Fakultät für Mathematik

Michael Jung and Ulrich Rüde

# Implicit Extrapolation Methods for Multilevel Finite Element Computations: Theory and Application

**Abstract.** Extrapolation methods for the solution of partial differential equations are commonly based on the existence of error expansions for the approximate solution. Implicit extrapolation, in the contrast, is based on applying extrapolation indirectly, by using it on quantities like the residual. In the context of multigrid methods, a special technique of this type is known as $\tau$-extrapolation. For finite element systems this algorithm can be shown to be equivalent to higher order finite elements. The analysis is local and does not use global expansions, so that the implicit extrapolation technique may be used on unstructured meshes and in cases where the solution fails to be globally smooth. Furthermore, the natural multilevel structure can be used to construct efficient multigrid and multilevel preconditioning techniques. The effectivity of the method is demonstrated for heat conduction problems and problems from elasticity theory.

**Key Words.** Finite Elements, Extrapolation, Multigrid, Elasticity.

**AMS(MOS) subject classification.** 65F10, 65F50, 65N22, 65N50, 65N55.

**Preprint Reihe der Chemnitzer DFG-Forschergruppe**
**"Scientific Parallel Computing"**

**SPC 94_11** **Juni, 1994**

**1. Introduction.** Multigrid methods have been shown to be very efficient solvers for elliptic partial differential equations (PDE). In this paper we are concerned with the so-called $\tau$-extrapolation multigrid method, see BRANDT [5] and HACKBUSCH [6], which is an extension of conventional multigrid that can improve the accuracy of the numerical result by implicitly using higher order approximations.

In contrast to conventional extrapolation methods for partial differential equations, as described in MARCHUK/SHAIDUROV [13] and BLUM/LIN/RANNACHER [3], the $\tau$-extrapolation algorithm is based on an *implicit* application of Richardson's deferred approach to the limit. We do not take linear combinations of computed approximations, but extrapolate the *residuals* of different levels. This is equivalent to forming a linear combination of the stiffness matrices. The precise meaning of this will be explained in detail later.

We show that one step of multigrid $\tau$-extrapolation for piecewise linear $C^0$ finite element (FE) methods is equivalent to using quadratic elements. This can be derived as a consequence of asymptotic error expansions for the numerical integration of the FE stiffness matrices, as shown in RÜDE [18]. Here we will follow a different approach and show that the quadratic stiffness matrix and the stiffness matrix which is implicitly constructed by $\tau$-extrapolation for linear elements coincide. Therefore the system solved by $\tau$-extrapolation is equivalent to using quadratic elements. Furthermore, we show the asymptotically optimal convergence of a multigrid solution of the extrapolated system.

Our experimental framework is the *Finite Element Multi-Grid Package* (FEMGP) (see STEIDTEN and JUNG [20]) developed at the Technische Universität Chemnitz-Zwickau for the solution of elliptic and parabolic problems arising in the computation of magnetic and thermomechanical fields. We focus on self-adjoint second order linear elliptic partial differential equations, using the heat conduction equation and the equations of elasticity as typical model problems. The equivalence of $\tau$-extrapolation to higher order finite elements justifies to use it even for unstructured meshes as produced with FEMGP, see also the results on $\tau$-extrapolation based higher order adaptive methods by MCCORMICK and RÜDE [14].

**2. Finite element discretizations of the boundary value problem.** We consider two–dimensional second order elliptic boundary value problems:

$$\text{Find } u \in V_0 \text{ such that} \quad a(u,v) = \langle F, v\rangle \quad \text{for all } v \in V_0, \tag{1}$$

with a symmetric, $V_0$–elliptic, and $V_0$–bounded bilinear form $a(.,.)$; $\langle .,. \rangle : V_0^* \times V_0 \to \mathbb{R}^1$ is the duality pairing, $V_0^*$ denotes the space which is dual to $V_0$, and $F \in V_0^*$ is a linear and bounded functional on $V_0$. Later we will describe more precisely which bilinear forms we want to investigate.

Let us first describe some finite element discretizations of problem (1). The starting point of the discretization process is a coarse triangular mesh $\mathcal{T}_1$. Then we generate a sequence of nested triangular meshes $\mathcal{T}_k = \{\delta_k^{(r)},\ r \in \mathcal{J}_k\}$, $k = 1,2,\dots,l$, $\mathcal{J}_k = \{1,2,\dots,R_k\}$, where $R_k$ denotes the number of triangles of the triangulation $\mathcal{T}_k$. We suppose that we obtain the triangulation $\mathcal{T}_l$ by dividing all triangles $\delta_{l-1}^{(r)}$, $r \in \mathcal{J}_{l-1}$, into four congruent subtriangles $\delta_l^{(r)}$. The nodes of the triangulations are numbered *hierarchically*, i.e. $P^{(1)}, P^{(2)}, \dots, P^{(\overline{N}_1)}, P^{(\overline{N}_1+1)}, \dots, P^{(\overline{N}_2)}, \dots, P^{(\overline{N}_{k-1}+1)}, \dots, P^{(\overline{N}_k)},\ \dots, P^{(\overline{N}_{l-1}+1)}, \dots,$ $P^{(\overline{N}_l)}$, where $P^{(\overline{N}_{k-1}+1)}, \dots, P^{(\overline{N}_k)}$ are the nodes of $\mathcal{T}_k$ that do not belong to $\mathcal{T}_{k-1}$ (but are naturally also nodes of $\mathcal{T}_{k+1}, \dots, \mathcal{T}_l$).

Corresponding to each triangulation $\mathcal{T}_k$, $k = 1, 2, \ldots, l-1$, we define the finite element subspaces $V_k \subset V_0$ as

$$V_k = \mathrm{span}\{p_k^{(i)} \ : i = 1, 2, \ldots, N_k\}, \tag{2}$$

where the trial functions $p_k^{(i)}$ are piecewise linear functions such that $p_k^{(i)}$ is linear in all triangles of $\mathcal{T}_k$, continuous, and satisfy the relations $p_k^{(i)}(x_1^{(j)}, x_2^{(j)}) = 1$ for $i = j$, $p_k^{(i)}(x_1^{(j)}, x_2^{(j)}) = 0$ for $i \neq j$, $i, j = 1, 2, \ldots, N_k$. Here $(x_1^{(j)}, x_2^{(j)})$ denotes the coordinates of the node $P^{(j)}$ and $N_k$ is the number of nodes belonging to $\Omega \cup \Gamma_N$, where $\Gamma_N$ is the part of the boundary $\partial\Omega$ on which natural boundary conditions are given.
The finite element subspace corresponding to the finest triangulation $\mathcal{T}_l$ we define for a moment only formally by

$$V_l = \mathrm{span}\{\tilde{p}_l^{(i)}, \ i = 1, 2, \ldots, N_l\}. \tag{3}$$

For the specific choice of the functions $\tilde{p}_l^{(i)}$ we consider four possibilities. The first one is the usual *nodal basis*, i.e. we set $\tilde{p}_l^{(i)} = p_l^{(i)}$, where the functions $p_l^{(i)}$ are defined in the same way as the functions $p_k^{(i)}$, $k = 1, 2, \ldots, l-1$. Consequently, we obtain the FE subspace

$$V_l = V_l^l = \mathrm{span}\{p_l^{(i)}, \ i = 1, 2, \ldots, N_l\}. \tag{4}$$

As second possibility we use the *two–level h–hierarchical basis*, i.e.

$$V_l = \hat{V}_l^l = \mathrm{span}\{p_{l-1}^{(i)}, \ i = 1, \ldots, N_{l-1}\} \cup \mathrm{span}\{p_l^{(i)}, \ i = N_{l-1}+1, \ldots, N_l\}. \tag{5}$$

Additionally, to these two approaches we introduce also FE subspaces spanned by piecewise quadratic functions $q_{l-1}^{(i)}$. These functions are polynomials of degree 2 in all triangles of $\mathcal{T}_{l-1}$, continuous, and satisfy the relations $q_{l-1}^{(i)}(x_1^{(j)}, x_2^{(j)}) = 1$ for $i = j$, $q_{l-1}^{(i)}(x_1^{(j)}, x_2^{(j)}) = 0$ for $i \neq j$, $i, j = 1, 2, \ldots, N_l$. Using these functions we can define the usual *quadratic nodal basis*

$$V_l = V_l^q = \mathrm{span}\{q_{l-1}^{(i)}, \ i = 1, 2, \ldots, N_l\}. \tag{6}$$

and the *two–level p–hierarchical basis*

$$V_l = \hat{V}_l^q = \mathrm{span}\{p_{l-1}^{(i)}, \ i = 1, \ldots, N_{l-1}\} \cup \mathrm{span}\{q_{l-1}^{(i)}, \ i = N_{l-1}+1, \ldots, N_l\}. \tag{7}$$

The sequence of FE subspaces $V_k$, $k = 1, 2, \ldots, l$, where $V_l$ stands for $V_l^l$, $\hat{V}_l^l$, $V_l^q$, or $\hat{V}_l^q$, respectively, results in a sequence of finite element schemes:

$$\text{Find } u_k \in V_k \text{ such that } \quad a(u_k, v_k) = \langle F, v_k \rangle \quad \text{for all } v_k \in V_k. \tag{8}$$

The determination of the unknown function $u_k$ is equivalent to the solution of the system

$$K_k \underline{u}_k = \underline{f}_k \tag{9}$$

of the algebraic finite element equations, where for $k = 1, 2, \ldots, l-1$

$$\underline{u}_k = [u_k^{(i)}]_{i=1,2,\ldots,N_k} \qquad \leftrightarrow \quad u_k = \sum_{i=1}^{N_k} u_k^{(i)} p_k^{(i)}, \tag{10}$$

$$K_k = [K_k^{(ij)}]_{i,j=1,2,\ldots,N_k} \quad , \quad K_k^{(ij)} = a(p_k^{(j)}, p_k^{(i)}), \ \text{ and} \tag{11}$$

$$\underline{f}_k = [f_k^{(i)}]_{i=1,2,\ldots,N_k} \quad , \quad f_k^{(i)} = \langle F, p_k^{(i)} \rangle. \tag{12}$$

For $k = l$ the stiffness matrix $K_l$ and the load vector $\underline{f}_l$ are defined in the same way, we set only the functions $\tilde{p}_l^{(i)}$ instead the functions $p_l^{(i)}$. Depending on the concrete choice of the functions $\tilde{p}_l^{(i)}$, see the possibilities (4) – (7), we get the stiffness matrices $K_l = K_l^l$, $\hat{K}_l^l$, $K_l^q$, or $\hat{K}_l^q$ and the load vectors $\underline{f}_l^l$, $\underline{\hat{f}}_l^l$, $\underline{f}_l^q$, or $\underline{\hat{f}}_l^q$, respectively.
Next we specify the bilinear form $a(.,.)$. In the following we will consider bilinear forms which are defined by

$$a(u,v) = \int_\Omega (A\nabla_x u, \nabla_x v)\, dx \tag{13}$$

where $A$ is a symmetric, positive definite $(2\times 2)$–matrix,

$$\nabla_x = \begin{pmatrix} \dfrac{\partial}{\partial x_1} & \dfrac{\partial}{\partial x_2} \end{pmatrix}^T, \tag{14}$$

and $(.,.)$ denotes the Euclidian scalar product in the space $\mathbb{R}^2$. Such bilinear forms arise from the derivation of the weak formulation of heat conduction problems. Let us suppose that the entries of the matrix $A$ are piecewise constant functions, i.e. constant in each triangle $\delta_{l-1}^{(r)}$, $r \in \mathcal{J}_{l-1}$. In this paper we will not discuss the variable coefficient case.
Next we prove an interesting relation between the matrices $K_{l-1}$, $\hat{K}_l^l$, and $\hat{K}_l^q$, which is useful for the investigation of the convergence properties of a multigrid algorithm with extrapolation.

Lemma 2.1. *Let $K_{l-1}$ $\hat{K}_l^l$, and $\hat{K}_l^q$ be defined by the bilinear form (13) as described above. Then the relation*

$$\hat{K}_l^q = \frac{4}{3}\hat{K}_l^l - \frac{1}{3}\tilde{K}_{l-1} \tag{15}$$

*holds, where* $\tilde{K}_{l-1} = \begin{pmatrix} K_{l-1} & 0 \\ 0 & 0 \end{pmatrix}$.

*Proof:* Recall the definition of the stiffness matrices

$$K_{l-1} = [K_{l-1}^{(ij)}]_{i,j=1,2,\ldots,N_{l-1}}\,, \quad K_{l-1}^{(ij)} = a(p_{l-1}^{(j)}, p_{l-1}^{(i)})\,, \tag{16}$$

$$\hat{K}_l^l = [\hat{K}_l^{l,(ij)}]_{i,j=1,2,\ldots,N_l}\,, \tag{17}$$

$$\hat{K}_l^{l,(ij)} = \begin{cases} a(p_{l-1}^{(j)}, p_{l-1}^{(i)}) & \text{for} \quad i,j = 1,2,\ldots,N_{l-1} \\ a(p_l^{(j)}, p_{l-1}^{(i)}) & \text{for} \quad j = N_{l-1}+1, N_{l-1}+2,\ldots,N_l,\ i = 1,2,\ldots,N_{l-1} \\ a(p_{l-1}^{(j)}, p_l^{(i)}) & \text{for} \quad j = 1,2,\ldots,N_{l-1},\ i = N_{l-1}+1, N_{l-1}+2,\ldots,N_l \\ a(p_l^{(j)}, p_l^{(i)}) & \text{for} \quad i,j = N_{l-1}+1, N_{l-1}+2,\ldots,N_l\,, \end{cases}$$

$$\hat{K}_l^q = [\hat{K}_l^{q,(ij)}]_{i,j=1,2,\ldots,N_l}\,, \tag{18}$$

$$\hat{K}_l^{q,(ij)} = \begin{cases} a(p_{l-1}^{(j)}, p_{l-1}^{(i)}) & \text{for} \quad i,j = 1,2,\ldots,N_{l-1} \\ a(q_{l-1}^{(j)}, p_{l-1}^{(i)}) & \text{for} \quad j = N_{l-1}+1, N_{l-1}+2,\ldots,N_l,\ i = 1,2,\ldots,N_{l-1} \\ a(p_{l-1}^{(j)}, q_{l-1}^{(i)}) & \text{for} \quad j = 1,2,\ldots,N_{l-1},\ i = N_{l-1}+1, N_{l-1}+2,\ldots,N_l \\ a(q_{l-1}^{(j)}, q_{l-1}^{(i)}) & \text{for} \quad i,j = N_{l-1}+1, N_{l-1}+2,\ldots,N_l. \end{cases}$$

All these stiffness matrices have the structure

$$K_l = \begin{pmatrix} K_{l,vv} & K_{l,vm} \\ K_{l,mv} & K_{l,mm} \end{pmatrix} \tag{19}$$

where $K_{l,vv}$ corresponds to the nodes of the triangulation $\mathcal{T}_{l-1}$, $K_{l,mm}$ corresponds to the new nodes in the triangulation $\mathcal{T}_l$, and $K_{l,mv}$, $K_{l,vm}$ are the coupling blocks.
From the definitions (16) – (18) of the matrix elements we see that

$$\frac{4}{3}\hat{K}_l^l - \frac{1}{3}\tilde{K}_{l-1} = \begin{pmatrix} K_{l-1} & \frac{4}{3}\hat{K}_{l,vm}^l \\ \frac{4}{3}\hat{K}_{l,mv}^l & \frac{4}{3}\hat{K}_{l,mm}^l \end{pmatrix} \tag{20}$$

and

$$\hat{K}_l^q = \begin{pmatrix} K_{l-1} & \hat{K}_{l,vm}^q \\ \hat{K}_{l,mv}^q & \hat{K}_{l,mm}^q \end{pmatrix}. \tag{21}$$

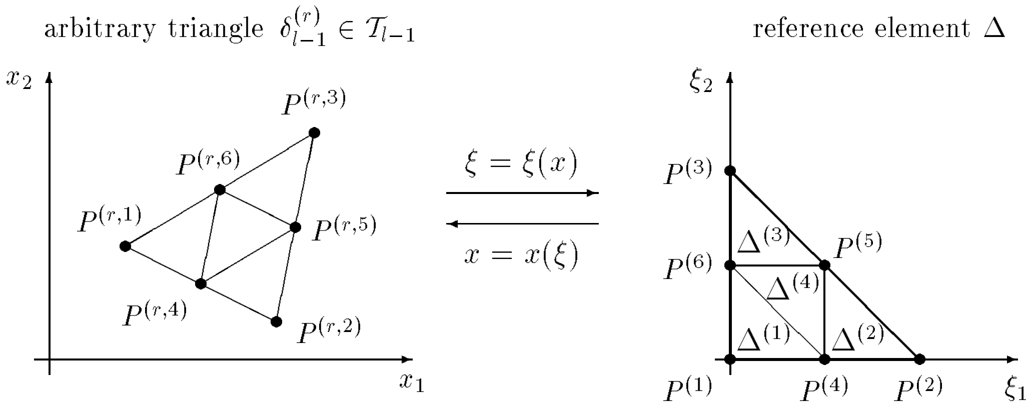
Taking into account that these matrices are symmetric, we have to prove that

$$\frac{4}{3}\hat{K}_{l,vm}^l = \hat{K}_{l,vm}^q \qquad \text{and} \qquad \frac{4}{3}\hat{K}_{l,mm}^l = \hat{K}_{l,mm}^q.$$

