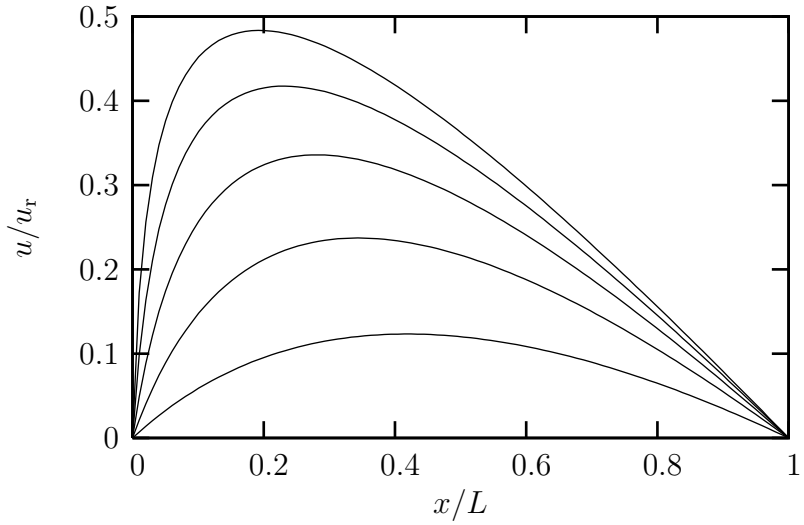


Figure 1. Solution profiles of the Rheinboldt's example.

Analytical solutions can be easily integrated for these problems, and the results are

$$\frac{u(x)}{u_\mathrm{r}} = \frac{1}{\lambda a^2} \ln \left[(\exp(\lambda a) - 1) \frac{x}{L} + 1 \right] - \frac{1}{a} \frac{x}{L}, \tag{47}$$

$$\frac{u(x)}{u_\mathrm{r}} = \frac{1}{b} \ln \left[1 + (\exp(\lambda b) - 1) \, \frac{x}{L} \right]. \tag{48}$$

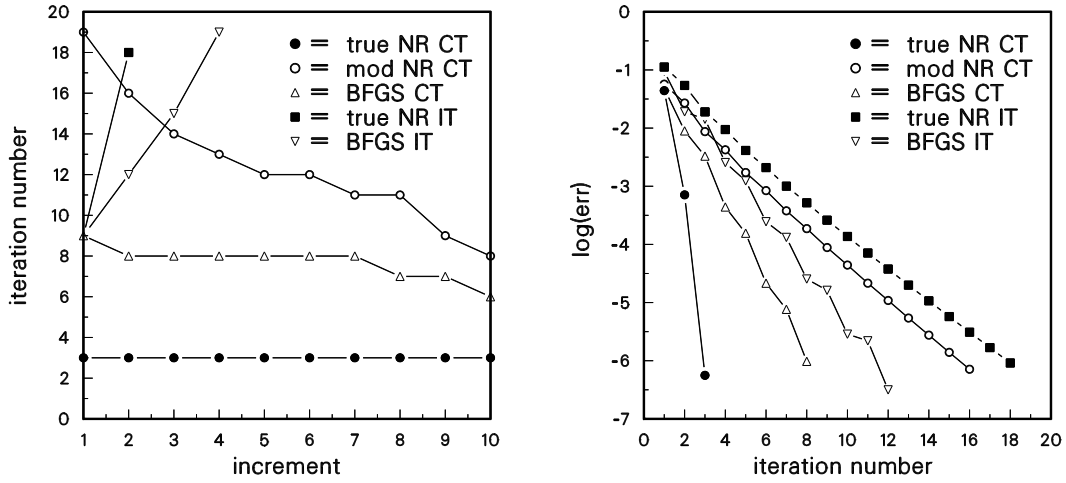
For growing value of the loading parameter λ solutions (47) and (48) exhibit sharp boundary leyers near $x = 0$.

The first example is due to Rheinboldt [27] and the solution is shown in Fig. 1 for values $\lambda = 1, 2, \ldots, 5$. Since the flux in this example depends only on the gradient of u, the consistent Jacobian matrix is symmetric. Therefore, from the quasi-Newton family only the symmetric rank-two BFGS update formula is tested.

