An Algebraic Multigrid Method for Nearly Incompressible Elasticity Problems in Two-Dimensions

Yingxiong Xiao[1,2], Shi Shu[2,*], Hongmei Zhang[3] and Yuan Ouyang[2]

[1] *Civil Engineering and Mechanics College, Xiangtan University, Xiangtan 411105, People's Republic of China.*
[2] *School of Mathematics and Computational Science, Xiangtan University, Xiangtan 411105, People's Republic of China.*
[3] *Department of Mathematics and Physics, Hunan University of Technology, Zhuzhou 412008, People's Republic of China.*

Abstract. In this paper, we discuss an algebraic multigrid (AMG) method for nearly incompressible elasticity problems in two-dimensions. First, a two-level method is proposed by analyzing the relationship between the linear finite element space and the quartic finite element space. By choosing different smoothers, we obtain two types of two-level methods, namely TL-GS and TL-BGS. The theoretical analysis and numerical results show that the convergence rates of TL-GS and TL-BGS are independent of the mesh size and the Young's modulus, and the convergence of the latter is greatly improved on the order p. However the convergence of both methods still depends on the Poisson's ratio. To fix this, we obtain a coarse level matrix with less rigidity based on selective reduced integration (SRI) method and get some types of two-level methods by combining different smoothers. With the existing AMG method used as a solver on the first coarse level, an AMG method can be finally obtained. Numerical results show that the resulting AMG method has better efficiency for nearly incompressible elasticity problems.

1   Introduction

In practice, a linear elasticity analysis is often required during simulations of multi-physics problems such as micro-electro-mechanical systems [1]. Typically, these mod-

---

ern devices involve complicated geometries, extremely high aspect ratios and disparate material properties and a great number of such problems must be solved by some numerical methods, in which the finite element method is the most commonly used numerical method for their analysis [2, 3]. There are many materials in applications such as rubber and plastic, which show nearly incompressible material properties, i.e., poisson's ratio v close to 0.5 or *Lamé* constant λ close to ∞. For the planar linear elasticity, it is well known that some finite element schemes result in poor convergence in the displacements (diverge or cannot obtain the optimal order of convergence) as λ is close to ∞. This is the so-called poisson's locking phenomenon in engineering. There are many literatures involving the locking phenomenon of the finite element method [6–10] and more detailed explanation of locking effects can be found in [4–6].

In order to overcome this poisson's locking, we need to construct some finite element schemes whose optimal error estimates are uniform with respect to $\lambda \in (0, \infty)$. Several approaches are developed in recent years, such as mixed finite element method based on Hellinger-Reissner variational principle [5, 7, 11–14], nonconforming finite element method [15–19], *p*-version and *hp*-version method of higher-order finite element scheme [4, 20–23], selective reduced integration method [24] that is often equivalent to a mixed method, and etc. Since discrete variation formulas, based on the minimization of the energy functional, are easier to be solved than the mixed formula, we consider this formula with pure displacement boundary condition. Moreover, the system matrix is positive definite, such that the CG method can be applied. In addition, while nonconforming finite elements has few degree of freedom, this method is sensitive to the adopted mesh. Therefore, in this paper we consider higher-order conforming finite element scheme to overcome this locking.

In 1983, M. Vogelius [20] considered conforming finite element approximations to the linear planar elasticity as λ is close to ∞, and showed that the piecewise linear conforming finite element scheme did not converge any more, and the quadratic and cubic conforming finite element schemes could not obtain optimal error estimates. In [21], it was shown that no locking results could be obtained when polynomials of degree $p \geq 4$ on a triangular mesh. Hence, *p*-version and *hp*-version method of higher-order ($p \geq 4$) conforming finite element scheme is an important approach to overcome the poisson's locking [23]. However, they have much higher computational complexity than the low-order elements, and the system matrix is often large-scale, symmetric and positive definite and ill-conditioned.

Algebraic multigrid (AMG) method for system of PDEs, such as the equations of linear elasticity, seems to be premature and the naive use of the scalar AMG does not lead to the robust and efficient solver. We would like to refer readers to [25–29] for the recent efforts to apply AMG methods for system of PDEs. But these methods are just for linear discretizations. For the higher-order discretization of system of PDEs, there are few studies on designing fast solvers. Recently, several efficient AMG methods for $v \leq 0.4$ are developed in [30] for higher-order discretizations by using geometric and algebraic information. Theoretically, it can be viewed as a two-level method proposed

in [31] where the coarse space is a linear finite element space. But these methods have poor performance when $\nu \to 0.5$ due to a combination of a reduction in the smoothing effect of the Gauss–Seidel relaxation and coarse mesh locking.

In the paper, a two-level method is first proposed by analyzing the relationship between the linear finite element space and quartic finite element space for nearly incompressible elasticity problems in two dimensions. By choosing different smoothers, we obtain two types of two-level methods, namely TL-GS and TL-BGS. The theoretical analysis and numerical results show that the convergence rates of TL-GS and TL-BGS are independent of the mesh size h, and the Young's modulus E, and the convergence of the latter is greatly improved on the order p. But the convergence of both methods still depends on Poisson's ratio ν. And then, we obtain a coarse level matrix with less rigidity based on selective reduced integration(SRI) method and get some types of two-level methods by combining different smoothers. With the existing AMG solver used as a solver on the first coarse level, an AMG method can be finally obtained as $\nu \to 0.5$. Numerical results have shown that the resulting AMG method has better efficiency for nearly incompressible elasticity problems.

The remainder of the paper is organized as follows. In the next section, we introduce poisson's locking in linear elasticity via a numerical example. In section 3, we numerically demonstrate that no locking results can be obtained when polynomials of degree $p \geq 4$. In section 4, by combining SRI method and different smoothers, several AMG methods can be obtained for the quartic finite element discretizations. In the final section, we conclude with some remarks.

2 Poisson's locking in linear elasticity

Consider the following pure displacement problem of the planar linear elasticity

$$\begin{cases} -\mu\Delta\boldsymbol{u} - (\lambda+\mu)\nabla\text{div }\boldsymbol{u} = \boldsymbol{f} & \text{in } \Omega, \\ \boldsymbol{u} = 0 & \text{on } \partial\Omega, \end{cases} \tag{2.1}$$

where λ and μ are *Lamé* constants, $\boldsymbol{u}$ is the displacement and $\boldsymbol{f}$ is an external force.

In what follows, we consider discretizations that are obtained from the weak formulation of the problem (2.1): find $\boldsymbol{u} \in (H_0^1(\Omega))^2$ such that

$$a(\boldsymbol{u},\boldsymbol{v}) = (\boldsymbol{f},\boldsymbol{v}), \quad \forall \boldsymbol{v} \in (H_0^1(\Omega))^2, \tag{2.2}$$

where

$$a(\boldsymbol{u},\boldsymbol{v}) = \int_\Omega (\boldsymbol{Lv})^T \boldsymbol{DLu}\, d\Omega, \quad (\boldsymbol{f},\boldsymbol{v}) = \int_\Omega \boldsymbol{f}\cdot\boldsymbol{v}\, d\Omega, \tag{2.3}$$

and

$$\boldsymbol{L}(\partial_x,\partial_y) = \begin{bmatrix} \partial_x & 0 \\ 0 & \partial_y \\ \partial_y & \partial_x \end{bmatrix}, \quad \boldsymbol{D} = \begin{bmatrix} \lambda+2\mu & \lambda & 0 \\ \lambda & \lambda+2\mu & 0 \\ 0 & 0 & \mu \end{bmatrix}.$$

We assume that $(\lambda, \mu) \in [0, \infty) \times [\mu_1, \mu_2], 0 < \mu_1 < \mu_2$. It is known that λ and μ can be expressed by Young's modulus E and Poisson's ratio ν ($\nu \in [0, 0.5)$) as follows

$$\mu = \frac{E}{2(1+\nu)}, \quad \lambda = \frac{E\nu}{(1+\nu)(1-2\nu)}. \tag{2.4}$$

By [33], we have the following lemma:

Lemma 2.1. *For any given $\boldsymbol{u} \in (H_0^1(\Omega))^2$, we have*

(1) $a(\boldsymbol{u}, \boldsymbol{v}) \lesssim (\lambda + \mu)\|\boldsymbol{u}\|_1\|\boldsymbol{v}\|_1, \quad \forall \boldsymbol{v} \in (H_0^1(\Omega))^2;$ (2.5)

(2) $a(\boldsymbol{u}, \boldsymbol{u}) \gtrsim \mu\|\boldsymbol{u}\|_1^2.$ (2.6)

From Lemma 2.1, we can see that the bilinear form (2.2) is bounded and V-elliptic and thus, the solution of (2.1) exists and is unique.