To do this we introduce some notations.
The transformation $x = x(\xi)$

$$\begin{aligned} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} &= \begin{pmatrix} x_1^{(r,2)} - x_1^{(r,1)} & x_1^{(r,3)} - x_1^{(r,1)} \\ x_2^{(r,2)} - x_2^{(r,1)} & x_2^{(r,3)} - x_2^{(r,1)} \end{pmatrix} \begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} + \begin{pmatrix} x_1^{(r,1)} \\ x_2^{(r,1)} \end{pmatrix} \\ &= J_{l-1}^{(r)} \begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} + \begin{pmatrix} x_1^{(r,1)} \\ x_2^{(r,1)} \end{pmatrix} \end{aligned} \tag{22}$$

realizes the mapping of the reference element $\Delta = \{(\xi_1, \xi_2) : 0 \le \xi_1 \le 1,\ 0 \le \xi_2 \le 1,\ \xi_1 + \xi_2 \le 1\}$ onto an element $\delta_{l-1}^{(r)}$ of the triangulation $\mathcal{T}_{l-1}$.




$$\begin{aligned} P^{(r,1)} &= P^{(r,1)}(x_1^{(r,1)}, x_2^{(r,1)}), \\ P^{(r,2)} &= P^{(r,2)}(x_1^{(r,2)}, x_2^{(r,2)}), \\ P^{(r,3)} &= P^{(r,3)}(x_1^{(r,3)}, x_2^{(r,3)}) \end{aligned} \qquad \begin{aligned} P^{(1)} &= P^{(1)}(0,0), \\ P^{(2)} &= P^{(2)}(1,0), \\ P^{(3)} &= P^{(3)}(0,1) \end{aligned}$$

FIG. 1. *The mapping between the reference element $\Delta$ and an arbitrary element $\delta_{l-1}^{(r)}$*

On the reference element $\Delta$ we define six shape functions $\tilde{\varphi}_\alpha$, $\alpha = 1, 2, \ldots, 6$. In the case of the $h$-hierarchical basis we have

$$
\begin{aligned}
\tilde{\varphi}_1(\xi_1,\xi_2) &= \varphi_1(\xi_1,\xi_2) = 1-\xi_1-\xi_2 \\
\tilde{\varphi}_2(\xi_1,\xi_2) &= \varphi_2(\xi_1,\xi_2) = \xi_1 \\
\tilde{\varphi}_3(\xi_1,\xi_2) &= \varphi_3(\xi_1,\xi_2) = \xi_2 \\
\tilde{\varphi}_4(\xi_1,\xi_2) &= \varphi_4(\xi_1,\xi_2) = \begin{cases} 2\xi_1 & \text{in } \Delta^{(1)} \\ 2-2\xi_1-2\xi_2 & \text{in } \Delta^{(2)} \\ 0 & \text{in } \Delta^{(3)} \\ 1-2\xi_2 & \text{in } \Delta^{(4)} \end{cases} \\
\tilde{\varphi}_5(\xi_1,\xi_2) &= \varphi_5(\xi_1,\xi_2) = \begin{cases} 0 & \text{in } \Delta^{(1)} \\ 2\xi_2 & \text{in } \Delta^{(2)} \\ 2\xi_1 & \text{in } \Delta^{(3)} \\ 2\xi_1+2\xi_2-1 & \text{in } \Delta^{(4)} \end{cases} \\
\tilde{\varphi}_6(\xi_1,\xi_2) &= \varphi_6(\xi_1,\xi_2) = \begin{cases} 2\xi_2 & \text{in } \Delta^{(1)} \\ 0 & \text{in } \Delta^{(2)} \\ 2-2\xi_1-2\xi_2 & \text{in } \Delta^{(3)} \\ 1-2\xi_1 & \text{in } \Delta^{(4)} \end{cases}
\end{aligned}
\tag{23}
$$

and in the case of the $p$–hierarchical basis

$$
\begin{aligned}
\tilde{\varphi}_1(\xi_1,\xi_2) &= \varphi_1(\xi_1,\xi_2), \quad \tilde{\varphi}_2(\xi_1,\xi_2) = \varphi_2(\xi_1,\xi_2), \quad \tilde{\varphi}_3(\xi_1,\xi_2) = \varphi_3(\xi_1,\xi_2) \\
\tilde{\varphi}_4(\xi_1,\xi_2) &= \psi_4(\xi_1,\xi_2) = 4\xi_1(1-\xi_1-\xi_2) \\
\tilde{\varphi}_5(\xi_1,\xi_2) &= \psi_5(\xi_1,\xi_2) = 4\xi_1\xi_2 \\
\tilde{\varphi}_6(\xi_1,\xi_2) &= \psi_6(\xi_1,\xi_2) = 4\xi_2(1-\xi_1-\xi_2)\,.
\end{aligned}
\tag{24}
$$

In order to calculate the elements of the stiffness matrices we need the derivatives of the shape functions. For the $h$–hierarchical functions we get the partial derivatives given in Table 1.

Table 1

*The partial derivatives of the piecewise linear shape functions*

| | $\frac{\partial\varphi_1}{\partial\xi_1}$ | $\frac{\partial\varphi_1}{\partial\xi_2}$ | $\frac{\partial\varphi_2}{\partial\xi_1}$ | $\frac{\partial\varphi_2}{\partial\xi_2}$ | $\frac{\partial\varphi_3}{\partial\xi_1}$ | $\frac{\partial\varphi_3}{\partial\xi_2}$ | $\frac{\partial\varphi_4}{\partial\xi_1}$ | $\frac{\partial\varphi_4}{\partial\xi_2}$ | $\frac{\partial\varphi_5}{\partial\xi_1}$ | $\frac{\partial\varphi_5}{\partial\xi_2}$ | $\frac{\partial\varphi_6}{\partial\xi_1}$ | $\frac{\partial\varphi_6}{\partial\xi_2}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\Delta^{(1)}$ | -1 | -1 | 1 | 0 | 0 | 1 | 2 | 0 | 0 | 0 | 0 | 2 |
| $\Delta^{(2)}$ | -1 | -1 | 1 | 0 | 0 | 1 | -2 | -2 | 0 | 2 | 0 | 0 |
| $\Delta^{(3)}$ | -1 | -1 | 1 | 0 | 0 | 1 | 0 | 0 | 2 | 0 | -2 | -2 |
| $\Delta^{(4)}$ | -1 | -1 | 1 | 0 | 0 | 1 | 0 | -2 | 2 | 2 | -2 | 0 |

For the computation of the matrix elements in the case of the $p$-hierarchical basis we use the following quadrature rule

$$
\int_\Delta \psi(\xi_1,\xi_2)\,d\xi \approx \sum_{k=1}^{3} \frac{1}{6}\,\psi(\xi^{(k)}) = \frac{1}{6}\left(\psi(0.5,0)+\psi(0,0.5)+\psi(0.5,0.5)\right),
\tag{25}
$$

which is exact for quadratic polynomials on $\Delta$. Therefore we present in the Table 2 the values of the partial derivatives of the functions $\psi_4$, $\psi_5$, and $\psi_6$ in the quadrature points $(0.5,0)$, $(0,0.5)$, and $(0.5,0.5)$.

TABLE 2
*The partial derivatives of the quadratic shape functions*

| | $\frac{\partial\psi_4}{\partial\xi_1}$ | $\frac{\partial\psi_4}{\partial\xi_2}$ | $\frac{\partial\psi_5}{\partial\xi_1}$ | $\frac{\partial\psi_5}{\partial\xi_2}$ | $\frac{\partial\psi_6}{\partial\xi_1}$ | $\frac{\partial\psi_6}{\partial\xi_2}$ |
|---|---|---|---|---|---|---|
| $(0.5,0)$ | 0 | -2 | 0 | 2 | 0 | 2 |
| $(0,0.5)$ | 2 | 0 | 2 | 0 | -2 | 0 |
| $(0.5,0.5)$ | -2 | -2 | 2 | 2 | -2 | -2 |

First we now prove $\frac{4}{3}\hat{K}^l_{l,mm} = \hat{K}^q_{l,mm}$. We have

$$a(\tilde{p}_l^{(j)},\tilde{p}_l^{(i)}) = \int\limits_\Omega \left(A\nabla_x\tilde{p}_l^{(j)},\nabla_x\tilde{p}_l^{(i)})\right)\,dx = \sum_{r\in\omega^{(ij)}}\int\limits_{\delta_{l-1}^{(r)}}\left(A\nabla_x\tilde{p}_l^{(j)},\nabla_x\tilde{p}_l^{(i)})\right)\,dx, \tag{26}$$

where

$$\omega^{(ij)} = \left\{r\in\mathcal{J}_{l-1}:\tilde{p}_l^{(i)}\not\equiv 0 \ \text{ and } \ \tilde{p}_l^{(j)}\not\equiv 0 \ \text{ on } \ \delta_{l-1}^{(r)}\right\}.$$

Obviously, the index sets $\omega^{(ij)}$ are the same for both the $h$-hierarchical functions $\tilde{p}_l^{(i)}=p_l^{(i)}$ and the $p$–hierarchical functions $\tilde{p}_l^{(i)}=q_{l-1}^{(i)}$, $i=N_{l-1}+1,\dots,N_l$. Using the mapping to the reference element it follows

$$a(\tilde{p}_l^{(j)},\tilde{p}_l^{(i)}) = \sum_{r\in\omega^{(ij)}}\int\limits_\Delta\left(A(J_{l-1}^{(r)})^{-T}\nabla_\xi\tilde{p}_l^{(j)}(x(\xi)),(J_{l-1}^{(r)})^{-T}\nabla_\xi\tilde{p}_l^{(i)}(x(\xi))\right)|\det J_{l-1}^{(r)}|\,d\xi = \sum_{r\in\omega^{(ij)}}\int\limits_\Delta\left(B\nabla_\xi\tilde{p}_l^{(j)}(x(\xi)),\nabla_\xi\tilde{p}_l^{(i)}(x(\xi))\right)\,d\xi,$$

with $B=(J_{l-1}^{(r)})^{-1}A(J_{l-1}^{(r)})^{-T}|\det J_{l-1}^{(r)}|$. Note that the entries of $A$, $J_{l-1}^{(r)}$, and $|\det J_{l-1}^{(r)}|$ are constants.
For $j=i$ and $\tilde{p}_l^{(i)}=p_l^{(i)}$, that is for the $h$-hierarchical basis, we have

$$a(p_l^{(i)},p_l^{(i)}) = \sum_{r\in\omega^{(ii)}}\sum_{k\in I(\alpha^{(r)})}\int\limits_{\Delta^{(k)}}\left(B\nabla_\xi\varphi_{\alpha^{(r)}},\nabla_\xi\varphi_{\alpha^{(r)}}\right)\,d\xi = \sum_{r\in\omega^{(ii)}}\sum_{k\in I(\alpha^{(r)})}\frac{1}{8}\left(B\nabla_\xi\varphi_{\alpha^{(r)}}|_{\Delta^{(k)}},\nabla_\xi\varphi_{\alpha^{(r)}}|_{\Delta^{(k)}}\right), \tag{27}$$

where $\alpha^{(r)}$ is the local number of node $P^{(i)}(x_1^{(i)},x_2^{(i)})$ in the triangle $\delta_{l-1}^{(r)}$, i.e.

$$a^{(r)}=4,5,\ \text{or}\ 6,\quad \text{and}\quad I(\alpha^{(r)})=\left\{k\in\{1,2,3,4\}:\varphi_{\alpha^{(r)}}\not\equiv 0\ \text{on}\ \Delta^{(k)}\right\}.$$

Obviously, $I(4)=\{1,2,4\}$, $I(5)=\{2,3,4\}$, and $I(6)=\{1,3,4\}$. Therefore, in all the cases, we have exactly three terms.
If we use quadrature rule (25), we obtain for the case of the $p$-hierarchical basis

$$
\begin{aligned}
(28)\qquad a(q_{l-1}^{(i)},q_{l-1}^{(i)}) &= \sum_{r\in\omega^{(ii)}} \int_{\Delta} (B\nabla_\xi\psi_{\alpha^{(r)}},\nabla_\xi\psi_{\alpha^{(r)}})\,d\xi \\
&= \sum_{r\in\omega^{(ii)}} \sum_{k=1}^{3} \frac{1}{6}\left(B\nabla_\xi\psi_{\alpha^{(r)}}(\xi^{(k)}),\nabla_\xi\psi_{\alpha^{(r)}}(\xi^{(k)})\right),
\end{aligned}
$$

where $\xi^{(k)}$ are the quadrature nodes of formula (25).
Now we compare the summands in the sums over $k$ in (27) and (28). If we examine the values of the partial derivatives $\partial\varphi_{\alpha^{(r)}}/\partial\xi_1$, $\partial\varphi_{\alpha^{(r)}}/\partial\xi_2$ in the triangles $\Delta^{(k)}$ and the values of the derivatives $\partial\psi_{\alpha^{(r)}}/\partial\xi_1$, $\partial\psi_{\alpha^{(r)}}/\partial\xi_2$ in the quadrature nodes (see Tables 1 and 2) then we can see that these summands differ only by the factor $\frac{4}{3}$. Therefore, we have

$$\frac{4}{3}a(p_l^{(i)},p_l^{(i)}) = a(q_{l-1}^{(i)},q_{l-1}^{(i)}) \qquad \text{for} \qquad i=N_{l-1}+1,\ldots,N_l.$$

For $i\neq j$, $i,j=N_{l-1}+1,\ldots,N_l$, we obtain

$$(29)\qquad a(q_{l-1}^{(j)},q_{l-1}^{(i)}) = \sum_{r\in\omega^{(ij)}} \int_{\Delta} \left(B\nabla_\xi\varphi_{\beta^{(r)}},\nabla_\xi\varphi_{\alpha^{(r)}}\right)\,d\xi,$$

where $\beta^{(r)}$ and $\alpha^{(r)}$ are the local numbers of the nodes $P^{(j)}$ and $P^{(i)}$ in the triangle $\delta_{l-1}^{(r)}$, respectively. Using again the quadrature formula (25) and the results from Table 2 we have

$$
(30)\qquad a(q_{l-1}^{(j)},q_{l-1}^{(i)}) = \begin{cases}
\displaystyle\sum_{r\in\omega^{(ij)}} -\frac{1}{3}\left(B\begin{pmatrix}0\\2\end{pmatrix},\begin{pmatrix}2\\2\end{pmatrix}\right) & \text{for} \quad \begin{matrix}\beta^{(r)}=4,\alpha^{(r)}=5\\ \beta^{(r)}=5,\alpha^{(r)}=4\end{matrix} \\
\displaystyle\sum_{r\in\omega^{(ij)}} \frac{1}{3}\left(B\begin{pmatrix}0\\2\end{pmatrix},\begin{pmatrix}2\\0\end{pmatrix}\right) & \text{for} \quad \begin{matrix}\beta^{(r)}=4,\alpha^{(r)}=6\\ \beta^{(r)}=6,\alpha^{(r)}=4\end{matrix} \\
\displaystyle\sum_{r\in\omega^{(ij)}} -\frac{1}{3}\left(B\begin{pmatrix}2\\2\end{pmatrix},\begin{pmatrix}2\\0\end{pmatrix}\right) & \text{for} \quad \begin{matrix}\beta^{(r)}=5,\alpha^{(r)}=6\\ \beta^{(r)}=6,\alpha^{(r)}=5\end{matrix}
\end{cases}
$$