Convergence plots are shown in Fig. 2. In the computations the value of $a = 1$ and the increment size $\Delta\lambda = 0.5$ have been used. It can be seen from the figures, that omission of the gradient dependent term $\mathbf{K}_g$ from the stiffness matrix causes severe convergence problems, which cannot be overcome by using the BFGS update scheme. It is also worth noticing that the convergence rate of the Newton's method with inconsistent Jacobian downgrades from quadratic to linear, i.e. as in the case of the simplified Newton-Raphson scheme. Convergence behaviour of the Newton's iteration is unaffected from the discretizations used.

The example case with the exponential diffusivity is demanding. Both uniform and geometrically graded meshes with 10, 30 and 100 linear elements, or 5, 15 and 50 quadratic elements or 3, 10 and 33 cubic elements have been used in the computations. The maximum step-size which can be used in this example is $\Delta u_0 = u_r$ which can be used with the true Newton's method with consistent Jacobian matrix.[4] As in the previous example the Broyden's quasi-Newton strategy is not successfull if the inconsistent Jacobian matrix is used. In figure 4 convergence behaviour is shown for four different strategies: true Newton and the Shamanski's higher order Newton with variable m (limited to $m \leq 3$) and

[4]As usual in the non-linear incremental solution procedure the starting solution for the Newton's iteration is chosen to be the last known equilibrium solution.



(a) Iteration history

(b) Convergence at increment 2

Figure 2. Rheinboldt's example: convergence plots, CT=consistent Jacobian (tangent) matrix, IT=inconsistent Jacobian matrix ($\mathbf{K}_g$ omitted).

started after the first corrector iteration, thus abbreviated as NR-3,1. Methods are tested with both consistent and inconsistent Jacobian matrix. It is observed that the MR-3,1-method is converging rapidly and it is more efficient than the full Newton's method when consistent Jacobian is used. Surprisingly it is also rather fastly converging if inconsistent tangent is used i.e. when $\mathbf{K}_u$ is missing.

The iteration procedure using the additive decomposition described in equation (39) is not successful. Similarly to the diffusion-convection equation, based on the unsymmetric part of the Jacobian matrix (19), a characteristic non-dimensional elementwise Péclet number has the expression

$$\mathrm{Pe}_h = \frac{\dfrac{\partial D}{\partial u}|\nabla u| h^{(\mathrm{e})}}{D}, \tag{49}$$

where $h^{(e)}$ is a characteristic length of an element. For the constitutive model (44), the result is

$$\mathrm{Pe}_h = b\left(\frac{|\Delta u^{(\mathrm{e})}|}{u_{\mathrm{r}}}\right), \tag{50}$$

where $\Delta u^{(\mathrm{e})}$ is the maximum nodal difference in u for an element. Therefore the condition (40) is not satisfied in the boundary layer of (48).

There is influence of the discretization to the convergence behaviour of the finite element solution. A boundary layer has a pollution effect which is clearly seen in Figs. 5 and 6. If a geometrically graded mesh is used the discretization error is more uniform with respect to the loading parameter λ.

The example with the constitutive model (44) is also computed in 2-dimensional domain $\Omega = (0, L) \times (0, L)$. At boundaries $x = L$ and $y = L$ temperature is prescribed by linear variation from zero to λu_r at (L, L). Analytical solution can be obtained by using a transformation $u = (u_r/b)\ln(1+v)$, giving a linear Poisson problem for v. Boundary conditions for this transformed problem are the following: $v(x, L) = \exp(\lambda x/L) - 1, v(L, y) = \exp(\lambda y/L) - 1, v(x, 0) = v(0, y) = 0$. Solution for v is then

$$v = \sum_{i=1}^{\infty} c_n \left(\sinh\frac{n\pi x}{L}\sin\frac{n\pi y}{L} + \sinh\frac{n\pi y}{L}\sin\frac{n\pi x}{L}\right), \tag{51}$$