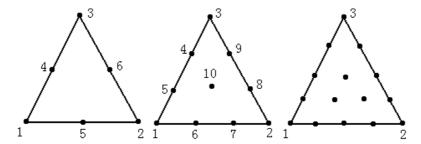Figure 1: Nodes of the typical quadratic, cubic and quartic elements.

Let T^h be a quasi-uniform triangular partition of the domain Ω, N the total number of nodes after the Dirichlet boundary conditions have been applied, where h is the maximal diameter of all elements in T^h. In what follows, we introduce the conforming Lagrangian finite element space of p ($p \geq 1$) order:

$$\boldsymbol{V}_h^{(p)} = \{\boldsymbol{v}_h^{(p)} | \boldsymbol{v}_h^{(p)} \in C(\overline{\Omega})^2, \boldsymbol{v}_h^{(p)}|_\tau \in P_p(\tau)^2, \forall \tau \in T^h\},$$

where P_p is the set of polynomials of degrees not more than p. Figure 1 illustrates nodes of the typical quadratic, cubic and quartic elements, respectively. The finite element solution $\boldsymbol{u}_h^{(p)} \in \boldsymbol{V}_h^{(p)}$ of problem (2.2) satisfies

$$a(\boldsymbol{u}_h^{(p)}, \boldsymbol{v}_h^{(p)}) = (\boldsymbol{f}, \boldsymbol{v}_h^{(p)}), \quad \forall \boldsymbol{v}_h^{(p)} \in \boldsymbol{V}_h^{(p)} \cap (H_0^1(\Omega))^2. \tag{2.7}$$

The corresponding linear algebraic system of equations can be written as

$$A_h^{(p)}\boldsymbol{u}_h^{(p)} = \boldsymbol{f}_h^{(p)}, \tag{2.8}$$

where $\boldsymbol{u}_h^{(p)}$ and $\boldsymbol{f}_h^{(p)}$ are $2N \times 1$ vectors, $A_h^{(p)}$ is $2N \times 2N$ matrix.

Furthermore, we can obtain the error estimate as follows:

Theorem 2.1. ([16]) *Assume that $\boldsymbol{u}$ and $\boldsymbol{u}_h^{(p)}$ are the solutions of (2.2) and (2.7), respectively. Then there exists a positive constant C independent of λ, μ and h, such that*

$$\|\boldsymbol{u} - \boldsymbol{u}_h^{(p)}\|_{1,\Omega} \leq C\sqrt{2 + \frac{\lambda}{\mu}} \cdot h|\boldsymbol{u}|_{2,\Omega}. \tag{2.9}$$

From Theorem 2.1, it can be seen that the solution $\boldsymbol{u}_h^{(p)}$ of the conforming finite element approximation (2.7) converges to the solution $\boldsymbol{u}$ of the problem (2.2), as $h \to 0$, for each fixed λ; but we can not say anything for convergence of $\boldsymbol{u}_h^{(p)}$ even if h is very very small when $\nu \to 0.5$ or $\lambda \to \infty$.

In [15], Brenner et al., have proved that the numerical solution $\boldsymbol{u}_h^{(1)}$ of the conforming linear finite element method converges to zero as $\lambda \to \infty$, and however, the exact solution does not converge to zero. Such phenomenon is known as numerical locking. In what follows, we will demonstrate the theoretical result above by a numerical example. It is also used in Reference [16] as an example.

Example 1. Consider the planar strain problem. Let the domain $\Omega = [0, 1]^2$ and $\boldsymbol{f} = (f_u, f_v)^T$, where $f_u = -[(12x^2 - 12x + 2)(4y^3 - 6y^2 + 2y) + (x - x^2)^2(24y - 12)]$, $f_v = (24x - 12)(y - y^2)^2 + (4x^3 - 6x^2 + 2x)(12y^2 - 12y + 2)$. The corresponding exact solution $\boldsymbol{u} = (u, v)^T$, where $u = (x - x^2)^2(4y^3 - 6y^2 + 2y)$, $v = -(y - y^2)^2(4x^3 - 6x^2 + 2x)$. In our numerical experiments, we assume that $\mu = 1$, and $\lambda = 2\nu/(1 - 2\nu)$.

The domain Ω is divided into squares uniformly. Let h be the edge length of each element, which is divided into two triangle uniformly. The L^2 norms of the approximate solution $\boldsymbol{u}_h^{(1)}$ of the conforming linear finite element scheme, the exact solution $\boldsymbol{u}$ and $\boldsymbol{u} - \boldsymbol{u}_h^{(1)}$ are listed in Table 1 when $\lambda = 10^3$, 10^5 and 10^{10}.

From these numerical results by the linear conforming finite element scheme, it can be shown that

$$\|\boldsymbol{u}_h^{(1)}\|_{0,\Omega} \to 0, \quad \|\boldsymbol{u} - \boldsymbol{u}_h^{(1)}\|_{0,\Omega} \to \|\boldsymbol{u}\|_{0,\Omega},$$

Table 1: Error comparison of $\mathbf{u}_h^{(1)}$ and $\mathbf{u}$ for $\lambda = 10^3$, 10^5 and 10^{10}.

λ	h^{-1}	$\|\boldsymbol{u}_h^{(1)}\|_{0,\Omega}$	$\|\boldsymbol{u} - \boldsymbol{u}_h^{(1)}\|_{0,\Omega}$	$\frac{\|\boldsymbol{u} - \boldsymbol{u}_h^{(1)}\|_{0,\Omega}}{\|\boldsymbol{u}\|_{0,\Omega}}$
	32	2.0238324E-03	3.3767358E-03	6.3330849E-01
10^3	64	3.6869725E-03	1.8053085E-03	3.3345329E-01
	128	4.8158052E-03	6.8984618E-04	1.2643926E-01
	32	3.7778397E-05	5.2961113E-03	9.9328832E-01
10^5	64	1.4175536E-04	5.2790122E-03	9.7507101E-01
	128	3.2119066E-04	5.1448044E-03	9.4297147E-01
	32	3.8159021E-10	5.3318968E-03	9.9999993E-01
10^{10}	64	1.4634898E-09	5.4139757E-03	9.9999974E-01
	128	3.3739834E-09	5.4559459E-03	9.9999940E-01

when $\lambda \to \infty$. The relative errors are almost equal to 1 even for very small h.

3   Locking-free elements for nearly incompressible elasticity problems

In the previous section, we show numerically that the linear finite element scheme does not converge any more when $\lambda \to \infty$. Besides, the quadratic and cubic conforming finite element schemes converge but can not obtain optimal error estimates.

