ICCM Proceedings

**Proceedings
of the International Conference
on Computational Methods**

(Vol. 9, 2022)

13th ICCM, 25th-28th July 2022

Editors: **G. R. Liu, Nguyen-Xuan Hung**

ICCM2022

Proceedings of the International Conference on Computational Methods (Vol. 9, 2022)

13th ICCM, 25th-28th July 2022, Virtual Conference

Edited by

G. R. Liu
University of Cincinnati, USA

Nguyen-Xuan Hung
HUTECH University of Technology, Vietnam

WELCOME MESSAGE

Dear Colleagues and Friends,

It is with great pleasure that we welcome you to the 13th International Conference on Computational Methods (ICCM2022) which will be held online via Zoom from July 25 to July 28, 2022, by Ho Chi Minh University of Technology (HUTECH), Vietnam.

The ICCM is an international conference which has been serving as an important forum for exchanging ideas on recent advances in areas related to the computational methods and the numerical modeling of both man-made and natural systems. The conference offers presentations for a wide range of topics to facilitate the exchange of ideas from multiple disciplines and foster academic collaborations. Publications, which have been peer-reviewed and accepted, will be showcased through oral presentations at the conference. All presentations, including abstracts and papers, will be published on our conference website. The online presentation schedule will be posted on the website one week before the conference.

The ICCM conference series were originated in Singapore in 2004, followed by ICCM2007 in Hiroshima, Japan; ICCM2010 in Zhangiajie, China; ICCM2012 in Gold Coast, Australia; ICCM2014 in Cambridge, England; ICCM2015 in Auckland, New Zealand; ICCM2016 in Berkeley, CA, USA; ICCM2017 in Guilin, China; ICCM2018 in Rome, Italy; ICCM2019 in Singapore; ICCM2020, ICCM2021 and ICCM2022 on the cloud.

We would like to express our appreciation to all members of the Organizing Committee, the International Scientific Committee, and all supporters who have been working tirelessly to make this conference possible. Also, we would like to thank the international reviewers for their meticulous work on reviewing the submitted abstracts and papers. Finally, we would like to thank you for your contribution to the ICCM conferences.

We look forward to welcoming you to the ICCM2022 and we hope to have your continued engagement for future ICCM conferences.

Professor Nguyen-Xuan Hung
Conference Chairman
CIRTECH Institute, HUTECH University of Technology
President of Vietnam Association of Computational Mechanics
Vietnam

Professor Gui-Rong Liu
Honorary Conference Chairman
University of Cincinnati
USA

ORGANIZATION COMMITTEES

Conference Chairman
Nguyen-Xuan Hung, Ho Chi Minh City University of Technology (HUTECH), Vietnam

Honorary Chairman
Guirong Liu, University of Cincinnati, United States

International Co-Chairs
Magd Abdel-Wahab, Ghent University, Belgium
Stephane P.A. Bordas, Luxembourg University, Luxembourg
Tinh Quoc Bui, Tokyo Institute of Technology, Japan
Ha Bui, Monash University, Australia
Daining Fang, Beijng Institute of Technology, China
Jaehong Lee, Sejong University, South Korea
Hua Li, Nanyang Technological University, Singapore
Tuan Ngo, The University of Melbourne, Australia
Hiroshi Okada, Tokyo University of Science, Japan
Timon Rabczuk, Bauhaus University Weimar, Germany
Dia Zeidan, German Jordanian University, West Asia

Local Co-Chairmen
Canh Van Le, International University-VNU-HCMC
Hung Quoc Nguyen, Vietnam-German University, Vietnam
Kien Trung Nguyen, Ho Chi Minh City University of Technology and Education, Vietnam
Van Hai Luong, VNUHCM-University of Technology, Vietnam

Secretary General
Nhung Ngoc Hoang, Ho Chi Minh City University of Technology (HUTECH), Vietnam Vuong Van Nguyen, Ho Chi Minh City University of Technology (HUTECH), Vietnam

Local Organizing Committee
Anh Ngoc Lai, Binh Anh Tran, Bang Quang Tao, Cuong Huu Ngo, Chien Hoang Thai, Bao Loi Dang, Hieu Van Nguyen, Long Minh Nguyen, Linh Ngoc Nguyen, Lieu Bich Nguyen, Phuc Hong Pham, Phuc Van Phung, Phuong Tran, Phuoc Trong Nguyen, Nam Van Hoang, Nghi Van Vu, Son Hoai Nguyen, Thanh Dinh Chau, Truong Van Vu, Viet Duc La, Binh Le, Hien Van Do, Tuan Ngoc Nguyen

**International Scientific Advisory Committee
(ordered by last name)**

Armas Rafael Montenegro (Spain)
Bui Ha (Australia)
Chen Bin (China)
Chen Jeng-Tzong (Taiwan)
Chen Jiye (UK)
Chen Shaohua (China)
Chen Songying (China)
Chen Weiqiu (China)
Chen Zhen (USA)
Cheng Yuan (Singapore)
Cheng Yumin (China)
Cui Fangsen (Singapore)
Dias-da-Costa Daniel (Australia)
Dong Leiting (China)
Fan Chia-Ming (Taiwan)
Fu Zhuojia (China)
Gan Yixiang (Australia)
Gao Wei (Australia)
Gravenkamp Hauke (Germany)
Gu Yuantong (Australia)
Gupta Murli (USA)
Hou Shujuan (China)
Jabareen Mahmood (Israel)
Jacobs Gustaaf (USA)
Jin Feng (China)
Jiang Chao (China)
Kanayama Hiroshi (Japan)
Kang, Zhan (China)
Kougioumtzoglou Ioannis (USA)
Le Van Canh (Vietnam)
Lee Chin-Long (New Zealand)
Lee Ik-Jin (South Korea)

Lenci Stefano (Italy)
Li Chenfeng (UK)
Li Eric (UK)
Li Qing (Australia)
Li Yue-Ming (China)
Liu Yan (China)
Liu Yinghua (China)
Lombardi Domenico (UK)
Miller Karol (Australia)
Nguyen Anh Dong (Vietnam)
Nguyen Duc Dinh (Vietnam)
Nguyen Giang (Australia)
Nithiarasu Parumal (UK)
Niu Yang-Yao (Taiwan)
Ogino Masao (Japan)
Onishi Yuki (Japan)
Peng Qing (USA)
Picu Catalin (USA)
Quek Jerry Sinsin (Singapore)
Reali Alessandro (Italy)
Rebielak Janusz (Poland)
Reddy Daya (South Africa)
Sadowski Tomasz (Poland)
Saitoh Takahiro (Japan)
Shen Lian (USA)
Shen Luming (Australia)
Shioya Ryuji (Japan)
Son Gihun (South Korea)
Song Chongmin (Australia)
Stefanou George (Greece)
Su Cheng (China)
Tadano Yuichi (Japan)
Tian Zhao-Feng (Australia)
Trung Nguyen-Thoi (Vietnam)
Tsubota Ken-Ichi (Japan)

Wan Decheng (China)
Wang Dongdong (China)
Wang Hu (China)
Wang Jie (China)
Wang Lifeng (China)
Wang Yue-Sheng (China)
Wu Bin (Italy)
Wu Wei (Austria)
Xiang Zhihai (China)
Xiao Feng (Japan)
Xiao Jinyou (China)
Xu Chao (ZJU, China)
Xu Chao (NPU, China)
Xu Xiangguo George (Singapore)
Yang Judy (Taiwan)
Yang Qingcheng (China)
Yang Qingsheng (China)
Yang Zhenjun (China)
Yao Jianyao (China)
Ye Hongling (China)
Ye Qi (China)
Yeo Jingjie (USA)
Zhang Chuanzeng (Germany)
Zhang Guiyong (China)
Zhang Jian (China)
Zhang Lihai (Australia)
Zhang Yixia Sarah (Australia)
Zhang Zhao (China)
Zhao Liguo (UK)
Zheng Hui (China)
Zhong Zheng (China)
Zhou Anna (Australia)
Zhou Kun (Singapore)
Zhuang Zhuo (China)

TABLE OF CONTENTS

Alternative formulations for finite elements for thick plate analysis

†*J. Petrolito and D. Ionescu

[1]Department of Engineering, La Trobe University, Australia.

*Presenting author: j.petrolito@latrobe.edu.au
†Corresponding author: j.petrolito@latrobe.edu.au

Abstract

This paper discusses the use of the free formulation and the deviatoric strain formulation for developing orthotropic thick plate finite elements. The interpolations for the transverse displacement and rotations are such that the governing equations of Mindlin's theory are exactly satisfied. Hence, the resulting element approximations are consistent for both thick and thin plates, and no locking occurs in the thin plate limit. The two formulations are used to derive rectangular and triangular elements. Examples are given in the paper to demonstrate the accuracy that is achieved with the proposed elements.

Keywords: Plate analysis; thick plates; shear deformation; finite elements.

Introduction

Plate structures are in common use in many areas of structural engineering, such as buildings, bridges and foundations. Engineers need reliable and efficient methods for plate analysis, and there is a vast literature in this area. While analytical solutions are available [1], they only apply to a restricted range of problems. Hence, practical plate problems are normally solved by numerical methods. Currently, the finite element method is normally used for solving plate problems, due to its generality and ability to deal with complex geometries [2].

The plate bending problem has been modelled by many different theories. In the classical theory [1], the plate thickness is assumed to be small in comparison with a typical plan dimension of the plate, and shear deformation is neglected. In this case, the problem leads to the biharmonic equation in terms of the transverse displacement as the primary unknown. A conforming finite element for this theory requires C_1 continuity, and this proved to be a stumbling block in the early development of plate finite elements. Methods of circumventing the C_1 continuity requirement have been the subject of much research over many years [2].

An approach that can be used to avoid the difficulties with the classical theory is to use a higher-order theory, such as Mindlin's shear deformation theory [1]. This theory is applicable to both thick and thin plates, and it incorporates independent assumptions for the transverse displacement and normal rotations. This allows C_0 approximations to be used for these variables, and hence the C_1 continuity problem is avoided. However, this approach can also suffer from some problems. The two major problems for thick plate elements are the possibility of spurious mechanisms that lead to unstable meshes, and locking of the solution in the case of thin plates. Numerous approaches, such as reduced integration methods, special shear interpolation methods and stabilization matrices methods have been used to alleviate these problems [2].

The current paper discusses the use of interpolations for the transverse displacement and rotations that are solutions of the governing equations of Mindlin's theory. The resulting element approximations are consistent for both thick and thin plates. Hence, no locking occurs in the thin plate limit. A number of alternative formulations are possible based on such interpolations. In this paper, we consider the use of the free formulation [3, 4] and the deviatoric strain formulation [5] for deriving thick plate elements. The details of the two formulations are presented and compared, and they are used to derive rectangular and triangular elements. Examples are given to demonstrate the accuracy that is achieved with the proposed elements.

Governing Equations

Mindlin's theory uses independent assumptions for the transverse displacement, w, and normal rotations, θ_x and θ_y, (see Fig. 1).
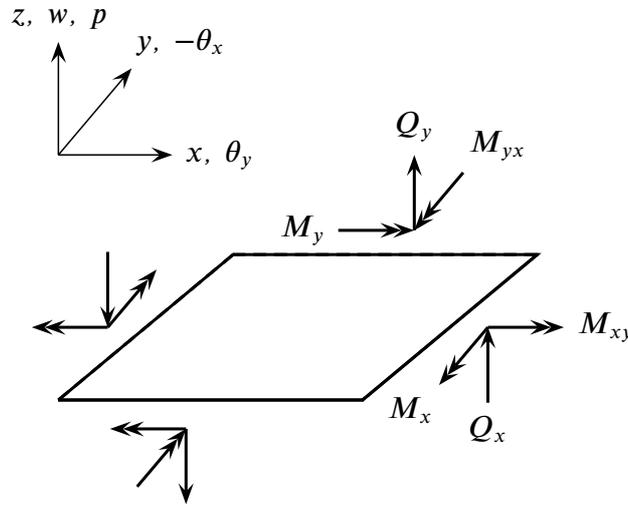
Figure 1. Sign convention.

Both bending and transverse shear strains are possible in this theory, and these are given by

$$\epsilon_b = \begin{bmatrix} \partial/\partial x & 0 \\ 0 & \partial/\partial y \\ \partial/\partial y & \partial/\partial x \end{bmatrix} \begin{Bmatrix} \theta_x \\ \theta_y \end{Bmatrix} = \mathbf{D}_1 \theta \tag{1a}$$

$$\epsilon_s = \begin{Bmatrix} \partial/\partial x \\ \partial/\partial y \end{Bmatrix} w - \theta = \mathbf{D}_2 w - \theta \tag{1b}$$

where ϵ_b is the bending strain vector, ϵ_s is the shear strain vector, which is zero in the thin plate limit, and $\mathbf{D}_1$ and $\mathbf{D}_2$ are matrices of differential operators.

For an orthotropic material, bending moments and shear forces are given in terms of the strains by

$$\mathbf{M} = \begin{Bmatrix} M_x \\ M_y \\ M_{xy} \end{Bmatrix} = \begin{bmatrix} D_x & D_1 & 0 \\ D_1 & D_y & 0 \\ 0 & 0 & D_{xy} \end{bmatrix} \epsilon_b = \mathbf{E}_b \epsilon_b \tag{2a}$$

$$\mathbf{Q} = \begin{Bmatrix} Q_x \\ Q_y \end{Bmatrix} = kt \begin{bmatrix} G_{xz} & 0 \\ 0 & G_{yz} \end{bmatrix} \epsilon_s = \mathbf{E}_s \epsilon_s \tag{2b}$$

2

with $M_{yx} = M_{xy}$. In Eq. (2), D_x, D_y, D_1 and D_{xy} are bending rigidities, G_{xz} and G_{yz} are transverse shear moduli, t is the plate thickness and k is the shear correction factor, which is usually taken as $5/6$ or $\pi^2/12$.

The terms in $\mathbf{E}_b$ are given in terms of the fundamental material properties by

$$D_x = \frac{E_x t^3}{12(1 - v_{xy}v_{yx})}, \quad D_y = \frac{E_y t^3}{12(1 - v_{xy}v_{yx})}, \quad D_1 = \frac{v_{xy}E_y t^3}{12(1 - v_{xy}v_{yx})}, \quad D_{xy} = \frac{G_{xy} t^3}{12} \quad (3)$$

where E_x and E_y are Young's moduli, v_{xy} and $v_{yx} = v_{xy}E_y/E_x$ are Poisson's ratios, and G_{xy} is the in-plane shear modulus. For an isotropic material, $D_x = D_y = D = Et^3/[12(1 - v^2)]$, $D_1 = vD$, $D_{xy} = D(1 - v)/2$ and $G_{xy} = G_{xz} = G_{yz} = E/[2(1 + v)]$.

The static equilibrium equations are

$$\mathbf{D}_3\sigma + \mathbf{p} = \mathbf{0} \quad (4)$$

where

$$\sigma = \begin{Bmatrix} \mathbf{M} \\ \mathbf{Q} \end{Bmatrix}, \quad \mathbf{p} = \begin{Bmatrix} \mathbf{0} \\ p \end{Bmatrix}, \quad \mathbf{D}_3 = \begin{bmatrix} \mathbf{D}_1^T & \mathbf{I} \\ \mathbf{0} & \mathbf{D}_2^T \end{bmatrix} \quad (5)$$

In Eq. (5), p is the applied transverse load, and $\mathbf{0}$ and $\mathbf{I}$ represent a zero matrix and an identity matrix of appropriate sizes, respectively.

Equations (1) to (5) are the governing equations for Mindlin's theory, and they are a system of coupled differential equations of order six. Hence, three boundary conditions need to be specified at any point on the boundary of the region being analysed.

In the thin plate limit, the only independent variable is w, and the governing differential equation in this case is

$$D_x \frac{\partial^4 w}{\partial x^4} + 2H \frac{\partial^4 w}{\partial x^2 \partial y^2} + D_y \frac{\partial^4 w}{\partial y^4} = p \quad (6)$$

where $H = D_1 + 2D_{xy}$ is the effective torsional rigidity. For an isotropic material, Eq. (6) reduces to the biharmonic equation

$$\frac{\partial^4 w}{\partial x^4} + 2\frac{\partial^4 w}{\partial x^2 \partial y^2} + \frac{\partial^4 w}{\partial y^4} = \frac{p}{D} \quad (7)$$

Element Approximations

The key feature of the elements developed in this paper is that the element approximations satisfy the governing differential equations for Mindlin's theory. The approximations for w, θ_x and θ_y are taken as

$$\mathbf{u} = \begin{Bmatrix} \theta \\ w \end{Bmatrix} = \mathbf{u}_0 + \mathbf{u}_p \quad (8)$$

where the subscript 0 denotes the homogenous solutions of Eq. (4) with $p = 0$, and the subscript p denotes a particular solution of Eq. (4) for the specified load p. The use of the particular solution is optional, and its inclusion can improve the accuracy of the approximation in some

cases [6]. However, we ignore $\mathbf{u}_p$ in this paper for simplicity and take the element approximations as $\mathbf{u} = \mathbf{u}_0$.

Using approximations based on Eq. (8) ensures that the approximations are consistent for all values of the plate thickness, including the thin plate limit when $t = 0$. Hence, locking is impossible for the elements discussed in this paper, and no special remedies, such as reduced integration or selective shear interpolations [2], are required with these elements. This represents a significant advantage of the elements.

The approximations for $\mathbf{u} = \mathbf{u}_0$ are generated as follows [6, 7]. Firstly, the terms for w are taken as solutions of Eq. (6) with $p = 0$. Secondly, using these solutions for w, the terms for θ are taken as

$$\theta_x = \frac{\partial w}{\partial x} + \frac{1}{ktG_{xz}} \left(D_x \frac{\partial^3 w}{\partial x^3} + H \frac{\partial^3 w}{\partial x \partial y^2} \right) \tag{9a}$$

$$\theta_y = \frac{\partial w}{\partial y} + \frac{1}{ktG_{yz}} \left(D_y \frac{\partial^3 w}{\partial y^3} + H \frac{\partial^3 w}{\partial x^2 \partial y} \right) \tag{9b}$$

$$\text{or } \theta = \mathbf{D}_4 w \tag{9c}$$

As the plate thickness goes to zero while keeping the bending rigidities finite, the rotation approximations reduce to the thin plate approximations

$$\theta_x = \frac{\partial w}{\partial x}, \quad \theta_y = \frac{\partial w}{\partial y} \tag{10}$$

Eqs. (9) and (10) confirm that the approximations are consistent for both thick and thin plates, and locking cannot occur in the thin plate limit. In addition, since Eq. (6) has an infinite number of solutions, finite elements of any approximation order can be derived from these approximations [6, 7].

Finite Element Formulations

A conforming finite element solution for Mindlin's theory requires that the element variables, $\mathbf{u}$, are C_0 continuous between elements. In general, elements that use approximations based on Eq. (8) will not be C_0 continuous, as the continuity of $\mathbf{u}$ across element boundaries will not be possible to achieve. Hence, a standard stiffness formulation [2] will generally lead to unacceptable elements, and alternative formulations must be used. One option is to use the hybrid-Trefftz formulation, and this leads to acceptable and robust elements [6, 7]. However, we will explore two other formulations in this paper, namely the free formulation and the deviatoric strain formulation [3–5].

A key feature of these formulations is that the element stiffness matrix, $\mathbf{K}$, is taken as

$$\mathbf{K} = \mathbf{K}_{\text{basic}} + S\mathbf{K}_{\text{high}} \tag{11}$$

where S is a positive scale factor. Under certain conditions to be discussed below, an element derived from Eq. (11) converges even if the approximations are non-conforming. The scale factor can be adjusted to improve the performance of the element, and it is also possible to use multiple scale factors [4].

The basic stiffness matrix is the same for both formulations, while $\mathbf{K}_{\text{high}}$ is different for the two formulations. The basic stiffness matrix is given by

$$\mathbf{K}_{\text{basic}} = \frac{1}{A}\mathbf{L}\mathbf{E}_b\mathbf{L}^T \tag{12}$$

where $\mathbf{L}$ is the linear force lumping matrix and A is the area of the element. $\mathbf{L}$ is based on the boundary interpolations for $\mathbf{u}$, and it depends on the external nodal configuration of the element and the material bending properties $\mathbf{E}_b$. Expressions for $\mathbf{L}$ for various elements are given in [4].

Free formulation

In the free formulation [3], the element approximation is taken as

$$\mathbf{u} = \mathbf{P}_{rc}\mathbf{q}_{rc} + \mathbf{P}_h\mathbf{q}_h = \mathbf{P}\mathbf{q} \tag{13a}$$
$$\epsilon = \mathbf{D}\mathbf{u} = \mathbf{D}\mathbf{P}\mathbf{q} = \mathbf{B}_q\mathbf{q} \tag{13b}$$

where

$$\epsilon = \left\{ \begin{array}{c} \epsilon_b \\ \epsilon_s \end{array} \right\}, \quad \mathbf{D} = \begin{bmatrix} \mathbf{D}_1 & \mathbf{0} \\ -\mathbf{I} & \mathbf{D}_2 \end{bmatrix} \tag{14}$$

In Eq. (13a), the terms in $\mathbf{P}$ are approximation functions and $\mathbf{q}$ are approximation coefficients. The approximations in $\mathbf{P}$ have been split into six rigid body and constant strain modes (denoted by rc) and a specified number of higher-order modes (denoted by h). This split is essential for the free formulation.

For the current problem, the terms in $\mathbf{P}_{rc}$ are given by

$$\mathbf{P}_{rc} = \begin{bmatrix} \mathbf{P}_{\theta rc} \\ \mathbf{P}_{wrc} \end{bmatrix} \tag{15}$$

where

$$\mathbf{P}_{wrc} = \begin{bmatrix} 1, & x, & y, & x^2, & xy, & y^2 \end{bmatrix} \tag{16}$$

and

$$\mathbf{P}_{\theta rc} = \mathbf{D}_4\mathbf{P}_{wrc} \tag{17}$$

follows from Eq. (9c).

The terms in Eq. (16) incorporate constant bending strains and zero shear strains, and they satisfy the governing differential equations of Mindlin's theory (eqs. (1) to (5)) with $p = 0$. The terms in $\mathbf{P}_h$ are typically chosen as higher-order polynomials, although other forms can also be used [3]. It is important to note that there is no requirement in the free formulation for the terms in $\mathbf{P}_h$ to satisfy the governing equations of the problem, and generally they do not. In contrast, all the approximations used in this paper satisfy the governing equations.

The element nodal degrees of freedom, $\mathbf{v}$, are related to $\mathbf{q}$ by substituting the appropriate nodal coordinates of the element into Eq. (13a). This gives

$$\mathbf{v} = \begin{bmatrix} \mathbf{G}_{rc} & \mathbf{G}_h \end{bmatrix} \left\{ \begin{array}{c} \mathbf{q}_{rc} \\ \mathbf{q}_h \end{array} \right\} = \mathbf{G}\mathbf{q} \tag{18}$$

Inverting Eq. (18) gives

$$\mathbf{q} = \begin{bmatrix} \mathbf{H}_{rc} \\ \mathbf{H}_h \end{bmatrix} \mathbf{v} = \mathbf{Hv} \tag{19}$$

Hence, the element approximations can be written in terms of the nodal parameters as

$$\mathbf{u} = \mathbf{Pq} = \mathbf{PHv} = \mathbf{Nv} \tag{20a}$$

$$\epsilon = \mathbf{Du} = \mathbf{DNv} = \mathbf{B}_v \mathbf{v} \tag{20b}$$

For a displacement-based finite element formulation [2], the element stiffness matrix in terms of the approximation variables, $\mathbf{q}$, follows from Eq. (13b) and is given by

$$\mathbf{k}_q = \int_A \mathbf{B}_q^T \mathbf{E} \mathbf{B}_q \, dA = \begin{bmatrix} \mathbf{k}_{qrc} & \mathbf{k}_{qrch} \\ \mathbf{k}_{qrch}^T & \mathbf{k}_{qh} \end{bmatrix} \tag{21}$$

where

$$\mathbf{E} = \begin{bmatrix} \mathbf{E}_b & \mathbf{0} \\ \mathbf{0} & \mathbf{E}_s \end{bmatrix} \tag{22}$$

Combining Eqs. (19) and (21) gives the element stiffness matrix in terms of the nodal variables $\mathbf{v}$, as

$$\mathbf{K} = \mathbf{H}^T \mathbf{k}_q \mathbf{H} \tag{23}$$

As noted previously, a thick plate element based on Eqs. (21) and (23) is generally only convergent if the element approximations are C_0 continuous.

The key feature of the free formulation is to modify the element stiffness matrix so that nonconforming approximations can be used, while ensuring a convergent solution. Hence, the element stiffness matrix for the free formulation is taken as Eq. (11), with $\mathbf{K}_{\text{high}}$ given by [3]

$$\mathbf{K}_{\text{high}} = \mathbf{H}_h^T \mathbf{k}_{qh} \mathbf{H}_h \tag{24}$$

Several elements for the analysis of isotropic thin plates were derived in [3] using this formulation. An approximate extension to isotropic thick plates was made in [8]. This was done by taking the isotropic plate equivalents of Eqs. (9a) and (9b) as constraints and thereby eliminating the rotations as fundamental variables. It was shown in [9] that these constraints lead to an alternative eighth-order isotropic thick plate theory in terms of the transverse displacement as the only independent variable. A conforming finite element formulation of this theory requires higher-order continuity [10]. In contrast, in this paper we use Eq. (9) for deriving element approximations and do not eliminate the rotations as fundamental variables. This approach avoids the requirement of higher-order continuity.

Deviatoric strain formulation

The use of substitute strain fields in the finite element method has a long history, and several versions of the method have been used [2]. It has been shown that some of these methods are equivalent to mixed or reduced integration methods [11]. One particular version of these

methods is the deviatoric strain formulation, which was used in [5] to derive some successful isotropic thin plate elements. This particular formulation uses independent strains that are defined in terms of the natural coordinates of the element in combination with the fundamental stiffness matrix split from Eq. (11). It should be noted that the strains in this formulation are not necessarily derived from a displacement field. The authors of [5] called the formulation the assumed natural deviatoric strain formulation or ANDES formulation.

In this paper, we use a modified version of the ANDES formulation that starts with a displacement field

$$\mathbf{u} = \mathbf{Pq} \tag{25}$$

Unlike the free formulation, it is not necessary to split the displacement field into rigid body and constant strain modes, and higher-order modes. In contrast to [5], the strains are not independently interpolated but are instead derived from Eqs. (1) and (25), giving

$$\epsilon = \mathbf{Du} = \mathbf{DPq} = \mathbf{B}_q \mathbf{q} \tag{26}$$

Hence, the strains in this formulation are the derivatives of the displacement field as is the case for the free formulation (see Eq. (13b)). The strain matrix, $\mathbf{B}_q$, in Eq. (26) is split into bending (denoted by qb) and shear (denoted by qs) components, that is,

$$\mathbf{B}_q = \begin{bmatrix} \mathbf{B}_{qb} \\ \mathbf{B}_{qs} \end{bmatrix} \tag{27}$$

The mean bending strain matrix is calculated as

$$\mathbf{B}_{qbmean} = \frac{1}{A} \int_A \mathbf{B}_{qb} \, dA \tag{28}$$

The deviatoric bending strain matrix is defined as

$$\mathbf{B}_{qbdev} = \mathbf{B}_{qb} - \mathbf{B}_{qbmean} \tag{29}$$

and the total deviatoric strain matrix is defined as

$$\mathbf{B}_{qdev} = \begin{bmatrix} \mathbf{B}_{qbdev} \\ \mathbf{B}_{qs} \end{bmatrix} \tag{30}$$

It should be noted that the shear strain matrix, $\mathbf{B}_{qs}$, is not modified in this formulation.

With these definitions, the higher-order stiffness matrix in terms of the approximation variables, $\mathbf{q}$, is given by

$$\mathbf{k}_{qdev} = \int_A \mathbf{B}_{qdev}^T \mathbf{E} \mathbf{B}_{qdev} \, dA \tag{31}$$

Transforming to the nodal variables, $\mathbf{v}$, by using Eq. (19) gives the higher-order stiffness matrix as

$$\mathbf{K}_{\text{high}} = \mathbf{H}^T \mathbf{k}_{qdev} \mathbf{H} \tag{32}$$

The total stiffness matrix is again given by Eq. (11), where the scale factor, S, is not necessarily the same as for the free formulation.

Rectangular and Triangular Finite Elements

In this section, we use the two formulations to derive several finite elements that are suitable for the analysis of thick orthotropic plates. Both formulations are applicable to elements of arbitrary geometry and approximation orders. However, we only consider low-order rectangular and triangular elements in this paper for simplicity.

Rectangular elements

Fig. 2 shows the geometry and degrees of freedom for the rectangular elements. The elements have three degrees of freedom at each node, making a total of twelve degrees of freedom.
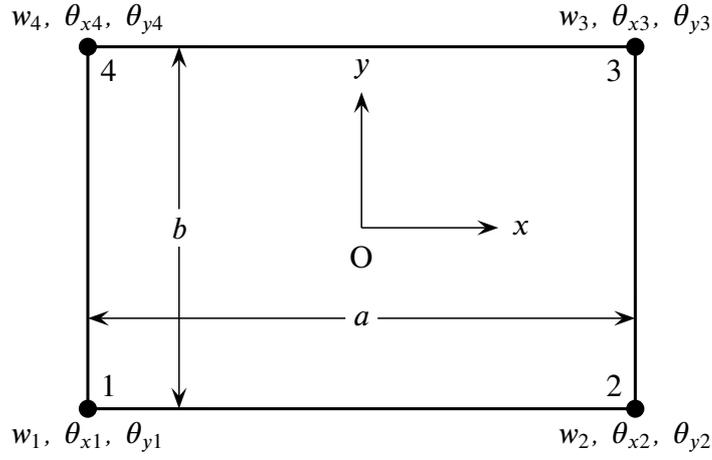
Figure 2. Rectangular element (origin is at centre of element).

The terms in $\mathbf{P}_h$ are taken as

$$\mathbf{P}_h = \begin{bmatrix} \mathbf{P}_{\theta h} \\ \mathbf{P}_{wh} \end{bmatrix} \tag{33}$$

where

$$\mathbf{P}_{wh} = \begin{bmatrix} x^3, & x^2y, & xy^2, & y^3, & x^3y, & xy^3 \end{bmatrix}, \quad \mathbf{P}_{\theta h} = \mathbf{D}_4\mathbf{P}_{wh} \tag{34}$$

Using Eqs. (13) and (33), the complete polynomial approximations for the displacement and rotations are

$$\mathbf{P}_w = [\mathbf{P}_{wrc}, \ \mathbf{P}_{wh}], \quad \mathbf{P}_\theta = \mathbf{D}_4\mathbf{P}_w \tag{35}$$

The terms in $\mathbf{P}_w$ are the displacement approximations for the well-known ACM thin plate element [2].

Two rectangular elements can now be derived using these approximations and either the free formulation or the deviatoric strain formulation. The resulting elements are denoted by the nomenclature RF12 (free formulation) and RD12 (deviatoric strain formulation), respectively. For isotropic thin plates, element RF12 reduces to element R3P from [12].

Triangular elements

Fig. 3 shows the geometry and degrees of freedom for the triangular elements. The elements again have three degrees of freedom at each node, making a total of nine degrees of freedom.

$x_3,\ y_3,\ w_3,\ \theta_{x3},\ \theta_{y3}$

$x_2,\ y_2,\ w_2,\ \theta_{x2},\ \theta_{y2}$

$x_1,\ y_1,\ w_1,\ \theta_{x1},\ \theta_{y1}$
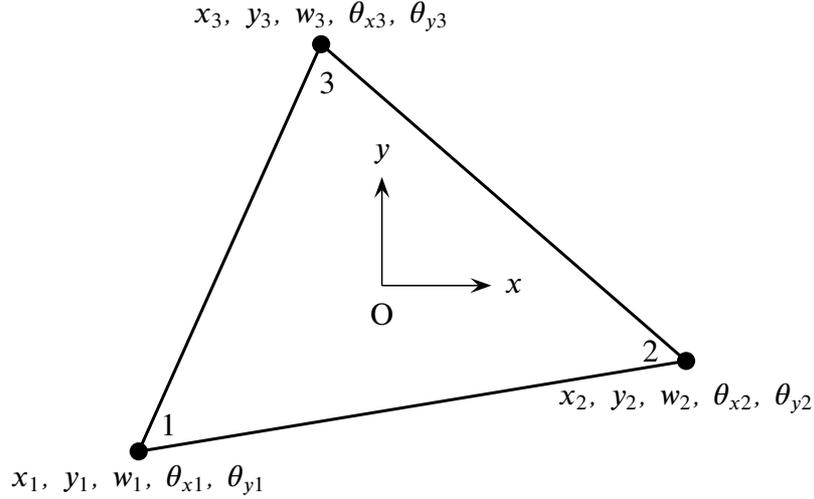
Figure 3. Triangular element (origin is at centre of element).

It is natural and convenient to use area coordinates [2] when working with triangular elements. Hence, for the triangular elements we use

$$\mathbf{P}_{wrc} = [\xi_1,\ \xi_2,\ \xi_3,\ \xi_1\xi_2,\ \xi_2\xi_3,\ \xi_3\xi_1]\,, \quad \mathbf{P}_{\theta rc} = \mathbf{D}_4\mathbf{P}_{wrc} \tag{36}$$

where ξ_1, ξ_2 and ξ_3 are the area coordinates.

Several options are possible for the higher-order modes, and we use

$$\mathbf{P}_{wh} = \left[\xi_1\xi_2^2,\ \xi_2\xi_3^2,\ \xi_3\xi_1^2\right]\,, \quad \mathbf{P}_{\theta h} = \mathbf{D}_4\mathbf{P}_{wh} \tag{37}$$

Other area functions, such as those used in [3] for isotropic thin plate elements, can also be used with the two formulations. The resulting elements have similar behaviour to those that use Eqs. (36) and (37). Following the previous nomenclature, the resulting triangular elements are denoted by TF9 and TD9.

Examples

We consider two examples to illustrate the behaviour of the proposed elements. The material properties were taken as $E_x = 10E_y$, $\nu_{xy} = 0.25$, $G_{xy} = 0.5E_y$, $G_{xz} = 0.5E_y$, $G_{yz} = 0.2E_y$ and $k = 5/6$ [1], and the scale factor was taken as $S = 1$ for all elements. The distributed load on the plate was modelled using a consistent load vector [2]. The boundary conditions are described as follows:

1. Simple support (S): $w = \theta_t = M_{nn} = 0$

2. Clamped support (C): $w = \theta_t = \theta_n = 0$

where n and t are the normal and tangential directions, respectively, at the boundary.

Fig. 4 shows a square plate size $a \times a$ and thickness t that is subjected to a uniform load p. All sides of the plate are either simply supported or clamped. Taking symmetry into account, a quarter of the plate was analysed using a uniform mesh of $N \times N$ rectangular elements or $2N \times 2N$ triangular elements.

As a first example, we consider the simply supported case. Results for the central displacement and bending moment for a thick plate ($a/t = 10$) and a thin plate ($a/t = 100$) are compared

Figure 4. Square plate of size $a \times a$ with typical rectangular and triangular meshes for quarter section ($N = 2$).

with the exact solutions in Tables 1 and 2. The displacement and bending moment converge rapidly for both thick and thin plates, with $N \geq 8$ being an acceptable mesh for the problem.

Table 1. Central displacement, w, for simply supported plate subjected to uniform load (multiplier $= 10^{-2} pa^4 / E_y h^3$).

N	$a/t = 10$				$a/t = 100$			
	RF12	RD12	TF9	TD9	RF12	RD12	TF9	TD9
2	1.832	1.842	1.402	1.631	1.560	1.561	1.239	1.385
4	1.712	1.718	1.586	1.667	1.451	1.451	1.376	1.408
8	1.684	1.688	1.649	1.675	1.424	1.424	1.405	1.413
16	1.677	1.681	1.668	1.677	1.417	1.417	1.412	1.414
32	1.675	1.679	1.674	1.677	1.415	1.415	1.413	1.414
Exact	1.679	1.679	1.679	1.679	1.415	1.415	1.415	1.415

Table 2. Central bending moment, M_x, for simply supported plate subjected to uniform load (multiplier $= 10^{-1} pa^2$).

N	$a/t = 10$				$a/t = 100$			
	RF12	RD12	TF9	TD9	RF12	RD12	TF9	TD9
2	1.289	1.298	1.044	1.217	1.344	1.344	1.023	1.194
4	1.170	1.173	1.130	1.161	1.199	1.199	1.122	1.163
8	1.142	1.144	1.144	1.145	1.166	1.166	1.149	1.157
16	1.136	1.137	1.142	1.139	1.158	1.158	1.155	1.156
32	1.134	1.136	1.140	1.138	1.156	1.156	1.155	1.155
Exact	1.135	1.135	1.135	1.135	1.155	1.155	1.155	1.155

The results for element RD12 are converging to the results from Mindlin's theory. In the case of a thick plate with $a/t = 10$, the results for the free formulation elements and element TD9 are converging to values that are slightly below the exact values. This implies that there is a small approximation to the shear deformation as a/t decreases and shear deformation increases. This observation is consistent with the results in [9, 10], where the results were also converging to values slightly different to the values from Mindlin's theory. However, it should be noted that Mindlin's theory is itself only an approximation to the true three-dimensional stress state in a plate, and other thick plate theories give slightly different results for the same problem [1]. For $a/t = 10$, the maximum difference is less than 0.3% for the quoted results. Hence, the differences can be ignored in a practical problem. In contrast, element RD12 is converging to the results from Mindlin's theory.

As the second example, we consider the clamped plate case. Results for the central displacement and bending moment for a thick plate ($a/t = 10$) and a thin plate ($a/t = 100$) are compared with the results obtained using the triangular thick plate element from [7] in Tables 3 and 4.
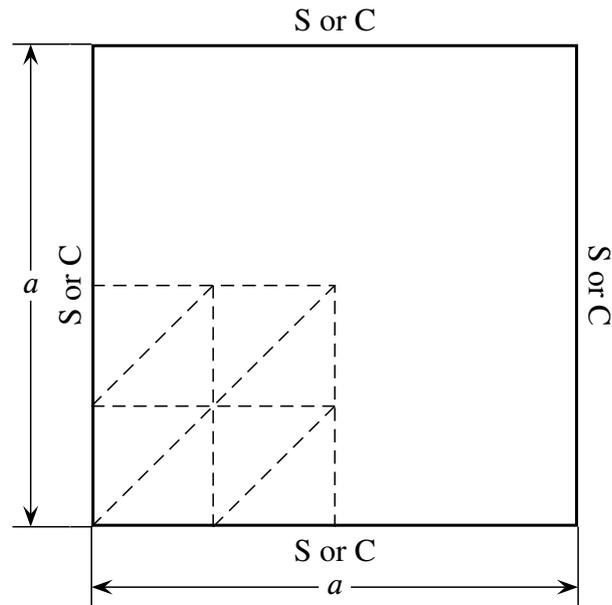
Table 3. Central displacement, w, for clamped plate subjected to uniform load (multiplier = $10^{-3}pa^4/E_yh^3$).

N	$a/t = 10$				$a/t = 100$			
	RF12	RD12	TF9	TD9	RF12	RD12	TF9	TD9
2	6.494	6.594	5.238	6.348	3.590	3.591	2.977	3.755
4	6.098	6.153	5.751	6.082	3.295	3.295	3.190	3.330
8	5.997	6.043	5.884	6.003	3.220	3.220	3.194	3.222
16	5.971	6.016	5.919	5.979	3.201	3.201	3.194	3.200
32	5.965	6.009	5.928	5.972	3.197	3.197	3.194	3.196
Ref. [7]	6.007	6.007	6.007	6.007	3.195	3.195	3.195	3.195

Table 4. Central bending moment, M_x, for clamped plate subjected to uniform load (multiplier = $10^{-2}pa^2$).

N	$a/t = 10$				$a/t = 100$			
	RF12	RD12	TF9	TD9	RF12	RD12	TF9	TD9
2	4.941	5.039	4.765	5.570	5.483	5.489	4.087	5.610
4	4.285	4.314	4.363	4.405	4.561	4.561	4.197	4.472
8	4.124	4.142	4.233	4.176	4.345	4.345	4.265	4.313
16	4.084	4.100	4.161	4.120	4.293	4.293	4.280	4.285
32	4.074	4.090	4.134	4.105	4.280	4.280	4.281	4.279
Ref. [7]	4.086	4.086	4.086	4.086	4.276	4.276	4.276	4.276

Once again, the displacement and bending moment converge rapidly for both thick and thin plates, with $N \geq 8$ being an acceptable mesh for the problem. Overall, the behaviour of the elements for this problem is similar to the simply supported case.

Conclusions

This paper has discussed the use of the free formulation and the deviatoric strain formulation for the development of finite elements for orthotropic thick and thin plate structures. The element interpolations are the solutions of the governing equations of Mindlin's theory, and locking cannot occur in the thin plate limit with these elements. Both formulations lead to reliable and accurate elements. Two examples were considered in the paper to demonstrate the accuracy that is achieved with the proposed elements.

References

[1] Reddy, J. (2007) *Theory and Analysis of Elastic Plates*, second edn., Taylor and Francis, Philadelphia.

[2] Zienkiewicz, O., Taylor, R. and Fox, D. (2014) *The Finite Element Method for Solid and Structural Mechanics*, vol. 2, seventh edn., Butterworth-Heinemann, Oxford.

[3] Bergan, P. and Nygård, M. (1984) Finite elements with increased freedom in choosing shape functions, *International Journal for Numerical Methods in Engineering*, **20**, 643–663.

[4] Felippa, C. and Bergan, P. (1987) A triangular bending element based on an energy-orthogonal free formulation, *Computer Methods in Applied Mechanics and Engineering*, **61**, 129–160.

[5] Militello, C. and Felippa, C. (1991) The first ANDES elements: 9-dof plate bending triangles, *Computer Methods in Applied Mechanics and Engineering*, **93**, 217–246.

[6] Petrolito, J. (2003) Analytical formulation of hybrid-Trefftz thick plate elements, *Computer Assisted Mechanics and Engineering Sciences*, **10**, 375–385.

[7] Petrolito, J. (2014) Vibration and stability analysis of thick orthotropic plates using hybrid-Trefftz elements, *Applied Mathematical Modelling*, **38**, 5858–5869.

[8] Bergan, P. and Wang, X. (1984) Quadrilateral plate bending elements with shear deformations, *Computers and Structures*, **19**, 25–34.

[9] Abdalla, H. and Hassan, K. (1988) On the Bergan-Wang approach for moderately thick plates, *Communications in Applied Numerical Methods*, **4**, 51–58.

[10] Hassan, K., Ali, E. and Tawfik, M. (2020) Finite elements for the one variable version of Mindlin-Reissner plate, *Latin American Journal of Solids and Structures*, **17**, 10.1590/1679–78256170.

[11] Hughes, T. (1987) *The Finite Element Method*, Prentice Hall, New Jersey.

[12] Bergan, P., Nygård, M. and Wang, X. (1981) A class of quadrilateral plate bending elements, *Proceedings, International Conference on Finite Element Methods*, 112–117, Shanghai.

GPU Parallelization of solving pressure Poisson Equation in MPS Method

†Zhe Sun[1,3], *Zi-kai Xu[1], Xi Zhang[2], Bi-ye Yang[1],†Gui-yong Zhang[1,3,4], Zhi-fan Zhang[1]

[1]School of Naval Architecture and Ocean Engineering, Dalian University of Technology, Dalian, 116024, P. R. China
[2]National Supercomputing Center , Guangzhou, 511400, P.R.China
[3]State Key Laboratory of Structural Analysis for Industrial Equipment, Dalian, 116024, PR China
[4]Collaborative Innovation Center for Advanced Ship and Deep-Sea Exploration, Shanghai, 200240, P.R. China

*Presenting author: a417851468@dlut.edu.cn
†Corresponding author: zsun@dlut.edu.cn   gyzhang@dlut.edu.cn

Abstract

In this paper, an improved form of explicit solution of pressure Poisson equation is introduced. On the basis of the existing program, the numerical simulation of 2D dam-break problem is carried out. The numerical results agree well with experiments and GMRES method,computing efficiency is greatly improved; At the same time, the MPS method is combined with the GPU parallel acceleration technique. Based on the CUDA programming language, the GPU is parallelized to solve the pressure Poisson equation. Compared with the CPU solver, the developed program greatly reduces the solution time of the pressure Poisson equation and improves the calculation efficiency. The maximum acceleration ratios of 11.486 can be obtained by numerical simulation for 2-D dam-break problems with different particle numbers. And the developed program has better reliability and good adaptability.

Keywords: Meshless particle method, MPS method, CUDA, parallel computing

1. Introduction

The meshless particle method has been developed rapidly in recent years. It has shown great advantages in dealing with large deformation of free surface and phase interface or strong nonlinear deformation and large motion of boundary. The basic idea of the method is to disperse the computational domain into a series of fluid particles with flow field information, and the transfer of flow field information is expressed by the interaction of particles. Smoothed Particle Hydrodynamics (SPH) and Moving Particle Semi-implicit (MPS) are two typical meshless particle class methods, of which the MPS method has attracted much attention as a new computational method. In recent years, some scholars have used MPS method to simulate problems such as dam break, slogging, fluid-structure coupling, and floating body motion on waves, and obtained good results, indicating that MPS method has great applicability.

However, there are still some problems in the MPS method, one of which is the computational efficiency. Because the MPS method is based on the incompressible condition, the pressure is obtained by solving the pressure Poisson equation, which needs to solve a sparse matrix. The

dimension of the matrix is positively correlated with the particle numbers. Therefore, how to improve computational efficiency has been a problem worthy of attention.

In recent years, the appearance of a parallel technology called GPU (Graphic Processing Unit) has aroused many people's attention. With the rapid development of GPU hardware and the development of related programming technology, GPU begins to play an important role in some general computing fields because of its powerful floating point computing ability and high efficiency. Because of the difference between GPU and CPU in design goal, there is a great difference between them in logic structure of hardware. As shown in figure 1 [1], GPUs are divided into more execution units (ALUs) and have more memory bandwidth than CPUs, which have more control and cache units. This difference, fundamentally, determines that CPU has more complex computing power, while GPU has more floating point computing power and parallel computing power. This natural multi-core architecture mode makes GPU suitable for large-scale parallel scientific computing.

Figure 1. Different hardware architectures for GPU and CPU

The numerical model of meshless particle method is relatively simple. Except for Poisson equation, the calculation of each particle is relatively independent, and it is easy to combine the whole method with GPU. The application of GPU parallel acceleration technology in SPH method has been relatively mature. Harada et al. [2] and Zhang et al. [3] first applied GPU technology to meshless particle method SPH. Then the application of GPU in SPH gradually increased. Crespo et al. [4] reviewed the theory of SPH and the different processing methods of some key problems, and compared the similarities and differences of SPH method on CPU and GPU. Herault et al. [5] used CUDA library to accelerate the SPH method, and compared the acceleration ratio ratios of neighbor particle search, integral calculation and particle movement through the dam-break problem in detail, which showed that the GPU parallel acceleration ratio technology has great potential in the SPH method. Wei et al. [6]used the GPUSPH program to simulate the impact of a tsunami on a vertical column. The effects of pier shape and upstream angle on free surface evolution and hydrodynamic load were studied. The DualSP Hysics solver using ISPH method, such as Chow [7], has carried out the research of 3D focusing wave slamming on vertical cylinder on a single GPU. However, compared with the explicit calculation process of SPH, the MPS method adopts semi-implicit calculation process, which brings difficulties to the combination of MPS method and GPU acceleration technology, so the related research work much less than SPH method. Hori et al.[8]developed the MPS program of GPU parallel acceleration by CUDA language. The

pressure Poisson equation and pressure gradient are solved by GPU parallel acceleration technique. Kakuda et al.[9]applied the MPS method based on GPU acceleration to the 3D simulation problem, and the acceleration ratio obtained by the improved MPS method was 17.33 when the number of particles was 200,000. Gou et al.[10]implemented the parallel optimization of MPS method on GPU, simulated the interaction of isothermal and multiphase fuel coolant, and the numerical results were in agreement with the experimental results, and achieved a higher acceleration ratio. Vieira-e-Silva et al.[11] used the improved MPS method to simulate the dam-break flow on GPU, but the number of particles in the numerical model was less. Taniguchi et al.[12] have developed a WCMPS algorithm program for multiple GPUs, and tested the reliability of the program through a standard 3D dam-break example. Kawamura et al.[13] and Hashimoto et al.[14] applied the GPU parallel technology to the WCMPS method, developed the GPGPU program, and simulated the 3D sloshing of oil storage tanks under earthquake excitation with 6 million particles.