For the $h$-hierarchical basis we get with $I(\beta^{(r)},\alpha^{(r)})=I(\beta^{(r)})\cap I(\alpha^{(r)})$

$$
\begin{aligned}
a(p_l^{(j)},p_l^{(i)}) &= \sum_{r\in\omega^{(ij)}} \sum_{k\in I(\beta^{(r)},\alpha^{(r)})} \int_{\Delta^{(k)}} \left(B\nabla_\xi\varphi_{\beta^{(r)}},\nabla_\xi\varphi_{\alpha^{(r)}}\right)\,d\xi \\
(31)\qquad &= \sum_{r\in\omega^{(ij)}} \sum_{k\in I(\beta^{(r)},\alpha^{(r)})} \frac{1}{8}\left(B\nabla_\xi\varphi_{\beta^{(r)}}|_{\Delta^{(k)}},\nabla_\xi\varphi_{\alpha^{(r)}}|_{\Delta^{(k)}}\right) \\
&= \begin{cases}
\displaystyle\sum_{r\in\omega^{(ij)}} -\frac{1}{4}\left(B\begin{pmatrix}0\\2\end{pmatrix},\begin{pmatrix}2\\2\end{pmatrix}\right) & \text{for} \quad \begin{matrix}\beta^{(r)}=4,\alpha^{(r)}=5\\ \beta^{(r)}=5,\alpha^{(r)}=4\end{matrix} \\
\displaystyle\sum_{r\in\omega^{(ij)}} \frac{1}{4}\left(B\begin{pmatrix}0\\2\end{pmatrix},\begin{pmatrix}2\\0\end{pmatrix}\right) & \text{for} \quad \begin{matrix}\beta^{(r)}=4,\alpha^{(r)}=6\\ \beta^{(r)}=6,\alpha^{(r)}=4\end{matrix} \\
\displaystyle\sum_{r\in\omega^{(ij)}} -\frac{1}{4}\left(B\begin{pmatrix}2\\2\end{pmatrix},\begin{pmatrix}2\\0\end{pmatrix}\right) & \text{for} \quad \begin{matrix}\beta^{(r)}=5,\alpha^{(r)}=6\\ \beta^{(r)}=6,\alpha^{(r)}=5.\end{matrix}
\end{cases}
\end{aligned}
$$

Comparing (30) and (31) we see that $\frac{4}{3}a(p_l^{(j)},p_l^{(i)}) = a(q_{l-1}^{(j)},q_{l-1}^{(i)})$. Consequently, we have shown

$$\frac{4}{3}\hat{K}^l_{l,mm} = \hat{K}^q_{l,mm}\,. \tag{32}$$

It remains to prove $\frac{4}{3}\hat{K}^l_{l,vm} = \hat{K}^q_{l,vm}$. For $j = N_{l-1}+1, N_{l-1}+2\ldots,N_l$, $i=1,2,\ldots,N_{l-1}$ we have

$$\begin{aligned} a(p_l^{(j)},p_{l-1}^{(i)}) &= \sum_{r\in\omega^{(ij)}} \int\limits_{\Delta} (B\nabla_\xi\varphi_{\beta^{(r)}},\nabla_\xi\varphi_{\alpha^{(r)}})\,d\xi \\ &= \sum_{r\in\omega^{(ij)}} \sum_{k\in I(\beta^{(r)})} \frac{1}{8}\left(B\nabla_\xi\varphi_{\beta^{(r)}}|_{\Delta^{(k)}},\nabla_\xi\varphi_{\alpha^{(r)}}|_{\Delta^{(k)}}\right) \end{aligned} \tag{33}$$

and

$$\begin{aligned} a(q_{l-1}^{(j)},p_{l-1}^{(i)}) &= \sum_{r\in\omega^{(ij)}} \int\limits_{\Delta} (B\nabla_\xi\psi_{\beta^{(r)}},\nabla_\xi\varphi_{\alpha^{(r)}})\,d\xi \\ &= \sum_{r\in\omega^{(ij)}} \sum_{k=1}^{3} \frac{1}{6}\left(B\nabla_\xi\psi_{\beta^{(r)}}(\xi^{(k)}),\nabla_\xi\varphi_{\alpha^{(r)}}(\xi^{(k)})\right)\,. \end{aligned} \tag{34}$$

From the Tables 1 and 2 we see again that the summands in the sums over $k$ differ only by the factor $\frac{4}{3}$. Hence, $\frac{4}{3}a(p_l^{(j)},p_{l-1}^{(i)}) = a(q_{l-1}^{(j)},p_{l-1}^{(i)})$, i.e.

$$\frac{4}{3}\hat{K}^l_{l,vm} = \hat{K}^q_{l,vm} \tag{35}$$

and in an analogous way

$$\frac{4}{3}\hat{K}^l_{l,mv} = \hat{K}^q_{l,mv}. \tag{36}$$

Combining the relations (20),(21),(32),(35), and (36) we obtain the statement of the Lemma. ∎

In Lemma 2.2 we formulate the corresponding property for the right–hand side.

LEMMA 2.2. *Let*

$$\langle F,v\rangle = \int\limits_{\Omega} fv\,dx + \int\limits_{\Gamma_N} g_2 v\,ds\,,$$

*where $f$ is a piecewise constant function, i.e. constant over all triangles $\delta_{l-1}^{(r)}$, and $g_2$ a piecewise constant function, i.e. constant over $\partial\delta_{l-1}^{(r)}\cap\partial\Omega$. Then the following relation holds*

$$\underline{\hat{f}}^q_l = \frac{4}{3}\underline{\hat{f}}^l_l - \frac{1}{3}\underline{\tilde{f}}_{l-1}\,, \qquad \underline{\tilde{f}}_{l-1} = \begin{pmatrix} \underline{f}_{l-1} \\ 0 \end{pmatrix}\,. \tag{37}$$

*Proof:* We have defined

$$\begin{aligned}
\underline{f}_{l-1} &= [f_{l-1}^{(i)}]_{i=1,2,\ldots,N_{l-1}}\,, & f_{l-1}^{(i)} &= \langle F, p_{l-1}^{(i)}\rangle \\
\underline{\hat{f}}_l^l &= [\hat{f}_l^{l,(i)}]_{i=1,2,\ldots,N_l}\,, & \hat{f}_l^{l,(i)} &= \begin{cases} \langle F, p_{l-1}^{(i)}\rangle & \text{for} \quad i=1,\ldots,N_{l-1} \\ \langle F, p_l^{(i)}\rangle & \text{for} \quad i=N_{l-1}+1,\ldots,N_l \end{cases} \\
\underline{\hat{f}}_l^q &= [\hat{f}_l^{q,(i)}]_{i=1,2,\ldots,N_l}\,, & \hat{f}_l^{q,(i)} &= \begin{cases} \langle F, p_{l-1}^{(i)}\rangle & \text{for} \quad i=1,2,\ldots,N_{l-1} \\ \langle F, q_{l-1}^{(i)}\rangle & \text{for} \quad i=N_{l-1}+1,\ldots,N_l \end{cases}
\end{aligned}$$

Consequently,

$$\frac{4}{3}\underline{\hat{f}}_l^l - \frac{1}{3}\underline{\tilde{f}}_{l-1} = \begin{pmatrix} \underline{f}_{l-1} \\ \frac{4}{3}\underline{\hat{f}}_{l,m}^l \end{pmatrix} \qquad \text{and} \qquad \underline{\hat{f}}_l^q = \begin{pmatrix} \underline{f}_{l-1} \\ \underline{\hat{f}}_{l,m}^q \end{pmatrix} \tag{38}$$

First we prove

$$\frac{4}{3}\int\limits_{\Omega} f p_l^{(i)}\, dx = \int\limits_{\Omega} f q_{l-1}^{(i)}\, dx \tag{39}$$

for $i = N_{l-1}+1,\ldots,N_l$. Using the notation from the proof of Lemma 2.1 we have

$$\begin{aligned}
\int\limits_{\Omega} f p_l^{(i)}\, dx &= \sum_{r\in\omega^{(i)}} \int\limits_{\delta_{l-1}^{(r)}} f p_l^{(i)}\, dx = \sum_{r\in\omega^{(i)}} \int\limits_{\Delta} f p_l^{(i)}(x(\xi))|\det J_{l-1}^{(r)}|\, d\xi \\
&= \sum_{r\in\omega^{(i)}} \sum_{k\in I(\alpha^{(r)})} \int\limits_{\Delta^{(k)}} f\varphi_{\alpha^{(r)}}(\xi)|\det J_{l-1}^{(r)}|\, d\xi\,,
\end{aligned}$$

where

$$\omega^{(i)} = \{r \in \mathcal{J}_{l-1} \;:\; p_l^{(i)} \not\equiv 0 \;\text{ on }\; \delta_{l-1}^{(r)}\},\quad I(\alpha^{(r)}) = \{k \in 1,2,3,4 \;:\; \varphi_{\alpha^{(r)}} \not\equiv 0 \;\text{ on }\; \Delta^{(k)}\},$$

and $\alpha^{(r)}$ is the local number of the node $P^{(i)}$ in the triangle $\delta_{l-1}^{(r)}$. Computing the integrals over $\Delta^{(k)}$ we obtain

$$\int\limits_{\Omega} f p_l^{(i)}\, dx = \sum_{r\in\omega^{(i)}} \sum_{k\in I(\alpha^{(r)})} \frac{1}{24} f|\det J_{l-1}^{(r)}| = \sum_{r\in\omega^{(i)}} \frac{1}{8} f|\det J_{l-1}^{(r)}|\,, \tag{40}$$

and using the quadrature formula (25) it follows

$$\int\limits_{\Omega} f q_{l-1}^{(i)}\, dx = \sum_{r\in\omega^{(i)}} \sum_{k=1}^{3} \frac{1}{6} f\psi_{\alpha^{(r)}}(\xi^{(k)})|\det J_{l-1}^{(r)}| = \sum_{r\in\omega^{(i)}} \frac{1}{6} f|\det J_{l-1}^{(r)}|\,, \tag{41}$$

i.e. the integrals in (40) and (41) differ by the factor $\frac{4}{3}$. Next we show

$$\frac{4}{3}\int\limits_{\Gamma_N} g_2 p_l^{(i)}\, ds = \int\limits_{\Gamma_N} g_2 q_{l-1}^{(i)}\, ds\,, \tag{42}$$

for $i = N_{l-1}+1,\ldots,N_l$. We have

$$\int\limits_{\Gamma_N} g_2 p_l^{(i)}\, ds = \sum_{e\in E_{l-1}} \int\limits_{\Gamma_{N,l-1}^{(e)}} g_2 p_l^{(i)}\, ds\,,$$

where $\Gamma_{N,l-1}^{(e)}$ is an edge of a triangle $\delta_{l-1}^{(r)}$, $r \in \mathcal{J}_{l-1}$, which is a part of the boundary $\Gamma_N$. The last integral we transform into an integral over the reference interval $[0,1]$. This transformation is described by

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_1^{(e,2)} - x_1^{(e,1)} \\ x_2^{(e,2)} - x_2^{(e,1)} \end{pmatrix} \xi_1 + \begin{pmatrix} x_1^{(e,1)} \\ x_2^{(e,1)} \end{pmatrix} .$$

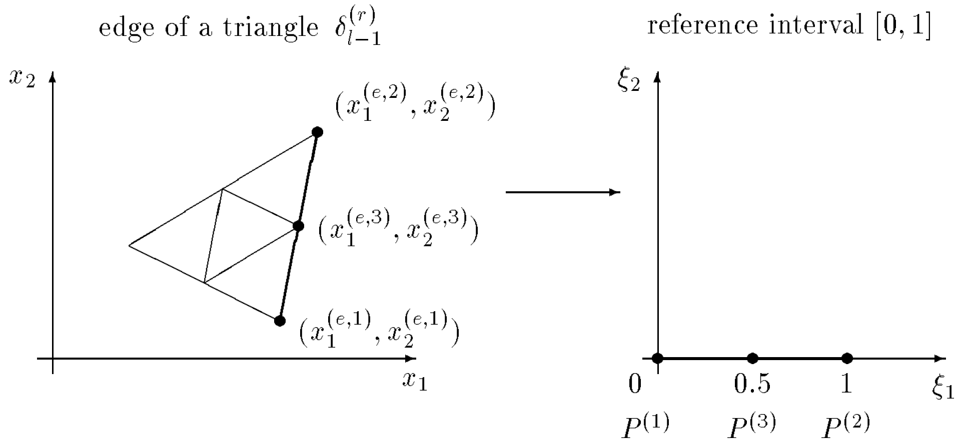



FIG. 2. *The mapping between an edge of a triangle and the reference interval* $[0,1]$

On the reference interval the piecewise linear shape function $\varphi_3(\xi_1)$ is defined as follows

$$\varphi_3(\xi_1) = \begin{cases} 2\xi_1 & \text{in} \quad [0, \frac{1}{2}) \\ 2 - 2\xi_1 & \text{in} \quad [\frac{1}{2}, 1] \end{cases}$$

and for the quadratic shape function $\psi_3(\xi_1)$ we have $\psi_3(\xi_1) = -4\xi_1^2 + 4\xi_1$ .
With $\sigma = [(x_1^{(e,2)} - x_1^{(e,1)})^2 + (x_2^{(e,2)} - x_2^{(e,1)})^2]^{0.5}$ we obtain

$$\begin{aligned} \int\limits_{\Gamma_N} g_2 p_l^{(i)} \, ds &= \sum_{e \in E_{l-1}} \int\limits_{\Gamma_{N,l-1}^{(e)}} g_2 p_l^{(i)} \, ds \\ &= \sum_{e \in E_{l-1}} \left\{ \int\limits_0^{0.5} g_2 2\xi_1 \sigma \, d\xi_1 + \int\limits_{0.5}^{1} g_2 (2 - 2\xi_1) \sigma \, d\xi_1 \right\} \qquad (43) \\ &= \sum_{e \in E_{l-1}} g_2 \sigma \left\{ \frac{1}{4} + \frac{1}{4} \right\} \;=\; \sum_{e \in E_{l-1}} \frac{1}{2} g_2 \sigma \end{aligned}$$

and

$$\begin{aligned} \int\limits_{\Gamma_N} g_2 q_{l-1}^{(i)} \, ds &= \sum_{e \in E_{l-1}} \int\limits_{\Gamma_{N,l-1}^{(e)}} g_2 q_{l-1}^{(i)} \, ds \qquad (44) \\ &= \sum_{e \in E_{l-1}} \int\limits_0^1 g_2 (-4\xi_1^2 + 4\xi_1) \sigma \, ds \;=\; \sum_{e \in E_{l-1}} \frac{2}{3} g_2 \sigma . \end{aligned}$$

Again both integrals differ only by the factor $\frac{4}{3}$. Combining the relations (38),(39), and (42) we get the statement of the Lemma. ■

THEOREM 2.3. *The FE systems of algebraic equations*

$$\left(\frac{4}{3}\hat{K}_l^l - \frac{1}{3}\tilde{K}_{l-1}\right)\underline{\hat{u}}_l = \left(\frac{4}{3}\underline{\hat{f}}_l^l - \frac{1}{3}\underline{\tilde{f}}_{l-1}\right) \qquad \textit{and} \qquad \hat{K}_l^q \underline{\hat{u}}_l = \underline{\hat{f}}_l^q$$

*have the same solution.*

*Proof:* The proof follows immediately from Lemma 2.1 and Lemma 2.2. ■

An analogous result can be proved for the FE systems in the nodal basis. Before we show this property, we state a lemma.