In [5,20] or [21], it was shown that no locking results could be obtained when polynomials of degree $p \geq 4$ on each element for the quasi-uniform triangular meshes. In what follows, we demonstrate the theoretical result above by the numerical example used in Section 2. For the simplicity of discussion here, let us now assume that $p = 4$. Figure 2 illustrates nodes of the typical quartic element. The corresponding linear algebraic system of equations can be written as

$$A_h^{(4)} u_h^{(4)} = f_h^{(4)}, \tag{3.1}$$

or

$$\begin{bmatrix} (A_{11})_h^{(4)} & (A_{12})_h^{(4)} & \cdots & (A_{1N})_h^{(4)} \\ (A_{21})_h^{(4)} & (A_{22})_h^{(4)} & \cdots & (A_{2N})_h^{(4)} \\ \vdots & \vdots & \ddots & \cdots \\ (A_{N1})_h^{(4)} & (A_{N2})_h^{(4)} & \cdots & (A_{NN})_h^{(4)} \end{bmatrix} \begin{bmatrix} (u_1)_h^{(4)} \\ (u_2)_h^{(4)} \\ \vdots \\ (u_N)_h^{(4)} \end{bmatrix} = \begin{bmatrix} (f_1)_h^{(4)} \\ (f_2)_h^{(4)} \\ \vdots \\ (f_N)_h^{(4)} \end{bmatrix}, \tag{3.2}$$

where $(u_i)_h^{(4)}$ and $(f_i)_h^{(4)}$ are 2×1 vectors, $(A_{ij})_h^{(4)}$ is 2×2 matrices with $i, j = 1, \cdots, N$.

The L^2 norms of the approximate solution $u_h^{(4)}$ of the conforming quartic finite element scheme, the exact solution u and $u - u_h^{(4)}$ are listed in Table 4 when $\lambda = 10^5$ and 10^8. Besides, we also present the corresponding error comparison for the conforming quadratic and cubic finite element schemes as shown in Tables 2 and 3, respectively. Furthermore, we list the theoretical convergence rates (TCR=h_1^p/h_2^p, $p = 2, 3, 4$) and numerical convergence rates (NCR=$\|u - u_{h_1}^{(p)}\|_{0,\Omega}/\|u - u_{h_2}^{(p)}\|_{0,\Omega}$) for comparison, where h_1 is the grid size of a coarse mesh and h_2 is that of the next refinement mesh.

From these results, it can be seen that p-version and hp-version method of higher-order ($p \geq 4$) conforming finite element scheme is an important approach to overcome the poisson's locking. It is the so-called locking-free finite element scheme. However, they have much higher computational complexity than the low-order elements, and the system matrix is often large-scale, symmetric and positive definite and ill-conditioned. Those commonly used iterative methods such as Gauss-Seidel and ILU-type PCG methods converge very slowly and are inefficient for practically interesting large-scale problems. In the next section, we will discuss the faster solver for this locking-free finite element equations.

Table 2: Error comparison of $\mathbf{u}_h^{(2)}$ and $\mathbf{u}$ for $\lambda = 10^5$ and $\lambda = 10^8$.

λ	h^{-1}	$\|\boldsymbol{u}_h^{(2)}\|_{0,\Omega}$	$\|\boldsymbol{u} - \boldsymbol{u}_h^{(2)}\|_{0,\Omega}$	$\frac{\|\boldsymbol{u}-\boldsymbol{u}_h^{(2)}\|_{0,\Omega}}{\|\boldsymbol{u}\|_{0,\Omega}}$	TCR	NCR
10^5	8	4.9294539E-03	3.9646695E-04	7.6622256E-02		
	16	5.2704340E-03	1.0047291E-04	1.8843745E-02	8	3.95
	32	5.3986161E-03	2.5155636E-05	4.6464245E-03	8	3.99
	64	5.4522263E-03	6.0895954E-06	1.1161385E-03	8	4.13
10^8	8	3.8574703E-03	1.5562202E-03	3.1919548E-01		
	16	4.8974149E-03	4.2830632E-04	8.2775618E-02	8	3.63
	32	5.2265775E-03	2.0354709E-04	3.8175358E-02	8	2.10
	64	5.3510403E-03	1.8042625E-04	3.3326010E-02	8	1.13

Table 3: Error comparison of $\mathbf{u}_h^{(3)}$ and $\mathbf{u}$ for $\lambda = 10^5$ and $\lambda = 10^8$.

λ	h^{-1}	$\|\boldsymbol{u}_h^{(3)}\|_{0,\Omega}$	$\|\boldsymbol{u} - \boldsymbol{u}_h^{(3)}\|_{0,\Omega}$	$\frac{\|\boldsymbol{u}-\boldsymbol{u}_h^{(3)}\|_{0,\Omega}}{\|\boldsymbol{u}\|_{0,\Omega}}$	TCR	NCR
10^5	4	5.0393995E-03	1.2371461E-04	2.4386539E-02		
	8	5.2763859E-03	1.4910161E-05	2.8247156E-03	16	8.30
	16	5.3862180E-03	1.8278631E-06	3.3935113E-04	16	8.16
	32	5.4418789E-03	2.2344258E-07	4.1059762E-05	16	8.18
10^8	4	5.0393309E-03	1.2350632E-04	2.4345480E-02		
	8	5.2737470E-03	3.0717491E-05	5.8193991E-03	16	4.02
	16	5.3852120E-03	1.4965499E-05	2.7784133E-03	16	2.05
	32	5.4415856E-03	7.5526470E-06	1.3878728E-03	16	1.98

Table 4: Error comparison of $\mathbf{u}_h^{(4)}$ and $\mathbf{u}$ for $\lambda = 10^5$ and $\lambda = 10^8$.

λ	h^{-1}	$\|\boldsymbol{u}_h^{(4)}\|_{0,\Omega}$	$\|\boldsymbol{u} - \boldsymbol{u}_h^{(4)}\|_{0,\Omega}$	$\frac{\|\boldsymbol{u}-\boldsymbol{u}_h^{(4)}\|_{0,\Omega}}{\|\boldsymbol{u}\|_{0,\Omega}}$	TCR	NCR
10^5	4	5.1726477E-03	1.5247966E-05	2.9468625E-03		
	8	5.3318877E-03	4.5916523E-07	8.6116669E-05	32	33.21
	16	5.4139770E-03	1.2905741E-08	2.3837820E-06	32	35.58
	32	5.4559293E-03	3.8073027E-10	6.9782845E-08	32	34.16
10^8	4	5.1726281E-03	1.5198541E-05	2.9373104E-03		
	8	5.3318874E-03	4.5927419E-07	8.6137104E-05	32	33.09
	16	5.4139757E-03	1.2986675E-08	2.3987311E-06	32	35.37
	32	5.4559682E-03	3.8177948E-10	6.9974653E-08	32	34.28

4 AMG method for the locking-free finite element equations

In practical computations, the system matrix in the finite element equations (3.1) is often large-scale and ill-conditioned. Those commonly used iterative methods such as Gauss-Seidel and ILU-type PCG methods converge very slowly and are inefficient for practically interesting large-scale problems. Therefore, it is necessary to develop some

special techniques to improve the performance of algebraic solvers. As we shall show later, the proposed AMG method in this paper will be more efficient for the solution of the resulting locking-free finite element equations.

4.1 Two-level method for compressible elasticity problems

In this subsection, we will design and analyze a class of two-level method for algebraic systems arising from the discretization of compressible elasticity problems ($\nu \leq 0.4$) by quartic finite element methods.

4.1.1 Two-level method by algebraic approaches

Let $V_f = V_h^{(4)}$ be the fine finite element space with the bilinear form $a_h(\cdot, \cdot) = a(\cdot, \cdot)$, where $a(\cdot, \cdot)$ is defined in (2.3). The corresponding finite element equations can be written as $A_f u_f = f_f$. In this paper, we take the linear finite element space $V_l = V_h^{(1)}$ as the first coarse level finite element space V_c, whose bilinear form is $a_c(\cdot, \cdot) = a(\cdot, \cdot)$. The finite element equations on the first coarse level can be written as $A_l u_l = f_l$. Thus, we can give a preliminary two-level algorithm for solving the equation (3.1) as follows.