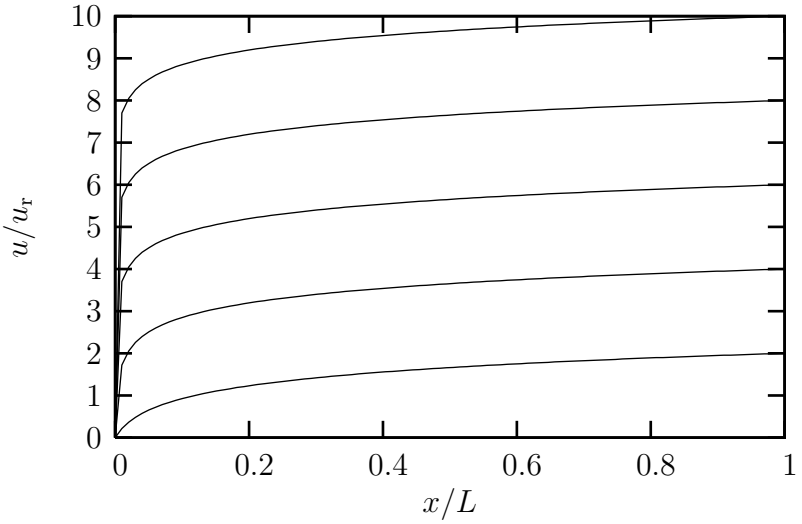
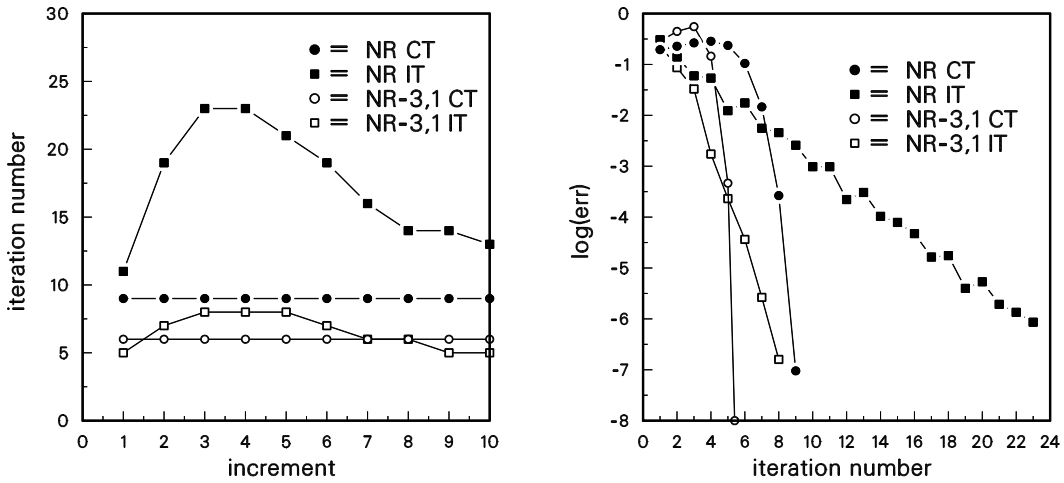
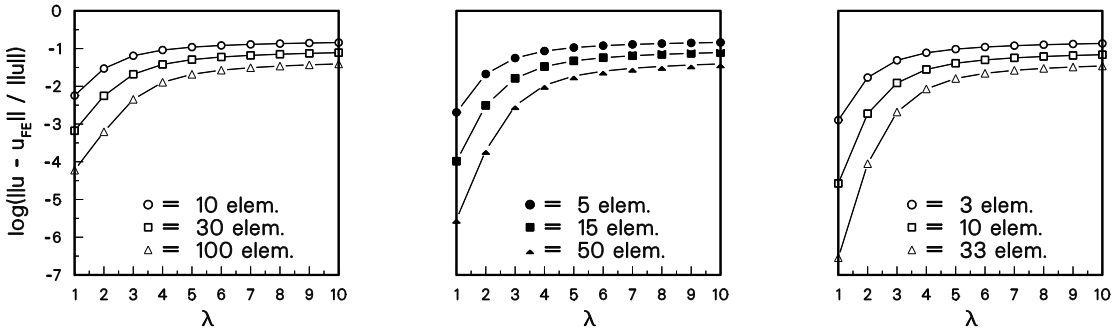


Figure 3. Solution profiles for the example with exponential diffusion coefficient.