LEMMA 2.4. *Between the p-hierarchical and the quadratic nodal shape functions on the reference element it holds*

$$\hat{\Phi}_\Delta = \Phi_\Delta S_\Delta \, , \tag{45}$$

*where*

$$\begin{aligned} S_\Delta &= [S_\Delta^{(\alpha\beta)}]_{\alpha,\beta=1,\ldots,6}\,, \quad S_\Delta^{(\alpha\beta)} = \begin{cases} 1 & \textit{for} \quad \alpha=\beta \\ \frac{1}{2} & \textit{for} \quad (\alpha,\beta)=(4,1),(4,2) \\ & \qquad\;\; (\alpha,\beta)=(5,2),(5,3) \\ & \qquad\;\; (\alpha,\beta)=(6,3),(6,1) \\ 0 & \textit{otherwise}\,. \end{cases} \\ \hat{\Phi}_\Delta &= (\varphi_1(\xi_1,\xi_2),\, \varphi_2(\xi_1,\xi_2),\, \varphi_3(\xi_1,\xi_2),\, \psi_4(\xi_1,\xi_2),\, \psi_5(\xi_1,\xi_2),\, \psi_6(\xi_1,\xi_2)) \\ \Phi_\Delta &= (\psi_1(\xi_1,\xi_2),\, \psi_2(\xi_1,\xi_2),\, \psi_3(\xi_1,\xi_2),\, \psi_4(\xi_1,\xi_2),\, \psi_5(\xi_1,\xi_2),\, \psi_6(\xi_1,\xi_2)) \end{aligned}$$

*with*

$$\begin{aligned} \varphi_1(\xi_1,\xi_2) &= 1-\xi_1-\xi_2\,, \quad \varphi_2(\xi_1,\xi_2) = \xi_1\,, \quad \varphi_3(\xi_1,\xi_2) = \xi_2\,, \\ \psi_1(\xi_1,\xi_2) &= 2\xi_1^2+2\xi_2^2-3\xi_1-3\xi_2+4\xi_1\xi_2+1\,, \\ \psi_2(\xi_1,\xi_2) &= 2\xi_1^2-\xi_1\,, \quad \psi_3(\xi_1,\xi_2) = 2\xi_2^2-\xi_2\,, \\ \psi_4(\xi_1,\xi_2) &= 4\xi_1(1-\xi_1-\xi_2)\,, \quad \psi_5(\xi_1,\xi_2) = 4\xi_1\xi_2\,, \\ \psi_6(\xi_1,\xi_2) &= 4\xi_2(1-\xi_1-\xi_2)\,. \end{aligned} \tag{46}$$

*Proof:* A simple calculation leads to

$$\begin{aligned} \varphi_1(\xi_1,\xi_2) &= \psi_1(\xi_1,\xi_2)+0.5\,(\psi_4(\xi_1,\xi_2)+\psi_6(\xi_1,\xi_2)) \\ \varphi_2(\xi_1,\xi_2) &= \psi_2(\xi_1,\xi_2)+0.5\,(\psi_4(\xi_1,\xi_2)+\psi_5(\xi_1,\xi_2)) \\ \varphi_3(\xi_1,\xi_2) &= \psi_3(\xi_1,\xi_2)+0.5\,(\psi_5(\xi_1,\xi_2)+\psi_6(\xi_1,\xi_2)) \end{aligned}$$

and therefore (45) holds. ■

Lemma 2.5. *For the p–hierarchical and the quadratic nodal basis the relation*

$$\hat{\Phi} = \Phi S_l \tag{47}$$

*holds, where*

$$\hat{\Phi} = (p_{l-1}^{(1)}, p_{l-1}^{(2)}, \ldots, p_{l-1}^{(N_{l-1})}, q_{l-1}^{(N_{l-1}+1)}, \ldots, q_{l-1}^{(N_l)}) \tag{48}$$

$$\Phi = (q_{l-1}^{(1)}, q_{l-1}^{(2)}, \ldots, q_{l-1}^{(N_l)}) \tag{49}$$

$$S_l = [S_l^{(ij)}]_{i,j=1,2,\ldots,N_l} \tag{50}$$

$$S_l^{(ij)} = \begin{cases} 1 & \textit{for } i = j,\ i,j = 1,2,\ldots,N_l \\ \frac{1}{2} & \textit{for } j = i_1 \textit{ and } j = i_2, N_{l-1} < i \le N_l, \textit{ where } P^{(i)} \textit{ is the} \\ & \textit{midpoint of that edge which is given by the vertices} \\ & \ P^{(i_1)} \textit{ and } P^{(i_2)} \textit{ of a triangle of } \mathcal{T}_{l-1} \\ 0 & \textit{otherwise.} \end{cases} \tag{51}$$

*Proof:* The FE functions are defined element by element, i.e.

$$\tilde{p}_l^{(i)}(x) = \begin{cases} \tilde{p}_\alpha^{(r)}(x) = \tilde{\varphi}_{\alpha^{(r)}}(\xi(x)) & x \in \delta_{l-1}^{(r)},\ r \in B_i \\ 0 & \text{otherwise,} \end{cases}$$

where $\tilde{p}_l^{(i)}$ stands for one function from (48) or (49), $\tilde{\varphi}_{\alpha^{(r)}}$ stands for the corresponding shape function on the reference element $\Delta$, i.e. for the corresponding function of (46), and $B_i = \{r \in \mathcal{J}_{l-1} : P^{(i)} \in \bar{\delta}_{l-1}^{(r)}\}$. Thus the statement of the Lemma follows from Lemma 2.4 immediately. ■

Theorem 2.6. *The FE systems of algebraic equations*

$$\left(\frac{4}{3}K_l^l - \frac{1}{3}\tilde{K}_{l-1}\right)\underline{u}_l = \left(\frac{4}{3}\underline{f}_l^l - \frac{1}{3}\underline{\tilde{f}}_{l-1}\right) \qquad \textit{and} \qquad K_l^q \underline{u}_l = \underline{f}_l^q$$

*have the same solution.*

*Proof:* Using Lemma 2.5 we get for arbitrary vectors $\underline{u}_l$, $\underline{v}_l \in \mathbb{R}^{N_l}$

$$(\hat{K}_l^q \underline{u}_l, \underline{v}_l) = a(\hat{\Phi}\underline{u}_l, \hat{\Phi}\underline{v}_l) = a(\Phi S_l\, \underline{u}_l, \Phi S_l\, \underline{v}_l) = (S_l^T K_l^q S_l\ \underline{u}_l, \underline{v}_l)$$

and

$$(\underline{\hat{f}}_l^q, \underline{v}_l) \quad = \quad \langle F, \hat{\Phi}\underline{v}_l \rangle \quad = \quad \langle F, \Phi S_l\, \underline{v}_l \rangle \quad = \quad (S_l^T \underline{f}_l^q, \underline{v}_l).$$

Therefore we have

$$\hat{K}_l^q = S_l^T K_l^q S_l \quad , \qquad K_l^q = S_l^{-T} \hat{K}_l^q S_l^{-1} \tag{52}$$

$$\underline{\hat{f}}_l^q = S_l^T \underline{f}_l^q \quad , \qquad \underline{f}_l^q = S_l^{-T} \underline{\hat{f}}_l^q\,. \tag{53}$$

Furthermore, from Yserentant [22] we know that

$$\hat{K}_l^l = S_l^T K_l^l S_l \qquad \text{and} \qquad \underline{\hat{f}}_l^l = S_l^T \underline{f}_l^l\,. \tag{54}$$

From (52), (54), Lemma 2.1, and Lemma 2.2 it follows that

$$K_l^q \;=\; S_l^{-T}\hat{K}_l^q S_l^{-1} \;=\; S_l^{-T}\left(\frac{4}{3}\hat{K}_l^l - \frac{1}{3}\tilde{K}_{l-1}\right)S_l^{-1}$$

$$(55)\qquad = \frac{4}{3}S_l^{-T}(S_l^T K_l^l S_l)S_l^{-1} - \frac{1}{3}S_l^{-T}\tilde{K}_{l-1}S_l^{-1} \;=\; \frac{4}{3}K_l^l - \frac{1}{3}\tilde{K}_{l-1}$$

and

$$\underline{f}_l^q \;=\; S_l^{-T}\hat{\underline{f}}_l^q \;=\; S_l^{-T}\left(\frac{4}{3}\hat{\underline{f}}_l^l - \frac{1}{3}\tilde{\underline{f}}_{l-1}\right)$$

$$(56)\qquad = \frac{4}{3}S_l^{-T}(S_l^T \underline{f}_l^l) - \frac{1}{3}S_l^{-T}\tilde{\underline{f}}_{l-1} \;=\; \frac{4}{3}\underline{f}_l^l - \frac{1}{3}\tilde{\underline{f}}_{l-1}\,. \blacksquare$$

**3. Multilevel algorithms with extrapolation.** In this Section we analyse a multigrid algorithm using FE discretizations with piecewise linear functions and an implicit extrapolation step. This algorithm converges to a FE solution which has the same discretization error as a FE solution obtained by a discretization with piecewise quadratic functions. Additionally, we will use this algorithm as a preconditioner in the preconditioned conjugate gradient (PCCG) method.

First we introduce some notations.

- Smoothing procedure
  pre–smoothing $G_l^V(\underline{u}_l^{(j)}, K_l^l, \underline{f}_l^l)$ :
  Let the initial guess $\underline{u}_l^{(j)} = (\underline{u}_{l,v}^{(j)}, \underline{u}_{l,m}^{(j)})^T$ be given.
  Set $\underline{u}_{l,v}^{(j+1)} = \underline{u}_{l,v}^{(j)}$ and compute an approximate solution $\tilde{\underline{z}}_{l,m}$ of the system

  $$(57)\qquad K_{l,mm}^l \underline{z}_{l,m} = \underline{f}_{l,m}^l - K_{l,mv}^l \underline{u}_{l,v}^{(j+1)} - K_{l,mm}^l \underline{u}_{l,m}^{(j)}$$

  by means of an iterative method, starting with the zero–vector. We suppose that the error transmission operator of the method is of the type $M_{l,m} = (I_{l,m} - B_{l,mm}^{-1}K_{l,mm}^l)$.
  Set $\underline{u}_l^{(j+1)} = (\underline{u}_{l,v}^{(j+1)}, \underline{u}_{l,m}^{(j)} + \tilde{\underline{z}}_{l,m})^T$.

  post–smoothing $G_l^N(\underline{u}_l^{(j)}, K_l^l, \underline{f}_l^l)$ :
  We want to use the same algorithm, however we suppose that the error transmission operator of the iterative method for solving the system (57) is of the type $M_{l,m} = (I_{l,m} - B_{l,mm}^{-T}K_{l,mm}^l)$ so that the overall multigrid operator becomes symmetric.
- Interpolation

  $$(58)\qquad I_{l-1}^l \,:\, \mathbb{R}^{N_{l-1}} \to \mathbb{R}^{N_l}, \qquad I_{l-1}^l = \begin{pmatrix} I_{l,v} \\ S_{l,mv} \end{pmatrix},$$

  where

  $$(59)\qquad (I_{l-1}^l)^{(ij)} = \begin{cases} 1 & \text{for } i=j,\ i,j=1,2,\dots,N_{l-1} \\ \frac{1}{2} & \text{for } j=i_1 \text{ and } j=i_2,\, N_{l-1} < i \le N_l, \text{ where } P^{(i)} \text{ is the midpoint of that edge which is given by the vertices } P^{(i_1)} \text{ and } P^{(i_2)} \text{ of a triangle from } \mathcal{T}_{l-1} \\ 0 & \text{otherwise.} \end{cases}$$

- Restrictions

$$I_l^{l-1} \, : \, \mathbb{R}^{N_l} \to \mathbb{R}^{N_{l-1}} \quad , \qquad I_l^{l-1} = (I_{l-1}^l)^T = (I_{l,v} \quad S_{l,mv}^T) \tag{60}$$

$$I_l^{l-1,inj} \, : \, \mathbb{R}^{N_l} \to \mathbb{R}^{N_{l-1}} \quad , \qquad I_l^{l-1,inj} = (I_{l,v} \quad 0) \tag{61}$$

Now we formulate the multigrid algorithm.

*Algorithm 1*

Let an initial guess $\underline{u}_l^{(k,0)}$ be given.

1. pre–smoothing

$$\underline{u}_l^{(k,1)} = G_l^V(\underline{u}_l^{(k,0)}, K_l^l, \underline{f}_l^l) \tag{62}$$

2. coarse–grid correction
   (a) Computation of the defect

$$\underline{d}_{l-1}^{(k)} = \frac{4}{3} I_l^{l-1}(\underline{f}_l^l - K_l^l \underline{u}_l^{(k,1)}) - \frac{1}{3}(\underline{f}_{l-1} - K_{l-1} I_l^{l-1,inj} \underline{u}_l^{(k,1)}) \tag{63}$$

   (b) Solution of the system

$$K_{l-1} \underline{w}_{l-1}^{(k)} = \underline{d}_{l-1}^{(k)} \tag{64}$$

   by means of $\mu$ iterations steps of an usual multigrid ($(l-1)$–grid) algorithm (see, e.g. [6]) which starts with the zero–vector and returns an approximate solution $\underline{\tilde{w}}_{l-1}^{(k)}$.

   (c) Computation of the correction

$$\underline{u}_l^{(k,2)} = \underline{u}_l^{(k,1)} + I_{l-1}^l \underline{\tilde{w}}_{l-1}^{(k)} \tag{65}$$

3. post–smoothing

$$\underline{u}_l^{(k,3)} = G_l^N(\underline{u}_l^{(k,2)}, K_l^l, \underline{f}_l^l) \tag{66}$$

Set $\underline{u}_l^{(k+1,0)} = \underline{u}_l^{(k,3)}$.

Before we present an alternative formulation of this algorithm, we analyse the smoothing step and the computation of the defect.

- The essential operation in the smoothing step is the approximate solution of system (57). Obviously, we can replace equation (57) by

$$\frac{4}{3} K_{l,mm}^l \underline{z}_{l,m} = \frac{4}{3} \underline{f}_{l,m}^l - \frac{4}{3} K_{l,mv}^l \underline{u}_{l,v}^{(j+1)} - \frac{4}{3} K_{l,mm}^l \underline{u}_{l,m}^{(j)} \, . \tag{67}$$

  Using the relations (55) and (56) in the proof of the Theorem 2.6 we get the equivalence of relation (67) to

$$K_{l,mm}^q \underline{z}_{l,m} = \underline{f}_{l,m}^q - K_{l,mv}^q \underline{u}_{l,v}^{(j+1)} - K_{l,mm}^q \underline{u}_{l,m}^{(j)} \, . \tag{68}$$

- Step 2(a) in Algorithm 1 can be formulated in terms of the quadratic nodal basis. We have

$$
\begin{aligned}
&\frac{4}{3} I_l^{l-1}(\underline{f}_l^l - K_l^l \underline{u}_l^{(k,1)}) - \frac{1}{3}(\underline{f}_{l-1} - K_{l-1} I_l^{l-1,inj} \underline{u}_l^{(k,1)}) \\
&= \frac{4}{3}\begin{pmatrix} I_{l,v} & S_{l,mv}^T \end{pmatrix} \left( \begin{pmatrix} \underline{f}_{l,v}^l \\ \underline{f}_{l,m}^l \end{pmatrix} - \begin{pmatrix} K_{l,vv}^l & K_{l,vm}^l \\ K_{l,mv}^l & K_{l,mm}^l \end{pmatrix} \begin{pmatrix} \underline{u}_{l,v}^{(k,1)} \\ \underline{u}_{l,m}^{(k,1)} \end{pmatrix} \right) \\
&\quad -\frac{1}{3}\left( \begin{pmatrix} \underline{f}_{l-1} \\ 0 \end{pmatrix} - \begin{pmatrix} K_{l-1} & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} (I_{l,v} \;\; 0)\underline{u}_l^{(k,1)} \\ 0 \end{pmatrix} \right) \qquad (69) \\
&= \begin{pmatrix} I_{l,v} & S_{l,mv}^T \end{pmatrix} \left[ \left( \frac{4}{3} \begin{pmatrix} \underline{f}_{l,v}^l \\ \underline{f}_{l,m}^l \end{pmatrix} - \frac{1}{3} \begin{pmatrix} \underline{f}_{l-1} \\ 0 \end{pmatrix} \right) \right. \\
&\quad \left. - \left( \frac{4}{3} \begin{pmatrix} K_{l,vv}^l & K_{l,vm}^l \\ K_{l,mv}^l & K_{l,mm}^l \end{pmatrix} - \frac{1}{3} \begin{pmatrix} K_{l-1} & 0 \\ 0 & 0 \end{pmatrix} \right) \begin{pmatrix} \underline{u}_{l,v}^{(k,1)} \\ \underline{u}_{l,m}^{(k,1)} \end{pmatrix} \right] \\
&= \begin{pmatrix} I_{l,v} & S_{l,mv}^T \end{pmatrix} (\underline{f}_l^q - K_l^q \underline{u}_l^{(k,1)}) \;=\; I_l^{l-1}(\underline{f}_l^q - K_l^q \underline{u}_l^{(k,1)}).
\end{aligned}
$$

Because of the equivalence of the relations (57) and (68) we can replace in Algorithm 1 the smoothing steps (62) and (66) by the equivalent steps

$$\underline{u}_l^{(k,1)} = G_l^V(\underline{u}_l^{(k,0)}, K_l^q, \underline{f}_l^q) \quad \text{and} \quad \underline{u}_l^{(k,3)} = G_l^N(\underline{u}_l^{(k,2)}, K_l^q, \underline{f}_l^q).$$

Furthermore, we can see from equation (69) that the computation of the defect (63) is equivalent to

$$\underline{d}_{l-1}^{(k)} = I_l^{l-1}(\underline{f}_l^q - K_l^q \underline{u}_l^{(k,1)}).$$

Therefore Algorithm 1 can be interpreted as an usual multigrid algorithm for solving the system $K_l^q \underline{u}_l = \underline{f}_l^q$ of algebraic finite element equations resulting from a discretization with piecewise quadratic functions. According to this interpretation we can formulate Algorithm 1 in a more abstract form. If we use a decomposition of the FE space $V_l^q$, i.e.

$$V_l^q = \hat{V}_l^q = V_{l-1} + T_l \ , \qquad T_l = \operatorname{span}\{q_{l-1}^{(i)} \, , \, i = N_{l-1}+1, \ldots, N_l\} \qquad (70)$$

we get the following equivalent algorithm

*Algorithm 1′*

Let an initial guess $u_l^{(k,0)} \in V_l$ be given.