In this paper, an improved explicit solution form of pressure Poisson equation is presented and compared with the implicit solution form. Secondly, the computational efficiency of pressure Poisson equation is discussed by developing GPU program and CPU solver. The second part introduces the basic theory of MPS and the improved method of explicit solution of Poisson equation, and compares it with the result of implicit solution of Poisson equation, which verifies its feasibility. In the third part, based on the CUDA programming language and the existing 2-D dam-break program, a meshless particle method program is developed to solve the pressure Poisson equation on the GPU equipment,and the computational efficiency of solving the pressure Poisson equation is compared. The reliability and adaptability of the program are verified by numerical simulation of 2-D dam break problems with different particle numbers, and the acceleration ratio under different particle numbers is discussed. The fourth part is the conclusion.

2. Numerical method

2.1 Basics of MPS method

The MPS method is a meshless method based on the Lagrange method, and the computational domain is represented by discrete particles. The particles are not connected by grids or nodes, but each carries its own physical information, such as mass, velocity, and acceleration. The numerical model of MPS method used in this paper is introduced below.

2.1.1 Governing equation

For incompressible fluids, the MPS method uses the Navier-Stokes equation and the continuity equation as the governing equations. The forms are as follows:

$$\nabla \cdot \boldsymbol{u} = 0 \tag{1}$$

$$\rho \frac{D\boldsymbol{u}}{Dt} = -\nabla P + \mu \nabla^2 \boldsymbol{u} + \rho g \tag{2}$$

D/Dt denotes the derivative of matter, t is time, ρ is fluid density, P is pressure, μ is dynamic viscosity, *V* is velocity vector, and g is gravitational acceleration vector.

2.1.2 Particle interaction model

In the MPS method, the computational domain is composed of a series of discrete particles whose interactions are realized by kernel functions. The kernel functions used in this article are as follows:

$$
w(r_{ij}) = \begin{cases} \dfrac{r_e}{r_{ij}} - 1, 0 < r_{ij} \leq r_e \\ \\ 0, r_{ij} > r_e \end{cases} \tag{3}
$$

In the formula, $r_{ij} = |r_i - r_j|$ is the distance between particles I and J, r_e is the radius of the particle's support domain. r_{e_lap} and r_{e_grad} adopted in this paper are 4.0 l_0 and 2.1 l_0 respectively[15], where is the initial particle spacing of l_0.

2.1.3 Boundary conditions

In the MPS method, when solving the pressure Poisson equation, we usually assign the pressure of the free surface particle and the second kind of boundary particle to 0 as the boundary condition.

2.1.3.1   Pressure Neumann condition on solid boundaries

In this study, the innermost solid boundary is treated by Newman boundary condition, and the pressure gradient between the current boundary particle and the closest fluid particle is calculated to avoid the deficiency of the particle in the support region. The formula is as follows:

$$
\mathbf{n} \cdot \nabla p^{(k+1)} = \rho_0 (\mathbf{n} \cdot \mathbf{g} - \mathbf{n} \cdot \boldsymbol{u}_b^{(k+1)}) \tag{4}
$$

Where $\boldsymbol{u}_b$ is the acceleration of the boundary and $\mathbf{n}$ is the normal vector of the boundary.

2.1.3.2   Laplacian operator compensation near solid boundary

For fluid particles close to the solid boundary, Laplace operator needs to be modified to meet the Neumann condition on the solid boundary and make up for the shortcomings of the adjacent particles.

The formula is as follows:

$$p_v = p_s + \rho_0 (\mathbf{n} \cdot \mathbf{g} - \mathbf{n} \cdot \boldsymbol{u}_b^{(k+1)}) r_0 \qquad (5)$$

As shown in Figure 2[16], p_v is the pressure of the virtual particle and p_s is the pressure of the corresponding solid particle.
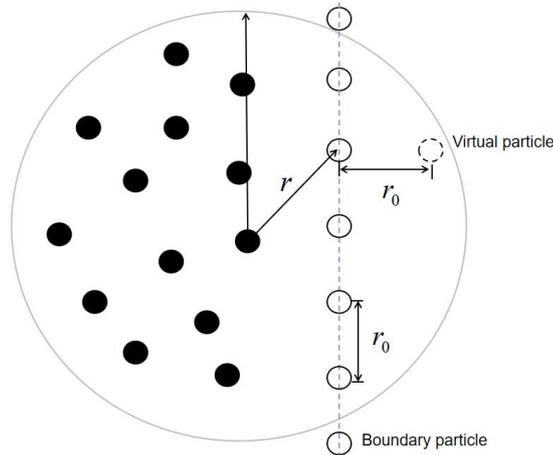
Fig 2. virtual particle for compensating the Lasplace operator near solid boundary
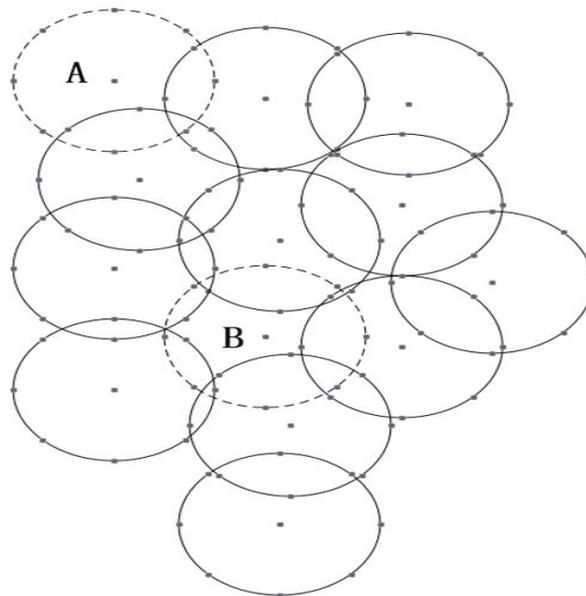
Fig 3. Demonstration of free surface particle identification

2.1.3.3 Free surface particle identification

A simplified version of the method used by Koh et al. [17]. Each self-centered particle is assigned a circle with a radius of $1.05r_0$, and the circle is discretized to 360 points and placed

evenly along the circle. If these points are completely covered by the circle of its neighbor particle, then it is considered an internal fluid particle, otherwise it is a free surface particle.

As shown in Figure 3, particle A is identified as a free-surface particle because it is not covered by a neighbor particle; accordingly, particle B is identified as an internal fluid particle because it is covered by a neighbor particle.

2.1.4 Basic flow of the MPS approach

In the MPS method, the control equation is solved by a predictor-corrector semi-implicit method. The general flow of a single time step is as follows:

(1) After completing the initial arrangement and entering the time step cycle, the first step is to correct the position and velocity of particles. The first correction is to calculate the intermediate velocity without considering the pressure and then move the particle to the intermediate position based on the increment of the velocity:

$$\mathbf{u}^{(*)} = \mathbf{u}^{(k)} + \Delta t g \tag{6}$$

$$\mathbf{r}^{(*)} = \mathbf{r}^{(k)} + \Delta t \mathbf{u}^{(*)} \tag{7}$$

(2) Calculate the particle numbers density n^{*}

Particle number density is the denseness of neighbouring particles around a particle, specifically refers to the accumulation of the nuclear function of the particle i and its neighbouring particles within the scope of the nuclear function. The formula is as follows:

$$\langle n \rangle_i = \sum_{j \neq i} W(|r_j - r_i|) \tag{8}$$

In order to provide the newest particle number density field in the next solution of pressure Poisson equation, we need to search the neighbor particle of each particle again, and calculate the middle step density of each fluid particle.

(3) Solving pressure Poisson equation
In the MPS method, the particle pressure is obtained by solving the pressure Poisson equation. The pressure Poisson equation used in this study is as follows[18][19][20]:

$$\nabla^2 p^{(k+1)} = \rho_0 \frac{\nabla \cdot u^{(*)}}{\Delta t} + \alpha \rho_0 \frac{n_0 - n^{(k)}}{n_0 \Delta t^2} \tag{9}$$

k+1 is the time step, and α is chosen by the following formula:

$$\alpha = \begin{cases} |\frac{n_0 - n^{(k)}}{n_0}| + \Delta t |\nabla \cdot u^{(k)}|, (n_0 - n^{(k)})\nabla \cdot u^{(k)} \geq 0 \\ |\frac{n_0 - n^{(k)}}{n_0}|, (n_0 - n^{(k)})\nabla \cdot u^{(k)} < 0 \end{cases} \tag{10}$$

The Laplacian model is used to discretize the second derivative terms in the governing equations. The Laplace operator is described as follows:

$$\left\langle \nabla^2 \phi \right\rangle_i = \frac{2d}{n_0 \lambda} \sum_{j \neq i} \left(\phi_j - \phi_i \right) w \left(\left\| \mathbf{r}_j - \mathbf{r}_i \right\| \right) \tag{11}$$

The λ in the formula is obtained by using the following formula:

$$\lambda = \frac{1}{n_0} \sum_{j \neq i}^{N} w(r_{ij}) r_{ij}^2 \tag{12}$$

(4)Pressure gradient
the following formulas are used to solve pressure gradients:

$$\left\langle \nabla p \right\rangle_i = \frac{d}{n_0} \sum_{j \neq i}^{N} \frac{p(r_j) - \tilde{p}(r_i)}{r_{ij}^2} (r_j - r_i) w(r_{ij}) \tag{13}$$

d is the dimension of the problem, n_0 is the initial particle number density, and $\tilde{p}(r_i)$ is the

minimum pressure among all particles in the support domain.
Similarly, the divergence model is used to discretize the divergence of velocity in the governing equation. The expression is:

$$\left\langle \nabla \cdot V \right\rangle_i = \frac{d}{n_0} \sum_{j \neq i} \frac{(V_j - V_i) \cdot (r_j - r_i)}{|r_j - r_i|^2} W(|r_j - r_i|) \tag{14}$$

After the pressure gradient of each particle is calculated, the position and velocity of the second particle can be corrected according to the following formula:

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(*)} - \Delta t \frac{\nabla p^{(k+1)}}{\rho_0} \tag{15}$$

$$\mathbf{r}^{(k+1)} = \mathbf{r}^{(k)} + \Delta t \mathbf{u}^{(k+1)} \tag{16}$$

2.2 Explicit solution of pressure Poisson equation

Because the MPS method adopts the semi-implicit method to solve the Poisson equation, we need to solve the linear equations in the process of solving, the calculation efficiency is not very high, so this paper adopts the explicit solution method to solve the pressure Poisson equation, that is, do not solve the equations, directly solve the pressure of particles through the formula iteration; and the MPS method adopts the implicit method to solve the Poisson equation brings difficulties for GPU parallelism, so we use the explicit solution form to speed up the process of GPU parallelism.

2.2.1 Relaxed Jacobi

As we all know, when we bring the Laplace model in this paper into the Poisson equation, it will form a linear equation system about pressure, and can be discretized into the form of linear equation system $Ax = b$. The original CPU solver is solved implicitly through the GMRES method. Here we use the Jacobian relaxation iteration method to solve the pressure Poisson equation, which is described below:

According to the Relaxed Jacobi method, we can iteratively solve for the particle pressure value:

$$p_i^{l+1} = (1-\omega)p_i^l + \omega \frac{b_i - \sum_{j\neq i} a_{ij} p_j^l}{a_{ii}} \qquad （17）$$

Where l is the number of iterations, ω is called the relaxation factor, b_i is the right-hand term of the pressure Poisson equation.

When the time step is over, we get the pressure of each particle p^{l+1}. When the next time step is done, we take the pressure of the last time step p^{l+1} as the initial pressure p^{l+1} of the current time step.

Iterations and relaxation factor are selected using the following formula：

$$R = \frac{\sum_{i=1}^{N}(Ap_i - b_i)}{N} \qquad (18)$$

In the formula, A is the coefficient matrix of the pressure Poisson equation, p is the pressure value solved by Jocabi method, N is the total number of particles involved in solving the pressure Poisson equation, i is the particle sequence number, and R is called the residual value. When R < 10, the selection of iteration times and relaxation coefficient is reasonable. In this paper, the selection of relaxation factor and iterations follow this formula,no further details are given below.

2.2.2   2-D dam-break simulation

In order to verify the feasibility of explicit solution of the pressure Poisson equation, in this paper, a 2-D dam-break example is used.

As shown in Figure 4, the liquid portion is 0.6m in length and height, the boundary portion is 1.61m in length and 0.6m in height, and a layer of solid particles is arranged outside to prevent particles from passing through. The particle spacing is 0.005m, the time step is dynamically arranged and the maximum is 0.001s. The total number of particles is 15530, among which liquid particles are 14400, internal solid particles are 563 and external solid

particles are 567. The acceleration of gravity g=9.81m/s, and the density of water is 1000kg/m³.After testing, we chose 0.5 relaxation factor and 150 iterations to get the best results.
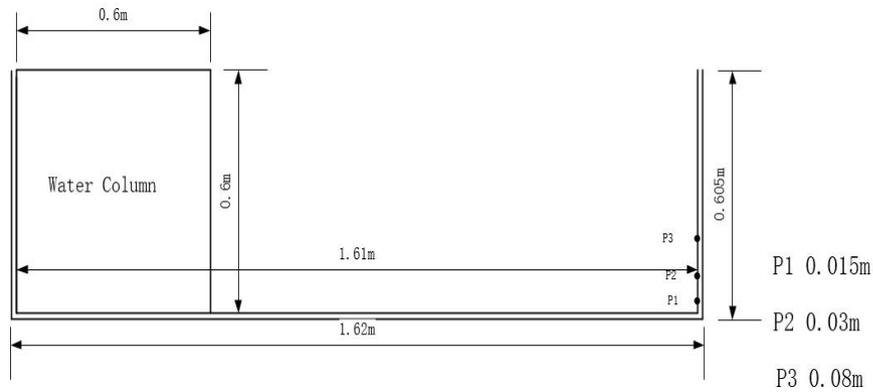
Fig 4. Sketch of 2-D dam-break
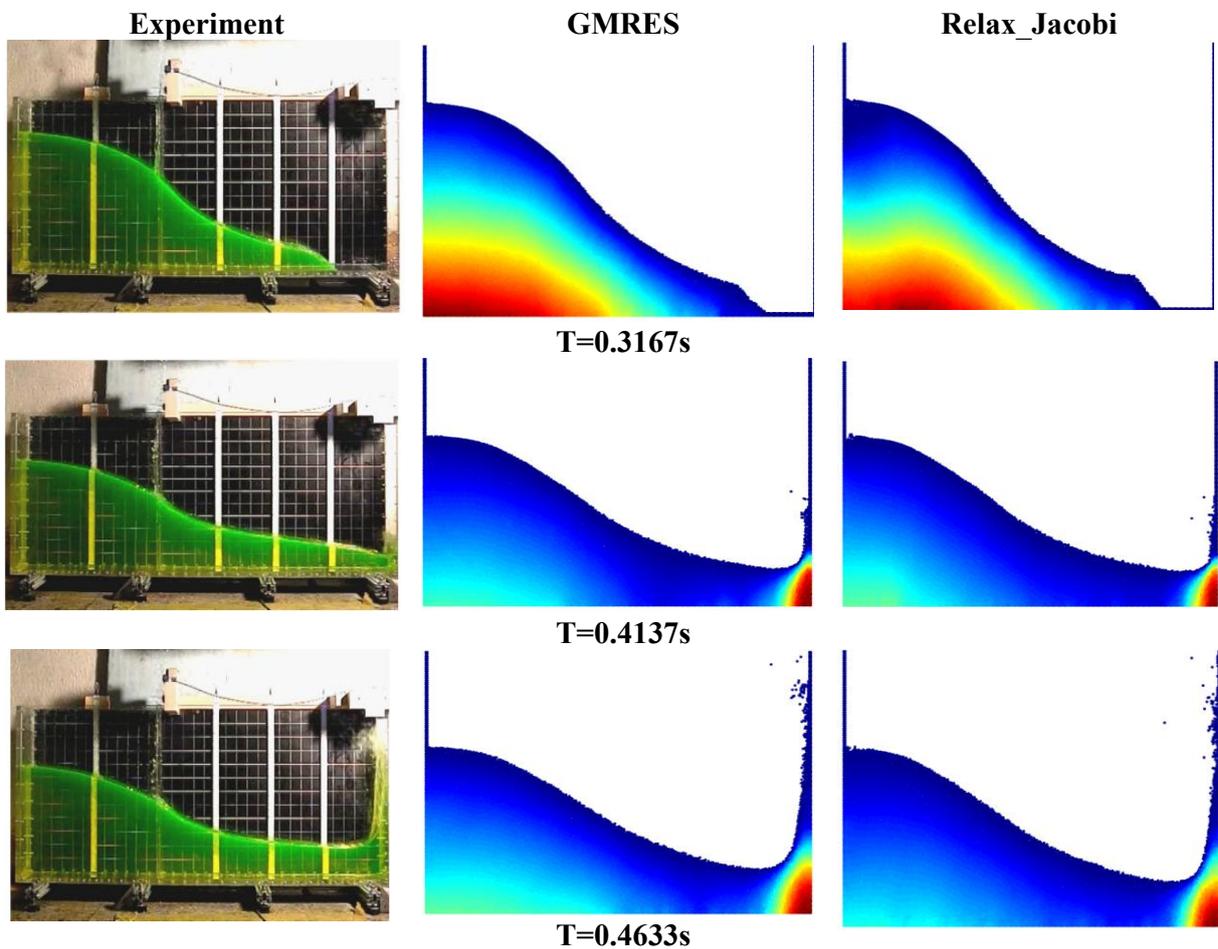
Fig 5. Comparison of numerical simulation Results at T = 0.3167s, T= 0.4137s and T = 0.4633s

In order to verify the accuracy of the explicit solution, we selected three pressure measurement points on the right side of the wall, and compared with the experimental results of Lobovsky et al.[21], as well as the pressure time curve, pressure cloud diagram and free

surface diagram of the implicit GMRES for solving the pressure Poisson equation, to prove the feasibility of the explicit solution of the pressure Poisson equation. The calculation results are as follows:
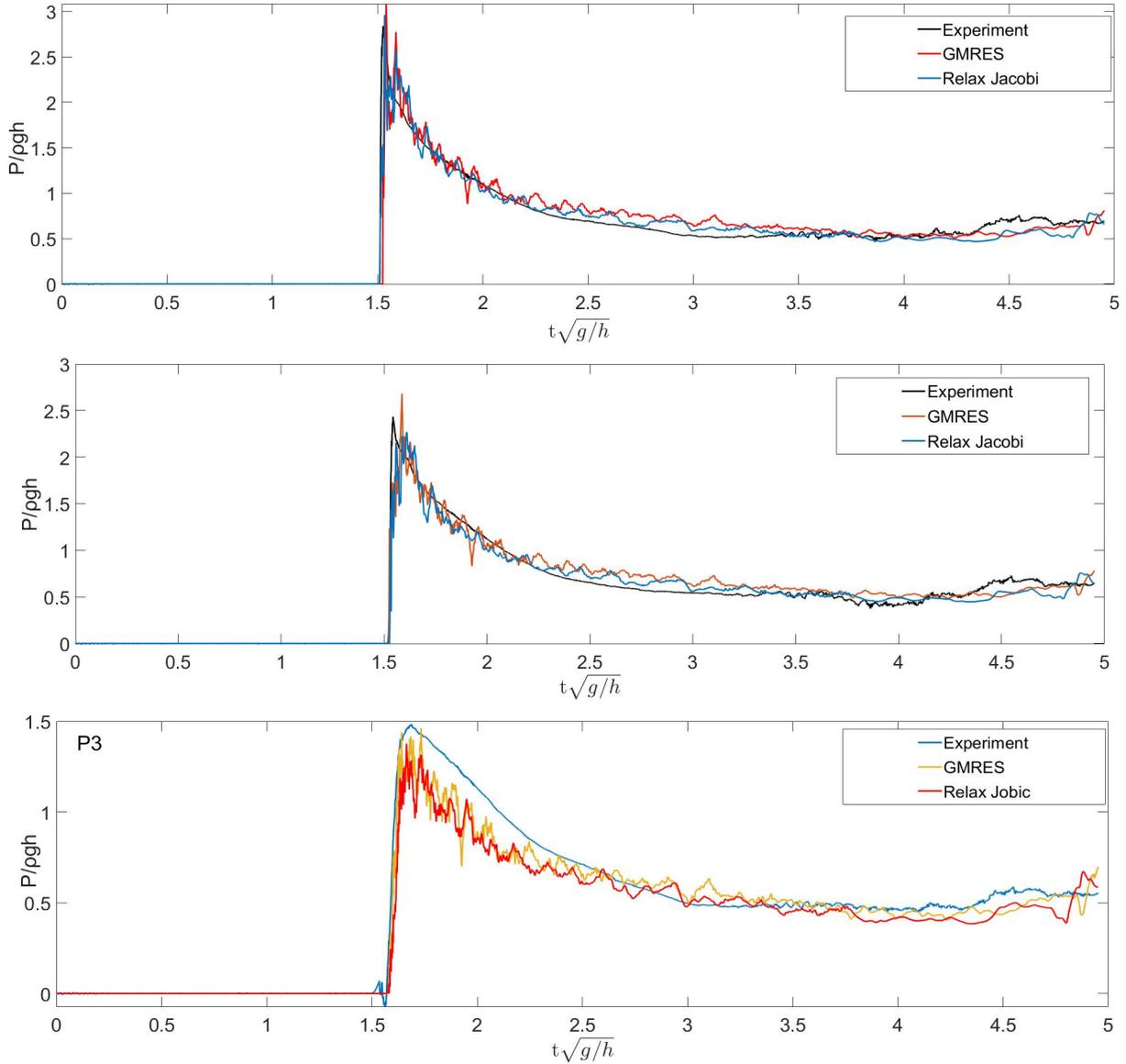
Fig 6. Comparison of pressure duration curves at P1, P2 and P3

As shown in Figure 5, according to Lobovsky et al. 's experiment, we selected three moments for analysis, t = 0.3167s, t = 0.4137s and t = 0.4633s, with a time error of no more than 5ms. According to the results, the explicit solution of pressure Poisson equation is consistent with the implicit solution and experiment, and the smoothness of pressure field is better.

As shown in Figure 6, the experimental, GMRES method, Relaxed Jacobi method under the P1, P2, P3 three measurement points under the pressure curve, horizontal and vertical coordinates are through dimensionless, respectively, $t\sqrt{(g/h)}$ and P/ρgh. After comparison, Relaxed Jacobi method is in good agreement with experiment and GMRES method.

In conclusion, the Relaxed Jacobi method is feasible to solve the pressure Poisson equation. Then, the time of solving the pressure Poisson equation by GMRES method and Relaxed Jacobi method is calculated. The results are shown in Figure 7:
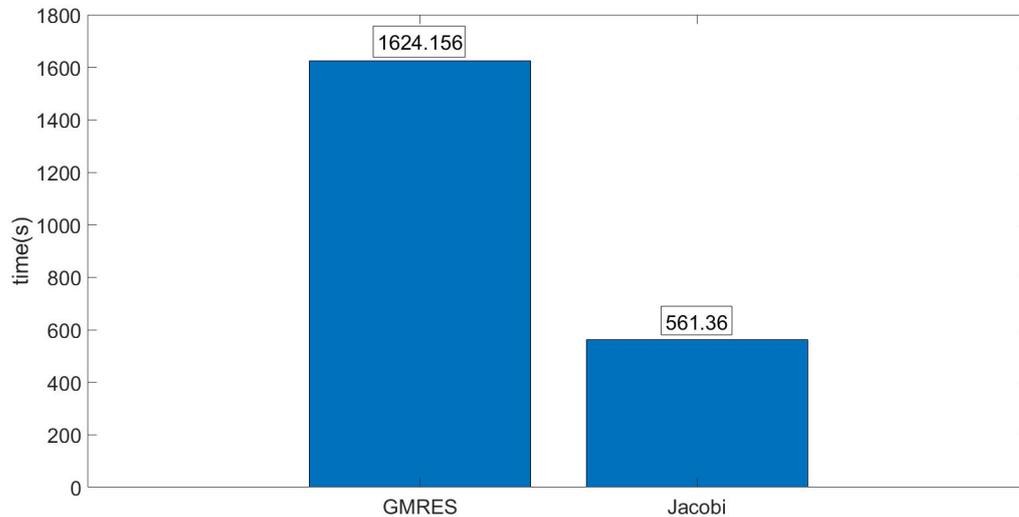
Figure 7. Solution time of pressure Poisson equation

Compared with the GMRES method, the Relaxed Jacobi method reduces the time of solving the pressure Poisson equation and improves the efficiency by 65.4%.

3. GPU parallelision for MPS

In 2.2, we verify the feasibility of explicitly solving Poisson's equations. In this section, we parallelize the explicit solution of Poisson's equations into GPU parallelization.

3.1 GPU parallel computing

3.1.1 CUDA architecture

CUDA (Compute Unified Device Architecture) is a hardware and software architecture released by NVIDIA to manipulate GPU computing. It is a general parallel computing platform and programming model based on NVIDIA's GPUs. It provides a simple interface for GPU programming. CUDA -based programming allows you to build GPU -based applications and use the GPUs parallel computing engine to solve more complex computing challenges more efficiently. It treats the GPU as a data-parallel computing device and does not need to map these calculations to a graphics API. The operating system's multitasking mechanism allows CUDA to access both the GPU and graphical runtime, and its computational features enable CUDA to visually write GPU core programs.

CUDA consists of a CUDA library, an application programming interface (API) and its runtime, and two high-level general-purpose mathematical libraries, CUFFT and CUBLAS. CUDA improves the read-write flexibility of DRAM so that the GPU fits the mechanism of

CPU. On the other hand, CUDA provides on-chip shared memory that allows threads to share data. Applications can utilize shared memory to reduce DRAM data transfers and rely less on DRAM memory bandwidth.

3.1.2 CUDA programming mode

The CUDA architecture introduces the concepts of host and device. CUDA programs contain both host and device programs. At the same time, host and device can communicate so that data can be copied between them. Among them, the memory of the CPU and the system (memory bar) is called the host, and the display memory of the GPU and the GPU itself is called the device.

A typical CUDA program executes as follows:
1. Allocating host memory and initializing data;
2. Allocating device memory and copying data from host to the device;
3. Call the kernel function of CUDA to complete the specified operation on the device;
4. Copy the operation result on the device to the host;
5. Free memory allocated on device and host.

3.2 GPU algorithm for solving pressure Poisson equation

3.2.1 Row compression method (CSR)

The pressure Poisson equation and the Laplace model are combined into a linear system. The coefficient matrix A of the Poisson equation is a typical sparse matrix. We cannot store all the elements in matrix A, otherwise the number of particles that can be simulated will be greatly reduced, which is not ideal for GPUs with smaller memory sizes. Therefore, this article will use the Compressed Sparse Row method to store sparse matrices. The matrix is represented by three one-dimensional arrays, A _ temp, JA _ temp, and IA _ temp. On the one hand, it can save storage space. On the other hand, three arrays are used for storage, which is easy to use. The array A _ temp stores all nonzero elements of the coefficient matrix; the JA _ temp array stores the column index of all nonzero elements of the coefficient matrix A; and the array IA _ temp stores the cumulative number of nonzero elements in row I, excluding row I. We found the number of non-zero elements on line I by looking at IA _ temp [I-1] - IA _ temp [I].
Using row compression (CSR) format to store matrix - related information facilitates the transfer of matrix information in the GPU parallel algorithm in this paper.

3.2.2 Introduction of GPU parallel algorithm for solving pressure Poisson equation

In Chapter 2, we introduce the explicit solution of the pressure Poisson equation by Relaxed Jacobi. In this section, we parallelize the pressure Poisson equation with GPU. The process is shown in Figure 8:
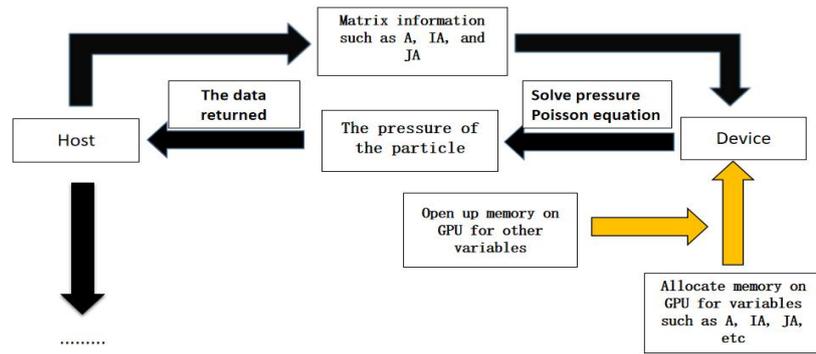
Fig 8. GPU parallel process for pressure Poisson equation

In the previous introduction, in the process of solving the pressure Poisson equation, we can get the linear equations about pressure by combining the Laplace model with the pressure Poisson equation, and then disperse them into the system of linear equations. Matrix information is already stored in A _ temp, JA _ temp, and IA _ temp through the CSR method, so we need to open memory for arrays A _ temp, JA _ temp, IA _ temp, and BM on the GPU and pass the Matrix information from the CPU to the GPU.

After the matrix information is transferred, we need to open up space in the GPU for the other variables in the Relaxed Jacobi method on the CPU solver to convert the host variable into the corresponding device variable.

To determine the accuracy of our calculations, we need to test the data transfer from the CPU to the GPU, and after each GPU invocation, we need to test the results, not verbose here.
After solving the pressure Poisson equation, we need to get the pressure back to facilitate the CPU solver to continue to calculate, and finally get the results.

3.3 2-D dam-break simulation

3.3.1 Computational verification

The numerical model still adopts the model in 2.2.2, The calculation configuration is as follows：

CPU	AMD Ryzen 9 3900X 12-Core Processor 3.97GHz 64GB
GPU	NVIDIA GeForce RTX 2060 6GB
compiler	CUDA 11.2 GCC 7.5.0

And other conditions remain unchanged, to verify the suitability of the GPU code, the CPU and GPU computations are as follows:

CPU   **CPU+GPU**
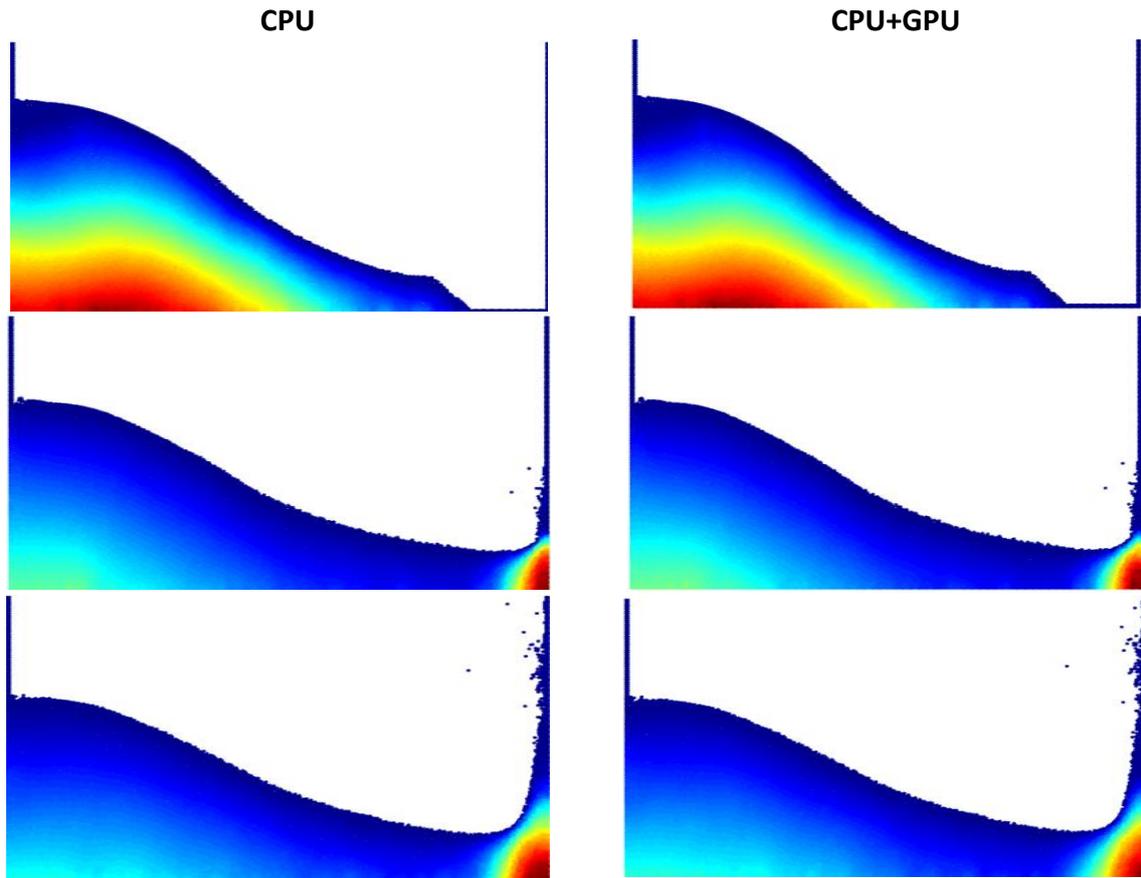
Fig 9. Comparison chart of calculation results at t = 0.3167s, t = 0.4137s and t = 0.4633s

As shown in Figure 9 , the results under CPU are basically consistent with those under GPU+ CPU, the outlines of flow field are basically the same, and the smoothness of pressure field is better.

In theory, the CPU and CPU+GPU calculations should be the same, but the data transmitted by the CPU and GPU, there will be data loss of the last digit, although the gap is small, but will still have an impact on the results of the calculation, so the pressure curve in Figure 140has a certain difference is normal, still can prove the adaptability of GPU code.
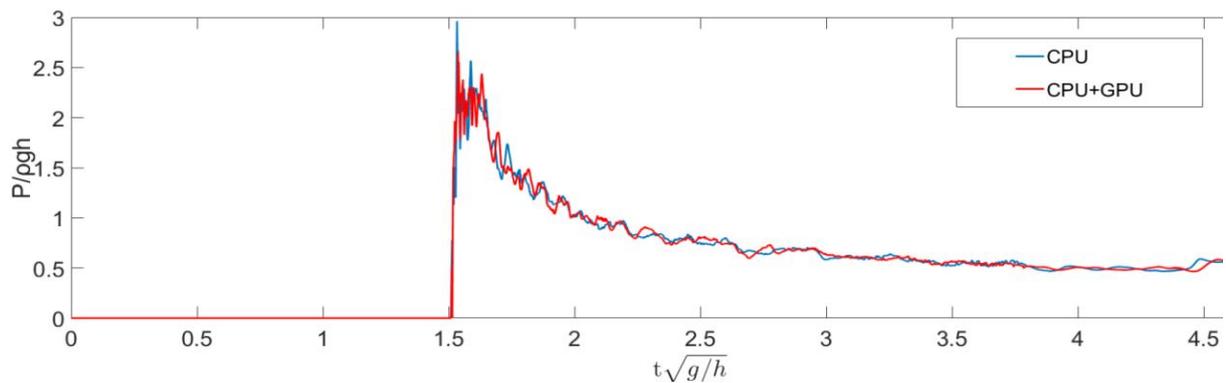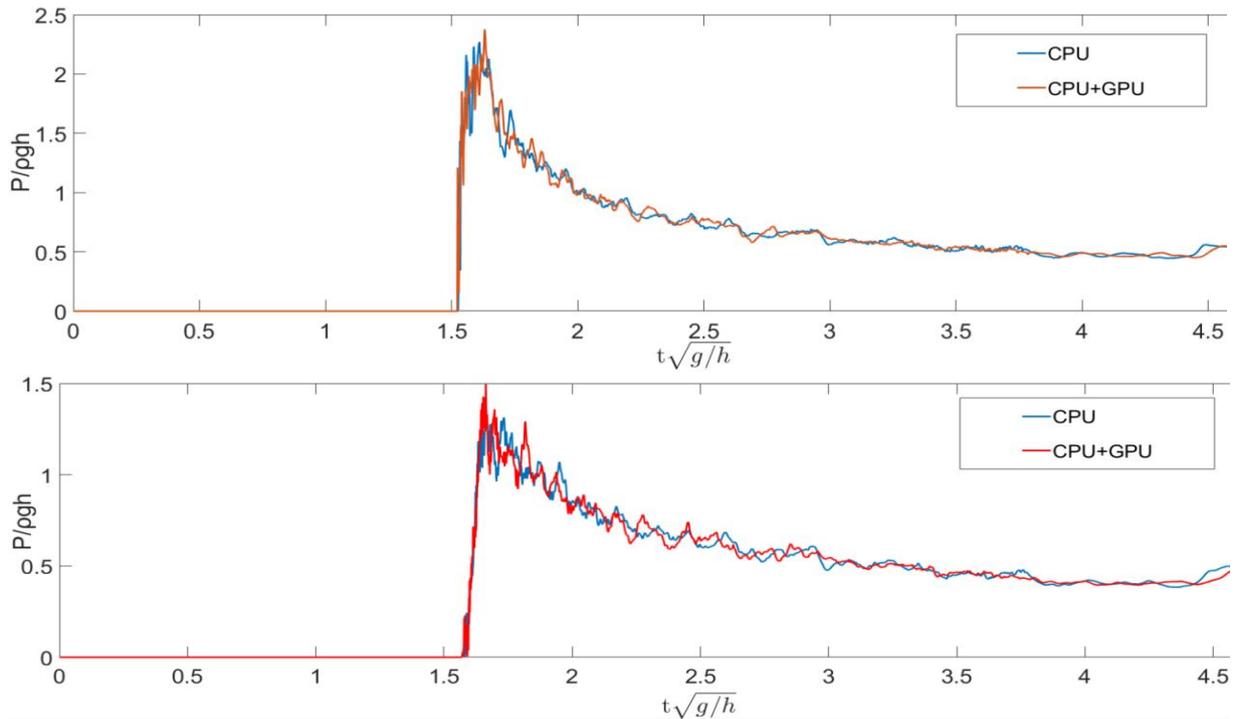
Figure. 10 Comparison of pressure duration curves at P1, P2 and P3
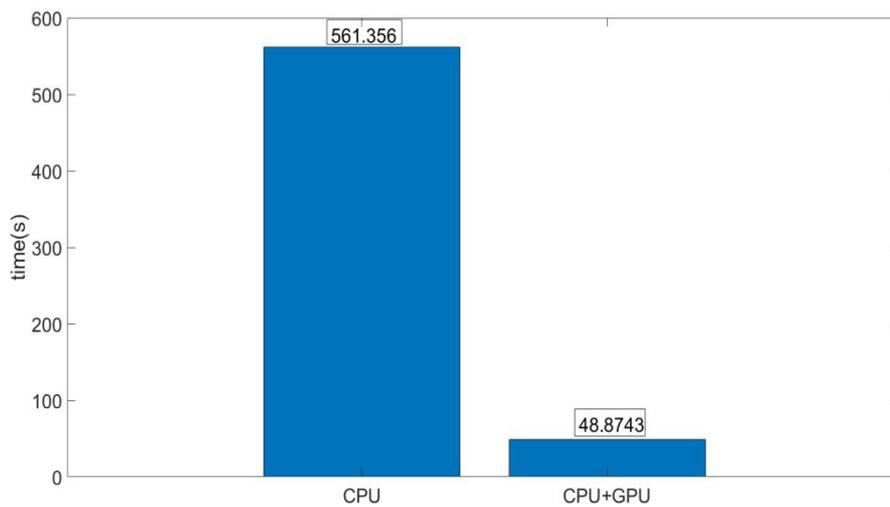
Figure 11. Pressure Poisson Equation Solution Time

As shown in Figure 11, By statistics, compared with the previous CPU serial code, the efficiency of GPU parallelization is improved by 91.3%.

3.3.2 The solution acceleration ratio of different particle numbers

After verifying the accuracy of the GPU parallel program, we discuss the influence of different particle numbers on the acceleration ratio. Table 1 lists the basic information of different 2-D dam-break examples.

Table 1.   Basic Parameters of the Calculation Example

dr	Particle numbers	iterations	Relax factors
0.005	15k	150	0.5
0.0075	7.1k	80	0.5
0.01	4.1k	45	0.5

The number of iterations here refers to the number of iterations in Relaxed Jacobi. Calculation and verification show that the number of iterations needs to be further increased with the increase of the number of particles in the example. Too small number of iterations will lead to the occurrence of calculation irregularities and particle erasure.

Through calculation, Fig. 12 shows the acceleration ratio of Poisson's equation under different particle numbers(Here the acceleration ratio is the ratio of the solution time of the pressure Poisson equation under CPU to the solution time of the pressure Poisson equation under GPU. The greater the acceleration, the better). It can be seen that the acceleration ratio increases with the increase of particle number,the maximum acceleration ratio can reach 11.486.
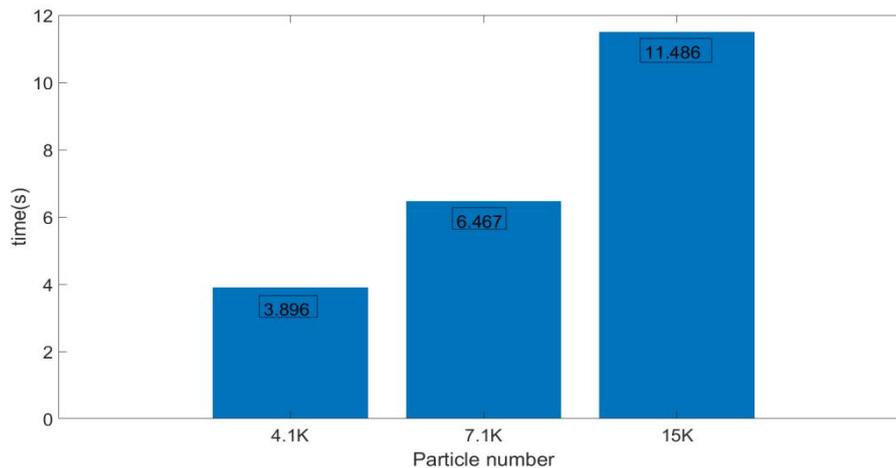
Figure 12. Acceleration ratio at different particle numbers

4. Conclusion

Based on the MPS — CPU solver, the Relaxed Jacobi method is used to solve the pressure Poisson equation explicitly instead of the GMRES implicit method. The simulation results are in good agreement with the GMRES implicit method, the efficiency of solving pressure Poisson equation is greatly improved. Based on the above research, the GPU parallelization of the Relaxed Jacobi method for the explicit solution of the pressure Poisson equation is completed, and the simulation results agree well with each other. The efficiency of the solution of the pressure Poisson equation is greatly improved. The maximum acceleration ratios of 11.486 can be obtained by simulating 2-D dam-break with different particle numbers.

On the whole, the overall computational efficiency has not been improved effectively due to the time consuming of matrix data transmission. Parallelization of a portion of the GPU does improve that portion of the computation. Therefore, on the basis of the current part of GPU, all the computations are given to GPU processing, so the potential of GPU high-speed computing can be more developed.

ACKNOWLEDGEMENTS

References:

[1] CUDA_C_Programming_Guide.NVIDIA,CUDA Toolkit Doc,2015.

[2] Harada T, Koshizuka S, Kawaguchi Y. Smoothed particle hydroddynamics on GPUs[C]. Computer Graphics International.Petropolis: SBC,2007: 63-70.

[3] Zhang,Y X, Solenthaler B, Pajarola R. GPU accelerated SPH particle simulation and rendering[C]. ACM SIGGRAPH 2007 posters. ACM, 2007.

[4] Crespo A J C, Dominguez J M, Barreiro A, et al. GPUs, a new tool of acceleration in CFD: efficiency and reliability on smoothed particle hydrodynamics methods[J]. PLoS One, 2011, 6(6): e20685.

[5] Herault, A., Bilotta, G., Dalrymple, R.A. SPH on GPU with CUDA[J]. Journal of Hydraulic Research, 2010, 48(Extra): 74-79.

[6] Wei, Z.P., Dalrymple, R.A., Hérault, A., Bilotta, G., Rustico, E., Yeh, H. SPH modeling of dynamic impact of tsunami bore on bridge piers[J]. Coastal Engineering, 2015, 104: 26-42.

[7] Chow, A.D., Rogers, B.D., Lind, S.J., Stansby, P.K. Numerical wave basin using incompressible smoothed particle hydrodynamics (ISPH) on a single GPU with vertical cylinder test cases[J]. Computers & Fluids, 2019, 179: 543-562.

[8] Hori, C., et al., GPU-acceleration for moving particle semi-implicit method[J]. Computers & Fluids, 2011. 51(1): p. 174-183.

[9] Kakuda, K., et al., Particle-based fluid flow simulations on GPGPU using CUDA[J]. CMES: Computer Modeling in Engineering and Sciences, 2012. 88(1): p. 17-28.

[10] Gou, W., S. Zhang and Y. Zheng, Simulation of isothermal multi-phase fuel-coolant interaction using MPS method with GPU acceleration[J]. Kerntechnik, 2016. 81(3): p. 330-336.

[11] Vieira-e-Silva, A.L.B., et al., Improved meshless method for simulating incompressible fluids on GPU[C]. 2017 19th Symposium on Virtual and Augmented Reality (SVR). Curitiba, Brazil.November 1-4,2017, p. 297-308.

[12] Taniguchi, D., Sato, L.M., Cheng, L.Y. Explicit moving particle simulation method on GPU cluster[C]. 10th World Congress on Computational Mechanics, Brazil, July 8-13, 2012: 1155-1168.

[13] Kawamura, K., Hashimoto, H., Onodera, N., Munesue, T. GPGPU simulation of oil tank sloshing based on explicit MPS method[C]. Proceedings of 3rd International Conference on Violent Flows, Osaka, Japan, March 9-11, 2016: Paper No.56.

[14] Hashimoto, H., Hata, Y., Kawamura, K. Estimation of oil overflow due to sloshing from oil storage tanks subjected to a possible Nankai Trough earthquake in Osaka bay area[J]. Journal of Loss Prevention in the Process Industries, 2017, 50: 337-346.

[15] Sun, Z., Djidjeli, K. & Xing, J. T.,A Mixed Particle-Mode function Method for Nonlinear Marine Fluid-Structure Interaction Problems with Free Surface[D],University of Southampton,2016.

[16] Koshizuka, S. and Oka, Y. (1996). Moving-particle semi-implicit method for fragmentation of incompressible flfluid.[J] *Nuclear Science and Engineering*, 123:421–434.

[17] Koh, C. G., Gao, M. & Luo, C., A new particle method for simulation of incompressible free surface flow problems[J]. *INT J NUMER METH ENG***89** 1582 (2012).

[18] Khayyer, A. and Gotoh, H. (2013). Enhancement of performance and stability of MPS mesh-free particle method for multiphase flflows characterized by high density ratios.[J] *Journal of Computational Physics*, 242:211–233.

[19] Kondo, M. and Koshizuka, S. (2011). Improvement of stability in moving particle semi-implicit method.[J] *International Journal for Numerical Methods in Fluids*, 65(6):638–654.

[20] Lee, B. H., Park, J. C., Kim, M. H., and Hwang, S. C. (2011). Step-by-step improvement of MPS method in simulating violent free-surface motions and impact-loads.[J] *Computer Methods in Applied Mechanics and Engineering*, 200(9-12):1113–1125.

[21] Lobovsky, L., Botia-Vera, E., Castellana, F., Mas-Soler, J., and Souto-Iglesias, A. (2014). Experimental investigation of dynamic pressure loads during dambreak. *Journal of Fluids and Structures*, 48:407–434.

Accuracy of a 3D adaptive mesh refinement method with analytical velocity fields

*Zhenquan Li[1] and Rajnesh Lal[2]

[1]School of Computing and Mathematics, Charles Sturt University, Thurgoona, NSW 2640, Australia.
[2]School of Mathematical and Computing Sciences, Fiji National University, Lautoka, Fiji.

*Presenting author and corresponding author: jali@csu.edu.au

Abstract

Meshing plays an important role on the accuracy and convergence of CFD solvers. The accuracy includes quantitative measures such as discretization and truncation errors and qualitative measures such as drawing closed streamline, identifying singular points, asymptotic lines/planes, and (symmetry) axis. The current study builds on previous work by further demonstrating the accuracy of the three-dimensional adaptive mesh refinement method by comparing the accuracy measures between the ones derived from analytical velocity fields and those identified by the refined meshes. The adaptive mesh refinement method presented in this study is proposed based on the law of mass conservation for three-dimensional incompressible or compressible steady fluid flows. The performance of the adaptive mesh refinement method is analysed using three-dimensional analytic velocity fields of four examples. The results provide evidence for the accuracy of the mesh refinement method in identifying the singular points, axes, and asymptote planes of the analytical velocity fields.

Keywords: Adaptive mesh refinement, Computational fluid dynamics (CFD), 3D velocity fields

Introduction

Discrete computational meshes are commonly employed in numerical high-performance computing modelling of physical processes to describe a specific problem's geometry or the general domain. To provide the requisite computation accuracy, discrete meshes are subjected to stringent requirements regarding the level and quality of discretization. Moreover, the simulation of computational fluid dynamics (CFD) problems is often largely reliant on mesh size for convergence and accuracy [1]. In many circumstances, the mesh size required grows to such proportions that the task becomes intractable for the computer resources available. For example, predicting the vortex trajectory of a large-scale flow, such as a tropical cyclone, may necessitate resolving the flow within and around the storm [2]. Furthermore, resolving localized features like vortex centers demands high resolution in areas where the numerical solution varies rapidly [3]. Such challenges can be addressed by adaptively refining the mesh during time-stepping.