Algorithm 4.1: (Two-level method)

1: Pre-smoothing: $\mathbf{u}_f = \mathbf{u}_f + \mathbf{S}_f(\mathbf{f}_f - \mathbf{A}_f\mathbf{u}_f)$, $j = 1, 2, \cdots, m_1$.

2: Solving the equation on first coarse level: $\mathbf{A}_l\mathbf{e}_l = \mathbf{P}_f^l(\mathbf{f}_f - \mathbf{A}_f\mathbf{u}_f)$.

3: Correcting: $\mathbf{u}_f = \mathbf{u}_f + \mathbf{P}_l^f\mathbf{e}_l$.

4: Post-smoothing: $\mathbf{u}_f = \mathbf{u}_f + \mathbf{S}_f(\mathbf{f}_f - \mathbf{A}_f\mathbf{u}_f)$, $j = 1, 2, \cdots, m_2$.

In Step 1 and Step 4 above, $\mathbf{S}_f$ is the smoother, m_1 and m_2 are the number of pre-smoothing and post-smoothing respectively, and $\mathbf{P}_f^l$ is the full restriction operator from V_f to V_l.

In what follows, we will discuss the restriction operator $\mathbf{P}_f^l$ and the smoother $\mathbf{S}_f$. For the convenience of discussion, we introduce some notations. For a given grid T^h, we can divide all interpolation nodes of the quartic finite element space into three classes. One consists of all vertices of the grid which are called as type-a nodes, and another consists of three quartering-points of all edges which are called as type-e nodes and a third one consists of interior-points of all elements which are called as type-c nodes as shown in Figure 2. Here, we assume that the finite element nodes are uniformly distributed along the element edges.

Let N_a, N_e and N_c be the total numbers of type-a, type-e and type-c nodes, respectively, and $N = N_a + N_e + N_c$. Assume that X_a, X_e and X_c are the sets of all type-a, type-e and type-c nodes, respectively, and S is the index set of all nodes. We divide S into three subsets: $S_a = \{i : i \in S, \mathbf{x}_i \in X_a\}$, $S_e = \{i : i \in S, \mathbf{x}_i \in X_e\}$ and $S_c = \{i : i \in S, \mathbf{x}_i \in X_c\}$, where $\mathbf{x}_i$ is the ith node with $i = 1, 2, \cdots, N$.
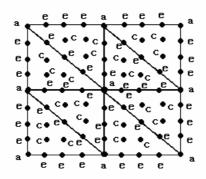
Figure 2: Type-a, type-e and type-c nodes.

Let $\{\Phi_i\}_{i\in S}$ be the vector nodal basis functions related to node $\mathbf{x}_i$ in V_f such that

$$\Phi_i = (\phi_i^1, \phi_i^2)^T \quad and \quad \phi_i^k(\mathbf{x}_j) = \delta_{i,j}^k, \quad i,j \in S, \quad k = 1,2, \tag{4.1}$$

and $\{\Psi_i\}_{i\in S_a}$ be the vector nodal basis functions in V_l such that

$$\Psi_i = (\psi_i^1, \psi_i^2)^T \quad and \quad \psi_i^k(\mathbf{x}_j) = \delta_{i,j}^k, \quad i,j \in S_a, \quad k = 1,2. \tag{4.2}$$

Since $V_l \subset V_f$, by using (4.1), (4.2) and the compact support properties of $\{\Psi_i\}_{i\in S_a}$, we have

$$\begin{aligned} \Psi_i(x) = \Phi_i(x) &+ \sum_{(l_c,j_c,k_c)\in S_i^{cc}} (\tfrac{1}{2}\Phi_{l_c}(x) + \tfrac{1}{4}\Phi_{j_c}(x) + \tfrac{1}{4}\Phi_{k_c}(x)) \\ &+ \sum_{(l_e,j_e,k_e)\in S_i^{ee}} (\tfrac{3}{4}\Phi_{l_e}(x) + \tfrac{1}{2}\Phi_{j_e}(x) + \tfrac{1}{4}\Phi_{k_e}(x)), \end{aligned} \tag{4.3}$$

where S_i^{cc} denotes the set of the index (l_c, j_c, k_c) related to three type-c nodes $\mathbf{x}_{l_c}$, $\mathbf{x}_{j_c}$ and $\mathbf{x}_{k_c}$ which are three interior-points of certain element belonging to $supp\ \Phi_i$ and node $\mathbf{x}_{l_c}$ is the geometrically closest to $\mathbf{x}_i$, and S_i^{ee} denotes the set of the index (l_e, j_e, k_e) related to three type-e nodes $\mathbf{x}_{l_e}$, $\mathbf{x}_{j_e}$ and $\mathbf{x}_{k_e}$ which are three quartering-points of certain edge related to the node $\mathbf{x}_i$.

Furthermore, (4.3) can be written as the following form:

$$(\Psi_1, \Psi_2, \cdots, \Psi_{N_a})^T = P_f^l \begin{pmatrix} \Phi^a \\ \Phi^e \\ \Phi^c \end{pmatrix}, \tag{4.4}$$

where P_f^l an $2N_a \times 2N$ matrix.

From the constructing process of the preliminary two-level method by applying geometrical approaches, in order to establish the AMG method for the finite element equation (3.1), we need to resolve the following two problems by algebraic approaches.

Problem 1: Find the index sets S_a, S_e and S_c from the system matrix A_f.

Problem 2: Find the index sets S_i^{cc} and S_i^{ee} for any fixed index $i \in S_a$.

Using the coefficient matrix $A_h^{(4)} = (A_{ij})_h^{(4)}$ in the system of equation (3.1), we first define an auxiliary grid matrix $\overline{A}_h^{(4)} = (\overline{a}_{ij})$ whose entries can be defined as follows

$$\overline{a}_{ij} := \begin{cases} 0, & \text{if } (A_{ij})_h^{(4)} = \mathbf{0}, \\ 1, & \text{if } (A_{ij})_h^{(4)} \neq \mathbf{0}, \end{cases} \tag{4.5}$$

where $"\mathbf{0}"$ denotes a 2×2 zero matrix.

By the auxiliary grid matrix $\overline{A}_h^{(4)}$, we can easily obtain the index sets S_a, S_e and S_c. For convenience, we introduce the following flag array $I_{aec}(k), k = 1, \cdots, N$ to mark all nodes with different types, which satisfy

$$I_{aec}(k) := \begin{cases} 1, & \mathbf{x}_k \in X_a, \\ 2, & \mathbf{x}_k \in X_e, \\ 3, & \mathbf{x}_k \in X_c. \end{cases} \tag{4.6}$$

Analogous to the method in [31] for cubic elements, we can obtain the following criterion for all nodes with different types by counting nodes using some geometrical information as the support properties of the nodal basis functions.