1. pre–smoothing

$$
\begin{aligned}
(71)\qquad &\text{Determine} \quad u_l^{(k,1)} \in u_l^{(k,0)} + T_l \;:\; \|u_l^{(k,1)} - u_{l,*}^{(k,1)}\| \le \rho_1 \|u_l^{(k,0)} - u_{l,*}^{(k,1)}\| \\
&\text{where} \quad u_{l,*}^{(k,1)} \in u_l^{(k,0)} + T_l \;:\; a(u_{l,*}^{(k,1)}, v) = \langle F, v\rangle \quad \text{for all } v \in T_l
\end{aligned}
$$

2. coarse–grid correction

$$
\begin{aligned}
(72)\qquad &\text{Determine} \quad u_l^{(k,2)} \in u_l^{(k,1)} + V_{l-1} \;:\; \|u_l^{(k,2)} - u_{l,*}^{(k,2)}\| \le \rho_2 \|u_l^{(k,1)} - u_{l,*}^{(k,2)}\| \\
&\text{where} \quad u_{l,*}^{(k,2)} \in u_l^{(k,1)} + V_{l-1} \;:\; a(u_{l,*}^{(k,2)}, v) = \langle F, v\rangle \quad \text{for all } v \in V_{l-1}
\end{aligned}
$$

3. post–smoothing

$$(73) \qquad \begin{array}{llll} \text{Determine} & u_l^{(k,3)} \in u_l^{(k,2)} + T_l & : & \|u_l^{(k,3)} - u_{l,*}^{(k,3)}\| \le \rho_3 \|u_l^{(k,2)} - u_{l,*}^{(k,3)}\| \\ \text{where} & u_{l,*}^{(k,3)} \in u_l^{(k,2)} + T_l & : & a(u_{l,*}^{(k,3)}, v) = \langle F, v\rangle \quad \text{for all} \;\; v \in T_l \end{array}$$

Set $u_l^{(k+1,0)} = u_l^{(k,3)}$.

In [19] SCHIEWECK has proved the following convergence result for this type of multigrid algorithm

$$\|u_l^{(k+1,0)} - u_l\| \le \eta \, \|u_l^{(k,0)} - u_l\| \, , \tag{74}$$

where

$$\eta = \rho_2 + (1-\rho_2)[\rho_1 + (1-\rho_1)\gamma][\rho_3 + (1-\rho_3)\gamma] \, , \tag{75}$$

$\|.\|^2 = a(.,.)$, and $u_l$ is the solution of the problem:
Find $u_l \in V_l \; : \; a(u_l, v_l) = \langle F, v_l\rangle \quad \text{for all} \;\; v_l \in V_l$,
and $\gamma$ is the constant in the strengthened Cauchy inequality

$$|a(v_l, w_{l-1})| \le \gamma \|v_l\| \, \|w_{l-1}\| \quad \text{for all} \;\; v_l \in T_l, \; \text{for all} \;\; w_{l-1} \in V_{l-1} \, . \tag{76}$$

Using this result we can prove the following convergence theorem for Algorithm 1.

THEOREM 3.1. *Let the smoothing procedures, the restriction, and the interpolation operators be defined as at the beginning of this Section. Then*

*(i) Algorithm 1 converges to an approximate solution of problem (1) which has the same discretization error as a piecewise quadratic FE solution.*

*(ii) The convergence estimate*

$$\|\underline{u}_l^{(k+1,0)} - \underline{u}_l\|_* \le \eta \|\underline{u}_l^{(k,0)} - \underline{u}_l\|_* \tag{77}$$

*holds, where* $\|.\|_*^2 = ((\frac{4}{3}K_l^l - \frac{1}{3}\tilde{K}_{l-1})\,.\,,\,.)$ *and* $\underline{u}_l$ *is the solution of the system of algebraic FE equations*

$$\left(\frac{4}{3}K_l^l - \frac{1}{3}\tilde{K}_{l-1}\right)\underline{u}_l = \left(\frac{4}{3}\underline{f}_l^l - \frac{1}{3}\underline{\tilde{f}}_{l-1}\right) \, .$$

*The convergence rate* $\eta$ *depends on the number of iteration steps for solving the systems (57), on the convergence rate of the* $(l-1)$*–grid algorithm used in step 2(b), and on the constant in the strengthened Cauchy inequality (76).*

*Proof:*

(i) This follows from the interpretation of Algorithm 1 as an usual multigrid algorithm for solving the FE system $K_l^q \underline{u}_l = \underline{f}_l^q$, immediately.

(ii) The convergence estimate (77) follows from estimate (75), because Algorithm 1 is equivalent to Algorithm 1′.
From [1] we know that the matrices $K_{l,mm}^l$ and $K_{l,mm}^q$ have a condition number which is independent of the discretization parameter. Therefore $\rho_1$ and $\rho_3$ in (71) and (73), respectively, do not depend on the discretization parameter. If additionally the convergence rate of the $(l-1)$–grid algorithm for solving the system (64) is independent of the discretization parameter $h_{l-1}$, then we get a $h_l$–independent convergence rate $\eta$ of the Algorithm 1. ∎

REMARK 3.1. The strengthened Cauchy inequality (76) for various bilinear forms $a(.,.)$ was analysed by many authors [1, 2, 4, 8, 11, 12, 19, 21]. MAITRE and MUSY [12] calculated the constant $\gamma$ for bilinear forms corresponding to scalar partial differential equations of second order. JUNG [8] and JUNG/LANGER/SEMMLER [11] studied the dependence of $\gamma$ on the Poisson ratio for linear elasticity problems in two- and three dimension.

REMARK 3.2. For different bilinear forms the dependence of $\rho_1$ and $\rho_3$ on problem specific parameters is studied in [8, 11, 19].

REMARK 3.3. The statements of Theorem 3.1 can also be proved for Algorithm 1 applied to FE equations resulting from the discretization of plane linear elasticity problems. To get these results we must prove the statements of Lemma 2.1 and Lemma 2.2 for the related matrices $\hat{K}_l^l$, $K_{l-1}$, and $\hat{K}_l^q$. These proofs are similar to the proofs given in Section 2. In Section 5 we will show some numerical experiments for plane linear elasticity problems.

REMARK 3.4. We can also use Algorithm 1 as preconditioner. The starting point is the PCCG method for solving the system of algebraic equations

$$\left(\frac{4}{3}K_l^l - \frac{1}{3}\tilde{K}_{l-1}\right)\underline{u}_l = \left(\frac{4}{3}\underline{f}_l^l - \frac{1}{3}\underline{\tilde{f}}_{l-1}\right) . \tag{78}$$

Since the matrix of the system of equations (78) is only used for matrix by vector multiplications within the PCCG method it is not necessary to assemble the matrix $\left(\frac{4}{3}K_l^l - \frac{1}{3}\tilde{K}_{l-1}\right)$. Also the right–hand side is needed for the computation of the defect in the initial step of the PCCG method only. Therefore, we can perform all operations of the PCCG method using the matrices $K_l^l$, $K_{l-1}$ and the right–hand sides $\underline{f}_l^l$ and $\underline{f}_{l-1}$. A priori we choose the matrix $\tilde{C}_l = \left(\frac{4}{3}K_l^l - \frac{1}{3}\tilde{K}_{l-1}\right)$ as preconditioner and solve the preconditioning systems $\tilde{C}_l\underline{w}_l = \underline{r}_l$ within the PCCG algorithm by means of the Algorithm 1. This approach we can interpret as a preconditioning with the matrix

$$C_l = \left(\frac{4}{3}K_l^l - \frac{1}{3}\tilde{K}_{l-1}\right)(I_l \; - M_l^m)^{-1} , \tag{79}$$

where $M_l^m$ is the error transmission operator of the Algorithm 1. We have to check whether the matrix $C_l$ is symmetric and positive definite. In [10] some conditions for the smoothing procedures, the restriction, and the interpolation operators are given, which guarantee these properties. If the conditions

$$\tilde{C}_l \; G_l^V = (G_l^N)^T \tilde{C}_l \quad , \tag{80}$$

where $G_l^V$ and $G_l^N$ are the error transmission operators of the smoothing procedures,

$$I_{l-1}^l = (I_l^{l-1})^T, \qquad \text{and} \tag{81}$$

$$K_{l-1}M_{l-1} = (M_{l-1})^T K_{l-1} , \tag{82}$$

where $M_{l-1}$ is the error transmission operator of the $(l-1)$–grid algorithm for solving system (64), are fulfilled then the matrix $C_l$ is a symmetric, positive definite one.

The pre–smoothing procedure introduced at the begin of this Section can be written in the following matrix form

$$\begin{pmatrix} \underline{u}_{l,v}^{(j+1)} \\ \underline{u}_{l,m}^{(j+1)} \end{pmatrix} = \begin{pmatrix} I_{l,v} & 0 \\ -(I_{l,m}-M_{l,m})K_{l,mm}^{-1}K_{l,mv} & I_{l,m}-(I_{l,m}-M_{l,m})K_{l,mm}^{-1}K_{l,mm} \end{pmatrix} \begin{pmatrix} \underline{u}_{l,v}^{(j)} \\ \underline{u}_{l,m}^{(j)} \end{pmatrix}$$

$$+ \begin{pmatrix} 0 & 0 \\ 0 & (I_{l,m}-M_{l,m})K_{l,mm}^{-1} \end{pmatrix} \begin{pmatrix} \underline{f}_{l,v} \\ \underline{f}_{l,m} \end{pmatrix}$$

$$(83) \qquad = \left\{ \left[ \begin{pmatrix} I_{l,v} & 0 \\ 0 & I_{l,m} \end{pmatrix} - \begin{pmatrix} 0 & 0 \\ -B_{l,mm}^{-1}K_{l,mv} & -B_{l,mm}^{-1}K_{l,mm} \end{pmatrix} \right] \begin{pmatrix} \underline{u}_{l,v}^{(j)} \\ \underline{u}_{l,v}^{(j)} \end{pmatrix} \right.$$

$$\left. + \begin{pmatrix} 0 & 0 \\ 0 & B_{l,mm}^{-1} \end{pmatrix} \begin{pmatrix} \underline{f}_{l,v} \\ \underline{f}_{l,m} \end{pmatrix} \right\}$$

$$= (I_l - B_l^{-1}K_l)\underline{u}_l^{(j)} + B_l^{-1}\underline{f}_l = G_l^V \underline{u}_l^{(j)} + B_l^{-1}\underline{f}_l ,$$

where

$$B_l^{-1} = \begin{pmatrix} 0 & 0 \\ 0 & B_{l,mm}^{-1} \end{pmatrix}$$

In an analog way we get for the post–smoothing

$$(84) \qquad \underline{u}_l^{(j+1)} = (I_l - B_l^{-T}K_l)\underline{u}_l^{(j)} + B_l^{-T}\underline{f}_l = G_l^N \underline{u}_l^{(j)} + B_l^{-T}\underline{f}_l ,$$

where

$$B_l^{-T} = \begin{pmatrix} 0 & 0 \\ 0 & B_{l,mm}^{-T} \end{pmatrix} .$$

Now we have

$$\begin{aligned} \tilde{C}_l G_l^V &= \tilde{C}_l - \tilde{C}_l B_l^{-1}K_l^l = \tilde{C}_l - \tfrac{4}{3}K_l^l B_l^{-1} K_l^l \\ &= \tilde{C}_l - K_l^l B_l^{-1}\tilde{C}_l = (I_l - K_l^l B_l^{-1})\tilde{C}_l = (G_l^N)^T \tilde{C}_l \quad , \end{aligned}$$

i.e. the condition (80) is fulfilled. The interpolation and restriction operators $I_{l-1}^l$ and $I_l^{l-1}$ we have defined in (58) – (60) such that condition (81) holds immediately. Condition (82) is fulfilled if the smoothing iterations, the interpolation and the restriction operators within the $(l-1)$–grid algorithm satisfy conditions analogous to (80) and (81) (see, also [10]).
Hence we know that the matrix $C_l$ is symmetric and positive definite. Furthermore, the spectral equivalence inequality

$$(85) \qquad (1-\eta^m)(C_l\underline{v}_l, \underline{v}_l) \le (\tilde{C}_l\underline{v}_l, \underline{v}_l) \le (C_l\underline{v}_l, \underline{v}_l) \qquad \text{for all } \underline{v}_l \in \mathbb{R}^{N_l}$$

holds. Therefore the number of iterations of the PCCG method needed to get an approximate solution with an relative accuracy $\varepsilon$ depends on the convergence factor of Algorithm 1. If the convergence factor of the $(l-1)$-grid method for solving the coarse grid system (64) is independent of the discretization parameter $h$ then the number of iterations of this PCCG method is independent of $h$.

REMARK 3.5. Using Theorem 2.3 and 2.6, we can also prove the convergence of an algorithm similar to YSERENTANT's PCCG method with an hierarchical preconditioner for FE schemes with piecewise linear elements. First, let us consider the system

$$\left(\frac{4}{3}\hat{K}_l^l - \frac{1}{3}\tilde{K}_{l-1}\right)\underline{\hat{u}}_l = \left(\frac{4}{3}\underline{\hat{f}}_l^l - \frac{1}{3}\underline{\tilde{f}}_{l-1}\right) \tag{86}$$

or the equivalent system

$$\hat{K}_l^q \underline{\hat{u}}_l = \underline{\hat{f}}_l^q . \tag{87}$$

We solve system (86) or (87) by means of the PCCG method with the preconditioner

$$\hat{C}_l = \begin{pmatrix} Q_{l-1}^{-T} C_{l-1} Q_{l-1}^{-1} & 0 \\ 0 & \mathrm{diag}(K_{l,mm}) \end{pmatrix}, \tag{88}$$

where $\mathrm{diag}(K_{l,mm}) = \mathrm{diag}\hat{K}_{l,mm}^q$ or $\mathrm{diag}(K_{l,mm}) = \frac{4}{3}\mathrm{diag}\hat{K}_{l,mm}^l$,

$$C_{l-1} = \begin{pmatrix} K_1 & 0 \\ 0 & I \end{pmatrix}, \quad Q_{l-1} = \hat{Q}_{l-1}\hat{Q}_{l-2}\cdots\hat{Q}_2.$$

The matrices $\hat{Q}_k,\ k = l-1,\ldots,2$ are defined in an analogous way as the matrix $S_l$ in (47). YSERENTANT [22] has shown the spectral equivalence inequality

$$\begin{aligned} \underline{c}\, l^{-2}(Q_{l-1}^{-T} C_{l-1} Q_{l-1}^{-1}\underline{v}_{l-1}, \underline{v}_{l-1}) &\le (K_{l-1}\underline{v}_{l-1}, \underline{v}_{l-1}) \\ &\le \overline{c}(Q_{l-1}^{-T} C_{l-1} Q_{l-1}^{-1}\underline{v}_{l-1}, \underline{v}_{l-1}) \quad \text{for all } \underline{v}_{l-1} \in \mathbb{R}^{N_{l-1}}, \end{aligned} \tag{89}$$

with constants $\underline{c}$ and $\overline{c}$ which do not depend on the discretization parameter. Furthermore, it can be shown (see [11])

$$\begin{aligned} \underline{\gamma}_m(\mathrm{diag}(K_{l,mm})\underline{v}_{l,m}, \underline{v}_{l,m}) &\le (K_{l,mm}\underline{v}_{l,m}, \underline{v}_{l,m}) \\ &\le \overline{\gamma}_m(\mathrm{diag}(K_{l,mm})\underline{v}_{l,m}, \underline{v}_{l,m}) \text{ for all } \underline{v}_{l,m} \in \mathbb{R}^{N_l - N_{l-1}}. \end{aligned} \tag{90}$$

From (89), (90), and the strengthened Cauchy inequality (76) follows immediately

$$\begin{aligned} (1-\gamma)\min\{\underline{\gamma}_m, \underline{c}\, l^{-2}\}(\hat{C}_l\,\underline{v}_l, \underline{v}_l) &\le (\hat{K}_l^q\underline{v}_l, \underline{v}_l) \\ &\le (1+\gamma)\max\{\overline{\gamma}_m, \overline{c}\}(\hat{C}_l\,\underline{v}_l, \underline{v}_l) \text{ for all } \underline{v}_l \in \mathbb{R}^{N_l}. \end{aligned} \tag{91}$$

Instead of solving the systems (86), (87) we can also solve the systems in the nodal basis

$$\left(\frac{4}{3}K_l^l - \frac{1}{3}\tilde{K}_{l-1}\right)\underline{u}_l = \left(\frac{4}{3}\underline{f}_l^l - \frac{1}{3}\underline{\tilde{f}}_{l-1}\right) \tag{92}$$

or, equivalently

$$K_l^q \underline{u}_l = \underline{f}_l^q, \tag{93}$$

by means of a PCCG method with the preconditioner

$$C_l = S_l^{-T}\hat{C}_l\, S_l^{-1}. \tag{94}$$

In this case we get the spectral equivalence inequality

$$
\begin{aligned}
(95)\,(1-\gamma)\min\{\underline{\gamma}_m, \underline{c}l^{-2}\}(C_l\,\underline{v}_l\,,\underline{v}_l\,) &\leq (K_l^q\underline{v}_l\,,\underline{v}_l\,) \\
&\leq (1+\gamma)\max\{\overline{\gamma}_m, \overline{c}\}(C_l\,\underline{v}_l\,,\underline{v}_l\,) \ \text{ for all } \underline{v}_l \in \mathbb{R}^{N_l},
\end{aligned}
$$

i.e. the number of PCCG iterations needed for solving the systems (92) or (93) is of the order $O(\log(h_{l-1}^{-1})\log(\epsilon^{-1}))$.