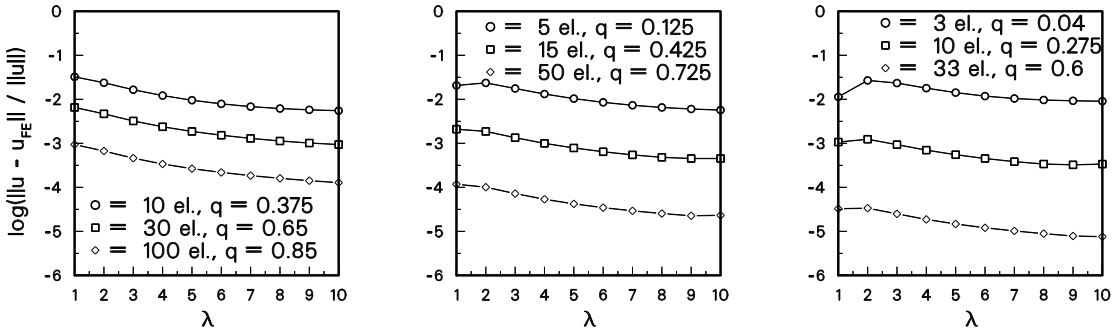
(a) Iteration history (b) Convergence at increment 3

Figure 4. Iteration history and convergence at increment 3 of the 1-D model problem with exponential diffusion coefficient.



(a) Linear elements (b) Quadratic elements (c) Cubic elements

Figure 5. Relative L_2-norm error of temperature u of the 1-D model problem with exponential diffusion coefficient, uniform meshes.



(a) Linear elements (b) Quadratic elements (c) Cubic elements

Figure 6. Relative L_2-norm error of temperature u of the 1-D model problem with exponential diffusion coefficient, geometrically graded meshes.

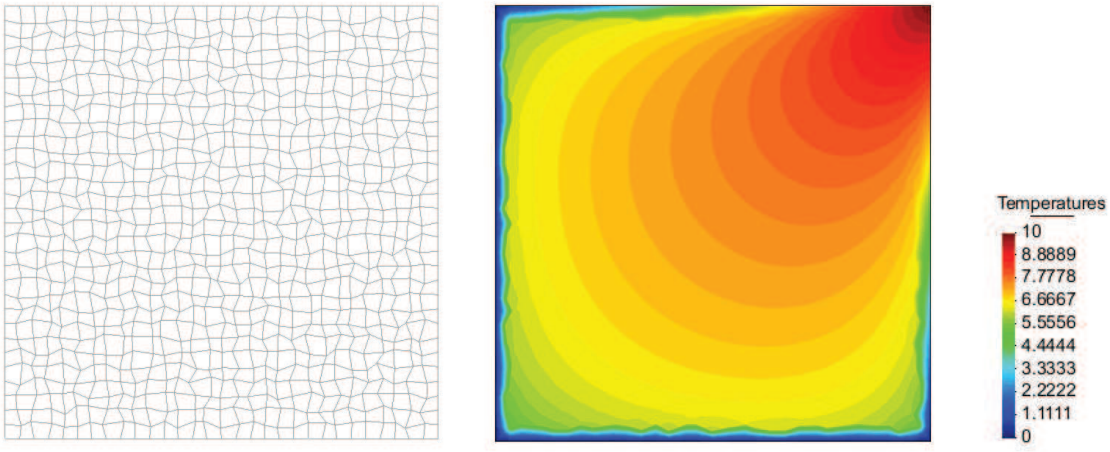


Figure 7. Randomly distorted 30×30-mesh and contour plot of the temperature field at $\lambda = 10$.

where

$$c_n = \frac{2}{n\pi \sinh n\pi} \left[\frac{1 - (-1)^n \exp \lambda b}{1 + (\lambda b / n\pi)^2} + (-1)^n - 1 \right]. \tag{52}$$

Behaviour of this 2-D problem is similar to the 1-D counterpart. For growing λ boundary layers emerge for all boundaries. Convergence of the Newton's iteration with consistent Jacobian is obtained in eight corrector iterations as for every step - stepsize $\Delta\lambda = 1$ - as in the 1-D case. Mesh distortion do not affect the convergence when consistent Jacobian is used, however, the discretization error is larger with distorted meshes. Distorted mesh and the contour plot of temperature are shown in Fig. 7.

