

Dear Editors and Reviewers:

Thank you to all the editors and reviewers for the constructive suggestions and comments on our manuscript, which greatly help us improve the quality of this paper. We have studied the comments carefully and have addressed all the comments on a point-by-point basis in this reply. All changes made are highlighted in the revised manuscript "Revised Manuscript.pdf", the clean version of manuscript is "Manuscript1.pdf".

Comments 1: In Subsection 2.1: If the derivations of the partial derivatives are novel, then the authors must mention that clearly at the beginning of that subsection. If not, then the authors must also clarify that they are using conventional derivations at this stage.

Response 1: Yes, this paper presents a new method for calculating partial derivatives, different from traditional numerical methods. We have added such an emphasis in Section 2.1 based on your comments.

Comments 2: As I can see in Section 2, the approach that the authors used is similar to the SPH method, and barely I can find a difference. Could you please write few sentences to show the main difference between your approach and the SPH method?

Response 2: I believe that the method proposed in this paper fundamentally differs from the SPH method, and below is my comparison of the two methods. Please correct me if I am wrong: The SPH method uses kernel functions to approximate the Dirac function through kernel approximation, while the kernel function introduced in this paper is not used to approximate the Dirac function. Additionally, the SPH method utilizes partial integration and the Gauss theorem among other weak form techniques to obtain the first derivative of the function, as shown in equation (1).

$$\langle \nabla \cdot f(x) \rangle = \int_S f(x') w(x - x', h) \cdot \vec{n} dS - \int_{\Omega} f(x') \nabla w(x - x', h) dx' \quad (1)$$

When the support domain of the kernel function w is completely within the problem domain, equation (2) can be derived:

$$\langle \nabla \cdot f(x) \rangle = - \int_{\Omega} f(x') \nabla w(x - x', h) dx' \quad (2)$$

However, at the boundary, since the support domain of the kernel function is truncated by the boundary, it loses its symmetry, making it impossible to neglect the surface integral terms, which leads to significant errors. This paper does not follow these steps; instead, it derives the various order partial derivatives of the function by multiplying the function by the kernel function and integrating using the Taylor expansion form:

$$\int d^T w_j f(x_j) d\Omega = \int w_j d^T d\mathcal{C} d\Omega \times \partial f \quad (3)$$

Where ∂f includes the various order partial derivatives of $f(x)$.

Comments 3: Lines 103-107: the explanation must be improved, especially for those reader with no background. For example, it is not clear enough what is h and the authors did not present it in figure 1. To help you to improve your explanation, I would recommend you use the same manner used in introducing the SPH method; for instance, check Section 3 in the following paper: "Alm rio Jos  Ven ncio Pains Soares Pamplona. Numerical Solution using SPH Method: An Application to Transient Heat Conduction 2019."

Response 3: Based on the reference materials you provided, we have revised the images

and text in the article to make it easier for readers without relevant background to understand the content.

Comments 4: In Eq. (1), I guess that the authors dropped accidentally the field function $f(X)$ after the high order derivation.

Response 4: You are right, and I apologize for this oversight. We have already made corrections in the manuscript.

Comments 5: In Eq. (2), on the right-hand side, the authors wrote the term " $\frac{1}{2} \xi_x^j \xi_y^j \frac{\partial^2}{\partial y} f(x)$ ".

I guess the correct term is " $\frac{1}{2} \xi_y^j \xi_y^j \frac{\partial^2}{\partial y} f(x)$ ".

Response 5: Thank you for your correction; we have made the changes in the manuscript.

Comments 6: If the authors think that Eq. (2) must be corrected, then Eq. (3) must be corrected accordingly. Moreover, if the authors think that Eq. (2) must be corrected, then they should check if this correction (modification) will affect the rest of the equations; Eqs. (4) to (35).

Response 6: Thank you for your suggestion. We have carefully reviewed all the equations in the article, and we confirm that the errors were only in equations (2) and (3).

Comments 7: Could you clarify how to calculate matrix K in Eq. (34)?