**4. An analysis of the number of arithmetical operations needed for the generation of the FE systems and the matrix–vector multiplication.** In the previous Section we have seen that Algorithm 1 can be interpreted as an usual multigrid algorithm for solving the system $K_l^q\underline{u}_l = \underline{f}_l^q$. Furthermore, we can formulate Algorithm 1 in terms of the $h$– or $p$–hierarchical basis, i.e. we have four possibilities for an implementation of this algorithm. To give an answer which implementation of the algorithm will be the most efficient with respect to the arithmetical work we analyse the number of arithmetical operations needed for the generation of the FE systems

$$
\left(\frac{4}{3}K_l^l - \frac{1}{3}\tilde{K}_{l-1}\right)\underline{u}_l = \left(\frac{4}{3}\underline{f}_l^l - \frac{1}{3}\underline{\tilde{f}}_{l-1}\right) \quad , \qquad \left(\frac{4}{3}\hat{K}_l^l - \frac{1}{3}\tilde{K}_{l-1}\right)\underline{\hat{u}}_l = \left(\frac{4}{3}\underline{\hat{f}}_l^l - \frac{1}{3}\underline{\tilde{f}}_{l-1}\right),
$$

$$
K_l^q\underline{u}_l = \underline{f}_l^q\,, \qquad\qquad \text{and} \qquad\qquad \hat{K}_l^q\underline{\hat{u}}_l = \underline{\hat{f}}_l^q\,.
$$

Additionally, we compare the number of operations needed for a matrix by vector multiplication.

The stiffness matrices and load vectors are computed element by element. Therefore, first we analyse the arithmetical work for the generation of an element stiffness matrix and an element load vector. The entries of the element stiffness matrices are defined by

$$
K_q^{(r),(\alpha\beta)} = \int\limits_{\Delta} \left(A(J_q^{(r)})^{-T}\nabla_\xi\tilde{\varphi}_\beta(\xi), (J_q^{(r)})^{-T}\nabla_\xi\tilde{\varphi}_\alpha(\xi)\right)|\det J_q^{(r)}|\,d\xi\,.
$$

For simplicity we consider the first boundary value problem, i.e. $\Gamma_N = \emptyset$, such that the entries of the element load vectors are given by

$$
f_q^{(r),(\alpha)} = \int\limits_{\Delta} f(x(\xi))\tilde{\varphi}_\alpha(\xi)|\det J_q^{(r)}|\,d\xi\,,
$$

where $\tilde{\varphi}_\alpha$, $\tilde{\varphi}_\beta$ stand for the shape functions corresponding to the piecewise linear nodal basis, the $h$–hierarchical basis, the piecewise quadratic nodal basis, or the $p$–hierarchical basis and $q = l-1$ or $q = l$. Using the representation

$$
(J_q^{(r)})^{-T} = \frac{1}{\det J_q^{(r)}}\begin{pmatrix} x_2^{(r,3)} - x_2^{(r,1)} & x_2^{(r,1)} - x_2^{(r,2)} \\ x_1^{(r,1)} - x_1^{(r,3)} & x_1^{(r,2)} - x_1^{(r,1)} \end{pmatrix} = \frac{1}{\det J_q^{(r)}}(\bar{J}_q^{(r)})^{-T}
$$

(see also (22)) we get

$$
K_q^{(r),(\alpha\beta)} = \frac{1}{|\det J_q^{(r)}|}\int\limits_{\Delta}\left(A(\bar{J}_q^{(r)})^{-T}\nabla_\xi\tilde{\varphi}_\beta(\xi), (\bar{J}_q^{(r)})^{-T}\nabla_\xi\tilde{\varphi}_\alpha(\xi)\right)\,d\xi\,.
$$

For the computation of $(\bar{J}_q^{(r)})^{-T}\nabla_\xi\tilde{\varphi}_\alpha$ and $\det J_q^{(r)}$ we have to calculate the differences of the coordinates

$$
\begin{aligned}
&x_1^{(32)} = x_1^{(r,3)} - x_1^{(r,2)}, \quad x_1^{(13)} = x_1^{(r,1)} - x_1^{(r,3)}, \quad x_1^{(21)} = x_1^{(r,2)} - x_1^{(r,1)}, \\
&x_2^{(23)} = x_2^{(r,2)} - x_2^{(r,3)}, \quad x_2^{(31)} = x_2^{(r,3)} - x_2^{(r,1)}, \quad x_2^{(12)} = x_2^{(r,1)} - x_2^{(r,2)},
\end{aligned} \tag{96}
$$

which requires $Q_{diff}^{+} = 6$ additions and for the computation of $\det J_q^{(r)}$ we need additionally $Q_J^{*} = 2$ multiplications and $Q_J^{+} = 1$ addition.
Next we discuss the arithmetical work for the generation of the matrix $\left(\frac{4}{3}K_l^l - \frac{1}{3}\tilde{K}_{l-1}\right)$. In this case we have to compute the element stiffness matrices

$$
K_q^{l,(r)} = \left[\frac{1}{|\det J_q^{(r)}|}\int_\Delta \left(A(\bar{J}_q^{(r)})^{-T}\nabla_\xi\varphi_\beta(\xi), (\bar{J}_q^{(r)})^{-T}\nabla_\xi\varphi_\alpha(\xi)\right)\, d\xi\right]_{\alpha,\beta=1}^{3},
$$

where the functions $\varphi_\alpha$, $\varphi_\beta$ are defined by the relations (23) and $q = l, l-1$, respectively. A simple calculation leads to

$$
\begin{aligned}
(\bar{J}_q^{(r)})^{-T}\nabla_\xi\varphi_1 &= (x_2^{(23)} \quad x_1^{(32)})^T, \\
(\bar{J}_q^{(r)})^{-T}\nabla_\xi\varphi_2 &= (x_2^{(31)} \quad x_1^{(13)})^T, \\
(\bar{J}_q^{(r)})^{-T}\nabla_\xi\varphi_3 &= (x_2^{(12)} \quad x_1^{(21)})^T.
\end{aligned} \tag{97}
$$

Because the element stiffness matrices $K_q^{l,(r)}$ are symmetric we have to calculate six entries per element stiffness matrix. We need $Q_{f1}^{*} = 1$ multiplication for the computation of $2|\det J_q^{(r)}|$. The element $K_q^{l,(r),(11)}$ of the element stiffness matrix is equal to

$$
\frac{1}{2|\det J_q^{(r)}|}\left[a_{11}\, x_2^{(23)}x_2^{(23)} + 2a_{12}\, x_2^{(23)}x_1^{(32)} + a_{22}\, x_1^{(32)}x_1^{(32)}\right],
$$

where $a_{ij}$ are the entries of the symmetric matrix $A$. The elements $K_q^{l,(r),(22)}$ and $K_q^{l,(r),(33)}$ can be computed in the same way. If we suppose that $2a_{12}$ is computed at the begin of the generation process, then we need for the computation of such an element

$$
Q_{e,1}^{+} = 2 \quad \text{additions} \qquad \text{and} \qquad Q_{e,1}^{*} = 7 \quad \text{multiplications}. \tag{98}
$$

For the element $K_q^{l,(r),(12)}$ we have

$$
K_q^{l,(r),(12)} = \frac{1}{2|\det J_q^{(r)}|}\left[a_{11}\, x_2^{(31)}x_2^{(23)} + a_{12}\,[x_1^{(13)}x_2^{(23)} + x_2^{(31)}x_1^{(32)}] + a_{22}\, x_1^{(13)}x_1^{(32)}\right].
$$

The elements $K_q^{l,(r),(13)}$ and $K_q^{l,(r),(23)}$ are defined in the same way. Therefore we need for the computation of each element $K_q^{l,(r),(\alpha\beta)}$, $(\alpha\beta) \in \{(12),(13),(23)\}$,

$$
Q_{e,2}^{+} = 3 \quad \text{additions} \qquad \text{and} \qquad Q_{e,2}^{*} = 8 \quad \text{multiplications}. \tag{99}
$$

To add an element stiffness matrix to the global stiffness matrix we need $Q_{e,add}^{+} = 6$ additions. Consequently, the total work for the generation of an element stiffness matrix are

$$
\begin{aligned}
Q_e^{+} &= Q_{diff}^{+} + Q_J^{+} + 3Q_{e,1}^{+} + 3Q_{e,2}^{+} + Q_{e,add}^{+} = 28 \\
Q_e^{*} &= Q_J^{*} + 3Q_{e,1}^{*} + 3Q_{e,2}^{*} + Q_{f1}^{*} = 48.
\end{aligned} \tag{100}
$$

Because we assumed that $f$ is constant over triangles $\delta_{l-1}^{(r)} \in \mathcal{T}_{l-1}$ (see Section 2) we get for the entries of the element load vectors

$$f_q^{l,(r),(\alpha)} = \frac{1}{6} f_{|_{\delta_q^{(r)}}} |\det J_q^{(r)}| \qquad \alpha = 1,2,3,$$

i.e. we have to perform $Q_f^* = 2$ multiplications to get all entries of the element load vector and for the addition of the element load vector to the global load vector we need $Q_{f,add}^+ = 3$ additions. The element stiffness matrices corresponding to the elements $\delta_{l-1}^{(r)}$ and $\delta_l^{(k)}$ with $\bar{\delta}_{l-1}^{l,(r)} = \cup_k \bar{\delta}_l^{l,(k)}$ are the same and the entries of the element load vectors $\underline{f}_{l-1}^{l,(r)}$, $\underline{f}_l^{l,(k)}$ differ only by the factor $\frac{1}{4}$. Therefore, the total arithmetical work for the generation of the stiffness matrices $K_{l-1}$, $K_l^l$ and the load vectors $\underline{f}_{l-1}$, $\underline{f}_l^l$ are

$$31R_{l-1} + 4(Q_{e,add}^+ + Q_{f,add}^+)R_{l-1} \quad \text{additions} \quad \text{and} \quad 50R_{l-1} + R_{l-1} \quad \text{multiplications},$$

i.e.

$$(101) \qquad 67R_{l-1} \quad \text{additions} \quad \text{and} \quad 51R_{l-1} \quad \text{multiplications},$$

where $R_{l-1}$ denotes the number of triangles of $\mathcal{T}_{l-1}$.

The element stiffness matrices corresponding to the $h$–hierarchical basis are

$$\hat{K}_l^{l,(r)} = \left[ \frac{1}{|\det J_{l-1}^{(r)}|} \int_\Delta \left( A(\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \varphi_\beta(\xi), (\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \varphi_\alpha(\xi) \right) d\xi \right]_{\alpha,\beta=1}^{6},$$

where $\varphi_\alpha$, $\varphi_\beta$ are defined by the relations (23). For the computation of the matrix elements $\hat{K}_l^{l,(r),(\alpha\beta)}$, $(\alpha\beta) \in \{(11),(12),(13),(21),(22),(23),(31),(32),(33)\}$ the arithmetical work is given by (100). For the matrix elements $\hat{K}_l^{l,(r),(\alpha\beta)}$, $(\alpha\beta) \in \{(14),(15),(16),(24),(25),(26),(34),(35),(36)\}$ we have

$$\begin{aligned}
\hat{K}_l^{l,(r),(\alpha\beta)} &= \sum_{k\in I(\beta)} \frac{1}{|\det J_{l-1}^{(r)}|} \int_{\Delta^{(k)}} \left( A(\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \varphi_\beta(\xi), (\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \varphi_\alpha(\xi) \right) d\xi \\
&= \frac{1}{8|\det J_{l-1}^{(r)}|} \sum_{k\in I(\beta)} \left( A(\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \varphi_\beta(\xi)_{|_{\Delta^{(k)}}}, (\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \varphi_\alpha(\xi)_{|_{\Delta^{(k)}}} \right),
\end{aligned}$$

where $I(4) = \{1,2,4\}$, $I(5) = \{2,3,4\}$, and $I(6) = \{1,3,4\}$ (see also Section 2). Using the special structure of $(J_{l-1}^{(r)})^{-T} \nabla_\xi \varphi_\alpha$, $\alpha = 1,2,3$, (see (97)) and $(J_{l-1}^{(r)})^{-T} \nabla_\xi \varphi_\beta$, $\beta = 4,5,6$, from Table 3 we see that $\hat{K}_l^{l,(r),(14)} = \hat{K}_l^{l,(r),(35)}$, $\hat{K}_l^{l,(r),(16)} = \hat{K}_l^{l,(r),(25)}$, and $\hat{K}_l^{l,(r),(24)} = \hat{K}_l^{l,(r),(36)}$. Consequently, the additional arithmetical work for the calculation of these 9 elements are

$$(102) \qquad 9Q_{e,2}^+ + 6Q_{s2}^+ \quad \text{additions} \quad \text{and} \quad 9Q_{e,2}^* + Q_{f2}^* \quad \text{multiplications},$$

where $Q_{s2}^+ = 2$ is the arithmetical work for the summation over $k \in I(\beta)$ and $Q_{f2}^* = 1$ is the multiplication $4|\det J_{l-1}^{(r)}|$.

Using again the vectors $(J_{l-1}^{(r)})^{-T} \nabla_\xi \varphi_\beta$, $\beta = 4,5,6$ from Table 3, it follows that

$$\hat{K}_l^{l,(r),(44)} = \hat{K}_l^{l,(r),(55)} = \hat{K}_l^{l,(r),(66)}.$$

TABLE 3
$(\bar{J}_{l-1}^{(r)})^{-T}\nabla_\xi\varphi_\alpha,\ \alpha=4,5,6$

| | $(\bar{J}_{l-1}^{(r)})^{-T}\nabla_\xi\varphi_4$ | $(\bar{J}_{l-1}^{(r)})^{-T}\nabla_\xi\varphi_5$ | $(\bar{J}_{l-1}^{(r)})^{-T}\nabla_\xi\varphi_6$ |
|---|---|---|---|
| $\Delta^{(1)}$ | $2(x_2^{(31)}\quad x_1^{(13)})^T$ | $(0\quad 0)^T$ | $2(x_2^{(12)}\quad x_1^{(21)})^T$ |
| $\Delta^{(2)}$ | $2(x_2^{(23)}\quad x_1^{(32)})^T$ | $2(x_2^{(12)}\quad x_1^{(21)})^T$ | $(0\quad 0)^T$ |
| $\Delta^{(3)}$ | $(0\quad 0)^T$ | $2(x_2^{(31)}\quad x_1^{(13)})^T$ | $2(x_2^{(23)}\quad x_1^{(32)})^T$ |
| $\Delta^{(4)}$ | $-2(x_2^{(12)}\quad x_1^{(21)})^T$ | $-2(x_2^{(23)}\quad x_1^{(32)})^T$ | $-2(x_2^{(31)}\quad x_1^{(13)})^T$ |

Therefore, the arithmetical operations needed for the computation of these elements are

$$(103)\qquad 3Q_{e,1}^{+}+Q_{s2}^{+}\quad\text{additions}\qquad\text{and}\qquad 3Q_{e,1}^{*}\quad\text{multiplications.}$$

For the matrix elements $\hat{K}_l^{l,(r),(\alpha\beta)}$, $(\alpha\beta)\in\{(45),(46),(56)\}$, we have

$$\hat{K}_l^{l,(r),(\alpha\beta)}=\frac{1}{|\mathrm{det}J_{l-1}^{(r)}|}\sum_{k\in I(\alpha)\cap I(\beta)}\int_{\Delta^{(k)}}\left(A(\bar{J}_{l-1}^{(r)})^{-T}\nabla_\xi\varphi_\beta(\xi)_{|_{\Delta^{(k)}}},(\bar{J}_{l-1}^{(r)})^{-T}\nabla_\xi\varphi_\alpha(\xi)_{|_{\Delta^{(k)}}}\right)d\xi,$$

where each sum consists of two identical summands. Therefore, we need for the computation of these elements

$$(104)\qquad 3Q_{e,2}^{+}\quad\text{additions}\qquad\text{and}\qquad 3Q_{e,2}^{*}\quad\text{multiplications}.$$

The addition of the element stiffness matrix into the global stiffness matrix requires

$$(105)\qquad Q_{e,add}^{+}=21$$

Consequently, we see from (100),(102),(103),(104), and (105) that the total arithmetical work for the generation of the element stiffness matrix in the $h$–hierarchical basis are

$$(106)\qquad \begin{array}{rcl} Q_e^{+} &=& Q_{diff}^{+}+Q_J^{+}+6Q_{e,1}^{+}+15Q_{e,2}^{+}+7Q_{s2}+Q_{e,add}^{+}=99\\ Q_e^{*} &=& Q_J^{*}+6Q_{e,1}^{*}+15Q_{e,2}^{*}+Q_{f1}^{*}+Q_{f2}^{*}=166\,. \end{array}$$

For the element load vector we have

$$\begin{array}{rclll} \underline{\hat{f}}_l^{l,(r),(\alpha)} &=& \frac{1}{6}f_{|_{\delta_{l-1}^{(r)}}}|\mathrm{det}J_{l-1}^{(r)}| & \text{for} & \alpha=1,2,3\quad\text{and}\\ \underline{\hat{f}}_l^{l,(r),(\alpha)} &=& \frac{1}{8}f_{|_{\delta_{l-1}^{(r)}}}|\mathrm{det}J_{l-1}^{(r)}| & \text{for} & \alpha=4,5,6 \end{array}$$

(see also (40)), i.e. we need $Q_f^{*}=3$ multiplications and for the addition of an element load vector to the global load vector $Q_{f,add}^{+}=6$ additions. Consequently, the total work for the generation of the global stiffness matrix $\hat{K}_l^l$ and the load vector $\underline{\hat{f}}_l^l$ is

$$(107)\qquad 105R_{l-1}\quad\text{additions}\qquad\text{and}\qquad 169R_{l-1}\quad\text{multiplications.}$$

In the case of the quadratic nodal basis the element stiffness matrices are defined by

$$K_l^{q,(r)} = \left[ \frac{1}{|\det J_{l-1}^{(r)}|} \int\limits_{\Delta} \left( A(\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \psi_\beta(\xi), (\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \psi_\alpha(\xi) \right) \, d\xi \right]_{\alpha,\beta=1}^{6} ,$$

where the functions $\psi_\alpha(\xi)$ and $\psi_\beta(\xi)$ are given by the formula (46). Because the integrands in these integrals are quadratic functions we will use the quadrature rule (25) for the computation of the matrix elements, i.e.

$$(108) \quad K_l^{q,(r)} = \left[ \frac{1}{6|\det J_{l-1}^{(r)}|} \sum_{k=1}^{3} \left( A(\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \psi_\beta(\xi^{(k)}), (\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \psi_\alpha(\xi^{(k)}) \right) \right]_{\alpha,\beta=1}^{6} .$$

The values of $(J_{l-1}^{(r)})^{-T} \nabla_\xi \psi_\alpha$, $\alpha = 1, 2, \ldots, 6$, in the quadrature points $\xi^{(k)}$ are given in Table 4.