Criterion 4.1. *For any index $i \in S$ ($i = 1, 2, \cdots, N$), let NZI be the nonzero entries in the row i of the auxiliary grid matrix $\overline{A}_h^{(4)}$. If*
(1) $NZI \geq 36$, *then the node x_i related to the index i belongs to a type-a node;*
(2) $15 < NZI \leq 25$, *then the node x_i belongs to a type-e node;*
(3) $NZI < 15$, *then the node x_i belongs to a type-c node.*

Thus, it is easy to obtain the flag array I_{aec} and the index sets S_a, S_e and S_c.
In what follows, we are to resolve Problem 2. We first define two index sets

$$S_i^e = \{j | \mathbf{x}_j \in supp\, \Phi_i \bigcap supp\, \Phi_j, \forall j \in S_e\},$$
$$S_i^c = \{j | \mathbf{x}_j \in supp\, \Phi_i \bigcap supp\, \Phi_j, \forall j \in S_c\}.$$

It is easy to find the index sets S_i^e and S_i^c by using the Criterion 4.2 as follows:

Criterion 4.2. *For any fixed index $i \in S_a$, we have that (1) the index $l \in S_i^e$ iff $l \in S_e$ and $supp\, \Phi_i \bigcap supp\, \Phi_l \neq \varnothing$; (2) the index $l \in S_i^c$ iff $l \in S_c$ and $supp\, \Phi_i \bigcap supp\, \Phi_l \neq \varnothing$.*

Secondly, we need to find the index sets S_i^{ee} and S_i^{cc} for any fixed index $i \in S_a$. The key of this problem is how to resolve the following problems:

1. For any index (l_e, j_e, k_e) (corresponding nodes are denoted by $\mathbf{x}_{l_e}$, $\mathbf{x}_{j_e}$ and $\mathbf{x}_{k_e}$), how to find that which is the geometrically closest to $\mathbf{x}_i$ between these nodes only by algebraic approach;
2. For any index (l_c, j_c, k_c) (corresponding nodes are denoted by $\mathbf{x}_{l_c}$, $\mathbf{x}_{j_c}$ and $\mathbf{x}_{k_c}$), how to find that which is the geometrically closest to $\mathbf{x}_i$ between these nodes only by algebraic approach.

Table 5: Iteration counts of TL-GS with $m_1=m_2=5$.

h^{-1}	ν						
	0.1	0.15	0.2	0.25	0.3	0.35	0.4
4	21	22	24	26	30	35	47
8	20	21	22	25	28	33	45
16	19	20	21	23	27	32	43
32	18	19	21	23	26	31	42

Completely analogous to the case of cubic elements [31], we can resolve the aforementioned two problems by solving certain minimization problem. But then, we need not solve the minimization problem and can easily obtain the restriction operator P_f^l via (4.3) or (4.4) by combining the geometric grid information as geometric coordinates with some relevant auxiliary arrays in our procedure. As a result, the performance of obtaining the restriction operator P_f^l is greatly improved.

Choosing $P_l^f(=(P_f^l)^T)$ as the prolongation operator from V_l to V_f, we can get the following matrix on the first coarse level related to the linear finite element:

$$A_l := (a(\Psi, \Psi)) = P_f^l A_h^{(4)} (P_f^l)^T, \qquad (4.7)$$

where $\Psi = (\Psi_1, \Psi_2, \cdots, \Psi_{N_a})^T$.

On the smoother S_f, we can choose Gauss-Seidel (GS) relaxation, which is efficient for compressible elasticity problem. Thus, we can give a preliminary two-level algorithm for solving the equation (3.1) as follows.

Algorithm 4.2:

In Algorithm 4.1, we take $S_f = GS$, $P_l^f = (P_f^l)^T$ and $A_l = P_f^l A_h^{(4)} (P_f^l)^T$. For convenience later, we denote this method by TL-GS.

In what follows, we first perform the aforementioned Example 1 when $\nu \leq 0.4$ to test the effectiveness and robustness of the resulting TL-GS method. The corresponding numerical results are given in Table 5.

Table 6: Iteration counts of TL-GS for elasticity problems with jumps in Young's modulus.

h	1/8			1/16			1/32		
E_1/E_2	10	100	1000	10	100	1000	10	100	1000
Iteration counts	21	29	30	20	30	32	20	31	33

Example 2. In this example, we examine the linear elasticity problem with jumps in the coefficients of the elasticity operator. Consider the decomposition of the unit square into four subdomains as shown in Figure 3. We set $\nu = 0.25$ and E to $E_2=10^3$MPa on the light subdomains and to $E = E_1$ on the dark subdomains. The constant body force $f = 2.78 \times 10^6$MPa is also used in this example. We then apply the TL-GS method to the solution of quartic finite element equations. The corresponding results are listed in
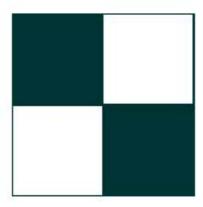
Figure 3: Jumps in Young's modulus, namely $E=E_1$ on dark subdomains and $E=E_2=10^3$ on light ones.

Table 6. We would like to point out here that more fine-level smoothing is required to eliminate the high frequency components of quartic elements. In our numerical tests, we take $m_1=m_2=9$ based in a trial-and-error manner.

From these results for compressible elasticity problems ($\nu \leq 0.4$), it can be seen that the convergence rate of TL-GS is independent of both the mesh size h and the Young's modulus E. This observed behavior is qualitatively similar to the theoretical estimate presented in the next subsection.

Besides, we can also use a special block Gauss-Seidel (BGS) smoothing with blocks corresponding to the supports of the canonical basis related to each node, especially for nearly incompressible problems. Thus, we can obtain another preliminary two-level algorithm for solving the equation (3.1) as follows.

Algorithm 4.3:

In Algorithm 4.1, we take $\mathbf{S}_f = BGS$, $\mathbf{P}_l^f = (\mathbf{P}_f^l)^T$ and $\mathbf{A}_l = \mathbf{P}_f^l \mathbf{A}_h^{(4)} (\mathbf{P}_f^l)^T$. Denote this method by TL-BGS.

And then, we can perform Example 1 when $\nu \leq 0.4$ to test the effectiveness and robustness of the TL-BGS method. In fact, we only need to perform one pre-smoothing and one post-smoothing in TL-BGS and can obtain the satisfactory results. The corresponding numerical results are given in Table 7.

We can see that the convergence rate of TL-BGS is independent of the mesh size h and Poisson's ratio ν for $\nu \leq 0.4$, and is also greatly improved on the order p. But a value of ν close to 0.5 significantly degrades the convergence of both methods due to coarse mesh locking, which means that a coarse mesh will not accurately capture the low frequency error component. Results of TL-BGS are listed in Table 8.

4.1.2   Convergence analysis of the TL-GS method

For a given T^h, let $\{\mathbf{x}_i\}_{i=1}^{n_f}$ and $\{\Phi_i\}_{i=1}^{n_f}$ be the sets of the interpolation nodes and the vector nodal basis functions in V_f, respectively, where $n_f = \dim(V_f)$. Define the

Table 7: Iteration counts of TL-BGS with different m_1 and m_2.

	h^{-1}	ν						
		0.1	0.15	0.2	0.25	0.3	0.35	0.4
$m_1 = 5$ $m_2 = 5$	4	1	1	1	2	2	2	2
	8	3	3	3	3	3	4	4
	16	4	4	4	4	4	5	6
	32	4	4	4	4	4	5	6
$m_1 = 1$ $m_2 = 1$	4	5	5	5	6	6	7	8
	8	6	6	6	7	7	8	9
	16	6	6	6	6	7	8	9
	32	5	5	5	5	6	6	7

Table 8: Iteration counts of TL-BGS when $0.4 < \nu \to 0.5$, where $m_1 = m_2 = 1$.

h^{-1}	ν				
	0.45	0.48	0.49	0.495	0.499
4	11	18	27	40	90
8	15	27	42	67	214
16	13	24	41	73	274
32	11	19	32	54	214

energy projection operators P_k $(k = f, l)$ from $(H_0^1(\Omega))^2$ to V_k as follows

$$a(P_k u, v_k) = a(u, v_k), \qquad \forall v_k \in V_k. \tag{4.8}$$

Furthermore, by introducing the subspaces $V_f^i = span\{\Phi_i\}$ with $i = 1, 2, \cdots, n_f$, we can define the projection operators P_f^i from $(H_0^1(\Omega))^2$ to V_f^i such that

$$a(P_f^i v, \Phi_i) = a(v, \Phi_i), \qquad \forall v \in (H_0^1(\Omega))^2. \tag{4.9}$$

In what follows, we present the convergence analysis for the TL-GS method. Note that we always assume that T^h is shape-regular [31]. For simplicity, we take $m_1 = 1$ and $m_2 = 0$ in the TL-GS method. From [35], we can know that the convergence rate δ of the TL-GS method can be expressed as

$$\delta^2 = \frac{C_0}{1 + C_0}, \tag{4.10}$$

where

$$C_0 = \sup_{v \in V_f, \, \|v\|_a = 1} \quad \inf_{v = v_l + \sum\limits_{i=1}^{n_f} v_f^i} \left(\left\| P_l \sum_{j=1}^{n_f} v_f^j \right\|_a^2 + \sum_{i=1}^{n_f} \left\| P_f^i \sum_{j=i+1}^{n_f} v_f^j \right\|_a^2 \right), \tag{4.11}$$

and $\| u \|_a^2 = a(u, u), \, \forall u = (u_1, u_2)^T \in (H_0^1(\Omega))^2$.