There are three common adaptive techniques in CFD to reduce and control numerical error, such as local refinement and coarsening, known as *h*-refinement, adjusting the local order of discretisation of the numerical method, known as *p*-refinement, or optimising the distribution of the computational nodes via grid relocating or moving, known as *r*-refinement [4]. Other quantitative refinement approaches, such as numerical entropy generation schemes and weak

local residuals, include refinement and coarsening indicators [5][6]. To dynamically obtain high accuracy in a domain of interest based on some pre-defined criteria, the *h*-adaptivity technique is used in adaptive mesh refinement (AMR) [7]. Since the initial work by Berger and Oliger [8] on the application of block-structured AMR to two dimensional (2D) hyperbolic partial differential equations, the AMR approach has been widely extended to several multiscale domains of CFD [9]-[12]. AMR techniques have been effectively employed to minimize computational time and memory requirements for numerous applications in computational fluid dynamics (CFD), computational structural dynamics (CSD), and other fields of computational mechanics [4].

This study uses the AMR method proposed by Li [13][14] to refine a given mesh based on the three-dimensional (3D) velocity fields computed numerically. The AMR method in [13][14] is derived from a theorem in the qualitative theory of differential equations (Theorem 1.14, page 18, Ye [15]) for accurate numerical computation of 2D and 3D velocity fields. The refinement process can be repeated as many times as necessary until the desired level of accuracy or a certain threshold is reached. The mesh refinement technique [13][14] has previously been verified using the accurate locations of singular points, asymptotic lines, and closed streamlines [16]-[18]. Moreover, the accuracy of the 2D AMR method has also been verified against the commonly used CFD benchmark experiments such as the lid-driven cavity flow [19]-[22], the 2D unsteady flow past a square cylinder [23], and the backward-facing step flow [24]. Additionally, the AMR proposed by Li [13][14] has been shown to capture the centre of vortices within the refined cells of once refined meshes and within the twice refined cells after applying the AMR algorithm twice [21][25].

This paper builds on Li's [13] work by further demonstrating the accuracy of the 3D AMR method. Since the error of CFD simulation comes from both the error of numerical methods and the error of meshing, we demonstrate the accuracy of the AMR method using the computational velocity fields without computational errors; that is, the computational velocity fields are calculated by substituting the coordinates of the nodes of a mesh into analytical velocity fields. We show the accuracy of the 3D AMR method by comparing the accuracy measures between the analytical velocity fields and the refined meshes. We provide four examples of 3D AMR using 3D analytic velocity fields from [26]. The four examples presented in this paper provide evidence for the accuracy of the mesh refinement method in identifying the singular points, axes, and asymptote planes of the analytical velocity fields.

The Mass Conservation Conditions for Linear Interpolations of Vector Fields Over Tetrahedral Domains

The 3D AMR method is an extension of the 2D AMR method [14] derived from a theorem in the qualitative theory of differential equations [15].

The continuity equation for incompressible or steady-state fluid is
$$\nabla . \boldsymbol{V} = 0$$
which is the statement of the law of mass conservation.

Let $\boldsymbol{V}_l$ be the linear interpolation of the values at the four vertices of tetrahedra in the domain of the velocity field. It follows that for every tetrahedron, the velocity field can be computed as

$$\boldsymbol{V}_l = \boldsymbol{A}\boldsymbol{Y} + \boldsymbol{B}'$$

where $A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$, $B' = \begin{pmatrix} b_1' \\ b_2' \\ b_3' \end{pmatrix}$, and $Y = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}$ is a matrix of constants, vector of

constants, and the vector of spatial variables, respectively.

V_l is unique if the volume of the tetrahedron is not zero [16]. Substitution of V_l into V of the continuity equation gets

$$\nabla \cdot V_l = \text{trace}(A) = 0 \qquad (1)$$

for incompressible or steady-state fluid. However, the interpolated numerical velocity vector field V_l generally does not satisfy Eq. (1).

Let f be a scalar function of spatial variables y_1, y_2, and y_3. We assume that fV_l satisfies the continuity equation

$$\nabla \cdot (fV_l) = 0$$

and then calculate the expressions of f. Solving the above equation for the eight different Jacobian forms of the constant matrix A results in eight distinct expressions of the function f that are given in Table 1 [13]. In Table 1, $(y_1, y_2, y_3)^T = V^{-1}X$ and $(b_1, b_2, b_3)^T = V^{-1}B$ where V satisfies $A\mathfrak{I} = V\mathfrak{I}$, and $\mathfrak{I}$ is one of the Jacobian matrices in Table 1. The Jacobian forms of the constant matrix A and corresponding expressions of f for the eight cases in which the linear interpolations of the vector fields over tetrahedral domains do not hold the law of mass conservation is summarized in Table 1.

For $f \neq \{0, \infty\}$, the vectors V_l and fV_l produce same streamlines (for more details we refer the readers to Section 2.2 of [17].

The conditions (SH) for subdividing a hexahedron are as follows:

- for a hexahedron shown in Fig. 1, subdivide it into five tetrahedra as shown in Fig. 2.

- calculate the Jacobian form of A in $V_l = AY + B'$ for each of the five tetrahedra, respectively.

- if there exist at least one of the five expressions of f corresponding to the particular Jacobian of A in Table 1 equalling zero or infinity, at some points on the corresponding tetrahedra, subdivision is performed on the hexahedron.
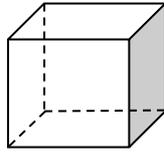
Fig. 1. A hexahedral cell.   **Fig. 2. Tetrahedral subdivision of a hexahedron.**
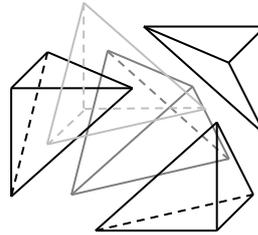
Table 1. Jacobian forms of the constant matrix A and expressions of f for all possible cases of a non-mass conservative linear field.

Case	Jacobian ($\Im$)	f
1	$\begin{pmatrix} r_1 & 0 & 0 \\ 0 & r_2 & 0 \\ 0 & 0 & r_3 \end{pmatrix}$ $(0 \neq r_1 \neq r_2 \neq r_3 \neq 0)$	$\left(y_1 + \dfrac{b_1}{r_1}\right)^{-1} \left(y_2 + \dfrac{b_2}{r_2}\right)^{-1} \left(y_3 + \dfrac{b_3}{r_3}\right)^{-1}$
2	$\begin{pmatrix} \mu & \lambda & 0 \\ -\lambda & \mu & 0 \\ 0 & 0 & r \end{pmatrix}$ $(r \neq 0, \lambda \neq 0)$	$\left\{\left(y_1 + \dfrac{\mu b_1 - \lambda b_2}{\mu^2 + \lambda^2}\right)^2 + \left(y_2 + \dfrac{\lambda b_1 + \mu b_2}{\mu^2 + \lambda^2}\right)^2\right\}^{-1} \left(y_3 + \dfrac{b_3}{r}\right)^{-1}$
3	$\begin{pmatrix} a & \delta & 0 \\ 0 & a & 0 \\ 0 & 0 & r \end{pmatrix}$ $(a \neq 0, r \neq 0)$ $(\delta = 0 \text{ or } 1)$	$\left(y_2 + \dfrac{b_2}{a}\right)^{-2} \left(y_3 + \dfrac{b_3}{r}\right)^{-1}$
4	$\begin{pmatrix} \mu & \lambda & 0 \\ -\lambda & \mu & 0 \\ 0 & 0 & 0 \end{pmatrix}$ $(\lambda \neq 0)$	$\left\{\left(y_1 + \dfrac{\mu b_1 - \lambda b_2}{\mu^2 + \lambda^2}\right)^2 + \left(y_2 + \dfrac{\lambda b_1 + \mu b_2}{\mu^2 + \lambda^2}\right)^2\right\}^{-1}$
5	$\begin{pmatrix} r & \delta & 0 \\ 0 & r & 0 \\ 0 & 0 & 0 \end{pmatrix}$ $(r \neq 0, \delta = 0 \text{ or } 1)$	$\left(y_2 + \dfrac{b_2}{r}\right)^{-2}$
6	$\begin{pmatrix} r & \delta & 0 \\ 0 & r & \delta \\ 0 & 0 & r \end{pmatrix}$ $(r \neq 0, \delta = 0 \text{ or } 1)$	$\left(y_3 + \dfrac{b_3}{r}\right)^{-3}$
7	$\begin{pmatrix} r & 0 & 0 \\ 0 & 0 & \delta \\ 0 & 0 & 0 \end{pmatrix}$ $(r \neq 0, \delta = 0 \text{ or } 1)$	$\left(y_1 + \dfrac{b_1}{r}\right)^{-1}$
8	$\begin{pmatrix} r_1 & 0 & 0 \\ 0 & r_2 & 0 \\ 0 & 0 & 0 \end{pmatrix}$ $(0 \neq r_1 \neq r_2 \neq 0)$	$\left(y_1 + \dfrac{b_1}{r_1}\right)^{-1} \left(y_2 + \dfrac{b_2}{r_2}\right)^{-1}$

The Adaptive Mesh Refinement Method

In practice, an unstructured mesh is typically employed, with most of the elements being hexahedra. The adaptive refinement approach is applied to every element in a mesh. A hexahedron to which the conditions (SH) applies can be decomposed into five or six

tetrahedra [27][28]. We divide a hexahedron into five tetrahedra in this study. The following algorithm describes how to refine a hexahedral cell in a mesh using the conditions (SH).

The refinement process of a hexahedral cell:
1) Subdivide the hexahedron into five tetrahedra and check if V_l satisfies the law of mass conservation on all five tetrahedra. If yes, no refinement for the hexahedron is required. If no, proceed to Step 2.
2) Apply the conditions (SH) to all tetrahedra. If the conditions (SH) are not satisfied on all tetrahedra, no subdivision is required. Otherwise, the cell is subdivided into a number of small elements such that the lengths of all sides of the small elements are truly reduced (e.g. half). Fig. 3 is an example that subdivides a hexahedron into eight smaller hexahedra by connecting the mid points of opposite sides on each of the six faces and O_1O_3, O_2O_4, O_5O_6. The new nodes are $AB, BC, CD, AD, AE, BF, CG, DH, EF, FG, GH, EH$ and $O_1, O_2, O_3, O_4, O_5, O_6, O$ as shown in Fig. 3.

The following is the algorithm of the adaptive mesh refinement method.
Algorithm of adaptive mesh refinement:
1. Let $T = 0$.
2. Calculate the values of a velocity field at nodes of an initial hexahedral mesh.
3. Perform the refinement process one by one for all cells in initial mesh and let $T = T + 1$.
4. Take the smaller hexahedra in the subdivided hexahedra in Fig. 3 as new cells of the initial mesh by replacing the cell in Fig. 1 if a cell is refined in Step 3. Otherwise, keep the cell in Fig. 1 in the initial mesh.
5. Repeat steps 2-5 until a pre-specified threshold number T is reached.
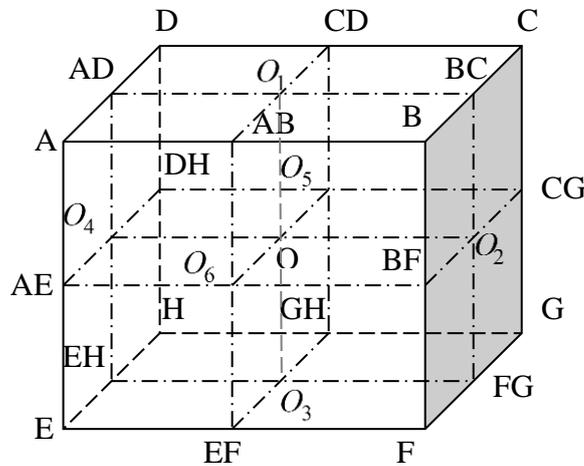
Figure 3. Subdivision of a hexahedron into eight small hexahedra.

In this study, we calculate the values of a velocity field in Step 2 of the algorithm at a point by substituting the coordinates of the nodes into the analytical velocity field. Since the number of refinements can be performed infinite times, we introduce a threshold number T in the algorithm. The choice of T depends on the required accuracy of the mesh, capacity of computers, or computational time.

Results

Four examples of 3D analytical velocity fields are shown here to demonstrate the effectiveness of the adaptive mesh refinement method. In these examples, the 3D analytical velocity fields, adopted from [26], are used to show that the values at the nodes of the refined meshes can present the fields very well by comparing the features shown in the refined meshes with the exact results. As the refinement process can be repeated as many times as necessary to achieve the desired level of accuracy or a certain threshold, we choose T as an integer. A higher threshold number, T, furnishes higher accuracy of numerical results based on the values at the nodes of the refined meshes.

Li [13] considered a different toroidal flow from the one considered in this paper. In [13], streamlines were drawn using the computational velocity fields on a refined mesh obtained by substituting the coordinates of nodes of the refined meshes into the analytical velocity field. For an exact closed streamline of the toroidal flow, a seed point was selected on the exact streamline. Then a streamline was drawn using the computational velocity field on a refined mesh, and the difference between the seed point and the end point (the x coordinate is the same as the x coordinate of the seed point) was compared. The distances between the seed and end points are smaller when the threshold number T is bigger.

In this section, we use four examples to provide evidence for the accuracy of the adaptive mesh refinement method in identifying the other qualitative measures for the accuracy of computational velocity fields.

Example 1: Helical flow

Velocity field: $\boldsymbol{V} = (-4y, x, 0.5)$.

Fig. 4 shows the initial mesh and three exact streamlines. These lines spiral around the z-axis. The variation of velocity fields at the points close to z-axis is smaller. Hence, more accurate computational velocity fields or a finer cell size are required for drawing accurate streamlines at the points close to the z-axis.
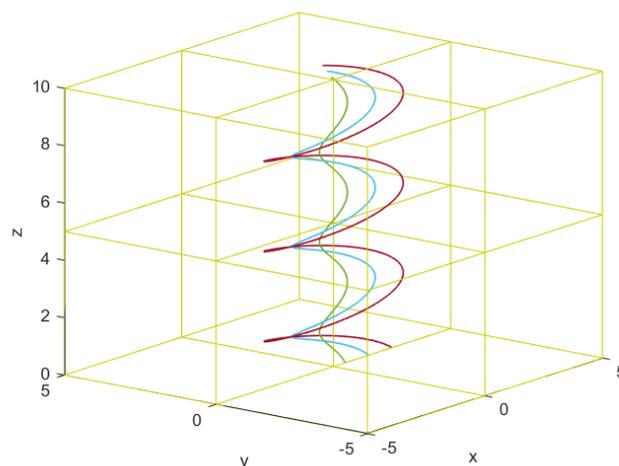
Figure 4. Initial mesh and the exact streamlines of helical flow.

Fig. 5 shows the refined mesh with the three streamlines. The cell sizes are smaller when cells are closer to z-axis. Fig. 6 shows the projection of refined mesh and the three streamlines on xy and yz-planes. The projection on the yz-plane indicates clearly that the cell sizes are getting smaller when cells are closer to the z-axis. The projection on the xz-plane is the same as that on the yz-plane. The projections of the three streamlines on the xy-plane are circles. The projection on the xy-plane again demonstrates that the cell sizes are getting smaller when cells are closer to the z-axis. This example demonstrates that the adaptive mesh refinement method can identify an axis accurately.
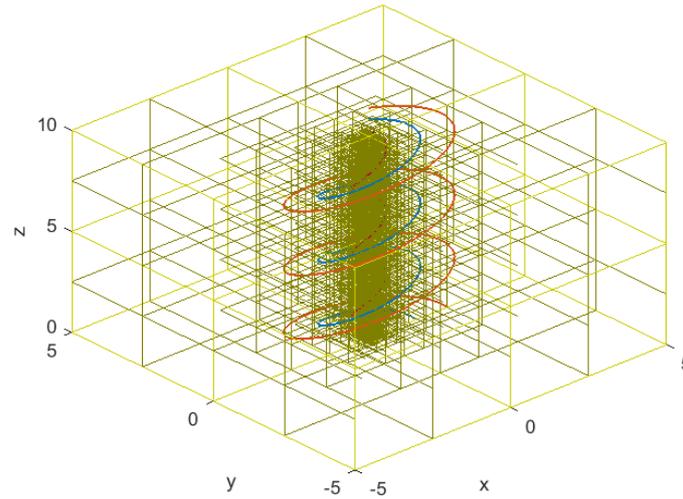
Figure 5. Refined mesh for $T = 7$ and streamlines of helical flow.
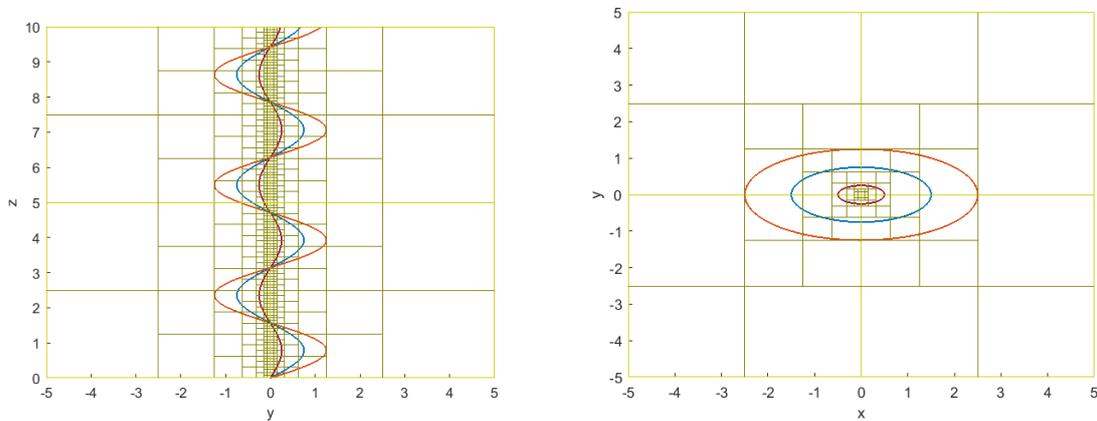
Figure 6. Projections of the graph in Fig. 5 on yz (left) and xy (right) planes.

Example 2: Saddle-spiral flow

Velocity field: $V = (-0.25xz - 5y, -0.25yz + 5x, -0.25z^2)$

Fig. 7 shows the initial mesh and two exact streamlines of the velocity field. The streamline on the top of xy-plane spirals down around the z-axis and gradually approximates the plane but never intersects with the plane. The streamline below xy-plane spirals up around z-axis and approximates the plane closer and closer but never intersects with the plane. Therefore,

xy-plane is asymptotic plane. Since the variations of velocity fields at the points closer to z-axis and xy-plane are smaller, the cell sizes must be smaller for drawing more accurate streamlines. Fig. 8 demonstrate that the finer cell size is closer to the z-axis and xy-plane. This example demonstrates that the adaptive mesh refinement method can identify axes and asymptotic planes. The singular point of this velocity field is the origin, and it is also identified in the refined mesh.
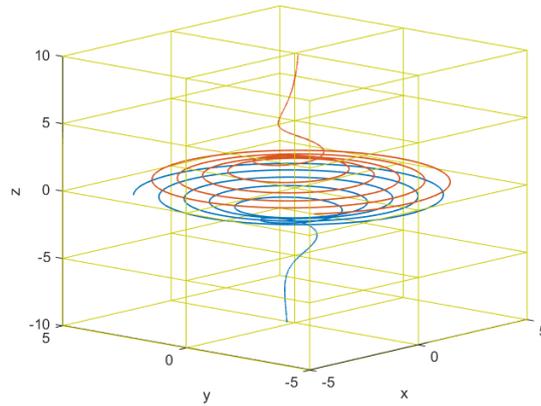
Figure 7. The initial mesh and the exact streamlines of Saddle-spiral flow.
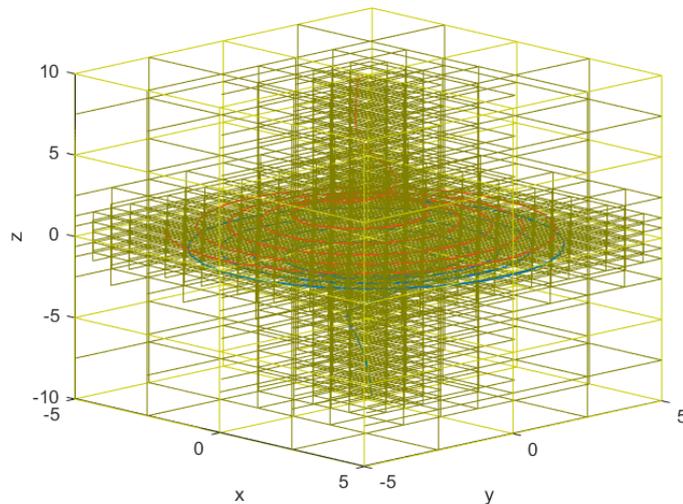
Figure 8. The refined mesh for $T = 5$ and the streamlines of Saddle-spiral flow.

Example 3: Toroidal flow

Velocity field:
$$V = \left(\frac{-x(z-4)}{r^2} - \frac{20y(r-2)}{r}, \frac{-y(z-4)}{r^2} + \frac{20x(r-2)}{r}, \frac{r-2}{r} \right)$$
where $r = \sqrt{x^2 + y^2}$.

Fig. 9 shows an exact streamline. When points are close to the z-axis, the velocity field at some of these points varies considerably, and when r is around 2, the variation of the

z component of velocity field is small. Therefore, we need a mesh with finer cells close to the z-axis and around $r = 2$.
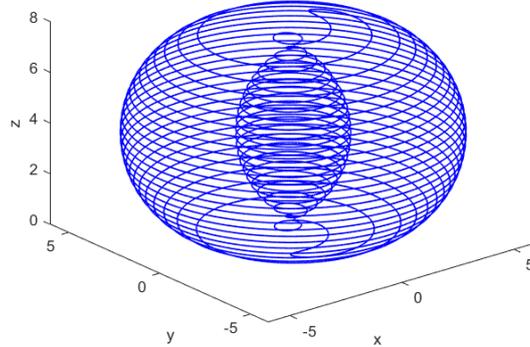
Figure 9. An exact streamline for Toroidal flow.

The refined mesh was generated for $T = 5$. The projections of the refined mesh on the yz- and xy-planes are shown in Fig. 10.
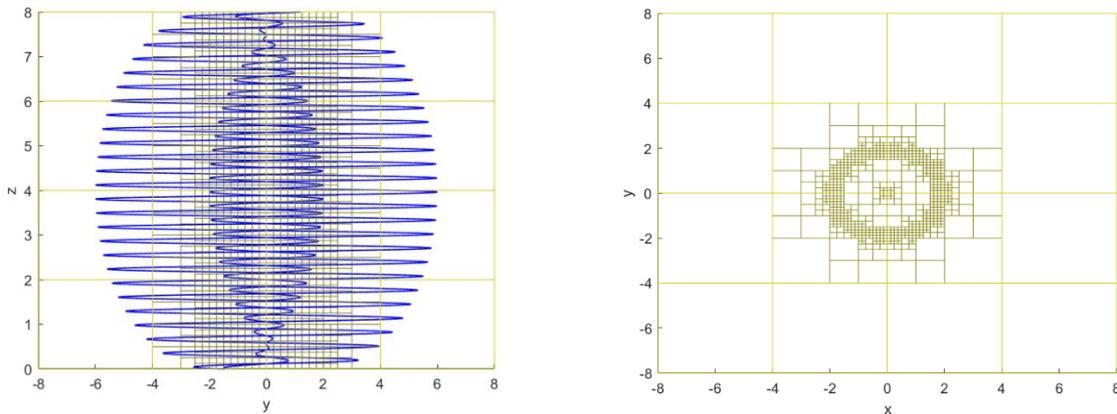
Figure 10. The projection on yz of the refined mesh with the streamline (left) and the projection on xy plane (right).

The projection of the refined mesh on the xy-plane shown on the right in Fig. 10 demonstrates that the cells sizes are getting smaller and smaller when r approaches 2 and cells are closer to the z-axis. The left figure in Fig. 10 shows that the refined cells are in the whole range of z coordinate in the domain. This example demonstrates that the adaptive mesh refinement can identify the areas where the velocity fields vary dramatically in value.

Example 4: Unstable focus-stretching flow

Velocity field: $\boldsymbol{V} = \left(\frac{-x}{2} - 40y, \frac{-y}{2} + 40x, -z\right)$.

Fig. 11 shows three exact streamlines. All streamlines spiral around the z-axis and towards the xy-plane. Some streamlines are close to the z-axis, and some of them are far away from the z-axis in the beginning and then close to the z-axis when they move close to the xy-plane

but never interact with the plane. Therefore, finer cells are required around the z-axis and xy-plane to compute a more accurate computational velocity field. The accuracy of computational velocity fields means the streamlines drawn using the computational velocity fields are accurate.
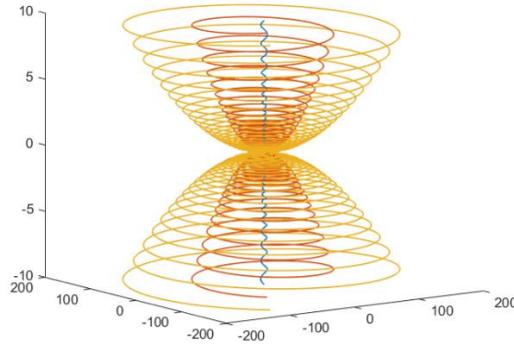
Figure 11. Three exact streamlines.
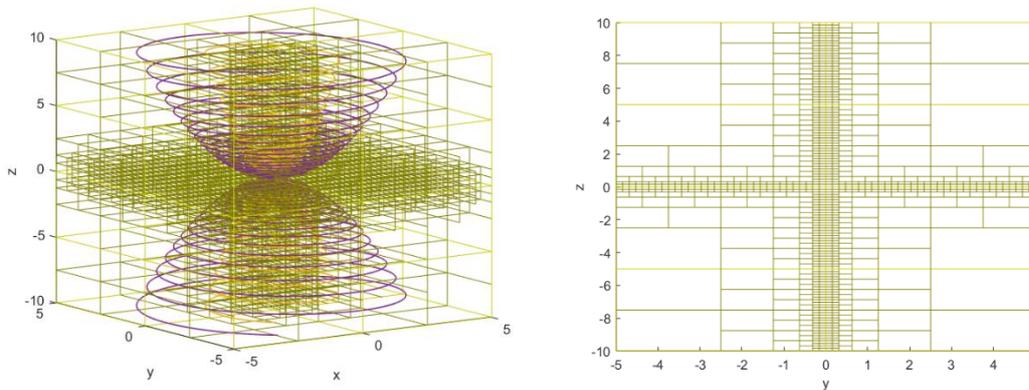
Figure 12. Refined mesh for $T = 7$ with streamlines (left) and the projection on yz plane (right).

The left figure in Fig. 12 shows the refined mesh for $T = 7$. Even though the refined mesh is similar to that of Saddle-spiral flow in Example 2, the two velocity fields are different. The right figure in Fig. 12 clearly shows that the z-axis and xy-plane are identified in the refined mesh. The singular point of this velocity is the origin, and it is identified in the refined mesh.

Discussion

The AMR method is implemented for three-dimensional unstructured meshes with hexahedra elements and extended to multi-level refinement. The results from the four examples are presented, which confirm the accuracy and efficiency of the 3D AMR method. Furthermore, we have assessed the performance of the adaptive mesh refinement method in identifying the accurate location of singular points, axes, asymptotic planes, and other features.

References

[1] Chakravarthy, S., and Akdag, V. (2015) Importance of accuracy in CFD simulations. *6th BETA CAE International Conference Importance*, 1-9.
[2] Fulton, S. R. (2001) An adaptive multigrid barotropic tropical cyclone track model. *Monthly weather review* **129(1),** 138-151.

[3] Amaziane, B., Bourgeois, M., and El Fatini, M. (2014) Adaptive mesh refinement for a finite volume method for flow and transport of radionuclides in heterogeneous porous media. *Oil & Gas Science and Technology–Revue d'IFP Energies nouvelles* **69(4),** 687-699.

[4] Löhner, R. (2008) Applied computational fluid dynamics techniques: an introduction based on finite element methods. John Wiley & Sons.

[5] Mungkasi, S., and Roberts, S. G. (2012) Behaviour of the numerical entropy production of the one-and-a-half-dimensional shallow water equations. *ANZIAM journal* **54**, C18-C33.

[6] Mungkasi, S., Li, Z., and Roberts, S. G. (2014) Weak local residuals as smoothness indicators for the shallow water equations. *Applied Mathematics Letters* **30**, 51-55.

[7] Rettenmaier, D., Deising, D., Ouedraogo, Y., Gjonaj, E., De Gersem, H., Bothe, D., Tropea, C. and Marschall, H. (2019) Load balanced 2D and 3D adaptive mesh refinement in OpenFOAM. *SoftwareX*, **10**, p.100317.

[8] Berger, M. J., and Oliger, J. (1984) Adaptive mesh refinement for hyperbolic partial differential equations. *Journal of computational Physics* **53(3),** 484-512.

[9] Bell, J., Berger, M., Saltzman, J., and Welcome, M. (1994) Three-dimensional adaptive mesh refinement for hyperbolic conservation laws. *SIAM Journal on Scientific Computing* **15(1)**, 127-138.

[10] Friedel, H., Grauer, R., and Marliani, C. (1997) Adaptive mesh refinement for singular current sheets in incompressible magnetohydrodynamic flows. *Journal of Computational Physics* **134**(1), 190-198.

[11] Rendleman, C. A., Beckner, V. E., Lijewski, M., Crutchfield, W., and Bell, J. B. (2000) Parallelization of structured, hierarchical adaptive mesh refinement algorithms. *Computing and Visualization in Science* **3**(3), 147-157.

[12] Berger, M. J., & Leveque, R. J. (1998) Adaptive mesh refinement using wave-propagation algorithms for hyperbolic systems. *SIAM Journal on Numerical Analysis* **35**(6), 2298-2316.

[13] Li, Z. (2007) An adaptive three-dimensional mesh refinement method based on the law of mass conservation. *Journal of Flow Visualization and Image Processing* **14(4)**, 375-395.

[14] Li, Z. (2008) An adaptive two-dimensional mesh refinement method based on the law of mass conservation. *Journal of Flow Visualization and Image Processing* **15(1)**, 17-33.

[15] Ye, Y. (1986) *Theory of Limit Cycles*, American Mathematical Society, Providence, Rhode Island.

[16] Li, Z. (2002) A mass conservative streamline tracking method for two-dimensional CFD velocity fields. *Journal of Flow Visualization and Image Processing* **9**, 75-87.

[17] Li, Z. (2006a) An adaptive streamline tracking method for two-dimensional CFD velocity fields based on the law of mass conservation. *Journal of Flow Visualization and Image Processing* **13**, 1-14.

[18] Li, Z. (2006b) An adaptive streamline tracking method for three-dimensional CFD velocity fields based on the law of mass conservation. *Journal of Flow Visualization and Image Processing* **13**, 359-376.

[19] Li, Z. (2014) Accuracy analysis of a mesh refinement method using benchmarks of 2-D lid-driven cavity flows and finer meshes. *Journal of Mathematical Chemistry* **52**, 1156-1170.

[20] Lal, R. and Li, Z. (2015) Sensitivity analysis of a mesh refinement method using the numerical solutions of 2-D steady incompressible driven cavity ow. *Journal of Mathematical Chemistry* **53**, 844-867.

[21] Li, Z. and Wood, R. (2017) Accuracy verification of a 2D adaptive mesh refinement method for incompressible or steady flow. *Journal of Computational and Applied Mathematics* **318**, 259-265.

[22] Li, Z. (2017a) Computational complexity of the algorithm for a 2D adaptive mesh refinement method using lid-driven cavity flows. Computational Thermal Sciences **9**, 395-403.

[23] Li, Z. (2017b) Analysis of 2D unsteady ow past a square cylinder at low Reynolds numbers with CFD and a mesh refinement method. *WSEAS Transactions on Fluid Mechanics* **12**, 150-157.

[24] Li, Z. and Li, M. (2021) Accuracy verification of a 2D adaptive mesh refinement method using backward-facing step ow of low Reynolds numbers. *International Journal of Computational Methods* **18**, 204-1012.

[25] Li, Z. and Wood, R. (2015) Accuracy analysis of an adaptive mesh refinement method using benchmarks of 2-D steady incompressible lid-driven cavity flows and coarser meshes. *Journal of computational and applied mathematics* **275**, 262-271.

[26] Reztsov, A. V., and Mallinson, G. D. (1998) Dual stream functions for 3D swirling flows. In *Proc. of 13th Australian Fluid Mech. Conference. Melbourne* 179-182.

[27] Knight, D., and Mallinson, G. (1996) Visualizing unstructured flow data using dual stream functions. *IEEE Transactions on Visualization and Computer Graphics* **2(4)**, 355-363.

[28] Nielson, G. M. (1994) Tools for Triangulations and Tetrahedrizations. In *Scientific Visualization* 429-525.

[29] Reyn, J. W. (1964) Classification and description of the singular points of a system of three linear differential equations. *Zeitschrift für angewandte Mathematik und Physik ZAMP* **15(5)**, 540-557.

Numerical study of the effect of shear keys on the stability of cantilever retaining walls

Changcheng Du[1], *Jianfeng Chen[1]

[1]Department of Geotechnical Engineering, College of Civil Engineering, Tongji University, China

*Presenting author: duchangcheng@tongji.edu.cn

Corresponding author: 01016@tongji.edu.cn

Abstract

Shear keys can enhance the slip resistance of cantilever retaining walls (CRWs), but their mechanisms are still not clear. In this paper, a numerical model for a CRW built and instrumented in Minnesota was carried out using a two-dimensional finite element program. The validated numerical model and Strength Reduction Method (SRM) were employed to investigate the effect of lengths and positions of the shear key on the stability of the CRW. The analysis shows that the slip surface passes the bottom of the shear key and is deepened and lengthened when a shear key is provided. The stability of the CRW with a shear key is improved because of better anti-slip capacities. The factor of safety of the CRW gradually increases from 1.038 to 1.268, approximately 22%, as the length of the shear key increases from 0 m to 0.6 m. The factor of safety of the CRW increases and then decreases as the shear key moves from the toe to the heel. The factor of safety is maximized when the shear key is set near the middle of the heel. This is because the rotation of the CRW results in a reduction in the effective length of the shear key at the end of the heel.

Keywords: Numerical method, Soil-structure interaction, Soil slope analysis

Introduction

Cantilever retaining walls (CRWs) are light in weight, low in cost, and simple in structure, which have been widely used in filling with low-bearing capacity foundations and limited construction space [1]-[3]. The anti-slip stability of the CRW is generally ensured by the friction resistance between the bottom of the footing and the foundation soil, but this kind of resistance is usually insufficient when the height of the wall stem is too large. Shear keys are often provided at the bottom of the footing to enhance the anti-slip resistance of CRW. For instance, the Minnesota Department of Transportation built a CRW with a shear key at the bottom of the footing to widen the width of an existing roadbed [1]. The monitoring results showed the CRW had been operating well.

At present, scholars have conducted some research on CRW mainly in terms of its earth pressure distribution characteristics, wall back thrust calculation methods, and wall movement characteristics. Kamiloğlu and Şadoğlu (2007) [4] proposed an earth pressure analytical solution considering the friction between the back of the wall and the backfill, based on the limit equilibrium method. Huang and Luo (2009, 2010) [5][6] experimentally investigated the effect of foundation settlement on the performance of CRWs. Al and Sitar (2010) [7] investigated the magnitude and distribution of lateral earth pressures acting on a CRW under

dynamic action. Bentler and Labuz [1] monitored earth pressures, wall displacements, and wall rotation angles of a CRW constructed in Minnesota.

To date, few studies have been carried out on the shear key of CRWs. Horvath (1991) [8] discussed the effect of footing shape on the performance of a CRW. It was found that the shape of the footing had a significant influence on the anti-slip resistance. The anti-slip resistance was enhanced in turn by shear keys located at the end of the toe, the underside of the wall stem, and the end of the heel, but there is no explanation for this phenomenon. Moreover, the anti-slip capacity is also not clear for the shear keys located in the middle of the heel. Although it is recognized that shear keys have a significant effect in improving slip resistance, current theory on the subject only treats them as a useful structural measure. It is still unclear where the shear keys should reasonably be located.

In this paper, a numerical model for a CRW built and instrumented in Minnesota was carried out using a two-dimensional finite element program. The validated numerical model and Strength Reduction Method (SRM) were employed to investigate the effect of lengths and positions of the shear key on the stability of the CRW. The research results are of some reference value for the reasonable consideration of the shear keys.

Project description

A 0.8 km long poured-in-place reinforced concrete CRW was constructed by the Minnesota Department of Transportation (Mn/DOT) on the south side of I - 494 near the West Bush Lake crossing in Bloomington, Minnesota (Latitude N 44°51′36″, Longitude W 93°22′36″) from 2002 to 2003. The location of the project is shown in Figure 1. The CRW consists of a wall stem, a footing, and a shear key. Displacements (2 survey reflectors 16 E and 25 E) and earth pressures (EPC_1, EPC_5, EPC_7, EPC_9) were monitored continuously for more than 12 months. The dimensions of the CRW, the location of the survey reflectors, and the location of EPCs are shown in Figure 2.

Figure 1. Project location [2]
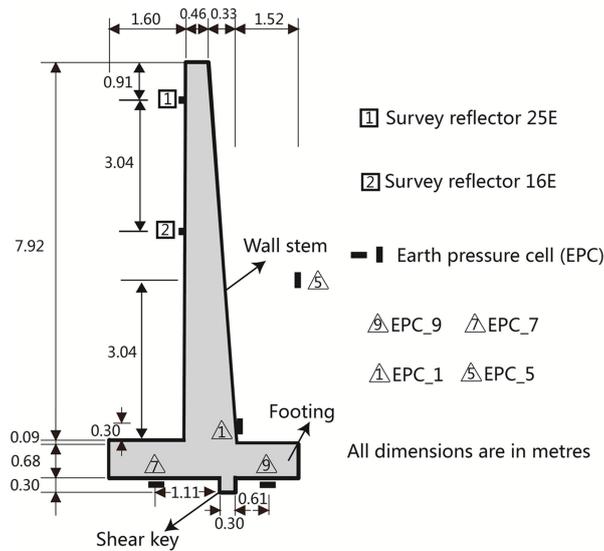
Figure 2. Schematic illustration of the CRW

Backfilling in layers after the wall construction. Drilling data shows that the foundation in the area is poorly graded sand and gravelly sand with an average moisture content of 12 %. The depth of bedrock is more than 7 m and the depth of groundwater burial is 1 - 1.5 m. The backfill and front fill are medium and fine sand with poor grading, gravity γ= 18.9 kN /m³, internal friction angle φ' = 35° - 39°, cohesion c' = 0.

Numerical modeling
Overview
The finite element program was employed to develop a plane strain model, as shown in Figure 3. The numerical model is 13.6 m in height and 30 m in width. It consists mainly of concrete CRW, front fill, backfill, and foundation. An unstructured finite element mesh consisting of triangular elements with 15 nodes was chosen. The wall and its surrounding soil were encrypted with group encryption and envelope point encryption respectively. In addition, the effect of groundwater was not considered in the model, so all materials were set to drain.
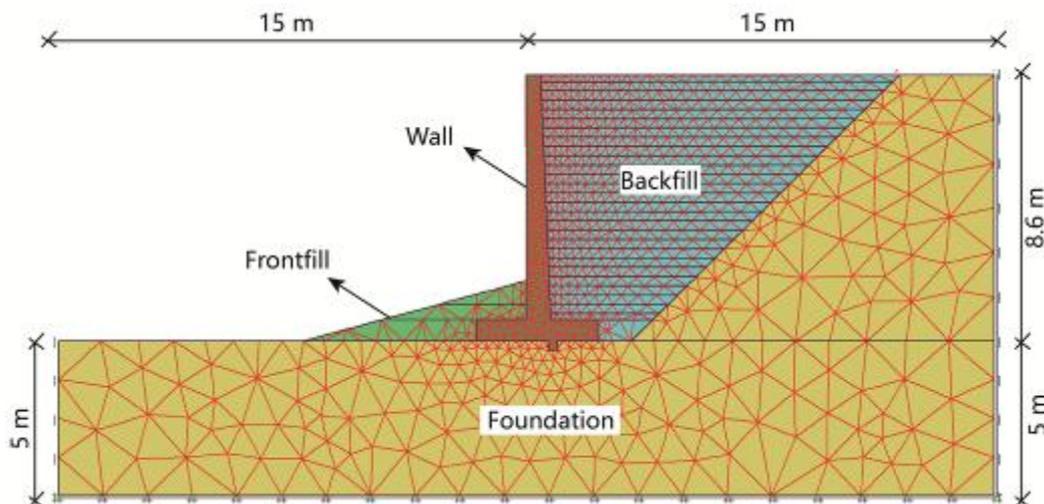
Figure 3. Numerical model dimensions

Material constitutive models and parameters

The Mohr-Coulomb (MC) constitutive model was selected to simulate the behavior of the front fill, the backfill, and the foundation. The isotropic linear elastic solid element was selected for the CRW. The material parameters for this model are summarized in Table 1. The material parameters were referenced from Chugh and Labuz (2011) [2]. It should be noted that some of the material parameters (e.g. the modulus of the soil) had been adjusted during the progress of numerical simulations in this paper.

Table 1 Material properties

Material	Constitutive model	γ_{unsat} (kN/m³)	v (-)	E_{ref} (MPa)	c' (kPa)	ϕ' (°)	Ψ (°)
Backfill	MC	18.81	0.33	35.1	0.1	38	8
Front fill	MC	17.64	0.33	30.2	0.1	36	6
Foundation	MC	15.68	0.33	70.0	10	30	0
Concrete	Elastic	23.52	0.16	32500	--	--	--

Notations: MC = Mohr-Coulomb model; γ_{unsat} = unit weight above the water table; v = Poisson's ratio; E_{ref} = Young's modulus; c' = effective cohesion; ϕ' = effective friction angle; ψ = dilation angle.

Interfaces and boundary conditions

Five types of interfaces were considered in the numerical model, as shown in Table 2. These interfaces were modeled as MC failure criteria. The interfaces were implemented by the interface strength reduction factor R_{inter} in the software PLAXIS 2D 8.5. It should be noted that the relationships between the actual parameters and the input parameters of the interfaces are as follows:

$$c_i = R_{inter} c_{soil} \tag{1}$$

$$\tan \varphi_i = R_{inter} \tan \varphi_{soil} \tag{2}$$

where c_i、φ_i is the actual parameters of the interfaces；c_{soil}、φ_{soil} is the input parameters of the interfaces

Table 2 Interface properties

Interface	Constitutive model	γ_{unsat} (kN/m³)	v (-)	E_{ref} (MPa)	c' (kPa)	ϕ' (°)	Ψ (°)	R_{inter} (-)
BW	MC	18.81	0.33	35.1	0.1	38	8	0.67
FFW	MC	17.64	0.33	30.2	0.1	36	6	0.67
FW	MC	15.68	0.33	70.0	10	30	0	0.67
FFF	MC	15.68	0.33	70.0	10	30	0	0.67
BF	MC	15.68	0.33	70.0	10	30	0	0.60

Notations: BW = backfill–wall interface; FFW = front fill–wall interface; FW = foundation–wall interface; FFF = front fill–foundation interface.; BF = backfill–foundation interface.; R_{inter} = reduction factor of property of interfaces.

In addition, the boundary conditions of the model were set as follows: The top surface was free displacement constraint; The left and right sides were horizontal displacement constraint ($u_x = 0$); The bottom surface was fully constrained for horizontal and normal displacements ($u_x = u_y = 0$).

Construction process

The construction process was simulated in the following order:

(i)   Gravity loading to generate the initial stress field in the foundation;
(ii) Freezing of the foundation at the shear key;
(iii) Activation of the footing and wall stem;
(iv) Activation of the front fill;
(v)   Activation of backfill on the side of the footing;
(vi) Activation of the rest of the backfill in layers.

Model validation

Comparisons of the results in this paper, the measured data in the field, and the results from Chugh and Labuz (2011) [2] are shown in Figure 4. The horizontal displacement, earth pressure of EPC_9, and earth pressure of EPC_7 are in good agreement with the measured data in the field. Although the earth pressures for EPC_1 and EPC_5 differ somewhat from the measured data in the field, the trend is consistent. In conclusion, the results in this paper are more satisfactory than those from Chugh and Labuz (2011) [2]. This numerical model is therefore acceptable and suitable as a typical example to analyze the mechanical behavior of CRWs

(a)   (b)

(c)

(d)

(e)
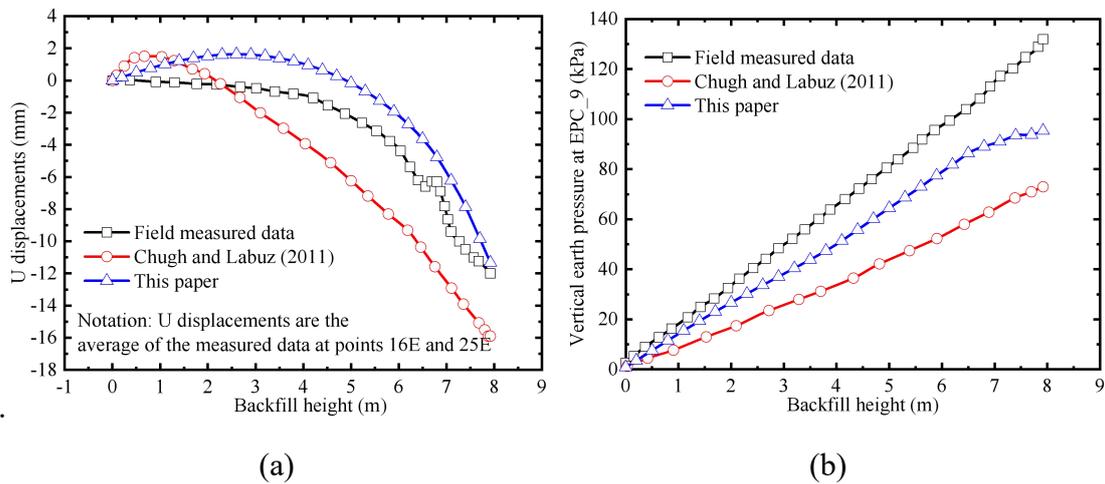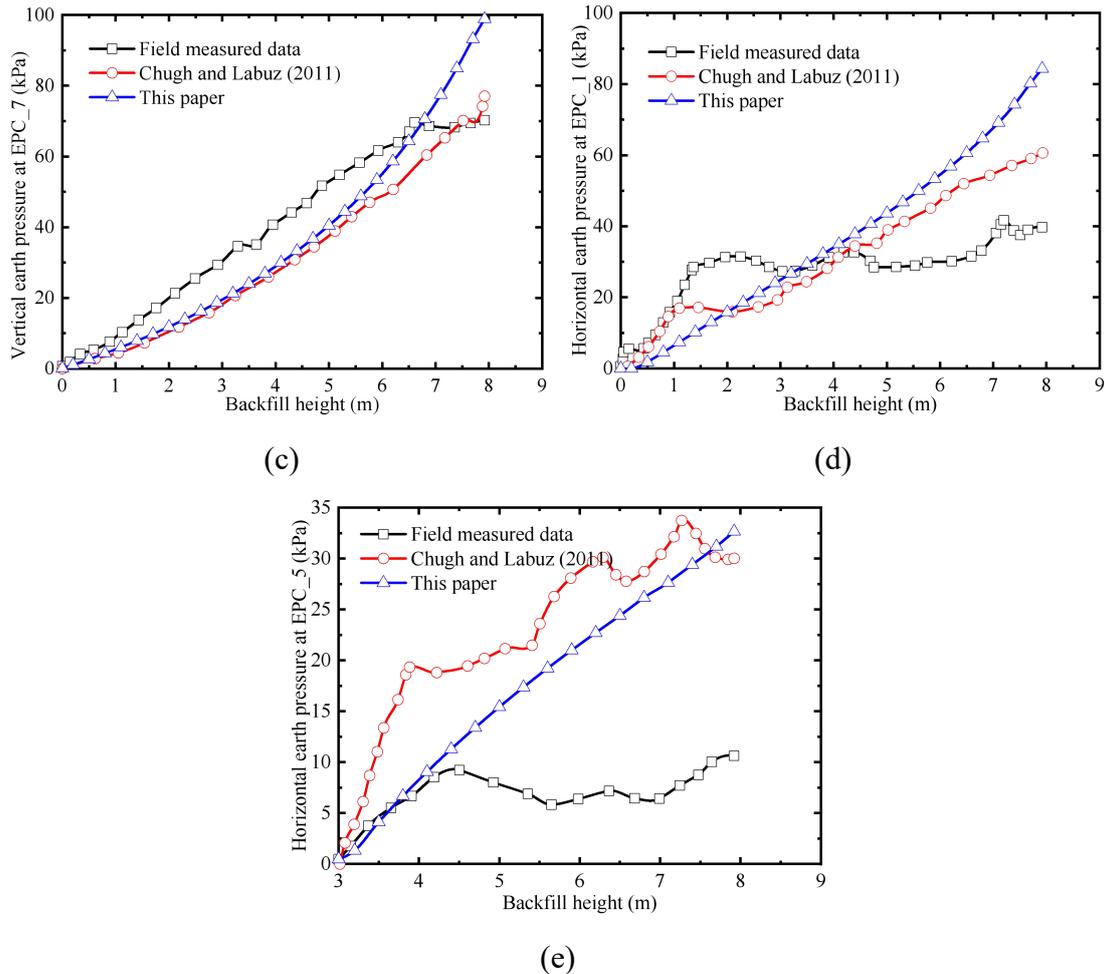
Fig. 4 Comparison between the results in this paper, the field measured data and the results from the literature [2]: (a) Horizontal displacement of the CRW; (b) Vertical earth pressure of EPC_9; (c) Vertical earth pressure of EPC_7; (d) Horizontal earth pressure of EPC_1; (e) Horizontal earth pressure of EPC_5.