Transient examples

To demonstrate the effect of using a two-stage algorithm in order to damp the oscillations in the Crank-Nicolson scheme the following time dependent problems have been solved. The first one is a simple bar with temperature dependent isotropic material properties [23]. Both the thermal conductivity and the heat capacity are assumed to vary according to $1 + \frac{1}{2}(u/u_{\mathrm{r}})$ (u in $^\circ C$). All other surfaces except the surface $x = 0$ are insulated. The initial temperature is $u = 0^\circ C$. The loading is a unit heat input through the surface $x = 0$. The spatial domain is discretized in 15 four node bilinear or eight node reduced biquadratic (serendipity) elements. Results are shown in Fig. 8, where the time step

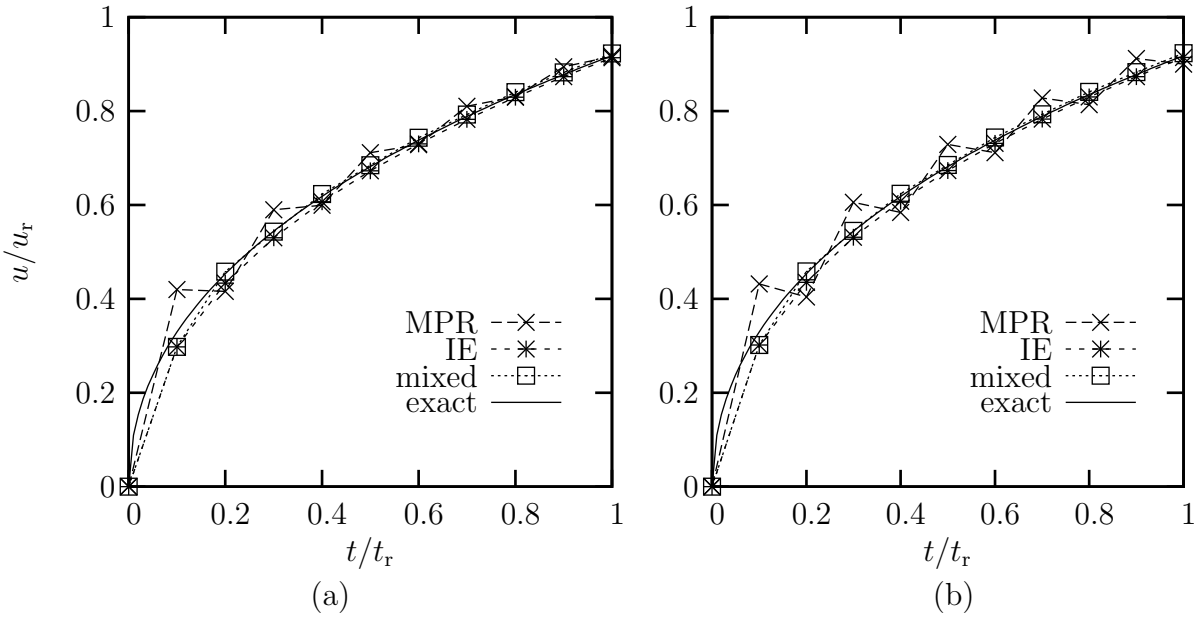


Figure 8. Temperature at the left end of the bar. (a) Numerical solution with bilinear elements and (b) with reduced biquadratic elements. Solid line indicates the exact analytical solution.

$\Delta t = 0.1$ s has been used, reference time is $t_r = 1$ s. Using the one step implicit Euler method before switching to the midpoint rule inhibits the oscillations completely. It is also seen that oscillations of the midpoint rule computations are more pronounced for quadratic than for linear elements.

