A Virtual Element Method for the Static Bending Analysis of Reissner-Mindlin Plates

Authors:
Xiaoxiao Du, duxiaoxiao@buaa.edu.cn, Beihang University
Gang Zhao, zhaog@buaa.edu.cn, Beihang University
Wei Wang, jrrt@buaa.edu.cn, Beihang University

Introduction:
The virtual element method (VEM), introduced in [3] is designed for solving numerical problems defined on arbitrarily shaped polygonal/polyhedral discretizations. Therefore, it will greatly alleviate the heavy burden placed on meshing complex CAD geometries when compared with the traditional finite element method. Furthermore, VEM could handle the non-conforming discretizations by allowing the existence of hanging nodes, which are treated as normal nodes in the element. The local h-refinement and p-version refinement could be easily implemented under the VEM framework. So far VEM has been successfully applied to solve various problems including topology optimization, contact, fracture, plate bending and vibration, inelasticity.

In this work, we develop an arbitrary order virtual element method for the static bending analysis of Reissner-Mindlin plates. The transverse displacement and rotations are independently interpolated with the functions defined in VEM spaces. The interpolation functions for transverse displacement are one degree higher than the functions for rotations. A benchmark problem is studied to verify the developed method. The optimal convergence rates for transverse displacement and rotations could be obtained from the numerical example.

Reissner-Mindlin Plate Problems:
Let Ω be the domain occupied by the middle plane of an elastic plate with thickness t. Let $\mathcal{W}$ and $\boldsymbol{\Theta}$ be the function spaces for the transverse displacement w and rotations $\boldsymbol{\theta}(\theta_x, \theta_y)$. Then the Reissner-Mindlin plate problems can be described as: Find $w \in \mathcal{W}, \boldsymbol{\theta} \in \boldsymbol{\Theta}$, such that

$$a(\boldsymbol{\theta}, \boldsymbol{\eta}) + b(\boldsymbol{\theta} - \nabla w, \boldsymbol{\eta} - \nabla v) = (g, v), \quad \forall (v, \boldsymbol{\eta}) \in \mathcal{W} \times \boldsymbol{\Theta}, \tag{2.1}$$

where the bilinear forms can be written as

$$a(\boldsymbol{\theta}, \boldsymbol{\eta}) = \int_\Omega \boldsymbol{\epsilon}^T(\boldsymbol{\theta}) \boldsymbol{D}_b \boldsymbol{\epsilon}(\boldsymbol{\eta}) d\Omega \tag{2.2}$$

$$b(\boldsymbol{\theta} - \nabla w, \boldsymbol{\eta} - \nabla v) = \int_\Omega (\boldsymbol{\theta} - \nabla w)^T \boldsymbol{D}_s (\boldsymbol{\eta} - \nabla v) d\Omega \tag{2.3}$$

in which $\boldsymbol{\epsilon}(\boldsymbol{\theta}) = 0.5(\nabla\boldsymbol{\theta} + \nabla\boldsymbol{\theta}^T)$ is the Voigt representation of the strain tensor. $\boldsymbol{D}_b$ and $\boldsymbol{D}_s$ are the material bending and shear constitutive matrices.

Virtual Element Spaces and Degrees of Freedom:

Let $\mathcal{T}_h$ be a decomposition of the domain Ω into a series of polygons. For each polygon $E \in \mathcal{T}_h$, edges $e \in E$ the local VEM spaces of degree $k+1(k \geq 1)$ for the transverse displacement w are defined as

$$\mathcal{W}_h^{k+1}(E) = \{w_h \in \mathcal{H}^1(E) : w_h|_e \in C^0(e), w_h|_E \in \mathcal{P}_{k+1}(E), \Delta w_h|_E \in \mathcal{P}_{k-1}(E)\} \tag{2.4}$$

with the associated degrees of freedom as follows

- Vertex DOFs: the values of w_h at each vertex of E;

- Edge DOFs: the values of w_h at k internal Gauss-Lobatto quadrature points on each edge of E;

- Face DOFs: the moments up to degree $k-1$ of w_h in E: $\dfrac{1}{|E|}\int_E w_h \cdot p dE, \forall p \in \mathcal{P}_{k-1}(E)$.