For any given $v \in V_f$, we make the following decomposition:

$$v = I_1 v + (v - I_1 v) = \begin{pmatrix} I_1 v_1 \\ I_1 v_2 \end{pmatrix} + \begin{pmatrix} v_1 - I_1 v_1 \\ v_2 - I_1 v_2 \end{pmatrix}, \tag{4.12}$$

where I_1 is the interpolation operator from the Sobolev space $(H_0^1(\Omega))^2$ to the linear finite element space V_l.

To discuss the convergence of TL-GS, we need to introduce some lemmas as follows.

Lemma 4.1. *For any given $v \in V_f$, we have*

$$\| v - I_1 v \|_1 \lesssim \| v \|_1, \tag{4.13}$$

and

$$\| v - I_1 v \|_{0,\tau} \lesssim h_\tau \mid v \mid_{1,\tau}, \quad \forall \tau \in T^h, \tag{4.14}$$

where h_τ is the diameter of element τ, and $\mid v \mid_{1,\tau}^2 = \int_\tau (v_{x_1}^2 + v_{x_2}^2) dx$.

Proof: Using the results of a scalar PDE in [31], we can easily complete the proof of Lemma 4.1. □

By Lemma 4.1, we can derive that

$$\| I_1 v \|_{1,\Omega} \lesssim \| v \|_{1,\Omega} + \| v - I_1 v \|_{1,\Omega} \lesssim \| v \|_{1,\Omega}. \tag{4.15}$$

Lemma 4.2. *For any given $v \in V_f$, we have*

$$h_\tau^2 \sum_{x_i \in \tau} \mid v(x_i) \mid^2 \cong \| v \|_{0,\tau}^2. \tag{4.16}$$

Lemma 4.3. *Assume that T^h is shape-regular, then for any given $v \in V_f$, we have*

$$\inf_{v = v_l + \sum_{i=1}^{n_f} v_f^i} \left(\left\| P_l \sum_{j=1}^{n_f} v_f^j \right\|_a^2 + \sum_{i=1}^{n_f} \left\| P_f^i \sum_{j=i+1}^{n_f} v_f^j \right\|_a^2 \right) \lesssim \frac{\lambda + \mu}{\mu} \| v \|_a^2, \tag{4.17}$$

where $v_l \in V_l$ and $v_f^i \in V_f^i$ for $i = 1, 2, \cdots, n_f$.

Proof: For any given $v \in V_f$, we make the decomposition as in (4.12) and denote by $v_1 = I_1 v$ and $v_2 = v - I_1 v := \sum_{i=1}^{n_f} v_f^i$, where $v_f^i \in V_f^i$. By using (4.13) and Lemma 2.1, we can derive that

$$\left\| P_l \sum_{j=1}^{n_f} v_f^j \right\|_a^2 = \| P_l (v - I_1 v) \|_a^2 \lesssim \| v - I_1 v \|_a^2$$

$$\lesssim (\lambda + \mu) \| v - I_1 v \|_1^2 \lesssim \frac{\lambda + \mu}{\mu} \| v \|_a^2. \tag{4.18}$$

Using $v_f^i(\mathbf{x}) = v(\mathbf{x}_i)\Phi_i(\mathbf{x})$ and the compact support property of $\Phi_i(\mathbf{x})$ and Lemmas 2.1, 4.1 and 4.2, we have

$$
\begin{aligned}
\sum_{i=1}^{n_f} \left\| \boldsymbol{P}_f^i \sum_{j=i+1}^{n_f} \boldsymbol{v}_f^j \right\|_a^2 &= \sum_{i=1}^{n_f} \left\| \boldsymbol{P}_f^i \sum_{j=i+1}^{n_f} \boldsymbol{v}_f^j \right\|_{a,\Omega_i}^2 \leq \sum_{i=1}^{n_f} \left\| \sum_{j=i+1}^{n_f} \boldsymbol{v}_f^j \right\|_{a,\Omega_i}^2 \\
&\lesssim \sum_{i=1}^{n_f} \| \boldsymbol{v}_f^i \|_{a,\Omega_i}^2 = \sum_{i=1}^{n_f} \sum_{x_i \in \tau} \| \boldsymbol{v}_f^i \|_{a,\tau} \lesssim \sum_{\tau \in T^h} \sum_{x_i \in \tau} \| \boldsymbol{v}_f^i \|_{a,\tau}^2 \\
&\lesssim (\lambda + \mu) \sum_{\tau \in T^h} \sum_{x_i \in \tau} \| v_f^i \|_{1,\tau}^2 \lesssim (\lambda + \mu) \sum_{\tau \in T^h} \sum_{x_i \in \tau} | v_2(x_i) |^2 \\
&= (\lambda + \mu) \sum_{\tau \in T^h} h_\tau^{-2} \sum_{x_i \in \tau} h_\tau^2 | v_2(x_i) |^2 \lesssim (\lambda + \mu) \sum_{\tau \in T^h} h_\tau^{-2} \| v_2 \|_{0,\tau}^2 \\
&\lesssim (\lambda + \mu) \sum_{\tau \in T^h} h_\tau^{-2} \| \boldsymbol{v} - I_1 \boldsymbol{v} \|_{0,\tau}^2 \lesssim (\lambda + \mu) \sum_{\tau \in T^h} | \boldsymbol{v} |_{1,\tau}^2 \lesssim (\lambda + \mu) |\mathbf{v}|_1^2 \\
&\lesssim \frac{\lambda + \mu}{\mu} \| v \|_a^2.
\end{aligned}
\tag{4.19}
$$

Thus, combining ((4.18) and (4.19), we can complete the proof of Lemma 4.2. □

Using (4.10), (4.11) and Lemma 4.3, we can get the convergence result as follows:

Theorem 4.1. *For a given shape-regular triangular grid T^h, the convergence rate δ of TL-GS for the finite element equations (3.1) satisfies*

$$
\delta^2 = \frac{C_0}{1 + C_0} \leq \frac{C(\lambda, \mu)}{1 + C(\lambda, \mu)},
\tag{4.20}
$$

where $C(\lambda, \mu) = C \cdot (\lambda + \mu)/\mu$ and C is a positive constant independent of the problem size.

Remark 4.1. From Theorem 4.1 above, we can obtain some conclusions as follows:

• The convergence rate of TL-GS method for the finite element equations (3.1) is independent of the problem size. It is essentially the same with the numerical results presented in Section 4.1.1;

• Since $(\lambda + \mu)/\mu = 1/(1 - 2\nu)$ which is independent of Young's modulus E, the convergence rate of TL-GS is independent of jumps in Young's modulus E, and this method has better convergence for Poisson's ratio ν less than 0.4. But this is not the case when Poisson's ratio ν tends to 0.5. As we have shown later, the performance of the resulting two-level method can be greatly improved by combining SRI method and BGS smoother.

4.2 Two-level method for incompressible elasticity problems

As we have shown previously, the convergence of TL-BGS method strictly depends on Poisson's ratio ν and a value of Poisson's ratio ν close to 0.5 significantly degrades the convergence of TL-BGS method due to coarse mesh locking. In order to overcome this coarse mesh locking, we still need to use selective reduced integration (SRI) method.