TABLE 4
$(\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \psi_\alpha$, $\alpha = 1, 2, \ldots, 6$

| $\xi^{(k)}$ | $(\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \psi_1$ | $(\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \psi_2$ | $(\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \psi_3$ |
|---|---|---|---|
| $(0.5, 0)$ | $(x_2^{(23)} \quad x_1^{(32)})^T$ | $(x_2^{(31)} \quad x_1^{(13)})^T$ | $-\,(x_2^{(12)} \quad x_1^{(21)})^T$ |
| $(0, 0.5)$ | $(x_2^{(23)} \quad x_1^{(32)})^T$ | $-\,(x_2^{(31)} \quad x_1^{(13)})^T$ | $(x_2^{(12)} \quad x_1^{(21)})^T$ |
| $(0.5, 0.5)$ | $-\,(x_2^{(23)} \quad x_1^{(32)})^T$ | $(x_2^{(31)} \quad x_1^{(13)})^T$ | $(x_2^{(12)} \quad x_1^{(21)})^T$ |
| $\xi^{(k)}$ | $(\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \psi_4$ | $(\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \psi_5$ | $(\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \psi_6$ |
| $(0.5, 0)$ | $-2(x_2^{(12)} \quad x_1^{(21)})^T$ | $2(x_2^{(12)} \quad x_1^{(21)})^T$ | $2(x_2^{(12)} \quad x_1^{(21)})^T$ |
| $(0, 0.5)$ | $2(x_2^{(31)} \quad x_1^{(13)})^T$ | $2(x_2^{(31)} \quad x_1^{(13)})^T$ | $-2(x_2^{(31)} \quad x_1^{(13)})^T$ |
| $(0.5, 0.5)$ | $2(x_2^{(23)} \quad x_1^{(32)})^T$ | $-2(x_2^{(23)} \quad x_1^{(32)})^T$ | $2(x_2^{(23)} \quad x_1^{(32)})^T$ |

For the matrix elements $K_l^{q,(r),(\alpha\alpha)}$, $\alpha = 1, 2, 3$, the three summands in the sum over $k$ in (108) are the same such that we have

$$K_l^{q,(r),(\alpha\alpha)} = \frac{1}{2|\det J_{l-1}^{(r)}|} \left( A(\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \psi_\beta(\xi^{(1)}), (\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \psi_\alpha(\xi^{(1)}) \right) .$$

Consequently, we need

(109) $\quad 3Q_{e,1}^{+}$ additions and $3Q_{e,1}^{*} + Q_{f3}^{*}$ multiplications

with $Q_{f3}^{*} = 1$ for the computation of $2|\det J_{l-1}^{(r)}|$.
From the special structure of $(J_{l-1}^{(r)})^{-T} \nabla_\xi \psi_\alpha$ we see that the computation of the matrix elements $K_l^{q,(r),(\alpha\beta)}$, $(\alpha\beta) \in \{(12), (13), (23)\}$ requires

(110) $\quad 3Q_{e,2}^{+}$ additions and $3Q_{e,2}^{*} + Q_{f4}^{*}$ multiplications

where $Q_{f4}^{*} = 1$ is the arithmetical work for the computation of $6|\det J_{l-1}^{(r)}|$.

From Table Tables 4 we see that the amount of arithmetical work for the computation of the matrix elements $K_l^{q,(r),(\alpha\beta)}$, $(\alpha\beta) \in \{(14),(15),(16),(24),(25),(26),(34),(35),(36)\}$ is

$$(111) \qquad 9Q^+_{e,2} + 9Q^+_{s2} \quad \text{additions} \qquad \text{and} \qquad 9Q^*_{e,2} + Q^*_{f5} \quad \text{multiplications}$$

with $Q^*_{f5} = 1$ for the computation of $3|\det J_{l-1}^{(r)}|$.
Furthermore, we see that the arithmetical work for the computation of the elements $K_l^{q,(r),(\alpha\alpha)}$, $\alpha = 4,5,6$, is the same as in the case of the $h$–hierarchical basis, i.e

$$(112) \qquad 3Q^+_{e,1} + Q^+_{s2} \quad \text{additions} \qquad \text{and} \qquad 3Q^*_{e,1} + Q^*_{f6} \quad \text{multiplications}$$

with $Q^*_{f6} = 1$ for the computation of $\frac{3}{2}|\det J_{l-1}^{(r)}|$.
The computation of the elements $K_l^{q,(r),(\alpha\beta)}$, $(\alpha\beta) \in \{(45),(46),(56)\}$, requires additionally

$$(113) \qquad 3Q^+_{s2} \quad \text{additions}$$

With (109),(110),(111),(112),(113), the amount of arithmetical work $Q^+_{e,add} = 21$ , for the addition of the element stiffness matrix to the global matrix, the arithmetical work for the computation of the differences of the coordinates (96) ($Q^+_{diff} = 6$) and the work for the computation of $\det J_{l-1}^{(r)}$ ($Q^+_J = 1, Q^*_J = 2$) leads to the total arithmetical work

$$(114) \qquad \begin{array}{lllllllll} Q^+_e &=& Q^+_{diff} + Q^+_J &+ 6Q^+_{e,1} &+ 12Q^+_{e,2} &+ 13Q^+_{s2} &+ Q^+_{e,add} & & = 102 \\ Q^*_e &=& Q^*_J &+ 6Q^*_{e,1} &+ 12Q^*_{e,2} &+ Q^*_{f3} &+ Q^*_{f4} &+ Q^*_{f5} + Q^*_{f6} & = 144\,. \end{array}$$

For the element load vector we have

$$\underline{f}_l^{q,(r)} = \left[ \int_\Delta f \psi_\alpha(\xi) |\det J_{l-1}^{(r)}| \, d\xi \right]_{\alpha=1}^6 .$$

Using again quadrature rule (25) we get

$$\begin{aligned} \underline{f}_l^{q,(r)} &= \left[ f|_{\delta_{l-1}^{(r)}} |\det J_{l-1}^{(r)}| \frac{1}{6} \sum_{k=1}^{3} \psi(\xi^{(k)}) \right]_{\alpha=1}^6 \\ &= \begin{pmatrix} 0 & 0 & 0 & \frac{1}{6} f|_{\delta_{l-1}^{(r)}} |\det J_{l-1}^{(r)}| & \frac{1}{6} f|_{\delta_{l-1}^{(r)}} |\det J_{l-1}^{(r)}| & \frac{1}{6} f|_{\delta_{l-1}^{(r)}} |\det J_{l-1}^{(r)}| \end{pmatrix}^T , \end{aligned}$$

i.e. we need $Q^*_f = 2$ multiplications for the computation of $\frac{1}{6} f|_{\delta_{l-1}^{(r)}} |\det J_{l-1}^{(r)}|$ and $Q^+_{f,add} = 3$ additions for the addition of the element load vector to the global load vector. Consequently, the total amount of arithmetical work for the computation of the stiffness matrix $K_l^q$ and the load vector $\underline{f}_l^q$ is

$$(115) \qquad 105 R_{l-1} \quad \text{additions} \qquad \text{and} \qquad 146 R_{l-1} \quad \text{multiplications.}$$

In the case of the $p$–hierarchical basis we have to compute the matrix elements

$$(116) \qquad \hat{K}_l^{q,(r),(\alpha\beta)} = \frac{1}{|\det J_{l-1}^{(r)}|} \left( A(\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \varphi_\beta(\xi), (\bar{J}_{l-1}^{(r)})^{-T} \nabla_\xi \varphi_\alpha(\xi) \right)$$

$$\alpha, \beta = 1,2,3$$

$$(117)\quad \hat{K}_l^{q,(r),(\alpha\beta)} = \frac{1}{|\det J_{l-1}^{(r)}|}\Big(A(\bar{J}_{l-1}^{(r)})^{-T}\nabla_\xi\psi_\beta(\xi),(\bar{J}_{l-1}^{(r)})^{-T}\nabla_\xi\varphi_\alpha(\xi)\Big)$$

$$\alpha = 1,2,3,\ \beta = 4,5,6$$

$$(118)\quad \hat{K}_l^{q,(r),(\alpha\beta)} = \frac{1}{|\det J_{l-1}^{(r)}|}\Big(A(\bar{J}_{l-1}^{(r)})^{-T}\nabla_\xi\psi_\beta(\xi),(\bar{J}_{l-1}^{(r)})^{-T}\nabla_\xi\psi_\alpha(\xi)\Big)$$

$$\alpha,\beta = 4,5,6\,.$$

The arithmetical work for the computation of the matrix elements in (116) is given by (100) and for the matrix elements in (118) by (112) and (113).
The integrands in the formula (117) are linear functions such that we use the midpoint rule for the computation of this integrals, i.e.

$$\int_\Delta v(\xi_1,\xi_2)\,d\xi \approx \frac{1}{2}\,v\left(\frac{1}{3},\frac{1}{3}\right).$$

We have

$$(119)\quad \begin{aligned}(J_{l-1}^{(r)})^{-T}\nabla_\xi\psi_4(\tfrac{1}{3},\tfrac{1}{3}) &= -\tfrac{4}{3}\begin{pmatrix}x_2^{(12)} & x_1^{(21)}\end{pmatrix}^T\\ (J_{l-1}^{(r)})^{-T}\nabla_\xi\psi_5(\tfrac{1}{3},\tfrac{1}{3}) &= -\tfrac{4}{3}\begin{pmatrix}x_2^{(23)} & x_1^{(32)}\end{pmatrix}^T\\ (J_{l-1}^{(r)})^{-T}\nabla_\xi\psi_4(\tfrac{1}{3},\tfrac{1}{3}) &= -\tfrac{4}{3}\begin{pmatrix}x_2^{(31)} & x_1^{(13)}\end{pmatrix}^T.\end{aligned}$$

Using the special structure of (97) and (119) we see that we need for the computation of the 9 matrix elements defined in (117)

$$(120)\qquad 6Q_{e,2}^{+}\ \text{ additions} \quad \text{and} \quad 6Q_{e,2}^{*}\ \text{ multiplications}.$$

From (100),(112),(113), (120) and the arithmetical work $Q_{e,add} = 21$ for the addition of the element stiffness matrix to the global stiffness matrix we obtain the total arithmetical work

$$(121)\quad \begin{aligned} Q_e^+ &= Q_{diff}^+ + Q_J^+ + 6Q_{e,1}^+ + 9Q_{e,2} + 4Q_{s2}^+ + Q_{e,add}^+ = 75\\ Q_e^* &= Q_J^* + 6Q_{e,1}^* + 9Q_{e,2}^* + Q_{f1}^* + Q_{f6}^* = 118\,.\end{aligned}$$

For the element load vector we have

$$\underline{\hat{f}}_l^{q,(r)} = \left[\frac{1}{6}\,f|_{\delta_{l-1}^{(r)}}\ |\det J_{l-1}^{(r)}|\right]_{\alpha=1}^{6},$$

i.e. we need $Q_f^* = 2$ multiplications and $Q_{f,add}^+ = 6$ additions for the assembling process. Consequently, we need for the computation of the global stiffness matrix $\hat{K}_l^q$ and the global load vector $\underline{\hat{f}}_l^q$

$$(122)\qquad 81R_{l-1}\ \text{ additions} \quad \text{and} \quad 120R_{l-1}\ \text{ multiplications.}$$

From (101),(107),(115), and (122) it follows that the generation of the stiffness matrix and the load vector in the piecewise linear nodal basis is the cheapest. Furthermore, we see that the application of the $p$–hierarchical basis is more efficient with respect to the arithmetical work than the quadratic nodal basis.

Next, we will estimate the number of arithmetical operations required for the matrix-vector multiplications

$$\left(\frac{4}{3}K_l^l - \frac{1}{3}\tilde{K}_{l-1}\right)\underline{u}_l \tag{123}$$

or

$$\hat{K}_l^q \underline{u}_l\,. \tag{124}$$

For the sake of simplicity we suppose that the domain $\Omega$ is a rectangle and the triangulation consists of isosceles rectangular triangles. If we neglect the influence of the boundary, each row in the matrices $K_l^l$ and $K_{l-1}$, respectively, have 7 nonzero elements. Therefore, a matrix–vector multiplication with (123) requires approximately

$$7N_l + N_l + 7N_{l-1} + N_{l-1} = 8(N_l + N_{l-1}) \approx 10N_l$$

multiplications, and

$$6N_l + 6N_{l-1} + N_{l-1} = 6N_l + 7N_{l-1} \approx 31/4N_l$$

additions. The matrix (124) has $1/4N_l$ rows with 19 non–zero elements and $3/4N_l$ rows with 9 non–zero elements, i.e. the matrix–vector multiplication with the matrix $\hat{K}_l^q$ requires

$$19/4N_l + 27/4N_l = 23/2 \text{ multiplications}$$

and

$$18/4N_l + 24/4N_l = 21/2 \text{ additions.}$$

Thus the equivalent multiplication with the extrapolated matrix in (123) is slightly cheaper than the multiplication with the hierarchical quadratic matrix of (124). In total, for the generation of the matrices, as well as for performing Algorithm 1, the computational work for the extrapolated system based on $(K_l^l, K_{l-1}, \underline{f}_l^l, \underline{f}_{l-1})$ is smaller than for the quadratic system with $(\hat{K}_l^q, \hat{\underline{f}}_l^q)$.

**5. Numerical results.** In this Section we want to demonstrate that the Algorithm 1 converges to a FE solution with a discretization error in the same order as we obtain by a discretization with quadratic elements. Furthermore, we show that the convergence rate of Algorithm 1 is independent of the discretization parameter. We compare our results with a multigrid algorithm applied to FE equations resulting from a discretization with quadratic elements. All algorithms are implemented within the multigrid package FEMGP [9, 20]. The computations were performed on a PC 80486 (33 MHz) using the LAHEY–Fortran compiler.

Let us first consider the problem:

Find $u \in H_0^1(\Omega)$ such that

$$\int\limits_{\Omega} (A\nabla_x u, \nabla_x v)\, dx = \int\limits_{\Omega} fv\, dx \tag{125}$$

for all $v \in H_0^1(\Omega)$ holds,

where $\Omega = (0,1) \times (0,1)$, $A = \begin{pmatrix} 4 & 4 \\ 4 & 5 \end{pmatrix}$, and $f = \pi^2(9 \sin \pi x \sin \pi y - 8 \cos \pi x \cos \pi y)$. The exact solution of this problem is $u = \sin \pi x \sin \pi y$.

Because we want to compare Algorithm 1 with an algorithm for solving the FE equations obtained by using quadratic elements, we discretize problem (125) by means of the usual nodal basis of piecewise linear functions and by means of the $p$–hierarchical basis. An analysis of the arithmetical work for the generation of the FE systems shows that it is almost the same in both cases. Table 5 demonstrates this fact.