Research programs

Ten types of shear keys were considered to investigate the effect of the length and position of the shear key on the anti-slip capacity of the CRW, as shown in Figure 5. The SRM built in the program was employed to analyze the stability of CRWs. Finally, the factor of safety was obtained [9].
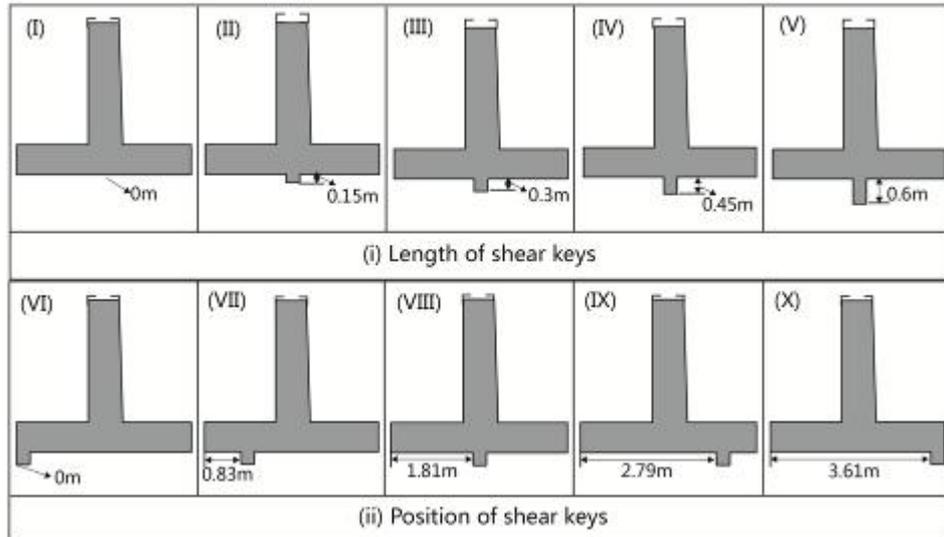
Figure 5. Ten types of shear keys considered in this paper

Results and discussion

Effect of the length of shear keys

Figure 6 shows the variation curve of the factor of safety of the CRWs with different lengths of shear keys. The lengths of the shear keys have a significant effect on the stability of CRWs. The factor of safety gradually increases from 1.038 to 1.268, an improvement of approximately 22%, as the lengths of the shear key increase from 0 m to 0.6 m.
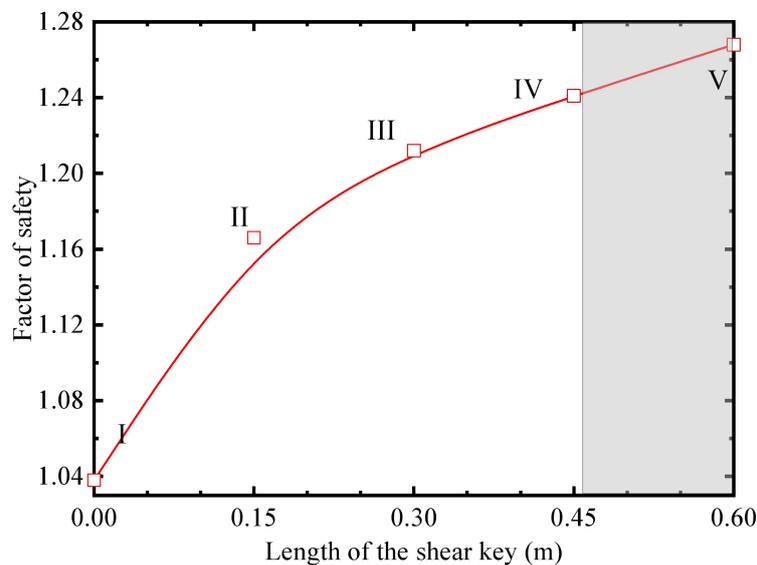
Figure 6 Variation curve of the factor of safety of the CRWs with different shear key lengths

The total displacement contours of CRWs based on SRM are plotted, as shown in Figure 7. The slip surface is the shortest and shallowest for a shear key of 0.15 m. The slip surface is the longest and deepest for a shear key of 0.6 m. The longer the shear key, the longer and deeper the slip surface. Therefore, foundation soils can provide a greater passive earth pressure and the stability of the retaining structure can be better. In practice, shear keys of a

certain length can be added to improve the anti-slip stability of CRWs.
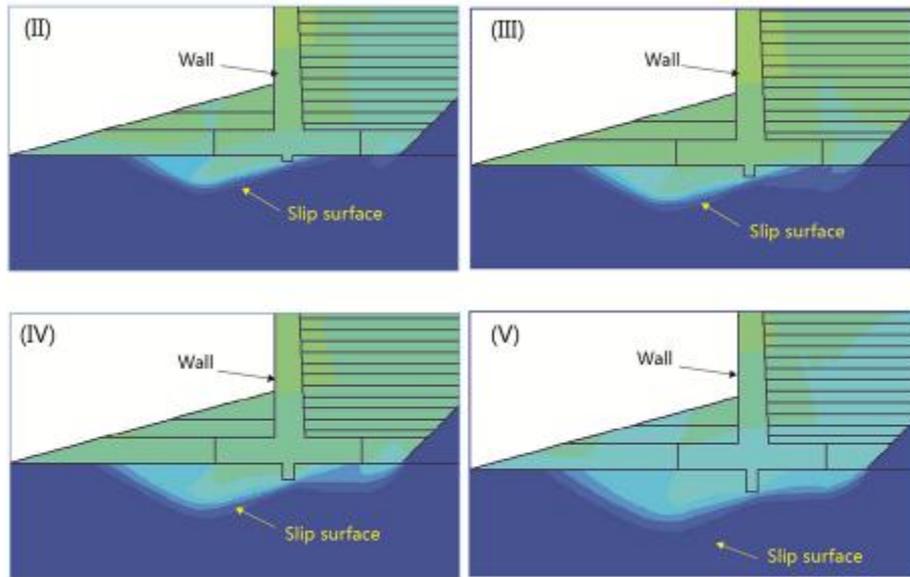
Figure 7 Total displacement contours for CRWs with different shear key lengths

Effect of the position of shear keys

Figure 8 shows the variation curve of the factor of safety of the CRWs with different shear key positions. The positions of the shear keys have a significant effect on the factor of safety. The factor of safety at X (1.215) and IX (1.228) is greater than that at VIII (1.212), VII (1.153), and VI (1.066). The factor of safety of the CRW with a shear key at the heel is greater than that with a shear key at the wall stem and the toe, which is consistent with the results of Horvath (1991) [8]. However, the factor of safety increases continuously and then decreases as the distance from the shear key to the end of the toe increases. An inflection point occurs at IX. This indicates that the factor of safety is maximum when the shear key is set near the middle of the heel. This did not describe in the analysis of Horvath (1991) [8].
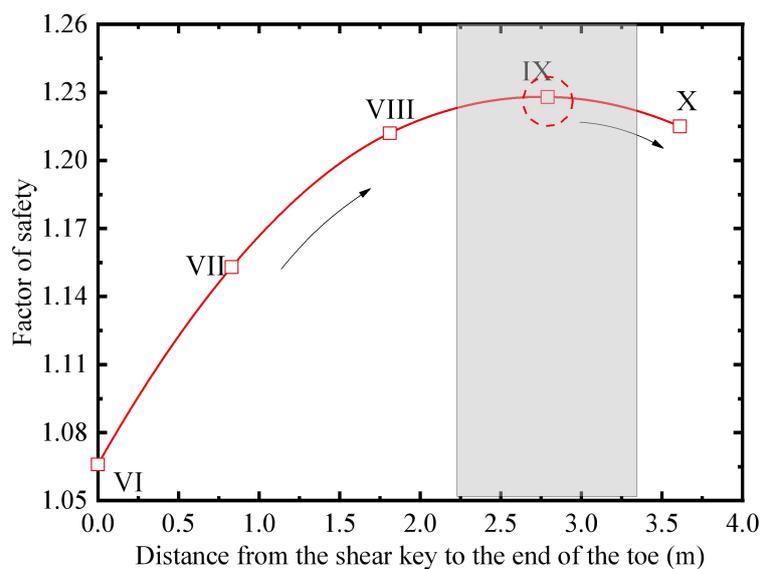
Figure 8 Variation curve of the factor of safety of the CRWs with different shear key positions

To investigate the reasons for these results, the total displacement contours of CRWs based on SRM are plotted, as shown in Figure 9. The depth and length of the slip surface first increase and then decreases, as the shear key moves from toe to heel. The slip surface of CRW with a shear key set near the middle of the heel is the deepest and longest. This is because CRWs not only translate but also rotate when backfilling, which results in a reduction of the effective length of the shear key set at the end of the heel, as shown in Figure 10. In practice, shear keys are suggested to be set near the middle of the heel to achieve the strongest slip resistance.
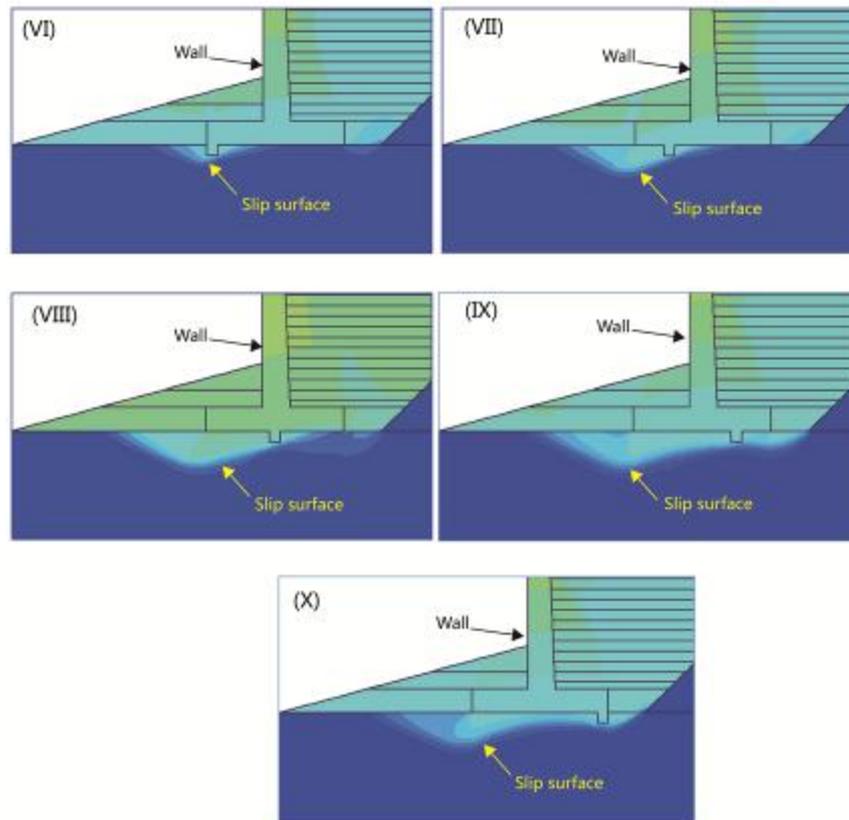
Figure 9 Total displacement contours for CRWs with different shear key positions
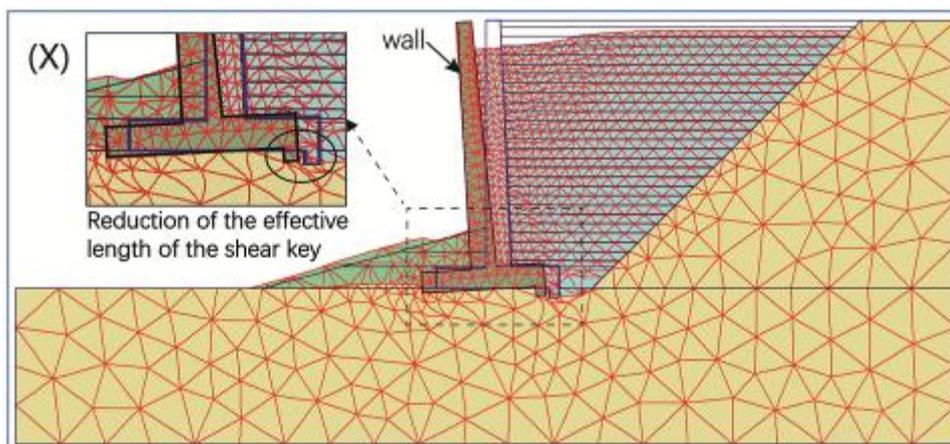
Figure 10 Grid deformation of the CRW with a shear key at the end of the heel

Conclusions

In this paper, a numerical model for a CRW built and instrumented in Minnesota was carried out using a two-dimensional finite element program. The validated numerical model and Strength Reduction Method (SRM) were employed to investigate the effect of lengths and positions of the shear key on the stability of the CRW. The main conclusions were summarized as follows:

(1) The analysis shows that the slip surface passes the bottom of the shear key and is deepened and lengthened when a shear key is provided. The stability of the CRW with a shear key is improved because of better anti-slip capacities.

(2) The factor of safety of the CRW gradually increases from 1.038 to 1.268, approximately 22%, as the length of the shear key increases from 0 m to 0.6 m. The factor of safety of the CRW increases and then decreases as the shear key moves from the toe to the heel.

(3) The factor of safety is maximized when the shear key is set near the middle of the heel. This is because the rotation of the CRW results in a reduction in the effective length of the shear key at the end of the heel.

Acknowledgments

Conflict of interest

The authors declare that they have no conflict of interest.

References

[1] Bentler, J. G. and Labuz, J. F. (2006) Performance of a cantilever retaining wall, Journal of geotechnical and geoenvironmental engineering 132(8), 1062-1070.

[2] Chugh, A. K. and Labuz, J. F. (2011) Numerical simulation of an instrumented cantilever retaining wall, Canadian geotechnical journal 48(9), 1303-1313.

[3] Greco, V. R. (2001) Active earth thrust on cantilever walls with a short heel, Canadian Geotechnical Journal 38(2), 401-409.

[4] Kamiloğlu, H. A. and Şadoğlu, E. (2017) Active earth thrust theory for horizontal granular backfill on a cantilever wall with a short heel, International Journal of Geomechanics 17(8), 04017018.

[5] Huang, C. C. and Luo, W. M. (2009) Behavior of soil retaining walls on deformable foundations, Engineering Geology 105(1-2), 1-10.

[6] Huang, C. C. and Luo, W. M. (2010) Behavior of cantilever and geosynthetic-reinforced walls on deformable foundations, Geotextiles and Geomembranes 28(5), 448-459.

[7] Al, A. L. and Sitar, N. (2010) Seismic earth pressures on cantilever retaining structures, *Journal of Geotechnical and Geoenvironmental Engineering* **136**(10), 1324-1333.

[8] Horvath, J. S. (1991) Effect of footing shape on the behavior of cantilever retaining wall, Journal of geotechnical engineering 117(6), 973-978.

[9] Brinkgreve, R. B. J., Broere, W., and Waterman, D. (2004) Plaxis Version 8 Reference Manual. Plaxis BV, Delft.

Semi-analytical solutions to 2D advection-dispersion-reaction equations in a finite domain subject to point-source and boundary-source

* Xianghong Ding [1], and † Shijin Feng [1,2]

[1] Department of Geotechnical Engineering, Tongji University, Shanghai 200092, China
[2] Key Laboratory of Geotechnical and Underground Engineering of the Ministry of Education, Tongji University, Shanghai 200092, China

* Presenting author: 1732201@tongji.edu.cn
† Corresponding author: fsjgly@tongji.edu.cn

Abstract

Advection-dispersion-reaction equations are widely used to simulate heat and mass transport problems in science and engineering. Analytical and semi-analytical solutions to such problems are highly desirable but are currently limited to a single type of source. This limitation poses significant challenges to the interaction analysis between different types of sources and the accurate inversion of the actual source zone. In this paper, we developed a two-dimensional analytical model for solute transport in a finite domain subject to both internal point sources and boundary sources. The solution approach applies Laplace transform combined with finite Fourier transform and variable substitution to obtain the generalized semi-analytical solution. An instantaneous point source system, together with Dirichlet and Robin inlet boundary, is selected to investigate the solute transport behavior in a multi-source scenario. Results reveal that the solute transport system with point source and Dirichlet boundary source has the largest predicted concentration. The selection of inlet boundaries for the model with low-permeability media (small Péclet number) or highly reactive (large Damköhler number) is of great importance, especially when performing long-term predictions.

Keywords: Solute transport, Advection-dispersion-reaction equation, Analytical solution, Point source, Boundary source

Introduction

Solute fate and transport in porous media are generally modeled using the advection-dispersion-reaction (ADR) equations. Analytical solutions to ADR equations are of great value as they provide more fundamental insight into migration behavior and can serve as a benchmark for complex numerical models. Consequently, a number of analytical or semi-analytical solutions to one- and multi-dimensional ADR equations have been developed to simulate various solute transport problems in porous media [1]. For example, Cleary and Adrian [2] and van Genuchten and Alves [3] presented several analytical solutions to the one-dimensional (1D) ADR equation with various combinations of boundary conditions. Batu [4][5] derived two-dimensional (2D) analytical solutions for solute transport in a unidirectional flow field subject to Dirichlet and Roin influent boundary conditions. Leij et al. [6] and Guerrero et al. [7] formulated analytical solutions for three-dimensional (3D) ADR equations in the semi-infinite and finite spatial domains, respectively. The solute sources in these models mentioned above are imposed by boundary conditions, and such problems are generally referred to as boundary-value problems.

In contrast to the boundary-value problems, another class of transport problems uses the source term of the ADR equation to introduce internal point sources. Bear [8] originally developed an analytical solution for instantaneous injection of a point source. Basha and El-Habel [9] proposed a 1D analytical solution for ADR equations with time-dependent source terms and dispersion coefficients in an infinite domain. Aral and Liao [10] generalized this solution to the two-dimensional infinite system and gave special solutions for instantaneous and constant-rate injection source scenarios. Employing the cosine Fourier series and Laplace transform, Fedi [11] derived an analytical solution for non-reactive solute transport in 2D semi-infinite domain with an instantaneous point injection source. Recently, Ding et al. [12] proposed a 2D analytical solution of ADR equations to investigate the reactive solute transport in a finite domain incorporating multiple arbitrary time-dependent point sources. However, the concentration gradient at the inlet boundary of the multi-point source model is set to zero, which cannot reflect the solute intrusion outside the transport system.

This study extends the method proposed by Ding et al. [12] and develops 2D analytical solutions for reactive solute transport in a finite field involving advection-dispersion-reaction processes subject to internal point sources and boundary sources. The validity of the present solutions is achieved by comparing them against corresponding numerical results. Using these solutions, the influent boundary conditions and transport parameters on the solute migration behavior will be investigated.

Transport model

This study presents a two-dimensional model for solute transport in a finite spatial domain with internal point sources and boundary sources. The groundwater flow is steady and uniform along the horizontal direction. The solute is injected through internal point mass sources and concentration sources at the inlet boundary. The injected solute migrates in the horizontal direction by advection and horizontal dispersion and undergoes vertical dispersion. The transport model also couples linear sorption and first-order reactions of solute, so it is described by the two-dimensional advection-dispersion-reaction equation as follows:

$$R\frac{\partial C(x,z,t)}{\partial t} = D_x\frac{\partial^2 C(x,z,t)}{\partial x^2} + D_z\frac{\partial^2 C(x,z,t)}{\partial z^2} - v\frac{\partial C(x,z,t)}{\partial x} - \mu C(x,z,t)$$
$$+ \sum_{i=1}^{n} q_i(t)\delta(x-x_i)\delta(z-z_i) \tag{1}$$

where C is the solute concentration [ML^{-3}] in the finite domain; R is the retardation factor [dimensionless]; D_x and D_z are the hydrodynamic dispersion coefficients [L^{2}T^{-1}], respectively; v is the pore-water seepage velocity [LT^{-1}]; μ is the first-order reaction constant [T^{-1}]. The time-dependent function $q_i(t)$ [ML^{-1}T^{-1}] and Dirac delta function $\delta(x-x_i)\delta(z-z_i)$ represents the strength and location of the i-th point sources in the finite domain, respectively.

Initially, the solute concentration in the finite field is assumed to be zero:

$$C(x,z,t=0) = 0 \tag{2}$$

The boundary conditions considered herein are:

$$\frac{\partial C(x,z=0,t)}{\partial z} = 0 \tag{3}$$

$$\frac{\partial C(x, z = H, t)}{\partial z} = 0 \tag{4}$$

$$C(x = 0, z, t) = c_s(z) \quad (Dirichlet\ boundary) \tag{5}$$

$$or\ -D_x \frac{\partial C(x = 0, z, t)}{\partial x} + vC(x = 0, z, t) = vc_s(z) \quad (Robin\ boundary) \tag{6}$$

$$\frac{\partial C(x = L, z, t)}{\partial x} = 0 \tag{7}$$

Two different inlet boundaries are adopted, namely the Dirichlet boundary (or concentration-type) condition of Eq.(5) and the Robin boundary (flux-type) condition of Eq.(6). The concentration of these two types of boundary sources can be an arbitrary depth-dependent function.

Solution method

For mathematical convenience and solution generalization, the following dimensionless parameters are introduced.

$$x_D = \frac{x}{L}, z_D = \frac{z}{H}, t_D = \frac{tv}{L}, \xi = \frac{L}{H}, \eta = \frac{D_z}{D_x}, \text{Pe} = \frac{vL}{D_x}, \text{Da} = \frac{\mu L^2}{D_x}, C_D = \frac{C}{C_0}, Q_i = \frac{q_i}{C_0 D_x} \tag{8}$$

where C_0 is the maximum concentration at the inlet boundary ($x = 0$), and thus the dimensionless concentration of boundary source $c_{s,D} = c_s/C_0$; Pe is the Péclet number, and Da is the Damköhler number. Then, substituting these above dimensionless parameters into Eqs. (1)-(7), one derives the dimensionless form as follows:

$$\text{Pe}R \frac{\partial C_D(x_D, z_D, t_D)}{\partial t_D} = \frac{\partial^2 C_D(x_D, z_D, t_D)}{\partial x_D^2} + \eta \xi^2 \frac{\partial^2 C_D(x_D, z_D, t_D)}{\partial z_D^2} - \text{Pe} \frac{\partial C_D(x_D, z_D, t_D)}{\partial x_D}$$
$$- \text{Da}C_D(x_D, z_D, t_D) + \sum_{i=1}^{n} \xi Q_i(t_D) \delta(x_D - x_{D,i}) \delta(z_D - z_{D,i}) \tag{9}$$

$$C_D(x_D, z_D, t_D = 0) = 0 \tag{10}$$

$$\frac{\partial C_D(x_D, z_D = 0, t_D)}{\partial z_D} = 0 \tag{11}$$

$$\frac{\partial C_D(x_D, z_D = 1, t_D)}{\partial z_D} = 0 \tag{12}$$

$$C_D(x_D = 0, z_D, t_D) = c_{s,D}(z_D) \tag{13}$$

$$or\ -\frac{\partial C_D(x_D = 0, z_D, t_D)}{\partial x_D} + \text{Pe}C_D(x_D = 0, z_D, t_D) = \text{Pe}c_{s,D}(z_D) \tag{14}$$

$$\frac{\partial C_D(x_D = 1, z_D, t_D)}{\partial x_D} = 0 \tag{15}$$

Applying the Laplace transform and finite Fourier transform techniques to the time variable t and spatial variable z of the governing equation (Eq. (9)), combining the initial condition of Eq. (10) and boundary conditions of Eqs. (11)-(12), one can give a second-order ordinary differential equation in the transform domain:

$$\frac{\partial^2 \hat{\bar{C}}_D(x_D, k, s)}{\partial x_D^2} - \mathrm{Pe}\frac{\partial \hat{\bar{C}}_D(x_D, k, s)}{\partial x_D} - (\mathrm{Pe}Rs + \mathrm{Da} + \eta\xi^2 k^2 \pi^2)\hat{\bar{C}}_D(x_D, k, s)$$

$$= -\sum_{i=1}^{n} \xi \bar{Q}_i(s)\delta(x_D - x_{D,i})\cos(k\pi z_{D,i}) \tag{16}$$

where s and k are the Laplace transform and finite Fourier cosine transform parameters, respectively, and the specific expressions of the two transform techniques are:

$$\bar{f}(s) = L_T\left[f(t_D)\right] = \int_0^{+\infty} f(t_D)e^{-st_D}dt_D \tag{17}$$

$$\hat{g}(k) = F_c\left[g(z_D)\right] = \int_0^1 g(z_D)\cos(k\pi z_D)dz_D \tag{18}$$

where L_T and F_c are the Laplace transform and Fourier cosine transform operators, respectively.

After, the above governing equation (Eq. (16)) can be transformed into:

$$\frac{d}{dx_D}\left(\frac{d\hat{\bar{C}}_D(x_D, k, s)}{dx_D} - \alpha_k \hat{\bar{C}}_D(x_D, k, s)\right) - \beta_k\left(\frac{d\hat{\bar{C}}_D(x_D, k, s)}{dx_D} - \alpha_k \hat{\bar{C}}_D(x_D, k, s)\right)$$

$$= -\sum_{i=1}^{n} \xi \bar{Q}_i(s)\delta(x_D - x_{D,i})\cos(k\pi z_{D,i}) \tag{19}$$

where

$$\begin{cases} \alpha_k = (\mathrm{Pe} + \sqrt{\mathrm{Pe}^2 + 4(\mathrm{Pe}\,Rs + \mathrm{Da} + \eta\xi^2 k^2 \pi^2)})/2 \\ \beta_k = (\mathrm{Pe} - \sqrt{\mathrm{Pe}^2 + 4(\mathrm{Pe}\,Rs + \mathrm{Da} + \eta\xi^2 k^2 \pi^2)})/2 \end{cases} \tag{20}$$

Using the substitution method [12], the following new variable is introduced:

$$Y(x_D, k, s) = \frac{d\hat{\bar{C}}_D(x_D, k, s)}{dx_D} - \alpha_k \hat{\bar{C}}_D(x_D, k, s) \tag{21}$$

Substituting the variable of Eq. (21) into Eq. (19) yields:

$$\frac{dY(x_D, k, s)}{dx_D} - \beta_k Y(x_D, k, s) + \sum_{i=1}^{n} \xi \bar{Q}_i(s)\delta(x_D - x_{D,i})\cos(k\pi z_{D,i}) = 0 \tag{22}$$

Solving the above two coupled first-order ordinary differential equations (Eqs. (21)-(22)) gives the general solution in the Laplace-Fourier transform domain:

$$\hat{\bar{C}}_D(x_D, k, s) = A_k e^{\alpha_k x_D} + B_k e^{\beta_k x_D} - \sum_{i=1}^{n} \frac{\xi \bar{Q}_i(s)H_v(x_D - x_{D,i})(e^{\alpha_k(x_D - x_{D,i})} - e^{\beta_k(x_D - x_{D,i})})\cos(k\pi z_{D,i})}{(\alpha_k - \beta_k)} \tag{23}$$

where the coefficients A_k and B_k can be determined by inlet and outlet boundary conditions, and they are provided as follows.

Dirichlet inlet boundary scenario:

Applying the Laplace and finite Fourier cosine transforms (Eqs. (17)-(18)) to the inlet and outlet boundary conditions (Eqs. (13) and (15)) yields:

$$
\begin{cases}
\hat{\bar{C}}_D(x_D = 0, k, s) = \hat{c}_{s,D}(k) \\
\dfrac{\partial \hat{\bar{C}}_D(x_D = 1, k, s)}{\partial x_D} = 0
\end{cases}
\tag{24}
$$

Substituting the boundary conditions of Eq. (24) in the transform domain into the general solution of Eq. (23), the coefficients A_k and B_k are solved as follows:

$$
\begin{cases}
A_k = \dfrac{G(k,s) - \beta_k e^{\beta_k} \hat{c}_{s,D}(k)}{\alpha_k e^{\alpha_k} - \beta_k e^{\beta_k}} \\
B_k = \dfrac{\alpha_k e^{\alpha_k} \hat{c}_{s,D}(k) - G(k,s)}{\alpha_k e^{\alpha_k} - \beta_k e^{\beta_k}}
\end{cases}
\tag{25}
$$

where

$$
G(k,s) = \sum_{i=1}^{n} \frac{\xi \bar{Q}_i(s)(\alpha_k e^{\alpha_k(1-x_{D,i})} - \beta_k e^{\beta_k(1-x_{D,i})})\cos(k\pi z_{D,i})}{(\alpha_k - \beta_k)}
\tag{26}
$$

Robin inlet boundary scenario:

Following a similar procedure as above, the Robin inlet and Neumann outlet boundary conditions (Eqs. (14) and (15)) can be rewritten in the Laplace-Finite cosine transform domain as follows:

$$
\begin{cases}
-\dfrac{\partial \hat{\bar{C}}_D(x_D = 0, k, s)}{\partial x_D} + \mathrm{Pe}\hat{\bar{C}}_D(x_D = 0, k, s) = \mathrm{Pe}\hat{c}_{s,D}(k) \\
\dfrac{\partial \hat{\bar{C}}_D(x_D = 1, k, s)}{\partial x_D} = 0
\end{cases}
\tag{27}
$$

and the corresponding coefficients A_k and B_k are:

$$
\begin{cases}
A_k = \dfrac{(\mathrm{Pe} - \beta_k)G(k,s) - \beta_k e^{\beta_k}\mathrm{Pe}\hat{c}_{s,D}(k)}{\alpha_k e^{\alpha_k}(\mathrm{Pe} - \beta_k) - \beta_k e^{\beta_k}(\mathrm{Pe} - \alpha_k)} \\
B_k = \dfrac{e^{\alpha_k}\mathrm{Pe}\hat{c}_{s,D}(k) - (\mathrm{Pe} - \alpha_k)G(k,s)}{\alpha_k e^{\alpha_k}(\mathrm{Pe} - \beta_k) - \beta_k e^{\beta_k}(\mathrm{Pe} - \alpha_k)}
\end{cases}
\tag{28}
$$

Finally, employing the inverse Fourier cosine transform to Eq. (23) yields a closed-form solution for the solute concentration in the Laplace domain, as follows:

$$
\begin{aligned}
\bar{C}_D(x_D, z_D, s) = {}& A_{k=0}e^{\alpha_{k=0}x_D} + B_{k=0}e^{\beta_{k=0}x_D} - \sum_{i=1}^{n} \frac{\xi \bar{Q}_i(s)H_v(x - x_{D,i})(e^{\alpha_{k=0}(x_D - x_{D,i})} - e^{\beta_{k=0}(x_D - x_{D,i})})}{(\alpha_{k=0} - \beta_{k=0})} \\
& + \sum_{k=1}^{\infty} 2(A_k e^{\alpha_k x_D} + B_k e^{\beta_k x_D} - \sum_{i=1}^{n} \frac{\xi \bar{Q}_i(s)H_v(x - x_{D,i})(e^{\alpha_k(x_D - x_{D,i})} - e^{\beta_k(x_D - x_{D,i})})\cos(k\pi z_{D,i})}{(\alpha_k - \beta_k)})\cos k\pi z_D
\end{aligned}
\tag{29}
$$

and the Laplace inversion is then implemented using the Stehfest inversion algorithm to give the transient concentration solution.

Result and discussion

In this section, an instantaneous-release source scenario is used as an example to investigate the correctness of the developed solution as well as its practical applications. The mathematical description of the instantaneous source strength is：

$$q_i(t) = M_i \delta(t - t_i) \tag{30}$$

where M_i is the released mass of the i-th point source; and $\delta(t-t_i)$ is a Dirac delta function where t_i is the release moment.

Two instantaneous point sources are located at the ($x = 5$m, $z = 7.5$m) and ($x = 5$m, $z = 12.5$m) in a finite domain, and each source has a total release mass of $M_1 = M_2 = 100$ g/m at $t_1 = t_2 = 0$ day. Other material and transport parameters for the finite domain model are given in Table 1.

Table 1. Input Parameters [12].

Parameter description	Symbol	Value
Length of the finite spatial domain	L	30 m
Height of the finite spatial domain	H	20 m
The average seepage velocity	v	0.1 m/day
Horizontal hydrodynamic dispersion coefficients	D_x	0.1 m²/day
Vertical hydrodynamic dispersion coefficients	D_z	0.03 m²/day
Sorption retardation factor	R_d	5.3
First-order decay rate coefficient	μ	0.002 day⁻¹

Comparison with analytical solutions and numerical results

As mentioned in the introduction, the Neumann inlet boundary is generally used in previous point source models, which does not reflect the influence of external sources on the transport system. Therefore, this study compared three types of inlet boundary sources on the solute concentration distribution in the transport system (Fig. 1). The predictive model with a constant concentration boundary source (i.e., Dirichlet inlet boundary) presented the maximum solute concentration. The difference in concentration prediction for different inlet boundary cases decreased with an increase in transport distance but gradually increased with time. This implied that solute transport models subject to both the boundary sources and the internal point sources should pay attention to the selection of the inlet boundary conditions, especially when performing long-term predictions. Fig. 1 also shows that the present analytical solutions for Dirichlet-boundary (displayed as solid curves) and Robin-boundary cases (dash-dot curves) agree well with the numerical results (open dots), providing some confidence in the reliability of the developed analytical solution.
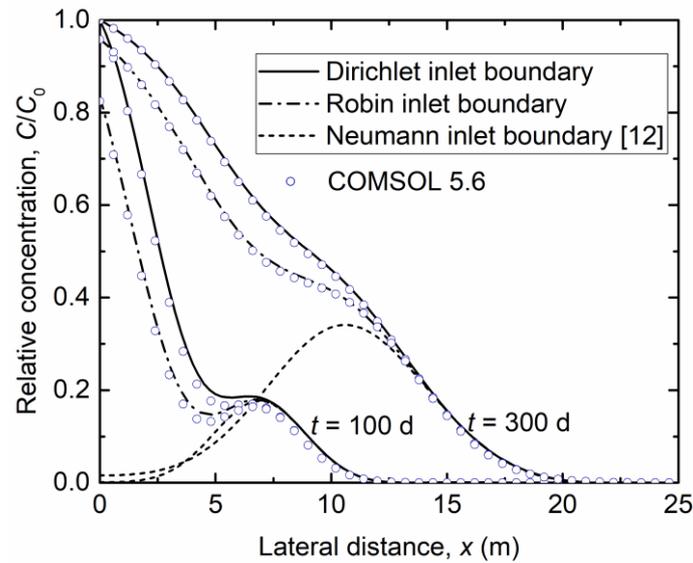
Figure 1. Effect of the inlet boundary condition on the concentration distributions

Effect of Péclet number (Pe) *and Damköhler number* (Da)

The Péclet number (Pe) is a dimensionless number that measures the relative importance of advection and diffusion, where a large number indicates an advection-dominated transport system and a small number indicates a diffuse flow. Fig. 2 investigated the effect of Pe numbers as well as inlet boundary types on the breakthrough curves at different observation locations. One observation is located upstream of the point source ($x = 4$ m), and the other is located downstream of $x = 10$ m. The solute concentration at $x = 4$ m increased significantly with increasing Péclet number Pe, especially in the case of the Robin inlet boundary condition (Fig. 2a). For transport systems with smaller Pe, the difference in predictions between the Dirichlet and Robin boundary source models was greater. For example, at $t = 2000$ days the relative difference between the predicted concentrations of the two cases for Pe = 20 was 4.4%, while for Pe = 5, this relative difference in prediction could be up to 58.3%. This suggested that the selection of inlet boundary conditions was of particular importance when performing contamination prediction for low permeability sites. For the downstream observation point ($x = 10$ m), the effect of the inlet boundary condition on the breakthrough curve could be seen after about 300 days (Fig. 2b). Although the observation locations were far from the entrance boundary ($x = 10$ m), the effect of the Péclet number on the breakthrough curve is still evident. However, the difference in concentrations at $x = 10$ m predicted by the Dirichlet and Robin boundary source models was relatively smaller compared to the case of a closer observation point ($x = 4$ m). An important reason is that the concentration at the downstream observation point of $x = 10$ m is affected by the coupling of point and boundary sources.

A dimensionless number, Damköhler number (Da), is generally used to indicate the rate of the first-order degradation reaction. The increase in Da caused a significant decrease in solute concentration, which is due to the accelerated consumption by biochemical reaction (Fig. 3). Moreover, for a larger Da, the difference in predictions between the Dirichlet and Robin boundary source models was greater. This illustrated that the prediction model for a strongly degradable system also requires careful selection of entrance boundary conditions.
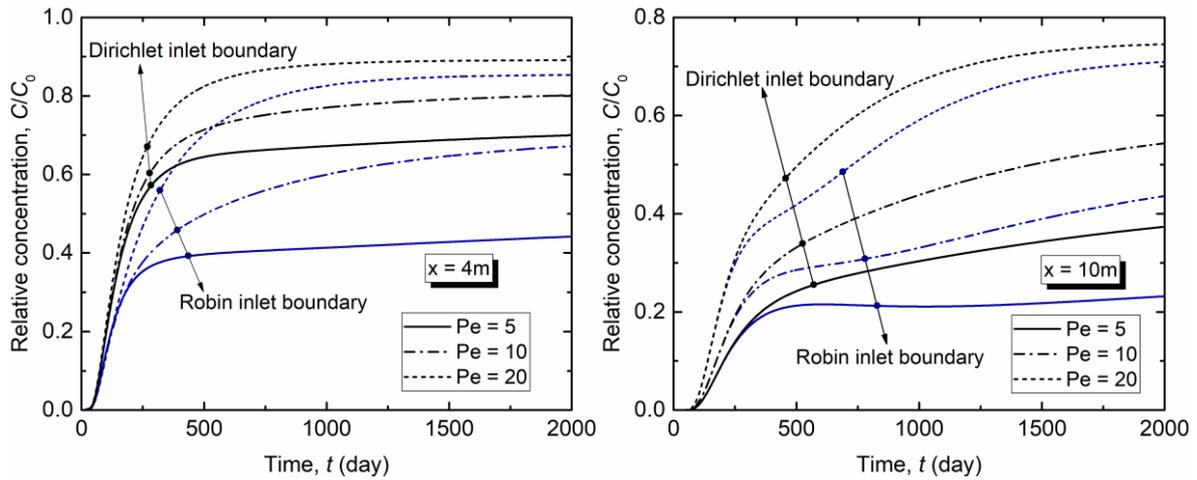
Figure 2. Effect of the Péclet number on the breakthrough curve: (a) the upstream observation point, x = 4 m; (b) the downstream observation point, x = 10 m.
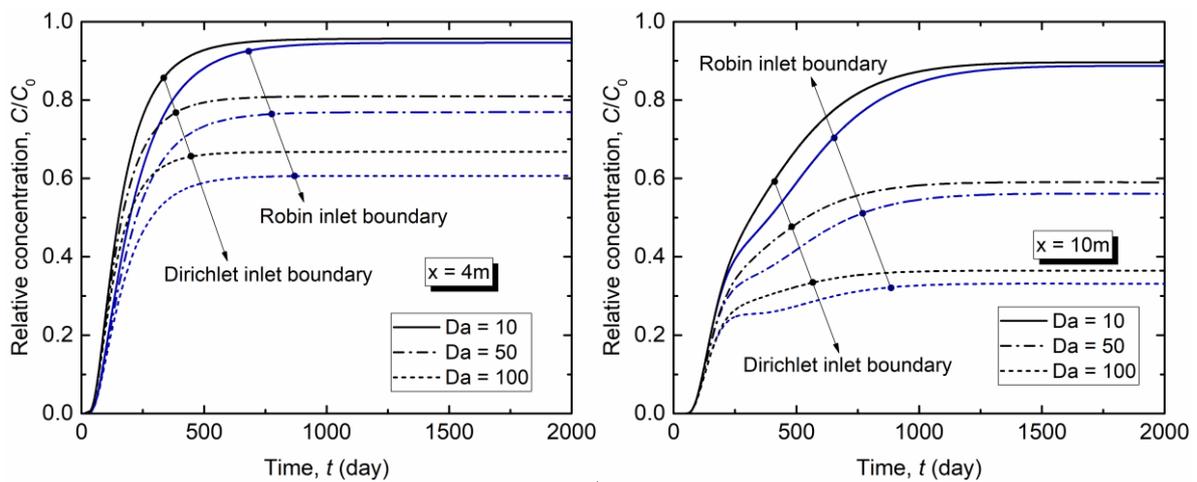
Figure 3. Effect of the Damköhler number on the breakthrough curve: (a) the upstream observation point, x = 4 m; (b) the downstream observation point, x = 10 m.

Conclusion

This study developed a generalized semi-analytical solution for advection-dispersion-reaction equations subject to point-source and boundary-source. Our solution strategy combined Laplace transform, finite Fourier transform, and variable substitution to solve multi-source coupled problems. The derived solutions were tested against numerical results for instantaneous point source scenarios with Dirichlet and Robin inlet boundary and were shown to be accurate and robust. The role of two essential dimensionless parameters was investigated using the proposed solutions. The following main conclusions are obtained:

(1) Point source systems with the Dirichlet boundary condition have a maximum predicted concentration. Predictive models of solute transport subject to both internal point sources and boundary sources should pay attention to the choice of inlet boundary conditions, especially when performing long-term predictions.

(2) For a low permeability system (small Péclet number) or strongly degraded system (large Damköhler number), the boundary source has a significant influence on the solute concentration distribution.

(3) The solutions developed in this paper were programmed into a MATLAB program to facilitate fast calculations. These solutions are mainly used to investigate the forward prediction problem, and they can also be used as a basis for the inverse problem of source zone identification, an essential topic in subsurface transport.

References

[1] Chen, K., Zhan, H. and Zhou, R. (2016) Subsurface solute transport with one-, two-, and three-dimensional arbitrary shape sources, *Journal of Contaminant Hydrology* **190**, 44-57.

[2] Cleary, R. W. and Adrian, D. D. (1973) Analytical Solution of the Convective-Dispersive Equation for Cation Adsorption in Soils, *Soil Science Society of America Journal,* **37**, 197-199.

[3] Van Genuchten, M. T. (1982) *Analytical solutions of the one-dimensional convective-dispersive solute transport equation*, US Department of Agriculture, Agricultural Research Service.

[4] Batu, V. (1989) A generalized two-dimensional analytical solution for hydrodynamic dispersion in bounded media with the first-type boundary condition at the source, *Water Resources Research* **25**, 1125-1132.

[5] Batu, V. (1993) A generalized two-dimensional analytical solute transport model in bounded media for flux-type finite multiple sources, *Water Resources Research* **29**, 2881-2892.

[6] Leij, F. J., Skaggs, T. H. and Van Genuchten, M. T. (1991) Analytical Solutions for Solute Transport in Three-Dimensional Semi-infinite Porous Media, *Water Resources Research* **27**, 2719-2733.

[7] Pérez Guerrero, J. S., Pimentel, L. C. G., Skaggs, T. H. and van Genuchten, M. T. (2009) Analytical solution of the advection–diffusion transport equation using a change-of-variable and integral transform technique, *International Journal of Heat and Mass Transfer* **52**, 3297-3304.

[8] Bear, J. (1972) *Dynamics of Fluids in Porous Media*, New York, USA: Dover Publications.

[9] Basha, H. A. and El-Habel, F. S. (1993) Analytical solution of the one-dimensional time-dependent transport equation, *Water Resources Research* **29**, 3209-3214.

[10] Aral, M. M. and Liao, B. (1996) Analytical Solutions for Two-Dimensional Transport Equation with Time-Dependent Dispersion Coefficients, *Journal of Hydrologic Engineering* **1**, 20-32.

[11] Fedi, A., Massabò, M., Paladino, O. and Cianci, R. (2010) A new analytical solution for the 2D advection-dispersion equation in semi-infinite and laterally bounded domain, *Applied Mathematical Sciences* **4**, 3733-3747.

[12] Ding, X. H., Feng, S. J. and Zheng, Q. T. (2021) A two-dimensional analytical model for contaminant transport in a finite domain subjected to multiple arbitrary time-dependent point injection sources, *Journal of Hydrology* **126318**.

A coupled SPH-DEM model for the simulation of abrasive water-jet impacting solid surface

*Ran Yu, †Xiangwei Dong, and Zengliang Li

College of Mechanical and Electronic Engineering, China University of Petroleum (East China), China

*Presenting author: yr_liquid@163.com
†Corresponding author: dongxiangweiupc@163.com

Abstract

Numerical model of the smoothed particle hydrodynamics method (SPH, a meshfree numerical method), has been approved its great advantages in fluid–particle-solid coupling problems with free surfaces, such as abrasive water-jet (AWJ) impacting process. However, the fully resolved SPH method needs a large amount of computation because of the requirement for fine resolution, which limits its application in practical problems. Coupling of the discrete element method (DEM) and SPH may be a more effective way to achieve the goal. In this study, a coupled SPH-DEM unresolved model is proposed for simulation of AWJ impact. The water-jet and the solid are discretized with a series of SPH particles, and each abrasive is modeled by a DEM particle. Different smoothing lengths are used for SPH-SPH particles and SPH-DEM particles, resulting in a multiple linked-list search method for neighborhood searching. The SPH and DEM particles are coupled through the so-called local averaging technique, in which the interaction forces between the two phases are related to the local porosity. Compared with the fully resolved SPH model, the new coupled model is more efficient, and is suitable for fluid-particle-solid simulation. The process of the single abrasive water-jet impact on the solid is simulated to verify the applicability of the model. The cases of single particle settlement is also involved. Results show that the proposed model can accurately capture the motion of particles in complex fluid flows, and has less computation time cost, which could be useful in the applications of AWJ machining and complex fluid-particle flow with free surfaces.

Keywords: Smoothed particle hydrodynamics; Discrete element method; Abrasive water-jet; Fluid–particle-solid interaction; Free surface flow

1 Introduction

The issue of the fluid–particle flows impacting solid surface is a common concern in several engineering fields, such as coastal, fluvial, and transportation engineering [1]. Abrasive water-jet (AWJ) is a typical fluid–particle flow, which has been widely used in various industries, such as cutting, mining and drilling [2,3]. It involves the interactions between fluid, abrasives, and the solid in free surface flows. Adjustment of various parameters makes the experimental study of AWJ time-consuming and expensive [4]. So the numerical simulation of AWJ impact process can be a valuable complement to the experiments to reveal the fundamental behaviors and predict the solid erosion performance [5].

As both Lagrangian methods, the smoothed particle hydrodynamics method (SPH) has more advantages than the finite element method (FEM) in dealing with large displacement and large deformation problems. In our previous research [6, 7, 8], a fully resolved SPH model for AWJ simulation was proposed and improved. Both fluid, abrasives and solid material were modeled by SPH particles. The water-jet was modeled as a continuous fluid flow, the solid was modeled as elastic–plastic material, and the abrasives were treated as rigid bodies. The model had the advantages of simple concept and strong robustness. The erosion process of the metallic surface by AWJ impact was reproduced. However, the large amount of computation is one of the disadvantages of the previous SPH model. In this study, we propose a coupled SPH-DEM model for AWJ simulation to improve the computational efficiency. Each abrasive is simplified to a single DEM particle, instead of a series of SPH particles. The locally averaged density algorithm based on the local porosity is adopted to simulate the movement of abrasives in water-jet flow. The new proposed model not only realize the detailed interaction among the water-jet, abrasives and solid, but also reduces the number of neighborhood SPH particle pairs, which reduces the computation cost.