Cooling of a cube initially at constant unit temperature $T_0 = 1°C$ and subjected to zero surface temperature when $t > 0$ is considered next. The analytical solution of this problem is given in Ref. [4] and finite element solutions e.g. in Ref. [30]. However, in their FE-analyses different initial conditions are used, in which the initial temperature varies from zero to one inside the outmost element layer. One octant has been discretized by 64 trilinear 8-node brick elements. Results are shown in Table 1. The time step has been $\Delta t = 0.0125$ s. Clearly, the use of the one step backward Euler method prior the midpoit rule does not inhibit oscillations completely as in the previous example.

Concluding remarks

Some Newton type iteration procedures have been tested for finite element solution of non-linear diffusion equation. A common procedure to solve non-linear diffusion problems is to use Picard's iteration. However, in highly non-linear problems the convergence of Picard's iteration is untolerably slow. A simple remedy is to use consistent linearization and Newton's method. For a certain class of non-linear constitutive models the consistent Jacobian matrix is unsymmetric. Numerical results of highly non-linear diffusion problems are shown and the convergence of the quasi-Newton updates has been investigated. Numerical experiments showed that the use of the inconsistent symmetric Jacobian with the proposed splitting strategy is not competitive. However, the higher-order Newton method, also known as Shamanski's method performed surprisingly well. Also a simple procedure to supress harmfull oscillations due to the temporal discretization by the midpoint or trapezoidal rules, originally introduced by Rannacher, is tested. It is recommended to be used if the problem data is irregular.

Table 1. Temperature at the center of the cube, 64 trilinear elements.

time	exact	MPR	IE	mixed (1)	MPR (2)
0.0000	1.000000	1.00000	1.00000	1.00000	1.00000
0.0125	1.000000	1.00033	0.99999	0.99999	1.00065
0.0250	0.999954	0.99743	0.99948	0.99931	0.99420
0.0375	0.998436	1.00290	0.99578	1.00306	1.01160
0.0500	0.990637	1.01214	0.98343	1.00339	1.01268
0.0750	0.942211	0.97488	0.92748	0.95188	0.95661
0.1000	0.855496	0.88099	0.84041	0.85258	0.85355
0.2000	0.460657	0.45931	0.46261	0.44011	0.43851
0.3000	0.223432	0.21907	0.23037	0.20971	0.20884
0.4000	0.106825	0.10356	0.11286	0.09912	0.09870
0.5000	0.050973	0.04890	0.05514	0.04681	0.04661
1.0000	0.001259	0.00115	0.00153	0.00110	0.00109

(1) = First step with the implicit Euler method (IE) and the following steps with the midpoint rule (MPR).
(2) = with different initial conditions, identical to the results of Ref. [30].

Kiitokset

Jouni Freundille ja Eero-Matti Saloselle kommenteista.

References

[1] L. Bernspång. Iterative and adaptive solution techniques in computational plasticity, Chalmers Univ. of Tech., Department of Structural Mechanics, Publication 91:8 (1991).

[2] K.W. Brodlie, A.R. Gourlay, and J. Greenstadt. Rank-one and rank-two corrections to positive definite matrices expressed in product form. J. Inst. Maths. Applics, 11 (1973), 73–82.

[3] C.G. Broyden. A class of methods of solving nonlinear simultaneous equations. *Mathematics of Computation*, 19 (1965) 577–592.

[4] H.S. Carslaw and J.C. Jaeger. *Conduction of heat in solids.* Oxford University Press, 1962

[5] J.E. Dennis and J.J. Moré. Quasi-Newton methods, motivation and theory. *SIAM Review*, 19 (1977) 46-89.

[6] J. Freund and E.-M. Salonen. Heikkojen muotojen johtamisesta. *Rakenteiden Mekaniikka*, 23 (3) 1990, 18–61.

[7] J. Freund and E.-M. Salonen. A logic for simple Petrov-Galerkin weighting functions. *Int. J. Numer. Meth. Engng*, 34 (3) 1992, 805–822.