Response 7: We have added a description of matrix K in the manuscript. K is actually a coefficient matrix, composed of the governing equations (4), displacement equations (5) and (6), and stress equations (7) and (8).

$$\frac{E}{1-\nu^2} \begin{bmatrix} G_1^4 + bG_1^6 & aG_1^5 \\ \vdots & \vdots \\ G_M^4 + bG_M^6 & aG_M^5 \\ aG_1^5 & bG_1^4 + G_1^6 \\ \vdots & \vdots \\ aG_M^5 & bG_M^4 + G_M^6 \end{bmatrix} \begin{bmatrix} u_{x_1} \\ \vdots \\ u_{x_M} \\ u_{y_1} \\ \vdots \\ u_{y_M} \end{bmatrix} = - \begin{bmatrix} \bar{f}_{x_1} \\ \vdots \\ \bar{f}_{x_M} \\ \bar{f}_{y_1} \\ \vdots \\ \bar{f}_{y_M} \end{bmatrix} \quad (4)$$

$$e \begin{bmatrix} G_i^1 & 0 \end{bmatrix} \begin{bmatrix} u_{x_1} \\ \vdots \\ u_{x_M} \\ u_{y_1} \\ \vdots \\ u_{y_M} \end{bmatrix} = e \begin{bmatrix} \bar{u}_{x_i} \end{bmatrix} \quad (5)$$

$$e \begin{bmatrix} 0 & G_i^1 \end{bmatrix} \begin{bmatrix} u_{x_1} \\ \vdots \\ u_{x_M} \\ u_{y_1} \\ \vdots \\ u_{y_M} \end{bmatrix} = e \begin{bmatrix} \bar{u}_{y_i} \end{bmatrix} \quad (6)$$

$$e \frac{E}{1-\nu^2} \begin{bmatrix} n_x G_i^2 + n_y b G_i^3 & n_y b G_i^2 + n_x \nu G_i^3 \end{bmatrix} \begin{bmatrix} u_{x_1} \\ \vdots \\ u_{x_M} \\ u_{y_1} \\ \vdots \\ u_{y_M} \end{bmatrix} = e \begin{bmatrix} \bar{t}_{x_i} \end{bmatrix} \quad (7)$$

$$e \frac{E}{1-\nu^2} \begin{bmatrix} n_y v G_i^2 + n_x b G_i^3 & n_x b G_i^2 + n_y G_i^3 \end{bmatrix} \begin{bmatrix} u_{x_1} \\ \vdots \\ u_{x_M} \\ u_{y_1} \\ \vdots \\ u_{y_M} \end{bmatrix} = e [\bar{t}_{y_i}] \quad (8)$$

The rows corresponding to internal points are typically composed only of the governing equations, while the rows corresponding to boundary points are composed of the governing equations combined with the corresponding displacement and stress boundary equations multiplied by penalty coefficients.

Comments 8: If Eq. (35) represents the new system of equations based on your novel approach, then please claim that clearly.

Response 8: Equation (35) is not a new set of equations, but rather the displacement components of each node obtained by inverting Equation (34). To avoid misunderstanding, we have made modifications, as shown in Equation (9):

$$U = \begin{bmatrix} u_{x_1} \\ \vdots \\ u_{x_M} \\ u_{y_1} \\ \vdots \\ u_{y_M} \end{bmatrix} = K^{-1} \bar{f} \quad (9)$$

Comments 9: In the numerical experiment, Cantilever Beam, I would recommend the author to add another figure which shows the beam in the initial position, and after the deformation. However, the authors can do the same with the other experiments if it is comfortable.

Response 9: Thank you for your suggestion. We have added comparison diagrams of the initial positions and the changes after deformation in the examples, to make the problem more visually understandable.

Comments 10: According to my experience in the fields of SPH and FDM for solving the heat equation, the real physical time to achieve the simulation of a system with large number of nodes (Particles in case of SPH) might take few minutes, to few hours in some cases. In your experiments, you did measure the real physical time to evaluate the computational cost of the methods. Instead, the authors used the CPU, which is, in my opinion, is not the proper way to evaluate the computational cost. As far as I know, when using Matlab, the **tic toc** stopwatch timer is used in order to measure the cost of coded scheme.