Let n_v be the number of vertices of the polygon E. The dimension of the space $\mathcal{W}_{h,E}^{k+1}$ is

$$dim(\mathcal{W}_h^{k+1}(E)) = n_v + kn_v + dim(\mathcal{P}_{k-1}(E)) = (k+1)n_v + \frac{k(k+1)}{2}. \tag{2.5}$$

Similarly, the local VEM spaces of degree k for rotations $\boldsymbol{\theta}$ can be given by

$$\boldsymbol{\Theta}_h^k(E) = \{\boldsymbol{\theta}_h \in [\mathcal{H}^1(E)]^2 : \theta_h^i|_e \in C^0(e), \theta_h^i|_E \in \mathcal{P}_k(E), \Delta\theta_h^i|_E \in \mathcal{P}_{k-2}(E), i = 1, 2\} \tag{2.6}$$

with the associated degrees of freedom as follows

- Vertex DOFs: the values of $\boldsymbol{\theta}_h$ at each vertex of E;

- Edge DOFs: the values of $\boldsymbol{\theta}_h$ at $k-1$ internal Gauss-Lobatto quadrature points on each edge of E;

- Face DOFs: the moments up to degree $k-2$ of $\boldsymbol{\theta}_h$ in E: $\dfrac{1}{|E|}\int_E \boldsymbol{\theta}_h \cdot \boldsymbol{p} dE, \forall \boldsymbol{p} \in \boldsymbol{\mathcal{P}}_{k-2}(E)$.

Then the dimension of the spaces $\boldsymbol{\Theta}_h^k$ is calculated as

$$dim(\boldsymbol{\Theta}_h^k(E)) = 2n_v + 2(k-1)n_v + 2dim(\mathcal{P}_{k-1}(E)) = 2kn_v + k(k-1). \tag{2.7}$$

The Projection Operators:

Let Π_r^∇ be the local projection operator for vertical displacement, mapping the functions from the local space $\boldsymbol{\Theta}_h^k(E)$ to $\boldsymbol{\mathcal{P}}_k(E)$. Given $\boldsymbol{\theta}_h \in \boldsymbol{\Theta}_h^k(E)$, the projection operator Π_r^∇ satisfies:

$$a_h^E(\boldsymbol{\theta}_h, \boldsymbol{p}) = a_h^E(\Pi_r^\nabla \boldsymbol{\theta}_h, \boldsymbol{p}), \quad \forall \boldsymbol{p} \in \boldsymbol{\mathcal{P}}_k(E). \tag{2.8}$$

Assume that the function $\boldsymbol{\theta}_h$ could be expressed by the bases $\{\boldsymbol{\varphi}_i\}_{i=1}^{n_d}$ as $\boldsymbol{\theta}_h = \sum_{i=1}^{n_d} \boldsymbol{\varphi}_i \bar{\boldsymbol{\theta}}_i$, where n_d denotes the total number of basis functions, $\bar{\boldsymbol{\theta}}_i$ denotes the unknown rotations at ith DOF. Using the polynomial functions $\boldsymbol{p}_\alpha \in \boldsymbol{\mathcal{P}}_k(E)$ to express the projected function with $\Pi_r^\nabla \boldsymbol{\varphi}_i = \sum_{\alpha=1}^{n_k} \pi_{i,\alpha}^r \boldsymbol{p}_\alpha$, and combining Eqs. (2.2) and (2.8), a system of linear equation can be written as

$$a_h^E(\boldsymbol{\varphi}_i, \boldsymbol{p}_\beta) = \sum_{i=1}^{n_k} \pi_{i,\alpha}^r a_h^E(\boldsymbol{p}_\alpha, \boldsymbol{p}_\beta), \quad \forall \boldsymbol{p}_\beta \in \boldsymbol{\mathcal{P}}_k(E), \forall \boldsymbol{\varphi}_i \in \boldsymbol{\Theta}_h^k(E). \tag{2.9}$$