Table 9: The number of iterations of several two-level methods with $m_1=m_2=5$ for the solution of the resulting locking-free finite element equations.

Two-level methods	h^{-1}	ν				
		0.45	0.48	0.49	0.495	0.499
TL(D_λ^e)-GS (TL-GS)	4	74 (82)	165 (187)	315 (362)	716 (878)	7769 (9483)
	8	76 (79)	177 (182)	410 (365)	1280 (1190)	11290 (11295)
	16	76 (76)	217 (211)	661 (656)	1785 (1774)	14194 (14200)
	32	74 (74)	308 (309)	840 (842)	2141 (2145)	16075 (16085)
TL(D_λ^e)-JCG (TL-JCG)	4	42 (46)	91 (100)	173 (190)	330 (367)	1698 (1818)
	8	42 (53)	98 (124)	194 (242)	370 (478)	2038 (2382)
	16	35 (39)	79 (92)	158 (184)	314 (370)	1629 (1851)
	32	35 (36)	72 (73)	138 (154)	281 (315)	1415 (1577)
TL(D_λ^e)-BGS (TL-BGS)	4	3 (3)	4 (4)	5 (6)	7 (9)	12 (17)
	8	6 (6)	9 (11)	12 (17)	15 (28)	42 (92)
	16	8 (8)	11 (14)	17 (23)	27 (38)	91 (132)
	32	6 (7)	10 (11)	15 (18)	25 (31)	86 (105)
TL(D_λ^e)-BGSCG (TL-BGSCG)	4	2 (2)	2 (2)	2 (2)	3 (2)	3 (3)
	8	3 (3)	3 (4)	5 (5)	6 (8)	11 (14)
	16	4 (4)	6 (7)	9 (10)	14 (18)	36 (48)
	32	5 (5)	7 (8)	11 (12)	18 (21)	59 (73)

4.2.1 Two-level method based on SRI method

For isotropic materials, when $\nu \to 0.5$, we rewrite the element stiffness matrix by introducing a threshold of θ as follows

$$K^e = \int_{\Omega^e} \left[\underbrace{(B^e)^T D_\mu^e B^e} + \theta \cdot \underbrace{(B^e)^T D_\lambda^e B^e} \right] dxdy, \qquad (4.21)$$

where $0 < \theta < 1$, the matrices D_μ^e and D_λ^e depend only on the shear modulus μ and Lamé constant λ, respectively, and B^e is the strain-displacement matrix. Thus, we can obtain a coarse level matrix with less rigidity by directly assembling together the element stiffness matrices from all elements based on the SRI method.

Two-level method that uses the coarse level matrix obtained by using SRI method and uses the GS relaxation as a smoother is denoted by TL(D_λ^e)-GS. Similarly, TL(D_λ^e)-BGS means that the coarse level matrix was assembled from the element stiffness matrices where SRI method is used on the D_λ^e term and that the BGS relaxation was used. Moreover, the smoothing of preconditioned conjugate gradient (PCG) method was found to be less sensitive to Poisson's ratio ν than that of GS relaxation [34]. So, if we choose the JCG and BGSCG as the smoothers, we can get four types of two-level methods, i.e., TL-JCG, TL(D_λ^e)-JCG, TL-BGSCG and TL(D_λ^e)-BGSCG, respectively.

In what follows, we perform the aforementioned Example 1 as $0.4 < \nu \to 0.5$ to test the effectiveness of the aforementioned eight types of two-level methods. The corresponding numerical results are summed in Table 9.

From these numerical results, it can be seen that the proposed two-level methods by combining SRI method and BGS smoother have better efficiency for nearly incom-

Table 10: The number of iterations of TL($\mathbf{D}_\lambda^e$)-BGSCG (TL-BGSCG) with $m_1=m_2=2$.

Two-level methods	h^{-1}	ν				
		0.45	0.48	0.49	0.495	0.499
TL(D_λ^e)-BGSCG (TL-BGSCG)	4	3 (3)	4 (4)	5 (5)	6 (7)	8 (8)
	8	5 (5)	7 (8)	10 (11)	14 (17)	41 (44)
	16	8 (7)	10 (11)	14 (17)	20 (26)	60 (77)
	32	6 (8)	9 (10)	14 (15)	20 (24)	61 (75)

pressible elasticity problems, in which the TL(D_λ^e)-BGSCG method is the most efficient one. For this method, in fact we only need to perform two pre-smoothing and two post-smoothing and then obtain the satisfactory results which are listed in Table 10.

Remark 4.2. The threshold θ used in the SRI method is an experiential parameter, which often affects the convergence of the resulting two-level methods. In our numerical experiments, we set $\theta = 0.5$ when $h < 1/32$ and $\theta = 0.8$ when $h = 1/32$ based in a trial-and-error manner.

4.2.2 Algebraic multigrid method

Generally speaking, the performance of the aforementioned two-level method depends critically on the algebraic solvers for the solution of the finite element equations on the first coarse level as follows:

$$A_l e_l = P_f^l(f_f - A_f u_f). \tag{4.22}$$

With the existing AMG method such as AMG01 presented in [29] used as a solver on the first coarse level, we can obtain a type of more efficient AMG method for the solution of the finite element equations (3.1). Note that we use linear elements on the first coarse level, the coarse problem still suffers from Poisson's locking. Therefore, we apply AMG01-BGS method by combining SRI method and BGS smoother to the coarse level equations (4.22). The corresponding algorithm can be describes below:

Algorithm 4.4: (AMG method)

0: (setup)

    (1) Find the flag array I_{aec} by Criterion 4.1;

    (2) Find the index sets S_i^c and S_i^e for any given $i \in S_a$ by Criterion 4.2;

    (3) Find $\mathbf{P}_f^l$ and the coarse level matrix $\mathbf{A}_l$ using SRI method.

1: Pre-smoothing: $\mathbf{u}_f = \mathbf{u}_f + \mathbf{S}_f(\mathbf{f}_f - \mathbf{A}_f\mathbf{u}_f) j = 1, 2, \cdots, m_1$

2: Solving $\mathbf{A}_l\mathbf{e}_l = \mathbf{P}_f^l(\mathbf{f}_f - \mathbf{A}_f\mathbf{u}_f)$ by calling k times AMG01-BGS method

3: Correcting: $\mathbf{u}_f = \mathbf{u}_f + \mathbf{P}_l^f\mathbf{e}_l$

4: Post-smoothing: $\mathbf{u}_f = \mathbf{u}_f + \mathbf{S}_f(\mathbf{f}_f - \mathbf{A}_f\mathbf{u}_f) j = 1, 2, \cdots, m_2$

Table 11: Iteration counts of AMG($\mathbf{D}_\lambda^e$)-BGSCG (AMG($\mathbf{D}_\lambda^e$)-BGS) with AMG01-BGS for the solution of the coarse level equations (4.22), where $m_1=m_2=2$.

h^{-1}	ν				
	0.45	0.48	0.49	0.495	0.499
4	3 (5)	4 (8)	5 (9)	6 (14)	8 (34)
8	5 (8)	7 (13)	10 (20)	14 (31)	34 (81)
16	6 (9)	10 (16)	14 (23)	20 (38)	60 (152)
32	7 (8)	9 (14)	14 (19)	22 (34)	62 (137)

In what follows, we solve Example 1 with the same stopping criteria as before using the proposed AMG method in Algorithm 4.4. The numerical results are listed in Table 11. Note that we find the convergence rate of the resulting AMG method cannot be improved when $k > 3$. Thus, in our numerical experiments, we take $k = 3$ in order to obtain best convergence. It can be seen that the proposed AMG method has better efficiency for nearly incompressible elasticity problems.