The remainder of this study is organized as follows. In Sec.2 and Sec.3, the basic theory of coupled SPH-DEM algorithm is presented, and the modeling process is described. In Sec.4, two cases of single particle settlement and single abrasive water-jet impact are presented to prove the validity of the coupled model. In Sec.5, the conclusions of the study are summarized.

2 Formulations for SPH model (Fluid and solid phase)

This section summarizes some fundamental parts of SPH model based on the local averaging technique.

2.1 Basic theory of SPH

In SPH model, materials in the computational domain are discretized by a set of particles, which carry field variables and material properties, such as velocity, density, stress, etc [9]. When a SPH particle is within another SPH particle's support domain Ω, those two particles interact with each other and move according to the governing equations, as shown in Fig.1. Each particle moves according to its own acceleration. Therefore, this method is not limited by the mesh factors and is suitable for large deformation simulation [10].
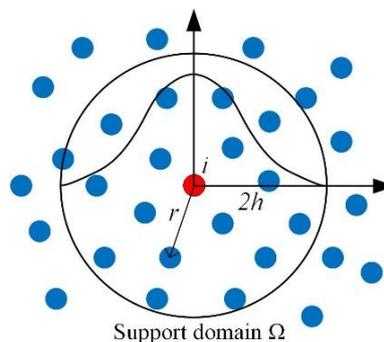
Figure 1. Kernel approximation in SPH method

There are two main steps for the SPH model establishment. The first step is the integral representation of field functions (kernel approximation). The value of the field function $f(\boldsymbol{x})$ can be approximated as the integral representation of $\boldsymbol{x'}$ in the support domain Ω of $\boldsymbol{x}$ [11]. The

second step is the particle approximation, which discretize the continuous integral function into the sum of the finite particles located in the support domain Ω of $\boldsymbol{x}$. As show in follows:

$$\langle f(\boldsymbol{x}_i) \rangle = \sum_{j=1}^{N} \frac{m_j}{\rho_j} f(\boldsymbol{x}_j) \cdot W(\boldsymbol{x}_i - \boldsymbol{x}_j, h), \tag{1}$$

$$\left\langle \frac{\partial f(\boldsymbol{x}_i)}{\partial \boldsymbol{x}} \right\rangle = \sum_{j=1}^{N} \frac{m_j}{\rho_j} f(\boldsymbol{x}_j) \cdot \frac{\partial W_{ij}}{\partial \boldsymbol{x}_i}, \tag{2}$$

where $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ in Eq. (1) and Eq. (2) are the position vectors of particle i and j, respectively. Particle j represents the SPH particle which is located in the support domain Ω of the particle i. m_j and ρ_j are the mass and the density of the particle j, respectively. N is the total number of the particles within the support domain Ω of particle i. The kernel gradient $\frac{\partial W_{ij}}{\partial \boldsymbol{x}_i}$ can be expressed as $\frac{\partial W_{ij}}{\partial \boldsymbol{x}_i} = \frac{x_i - x_i}{r_{ij}} \frac{\partial W_{ij}}{\partial r_{ij}} = \frac{x_{ij}}{r_{ij}} \frac{\partial W_{ij}}{\partial r_{ij}}$, where r_{ij} is the distance between particle i and j.

There are many available smoothing functions for SPH model. The cubic spline function, which was proposed by Monaghan and Lattanzio [12], is used in this study:

$$W(\boldsymbol{x}_i - \boldsymbol{x}_j, h) = W_{ij} = a_d \times \begin{cases} \dfrac{2}{3} - q^2 + \dfrac{1}{2}q^3, & 0 \le q < 1 \\ \dfrac{1}{6}(2-q)^3, & 1 \le q < 2 \\ 0, & q \ge 2 \end{cases} \tag{3}$$

where q is the relative distance between the particle i and j ($q=r_{ij}/h$), a_d is the normalization constant which is expressed as $a_d = 15/(7\pi h^2)$ for the two-dimensional simulation.

2.2 Local averaging technique for SPH model (Fluid phase)

In this paper, the fluid and the solid phase are treated as continuous media. To calculate the coupling force between SPH and DEM particles, a local averaging technique is introduced. The concept was established by Anderson and Jackson [13] to deal with the momentum exchange and balance between different phases. For ease of reading, the SPH particles are labeled as particle a and b while the DEM particles are labeled as particle i and j. For the fluid SPH particle a, the locally averaged fluid density $\bar{\rho}_a$ is shown as:

$$\bar{\rho}_a = \varepsilon_a \rho_f, \tag{4}$$

where ε_a is the local porosity of the fluid particle a, and ρ_f is the actual fluid density. The local porosity ε_a depends on the volume fraction of nearby DEM particles smoothed by the kernel function, as shown in follows [14]:

$$\varepsilon_a = 1 - \sum_{j} W_{aj}(h_c) V_j, \tag{5}$$

where $W_{aj}(h_c)$ is the SPH kernel function and h_c is the coupling smoothing length between SPH and DEM particles. h_c should be larger than the diameter of DEM particle but small enough to capture local feature of the porosity field. V_j is the volume of DEM particle.

For fluid SPH particles, the conservation equations of mass and momentum based on local averaging technique are expressed as:

$$\frac{\mathrm{d}\overline{\rho}_a}{\mathrm{d}t} = \overline{\rho}_a \sum_{b=1}^{N} \frac{m_b}{\overline{\rho}_b} (v_a^\beta - v_b^\beta) \cdot \frac{\partial W_{ab}}{\partial x_a^\beta}, \tag{6}$$

$$\frac{\mathrm{d}v_a^\alpha}{\mathrm{d}t} = \sum_{b=1}^{N} m_b \left[-\left(\frac{P_a}{\overline{\rho}_a^2} + \frac{P_b}{\overline{\rho}_b^2} \right) \delta^{\alpha\beta} + \Pi_{ab}^{visc} - \pi_{ab}^{art} \delta^{\alpha\beta} \right] \cdot \frac{\partial W_{ab}}{\partial x_a^\beta} + f_a^\alpha, \tag{7}$$

where α and β are the Cartesian coordinates x and y, and t represents the time. $\delta^{\alpha\beta}$ is the Kronecker tensor (if $\alpha = \beta$, $\delta^{\alpha\beta} = 1$, otherwise, $\delta^{\alpha\beta} = 0$). f_a^α is the external force of the particle a, such as gravity or coupling forces. The first term in Eq.(7) is the pressure term. The second term (Π_{ab}^{visc}) is the dissipative force, which is treated as the viscosity force in Newtonian fluids [15]. The third term (π_{ab}^{art}) is the artificial viscosity term, which is proposed by Monaghan to reduce unphysical spurious oscillations and improve the numerical stability.

For fluid SPH particles, the pressure P is a function of the actual density ρ, which is computed by the eqution of state (EOS). A Mie-Grüneisen form of the EOS for fluid particles is shown as [16]:

$$P = \frac{\rho_0 c_0^2 \eta \left[1 + \left(1 - \frac{\Gamma_0}{2} \right) \eta - \frac{a}{2} \eta^2 \right]}{\left[1 - (S_1 - 1)\eta - S_2 \frac{\eta^2}{\eta+1} - S_3 \frac{\eta^2}{(\eta+1)^2} \right]^2} + (\Gamma_0 + a\eta)e, \tag{8}$$

where $\eta = \rho/\rho_0$, ρ_0 is the reference density and ρ is the actual density. e is the internal energy per unit of mass. Table 1 lists the EOS parameters of fluid phase.

Table 1. EOS parameters for fluid phase[16]

Parameters	Value
Reference density	$\rho_0 = 1000 \mathrm{kg/m^3}$
Velocity of sound	$c_0 = 1480 \mathrm{m/s}$
Grüneisen gamma	$\Gamma_0 = 0.5$
Volume correction coefficient	$a = 0$
Coefficient	$S_1 = 2.56$
Coefficient	$S_2 = 1.98$
Coefficient	$S_3 = 1.23$

2.3 Formulations for SPH model (Solid phase)

In this study, the solid phase is modeled as a elastic–plastic material to investigate the erosion process by AWJ impact. Similar to Section 2.2, the conservation equations for solid SPH particles are written as:

$$\frac{\mathrm{d}\rho_a}{\mathrm{d}t} = \rho_a \sum_{b=1}^{N} \frac{m_b}{\rho_b} (v_a^\beta - v_b^\beta) \cdot \frac{\partial W_{ab}}{\partial x_a^\beta}, \tag{9}$$

$$\frac{\mathrm{d}v_a^\alpha}{\mathrm{d}t} = \sum_{b=1}^{N} m_b \left[-\left(\frac{P_a}{\rho_a^2} + \frac{P_b}{\rho_b^2} \right) \delta^{\alpha\beta} + \frac{\tau_a^{\alpha\beta} + \tau_b^{\alpha\beta}}{\rho_a \rho_b} - \Pi_{ab} \delta^{\alpha\beta} \right] \cdot \frac{\partial W_{ab}}{\partial x_a^\beta} + f_a^\alpha, \tag{10}$$

where the second term in Eq.(10) is the shear force term. $\tau_a^{\alpha\beta}$ is the deviatoric stress of the particle a. f_a^α is the external force, such as gravity or contact force with DEM particles.

In the elastic-plasticity mechanics, the deviatoric stress rate $\dot{\tau}_a^{\alpha\beta}$ is a function of the strain rate tensor $\varepsilon_a^{\alpha\beta}$ and the rotation rate tensor $R_a^{\alpha\beta}$. The incremental formulation with the Jaumann rate correction is shown as follows [11]:

$$\dot{\tau}_a^{\alpha\beta} = \frac{d\tau_a^{\alpha\beta}}{dt} = 2G\left(\varepsilon_a^{\alpha\beta} - \frac{1}{3}\delta^{\alpha\beta}\varepsilon_a^{\gamma\gamma}\right) + \tau_a^{\alpha\gamma}\cdot R_a^{\beta\gamma} + \tau_a^{\gamma\beta}\cdot R_a^{\alpha\gamma},$$ (11)

where G is the shear modulus. $\varepsilon_a^{\alpha\beta}$ is the strain rate tensor and $R_a^{\alpha\beta}$ is the rotation rate tensor, respectively.

The Johnson–Cook constitutive model (J-C) is selected to describe the plastic deformation of OFHC copper [17], which is numerically robust and easily implemented in the SPH formulations. The yield stress σ_y in J-C model is written as:

$$\sigma_y = \left[A + B\left(\varepsilon_{eff}^p\right)^N\right]\left[1 + C\ln\left(\frac{\dot{\varepsilon}_{eff}^p}{\dot{\varepsilon}_0}\right)\right],$$ (12)

where ε_{eff}^p is the equivalent plastic strain, $\dot{\varepsilon}_{eff}^p$ is the equivalent plastic strain rate, and $\dot{\varepsilon}_0$ is the reference strain rate. A, B, C and N are material dependent constants.

The oxygen-free high-thermal-conductivity (OFHC) copper is selected as the solid phase material. The Mie-Grüneisen EOS equation for OFHC copper is employed as [18]:

$$P = \frac{\rho_0 c_0^2 \eta\left[1 + \left(1 - \frac{\Gamma_0}{2}\right)\eta\right]}{\left[1 - (S_a - 1)\eta\right]^2} + \rho_0\Gamma_0 e,$$ (13)

where S_a is a linear Hugoniot slope coefficient, the EOS parameters for OFHC copper are shown in Table 2.

Table 2. EOS parameters for OFHC copper[18]

Parameters	Value
Reference density	ρ_0=8960kg/m^3
Velocity of sound	c_0=3940m/s
Grüneisen gamma	Γ_0=1.99
Linear Hugoniot slope coefficient	S_a=1.5

3 Formulations for DEM model (Abrasive phase) and phase coupling

DEM is a Lagrangian method proposed by Cundall [19] to study discontinuous mechanical efforts of rock by assemblies of discs (2D) or spheres (3D). Each abrasive is simplified to a single DEM particle. Contact forces occur when the particles overlap. The abrasive and fluid phase are coupled by local averaging algorithm based on porosity, which can be used to simulate the motion of abrasives in free surface flow.

3.1 DEM governing equations

The basic governing equations of DEM particles follow Newton's second law. In this study, the forces acting on the DEM particle i are listed as:

$$m_i\frac{d\boldsymbol{v}_i}{dt} = \sum_j \boldsymbol{F}_{ij}^c + \boldsymbol{F}_{fa} + \boldsymbol{F}_{sa}^c + m_i\boldsymbol{g},$$ (14)

where m_i and v_i are the mass and velocity of DEM particle i, respectively. F_{ij}^c is the contact force for abrasive-abrasive interaction, and j represents other DEM particles contact with particle i. F_{fa} is the coupling force with fluid SPH particle, including drag force and buoyancy. F_{sa}^c is the contact force with solid SPH particles. For rotational motion, the angular acceleration of the DEM particle i is expressed as:

$$I_i \frac{d\boldsymbol{\omega}_i}{dt} = \sum_j \boldsymbol{T}_{ij} , \tag{15}$$

where I_i, ω_i, and T_{ij} are the moment of inertia, angular velocity, and torque of contact forces.

3.2 Contact force for abrasive-abrasive interaction

The details of contact force of DEM have been described in many literatures. To simulate interaction and rotation behavior among abrasives in water-jet flow, the soft-sphere contact force model is adopted in this study. The normal and tangential contact forces for abrasive-abrasive interaction are determined from the particle overlap by a spring-dashpot model [20]. The schematic illustration is shown in Fig.2.

(a) Normal contact force   (b) tangential contact force
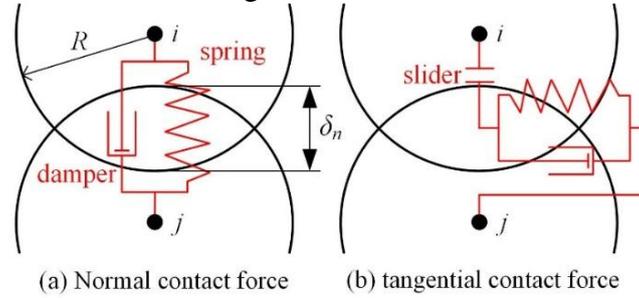
Figure 2. Schematic illustration of soft-sphere contact force model

The abrasives are treated as rigid bodies and the surface deformation is ignored. The soft-sphere model is suitable for numerical simulation of engineering problems. The contact force for the DEM particle i is the sum of normal and tangential forces, as follows:

$$\boldsymbol{F}_{ij}^c = \boldsymbol{F}_{ij}^n + \boldsymbol{F}_{ij}^t , \tag{16}$$

where the superscript n and t denote normal and tangential forces, respectively. The normal contact force is given by

$$\boldsymbol{F}_{ij}^n = -k_n \delta_n \boldsymbol{n}_{ij} - d_n \boldsymbol{v}_{ij}^n , \tag{17}$$

where k_n and d_n denote the normal stiffness and damping coefficient, respectively. δ_n is the normal overlap size between DEM particle i and j, as shown in Fig.2. $\boldsymbol{n}_{ij}$ is the unit normal vector, from i to j. $\boldsymbol{v}_{ij}^n$ is the normal relative velocity, which is determined by the relative velocity $\boldsymbol{v}_{ij}$, as shown in follows:

$$\boldsymbol{v}_{ij} = \boldsymbol{v}_i - \boldsymbol{v}_j + (R_i\boldsymbol{\omega}_i + R_j\boldsymbol{\omega}_j) \times \boldsymbol{n}_{ij} , \tag{18}$$

$$\boldsymbol{v}_{ij}^n = (\boldsymbol{v}_{ij} \cdot \boldsymbol{n}_{ij})\boldsymbol{n}_{ij} , \tag{19}$$

where R and ω are the radius and angular velocity of DEM particle, respectively.

Similar to normal force, the tangential contact force is written as:

$$\boldsymbol{F}_{ij}^{t} = -k_{t}\delta_{t}\boldsymbol{t}_{ij} - d_{t}\boldsymbol{v}_{ij}^{t}, \tag{20}$$

where k_t and d_t denote the tangential stiffness and damping coefficient, respectively. δ_t is the normal overlap size between DEM particle i and j. $\boldsymbol{t}_{ij}$ is the unit tangential vector, $\boldsymbol{t}_{ij} = \dfrac{\boldsymbol{v}_{ij}^{t}}{\left|\boldsymbol{v}_{ij}^{t}\right|}$. $\boldsymbol{v}_{ij}^{t}$ is the tangential relative velocity, $\boldsymbol{v}_{ij}^{t} = \boldsymbol{v}_{ij} - \boldsymbol{v}_{ij}^{n}$.

In addition, the maximum tangential force is limited by the slip condition:

$$\boldsymbol{F}_{ij\,\max}^{t} = \mu\left|\boldsymbol{F}_{ij}^{n}\right|\boldsymbol{t}_{ij}, \tag{21}$$

where μ is the friction coefficient at the contact.

The contact torque $\boldsymbol{T}_{ij}$ shown in Eq.(15) is determined by the tangential contact force $\boldsymbol{F}_{ij}^{t}$:

$$\boldsymbol{T}_{ij} = \boldsymbol{F}_{ij}^{t} \times (\boldsymbol{x}_{i} - \boldsymbol{x}_{p}), \tag{22}$$

where $\boldsymbol{x}_i$ is the center of gravity of the DEM particle i. $\boldsymbol{x}_p$ is the position of the contact point, which is on the line between particle i and j, and the distance from i is R.

3.3 Contact force for solid-abrasive interaction

The contact force $\boldsymbol{F}_{sa}^{c}$ for the solid-abrasive interaction is based on the penalty algorithm [21], as shown in Fig.3. The solid material is modeled as a elastic–plastic material (as shown in Sec.2.3), and the abrasives are modeled as rigid particles. When the distance between the solid SPH particle and the abrasive DEM particle is within the threshold (in this study, the threshold is set to $R+d_{ini}$, where R is the DEM particle's radius and d_{ini} is the initial spacing of two adjacent SPH particles), the contact force $\boldsymbol{F}_{sa}^{c}$ is generated. $\boldsymbol{F}_{sa}^{c}$ can be decomposed into the normal force $\boldsymbol{F}_{sa}^{n}$ and tangential force $\boldsymbol{F}_{sa}^{t}$.

$\boldsymbol{F}_{sa}^{n}$ and $\boldsymbol{F}_{sa}^{t}$ are expressed as:

$$\begin{cases} \boldsymbol{F}_{sa}^{n} = (1-\chi)\left[\dfrac{2m_{a}}{(\Delta t)^{2}}\left(R + d_{ini} - d_{p}\right)\right]\boldsymbol{n}_{p} \\ \boldsymbol{F}_{sa}^{t} = \dfrac{2m_{a}}{\Delta t}\left(\boldsymbol{v}_{pi} \cdot \boldsymbol{\tau}_{p}\right)\boldsymbol{\tau}_{p} \end{cases}, \tag{23}$$

where χ is the index of the penetration, if $\chi = 0$ means no penetration is allowed. m_a is the mass of particle a, and Δt is the time step. $\boldsymbol{v}_{pi} = \boldsymbol{v}_p - \boldsymbol{v}_i$, where $\boldsymbol{v}_p$ is the velocity vector at point p.

$\bigcirc$ Abrasive DEM particle i
$\bullet$ Solid SPH particle a
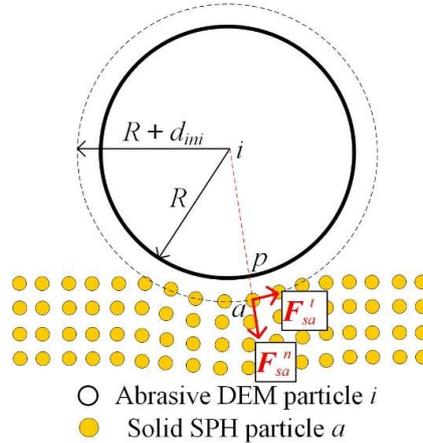
Figure 3. Illustration of solid-abrasive interaction

3.4 Coupling force for fluid-abrasive interaction

For the abrasive DEM particle i, coupling force $\boldsymbol{F}_{fa}$ due to fluid flow is modelled, which can be split into a hydrodynamic force and a drag force [22]:

$$\boldsymbol{F}_{fa} = V_i(-\nabla P + \nabla \cdot \tau)_i + \boldsymbol{f}_d(\varepsilon_i, \boldsymbol{u}_i), \tag{24}$$

where V_i is the volume of DEM particle i. For the hydrodynamic force, ∇P is the pressure gradient, and $\nabla \cdot \tau$ is the viscosity force. Drag force $\boldsymbol{f}_d$ depends on the local porosity ε_i and relative velocity $\boldsymbol{u}_i$ between fluid and abrasive.

The hydrodynamic force is evaluated at particle i using a Shepard corrected SPH interpolation, given as:

$$V_i(-\nabla P + \nabla \cdot \tau)_i = \frac{1}{\sum\limits_a \frac{m_a}{\rho_a} W_{ia}(h_c)} \sum_a m_a \theta_a W_{ia}(h_c), \tag{25}$$

$$\theta_a = \sum_b m_b \left[-\left(\frac{P_a}{\overline{\rho}_a^2} + \frac{P_b}{\overline{\rho}_b^2} \right) + \Pi_{ab}^{visc} \right] \cdot \nabla_a W_{ab}(h), \tag{26}$$

where the subscript a is the fluid SPH particle nearby the DEM particle i. The subscript b is the fluid SPH particle nearby the particle a. h is the smoothing length among SPH particles, and h_c is the coupling smoothing length between SPH and DEM particles.

The drag force $\boldsymbol{f}_d$ is a function of local porosity ε_i and relative velocity $\boldsymbol{u}_i$. The local porosity ε_i at the position of DEM particle i is estimated by smoothing the nearby values of SPH particles: The expression for the drag force $\boldsymbol{f}_d$ is shown below:

$$\boldsymbol{f}_d = \frac{\beta_i}{1-\varepsilon_i} V_i \boldsymbol{u}_i, \tag{27}$$

where β_i is the interphase momentum transfer coefficient, which is given by:

$$\beta_i = \begin{cases} 150\dfrac{(1-\varepsilon_i)^2}{\varepsilon_i}\dfrac{\mu_f}{d_i^2} + 1.75(1-\varepsilon_i)\dfrac{\rho_f}{d_i}|\boldsymbol{u}_i|, & \varepsilon_i \leqslant 0.8 \\[4mm] 0.75 C_d \dfrac{\varepsilon_i(1-\varepsilon_i)}{d_i}\rho_f |\boldsymbol{u}_i|\varepsilon_i^{-2.65}, & \varepsilon_i > 0.8 \end{cases}, \tag{28}$$

where μ_f, ρ_f are the viscosity, reference density of the fluid, respectively. C_d, d_i are the drag coefficient, diameter for DEM particle i, respectively.

The coupling force on fluid SPH particle a is calculated by a weighted average fluid-abrasive coupling force $\boldsymbol{F}_{fa}$ acting on the DEM particles nearby within the coupling length h_c. The contribution of each DEM particle to this average is scaled by the value of the SPH kernel function:

$$\boldsymbol{f}_a = -\frac{m_a}{\overline{\rho}_a} \sum_i \frac{1}{S_i} \boldsymbol{F}_{fa} W_{ai}(h_c), \tag{29}$$

where $\boldsymbol{F}_{fa}$ is the coupling force acting on DEM particles in Eq.(24). The scaling factor S_i is calculated to ensure that the force acting on the fluid particles is balanced with the force on the DEM particle, which is given by:

$$S_i = \sum_b \frac{m_b}{\rho_b} W_{bi}(h_c) , \tag{30}$$

where the subscript b is the fluid SPH particle nearby the DEM particle i within the coupling length h_c.

4 Validations of the coupled SPH-DEM model

According to the formulations presented above, the SPH-DEM 2D numerical model is implemented by a Fortran code in this study. In this section, two numerical cases are proposed to verify the coupled model correctly handle the interactions among different phases. Meanwhile, compared with the fully resolved SPH model [6~8], this model has the advantage of less computation and higher computational efficiency.

4.1 Case 1: single particle sedimentation

The first case simulates the process of a single particle sedimentation in a fluid domain by gravity. The results are compared with the data in Reference [23] to verify the coupled model can simulate the coupling force between DEM and SPH phases.

Computational domain of the single particle sedimentation is shown in Fig.4. The water area is 0.04×0.06m, and the initial height for the single DEM particle is 0.04m. The gravity acts in the negative Y direction. Other detailed parameters are listed in Table 3.
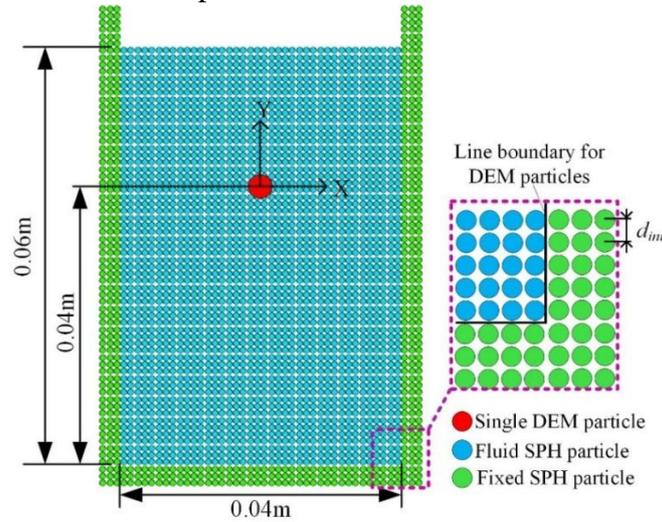
Figure 4. SPH-DEM Coupled model for single particle sedimentation

Table 3. Parameters for single particle sedimentation

Parameters	Value
Initial density for SPH particle, ρ_0	$1.0 \times 10^3 \text{kg/m}^3$
Viscocity for SPH particle, μ_f	$1.0 \times 10^{-2} \text{Pa·s}$
Initial spacing for SPH particle, d_{ini}	$1.0 \times 10^{-3} \text{m}$
Smoothing length for SPH-SPH, h	$1.25 \times d_{ini}$
Density for DEM particle, ρ_{dem}	$1.25 \times 10^3 \text{kg/m}^3$
Diameter for DEM particle, d	$2.5 \times 10^{-3} \text{m}$
Coupling smoothing length for SPH-DEM, h_c	$2.5 \times d$
Time step, Δt	$2.5 \times 10^{-7} \text{s}$

Boundary treatments for SPH and DEM are separately. For SPH particles, three layers of particles are fixed on the boundary to prevent fluid particles from penetrating. The velocity and

acceleration of the boundary particles are fixed at zero, while other parameters (initial density, pressure, etc.) evolve through the kernel function as fluid particles do. A line boundary for DEM is placed at the boundary of water area, as shown in Fig.4. When the DEM particle's centroid is within the radius (radius of DEM particle) from the boundary. The particle is subjected to spring and damping forces from normal and tangential directions, as mentioned in Sec.3.2.

The DEM particle is released at $t = 0$s, moves along the negative Y direction under the action of gravity and coupling force, and finally stops at the bottom boundary. Fig.5 shows the time history of the DEM particle's velocity in Y direction. The simulation results are compared with the data in Reference [23]. The two processes of sedimentation are largely the same.

There are 3 different places. The first place is found at $0 < t < 0.20$s when the particle accelerates down, which is caused by the drag force term f_d (Eq.(27)). Drag force increases with the relative speed between fluid and particle, so the acceleration is not a linear process. The second place occurs at $0.60 < t < 0.74$s. When the DEM particle approaching the bottom boundary, the local fluid pressure at the bottom increases, resulting in the hydrodynamic force increase (Eq.(25)), which causes the particle to decelerate. The third place is at $t = 0.74$s when the particle contacts the bottom boundary. The particle rebounds by the contact force of spring-dashpot model, and finally stops at the bottom boundary. The results show that the coupled model can simulate the free movement of DEM particles in fluid phase.
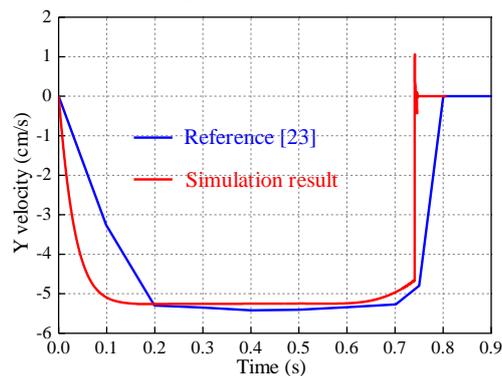
Figure 5. Time history of the velocity in Y direction

4.3 Case 2: high speed water-jet flow containing a single circular abrasive

The second case investigates the erosion process of solid phase by high speed water-jet impact. The computational domain is shown in Fig.6. The 2D numerical model is simplified, and the water-jet contains only one circular abrasive. The OFHC copper is set as the solid phase material with a size of 2.40×8.04mm. The water-jet diameter (d_{jet}) is 1.02mm, impacts the solid vertically at a speed of 200m/s. Water-jet SPH particles enter computational domain periodically from the inlet, and disappear at the outlet on both sides. These measures keep the particle number within a certain range and improve the computational efficiency.

Two types of rigid abrasive model are compared in this section. For Type 1 model, the circular abrasive is modeled by a single DEM particle. In Type 2 model, the abrasive is discretized with a series of SPH particles, as shown in Fig.6. The density and pressure evolution between abrasive SPH particles and water-jet particles are carried out by the kernel function with the smooth length h, and the contact force model for solid-abrasive interaction is the same as Eq.(23). Other details of Type 2 model are discussed in our earlier research [8]. The parameters for the single water-jet impact are listed in Table 5.
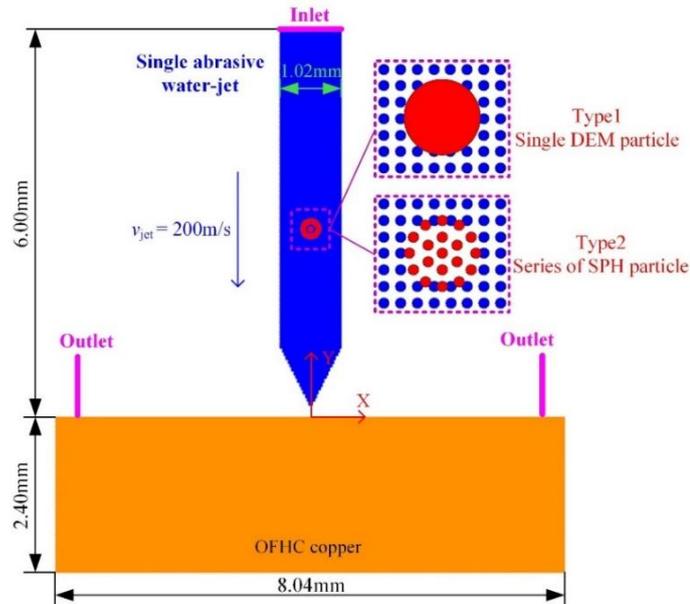
Figure 6. Single abrasive water-jet impact using 2 types of abrasive model

Table 5. Parameters for single abrasive water-jet impact

Parameters	Value
Initial density for water-jet SPH particle, ρ_0	$1.0 \times 10^3 \text{kg/m}^3$
Viscocity for water-jet SPH particle, μ_f	$1.0 \times 10^{-3} \text{Pa·s}$
Initial spacing for SPH particle, d_{ini}	0.06mm
Smoothing length for SPH-SPH, h	$1.25 \times d_{ini}$
Density for circular abrasive, ρ_{ab}	$7.8 \times 10^3 \text{kg/m}^3$
Diameter for circular abrasive, d_{ab}	0.24mm
Coupling smoothing length for SPH-DEM, h_c	$2.0 \times d$
Time step, Δt	$2.0 \times 10^{-9} \text{s}$

The simulation is conducted on a 16-core PC (E5-2667,3.20GHz), running about 6,0000 time steps with a corresponding physical time of 120μs. Fig.7 shows the evolution of the single abrasive water-jet impact. Type 1 single DEM particle model is shown in Fig.7(a~d), and Type 2 SPH particle model is shown in Fig.7(e~h). In these two models, the initial velocity of the abrasive is 200m/s. Under the action of fluid-abrasive interaction force, the abrasive impacts the solid SPH particles at a velocity of 170m/s at $t = 30$μs, finally flows to the side at $t = 120$μs.

(a) 15μs

(e) 15μs

(b) 30μs

(f) 30μs
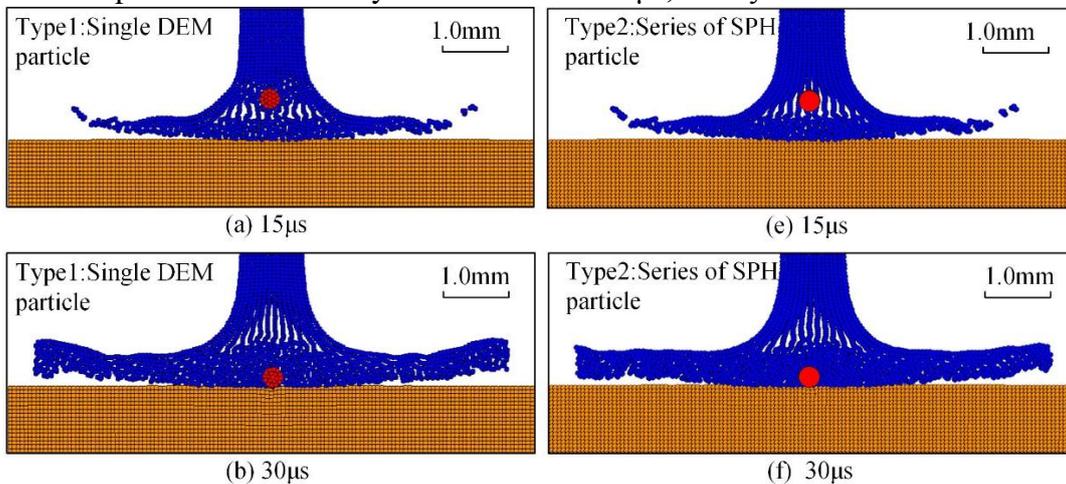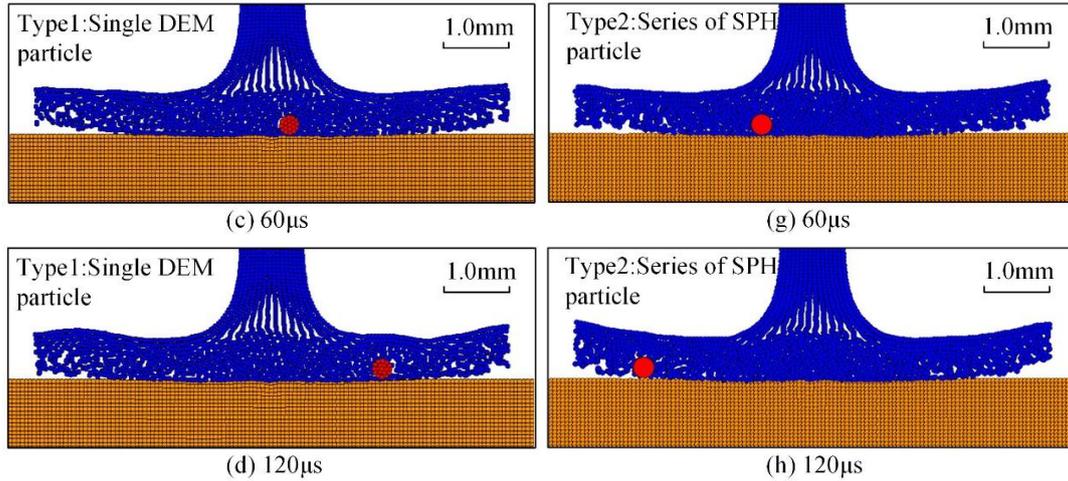
Figure 7. Evolution of single abrasive water-jet impact using 2 types of abrasive model

Surface morphology of OFHC copper by water-jet impact at $t = 120\mu s$ is shown in Fig.8. The single abrasive impact solid at the center, causing plastic deformation on the surface, and leaving a small crater. The plastic strain distribution in both two types is basically the same, and maximum plastic strain is about 0.13. It can be concluded from Fig.7 and Fig.8 that the simulation results of the newly proposed coupled SPH-DEM model (Type 1) for single abrasive water-jet impact are consistent with the fully resolved SPH model (Type 2) in our previous study [8]. The main difference is the computation time. For the coupled SPH-DEM model, it takes about 840s to run 6,0000 time steps. While for the fully SPH model, the computation time cost is 894s. The proposed model saves 6% of computation time for the simulation of single abrasive water-jet impact.
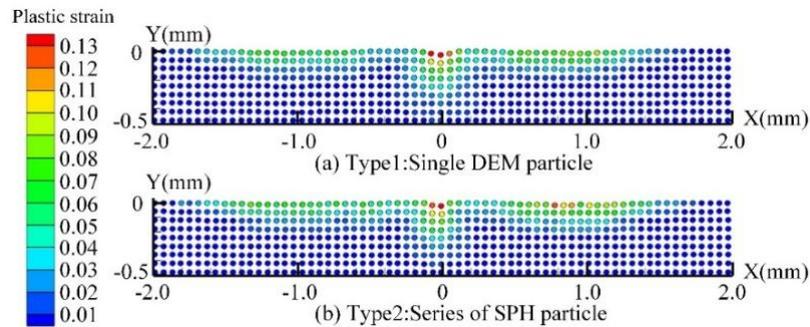
Figure 8. Surface morphology of OFHC copper at $t = 120\mu s$

For the meshfree particle method, the field variables of the particles (such as density, acceleration, stress, etc.) are calculated in pairs and accumulated. The most time-consuming part of the calculation is the neighborhood particle pair search. The link-list search method is adopted in this study. The information of two particles that are within 2 times the smoothing length is stored in memory in pairs for subsequent calculations. Obviously, the greater the number of particle pairs, the greater the amount of computation during the particle search, and the more time-consuming the computation will be. The number of particle pairs in two types (Niac1 for Type 1, and Niac2 for Type 2) are counted. Subtract these two numbers, as shown in Fig.9. Most of the time steps during the simulation, the Niac2−Niac1 value is greater than 0, which indicates that the coupled SPH-DEM model (Type 1) has fewer number of particle pairs. This may be the main reason for the less computation time cost of Type 1 model.
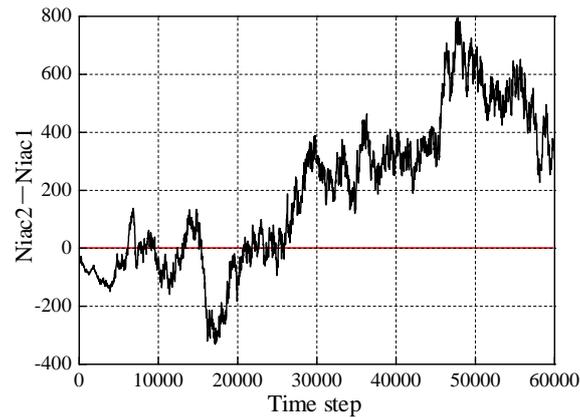
Figure 9. Difference of particle pair number in two types

It should be emphasized that the abrasive water-jet impact simulation in this study contains only one abrasive. If the water-jet contains multiple abrasives, the value of Niac2−Niac1 will be further increased, and the advantage of the coupled SPH-DEM model in less computation time cost will be more obvious, which will be involved in our further study.

5 Conclusions

In this study, a coupled SPH-DEM numerical model for AWJ impact simulation is proposed. The fluid and solid phases are discretized with a series of SPH particles, and the abrasive is modeled by a single DEM particle. Two numerical cases are carried out with the new model. The simulation results of single particle sedimentation are basically consistent with the reference. Compared with the fully resolved SPH model in our earlier research, the coupled model has less computation and higher numerical efficiency in single abrasive water-jet impact simulation. We believe that it has better effect in dealing with multiple abrasive water-jet impact. The present SPH-DEM model can reasonably describe the features of fluid–particle-solid coupling under free surface flow conditions.

References

[1] Canelas R B, Crespo A, JM Domínguez, et al. SPH–DCDEM model for arbitrary geometries in free surface solid–fluid flows[J]. Computer Physics Communications, 2016:131-140.

[2] Liu H T. Waterjet technology for machining fine features pertaining to micromachining[J]. Journal of Manufacturing Processes, 2010, 12(1):8-18.

[3] Tirumala D, Gajjela R, Das R. ANN and RSM approach for modelling and multi objective optimization of abrasive water jet machining process[J]. Decision Science Letters, 2018, 7(4): 535-548.

[4] Nyaboro J, Ahmed M, El-Hofy H, et al. Experimental and numerical investigation of the abrasive waterjet machining of aluminum-7075-T6 for aerospace applications[J]. Advances in Manufacturing, 2021, 9(2): 286-303.

[5] Kurnenkov A , Shurigin A , Glebov V . Investigation of the process of abrasive waterjet cutting of steels based on numerical simulation[J]. MATEC Web of Conferences, 2019, 298(2):00103.

[6] Dong X W, Li Z L, Jiang C , et al. Smoothed particle hydrodynamics (SPH) simulation of impinging jet flows containing abrasive rigid bodies[J]. Computational Particle Mechanics, 2019, 6(3): 479-501.

[7] Feng L, Liu G R, Li Z L, et al. Study on the effects of abrasive particle shape on the cutting performance of Ti-6Al-4V materials based on the SPH method[J]. The International Journal of Advanced Manufacturing Technology, 2019, 101(9): 3167-3182.

[8]   Yu R, Dong X, Du M, et al. Improved smoothed particle hydrodynamics (SPH) model for simulation of abrasive water-jet (AWJ)[J]. International Journal of Computational Methods, 2022: 2143002.

[9]   Takaffoli M, Papini M. Material deformation and removal due to single particle impacts on ductile materials using smoothed particle hydrodynamics[J]. Wear, 2012, 274: 50-59.

[10] Mao Z, Liu G R, Dong X W. A comprehensive study on the parameters setting in smoothed particle hydrodynamics (SPH) method applied to hydrodynamics problems[J]. Computers and Geotechnics, 2017, 92: 77-95.

[11] Liu G R, Liu M B. Smoothed particle hydrodynamics: a meshfree particle method[M]. World scientific, 2003.

[12] Monaghan J J, Lattanzio J C. A refined particle method for astrophysical problems[J]. Astronomy and astrophysics, 1985, 149: 135-143.

[13] T.B. Anderson, R. Jackson, Fluid mechanical description of fluidized beds. Equations of motion, Industrial and Engineering Chemistry Fundamentals 6(1967) 527–539.

[14] He Y, Bayly A E, Hassanpour A, et al. A GPU-based coupled SPH-DEM method for particle-fluid flow with free surfaces[J]. Powder technology, 2018, 338: 548-562.

[15] E.Y.M. Lo, S. Shao, Simulation of near-shore solitary wave mechanics by an incompressible SPH method, Applied Ocean Research 24 (2002) 275–286.

[16] Liu X H, Liu S Y, Ji H F. Numerical research on rock breaking performance of water jet based on SPH[J]. Powder Technology, 2015:181-192.

[17] Johnson G R, Cook W H. Fracture characteristics of three metals subjected to various strains, strain rates, temperatures and pressures[J]. Engineering fracture mechanics, 1985, 21(1): 31-48.

[18] Randles P W, Libersky L D. Smoothed particle hydrodynamics: some recent improvements and applications[J]. Computer methods in applied mechanics and engineering, 1996, 139(1-4): 375-408.

[19] Cundall PA, Strack ODL. A discrete numerical model for granular assemblies. Geotechnique 1979;29:47–65

[20] Norouzi H R, Zarghami R, Sotudeh-Gharebagh R, et al. Coupled CFD-DEM modeling: formulation, implementation and application to multiphase flows[M]. John Wiley & Sons, 2016.

[21] Campbell J, Vignjevic R, Libersky L. A contact algorithm for smoothed particle hydrodynamics[J]. Computer methods in applied mechanics and engineering, 2000, 184(1): 49-65.

[22] Anderson T B, Jackson R. Fluid mechanical description of fluidized beds. Equations of motion[J]. Industrial & Engineering Chemistry Fundamentals, 1967, 6(4): 527-539.

[23] Wu K, Yang D, Wright N, et al. An integrated particle model for fluid–particle–structure interaction problems with free-surface flow and structural failure[J]. Journal of fluids and structures, 2018, 76: 166-184.

Averaging Navier-Stokes equations system by a dual approach

†*The Hung NGUYEN[1], Dong Anh NGUYEN[2]

[1]Faculty of Water Resources Engineering, University of Science and Technology, The University of Danang
[2]Institute of Mechanics, VAST.

*Presenting author: ngthung@dut.udn.vn; profhungthenguyen@gmail.com

†Corresponding author: ngthung@dut.udn.vn; profhungthenguyen@gmail.com

Abstract

The current three-dimensional averaging mathematical model of flow, also known as the Reynolds equations, was developed based on the idea of Reynolds in 1895. This model is given by the classical averaging of velocity and pressure parameters from the three-dimensional Navier-Stokes equations system. However, by doing this, these averaging parameters obtained by this classical approach are not generalized in comparison to ones estimated by the dual approach. This paper proposes a dual approach to establishing the three-dimensional flow equation. The model setup is more complicated than the classical model in terms of integration because the procedure can be repeated several times. In this paper, the authors perform twice: (i) first, integration of the velocity and pressure parameters from time t to Tm, with time Tm < t+T, where T is the repeated period of parameters; (ii) second, integration from time t to t+T. Fluctuating quantities such as velocity and pressure in turbulent flow, over time, are simulated using trigonometric Fourier series. The three-dimensional flow model obtained from this dual approach could provide more accurate results than those given by the Reynolds equations.

Keywords: Classical averaged method, dual approach, three-dimensional Navier-Stokes equations

Introduction

Flow in nature is usually turbulent and three-dimensional (3D) and it is often described by the Reynolds equations system [1, 2], classically averaged from the 3D Navier-Stokes equations system [1, 2].

With the classical averaging, quantities such as velocity and pressure are simply arithmetically averaged. In this research, these quantities will be averaged based on the dual approach, meaning that physical quantities such as velocity, pressure are integrated many times, have both local and global integration [3, 4]. In this paper, they are integrated only twice.

With this dual approach, it will be more complicated than the classical approach, but in return, we will receive better physical flow quantities than the classical approach [3, 4, 5, 6, 7, 8].