The right side term $a_h^E(\boldsymbol{p}_\alpha, \boldsymbol{p}_\beta)$ can be calculated directly and the left side term can be computed by using the rule of integration by parts as

$$a_h^E(\boldsymbol{\varphi}_i, \boldsymbol{p}_\beta) = \int_E \boldsymbol{\epsilon}^T(\boldsymbol{\varphi}_i) \boldsymbol{D}_b \boldsymbol{\epsilon}(\boldsymbol{p}_\beta) dE = - \int_E \boldsymbol{\varphi}_i^T \nabla \boldsymbol{D}_b \boldsymbol{\epsilon}(\boldsymbol{p}_\beta) dE + \int_e \boldsymbol{\varphi}_i^T \boldsymbol{D}_b \boldsymbol{\epsilon}(\boldsymbol{p}_\beta) \bar{\boldsymbol{n}} de, \tag{2.10}$$

where $\bar{\boldsymbol{n}}$ denotes the unit normal vector to the edges of E. Note that the integrand of the first term in the right side $\nabla \boldsymbol{D}_b \boldsymbol{\epsilon}(\boldsymbol{p}_\beta)$ could be expressed by using polynomials of degree $k-2$. Then we can compute the integration according to the predefined face DOFs. The integrand of the second term in the right side could be expressed by using polynomials of degree $2k-1$ and could be precisely calculated by using the $k+1$ Gauss-Lobatto quadrature points, namely the edge DOFs. Eventually, the coefficient $\boldsymbol{\pi}_r(\pi_{i,a})$ can be obtained by solving a system of linear equations. Considering the orthogonality condition of the projection operator Π_r^∇, the bilinear form $a_h^E(\boldsymbol{\theta}_h, \boldsymbol{\eta}_h)$ is computed by

$$a_h^E(\boldsymbol{\theta}_h, \boldsymbol{\eta}_h) = a_h^E(\Pi_r^\nabla \boldsymbol{\theta}_h, \Pi_r^\nabla \boldsymbol{\eta}_h) + a_h^E(\boldsymbol{\theta}_h - \Pi_r^\nabla \boldsymbol{\theta}_h, \boldsymbol{\eta}_h - \Pi_r^\nabla \boldsymbol{\eta}_h), \tag{2.11}$$

where the first term is called consistent term and the second term is called stability term. Combining Eqs. (2.2) and (2.8), the consistent term can be written as

$$a_h^E(\Pi_r^\nabla \boldsymbol{\theta}_h, \Pi_r^\nabla \boldsymbol{\eta}_h) = \bar{\boldsymbol{\theta}}^T \boldsymbol{\pi}_r^T \left\{ \int_E (\partial \boldsymbol{N}_p^r)^T \boldsymbol{D}_b (\partial \boldsymbol{N}_p^r) dE \right\} \boldsymbol{\pi}_r \bar{\boldsymbol{\eta}} = \bar{\boldsymbol{\theta}}^T \boldsymbol{K}_a^c \bar{\boldsymbol{\eta}}, \tag{2.12}$$

and the stability stiffness matrix derived from the stability term is written as

$$\boldsymbol{K}_a^s = \tau tr(\boldsymbol{K}_a^c)(\boldsymbol{I} - \boldsymbol{D}_r \boldsymbol{\pi}_r)^T(\boldsymbol{I} - \boldsymbol{D}_r \boldsymbol{\pi}_r), \tag{2.13}$$

in which τ is a positive real number and is chosen as $\tau = 0.5$ according to the recommendation given in [2]. The matrix $\boldsymbol{D}_r$ is constituted by the value of polynomials $\boldsymbol{p}_j$ at i-th DOFs as $\boldsymbol{D}_r^{ij} = dof_i(\boldsymbol{p}_j)$. ∂ is the gradient matrix and $\boldsymbol{N}_p^r$ is the matrix consisting of polynomials $\boldsymbol{p}_j$.