5   Concluding remarks

The aim of this paper has been to propose a more efficient AMG solver for nearly incompressible elasticity problems in two dimensions. Most commonly used AMG methods have poor performance when conditions of near incompressibility are encountered due to a reduction in the smoothing capability of GS relaxation and to coarse mesh locking. These effects can be greatly improved by relaxing with the BGSCG method and by using coarse mesh stiffness matrices assembled from elements integrated with reduced integration on the $\mathbf{D}_\lambda^e$ term. Our method is less algebraic or generic than the classical AMG method in the sense that we need to know a priori what type of finite element spaces, and what type of basis functions are used in generating the stiffness matrices.

While only limited numerical experiments have been performed, the numerical results are quite encouraging. More extensive numerical experiments for general problems discretized on unstructured meshes and developing other techniques to obtain more robust and more efficient AMG methods, in particular, to obtain AMG methods independent of Poisson's ratio ν by choosing the P_2/P_0 finite element space as the first coarse level finite element space, are also expected. Such area will be a subject of our future research.

Acknowledgments

References

[1] S. SENTURIA, N. ALURU AND J. WHITE, *Simulating the behavior of MEMS devices: Computational methods and needs*, IEEE Comput. Sci. Eng., 4 (1997), pp. 30–43.

[2] S. SENTURIA, R. HARRIS, B. P. JOHNSON, SONGMIN KIM, K. NABORS, M. A. SHULMAN, J. K. WHITE , *A computer-aided design system for microelectromechanical systems (MEM-CAD)*, J. Microelectromech. S., 1 (1992), pp. 3–13.

[3] C. BRENNER AND L.Y. SUNG, *Linear finite element methods for planar linear elasticity*Math. Comput., 1992, 29 (1992), pp. 321–338.

[4] I. BABUSKA AND M. SURI, *On locking and robustness in the finite element method*, SIAM J. Numer. Anal., 29 (1992), pp. 1261–1293.

[5] I. BABUSKA AND M. SURI, *Locking effects in the finite element approximation of elasticity problems*, Numer. Math, 62 (1992), pp. 439–463.

[6] D. N. ARNOLD, *Discretization by finite elements of a model parameter dependent problem*, Numer. Math., 37 (1981), pp. 405–421.

[7] F. BREZZI, K. BATHE AND M. FORTIN, *Mixed-interpolated elements for Reissner Mindlin plates*, Internat. J. Numer. Methods Engrg., 28 (1989), pp. 1787–1801.

[8] L. LI, *Discretization of the Timoshenko beam problem by the p and h-p version of the finite element*, Numer. Math., 57 (1990), pp. 1-8.

[9] M. VOGELIUS, *An analysis of the p-version of the finite element method for nearly incompressible materials, uniformly valid, optimal order estimates*, Numer. Math., 41 (1983), pp. 39–53.

[10] L. R. SCOTT AND M. VOGELIUS, *Primal hybrid finite element methods for 2nd order elliptic problems*, Math. Comput., (1985), 19: 111-143.

[11] D. N. ARNOLD AND R. S. FALK, *A new mixed formulation for elasticity*, Numer. Math., 53 (1988), pp. 13–30.

[12] R. KOUHIA AND R. STENBERG, *A linear nonconforming finite element method for nearly incompressible elasticity and Stokes flow*, Comput. Meths. Appl. Mech. Engrg., 124 (1995), pp. 195–212.

[13] J. PITKDRANTA AND R. STENBERG, *Analysis of some mixed finite element methods for plane elasticity equations*, Math. Comp., 41 (1983), pp. 399–423.

[14] M. MORLEY, *A mixed family of elements for linear elasticity*, Numer. Math., 55 (1989), pp. 633–666.

[15] S. C. BRENNER AND L. Y. SUNG, *Linear finite element methods for planar linear elasticity*, Math. Comput., (1992), 59: 321-330.

[16] L. H. WANG AND H. QI, *On Locking finite element schemes for the pure displacement boundary value problem in the planar elasticity*, Mathematica Numerica Sinica, 24 (2002), pp. 243–256.

[17] H. QI, L. H. WANG AND W. Y. ZHENG, *On locking-free finite element schemes for three-dimensional elasticity*, J. Comp. Math., 23 (2005), pp. 101–112.

[18] R. S. FALK, *Noncomforming finite element methods for the equations of linear elasticity*, Math. Comput., 57 (1991), pp. 529–556.

[19] C. O. LEE, J. W. LEE AND D. W. SHEEN, *A locking free nonconforming finite element method for planar linear elasticity*, Advances in Comput. Math., (2003), 19: 277-291.

[20] L. R. SCOTT AND M. VOGELIUS, *Norm estimates for a maximal right inverse of the divergence operator in spaces of piecewise polynomials*, RAIRO Math. Modeling Numer. Anal., 19 (1985),

pp. 111–143.

[21] L. R. SCOTT AND M. VOGELIUS, *Conforming finite element methods for incompreeeible and nearly incompressible continua*, In Large Scale Computations in Fluid Mechanics, Lectures in Applied Mathematics, 22 (1985), pp. 221–244.

[22] D. N. ARNOLD AND F. BREZZI, *Mixed and nonconforming finite element methods: implementation, postprocessing and error estimates*, Math. Anal. Numer., 19 (1985), pp. 7–32.

[23] R. STENBERG AND M. SURI, *Mixed h-p finite element methods for problems in elasticity and Stokes flow*, Numer. Math., 72 (1996), pp. 367–390.

[24] O. C. ZIENKIEWICZ, R. L. TAYLOR AND J. M. TOO, *Reduced integration technique in general analysis of plates and Shells*, Int. J. Numer, Methods Engrg., 3 (1971), pp. 575–586.

[25] M. GRIEBEL, D. OELTZ AND M. A. SCHWEITZER, *An algebraic multigrid method for linear elasticity*, SIAM J. Sci. Comput., 25 (2003), pp. 385–407.

[26] M. BREZINA, A. J. CLEARY, R. D. FALGOUT, V. E. HENSON, J. E. JONES, T. A. MANTEUFFEL, S. F. MCCORMICK, AND J. W. RUGE, *Algebraic multigrid based on element interpolation (AMGe)*, SIAM J. Sci. Comput., 22 (2000), pp. 1570–1592.

[27] J. JONES, P. VASSILEVSKI, *AMGE based on element agglomeration*, SIAM J. Sci. Comput., 23 (2001), pp. 109–133.

[28] Y. X. XIAO, P. ZHANG AND S. SHU, *An algebraic multigrid method with interpolation reproducing rigid body modes for semi-definite problems in two dimensional linear elasticity*, J. Comput. Appl. Math., 200 (2007), pp. 637–652.

[29] Y. X. XIAO, P. ZHANG AND S. SHU, *Algebraic multigrid methods for elastic structures with highly discontinuous coefficients*, Math. Comput. Simul., 76 (2007), pp. 249–262.

[30] Y. X. XIAO, S. SHU AND T. Y. ZHAO, *A geometric-based algebraic multigrid for higher-order finite element equations in two dimensional linear elasticity*, Numer. Linear Algebr., 2008, DOI: 10.1002/nla.629.

[31] S. SHU, D. D. SUN AND J. C. XU, *An algebraic multigrids for higher order Finite Element discretizations*, Computing, 77 (2006), pp. 347–377.

[32] J. SCHÖBERL, *Multigrid methods for a parameter dependent problem in primal variables*, Numer. Math., 84 (1999), pp. 97–119.

[33] Z. D. LUO, *Basis of theory and application of mixed finite element methods*, Education press of Shandong Province: Shandong, 1996.

[34] I. PARSONS AND J. HALL, *The multigrid method in solid mechanics: Part I- Algorithm description and behaviour*, Int. J. Numer., Methods Engrg., 29 (1990), pp. 719–737.

[35] J. XU AND L. ZIKATANOV, *The method of alternating projections and the method of subspace corrections on Hilbert space*, J. Am. Math. Soc., 15 (2002), pp. 573–597.