Averaging Navier-Stokes equations by a dual approach

The 3D Navier-Stokes equations [2] describes liquid motion written as follows:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial \mathrm{w}}{\partial z} = 0 \tag{1a}$$

$$\frac{\partial u}{\partial t} + \frac{\partial (u.u)}{\partial x} + \frac{\partial (u.v)}{\partial y} + \frac{\partial (u.\mathrm{w})}{\partial z} = \frac{1}{\rho}.F_x - \frac{1}{\rho}.\frac{\partial p}{\partial x} + \frac{1}{\rho}\frac{\partial \tau_{xx}}{\partial x} + \frac{1}{\rho}\frac{\partial \tau_{xy}}{\partial y} + \frac{1}{\rho}\frac{\partial \tau_{xz}}{\partial z} \tag{1b}$$

$$\frac{\partial v}{\partial t}+\frac{\partial(u.v)}{\partial x}+\frac{\partial(v.v)}{\partial y}+\frac{\partial(v.w)}{\partial z}=\frac{1}{\rho}.F_y-\frac{1}{\rho}.\frac{\partial p}{\partial y}+\frac{1}{\rho}\frac{\partial\tau_{yx}}{\partial x}+\frac{1}{\rho}\frac{\partial\tau_{yy}}{\partial y}+\frac{1}{\rho}\frac{\partial\tau_{yz}}{\partial z}$$

(1c)

$$\frac{\partial w}{\partial t}+\frac{\partial(u.w)}{\partial x}+\frac{\partial(v.w)}{\partial y}+\frac{\partial(w.w)}{\partial z}=\frac{1}{\rho}.F_z-\frac{1}{\rho}.\frac{\partial p}{\partial z}+\frac{1}{\rho}\frac{\partial\tau_{zx}}{\partial x}+\frac{1}{\rho}\frac{\partial\tau_{zy}}{\partial y}+\frac{1}{\rho}\frac{\partial\tau_{zz}}{\partial z}$$

(1d)

With turbulent flow, when the velocity and pressure quantities are averaged by the classical method, we get the Reynolds equations system as follows [1, 2]:

$$\frac{\partial\overline{u}}{\partial x}+\frac{\partial\overline{v}}{\partial y}+\frac{\partial\overline{w}}{\partial z}=0$$

(2a)

$$\frac{\partial\overline{u}}{\partial t}+\frac{\partial(\overline{u.u})}{\partial x}+\frac{\partial(\overline{u.v})}{\partial y}+\frac{\partial(\overline{u.w})}{\partial z}=\frac{1}{\rho}.F_x-\frac{1}{\rho}.\frac{\partial\overline{p}}{\partial x}+\frac{1}{\rho}\frac{\partial\overline{\tau}_{xx}}{\partial x}+\frac{1}{\rho}\frac{\partial\overline{\tau}_{xy}}{\partial y}+\frac{1}{\rho}\frac{\partial\overline{\tau}_{xz}}{\partial z}-\frac{\partial(\overline{u'.u'})}{\partial x}-\frac{\partial(\overline{u'.v'})}{\partial y}-\frac{\partial(\overline{u'.w'})}{\partial z}$$

(2b)

$$\frac{\partial\overline{v}}{\partial t}+\frac{\partial(\overline{v.u})}{\partial x}+\frac{\partial(\overline{v.v})}{\partial y}+\frac{\partial(\overline{v.w})}{\partial z}=\frac{1}{\rho}.F_y-\frac{1}{\rho}.\frac{\partial\overline{p}}{\partial y}+\frac{1}{\rho}\frac{\partial\overline{\tau}_{yx}}{\partial x}+\frac{1}{\rho}\frac{\partial\overline{\tau}_{yy}}{\partial y}+\frac{1}{\rho}\frac{\partial\overline{\tau}_{yz}}{\partial z}-\frac{\partial(\overline{v'.u'})}{\partial x}-\frac{\partial(\overline{v'.v'})}{\partial y}-\frac{\partial(\overline{v'.w'})}{\partial z}$$

(2c)

$$\frac{\partial\overline{w}}{\partial t}+\frac{\partial(\overline{w.u})}{\partial x}+\frac{\partial(\overline{w.v})}{\partial y}+\frac{\partial(\overline{w.w})}{\partial z}=\frac{1}{\rho}.F_z-\frac{1}{\rho}.\frac{\partial\overline{p}}{\partial z}+\frac{1}{\rho}\frac{\partial\overline{\tau}_{zx}}{\partial x}+\frac{1}{\rho}\frac{\partial\overline{\tau}_{zy}}{\partial y}+\frac{1}{\rho}\frac{\partial\overline{\tau}_{zz}}{\partial z}-\frac{\partial(\overline{w'.u'})}{\partial x}-\frac{\partial(\overline{w'.v'})}{\partial y}-\frac{\partial(\overline{w'.w'})}{\partial z}$$

(2d)

Where:

$u=\overline{u}+u'$, $v=\overline{v}+v'$, $w=\overline{w}+w'$, $p=\overline{p}+p'$

u, v, w, p are the instantaneous velocity (in the x, y and z direction respectively) and p is the pressure;

$\overline{u}, \overline{v}, \overline{w}, \overline{p}$ are the averaged quantities of velocity (in the x, y and z direction respectively) and p is pressure;

u', v', w', p' are the fluctuating quantities of velocity (in the x, y, and z - direction respectively) and pressure;

$\overline{y_i'.y_j'}$ is the mean product of fluctuating quantities of y_i, y_j ; is the friction components generated by the turbulent flow; y_i, y_j represent velocity u, v, w or pressure p which can be approximated in a variety of ways [2];

$\overline{\tau}_{i,j}$ is the shear stress, where the subscript i, j represent the x, y or z directions; if $i\equiv j$ we have the normal stress; if $i\neq j$ we have the shear stress.

In this paper, the average pressure and velocity quantities are obtained by dual approach by twice integration [3, 4] as follows:

$$\overline{y}(t)=\frac{1}{T}\int_0^T\frac{1}{r}\int_0^r y(t).dt.dr$$

(3)

First integral: these quantities are averaged by integration from time t (for simplicity we choose t = 0) to time (t + r), where r < T, and T is the fluctuating period of the quantities to be integrated:

$$\overline{y}_r=\frac{1}{r}\int_0^r y(t).dt$$

(4)

Second integral: integrations of these quantities are performed from time t to t+T:

$$\overline{y}(t)=\frac{1}{T}\int_0^T\overline{y}_r.dr$$

(5)

The fluctuating pressure or velocity quantities are approximated as a trigonometric Fourier series [9, 10, 11]; so the instantaneous velocity or pressure y(t) is approximate as:

$$y(t) = A_0 / 2 + \sum_{p=1}^{\infty} [A_p.\cos(p\omega t) + B_p.\sin(p\omega t) \tag{6}$$

where:

$$A_0 = \frac{2}{T} \int_0^T y(t)dt;$$

$$A_p = \frac{2}{T} \int_0^T y(t).\cos(p\omega t); \; p = 1, 2, ...$$

$$B_p = \frac{2}{T} . \int_0^T y(t).\sin(p\omega t)dt; \; p = 1, 2, ...$$

Substituting the quantity $y(t)$ into (4), we have the average quantity $\overline{y}_r(t)$ as follows:

$$\overline{y}_r(t) = A_0 / 2 + \sum_{p=1}^{\infty} [\frac{A_p}{p\omega T_m}.\sin(p\omega T_m) - \frac{B_p}{p\omega T_m}.\cos(p\omega T_m)]$$

and then, substituting $\overline{y}_r(t)$ into (5), we receive the average value $\overline{y}_j = \overline{y}(t_j)$ by dual approach as follows:

$$\overline{y}_j \equiv \overline{y}(t_j) = A_0 / 2 - \frac{1}{2\Pi T^2} \left(\sum_{p=1}^{N/2} \frac{B_p}{p} \right)$$

In this paper, the instantaneous velocity or pressure quantities at time $t = t_n$, denoted by $y(t_n)$ are discrete and finite measurement data; so we have the corresponding variables:

$$\frac{t}{T} \equiv \frac{t_n}{T} \sim \frac{n}{N} \tag{7}$$

Therefore, they are approximated by a discrete, finite trigonometric Fourier series as follows [10, 11]:

$$y(t_n) = A_0 / 2 + \sum_{p=1}^{N/2} [A_p.\cos(2\pi pn/N) + B_p.\sin(2\pi pn/N) \; ; \; n = 1, 2, ..., N \tag{8}$$

where:
A_0, A_p, B_p are the coefficients;

$$A_0 = \frac{2}{N} \sum_{n=1}^{N} y_n; \quad B_0 = 0 \tag{9a}$$

$$A_p = \frac{2}{N} \sum_{n=1}^{N} [y_n.\cos(2\pi pn/N); \; p = 1, 2, ...N / 2 \tag{9b}$$

$$B_p = \frac{2}{N} \sum_{n=1}^{N} [y_n.\sin(2\pi pn/N); \; p = 1, 2, ...N / 2 \tag{9c}$$

N sum of instantaneous velocity or pressure data values (u, v, w, p) mesures in time T;
p indicates the pth (for $p > $ N/2 these trigonometric harmonics will repeat);
t_n time to calculate, $t_n = $ n.Δt;
n time increment step.

The average velocity or pressure value obtained by the dual approach $\overline{y}_j = \overline{y}(t_j)$ will be as follows:

$$\overline{y}_j \equiv \overline{y}(t_j) = A_0 / 2 + \Delta DA \tag{10}$$

Where:
$A_0 / 2$ is the average quantity of Σy_j, calculated by the original ideal of Reynolds (1895), also called classical RANS equations;

$$\Delta DA = \frac{-1}{2\Pi N^2}\left(\sum_{p=1}^{N/2}\frac{B_p}{p}\right) \tag{11}$$

ΔDA is the difference of velocity or pressure quantities, when averaging these quantities with respect to the dual approach, compared with the classical RANS equations.

Algorithm used to solve the 3D Navier-Stokes equations system which is averaged by the dual approach

There are two methods to solve the 3D Naviers-Stokes equations system which is averaged by the dual approach.
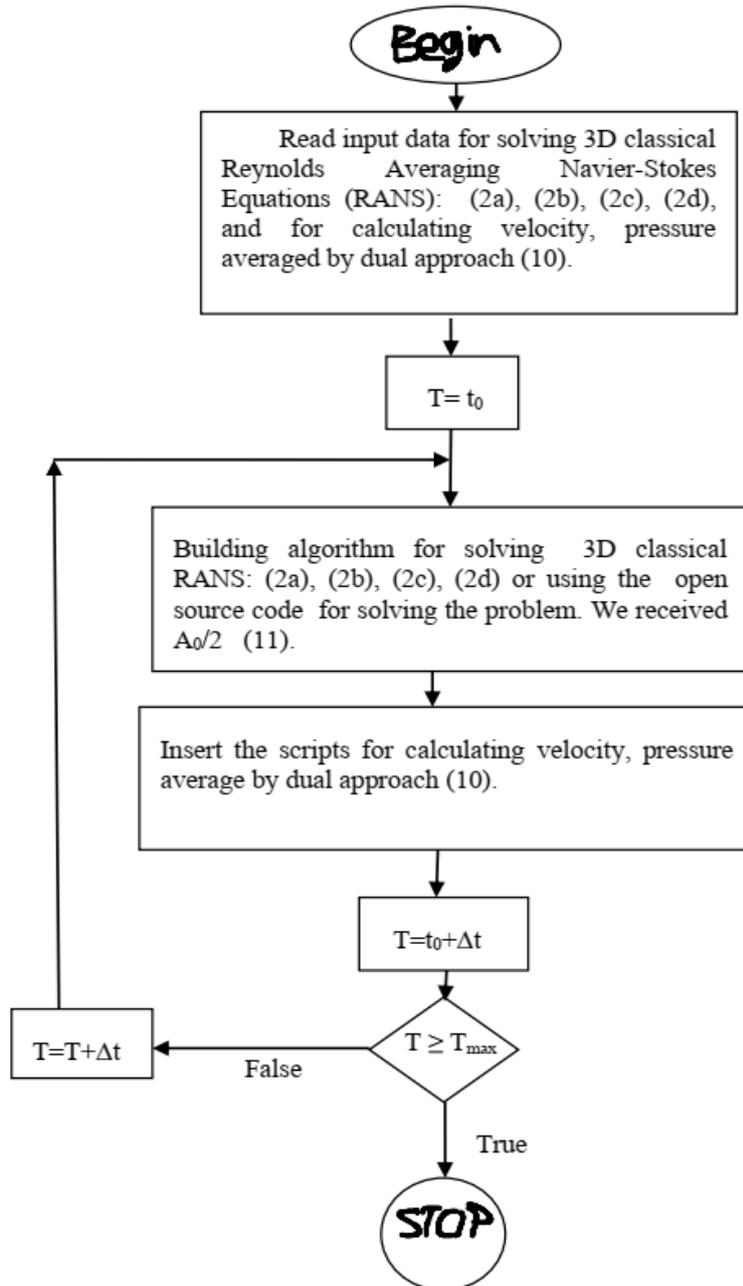
Figure 1. Flow chart of solving the 3D averaging Naviers-Stokes equations by dual approach

(i) *Method 1*: By relying on the numerical solution results of the 3D RANS equations by open source code, inserting command line segments to add differential increments ΔDA (11) to the

velocity and pressure solutions calculated at each time step to obtain the numerical solution of the 3D Naviers-Stokes equations system averaging by the dual approach, formula (10) (see Step 3, Fig. 1).

(ii) *Method 2*: Completely build a new algorithm to solve the 3D Naviers-Stokes equations system which is averaged by the dual approach as follows:

At each time step, constructing an algorithm to solve the 3D classical RANS equations, after that, inserting command line segments to add differential increments ΔDA (11) to the velocity and pressure solutions calculated at each time step to obtain the numerical solution of the 3D Naviers-Stokes equations averaging by the dual approach, formula (10) (see Step 3, Fig. 1).

Case study

In this case study, we approximate the instantaneous pressure quantity by a finite trigonometric Fourier series (8):

$$y(t_n) = A_0/2 + \sum_{p=1}^{N/2}[A_p.\cos(2\pi pn/N) + B_p.\sin(2\pi pn/N); \quad n = 1, 2, ..., N \tag{8}$$

The instantaneous pressure at a fixed point (MC7-2) measured over time is given in Table 1 and is shown in Fig. 2.

Table 1: Pressure measurement data over time at a fixed point (MC7-2) in the first twenty (N = 20) measurement steps [12]

Time (sec)	1/5	2/5	3/5	4/5	5/5
Pressure (mm)	354.5706523	356.9267077	359.4528209	357.827718	354.8137758
Time (sec)	6/5	7/5	8/5	9/5	10/5
Pressure (mm)	354.4195018	357.937984	359.5774499	360.402683	360.8192235
Time (sec)	11/5	12/5	13/5	14/5	15/5
Pressure (mm)	359.196406	358.0766566	358.3154676	356.2717004	356.2604194
Time (sec)	16/5	17/5	18/5	19/5	20/5
Pressure (mm)	359.5143341	361.8665435	360.5558524	359.1800103	357.4757616

With the pressure measurement interval over time $\Delta T = 1/5$ sec, the arithmetic mean pressure value (classical RANS) is:

$$\bar{p}_{CA} = A_0/2 = \frac{1}{N}\sum_{p_i=1}^{N} p_i = 358.1730834 mm;$$

Average value of pressure according to dual approach is:

$$\overline{y_j} \equiv \overline{y(t_j)} = \overline{p_{CA}} + \Delta DA = \overline{p_{CA}} - \frac{1}{2\Pi N^2}\left(\sum_{p=1}^{N/2}\frac{B_p}{p}\right) = A_0/2 - \frac{1}{2\Pi N^2}\left(\sum_{p=1}^{N/2}\frac{B_p}{p}\right) \tag{10}$$

$$\overline{y_j} = 358.1730834 - 0.4384605 = 357.7346229 mm$$

Comment

(i) From the solution obtained by the dual approach (10), when we arithmetically average the fluctuating velocity and pressure quantities as Reynolds proposed in 1895 (i.e. for $\Delta DA = 0$), we get the classical RANS equations [1, 2].

(ii) In order to calculate the increment ΔDA, we need to known the values of the fluctuating velocity and pressure quantities in the cycle T.

(iii)   The results here are for the case of a normal measuring point in the flow; in fact there are cases where the flow has a very large fluctuating values; and so the increments ΔDA will also be considerably large.
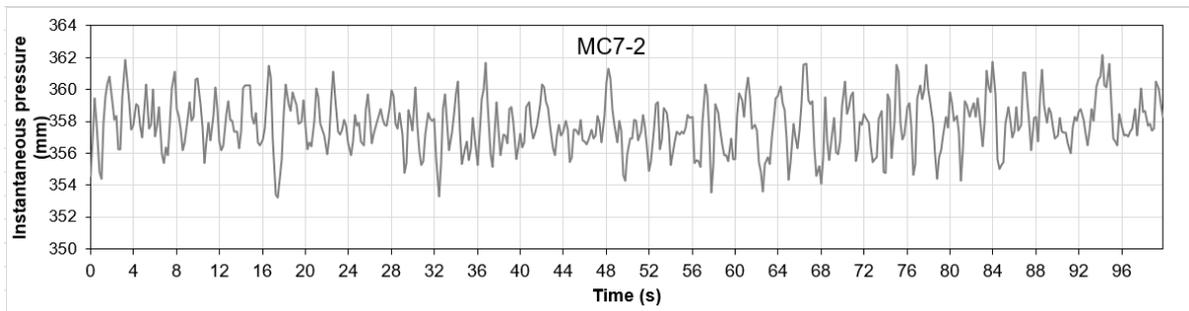
Figure 2. Pressure fluctuation at measuring point MC7-2 over time [12]

Conclusion

In this paper, the velocity and pressure fluctuations are approximated by a trigonometric Fourier series; the velocity or pressure quantities received by the dual approach (10) are more general than those obtained using the classical RANS method.

References

[1] Reynolds (1895), "On the Dynamical Theory of Incompressible Viscous Fluids and the Determination of the Criterion," Philosophical Transactions of the Royal Society of London, Series A, vol. 186, pp. 123–164.

[2] Weiming Wu (2007), *Computation river dynamics*, Taylor and Francis / Balkema.

[3] Nguyen Dong Anh (2012), Dual approach to averaged values of functions, Vietnam Journal of Mechanics, VAST, Vol. 34, No. 3, pp. 211 – 214.

[4] Nguyen Dong Anh (2012), Dual approach to averaged values of functions: Advanced formulas, Vietnam Journal of Mechanics, Vast, Vol. 34, No. 4, pp. 321 – 325.

[5] Hung, NGUYEN The (2017), A dual approach to modeling solute transport, The International Conference on Advances in Computational Mechanics, pp. 821-834.

[6] Tinh Ton That[1], The Hung Nguyen[1*], Dong Anh Nguyen[2], A dual approach for model construction of two-dimensional horizontal flow, *Proceedings of the 10th International Conference on Asian and Pacific Coasts (APAC 2019) Hanoi, Vietnam, Sept. 25-28, 2019, 115-120.*

[7] *H. Nguyen-The*, A dual approach for modeling two- and one-dimensional solute transport, The International Conference on modern Mechanics and Applications, ICOMMA2020 December 02 – 04, HoChiMinh city, Vietnam, Lecture notes in Mechanical Engineering, Pp(978-981).

[8] Nguyen The Hung (2022), Building mathematical model of two-dimensional vertical flow by dual approach, Journal of Water Resources Science and Technology, No.71, pp(109-116).

[9] Hiroyuki Shima et Tsuneyoshi Nakayama, Higher Mathematics for Physics and Engineering, Springer 2010.

[10] Phan Quoc Khanh, Specialized mathematics, HoChiMinh National University, 2000.

[11] William J. Emery et Richard E. Thomson, Data Analysis Methods in Physical Oceanography, Elsevier 2004.

[12] LE Van Nghi et al., General report – Building physical model, Experimenting with cross-sectional model and Do Luong Focal cluster (sectional model + overall model) – Technical phase – Hanoi 2016.

An extension of radial basis function based finite element method on quadtree mesh refinement

*†D. Nguyen Kien[1], Xiaoying Zhuang[1,2,3] and Timon Rabczuk[4]

[1]Department of Geotechnical Engineering, Tongji University, China
[2]Institute of Photonics, Faculty of Mathematics and Physics, Leibniz University Hannover, Germany
[3]Hannover Center for Optical Technologies, Leibniz University Hannover, Germany
[4]Chair of Computational Mechanics, Bauhaus University Weimar, Germany

[*]Presenting author, [†]Corresponding author: nkdung@hotmail.com, kiendung.nguyen@tongji.edu.cn

Abstract

The recent developed method, radial basis function based finite element method (RBF-FEM) could solve the partial differential equation engineering problems [12, 31, 7]. The RBF-FEM method utilizes the radial basis functions to generate the shape functions from the node in the element (the nonoverlapping region). When the problems in terms of localized deformation, or steep gradients, or even discontinuity regions are posed, a fine mesh is created to produce more accurate results. Therefore a mesh refinement technique is employed in order to reduce the computational time cost. This study introduces an extension of the RBF-FEM on quadtree mesh. The proposed approach combines RBF-FEM and the polygonal finite element method, which is the so-called RBF-PolyFEM, to construct conforming approximations on quadtree mesh refinement. Numerical examples are presented to demonstrate the accuracy and performance of the proposed h-adaptive RBF-FEM approach in comparison to other numerical methods.

Keywords: Radial basis function, finite element method, polygonal finite element method, quadtree mesh, adaptive mesh refinement.

1 Introduction

1.1 Background

The recent developed method, radial basis function based finite element method (RBF-FEM) could solve the partial differential equation engineering problems ([12, 31, 7]). The RBF-FEM method utilizes the radial basis functions to generate the shape functions from the node in the element (the nonoverlapping region). The connection of the elements creates topologically the mesh. When the problems in terms of localized deformation, or steep gradients, or even discontinuity regions are posed, a fine mesh is created to produce more accurate results. Therefore a technique on the mesh is utilized in order to reduce the computational time cost. A mesh refinement strategy is preferred over the usage of uniform mesh.

Today the preferred and advanced mesh refinement technique relies on natural refinement. One of them is the quadtree mesh technique which uses the recursive decomposition principle [21] to create an efficient way for mesh refinement. In application to numerical methods relying on this quadtree mesh (for example, the finite element method), a special shape function must be constructed that adapts with hanging nodes. The hanging nodes appear during the mesh refinement. Several authors built their techniques based on finite element methods that can solve the partial differential equations problems with quadtree mesh refinement [18, 19, 6].

However, when applying the RBF-FEM in order to solve the partial differential equations with the quadtree mesh, the results are not as accurate as expected. The reason is because of the

element in which the hanging node laying on the edge. As shown in the paper [59], the number of nodes in the element that should be recommended to utilize are three, four, or eight and they should be well-positioned on the edge of the element. The other element types, five- and six-node elements, could be used with acceptable results in computation (Figs. 1a and 1b). Based on this RBF-FEM approach, a slight change in the position of nodes in the element (Figs. 1c and 1d) are applied to test the partial differential equation cases with quadtree meshes. However, in this practice, they do not show good results. Two examples are carried out to illustrate that. Laplace and Poisson II equations with Dirichlet boundary condition (Eq. (27) and Eq. (31), respectively) are tested. With Laplace equation, six quadtree meshes are utilized (Figs. 10a-10f). With Poisson II equation, six meshes are applied (Figs. 16a-16f). The mesh starts at the initial simple uniform mesh, and then refines and focuses on the critical area. The results are subsequently shown in comparison to the proposed method, adaptive RBF-PolyFEM or in short ARBF-PolyFEM (which will be mentioned in the following sections of the current paper) in Table 1. By RBF-FEM method, the L^2-error norm increases in Laplace equation with the d mesh and in Poisson II with f mesh (Table 1).
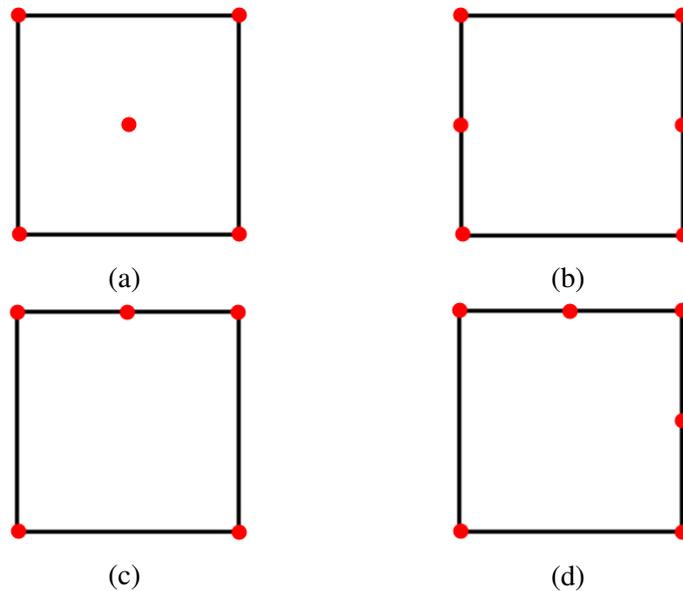
Figure 1: *RBF-FEM element: (a) 5-node element, (b) 6-node element, (c) Quadtree 5-node element, (d) Quadtree 6-node element*

This is, therefore, a requirement for the extension of the RBF-FEM method for solving the system of partial differential equations, with efficient way on the regions of steep gradients, singularities or even discontinuities.

1.2   Appoarch

Several efficient approaches can treat the hanging node in a quadtree mesh. A simple and efficient approach is that it can be added rectangular or triangular elements inside the hanging node elements to get the compatible meshes [18, 19]. Then the traditional finite element method can possibly be applied to solve the problems with this mixed mesh. Another novel classical approach introduced by Gupta, 1978 [6] which is deriving the conforming shape function of the transition hanging node quadrilateral elements. Another advanced approach is using polygonal interpolation in the quadtree element nodes to construct the shape function [24, 25]. The method applies the natural neighbor Laplace interpolation to approximate the node values along the edge with conforming C^0 approximation [3, 22, 8]. Then solving the equation by polygonal

Table 1: *Rrelative error in the L^2 norm for the patch test.*

Problem	Mesh	No. nodes	Relative L^2 error norm	
			RBF-FEM	ARBF-PolyFEM
Laplace	a	16	2.20×10^{-2}	2.20×10^{-2}
	b	49	5.55×10^{-3}	5.55×10^{-3}
	c	138	2.01×10^{-3}	1.40×10^{-3}
	d	292	$\mathbf{2.91 \times 10^{-3}}$	4.75×10^{-4}
	e	343	1.81×10^{-3}	4.23×10^{-4}
	f	485	1.63×10^{-3}	2.75×10^{-4}
Poisson II	a	9	2.33×10^{-1}	2.33×10^{-1}
	b	25	7.27×10^{-2}	7.27×10^{-2}
	c	81	1.93×10^{-2}	1.92×10^{-2}
	d	253	6.92×10^{-3}	5.02×10^{-3}
	e	697	6.90×10^{-3}	1.50×10^{-3}
	f	1265	$\mathbf{7.19 \times 10^{-3}}$	9.13×10^{-4}

finite element methods can obtain the solution. Recently, polygonal finite element method and its polygonal adaptive mesh refinement techniques are utilized to solve the partial differential equations problems [2, 15, 17, 16].

The proposed method is the combination of radial basis function-based finite element method (RBF-FEM) and polygonal finite element method in solving partial differential equations (Laplace and Poisson equations) with mesh refinement. The sub-domain element in which the node number equals four will be applied by RBF-FEM and the other (five- and/or six-node elements) will be utilized by the polygonal finite element method. The Wachspress method is used to construct the shape function in the polygonal finite element method [26, 23]. In this study, we would like to investigate how the combination could solve the Poisson/Laplace equations with quadtree mesh refinement. Several numerical studies are then carried out in comparison to other numerical methods.

2   Adaptive RBF-polyFEM approach

The whole domain is divided into 2 domains, Ω^R and Ω^P where RBF-FEM and Polygonal FEM can be applied, respectively. The idea of the adaptive RBF-polyFEM approach which combines the RBF-FEM and the polygonal finite element method to construct the conforing approximations on quadtress mesh refinement is presented in the following.

2.1   RBF-FEM shape function

The formulation of the radial basis function interpolation for the physical degree of freedom is presented. The idea is illustrated by a simple two-dimension example. Suppose that we construct a convex four-edge polygonal element with n_p nodes (Fig. 2). The value $s(\mathbf{x})$ at each node in the element (n_p degree of freedom) is interpolated as (based on radial basis function)

$$s(\mathbf{x}) = \sum_{i=1}^{n_p} a_i \, \phi(||\mathbf{x} - \mathbf{x}_i||) \tag{1}$$

where $\mathbf{x} = [x, y]$, and the vertex at the ith node $\mathbf{x}_i = [x_i, y_i]$ (Fig. 2), and a_i is the unknown weight coefficient. The radial basis function $\phi(||\mathbf{x} - \mathbf{x}_i||)$ of which the commonly-used types list in Table 2. In the table, the distance $\mathbf{r} = ||\mathbf{x} - \mathbf{x}_i||$, and the shape parameter, ε, can be used to scale the input of the radial kernel and can influence the fitting accuracy. By evaluating the
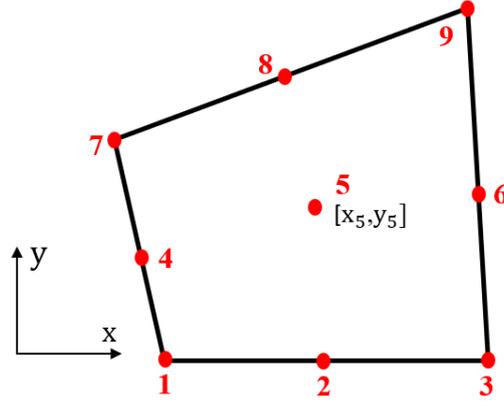
Figure 2: *An element with nodes in global coordinate system.*

Table 2: *Types of radial basis functions.*

RBF type	Abbreviation	Definition
Gaussians	GA	$\phi(\mathbf{r}, \varepsilon) = e^{-(\varepsilon/\mathbf{r})^2}$
Multiquadrics	MQ	$\phi(\mathbf{r}, \varepsilon) = \sqrt{1 + (\varepsilon\mathbf{r})^2}$
Inverse Multiquadrics	IMQ	$\phi(\mathbf{r}, \varepsilon) = \frac{1}{\sqrt{1+(\varepsilon\mathbf{r})^2}}$
Inverse Quadratics	IQ	$\phi(\mathbf{r}, \varepsilon) = \frac{1}{1+(\varepsilon\mathbf{r})^2}$

model at each node we can find the generalized coordinates as

$$
\begin{aligned}
s_1(\mathbf{x}) &= \sum_{i=1}^{n_p} a_i\, \phi(||\mathbf{x}_1 - \mathbf{x}_i||) \\
s_2(\mathbf{x}) &= \sum_{i=1}^{n_p} a_i\, \phi(||\mathbf{x}_2 - \mathbf{x}_i||) \\
&\cdots \\
s_{n_p-1}(\mathbf{x}) &= \sum_{i=1}^{n_p} a_i\, \phi(||\mathbf{x}_{n_p-1} - \mathbf{x}_i||) \\
s_{n_p}(\mathbf{x}) &= \sum_{i=1}^{n_p} a_i\, \phi(||\mathbf{x}_{n_p} - \mathbf{x}_i||)
\end{aligned}
\tag{2}
$$

or in tensor notation it could be written as

$$
\mathbf{s} = \mathbf{\Phi}\,\mathbf{a} \tag{3}
$$

where

$$
\mathbf{s} = [s_1(\mathbf{x})\ s_2(\mathbf{x})\ \ldots\ s_{n_p-1}(\mathbf{x})\ s_{n_p}(\mathbf{x})]^T, \qquad \mathbf{a} = [a_1\ a_2\ \ldots\ a_{n_p-1}\ a_{n_p}]^T \tag{4}
$$

and

$$\boldsymbol{\Phi} = \begin{bmatrix} \phi(||\mathbf{x}_1 - \mathbf{x}_1||) & \dots & \phi(||\mathbf{x}_1 - \mathbf{x}_{n_p}||) \\ & \cdot & \cdot \\ & \cdot & \cdot \\ & \cdot & \cdot \\ \phi(||\mathbf{x}_i - \mathbf{x}_1||) & \dots & \phi(||\mathbf{x}_i - \mathbf{x}_{n_p}||) \\ & \cdot & \cdot \\ & \cdot & \cdot \\ & \cdot & \cdot \\ \phi(||\mathbf{x}_{n_p} - \mathbf{x}_1||) & \dots & \phi(||\mathbf{x}_{n_p} - \mathbf{x}_{n_p}||) \end{bmatrix} \tag{5}$$

The generalize coordinates could be expressed as the solution of Eq. (3) for $\mathbf{a}$

$$\mathbf{a} = \left(\boldsymbol{\Phi} \right)^{-1} \mathbf{s} \tag{6}$$

Expressing the term in Eq. (1) as the row vector multiplied by the column vector, the interpolation $s(\mathbf{x})$ could be expressed as

$$s(\mathbf{x}) = \boldsymbol{\phi}(\mathbf{x}) \, \mathbf{a} \tag{7}$$

where

$$\boldsymbol{\phi}(\mathbf{x}) = \begin{bmatrix} \phi(||\mathbf{x} - \mathbf{x}_1||) & \dots & \phi(||\mathbf{x} - \mathbf{x}_i||) & \dots & \phi(||\mathbf{x} - \mathbf{x}_{n_p}||) \end{bmatrix} \tag{8}$$

Therefore, from Eq. (6) and Eq. (7) we have

$$s(\mathbf{x}) = \boldsymbol{\phi}(\mathbf{x}) \left(\boldsymbol{\Phi} \right)^{-1} \mathbf{s} \tag{9}$$

The RBF-FEM shape function is then obtained as

$$\mathbf{N}_R = \boldsymbol{\phi}(\mathbf{x}) \left(\boldsymbol{\Phi} \right)^{-1} \tag{10}$$

and its gradients are written as

$$\nabla \mathbf{N}_R = \nabla \boldsymbol{\phi}(\mathbf{x}) \left(\boldsymbol{\Phi} \right)^{-1} \tag{11}$$

where $\nabla \boldsymbol{\phi}(\mathbf{x}) = \begin{bmatrix} \nabla \phi(||\mathbf{x} - \mathbf{x}_1||) \dots \nabla \phi(||\mathbf{x} - \mathbf{x}_i||) \dots \nabla \phi(||\mathbf{x} - \mathbf{x}_{n_p}||) \end{bmatrix}$.

Therefore the displacement field approximation based on the RBF-FEM shape function on each element could be given as

$$\mathbf{u} = \mathbf{u}(\mathbf{x}) \approx \mathbf{u}^e(\mathbf{x}) = \mathbf{N}_R^e(\mathbf{x})\mathbf{d}^e, \qquad \mathbf{x} \in \Omega^e \tag{12}$$

Then the governing equation on each RBF-FEM element can simply be expressed as

$$\mathbf{k}^e \mathbf{u}^e(\mathbf{x}) = \mathbf{f}^e \quad \text{in} \quad \Omega_e^R \tag{13}$$

Plots of the shape function $\mathbf{N}_R$ and the derivatives of shape function $\partial \mathbf{N}_R / \partial \boldsymbol{\xi}$ for 1D element and 2D element are depicted in Figs. 3 and 4, respectively. It shows the basis function is non-linear and C^0 continuous.

(a) N_R

(b) $\partial N_R/\partial \xi$
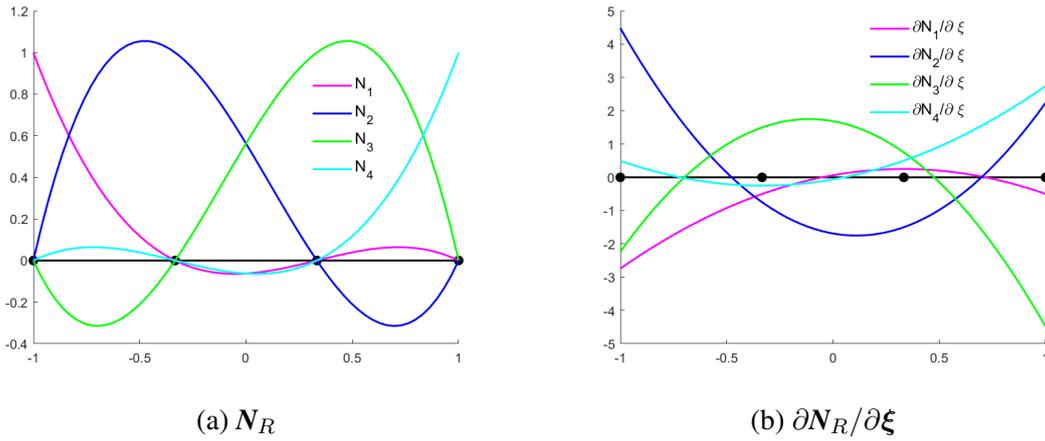
Figure 3: *MQ RBF-FEM shape functions and its derivatives of 1D four-node element.*
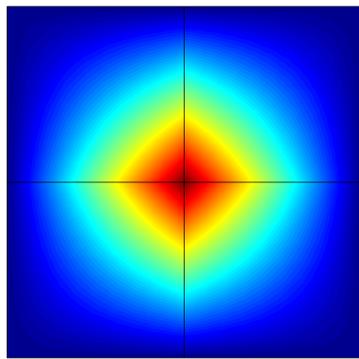
(a) *Contour of the shape function*
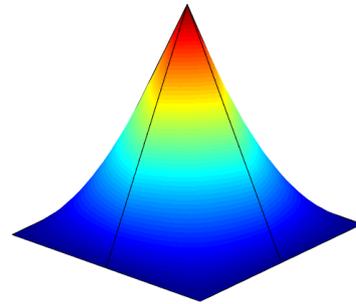
(b) *3D surface plot of the shape function*

(c) *The derivaties of shape function $dN_{Ri}/d\xi$*

(d) *The derivaties of shape function $dN_{Ri}/d\eta$*

Figure 4: *Plot of the shape function of a connecting node and the shape function derivatives of the MQ RBF-FEM 2D four-node element.*

2.2 Polygonal Wachspress shape function

Consider the polygonal domain Ω^P splits into another type of subdomain on polygonal higher number node Ω_e^P. Each subdomain is discretized by n_P nodes.

Let us define a polygon (vertices, $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{n_P}$, where n_P is the number of nodes in the poly-

gon). Then the Wachspress shape function, for any $\mathbf{v} \in \Omega^P$, could be written as [26, 27, 13]

$$N_{Pi} = \frac{w_i}{\sum_{j=1}^{n_P} w_j}, \quad \text{with } w_i = \frac{A(\mathbf{x}_{i-1}, \mathbf{x}_i, \mathbf{x}_{i+1})}{A(\mathbf{v}, \mathbf{x}_{i-1}, \mathbf{x}_i)A(\mathbf{v}, \mathbf{x}_i, \mathbf{x}_{i+1})} \tag{14}$$

where $A(\mathbf{x}_{i-1}, \mathbf{x}_i, \mathbf{x}_{i+1})$ represents the area of the triangle inside the polygon (Fig. 5a). To get the Wachspress coordinates, the approach regarding the perpendicular distance to the edge of the polygon is utilized [4]. Fig. 5b shows the perpendicular distance $h_i(\mathbf{x})$ of $\mathbf{v}$ to the edge. Let us define $\mathbf{p}_i(\mathbf{x}) = \mathbf{n}_i/h_i(\mathbf{x})$. In this definition, the unit normal vector that outwards the edge is denoted by $\mathbf{n}_i$. The Wachspress shape function and its gradients are therefore respectively written as

$$N_{Pi} = \frac{\bar{w}_i}{\sum_{j=1}^{n_P} \bar{w}_j}, \quad \text{with } \bar{w}_i = \det(\mathbf{p}_{i-1}, \mathbf{p}_i) \tag{15a}$$

$$\nabla N_{Pi} = N_{Pi}\left(\boldsymbol{\vartheta}_i - \sum_{j=1}^{n_P} N_{Pj}\boldsymbol{\vartheta}_i\right), \quad \text{where } \boldsymbol{\vartheta}_i = \mathbf{p}_{i-1} + \mathbf{p}_i \tag{15b}$$

Fig. 5c shows 3D surface plot of Wachspress shape function.

(a) *Triangles.* (b) *Perpendicular diameters.* (c) *3D surface plot.*

Figure 5: *Polygonal Wachspress approximation.*
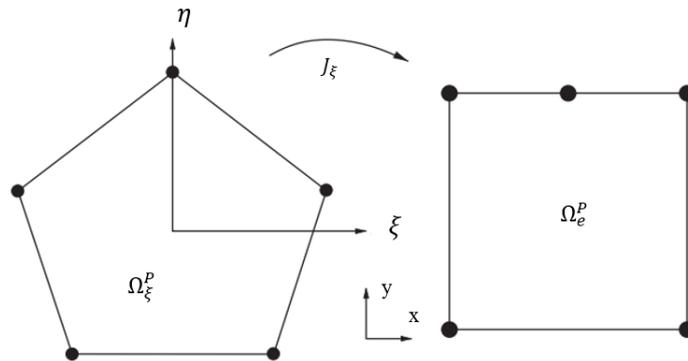
Figure 6: *Mapping from a reference pentagonal element (natural coordinates) to a five-node quadtree mesh element (physical coordinates) with a hanging node.*

The Wachspress shape function constructs the conforming approximation over the polygonal reference element. The pentagon is considered the reference element and is shown in Fig. 6. All

the vertice nodes of the reference pentagonal element lay on the circumcircle. The interpolation to the physical element is obtained via isoparametric mapping $\mathbf{J}_\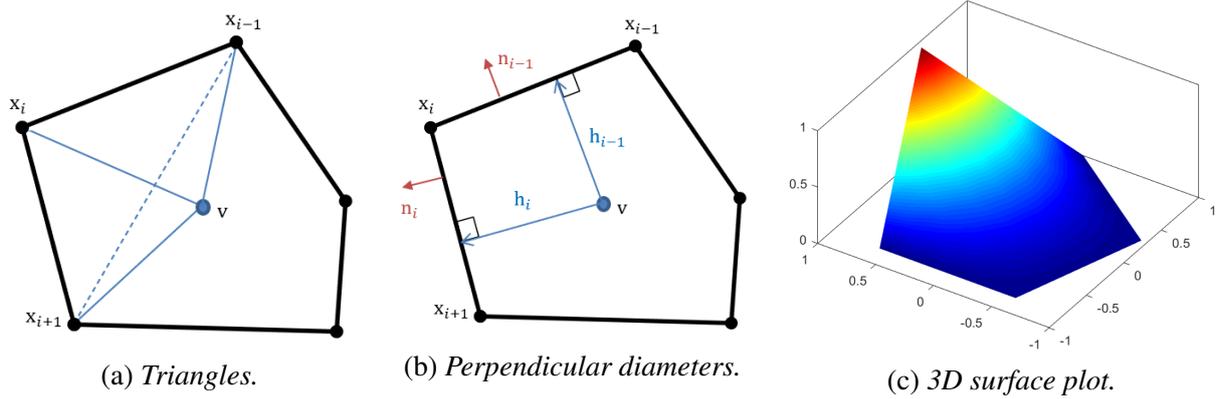xi = \partial\mathbf{x}/\partial\boldsymbol{\xi}$. Fig. 6 depicts an example of mapping from a reference pentagonal element (natural coordinates) to a five-node quadtree mesh element (physical coordinates) with a hanging node. The interpolation remains linear on the physical element edges when it has the affine mapping.

2.3 RBF-PolyFEM: discretization

Laplace and Poisson equations are utilized to evaluate the approach. The discretization process is described shortly and it is similar to the traditional finite element method.

The whole domain Ω is divided into two main domains. The first domain (RBF-FEM domain, Ω^R) splits into finer subdomains (Ω_e^R) in which the node number equals to four will be applied by RBF-FEM method (white elements in Fig. 7), and the second (polygonal domain, Ω^P) divided into subdomains (five- and/or six-node elements of Ω_e^P) will be utilized by the polygonal finite element method (light blue elements in Fig. 7).
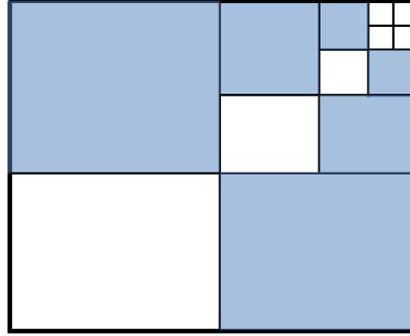
Figure 7: *Adaptive RBF-PolyFEM mesh.*

Laplace/Poisson equations problem (applications to engineering problems such as heat conduction, electromagnetics, and so on.) is considered. The boundary value problem of elliptic equations in two-dimension is given as follows.

$$\Delta^2 u(\mathbf{x}) = f(\mathbf{x}), \quad \text{in } \Omega \tag{16a}$$

$$\frac{\partial u}{\partial n} = g, \quad \text{on } \Gamma_N \tag{16b}$$

$$u = 0, \quad \text{on } \Gamma_D \tag{16c}$$

where $\Omega = \Omega^R \cap \Omega^P$, denotes the problem domain, and the Dirichlet and Neumann boundaries are, respectively, denoted by Γ_D and Γ_N, with $\Gamma_D \cup \Gamma_N = \partial\Omega$. Therefore, the variational weak form of the equations Eq. (16) could be obtained as

$$\mathbf{B}(u, \delta u) = \mathbf{L}(\delta u), \quad \forall \delta u \in H_0^1(\Omega) \tag{17a}$$

$$\mathbf{B}(u, \delta u) = \int_\Omega \nabla u \cdot \nabla \delta u \, d\Omega \tag{17b}$$

$$\mathbf{L}(\delta u) = \int_\Omega f \delta u \, d\Omega + \int_{\Lambda_N} g \delta u \, ds \tag{17c}$$

where δu the test function in the Sobolev space $H_0^1(\Omega)$.

The unknown variable u is approximated over the arbitrary element in the domain using either

the RBF-FEM shape function Eq. (10) or polygonal Wachspress shape function Eq. (15)). The nodal variable u_i could then be obtained as

$$u \cong u^h = \sum_i^{n_p} N_i u_i \tag{18}$$

where n_p represents the node number in the element, N_i denotes the shape function of either RBF-FEM or polygonal Wachspress finite element method depending on the element approximation. If the approximated field variable u_h is substituted from Eq. (18) into Eq. (17), applying Galerkin method, considering Neumann boundary condition, assembling the global coefficient matrix and load vector, then the PDE discretized form can be obtained as [20]

$$\mathbf{Ku} = \mathbf{F} \tag{19}$$

where $\mathbf{u}$ represents the global nodal vector, $\mathbf{K}$ the global coefficient matrix, $\mathbf{F}$ the loading vector. In index notation, the components of $\mathbf{K}$ matrix and $\mathbf{F}$ vector are written as

$$K_{ij}^e = \int_{\Omega_e} \left(\frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y} \right) d\Omega_e \tag{20}$$

$$F_i^e = \int_{\Omega_e} \left(f_i \partial N_i \right) d\Omega + \int_{\Gamma_N^e} \left(g N_i \right) d\Gamma \tag{21}$$

The shape function of hanging node is shown in Figure. 8
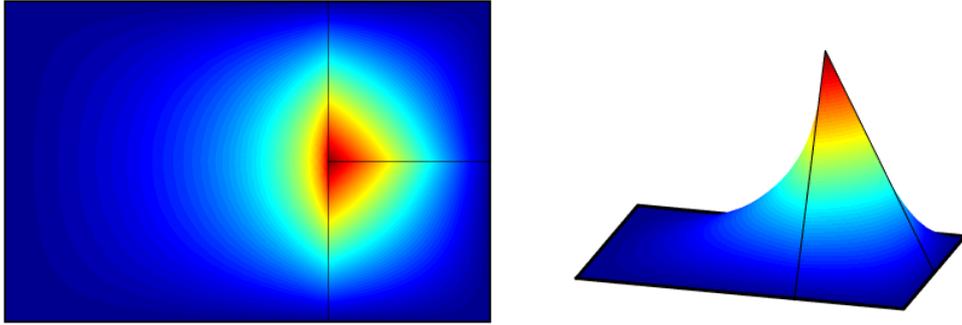
Figure 8: *Contour (LSH) and 3D surface plot (RHS) of the shape function of a hanging node.*

2.4 Adaptive strategy

To measure the error, the L^2-norm and energy-norm errors are commonly used. The L^2-norm error is introduced and it is defined as

$$||\mathbf{e}||_{L^2}(\Omega) = \left(\int_\Omega (\mathbf{u} - \mathbf{u}^h)^T (\mathbf{u} - \mathbf{u}^h) \, d\Omega \right)^{1/2} \tag{22}$$

where $\mathbf{u}$ denotes exact displacement solution and $\mathbf{u}^h$ the numerical displacement approximation solution. In the strategy of adaptive mesh refinement, the energy-norm error is utilized and it is written as

$$||\mathbf{e}|| = \left(\int_\Omega \nabla(\mathbf{u} - \mathbf{u}^h)^T \nabla(\mathbf{u} - \mathbf{u}^h) \, d\Omega \right)^{1/2} \tag{23}$$

The relative energy-norm error is given as

$$\eta = \frac{||\mathbf{e}||}{(||\mathbf{u}^h||^2 + ||\mathbf{e}||^2)^{1/2}} \tag{24}$$

In this study, we apply the technique of Zienkiewicz-Zhu error estimation [29, 30] as the criteria of mesh refinement to solve the problems of the adaptive scheme. The element permissible error is defined as

$$||e||_{all} = \eta_{\text{all}} \left(\frac{||\mathbf{u}^h||^2 + ||\mathbf{e}||^2}{n_E} \right)^{1/2} \tag{25}$$

Where n_E denotes the number of elements in the mesh, η_{all} represents the allowable relative error norm ([28, 10]).

The index ξ_i (in different definition from the previous sections and chapters about the natural coordinate system) is defined to mark the element in the adaptive refinement technique.

$$\xi_i = \frac{||e||_i}{||e||_{all}} \tag{26}$$

where $||e||_i$, the energy-norm error of the ith element, is calculated by Eq. (23). The element will be refined if $\xi_i > 1$. Otherwise, the element will be merged.

The adaptive mesh refinement stratergy is shown in Algorithm 1.

Algorithm 1 *The adaptive strategy.*

    Input: Initial mesh $\mathcal{M}_I$, maximum number of mesh elements $n_{\max}$
    Ouput: Numerical approximation u_J^h, the finesh mesh $\mathcal{M}_J$
    Algorithm:
        $K = 0$;
        while $n < n_{\max}$ **do**
            K = K+1;
            SOLVE Eq. (19) on $\mathcal{M}_K$ to get the numerical approximation solution u_K^h;
            ESTIMATE the energy-norm error of the ith element on its domain Ω_i (Eq. (23)),
                and calculate the error in the L^2-norm $||e||_{L^2(\Omega)}$ (Eq. (22)) ;
            MARK a set $\mathcal{S}_K \in \mathcal{M}_K$ with minumum number such that $\xi_k > 1$ (Eq. (26))
            REFINE mesh elements $\tau \in \mathcal{S}_K$ to generate a new mesh $\mathcal{M}_{K+1}$
                (based on the quadtree mesh refinement algorithm from Funken
                and Schmidt, 2020 [5])
        end while
        $u_J^h = u_K^h, \mathcal{M}_J = \mathcal{M}_K$
    end Algorithm.