Next we consider the discrete bilinear form $b_h^E(\boldsymbol{\theta} - \nabla w, \boldsymbol{\eta} - \nabla v)$, which can be expanded as

$$b_h^E(\boldsymbol{\theta} - \nabla w, \boldsymbol{\eta} - \nabla v) = b_h^E(\nabla w, \nabla v) - b_h^E(\nabla w, \boldsymbol{\eta}) - b_h^E(\boldsymbol{\theta}, \nabla v) + b_h^E(\boldsymbol{\theta}, \boldsymbol{\eta}), \tag{2.14}$$

where the first term $b_h^E(\nabla w, \nabla v)$ can be calculated using the similar way for computation of $a_h^E(\boldsymbol{\theta}_h, \boldsymbol{\eta}_h)$. The derived consistent stiffness matrix $\boldsymbol{K}_{bw}^c$ and stability stiffness matrix $\boldsymbol{K}_{bw}^s$ are written as

$$\boldsymbol{K}_{bw}^c = \boldsymbol{\pi}_w^T \left\{ \int_E (\nabla \boldsymbol{N}_p^w)^T \boldsymbol{D}_s (\nabla \boldsymbol{N}_p^w) dE \right\} \boldsymbol{\pi}_w \tag{2.15}$$

$$\boldsymbol{K}_{bw}^s = \tau tr(\boldsymbol{K}_{bw}^c)(\boldsymbol{I} - \boldsymbol{D}_w \boldsymbol{\pi}_w)^T(\boldsymbol{I} - \boldsymbol{D}_w \boldsymbol{\pi}_w) \tag{2.16}$$

The fourth term $b_h^E(\boldsymbol{\theta}, \boldsymbol{\eta})$ is computed through an equivalent projection operator Π_r^0 as introduced in [1], and the derived stiffness matrix $\boldsymbol{K}_{br}^c$ and stability stiffness matrix $\boldsymbol{K}_{br}^s$ are given as

$$\boldsymbol{K}_{br}^c = \boldsymbol{\pi}_r^{0T} \int_E (\nabla \boldsymbol{N}_p^r)^T \boldsymbol{D}_s (\nabla \boldsymbol{N}_p^r) dE \boldsymbol{\pi}_r^0 \tag{2.17}$$

$$\boldsymbol{K}_{br}^s = \tau tr(\boldsymbol{K}_{br}^c)(\boldsymbol{I} - \boldsymbol{D}_r \boldsymbol{\pi}_r^0)^T(\boldsymbol{I} - \boldsymbol{D}_r \boldsymbol{\pi}_r^0) \tag{2.18}$$

The third term $b_h^E(\boldsymbol{\theta}, \nabla v)$ is the symmetric part of the second term $b_h^E(\nabla w, \boldsymbol{\eta})$, which can be computed using the rule of integration by parts as

$$b_h^E(\nabla w, \boldsymbol{\eta}) = \int_e w^T \boldsymbol{\eta} \boldsymbol{n} de - \int_E w^T \nabla \boldsymbol{\eta} dE = \bar{\boldsymbol{w}} \boldsymbol{K}_{wr} \bar{\boldsymbol{\eta}}. \tag{2.19}$$

Note that w is a polynomial of $k+1$ and $\boldsymbol{\eta}$ is the polynomial of k. Therefore the term $\int_{\partial E} w^T \boldsymbol{\eta} \boldsymbol{n} d\partial E$ can be obtained by computing the integrations on the $k+2$ Gauss-Lobatto quadrature points (edge DOFs), and the term $\int_E w^T \nabla \boldsymbol{\eta} dE$ can be calculated by using the face DOFs predefined for the space $\mathcal{W}_h^{k+1}(E)$.