[8] Z.-J. Fu, Q.-H. Qin, and W. Chen. Hybrid-Trefftz finite element method for hear conduction in non-linear functionally graded materials. *Engineering Computations*. 28 (5) 2011, 578–599.

[9] M. Geradin, S. Idelsohn and M. Hogge. Computational strategies for the solution of large non-linear problems via quasi-Newton methods. *Comput. Struct.*, 13 (1981) 73-81

[10] M. Geradin, M. Hogge and S. Idelsohn. Implicit finite element methods. Chapter 7 in *Computational Methods for Transient Analysis*. Eds. T. Belytchko and T.J.R. Hughes, North-Holland (1983).

[11] A.R. Gourlay. A note on trapezoidal methods for the solution of initial value problems. *Math. Comp.* 24 (1970) 629-633.

[12] T.J.R. Hughes. Unconditionally stable algorithms for non-linear heat conduction. *Comp. Meth. Appl. Mech. Engng*, 10 (1977) 135-139.

[13] B. Irons and A. Elsawaf, Proc. U.S.-German Symp. on Formulation and Algorithms in Finite Element Analysis, Eds. K.J. Bathe et al., MIT, pp. 656-672 (1977)

[14] C.T. Kelley. *Iterative Methods for Linear and Nonlinear Equations*. SIAM, Frontiers in Applied Mathematics, vol. 16, 1995

[15] R. Kouhia. Newtonin iteraatiot epälineaarisessa rakenneanalyysissä. *Rakenteiden Mekaniikka*. 19 (4) 1986, 15–51.

[16] R. Kouhia and M. Mikkola. Some aspects on efficient path-following. *Comp. Struct.*, 72, 1999, 509–524.

[17] R.W. Lewis, K. Morgan, H.R. Thomas, K. Seetharamu. *The Finite Element Method in Heat Transfer Analysis*. Wiley, 1996.

[18] Q.H. Li and J. Wang. Weak Galerkin finite element methods for parabolic equations. *Numerical Methods for Partial Differential Equations*. 29 (2013) 6, 2004–2024.

[19] M. Luskin and R. Rannacher. On the smoothing property of the Crank-Nicolson scheme. *Applicable Analysis*. 14 (1982/83) 2, 117-135.

[20] H. Matthies and G. Strang. The solution of nonlinear finite element equations. *Int. J. Numer. Meth. Engng*, 14 (1979) 1613-1626

[21] G. Meurant. *Computer solution of large linear systems*. North-Holland, 1999.

[22] M. Papadrakakis and C.J. Gantes. Preconditioned conjugate- and secant-Newton methods for non-linear problems. *Int. J. Numer. Meth. Engng*, 28 (1989) 1299-1316

[23] S. Orivuori. Efficient methods for solution of non-linear heat conduction problems. *Int. J. Numer. Meth. Engng*, 14 (1979) 1461-1476

[24] J. Ortega and W.C. Rheinboldt. *Iterative Solution of Nonlinear Equations in Several Variables*. Academic Press, 1970

[25] R. Rannacher. Finite element solution of diffusion problems with irregular data. *Numerische Mathematik*. 43 (1984) 2, 309–327.

[26] J.N. Reddy, D.K. Gartling. *The Finite Element Method in Heat Transfer and Fluid Dynamics*. CRC-press, Computational Mechanics and Applied Analysis series, 3^{th} edition, 2010.

[27] W.C. Rheinboldt. *Numerical Analysis of Parametrized Nonlinear Equations*. John Wiley, New York, 1986.

[28] Y. Saad. *Iterative methods for sparse linear systems*. PWS Publishing Company, 1996.

[29] H.A. van der Vorst. *Iterative Krylov methods for large linear systems*. Cambridge University Press, 2003.

[30] O.C. Zienkiewicz and C.J. Parekh. Transient field problems: Two-dimensional and three-dimensional analysis by isoparametric elements. *Int. J. Numer. Meth. Engng*, 1 (1970) 61-71

Reijo Kouhia
Tampere University of Technology
Department of Engineering Design
P.O. Box 589, FI-33101 Tampere
Finland
reijo.kouhia@tut.fi