---

3   Numerical studies

Several numerical test cases on Laplace/Poisson equations are carried out to investigate the adaptive scheme performance. The patch test is first studied, then three Poisson and Laplace equations involving steep gradients are solved. The direct solver is applied to solve the linear algebraic equations. The Matlab codes are built to analyze these test case performance. The maximum energy-norm permissible error is configured to six percent ($\eta_{\max} = 0.06$). The MQ radial basis function with shape parameter, $\epsilon = 400$, is employed to run the cases.

3.1 Patch test

Let us consider a patch test for the Laplace equation, $\Delta u = 0$ in the square domain $\Omega = (0,1)^2$, imposing the essential boundary conditions on the square boundaries with $u = g(\mathbf{x}) = x + y$ [25]. Regularized (2:1 rule) meshes and non-regularize meshes are used for patch test as shown in Fig. 9. The results listed in Table. 3 shows that they pass the patch test over quadtree meshes. The L^2-norm error are $\mathcal{O}(10^{-9})$ for the case with 2:1 rule quadtree mesh, and $\mathcal{O}(10^{-6})$ with no 2:1 rule.
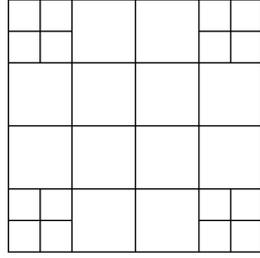
(a) *Regularized meshes: 45 nodes*
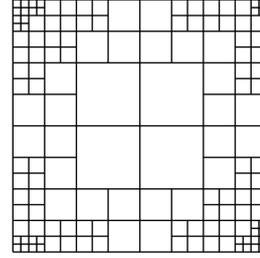
(b) *Regularized meshes: 203 nodes*
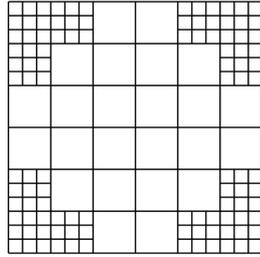
(c) *Non regularized meshes: 177 nodes*
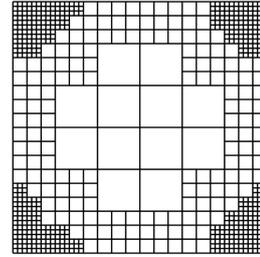
(d) *Non regularized meshes: 793 nodes*

Figure 9: *Patch test on regularized and non-regularized meshes.*

Table 3: *The relative L^2-norm error for the patch test.*

Ratio	Mesh	No. nodes	Relative L^2-norm error
2:1	a	45	5.1×10^{-9}
	b	203	7.3×10^{-9}
No 2:1	c	177	8.6×10^{-6}
	d	793	1.2×10^{-6}

3.2 Laplace problem

Another test is solving the equation of Laplacian with Dirichlet boundary condition as follows

$$\nabla^2 u = 0, \qquad \text{in } \Omega = (0,1)^2, \tag{27a}$$

$$u(0,y) = 0, \qquad \text{for } 0 \leq y \leq 1 \tag{27b}$$

$$u(x,0) = 0, \qquad \text{for } 0 \leq x \leq 1 \tag{27c}$$

$$u(1,y) = 0, \qquad \text{for } 0 \leq y \leq 1 \tag{27d}$$

$$u(x,1) = w_0 \sin(\pi x), \qquad \text{for } 0 \leq y \leq 1 \tag{27e}$$

The exact solution is given as

$$u(\mathbf{x}) = \frac{w_0}{\sinh(\pi)}\sin(\pi x)\sinh(\pi y) \tag{28}$$

In this study, we set $w_0 = 1$. The successive adaptive refinement meshes are shown in Fig. 10. And Fig. 11 shows the adaptive numerical solution of the problem after 5th iteration with 485-node quadtree mesh.

(a) *Initial mesh: 16 nodes*   (b) *Iteration 01: 49 nodes*   (c) *Iteration 02: 138 nodes*

(d) *Iteration 03: 292 nodes*   (e) *Iteration 04: 343 nodes*   (f) *Iteration 05: 485 nodes*

Figure 10: *Adaptive refinement meshes by RBF-PolyFEM for Laplace problem.*

Figure 11: *The approximation solution u^h after 5th iteration for Laplace problem. (LHS: contour plot, RHS: 3D plot).*

The error in L^2-norm of the adaptive RBF-PolyFEM will then be compared with that from uniform finite element method and polygonal finite element method. The errors are shown in Fig. 12. It shows that the error from the proposed method is lower than the two others. When the node is increasing, the errors are more different. The reason is that the quadtree mesh is refined and focused on the critical region.
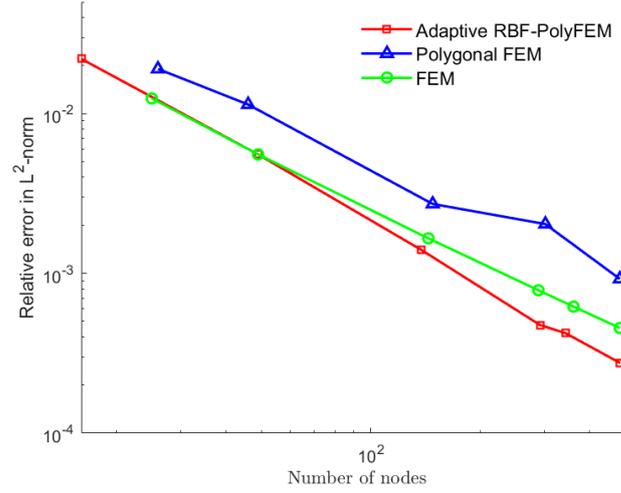
Figure 12: *Convergence of the relative error for different approaches for Laplacian problem.* (*log scale on the y-axis*).

3.3 Poisson problem I

A numerical test on the Poisson equation with the Dirichlet boundary condition is then carried out. The equation problem is given as

$$\nabla^2 u = f, \qquad \text{in } \Omega = (0,1)^2, \tag{29a}$$

$$u = 0, \qquad \text{on } \partial\Omega \tag{29b}$$

where the source term f is chosen such that the problem exact solution is obtained as [9]

$$u(\mathbf{x}) = x^{10}y^{10}(1-x)(1-y) \tag{30}$$

The regularized meshes (2:1 rule) are depicted in Fig. 13. The critical region is focused on mesh refinement. Fig. 14 shows the adaptive numerical solution of the problem with 825 nodes. In comparison, Fig. 15 illustrates the convergence of the relative error for three different approaches. It again shows that the result of the RBF-PolyFEM regarding the quadtree mesh refinement is by far better than that of the traditional FEM and polygonal finite element method.

3.4 Poisson problem II

The following example is solving the Poission equation with Dirichlet boundary condition under a different source term. The equation is given as follows

$$\nabla^2 u = f, \qquad \text{in } \Omega = (0,1)^2, \tag{31a}$$

$$u = 0, \qquad \text{on } \partial\Omega \tag{31b}$$

where the source term $f = 2\pi^2 \sin(\pi x)\sin(\pi y)$. Then the problem exact solution is given as

$$u(\mathbf{x}) = -\sin(\pi x)\sin(\pi y), \tag{32}$$

The regularized meshes (2:1 rule) are shown in Fig. 16. Fig. 17 shows the adaptive numerical solution of the problem with 1265 nodes. Fig. 18 shows the convergence of the relative L^2-

(a) *Initial mesh: 16 nodes* (b) *Iteration 01: 46 nodes* (c) *Iteration 02: 90 nodes*

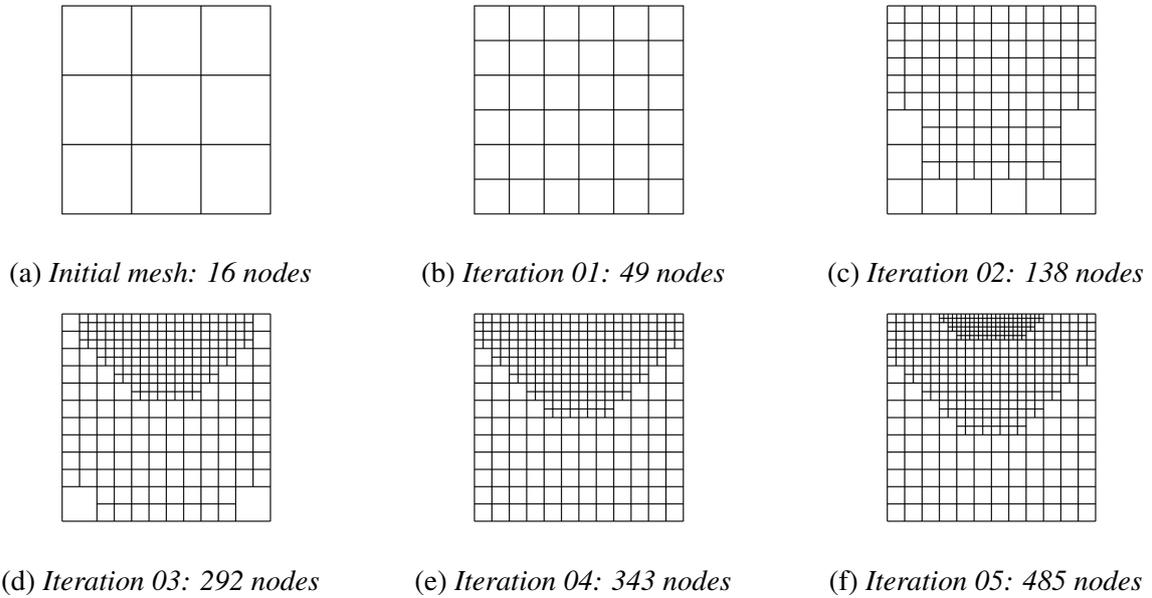(d) *Iteration 03: 177 nodes* (e) *Iteration 04: 375 nodes* (f) *Iteration 05: 825 nodes*
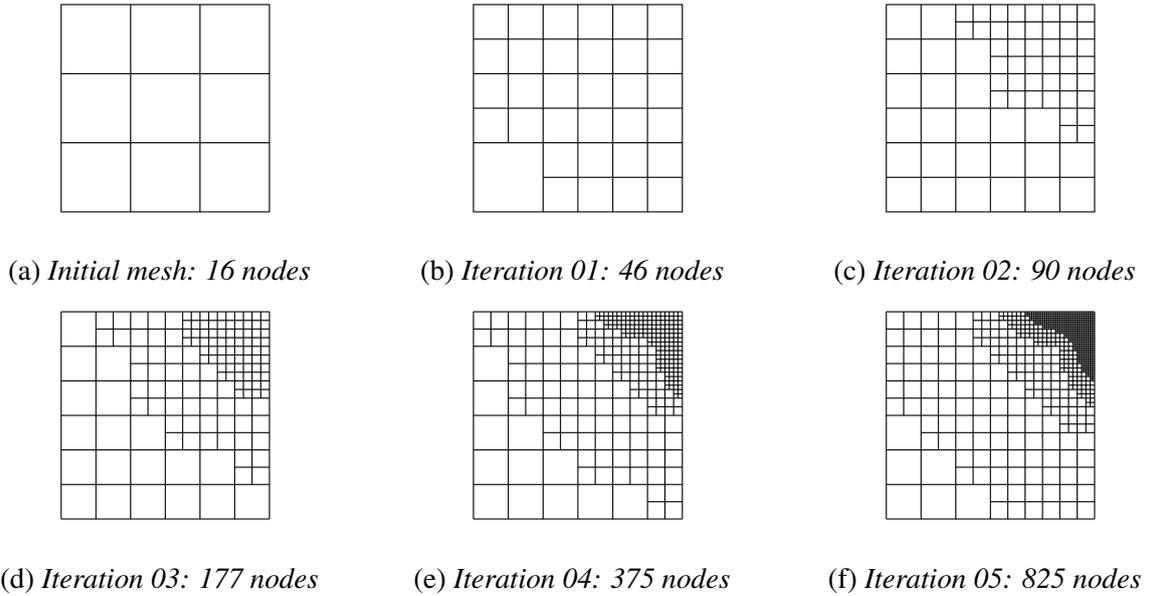
Figure 13: *Adaptive refinement meshes by RBF-polyFEM for Poisson problem I.*
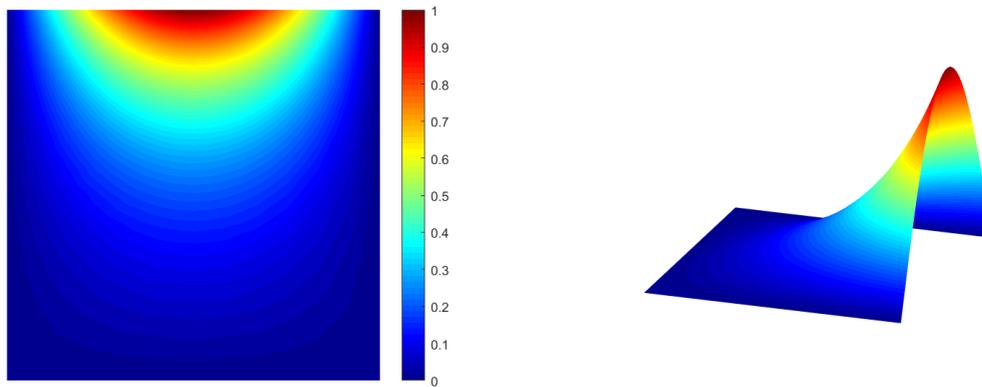

Figure 14: *The approximation solution u^h after 5th iteration for Poisson problem I. (LHS: contour plot, RHS: 3D plot).*

norm error for three different approaches. The results agree with that in the previous numerical examples.

3.5   L-Shape domain

The last example is solving the Laplace equation in L-shape domain. The exact solution is obtained as [1]

$$u(r, \theta) = r^{2/3} \sin\left(\frac{2\theta}{3}\right) \tag{33}$$

where r and θ are the polar coordinates. The essential boundary conditions on the boundary with the exact solution is imposed. Fig. 19 shows the refined meshes after computation and Fig. 20 depicts the contour plot of the solution. As expected, the quadtree mesh refines surrounding the sharp corner.

4   Conclusions

The recently developed method, the radial basis function-based finite element method (RBF-FEM) [12], could not well solve the steep gradient and singularity problems of partial differ-
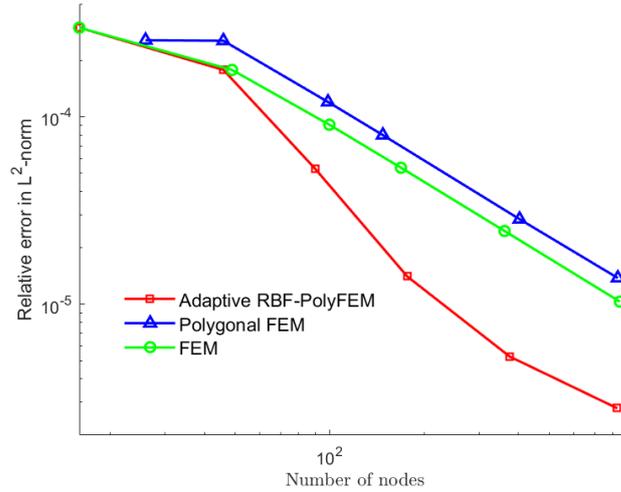
Figure 15: *Convergene of the relative error for different approaches for Poisson I problem. ($\log$ scale on the y-axis).*

(a) *Initial mesh: 9 nodes* (b) *Iteration 01: 25 nodes* (c) *Iteration 02: 81 nodes*

(d) *Iteration 03: 253 nodes* (e) *Iteration 04: 697 nodes* (f) *Iteration 05: 1265 nodes*
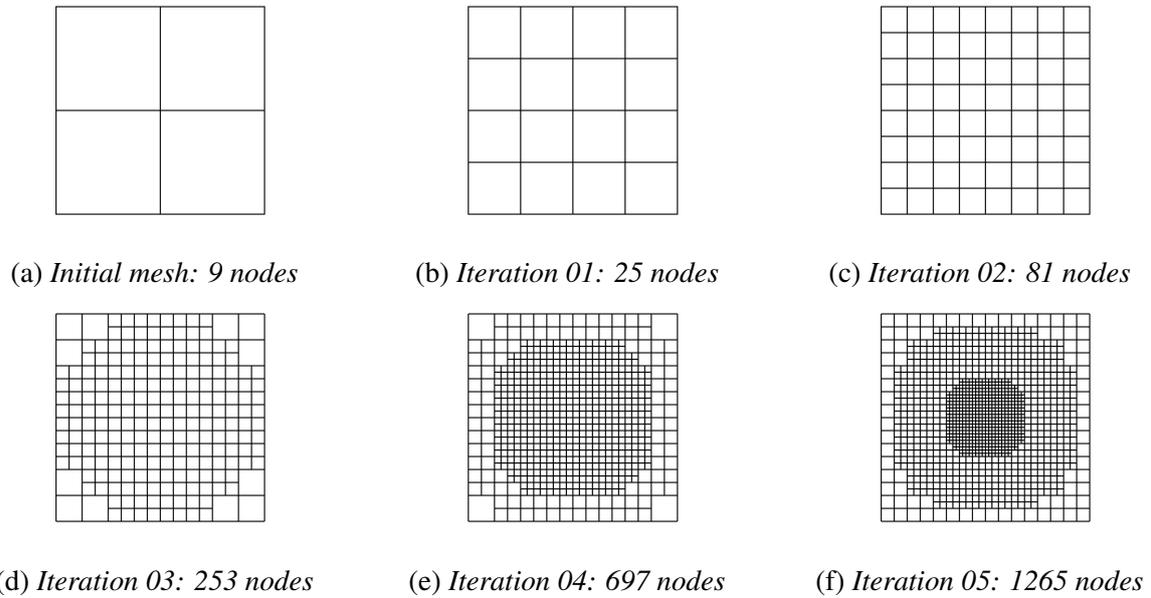
Figure 16: *Adaptive refinement meshes by RBF-polyFEM for Poisson problem II.*

ential equations by mesh refinement. The method even shows good results in regular uniform mesh, however, with a fine mesh, the time load will be increased. It is therefore inefficient and loading.

In this paper, we have presented a method that is a combination of the radial basis function-based finite element method (RBF-FEM) and the polygonal finite element method (as is the so-called RBF-polyFEM) to solve the steep gradient and singularity problem of the partial differential equation by quadtree mesh refinement. The sub-domain elements in which the node number equals four were applied by RBF-FEM and the other elements (five- and/or six-node elements) were utilized by the polygonal finite element method. Several test cases for the Laplace and Poisson equations were presented. The results were then compared with the finite element method with uniform meshes and with the polygonal finite element method. It is shown that the combination did not conflict in solving the numerical problems. The L^2 error norm results were
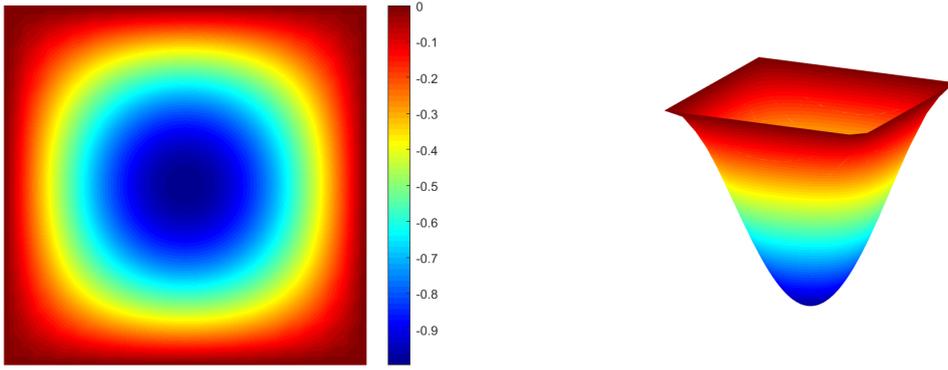
Figure 17: *The approximation solution u^h after 5th iteration for Poisson problem II. (LHS: contour plot, RHS: 3D plot).*
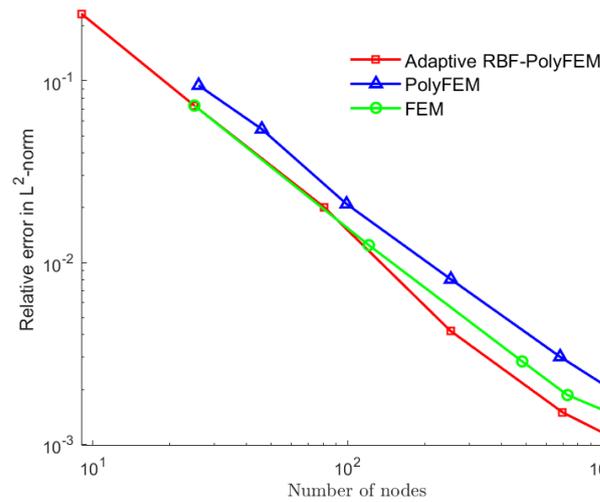
Figure 18: *Convergene of the relative error for different approaches for Poisson II problem. ($\log$ scale on the y-axis).*

sufficiently accurate, and the proposed approach could capture the sharp gradients and singularity in the problems. The implementation of this method for solving elasticity problems that involve crack discontinuities and singularities will be investigated. Applications of the method to more advanced physical engineering problems are also interested [11, 14].

References

[1] Carey, G. (1997). *Computational Grids, Generation, Adaptation, and Solution Strategy*. Taylor and Francis.

[2] Chau, K. N., Chau, K. N., Ngo, T., Hackl, K., and Nguyen-Xuan, H. (2018). A polytree-based adaptive polygonal finite element method for multi-material topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 332:712–739.

[3] Christ, N., Friedberg, R., and Lee, T. (1982). Weights of links and plaquettes in a random lattice. *Nuclear Physics B*, 210(3):337–346.

[4] Floater, M., Gillette, A., and Sukumar, N. (2014). Gradient Bounds for Wachspress Coordinates on Polytopes. *SIAM Journal on Numerical Analysis*, 52(1):515–532.

(a) *Initial mesh: 8 nodes*   (b) *Iteration 01: 21 nodes*   (c) *Iteration 02: 42 nodes*

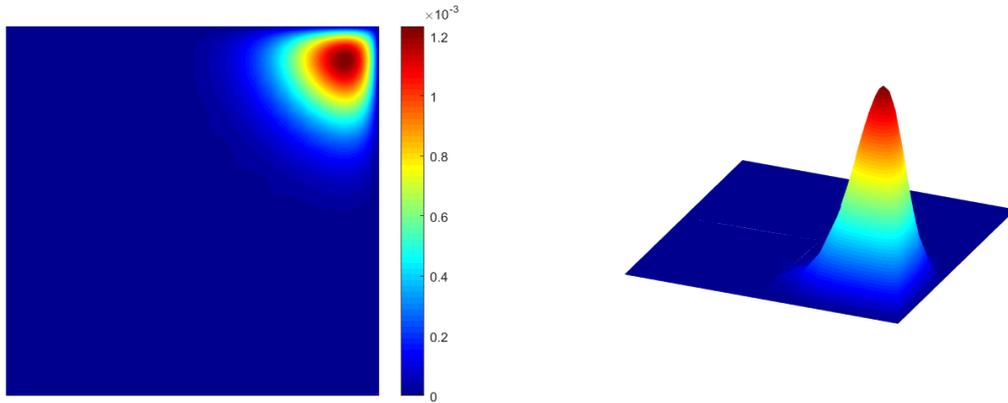(d) *Iteration 03: 75 nodes*   (e) *Iteration 04: 155 nodes*   (f) *Iteration 05: 349 nodes*
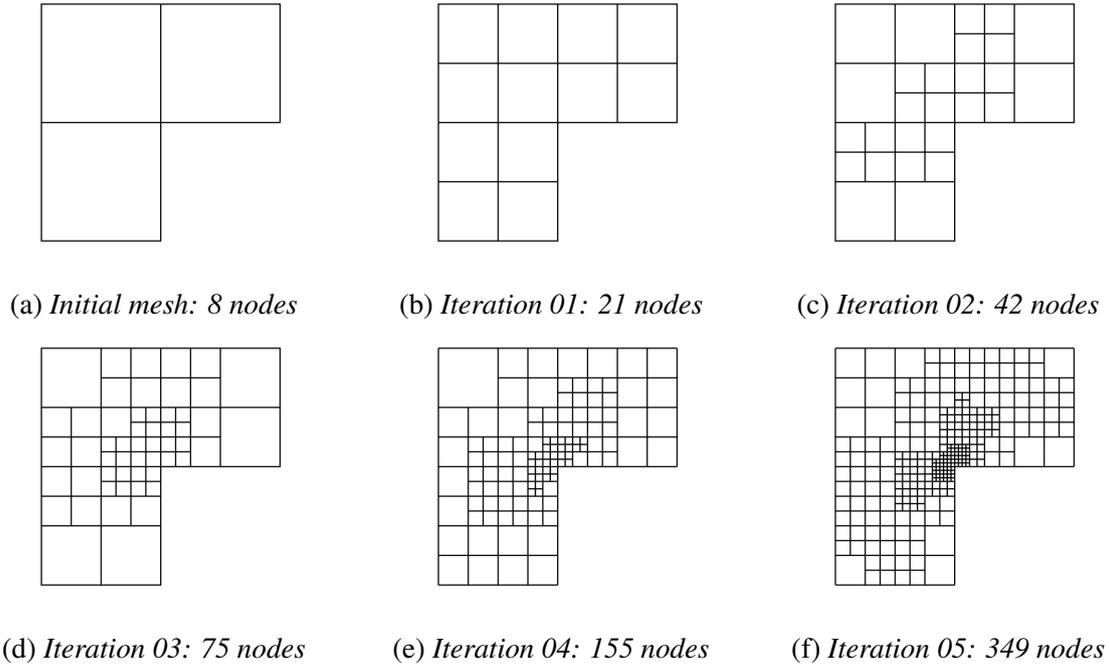
Figure 19: *Successive adaptive refinement by RBF-polyFEM for L-shape problem.*
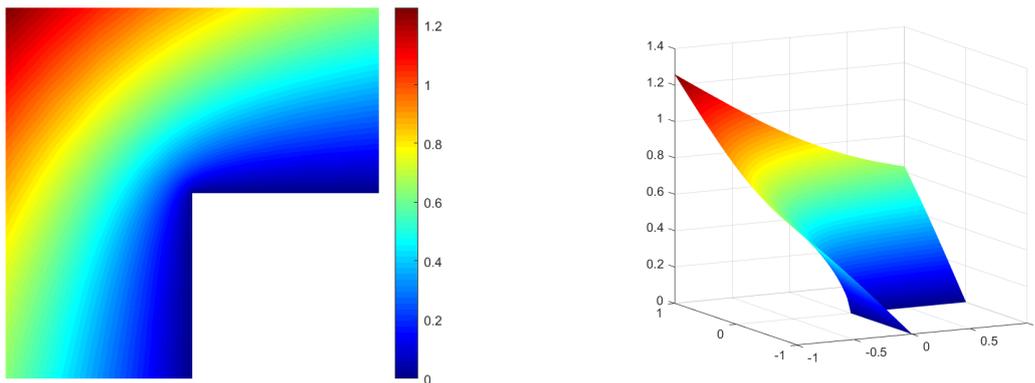
Figure 20: *The approximation solution u^h after 5th iteration for L-shape problem. (LHS: contour plot, RHS: 3D plot).*

[5] Funken, S. A. and Schmidt, A. (2020). Adaptive Mesh Refinement in 2D – An Efficient Implementation in Matlab. *Computational Methods in Applied Mathematics*, 20(3):459–479.

[6] Gupta, A. K. (1978). A finite element for transition from a fine to a coarse grid. *International Journal for Numerical Methods in Engineering*, 12(1):35–45.

[7] Han, X., Liu, Z., Li, G., Wang, Z., Wu, Z., and Wu, Y. (2019). Radial basis function and finite element method hybrid approach for three-dimensional electromagnetics problems. *International Journal of Applied Electromagnetics and Mechanics*, 60(3):445–455.

[8] Hiyoshi, H. and Sugihara, K. (1999). Two Generalizations of an Interpolant Based on Voronoi Diagrams. *International Journal of Shape Modeling*, 05(02):219–231.

[9] Kagan, P., Fischer, A., and Bar-Yoseph, P. Z. (2003). Mechanically based models: Adaptive refinement for B-spline finite element. *International Journal for Numerical Methods in Engineering*, 57(8):1145–1175.

[10] Kazemzadeh-Parsi, M. J. (2020). New elements with harmonic shape functions in adaptive mesh refinement. *Finite Elements in Analysis and Design*, 170:103366.

[11] Kien, D. N. and Zhuang, X. (2021). A deep neural network-based algorithm for solving structural optimization. *Journal of Zhejiang University-SCIENCE A*, 22(8):609–620.

[12] Kien, D. N. and Zhuang, X. (2022). Radial basis function based finite element method: Formulation and applications (submitted). *Engineering Structure*.

[13] Meyer, M., Barr, A., Lee, H., and Desbrun, M. (2002). Generalized Barycentric Coordinates on Irregular Polygons. *Journal of Graphics Tools*, 7(1):13–22.

[14] Nguyen-Thanh, V. M., Zhuang, X., and Rabczuk, T. (2020). A deep energy method for finite deformation hyperelasticity. *European Journal of Mechanics - A/Solids*, 80:103874.

[15] Nguyen-Xuan, H. (2017). A polytree-based adaptive polygonal finite element method for topology optimization. *International Journal for Numerical Methods in Engineering*, 110(10):972–1000.

[16] Nguyen-Xuan, H., Do, H. V., and Chau, K. N. (2018). An adaptive strategy based on conforming quadtree meshes for kinematic limit analysis. *Computer Methods in Applied Mechanics and Engineering*, 341:485–516.

[17] Nguyen-Xuan, H., Nguyen-Hoang, S., Rabczuk, T., and Hackl, K. (2017). A polytree-based adaptive approach to limit analysis of cracked structures. *Computer Methods in Applied Mechanics and Engineering*, 313:1006–1039.

[18] Palle, N. and Dantzig, J. A. (1996). An adaptive mesh refinement scheme for solidification problems. *Metallurgical and Materials Transactions A*, 27(3):707–717.

[19] Provatas, N., Goldenfeld, N., and Dantzig, J. (1999). Adaptive Mesh Refinement Computation of Solidification Microstructures Using Dynamic Data Structures. *Journal of Computational Physics*, 148(1):265–290.

[20] Reddy, J. N. (2018). *Introduction to the Finite Element Method*. McGraw-Hill Education, 4th edition.

[21] Samet, H. (1990). *Applications of Spatial Data Structures*. Addison-Wesley.

[22] Semenov, V. B., Ivanov, V. D., Kontorovich, V. K., Korytnik, S. A., and A. (1997). The non-Sibsonian interpolation : A new method of interpolation of the values of a function on an arbitrary set of points. *Computational Mathematics and Mathematical Physics*, 37:9–15.

[23] Sukumar, N. and Malsch, E. A. (2006). Recent Advances in the Construction of Polygonal Finite Element Interpolants. *Archives of Computational Methods in Engineering*, 13:129–163.

[24] Sukumar, N. and Tabarraei, A. (2004). Conforming polygonal finite elements. *International Journal for Numerical Methods in Engineering*, 61(12):2045–2066.

[25] Tabarraei, A. and Sukumar, N. (2005). Adaptive computations on conforming quadtree meshes. *Finite Elements in Analysis and Design*, 41(7-8):686–702.

[26] Wachspress, E.L. (1975). *A Rational Finite Element Basis*. Academic Press, New York.

[27] Warren, J. (1996). Barycentric coordinates for convex polytopes. *Advances in Computational Mathematics*, 6(1):97–108.

[28] Zienkiewicz, O., Taylor, R., and Zhu, J. (2013). *The Finite Element Method: its Basis and Fundamentals*. Elsevier.

[29] Zienkiewicz, O. C. and Zhu, J. Z. (1992a). The superconvergent patch recovery anda posteriori error estimates. Part 1: The recovery technique. *International Journal for Numerical Methods in Engineering*, 33(7):1331–1364.

[30] Zienkiewicz, O. C. and Zhu, J. Z. (1992b). The superconvergent patch recovery anda posteriori error estimates. Part 2: Error estimates and adaptivity. *International Journal for Numerical Methods in Engineering*, 33(7):1365–1382.

[31] Zou, Y., Lei, G., Shao, K., Guo, Y., Zhu, J., and Chen, X. (2015). Hybrid approach of radial basis function and finite element method for electromagnetic problems. *IEEE Transactions on Magnetics*, 51(3):1–4.

Studies on interface of pipe joints based on exponential softening bond-slip law under torsional loads

Hong Yuan[1], Jun Han[1], *Ziyong Mo[1], †Lan Zeng[1]

[1]MOE Key Laboratory of Disaster Forecast and Control in Engineering, School of Mechanics and Construction Engineering, Jinan University, China

*Presenting author: moxph33@163.com
†Corresponding author: zenglan@jnu.edu.cn

Abstract

The mechanical behavior and debonding process of pipe joints' interface are the key points for pipe system. In order to better understand and describe the debonding failure of pipe joints subjected to torsional loads for safety design, the theoretical and numerical studies have been conducted. Firstly, based on the exponential softening bond-slip law, the analytical expressions of the interfacial shear stress and the load-displacement relationship at loaded end were obtained. Thus the shear stress propagation and the debonding progress of the whole interface for different bond lengths could be predicted. Secondly, a simplified interface bond-slip law was used by changing the exponential softening law into a bilinear model. The analytical solutions for the simplified model were also obtained. Based on the analytical solutions, the influence of bond length and stiffness on load-displacement curve and ultimate load were discussed. The stress transfer mechanism, the interface crack propagation and the ductility behavior of the joints were further explained.

Keywords: interface; torsion; load-displacement curve; bond-slip law; pipe joints

Introduction

Pipe structures are a very important structural form for energy industry and construction industry. The limitations of the overall system performance usually come from the capacity of pipe joints [1]. Therefore, the pipe joints play the most important role in the overall integrity of most piping systems [2-4].

Based on the mechanics of composite materials and the maximum strain failure criterion, an analytical and experimental study was conducted to investigate the elastic and failure behavior of composite laminated pipe under torsion [5]. Based on the general composite shell theory, Zou et al. [6] studied the stress concentrations at and near the end of the joints as functions of various parameters, such as the overlap length, and thickness of the adhesive layer. Pugno et al. [7] confirmed that the maximum stresses were attained at the ends of the adhesive and that the peak of maximum stress was reached at the end of the stiffer tube and does not tend to zero as the adhesive length approaches infinity. Cheng [8] developed an adhesively bonded smart composite pipe joint system by integrating electromechanical coupling piezoelectric layers with the connection coupler. He et al. [9] studied the failure analysis for thermoplastic composite pipes under combined pure torsion and thermomechanical loading from operational thermal gradients.

A finite element analysis was used to calculate the residual thermal stresses generated by cooling down from the adhesive cure temperature and a nonlinear analysis incorporating the

nonlinear adhesive behavior was performed [10]. Based on a parametric study conducted by 2D and 3D finite element analysis, Hosseinzadeh et al. [11-14] developed a simple method for assessing the behavior of adhesively bonded tubular joints under torsion. A finite difference method was utilized to solve the system of equilibrium equations and it was modeled as a separate 3D elastic body without the uniform stress assumption [15]. Considering individual and combined effect of internal pressure and torsional loadings, Baishya et al. [16] analyzed the failure process of the laminated composite tubes by finite element analysis.

Studies of interfacial mechanical properties are mainly for simple shear model. Yuan et al. [17-19] gave analytical solutions in closed-form of interfacial behavior of adhesive joints. Other researchers made some improvements by considering interfacial normal stress [20-23].

The latest experimental study proposes that exponential softening may appear for interface of some bonded joints [24, 25]. Based on the exponential softening bond-slip law, this paper studies the interface behavior of pipe joints under torsion loads. Closed-form solutions are given.

Interface model of pipe joint

2.1. Interface model

The inner and outer pipe is bonded together by a thin and soft adhesive layer shown in Fig. 1. Here the inner and outer pipe are defined as pipe 1 and 2 respectively. Due to symmetry, only the right half of the pipe joint is considered. We assume that the distance between the left end of pipe 1 and the right end of the pipe 2 is L. For the sake of clarification, the bond length in this paper is denoted by L for only the right half of the pipe joint is considered.

(a) A cross-sectional view

(b) A side view
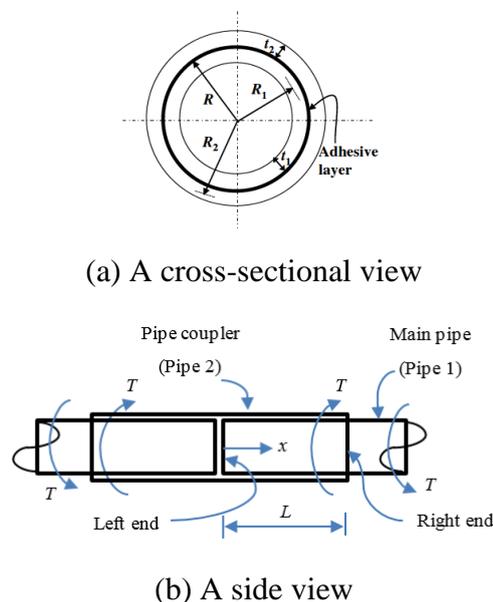
Fig. 1. Adhesively bonded pipe joint.

Before starting the derivations, the following assumptions can be made for the simplicity of problems:

    (1) The adherents are homogeneous and linear elastic;

    (2) The adhesive is only exposed to shear forces;

    (3) The torsion load carried by the thin and soft adhesive layer is ignored and the external torsion load is assumed to be resisted by the main pipe and coupler pipe;

(4) Local bending effects in the pipe joint under torsion load are neglected.

According to the classical torsion theory, the internal torsion T_1 and T_2 of the pipe and the coupler can be expressed respectively as follows:

$$T_1 = \varphi_1 G_1 J_1 \tag{1}$$

$$T_2 = \varphi_2 G_2 J_2 \tag{2}$$

Where G_1 and G_2, φ_1 and φ_2 are the shear modulus and the rotation angle of pipe 1 and pipe 2 respectively; J_1 and J_2 are the polar moment of inertia of the thin-walled pipe 1 and pipe 2 respectively written as follows:

$$J_1 = 2\pi R_1^3 t_1 \tag{3}$$

$$J_2 = 2\pi R_2^3 t_2 \tag{4}$$

In which, t_1 and t_2 are the thickness of the thin-walled pipe 1 and pipe 2 respectively; R_1 and R_2 are the average radius of pipe 1 and pipe 2 respectively (Figs. 1a).

According to the assumption above, the torsion load carried by the soft and thin adhesive layer is ignored. Thus, the equilibrium between external and internal torsion load in the pipe joint requires:

$$T_1 + T_2 = 0 \tag{5}$$

2.2. Governing equations

If at the given cross-section, the rotations of pipe 1 and pipe 2 are different from each other, a relative rotation occurs accompanied by a circumferential relative displacement at the bond layer. Let's introduce the relative interfacial rotation φ, which equals to the difference of the individual rotation angle of pipe 1 and pipe 2 at the cross-section x as illustrated in Fig. 2. Consider the torsional equilibrium in pipe 1 of an infinitely small section dx as illustrated in Fig. 2:

$$2\pi R \tau R dx = dT_1 \tag{6}$$
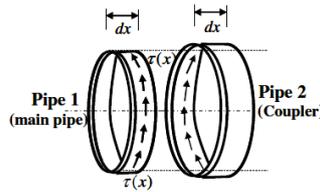
Fig. 2. Equilibrium of the local interfacial shear stresses.

Where τ is the interfacial shear stress along the circumferential direction and R is the distance between the center of the pipe and mid-height of the adhesive layer which can be calculated by:

$$R = \frac{1}{2}\left[\left(R_1 + \frac{t_1}{2}\right) + \left(R_2 - \frac{t_2}{2}\right)\right] \tag{7}$$

Denote this relative slip at the bond layer interface along circumferential direction as δ. This interfacial slip δ can thus be expressed as a function of the relative interfacial rotation φ as follow:

$$\delta = R\varphi = R\varphi_1 - R\varphi_2 \tag{8}$$

By introducing two parameters of local bond strength τ_f and interfacial fracture energy G_f, we have:

$$\frac{d^2\delta}{dx^2} - \frac{2G_f}{\tau_f^2}\lambda^2 f(\delta) = 0 \tag{9}$$

$$\varphi_1' = \frac{\tau_f^2}{2G_f\lambda^2}\frac{2\pi R^2}{G_1 J_1}\frac{d\delta}{dx} \tag{10}$$

where

$$\lambda^2 = 2\pi R^3 \frac{G_1 J_1 + G_2 J_2}{G_1 J_1 G_2 J_2}\frac{\tau_f^2}{2G_f} \tag{11}$$

Substituting Eq. (10) into (1), the relationship of T_1 and derivative of δ can be obtained:

$$T_1 = 2\pi R^2 \frac{\tau_f^2}{2G_f\lambda^2}\frac{d\delta}{dx} \tag{12}$$

Eq. (9) is the governing differential equation of the adhesive bonded joint in Fig. 2. When the local bond-slip model is found, this equation can be solved.

2.3. Bond-slip model

The exponential bond-slip law can be described as the dashed line in Fig. 3. The interfacial shear stress increases linearly to τ_f at which the value of the slip is denoted by δ_1. It is called an elastic stage. Then interface softening appears and the interfacial shear stress decays exponentially with the interfacial slip. It is called a softening stage. The mathematical expressions of the interfacial bond-slip law in Fig. 3 are:

$$\tau = f(\delta) = \begin{cases} \dfrac{\tau_f}{\delta_1}\delta & (0 \le \delta \le \delta_1) \\ \tau_f e^{-2\alpha^2\left(\frac{\delta}{\delta_1}-1\right)} & (\delta > \delta_1) \end{cases} \tag{13}$$
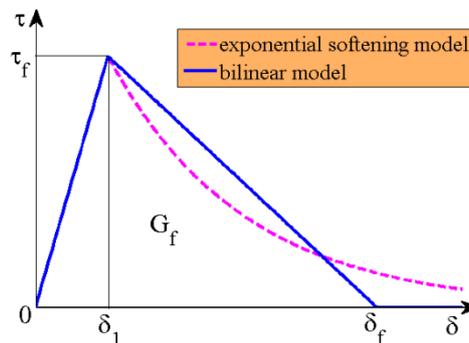
Fig. 3. Bond-slip models.

The bilinear model shown as blue real line in Fig. 3 which features a linear ascending branch followed by a linear descending branch provides a close approximation. It is a simplified model of exponential softening model by letting G_f be equal. According to this model, the bond shear stress increases linearly with the interfacial slip which is the same as exponential model.

Interfacial softening (or micro-cracking) then starts with the shear stress reducing linearly with the interfacial slip. The shear stress reduces to zero when the slip exceeds δ_f, signifying the shear fracture (debonding or macro-cracking) of a local bond element. This bond-slip model shown in Fig. 3 is mathematically described by the following:

$$\tau = f(\delta) = \begin{cases} \dfrac{\tau_f}{\delta_1}\delta & (0 \le \delta \le \delta_1) \\[2mm] \dfrac{\tau_f}{\delta_f - \delta_1}(\delta_f - \delta) & (\delta_1 < \delta \le \delta_f) \\[2mm] 0 & (\delta > \delta_f) \end{cases} \tag{14}$$

The expression α^2 which is a positive coefficient characterizing the exponential decay could be obtained by letting the interfacial fracture energy G_f in Fig. 3 be equal:

$$\alpha^2 = \frac{\delta_1}{\delta_f - \delta_1} \tag{15}$$

3. Analysis of the debonding process for the exponential model

3.1. Elastic stage

As small loads, there is no interfacial softening or debonding along the interface, so the entire length of the interface is in an elastic stress state. Substituting the relationship of Eq. (13) for the case of $0 \le \delta \le \delta_1$ into (9), the following differential equation is obtained:

$$\delta''(x) - \lambda_1^2 \delta(x) = 0 \quad (0 \le \delta \le \delta_1) \tag{16}$$

where

$$\lambda_1^2 = 2\pi R^3 \frac{G_1 J_1 + G_2 J_2}{G_1 J_1 G_2 J_2} \frac{\tau_f}{\delta_1} \tag{17}$$

And the boundary conditions are:

$$\varphi_1'(0) = 0 \tag{18}$$

$$\varphi_1'(L) = \frac{T}{G_1 J_1} \tag{19}$$

The solution of Eq. (16) for the relative shear displacement as well as the shear stress of the adhesive layer can be written in the form:

$$\delta(x) = \frac{T\delta_1 \lambda_1}{2\pi R^2 \tau_f} \frac{\cosh(\lambda_1 x)}{\sinh(\lambda_1 L)} \tag{20}$$

$$\tau(x) = \frac{T\lambda_1}{2\pi R^2} \frac{\cosh(\lambda_1 x)}{\sinh(\lambda_1 L)} \tag{21}$$

The slip at the loaded end (i.e. the value of δ at $x=L$) is defined as the displacement of the bonded joint and is denoted by Δ. According to this definition, the relationship of the load-displacement can be obtained from Eq. (20):

$$T = \frac{2\pi R^2 \tau_f}{\delta_1 \lambda_1} \tanh(\lambda_1 L)\Delta \tag{22}$$

3.2. Elastic-softening stage

As the load increases, the interfacial slip reaches δ_1 at the loaded end and softening appears at $x=L$, thus the whole interface is in an elastic-softening stage. The load T increases as the length of the softening region a increases. Substituting the relationship given in Eq. (13) into (9) gives differential equation (16) for the elastic region and the following equation (23) for the softening region.

$$\delta''(x) - \lambda_2^2 e^{-2\alpha^2 \frac{\delta(x)}{\delta_1}} = 0 \quad (\delta > \delta_1) \tag{23}$$

where

$$\lambda_2^2 = 2\pi R^3 \frac{G_1 J_1 + G_2 J_2}{G_1 J_1 G_2 J_2} \tau_f e^{2\alpha^2} \tag{24}$$

With the boundary conditions of Eqs. (18), (19) and continuous conditions:

$$\delta(L-a) = \delta_1 \tag{25}$$

$$\delta'(x) \text{ is continues at } x = L-a \tag{26}$$

The solution for the elastic region of the interface ($0 \le \delta \le \delta_1$, *i.e.* $0 \le x \le L$-a) is given by:

$$\delta(x) = \frac{\delta_1 \cosh(\lambda_1 x)}{\cosh[\lambda_1(L-a)]} \tag{27}$$

$$\tau(x) = \frac{\tau_f \cosh(\lambda_1 x)}{\cosh[\lambda_1(L-a)]} \tag{28}$$

and the solution for the softening region of the interface ($\delta > \delta_1$, *i.e.* L-$a \le x \le L$) is shown as follows:

$$\delta(x) = \frac{1}{n} \ln\left(\frac{2m}{nc_1}\right) + \frac{2}{n} \ln\left\{\cosh\left[\frac{n}{2}\sqrt{c_1}(x-c_2)\right]\right\} \tag{29}$$

$$\tau(x) = \tau_f e^{-2\alpha^2\left[\frac{\delta(x)}{\delta_1}-1\right]} \tag{30}$$

where

$$m = \lambda_2^2 \tag{31}$$

$$n = \frac{2\alpha^2}{\delta_1} \tag{32}$$

Based on the conditions Eqs. (25) and (26), the constants c_1 and c_2 can be obtained:

$$c_1 = \left\{\delta_1 \lambda_1 \tanh[\lambda_1(L-a)]\right\}^2 + \frac{2m}{n}e^{-n\delta_1} \tag{33}$$

$$c_2 = L - a - \frac{2}{n\sqrt{c_1}} \text{arccosh}\left(\sqrt{\frac{nc_1}{2m}e^{n\delta_1}}\right) \tag{34}$$

The expression of slip at loaded end can be obtained from Eq. (29) when $x=L$:

$$\Delta = \delta(L) = \frac{1}{n}\ln\left(\frac{2m}{nc_1}\right) + \frac{2}{n}\ln\left\{\cosh\left[\frac{n}{2}\sqrt{c_1}(L-c_2)\right]\right\} \tag{35}$$

Substituting Eqs. (19) and (29) into (10) yields:

$$T = \frac{2\pi R^2 \tau_f}{\delta_f \lambda^2}\sqrt{-\frac{2m}{n}e^{-n\Delta} + c_1} \tag{36}$$

For exponential model, the expression of T above can be rewritten as:

$$T = \frac{2\pi R^2 \tau_f}{\delta_f \lambda^2}\sqrt{c_1}\tanh\left[\frac{n}{2}\sqrt{c_1}(L-c_2)\right] \tag{37}$$

T reaches its maximum when L is large enough so Eqs. (37) and (33) converge to:

$$T_u = \frac{2\pi R^2 \tau_f}{\delta_f \lambda^2}\sqrt{c_1} \tag{38}$$

where

$$c_1 = (\delta_1 \lambda_1)^2 + \frac{2m}{n}e^{-n\delta_1} \tag{39}$$

The length of the interface that is mobilized to resist the applied load is generally referred to as the effective bond length. This effective bond length is defined here as the bond length over which the shear stresses offer a total resistance which is at least 97% of the applied load for a joint with an infinite bond length. The effective bond length when T_u is reached can be obtained from Eqs. (34) and (37)-(39) to give:

$$l_e = a_e + \frac{2}{\lambda_1} \tag{40}$$

where

$$a_e = \frac{4}{n\sqrt{c_1}} - \frac{2}{n\sqrt{c_1}}\text{arccosh}\left(\sqrt{\frac{nc_1}{2m}e^{n\delta_1}}\right) \tag{41}$$

3.3. Softening stage

As the load increases, the peaks of shear stress move to the unloaded end ($x=0$). When the interfacial slip at $x=0$ reaches δ_1, the whole interface enters into softening zone. This stage is governed by Eq. (23) with boundary conditions (18) and (19).