Eventually the stiffness matrix induced by bilinear forms $a_h^E(\boldsymbol{\theta}_h, \boldsymbol{\eta}_h)$ and $b_h^E(\boldsymbol{\theta} - \nabla w, \boldsymbol{\eta} - \nabla v)$ can be expressed as

$$\boldsymbol{K} = \begin{bmatrix} \boldsymbol{K}_{bw}^c + \boldsymbol{K}_{bw}^s & \boldsymbol{K}_{wr} \\ \boldsymbol{K}_{wr}^T & \boldsymbol{K}_a^c + \boldsymbol{K}_a^s + \boldsymbol{K}_{br}^c + \boldsymbol{K}_{br}^s \end{bmatrix}. \tag{2.20}$$

The external force vector $\boldsymbol{F}$ can be computed using the scheme presented in [5].
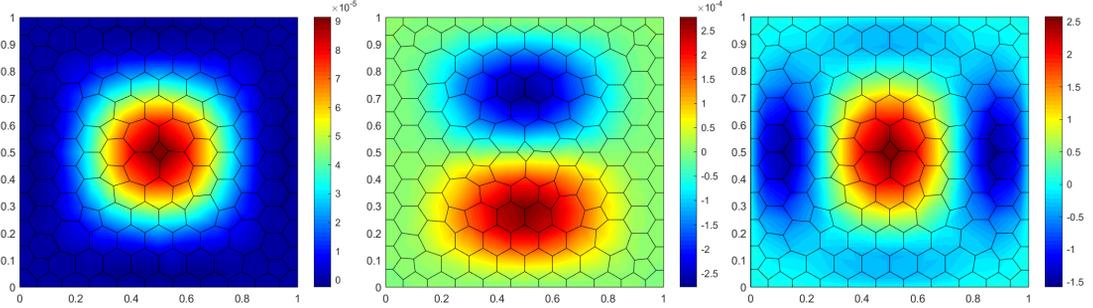
Fig. 1: Plots of numerical results for the unit square plate under a transverse load. Left: Vertical displacement w; Middle: Rotation θ_x; Right: Bending moment M_x.

Numerical Examples:

A benchmark problem is studied in this section to verify the developed method on the static bending analysis of Reissner-Mindlin plates. Consider a clamped unit square plate $\Omega \in [0,1]^2$ subjected to a transverse load g with the expression

$$\begin{aligned} f(x,y) = &\frac{E}{12(1-\nu^2)}[12y(y-1)(5x^2-5x+1)(2y^2(y-1)^2 + x(x-1)(5y^2-5y+1)) \\ &+ 12x(x-1)(5y^2-5y+1)(2x^2(x-1)^2 + y(y-1)(5x^2-5x+1))]. \end{aligned} \tag{2.21}$$

The analytical solutions of vertical displacement and rotations could be found in [4]. The material parameters are taken as: $E = 10.92 \times 10^6, \nu = 0.3, t = 0.1$. Figure 1 shows the color plots of vertical displacement w, rotation θ_x and bending moment M_x obtained by using the developed p_w^3/p_r^2 virtual element method. Here p_w^{k+1}/p_r^k is used to state that the vertical displacement is interpolated with functions of degree $k+1$ and rotations are interpolated with functions of degree k. The total number of elements and DOFs are 137 and 3413, respectively.