TABLE 5

*Comparison of the CPU–time needed for the generation of the FE systems*

| level $l$ | number of degrees of freedom | number of triangles of $\mathcal{T}_{l-1}$ | CPU–time for the generation of $K_{l-1}$, $K_l^l$, $\underline{f}_{l-1}$, $\underline{f}_l^l$ | CPU–time for the generation of $\hat{K}_l^q$, $\underline{\hat{f}}_l^q$ |
|---|---|---|---|---|
| 3 | 49 | 32 | 0.007 sec | 0.011 sec |
| 4 | 225 | 128 | 0.029 sec | 0.044 sec |
| 5 | 961 | 512 | 0.118 sec | 0.178 sec |
| 6 | 3969 | 2048 | 0.473 sec | 0.713 sec |
| 7 | 16129 | 8192 | 1.892 sec | 2.851 sec |

In the Table 6 the number of iterations and the CPU–time needed by the application of Algorithm 1 are given. Within the Algorithm 1 we used for the pre–smoothing (62) two iteration steps of the lexicographically forward Gauss–Seidel method, one iteration step of a $(l-1)$–grid algorithm for solving the coarse–grid system (64), and two iteration steps of the lexicographically backward Gauss–Seidel method for the post–smoothing (66). The results show that the number of iterations is independent of the discretization parameter. If we use one iteration step of Algorithm 1 as preconditioner in the PCCG method for solving the system

$$\left(\frac{4}{3}K_l^l - \frac{1}{3}\tilde{K}_{l-1}\right)\underline{u}_l = \left(\frac{4}{3}\underline{f}_l^l - \frac{1}{3}\underline{\tilde{f}}_{l-1}\right), \tag{126}$$

we get an algorithm with better convergence, the so–called MG(1)–PCCG method (see also Remark 3.4). For comparison we use Algorithm $1'$ as preconditioner in the PCCG method for solving the system

$$\hat{K}_l^q \underline{\hat{u}}_l^q = \underline{\hat{f}}_l^q \tag{127}$$

The results are presented in Table 6.

TABLE 6

*Comparison of the Algorithm 1, of the PCCG method (Algorithm 1 as preconditioner), and the PCCG method (Algorithm $1'$ as preconditioner). The algorithms are terminated, when the relative defect becomes smaller than $10^{-4}$.*

| $l$ | Algorithm 1 | | MG(1)–PCCG method for solving the systems (126) | | (127) | |
|---|---|---|---|---|---|---|
| | number of iterations | CPU–time | number of iterations | CPU–time | number of iterations | CPU–time |
| 3 | 13 | 0.11 sec | 5 | 0.06 sec | 5 | 0.05 sec |
| 4 | 14 | 0.54 sec | 6 | 0.28 sec | 6 | 0.28 sec |
| 5 | 14 | 2.36 sec | 6 | 1.15 sec | 6 | 1.10 sec |
| 6 | 14 | 9.83 sec | 6 | 4.83 sec | 6 | 4.66 sec |
| 7 | 14 | 41.58 sec | 6 | 19.83 sec | 6 | 19.49 sec |

Finally, we compare the discretization errors $u - u_l^l$ and $u - u_l^q$ in the $H^1$– and $L_2$–norm, respectively. Here $u_l$ denotes the FE solution obtained by means of Algorithm 1 and $u_l^q$ the FE solution by a discretization with piecewise quadratic functions. We remark that in our example the right–hand side $f$ is not constant on triangles $\delta_{l-1}^{(r)}$, which we had assumed in the proofs of the Theorems 2.3 and 2.6. Therefore, in our example the right–hand sides $\left(\frac{4}{3}\underline{f}_l^l - \frac{1}{3}\underline{\tilde{f}}_{l-1}\right)$ and $\underline{\hat{f}}_l^q$ are not identical. But the system (126) gives also a FE solution with almost the same discretization error as the system (127).

TABLE 7

*Comparison of the discretization errors*

| Level $l$ | $\Vert u - u_l^l\Vert_{H^1}$ | $\Vert u - u_l^q\Vert_{H^1}$ | $\Vert u - u_l^l\Vert_{L_2}$ | $\Vert u - u_l^q\Vert_{L_2}$ |
|---|---|---|---|---|
| 3 | 0.1306 | 0.1426 | 0.4074-02 | 0.4226-02 |
| 4 | 0.3347-01 | 0.3481-01 | 0.5404-03 | 0.5440-03 |
| 5 | 0.8426-02 | 0.8539-02 | 0.6850-04 | 0.6864-04 |
| 6 | 0.2110-02 | 0.2118-02 | 0.8577-05 | 0.8582-05 |
| 7 | 0.5278-03 | 0.5283-03 | 0.9328-06 | 0.9331-06 |

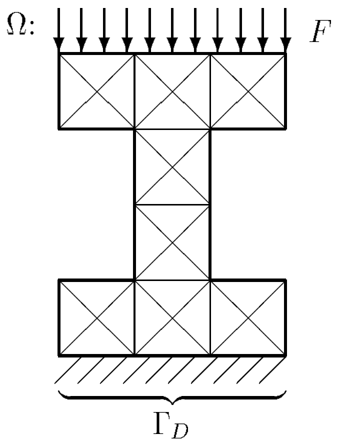
As second example we consider a linear elasticity problem, i.e.:
Find the displacement field $u = (u_1, u_2)^T \in V_0$, such that

$$\frac{E}{1+\nu}\int\limits_{\Omega}\left[\frac{\partial u_1}{\partial x_1}\frac{\partial v_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2}\frac{\partial v_2}{\partial x_2} + \frac{\nu}{1-\nu}\,\mathrm{div}u\,\mathrm{div}v \right. \tag{128}$$

$$\left. +\frac{1}{2}\Big(\frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1}\Big)\Big(\frac{\partial v_1}{\partial x_2} + \frac{\partial v_2}{\partial x_1}\Big)\right] dx = \int\limits_{\Gamma_N} g_{2,1}v_1 + g_{2,2}v_2\, ds$$

holds for all test functions $v \in V_0$.

Here $g_2 = (g_{2,1}, g_{2,2})^T$ denotes the surface tractions, $E$ is Young's elasticity modulus, and $\nu$ is the Poisson ratio. The space $V_0$ is defined by $V_0 = \{v \in [H^1(\Omega)]^2 : v_1(x) = v_2(x) = 0$ on $\Gamma_D\}$ and $\partial\Omega = \bar{\Gamma}_D \cup \bar{\Gamma}_N$, $\Gamma_D \cap \Gamma_N = \emptyset$.




$$
\begin{aligned}
E &= 196\ GPa \\
\nu &= 0.3 \\
g_{2,1} &= 0 \\
g_{2,2} &= \begin{cases} F = 1000\ N & \text{on the upper part of the boundary} \\ 0 & \text{otherwise} \end{cases}
\end{aligned}
$$

FIG. 3. *Shape of the domain and data for the test problem*

Again we compare the CPU–time needed for the generation of the FE systems in the nodal basis of piecewise linear functions and in the $p$–hierarchical basis.

TABLE 8
*Comparison of the CPU–time needed for the generation of the FE systems*

| level $l$ | number of degrees of freedom | number of triangles of $\mathcal{T}_{l-1}$ | CPU–time for the generation of $K_{l-1}$, $K_l^l$, $\underline{f}_{l-1}$, $\underline{f}_l^l$ | CPU–time for the generation of $\hat{K}_l^q$, $\hat{\underline{f}}_l^q$, |
|---|---|---|---|---|
| 3 | 586 | 128 | 0.27 sec | 0.24 sec |
| 4 | 2194 | 512 | 0.99 sec | 0.96 sec |
| 5 | 8492 | 2048 | 4.01 sec | 3.89 sec |

Furthermore, we give results concerning the application of the Algorithm 1 and its use as preconditioner in the PCCG method. We mention here, that the constant in the strengthened Cauchy inequality (76) is relatively large, namely $\gamma = 0.94$ (see [11]), and therefore the convergence of the Algorithm 1 is poor. In Table 9 we summarise some results for Algorithm 1 and we compare the MG(1)–PCCG method for the systems (126) and (127). Additionally, we compare these algorithms with the PCCG method discussed in Remark 3.5, i.e. the HB–PCCG method.
Finally, we compare the energy of the FE solutions obtained by solving the systems (126) and (127), respectively. From Table 10 we see that we have in both cases the same FE solution.

Table 9

*Comparison of the Algorithm 1, of the MG(1)–PCCG method (Algorithm 1 as preconditioner), the MG(1)–PCCG method (Algorithm 1′ as preconditioner), and the HB–PCCG method. The algorithms are terminated, when the relative defect becomes smaller than* $10^{-4}$.

| $l$ | Algorithm 1 | MG(1)–PCCG method for solving the system (126) | MG(1)–PCCG method for solving the system (127) | HB–PCCG method for solving the system (126) |
|---|---|---|---|---|
| | number of iterations / CPU–time | number of iterations / CPU–time | number of iterations / CPU–time | number of iterations / CPU–time |
| 3 | 25 / 3.51 sec | 9 / 1.32 sec | 9 / 1.32 sec | 37 / 1.43 sec |
| 4 | 26 / 15.37 sec | 9 / 5.87 sec | 9 / 5.66 sec | 46 / 6.87 sec |
| 5 | 25 / 63.11 sec | 9 / 23.40 sec | 9 / 23.67 sec | 54 / 32.63 sec |

Table 10

*Comparison of the energy norms of the solutions*

| Level $l$ | energy norm of the solution of the system (126) | energy norm of the solution of the system (127) |
|---|---|---|
| 3 | 6.34388 | 6.34386 |
| 4 | 6.41933 | 6.41931 |
| 5 | 6.45374 | 6.45378 |

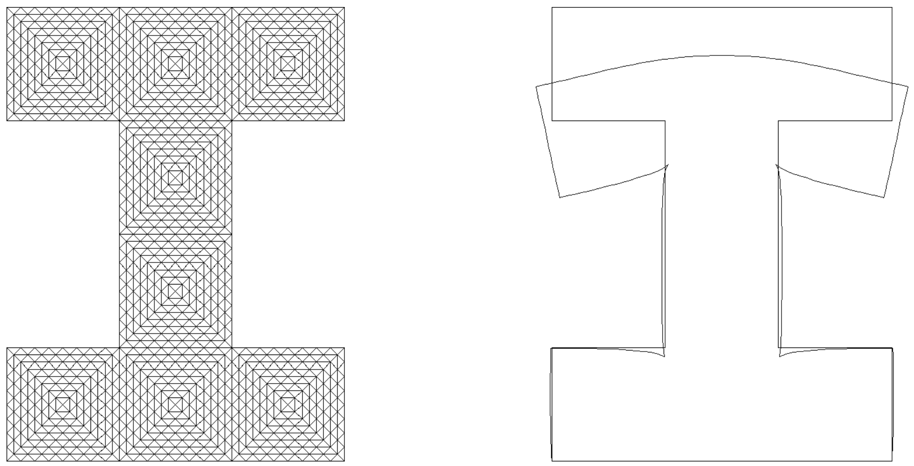


Fig. 4. *The triangulation (level 4) and the contour of the domain with its deformation*

**6. Conclusions.** We have shown that multigrid $\tau$-extrapolation can be interpreted as an implicit method to form higher order FE stiffness matrices. This is not only of theoretical interest, but leads to an efficient higher order multilevel solution technique for PDE. In particular, this extrapolation technique can be used on unstructured meshes.
The resulting algorithm is competitive with multilevel methods that use higher order elements directly. The convergence rate and numerical work per iteration are comparable, but the algorithm has the advantage of a possibly simpler structure. In particular, the $\tau$-extrapolation method is easy to incorporate into existing low order methods, because it differs from the basic algorithm for linear elements only by a slight modification of the fine-to-coarse restriction process.
The alternative analysis for $\tau$-extrapolation given in Rüde [18] is based on asymptotic expansions for quadrature rules over the triangle, and shows that the method can be generalized when the coefficients are not piecewise constant. In this case the linear combination of the stiffness matrices constitute an appropriate numerical quadrature formula for the quadratic stiffness matrix. This analysis also opens the possibility to generalize this technique to higher order. Some preliminary results for these extensions are contained in Rüde [15], [16], and [17].

## REFERENCES


[1] O. Axelsson and I. Gustafsson, *Preconditioning and two–level multigrid methods of arbitrary degree of approximation*, Mathematics of Computation, 40 (1983), pp. 219–242.

[2] R. E. Bank and T. F. Dupont, *An optimal order process for solving elliptic finite element equations*, Mathematics of Computations, 36 (1981), pp. 35–51.

[3] H. Blum, Q. Lin, and R. Rannacher, *Asymptotic error expansions and Richardson extrapolation for linear finite elements*, Numer. Math., 49 (1986), pp. 11–37.

[4] D. Braess, *The contraction number of a multigrid method for solving the Poisson equation*, Numerische Mathematik, 37 (1981), pp. 387–404.

[5] A. Brandt, *Multigrid techniques: 1984 guide with applications to fluid dynamics*, GMD Studien, 85 (1984).

[6] W. Hackbusch, *Multi–Grid methods and applications*, vol. 4 of Springer Series in Computational Mathematics, Springer–Verlag, Berlin, 1985.

[7] W. Hackbusch and U. Trottenberg, eds., *Multigrid methods*, vol. 960 of Lecture Notes in Mathematics, Berlin–Heidelberg–New York, 1982, Springer–Verlag. Proc. of the Conference held at Köln–Porz, November 23–27, 1981.

[8] M. Jung, *Konvergenzfaktoren von Mehrgitterverfahren für Probleme der ebenen linearen Elastizitätstheorie*, ZAMM, 67 (1987), pp. 165–173.

[9] M. Jung, *Eine Einführung in die Theorie und Anwendung von Mehrgitterverfahren*, Wissenschaftliche Schriftenreihe 9, Technische Universität Karl–Marx–Stadt, 1989.

[10] M. Jung, U. Langer, A. Meyer, W. Queck, and M. Schneider, *Multigrid preconditioners and their applications*, in Third Multigrid Seminar, Biesenthal 1988, G. Telschow, ed., Berlin, 1989, Karl–Weierstrass–Institut, pp. 11–52. Report R–MATH–03/89.

[11] M. Jung, U. Langer, and U. Semmler, *Two–level hierarchically preconditioned conjugate gradient methods for solving linear elasticity finite element equations*, BIT, 29 (1989), pp. 748–768.

[12] J. F. Maitre and F. Musy, *The contraction number of a class of two–level methods, an exact evaluation for some finite element subspaces and model problems*, in [7], 1982, pp. 535–544.

[13] G. Marchuk and V. Shaidurov, *Difference Methods and their Extrapolations*, Springer, New York, 1983.

[14] S. McCormick and U. Rüde, *On local refinement higher order methods for elliptic partial differential equations*, International Journal of High Speed Computing, 2 (1990), pp. 311–334. Also available as Bericht I-9034, TU München.

[15] U. Rüde, *Multiple tau-extrapolation for multigrid methods*, Bericht I-8701, Institut für Informatik, TU München, January 1987.

[16] U. Rüde, *On the accurate computation of singular solutions of Laplace's and Poisson's equation*, in Multigrid Methods: Theory, Applications, Supercomputing: Proceedings of the Third Copper Mountain Conference on Multigrid Methods, April 5-10, 1987, S. McCormick, ed., New York, 1988, Marcel Dekker.

[17] U. Rüde, *Extrapolation and related techniques for solving elliptic equations*, Bericht I-9135, Institut für Informatik, TU München, September 1991.

[18] U. Rüde, *Extrapolation techniques for constructing higher order finite element methods*, Bericht I-9304, Institut für Informatik, TU München, Juli 1993.

[19] N. Schieweck, *A multigrid convergence proof by a strengthened Cauchy inequality for symmetric elliptic boundary value problems*, in Second Multigrid Seminar, Garzau, November 5–8, 1985, G. Telschow, ed., Karl–Weierstrass–Institut für Mathematik, Berlin, 1986, pp. 49–62. Report R–Math–08/86.

[20] T. Steidten and M. Jung, *Das Multigrid–Programmsystem FEMGPM zur Lösung elliptischer und parabolischer Differentialgleichungen einschließlich mechanisch–thermisch gekoppelter Probleme (Version 06.90)*, Programmdokumentation, Technische Universität Karl–Marx–Stadt, Sektion Mathematik, 1990.

[21] C.-A. Thole, *Beiträge zur Fourieranalyse von Mehrgittermethoden: V–cycle, ILU–Glättung, anisotrope Operatoren*, Diplomarbeit, Institut für Angewandte Mathematik, Universität Bonn, 1983. Diplomarbeit.

[22] H. Yserentant, *On the multi–level splitting of finite element spaces*, Numer. Math., 49 (1986), pp. 379–412.

**Authors' address:**

Dr. rer. nat. Michael Jung
Technische Universität Chemnitz–Zwickau
Fakultät für Mathematik
PSF 964
D – 09009 Chemnitz

e–mail: dr.michael.jung@mathematik.tu–chemnitz.de

Dr. rer. nat. habil. Ulrich Rüde
Technische Universität Chemnitz–Zwickau
Fakultät für Mathematik
PSF 964
D – 09009 Chemnitz

e–mail: ruede@mathematik.tu–chemnitz.de