Response 10: Your suggestion is highly professional and cautious. In our source code, we erroneously used the tic-toc method as a way to measure CPU computation time, whereas the correct function to accurately measure CPU computation time is `cputime`. Tic-toc actually evaluates the execution time of the program, so we have corrected the term "CPU" to "execution time" in the original text. You also mentioned that your numerical simulations usually take several minutes or even hours, which might be due to differences in our respective research fields and specific cases.

Comments 11: In Subsection 3.1.2, I really appreciate the work you did to analyse the effect of kernel, but still, you have to explain the reason why the method is less accurate when using quintic kernel. I know that this idea can be a research topic by itself, but the authors should introduce a simple experiment that might encourage other researchers to take the

reasons. I have a suggestion that might be useful to detect the reason. For a system with $N = 1701$ nodes, First, you can calculate the density for a uniform system, when $\mu = 0$, which can be used as a reference density. Second, for the same number of nodes $N = 1701$, but $\mu = 0.4$, we calculate the density at each point source using three Kernel functions which are Gaussian, Cubic spline, and Quintic spline. The density can be calculated by modifying Eq. (16) to become $\rho_i = \sum W(x_i - x_j, h)m_j$, where i refers to the source point and j refers to its neighbours. Now we have the density as a function of the node source points; on other words, the density is function of the position of each node $r_i = \sqrt{x_i^2 + y_i^2}$. Comparing those functions with reference density can provide us with an estimation about the low accuracy of Quintic Kernel. I guess the density calculated by Quintic Kernel will fluctuate more than others.

Response 11: Thank you very much for your detailed and insightful suggestions. After further reviewing the relevant literature, we believe that the poor robustness of the quintic kernel in experiments may be due to its high order, which likely causes overfitting. This will become an important direction for our future research. However, the main focus of this paper is to verify the feasibility and significant advantages of the proposed mesh-free method in static applications, so we may not extensively discuss the robustness of different kernel functions in this paper.

Comments 12: Line 33: The authors mentioned that „ computational cost of the FEM is usually too high ...“. In my opinion the authors should clarify comparing to which method the cost is high, for example, is it expensive comparing to FVM or FDM...? Also, I would recommend you to mention at least one reference.

Response 12: Thank you for your reminder. We have revised the relevant text and added references comparing the computational efficiency and cost of the finite element method to the mesh-free method.

Comments 13: When numbering the equations, the notation “#” appears unnecessarily. Please check this typo in all equations.

Response 13: I'm very sorry for the inconvenience caused to your reading. I believe this issue arose from errors that occurred during the conversion from Word to PDF format. We will pay attention to this problem in the manuscripts we submit in the future.

Comments 14: In Eq. (3), since each internal node has its own vector d , then I would recommend you to label the vector d with subscript j .

Response 14: Thank you very much. We have made revisions to the relevant text.

Comments 15: I guess there should be spaces between equations, for example, Eq. (3) and (4). Another space is necessary between line 126 and Eq. (8). Please check this type of spaces everywhere in the paper.

Response 15: Like Comment 13, I believe this was caused by errors during the conversion from Word to PDF format, and we will make improvements.

Comments 16: In Eq. (6), I would prefer to separate the matrices multiplication dC by $\times$ to avoid any misunderstanding.

Response 16: We have added a multiplication sign $\times$ to the relevant equations to avoid misunderstanding.

Comments 17: line 124, Please mention at least one reference where the reader can check the

formula of Kernel function. I can suggest the following paper “Philip Mocz, Smoothed Particle Hydrodynamics: Theory, Implementation, and Application to Toy Stars. 2011”. In that reference you can find the kernel formulas in Eqs. (6) and (7). However, you can choose another suitable reference

Response 17: Thank you, we have added references that include all the kernel functions discussed in this paper.

Comments 18: In figure 5. (a), I guess that the label of the vertical axis should be “*erroru*” instead of *errorμ*. In figure 5. (b), I guess that the label of horizontal axis should be “*μ*” instead of “*σ*”.

Response 18: We have reviewed all the images in the article and replaced all incorrect ones. Thank you for your correction.