To investigate the convergence, we first define a L_2-like relative error e_h^w for the vertical displacement as

$$(e_h^w)^2 = \frac{\sum_{E \in \mathcal{T}_h} \int_E (w_{ex} - \Pi w_h)^2 dE}{\sum_{E \in \mathcal{T}_h} \int_E w_{ex}^2 dE}. \tag{2.22}$$

The error for rotations can be similarly defined by substituting $\boldsymbol{\theta}$ for w. To better describe the mesh size and compute the errors, we discretize the square plate into structured rectangular mesh. Figure 2 presents the relative errors e_w and e_{θ_x} with respect to mesh size h under three cases: $p_w^2/p_r^1, p_w^3/p_r^2, p_w^4/p_r^3$. It can be found that both the vertical displacement w and rotation θ_x could achieve the optimal convergence rate.
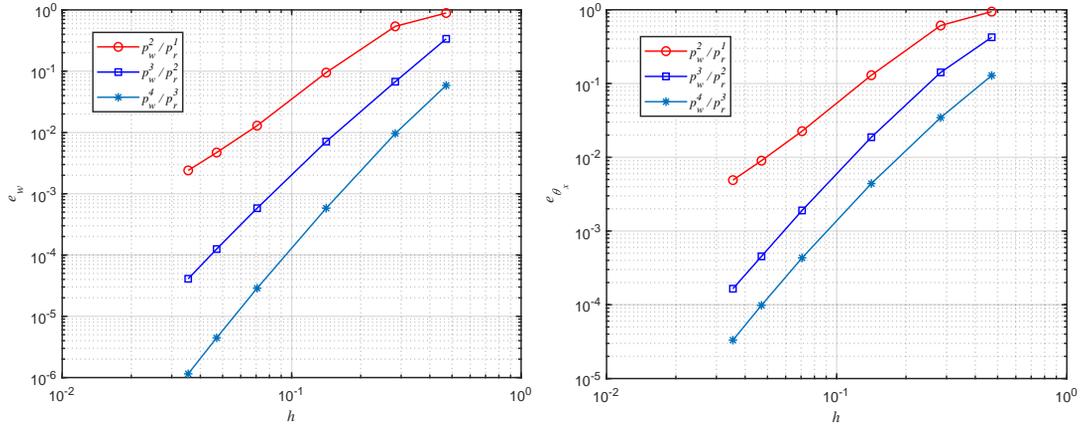
Fig. 2: Relative errors of vertical displacement (Left) and rotation θ_x (Right) with respect to mesh size.

Conclusions:

A novel virtual element method is developed for the static bending analysis of Reissner-mindlin plate by using k-degree functions for interpolation of rotations and $(k+1)$-degree for interpolation of vertical displacement. The numerical results show optimal convergence rates for the vertical displacement and rotations.

Acknowledgement:

References:

[1] Ahmad, B.; Alsaedi, A.; Brezzi, F.; Marini, L.D.; Russo, A.: Equivalent projectors for virtual element methods. Computers & Mathematics with Applications, 66(3), 376–391, 2013. http://doi.org/10.1016/j.camwa.2013.05.015.

[2] Artioli, E.; da Veiga, L.B.; Lovadina, C.; Sacco, E.: Arbitrary order 2d virtual elements for polygonal meshes: part i, elastic problem. Computational Mechanics, 60(3), 355–377, 2017. http://doi.org/10.1007/s00466-017-1404-5.

[3] da Veiga, L.B.; Brezzi, F.; Cangiani, A.; Manzini, G.; Marini, L.D.; Russo, A.: Basic principles of virtual element methods. Mathematical Models and Methods in Applied Sciences, 23(01), 199–214, 2013. http://doi.org/10.1142/S0218202512500492.

[4] Du, X.; Zhao, G.; Wang, W.: Nitsche method for isogeometric analysis of reissner–mindlin plate with non-conforming multi-patches. Computer Aided Geometric Design, 35, 121–136, 2015. http://doi.org/10.1016/j.cagd.2015.03.005.

[5] Mengolini, M.; Benedetto, M.F.; Aragón, A.M.: An engineering perspective to the virtual element method and its interplay with the standard finite element method. Computer Methods in Applied Mechanics and Engineering, 350, 995–1023, 2019. http://doi.org/10.1016/j.cma.2019.02.043.