Based on the boundary conditions, the solution for the interfacial slip and the shear stress of the adhesive layer can be written in the form:

$$\delta(x) = \frac{1}{n}\ln\left(\frac{2m}{nc_3}\right) + \frac{2}{n}\ln\left\{\cosh\left[\frac{n}{2}\sqrt{c_3}(x-c_4)\right]\right\} \tag{42}$$

$$\tau(x) = \tau_f e^{-2\alpha^2\left[\frac{\delta(x)}{\delta_1}-1\right]} \tag{43}$$

where the constants c_3 and c_4 are shown as follows:

$$c_3 \tanh^2\left[\frac{n\sqrt{c_3}}{2}(L-c_4)\right] = \left(\frac{T\delta_f\lambda^2}{2\pi R^2\tau_f}\right)^2 \tag{44}$$

$$c_4 = 0 \tag{45}$$

Substituting Eqs. (19) and (42) into (10) yields:

$$T = \frac{2\pi R^2\tau_f}{\delta_f\lambda^2}\sqrt{-\frac{2m}{n}e^{-n\Delta}+c_3} \tag{46}$$

The expression of slip at loaded end can be obtained from Eq. (42) when $x=L$:

$$\Delta = \frac{1}{n}\ln\left(\frac{2m}{nc_3}\right) + \frac{2}{n}\ln\left\{\cosh\left[\frac{n}{2}\sqrt{c_3}(L-c_4)\right]\right\} \tag{47}$$

4. Analysis of the debonding process for the bilinear model

4.1. Elastic stage

The elastic stage is the same as the elastic stage in 3.1, thus the expressions of the interfacial slip, shear stress and the relationship of the load-displacement are the same.

4.2. Elastic-softening stage

As the load increases, softening commences at the loaded end once the shear stress reaches τ_f at $x=L$. The load T increases as the softening length a increases. Substituting the relationship given in Eq. (14) into (9) gives Eqs. (16) for the elastic region and (48) for the softening region.

$$\delta''(x) - \lambda_3^2\left[\delta_f - \delta(x)\right] = 0 \quad (\delta_1 < \delta \leq \delta_f) \tag{48}$$

where

$$\lambda_3^2 = 2\pi R^3 \frac{G_1 J_1 + G_2 J_2}{G_1 J_1 G_2 J_2}\frac{\tau_f}{\delta_f - \delta_1} \tag{49}$$

With the same boundary conditions (18), (19) and continuous conditions (25), (26) used in 3.2. The solution for the elastic region of the interface ($0 \leq \delta \leq \delta_1$, i.e. $0 \leq x \leq L-a$) is the same as in 3.2. And the solution for the softening region of the interface ($\delta_1 < \delta \leq \delta_f$, i.e. $L-a \leq x \leq L$) is given by:

$$\delta(x) = (\delta_f - \delta_1)\left\{\frac{\lambda_3}{\lambda_1}\tanh\left[\lambda_1(L-a)\right]\sin\left[\lambda_3(x-L+a)\right] - \cos\left[\lambda_3(x-L+a)\right] + \frac{\delta_f}{\delta_f-\delta_1}\right\} \tag{50}$$

$$\tau(x) = -\tau_f\left\{\frac{\lambda_3}{\lambda_1}\tanh\left[\lambda_1(L-a)\right]\sin\left[\lambda_3(x-L+a)\right] - \cos\left[\lambda_3(x-L+a)\right]\right\} \tag{51}$$

Substituting Eqs. (19) and (50) into (10) yields:

$$T = \frac{2\pi R^2 \tau_f}{\lambda_3} \left\{ \frac{\lambda_3}{\lambda_1} \tanh\left[\lambda_1(L-a)\right]\cos(\lambda_3 a) + \sin(\lambda_3 a) \right\} \tag{52}$$

The expression of the slip at the loaded end could be got from Eq. (50) when $x=L$:

$$\Delta = (\delta_f - \delta_1) \left\{ \frac{\lambda_3}{\lambda_1} \tanh\left[\lambda_1(L-a)\right]\sin(\lambda_3 a) - \cos(\lambda_3 a) + \frac{\delta_f}{\delta_f - \delta_1} \right\} \tag{53}$$

During this stage, the load-displacement curve could be drawn from Eqs. (52) and (53). When the interfacial slip increases to δ_f at $x=L$ and the slip at $x=0$ less than δ_1, we can get $L > \frac{\pi}{2\lambda_3}$ and the interface enters into elastic-softening-debonding stage. When the slip at $x=0$ reaches δ_1 and the slip at loaded end less than δ_f, we can get $L < \frac{\pi}{2\lambda_3}$ and the interface enters into softening stage. Therefore, there exists a critical bond length to distinguish the coming failure process:

$$L_{cr} = \frac{\pi}{2\lambda_3} \tag{54}$$

For bilinear model, T reaches its maximum when the derivative of Eq. (52) with respect to a equal zero. Therefore, a at the ultimate load can be found from the following relationship:

$$\tanh\left[\lambda_1(L-a)\right] = \frac{\lambda_3}{\lambda_1}\tan(\lambda_3 a) \tag{55}$$

Substituting Eq. (55) into (52) yields:

$$T = \frac{2\pi R^2 \tau_f}{\lambda_3} \frac{\delta_f}{\delta_f - \delta_1}\sin(\lambda_3 a) \tag{56}$$

It can be shown from Eq. (55) that for large values of L Eq. (56) converges to:

$$T_u = \frac{2\pi R^2 \tau_f}{\lambda} \tag{57}$$

Based on same definition of effective bond length in 3.2 and considering that $\tanh(2)\approx0.97$, the effective bond length when T_u is reached can be obtained from Eqs. (55)-(57) to give:

$$l_e = a_e + \frac{1}{2\lambda_1}\ln\left[\frac{\lambda_1 + \lambda_3\tan(\lambda_3 a_e)}{\lambda_1 - \lambda_3\tan(\lambda_3 a_e)}\right] \tag{58}$$

where

$$a_e = \frac{1}{\lambda_3}\arcsin\left[0.97\sqrt{\frac{\delta_f - \delta_1}{\delta_f}}\right] \tag{59}$$

4.3. $L>L_{cr}$

4.3.1. Elastic-softening-debonding stage

If $L>L_{cr}$, as the load increases the interfacial slip at loaded end reaches δ_f and debonding (or

macro-cracking or fracture) commences and propagates along the interface. At the initiation of debonding $\Delta=\delta_f$ and by making use of this condition, the corresponding value of a, denoted by a_d, can be obtained from Eq. (53) as:

$$\frac{\lambda_3}{\lambda_1}\tanh\left[\lambda_1(L-a_d)\right]\sin(\lambda_3 a_d)-\cos(\lambda_3 a_d)=0 \tag{60}$$

As debonding propagates, the peak shear stress moves towards the unloaded end. Assuming that the debonded length of the interface starting at the loaded end is d, Eqs. (27), (28), (50) and (51) are still valid if replacing L by L-d. Therefore, the load-displacement relationship can be written as:

$$T=\frac{2\pi R^2 \tau_f}{\lambda_3}\left\{\frac{\lambda_3}{\lambda_1}\tanh\left[\lambda_1(L-d-a)\right]\cos(\lambda_3 a)+\sin(\lambda_3 a)\right\} \tag{61}$$

$$\Delta=\delta_f+\frac{\delta_f T\lambda^2}{2\pi R^2 \tau_f}d \tag{62}$$

As the interfacial shear stress at x=L-d is zero, the following relationship can be obtained:

$$\frac{\lambda_3}{\lambda_1}\tanh\left[\lambda_1(L-d-a)\right]\sin(\lambda_3 a)-\cos(\lambda_3 a)=0 \tag{63}$$

Substituting Eq. (63) into (61) yields the following simplified form:

$$T=\frac{2\pi R^2 \tau_f}{\lambda_3}\frac{1}{\sin(\lambda_3 a)} \tag{64}$$

At the end of this stage, the softening-debonding stage starts when L-d=a_u. Substituting the relation into Eq. (63) yields:

$$a_u=\frac{\pi}{2\lambda_3} \tag{65}$$

Moreover, Eq. (64) can be written as:

$$T=\frac{2\pi R^2 \tau_f}{\lambda_3} \tag{66}$$

4.3.2. Softening-debonding stage

This stage is governed by Eq. (48) with boundary conditions of Eq. (18) and:

$$\varphi_1'(a)=\frac{T}{G_1 J_1} \tag{67}$$

$$\delta(a)=\delta_f \tag{68}$$

The following solution can thus be found:

$$a=\frac{\pi}{2\lambda_3}=a_u \tag{69}$$

$$\delta(x) = \delta_f - \frac{\delta_f T\lambda^2}{2\pi R^2 \tau_f \lambda_3} \cos(\lambda_3 x) \quad (0 \leq x \leq a_u) \tag{70}$$

$$\tau(x) = \frac{\lambda_3 T}{2\pi R^2 \cos(\lambda_3 x)} \quad (0 \leq x \leq a_u) \tag{71}$$

It can be concluded from Eq. (69) that the length of softening zone remains constant during this stage. The load-displacement relationship can be simply obtained by displacement superposition along the bonded joint:

$$\Delta = \delta_f + \frac{\delta_f T\lambda^2}{2\pi R^2 \tau_f}(L - a_u) \tag{72}$$

4.4. L<L_cr

4.4.1. Softening stage

As the load increases, the peaks of shear stress move to the unloaded end. When the interfacial slip at $x=0$ reaches δ_1, the whole interface enters into softening zone. This stage is governed by Eq. (48) with boundary conditions of Eqs. (18) and (19). The following solution can be obtained:

$$\delta(x) = \delta_f - \frac{T\lambda^2 \delta_f \cos(\lambda_3 x)}{2\pi R^2 \tau_f \lambda_3 \sin(\lambda_3 L)} \tag{73}$$

$$\tau(x) = \frac{T\lambda_3 \cos(\lambda_3 x)}{2\pi R^2 \sin(\lambda_3 L)} \tag{74}$$

The expression of the slip at the loaded end could be obtained from Eq. (73) when $x=L$:

$$\Delta = \delta_f - \frac{T\lambda^2 \delta_f}{2\pi R^2 \tau_f \lambda_3}\cot(\lambda_3 L) \tag{75}$$

5. Numerical simulations

The typical inner diameter and thickness are assumed to be 290 and 10 mm for the main pipe, respectively. And the inner diameter and thickness of the coupler are assumed to be 311 and 15 mm, respectively. The shear moduli G_1 and G_2 are assumed to be 28 GPa for both main pipe and coupler. The parameters for interfacial bond-slip laws are identified as: τ_f=7.2 MPa, δ_1=0.034 mm, δ_f=0.16 mm.

According to the material properties and geometry parameters given above, the critical bond length for bilinear model can be calculated as L_{cr}=85 mm. Therefore, bond length of 50 and 100 mm are chosen.

5.1. Load-displacement curves

5.1.1. Load-displacement curves for exponential model

The load-displacement curve for exponential model is shown in Fig. 4. OA, AB and BC are elastic, elastic-softening and softening stages, respectively.

(a) *L*=50 mm   (b) *L*=100 mm
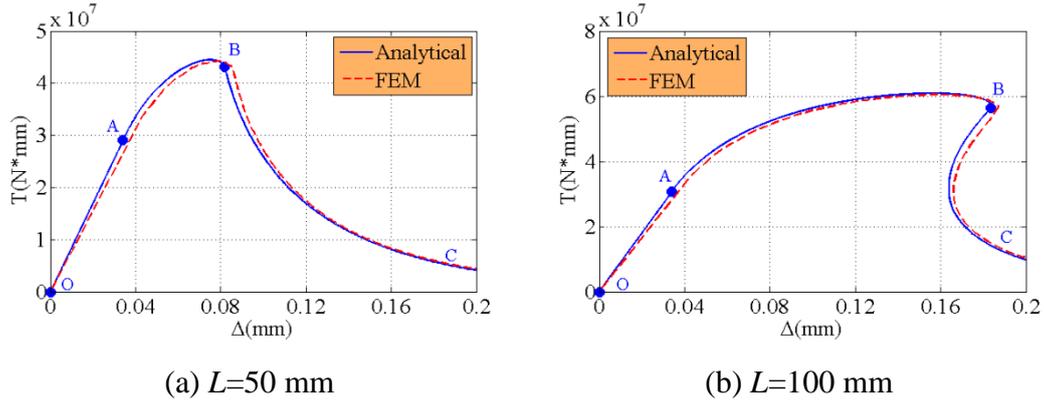
Fig. 4. Load-displacement curves for exponential model.

5.1.2. Load-displacement curves for bilinear model

When the bond length is shorter than L_{cr}, take *L*=50 mm and the load-displacement curve is shown in Fig. 5(a). OA, AB and BC are elastic, elastic-softening and softening stages, respectively. When the bond length is longer than L_{cr}, take *L*=100 mm and the load-displacement curve is shown in Fig. 5(b). OA, AB, BC and CD are elastic, elastic-softening, elastic-softening-debonding and softening-debonding stages, respectively. The FEA results by using commercial software ABAQUS are also given for comparison in Fig. 4 and Fig. 5. In ABAQUS modelling, C3D8R, which is an 8-node linear brick element with reduced integration scheme, is used for both main pipe and coupler. And for adhesive layer, COH3D8, which is an 8-node three-dimensional cohesive element, is used.

(a) *L*=50 mm   (b) *L*=100 mm
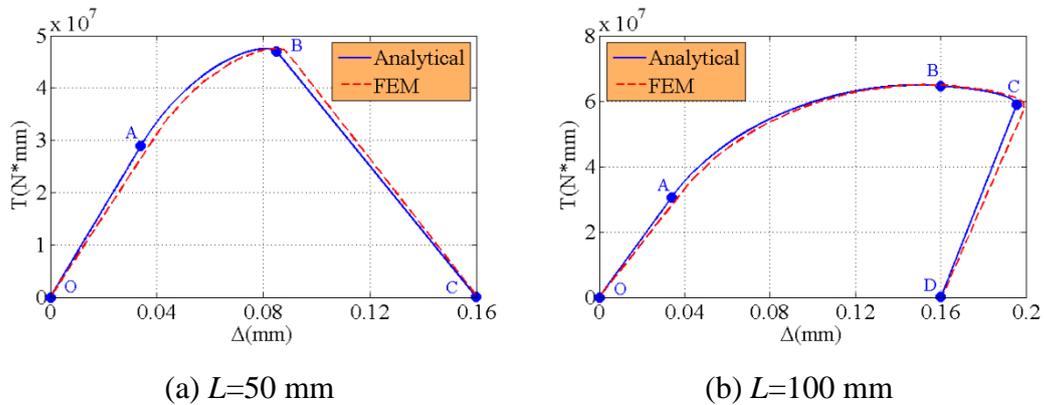
Fig. 5. Load-displacement curves for bilinear model.

5.2. Shear stress distribution

The shear stress distribution for exponential model is shown in Fig. 6. When the load is small, the interfacial shear stress at loaded end is less than peak stress and the interface is in an elastic stage. When the interfacial shear stress reaches peak stress at loaded end, the interface enters into elastic-softening stage. As load increases, the peak stress moves from loaded end to unloaded end and the length of softening zone increases. When the interfacial shear stress at unloaded end reaches peak stress, the interface enters into softening stage.
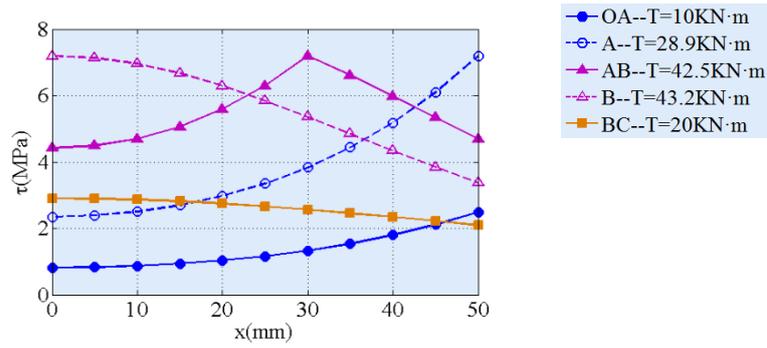
(a) *L*=50 mm
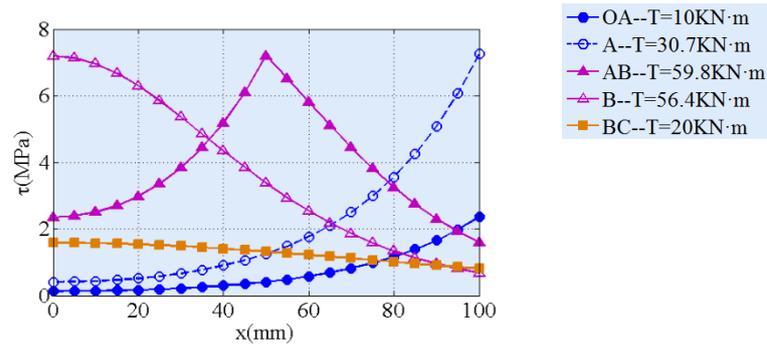
(b) *L*=100 mm

Fig. 6. Shear stress distribution for exponential model.

The shear stress distribution of *L*=50 mm and *L*=100 mm for bilinear model are shown in Fig. 7(a) and (b) respectively. When the load is small, the interfacial shear stress at loaded end is less than peak stress and the interface is an in elastic stage. When the interfacial shear stress reaches peak stress at loaded end, the interface enters into elastic-softening stage. As load increases, the peak stress moves from loaded end to unloaded end and the length of softening zone increases. When the interfacial shear stress at unloaded end reaches peak stress while $\Delta < \delta_f$, the interface enters into softening stage shown as in Fig. 7(a). When $\Delta = \delta_f$ and the interfacial shear stress at unloaded end is less than peak stress, then the interface enters into elastic-softening-debonding stage shown as in Fig. 7(b). The length of debonding zone increases as the peak stress moves to unloaded end. When the interfacial shear stress at unloaded end reaches peak stress, the interface enters into softening-debonding stage.

(a) *L*=50 mm

(b) *L*=100 mm

Fig. 7. Shear stress distribution for bilinear model.

5.3. Parametric study

Fig. 8(a) shows the influence of bond lengths on the load-displacement curves for bilinear model. From the figure, the significant influence for bond length on the curves could be observed. In the range of the effective bond length, as the bond length increases, not only the interface failure processes change but also the ultimate load and interfacial slip. Specifically, the increase of the bond length can increase damage ductility. However, when the bond length reaches a certain length (effective bond length), the ultimate load will hardly change. Fig. 8(b) shows the influence of bond lengths on the load-displacement curves for exponential model. From the figure we know that the ultimate load increases as the bond length increases. But the failure processes are all the same. The increase of the bond length can also increase damage ductility.

(a) bilinear model

(b) exponential model

Fig. 8. Load-displacement curves for different bond lengths.

Fig. 9 shows the comparison of the load-displacement curves between the two models for different bond lengths. The load-displacement curves are different when a softening area exists. In addition, since there is no debonding initiation in the exponential model, the displacement can increase unlimitedly, with this being different from bilinear model in which the slip would approach δ_f. From the figures, the load increases faster in the bilinear model as the bond lengths increases. However, when the bond length is long, the ultimate loads between the two models seem no more difference.

(a) *L*=25 mm

(b) *L*=75 mm

(c) *L*=125 mm

(d) *L*=175 mm

Fig. 9. Comparison of the load-displacement curves of the two models for different bond lengths.

Fig. 10 shows the load-displacement curves for different ratios of torsion stiffness ($\beta = G_2J_2/G_1J_1$). From the figures we can see that as the ratio increases, the ultimate load increases but the slip decreases, namely the ductility reduces.

(a) bilinear model

(b) exponential model
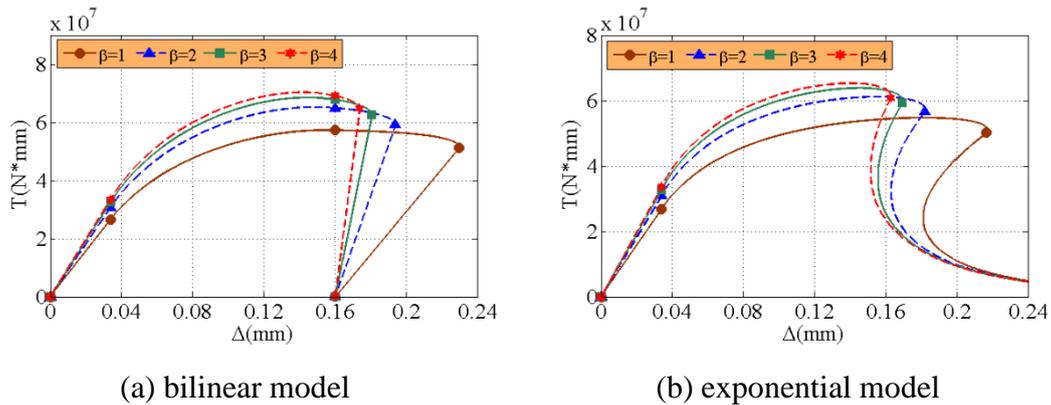
Fig. 10. Load-displacement curves for different ratios of torsion stiffness.

Fig. 11 shows the relationship between effective bond length and ratio of torsion stiffness. From the figure we can see that a stiffer coupler leads to a longer effective bond length. But as the ratio getting larger, the effective bond length increases not obviously. As the ratio increases, the effective bond length of two models have the similar trend, but the effective bond length of exponential model is longer than that of bilinear model.
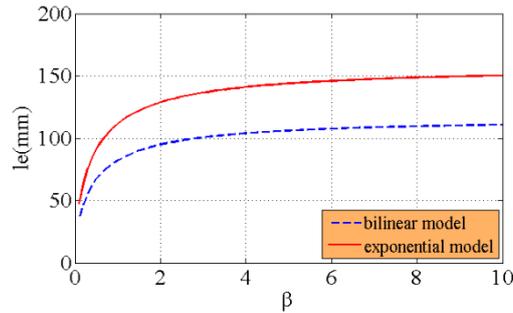
Fig. 11. Effect of ratio of torsion stiffness on the effective bond length.

Through the numerical computation, the ultimate load of exponential and bilinear models for different bond lengths could be obtained. Fig. 12 shows the ultimate load for different bond lengths. From the figure we can see that for bilinear model, when the bond length is short, the ultimate load increases significantly with the bond length. When the bond length is long, the ultimate load stays essentially unchanged. For the exponential model, when the bond length is short, the trend is similar to the bilinear model, but the ultimate load is relatively smaller. Both models have the same ultimate load when the bond length is relatively long.
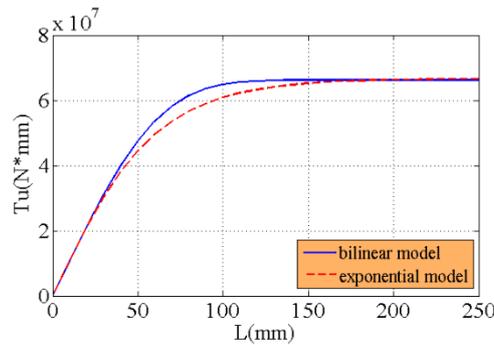
Fig. 12. The ultimate load of two models for different bond lengths.

Conclusions

On the basis of fully understanding the mechanical behavior of the pipe joints' interface, this paper gives a further understanding of the key factors of interfacial debonding. By modifying the torsion stiffness, the present models may be further extended to orthotropic materials, such as fiber-reinforced composite pipe joints. Based on the derivations in the current study, some important conclusions are summarized as follows:

(1) Through the nonlinear fracture mechanics, the analytical expressions of the interfacial shear stress and the load-displacement relationship at loaded end of pipe joints under torsion loads could be got. Thus the shear stress propagation and the debonding progress of the whole interface for different bond lengths could be predicted.

(2) The influences of different bond length on the load-displacement curve and the ultimate load are studied through the analytical solutions. The stress transfer mechanism, the interface crack propagation and the ductility behavior of the joints could be explained.

Acknowledgements

References

[1] Lv Y., Liu C.F., Huang X., Chen Y., Chouw N. (2021) Experimental and Finite-Element Studies of Buried Pipes Connected by a Bellow Joint under Cyclic Shear Loading, *Journal of Pipeline Systems Engineering and Practice* **12**.

[2] Ouyang Z.Y., Li G.Q. (2009) Interfacial debonding of pipe joints under torsion loads: a model for arbitrary nonlinear cohesive laws, *International Journal of Fracture* **155**, 19-31.

[3] Ouyang Z.Y., Li G.Q. (2009) Cohesive zone model based analytical solutions for adhesively bonded pipe joints under torsional loading, *International Journal of Solids and Structures* **46**, 1205-1217.

[4] Fame C.M., Correia J.R., Ghafoori E., Wu C. (2021) Damage tolerance of adhesively bonded pultruded GFRP double-strap joints, *Composite Structures* **263**,113625.

[5] Zhao Y., Pang S.S. (1995) Stress-strain and failure analyses of composite pipe under torsion, *Journal of Pressure Vessel Technology-Transactions of the Asme* **117**, 273-278.

[6] Zou G.P., Taheri F. (2006) Stress analysis of adhesively bonded sandwich pipe joints subjected to torsional loading, *International Journal of Solids and Structures* **43**, 5953-5968.

[7] Pugno N., Surace G. (2001) Tubular bonded joint under torsion: Theoretical analysis and optimization for uniform torsional strength, *Journal of Strain Analysis for Engineering Design* **36**, 17-24.

[8] Cheng J.Q., Li G.Q. (2008) Stress analysis of a smart composite pipe joint integrated with piezoelectric composite layers under torsion loading, *International Journal of Solids and Structures* **45**, 1153-1178.

[9] He Y., Vaz M.A., Caire M. (2021) Stress and failure analyses of thermoplastic composite pipes subjected to torsion and thermomechanical loading, *Marine Structures* **79**.

[10] Oh J.H. (2007) Strength prediction of tubular composite adhesive joints under torsion, *Composites Science and Technology* **67**, 1340-1347.

[11] Hosseinzadeh R., Taheri F. (2009) Non-linear investigation of overlap length effect on torsional capacity of tubular adhesively bonded joints, *Composite Structures* **91**, 186-195.

[12] Hosseinzadeh R., Cheraghi N., Taheri F. (2006) An engineering approach for design and analysis of metallic pipe joints under torsion by the finite element method. *Journal of Strain Analysis for Engineering Design* **41**, 443-452.

[13] Hosseinzadeh R., Shahin K.,Taheri F. (2007) A simple approach for characterizing the performance of metallic tubular adhesively-bonded joints under torsion loading, *Journal of Adhesion Science and Technology* **21**, 1613-1631.

[14] Esmaeel R.A., Taheri F. (2011) Influence of adherend's delamination on the response of single lap and socket tubular adhesively bonded joints subjected to torsion, *Composite Structures* **93**, 1765-1774.

[15] Xu W., Li G.Q. (2010) Finite difference three-dimensional solution of stresses in adhesively bonded composite tubular joint subjected to torsion, *International Journal of Adhesion and Adhesives* **30**, 191-199.

[16] Baishya N., Das R.R., Panigrahi S.K. (2017) Failure analysis of adhesively bonded tubular joints of laminated FRP composites subjected to combined internal pressure and torsional loading, *Journal of Adhesion Science and Technology* **31**, 2139-2163.

[17] Yuan H., Wu Z.S.,Yoshizawa H. (2001) Theoretical solutions on interfacial stress transfer of externally bonded steel/composite laminates, *Doboku Gakkai Ronbunshu* **2001**, 27-39.

[18] Wu Z.S., Yuan H., Niu H.D. (2002) Stress transfer and fracture propagation in different kinds of adhesive joints, *Journal of Engineering Mechanics* **128**, 562-573.

[19] Yuan H., Teng J.G., Seracino R., Wu Z.S., Yao J. (2004) Full-range behavior of FRP-to-concrete bonded joints, Engineering Structures **26**, 553-565.

[20] Caggiano A., Martinelli E.,Faella C. (2012) A fully-analytical approach for modelling the response of FRP plates bonded to a brittle substrate, International Journal of Solids and Structures **49**, 2291-2300.

[21] Chen F.L., Qiao P.Z. (2009) Debonding analysis of FRP-concrete interface between two balanced adjacent flexural cracks in plated beams, International Journal of Solids and Structures **46**, 2618-2628.

[22] Lorenzis L.D., Zavarise G.: Cohesive zone modeling of interfacial stresses in plated beams. Int. J. Solids Struct.**46**, 4181-4191 (2009)

[23] Wang J.L. (2007) Cohesive zone model of FRP-concrete interface debonding under mixed-mode loading, International Journal of Solids and Structures **44**, 6551-6568.

[24] Woo, S., Lee, Y. (2010) Experimental study on interfacial behavior of CFRP-bonded concrete, KSCE Journal of Civil Engineering **14**, 385-393.

[25] Yuan H., Lu X.S., Hui D., Feo L. (2012) Studies on FRP-concrete interface with hardening and softening bond-slip law, Composite Structures **94**, 3781-3792.

Molecular dynamics study on entangled structure of polymer chains in soft material

*†I. Riku and K. Mimura

Department of Mechanical Engineering, Osaka Metropolitan University, Japan

*†riku@omu.ac.jp

Abstract

It is well known that the mechanical resistance of soft materials such as rubber and elastomer can be improved by cross-linkages or fillers, which will lead to a construction of entangled structure of polymer chains. However, the correlation between the amount of cross-linkages or fillers with the toughness strength of the resultant material has not been clarified. Therefore, in this study, we at first construct a computational model for the resultant material with molecular dynamics method. Then, a series of simulations are performed for the resultant materials with different amount of cross-linkages or fillers under a cyclic loading condition. Finally, the development of entangled structure of polymer chains is investigated and the effect of such development on the toughness strength of soft materials is evaluated.

Keywords: Molecular dynamics method, Polymer chain, Entangled structure, Toughness strength.

Introduction

Soft materials such as rubber and elastomer are mostly used in the cross-linked state of polymer chains, i.e. lots of cross-linkages or fillers have been introduced and the entangled structures of polymer chains have been constructed completely. On the other hand, there are two types of entangled points in the entangled structure of polymer chains: one is the chemical type which is due to the existance of cross-linkages or fillers and obeys the affine deformation; the other one is the physical type which is due to the existance of dangling chains and obeys the nonaffine deformation.

To clarify the effect of cross-linkages or fillers on the mechanical behavior of soft materials, the investigation of the development of the entanlged structure of polymer chains during the deformation is indispensable. Therefore, in this study, we at first constitute a coarse-grained molecular dynamics model for the soft material with different amout of cross-linkages or fillers. Then, a series of simulations are performed for the resultant materials under a cyclic loading condition. Finally, the development of the entangled structure of polymer chains is investigated and the effect of such development on the toughness strength of soft material is evaluated.

Molecular Dynamics Model

When one is studying the mechanical properties of a polymeric material, the behavior is mainly governed by the topological features of the entangled structure of polymer chains and using a coarse graining procedure [1] can therefore be a successful approach.

Polymer Chain

In this study, the configuration of polymer chains is represented by a set of the Cartesian coordinates of grains and each grain represents a group of atoms. Two types of grains, i.e. A and B, are consisted in each polymer chain and the chemical formula of each monomer is specified as $+$ABBBB$+$. The bonding potential, U^{bond}, which represents the energy associated with the chemically connected grains separated by a distance r is given by[1]

$$U^{\text{bond}}(r) = U^{\text{FENE}}(r) + U^{\text{LJ}}(r) \tag{1}$$

$$U^{\text{FENE}}(r) = \begin{cases} -\frac{1}{2}kR_0^2\ln\left[1 - \left(\frac{r}{R_0}\right)^2\right] & r \leq R_0 \\ \infty & r > R_0 \end{cases} \tag{2}$$

$$U^{\text{LJ}}(r) = 4\epsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^6 + \frac{1}{4}\right] \tag{3}$$

where k is spring constant, R_0 is finite extended length, σ is Lennard-Jones diameter, ϵ is Lennard-Jones energy. On the other hand, the non-bonding potential, $U^{nonbond}$, which represents the interaction between the grains not chemically connected, or the grains separated far along the polymer backbone is given by

$$U^{\text{nonbond}}(r) = \begin{cases} 4\epsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^6\right] & r \leq r_c \\ 0 & r > r_c \end{cases} \tag{4}$$

where r_c is the cutoff distance.

Cross-linkage Point

In this study, the cross-linkage points are represented by the third type of grains, C, which react and make the chemical connections with the grains of type A when the distance between them is less than 1.3σ[1]. By the way, the bonding and non-bonding potential between the grains of type C with the other grains are the same as shown in Eq.(1)-(4).

Molecular Dynamics Algorithm

The molecular dynamics simulations consider a canonical ensemble (N, V, T), in which N is the total number of grains in the unit cell, V is the volume of the unit cell, T is the temperature. The (N, V, T) ensemble necessitates a modification of the equations of motion with an additional degree of freedom representing a kinetic mass[2][3]

$$m_i\frac{d^2\mathbf{r}_i}{dt^2} = -\frac{\partial U}{\partial \mathbf{r}_i} - m_i\Gamma\frac{d\mathbf{r}_i}{dt} + \mathbf{W}_i(t) \tag{5}$$

where m_i is the mass of the grain i, Γ is the friction constant, $\mathbf{W}_i(t)$ is a Gaussian white noise which is generated according to [1]

$$\langle\mathbf{W}_i(t) \cdot \mathbf{W}_j(t')\rangle = \delta_{ij}\delta(t - t')6k_BT\Gamma \tag{6}$$

where k_B is Boltzmann's constant.

Parameters

The molecular dynamics simulations are carried out in the reduced units, in which σ, ϵ and m are taken as the unit of lenght, energy and mass, respectively. The unit of time is given by $\tau = \sigma(m/\epsilon)^{1/2}$. The unit of temperature is given by $T = \epsilon/k_B$. In Table 1, the value of the parameters employed in the simulation is given with the corresponding reduced unit.

Table 1: Value of parameters

σ (σ)	ϵ (ϵ)	k (ϵ/σ^2)	R_0 (σ)	r_c (σ)	Γ (τ^{-1})	m (m)	T (T)
1	1	30	1.5	$2^{1/6}$	0.5	1	1

Boundary Conditions

Fig. 1 shows the unit cell and the coordinate system. The density of the grains consisted in the unit cell is specified as $\rho = 0.85(m/\sigma^3)$, from which the volume of the unit cell can be calculated based on the relation of $V = Nm/\rho$. In this study, the number of monomers in each polymer chain is $N^{monomer} = 200$, the number of polymer chains is $N^{chain} = 5$, the number of cross-linkage points is $N^{point} = 0, 20, 50$. Because the number of grains in each monomer is 5, the total number of grains in the unit cell is $N = N^{monomer} \times 5 \times N^{chain} + N^{point}$.

In order to perform the simulation under a cyclic loading condition, the unit cell is elongated and then compressed in the z direction with a constant stretch rate $d\lambda/dt = 10^{-4}\tau^{-1}$. When the unit cell is elongated or compressed in the z direction, the lengths of the unit cell in the x and y directions are changed simultaneously to keep the density of the grains constant. The time step is specified as $dt = 0.01\tau$. Periodic boundary condition is applied to the unit cell throughout the simulation.
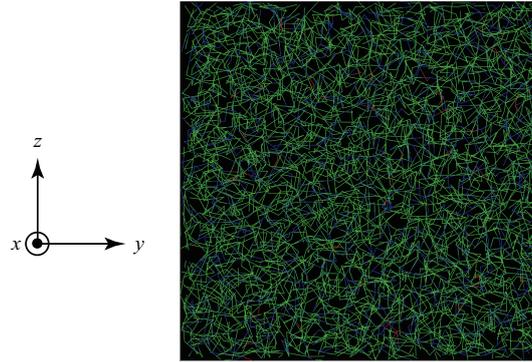
Figure 1: Configuration of polymer chains in the unit cell

Results

Fig. 2 shows the mechanical responses of polymer chains under the cyclic loading condition. When the number of the cross-linkage points increases, the deformation resistance increases. Moreover, the hysteresis loops can be observed and the dissipated energy during each cyclic loading process, W^{dis}, can be calculated from the area of each loop. Fig. 3 shows the effect of the amount of the cross-linkage points on the dissipated energy ratio which is defined by W^{dis}/W^{load}, where W^{load} means the work done during the loading process. When the number of the cross-linkage points increases, the dissipated energy ratio decreases. As the dissipated energy correlates closely with the toughness strength of soft material, a relatively loose connections between polymer chains maybe helpful for the improvement of the toughness strength of soft material.

To study the configurational change of polymer chains, the average radius of gyration over all polymer chains, R_g, is calculated and its development during loading process is shown in Fig.4, in which the value of R_g is normalized by its initial value R_{g0}. It can be found that the value of R_g increases together with the stretch. When the number of the cross-linkage points increases,

such increase of the value of R_g is suppressed.
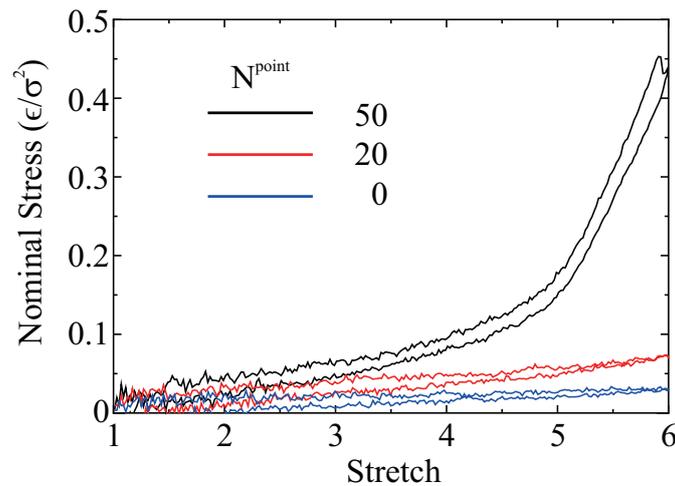
Figure 2: Nominal stress-stretch relations of polymer chains with different amount of cross-linkage points under the cyclic loading condition
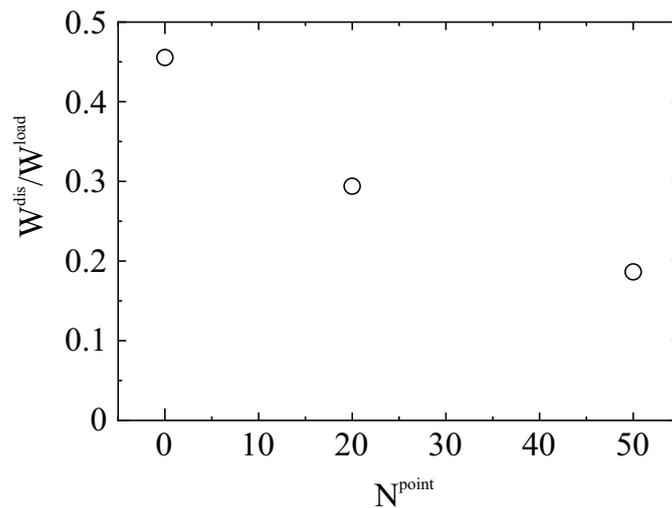
Figure 3: Relation between dissipated energy ratio and amount of cross-linkage points

Conclusions

In this study, the effect of the amount of cross-linkages on the mechanical behavior of soft materials under cyclic loading condition is investigated with molecular dynamics method. The results show that even though the increase of the amount of cross-linkages leads to the increase of the mechanical resistance, the development of the average radius of gyration of polymer chains is suppressed. As a result, the relative movement of polymer chains is limited in the soft material with large amount of cross-linkages and the toughness strength decreases eventually.

References

[1] Kremer, K. and Grest. G. S. (1990) Dynamics of entangled linear polymer melts: A molecular‐dynamics simulation. *The Journal of Chemical Physics* **92**, 5057—5086.
[2] Nose, S. (1984) A molecular dynamics method for simulations in the canonical ensemble. *Molecular Physics* **52**, 255—268.
[3] Hoover, W. G. (1985) Canonical dynamics: Equilibrium phase-space distributions. *Physical Review A* **31**, 1695—1697.
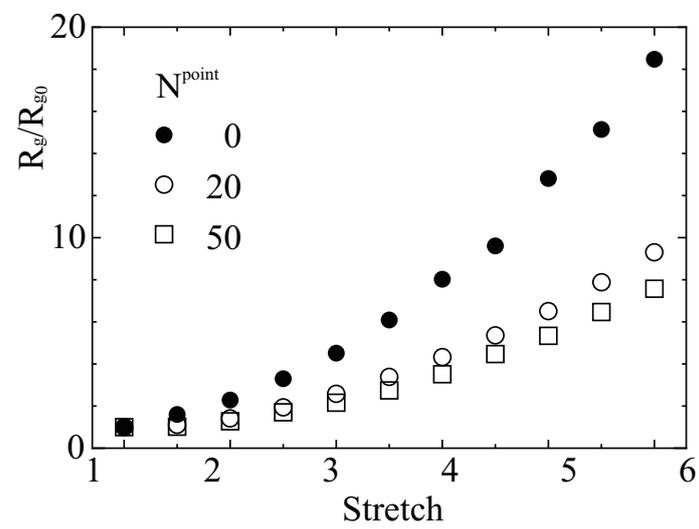
Figure 4: Development of R_g of polymer chains with different amount of cross-linkage points during loading process

Study on energy conservation in dynamic ultra-large deformation analysis of plane frame structures

* Erjon Krasniqi [1], Shuhei Yamashita[2] and †Hiroyuki Obiya[1]

[1]Department of Civil Engineering and Architecture, Saga University, Japan
[2]Miyaji Engineering Co., Ltd. Japan

*Presenting author: 21737001@edu.cc.saga-u.ac.jp
†Corresponding author: obiyah@cc.saga-u.ac.jp

Abstract

To analyze the time history response of flexible structures, it's required an algorithm with high accuracy to reproduce the geometrical nonlinear behavior. In order to check the accuracy of dynamic analyses, the conservation of energy or momentum under undamped conditions may be an important indicator. On the other hand, for ultra-large deformation analysis, the tangent stiffness method (TSM) has already enough achievement with strict evaluation of the rigid body displacement. As a time-integration algorithm applied to dynamic analysis is used Newmark β, which is unconditionally stable over time in the case of linear analysis under the condition of β=1/4. In this study, the combination of TSM and Newmark β is examined, and the conservation of energy through some numerical examples is verified. This study reveals that applying β>1/4 in the case of ultra-large deformations, longer duration of energy conservation is obtained. Also, when β=1/2 is applied, rough time increment leads to lower level of numerical damping compared to fine time increment, which is convenient for computational efficiency.

Introduction

Flexible structures such as cables, membranes or some thin structures are subject to ultra-large deformation with strong nonlinearity. In order to explore the time history response of these flexible structures, an algorithm with sufficient accuracy in terms of reproducibility of geometrically non-linear behavior is required [1]. It is common to check the continuity of energy or momentum conservation laws under undamped conditions as an indicator to ensure the accuracy of time history response analysis with ultra-large deformations [2].

As corresponding time integration methods, the energy-momentum method [3][4] and the symplectic integration method [5] have been proposed. The energy-momentum method is a time integration method, which uses the energy and momentum principles as an indicator of time integration stability, ensuring numerical stability by modifying the equations of motion to satisfy both principles and integrating over time. However, because the technique is based on the premise that non-linearity of finite strain is treated in the finite element method for each time history, there is no guarantee that a balanced solution with complete convergence of unbalanced forces at all nodes is attained. Even though the symplectic integration method is a semi-positive solution approach based on Hamilton's canonical equation, it is thought that a satisfactory response can only be reached if the time increments are split to an unrealistic degree.

Considering the above, the combination of the Newmark β method [7] and the Tangent stiffness method [8] can be a better way to obtain rational solutions for dynamic analysis with ultra-large displacement. The Newmark β method (β=1/4) is widely used in linear analysis due to its unconditional stability over time. Also, the tangent stiffness method is a geometric non-linear

theory that can strictly evaluate the rigid body displacement of an element and has made many significant achievements in static analysis.

However, in the case of geometrical non-linear analysis of flexible structures with ultra-large deformation, convergent solutions cannot be achieved at each time unless the time increment is set finely, and chaotic responses with loss of continuity appear over time, regardless of the presence of damping or external forces. For geometrical nonlinear analysis even in static analysis case, we have many problems, while in the dynamic analysis we need to add the time integration factor. Time integration in itself contains numerical problems which increase the difficulties of the analyses.

In this study, numerical experiments of free vibration without damping are carried out on the conservation of total dynamic energy in the dynamic large-deformation analysis of a plane frame structure using the Newmark β method and the tangential stiffness method to investigate the effect of time increments and value of β on the energy. Unlike in the case of small deformations where the Newmark β method (β=1/4) is better for getting more accurate results, in the case of ultra-large deformations, if β>1/4 is applied was found to bring better results. Therefore, in this study, we adopted Δt=0.01s as a time increment which is corresponding to the earthquake acceleration data issued by Japan Meteorological Agency [8], and examined the β values whose range of 0.2 to 0.5. As a result, we could detect the tendency that larger β realize longer period to keep the energy conservation.

Tangent Stiffness Method

The geometric nonlinearity of structures in the finite element method can be separated into the first nonlinear component caused by the rigid-body displacement of each element divided into finite elements and the second nonlinear component caused by the elements' own deformation described in the element local coordinate. In conducting a geometric nonlinear analysis, these two nonlinear components must be precisely evaluated. Element deformation can be reduced by making the element partitioning of the finite element structure denser, whereas rigid body displacement, which increases due to incremental nodal displacement, cannot be reduced. In other words, it is necessary to strictly evaluate the first nonlinear component by rigid-body displacement in ultra-large deformation regions where nodal displacement increases.

In the tangent stiffness method, each element is subjected to stable and static equilibrium conditions that constrain only rigid body displacements. Eq.(1) is the element force equation that shows the stiffness relation between the element force vector S and element deformation vector s, both defined in the element coordinate with stable support conditions.

$$\mathbf{S} = \mathbf{ks} \tag{1}$$

The tangent stiffness equation can be easily formulated by the first-order derivative of the equilibrium equation between the nodal vector in the global coordinate system representation and the element end force vector in the element coordinate system representation, provided that the element force equation of Eq. (1) is complete. Let D be the nodal displacement vector from the previous equilibrium displayed in the global coordinate system, and the equilibrium condition that shows the relationship between the two vectors of D and S can be expressed as follows:

$$\mathbf{D} = \mathbf{JS} \tag{2}$$

Here, J is the equilibrium condition. And by differentiating this equilibrium equation, we get:

$$\delta \mathbf{D} = \delta \mathbf{J} \cdot \mathbf{S} + \mathbf{J} \cdot \delta \mathbf{S} \tag{3}$$

Furthermore, since the first and second terms on the right-hand side can be expressed exactly as linear functions of the reference coordinate system-displayed incremental displacements δ**d,** the tangent stiffness equation can be expressed as follows:

$$\delta \mathbf{D} = (\mathbf{K_G} + \mathbf{K_0})\delta \mathbf{d} = \mathbf{K}\delta \mathbf{d} \tag{4}$$

Here, $\mathbf{K_G}$ is called the geometric stiffness matrix and is caused by the first nonlinear component caused by the rigid-body displacement. Further, $\mathbf{K_0}$ is called the element stiffness matrix and is the second nonlinear component caused by elements' own deformation described in the element local coordinate.

In this rigorous evaluation of geometric nonlinearity, tangential element stiffness due to element-specific stiffness can be expressed and evaluated in a separate form from tangential geometric stiffness due to rigid body displacement, which is the main factor.

In the case of plane frame structures, the compatibility equation Eq. (2) becomes as follows:

$$\begin{bmatrix} \Delta L \\ \theta_i \\ \theta_j \end{bmatrix} = \begin{bmatrix} -\alpha & -\beta & 0 & \alpha & \beta & 0 \\ \dfrac{\beta}{L} & \dfrac{\alpha}{L} & 1 & \dfrac{\beta}{L} & -\dfrac{\alpha}{L} & 0 \\ -\dfrac{\beta}{L} & \dfrac{\alpha}{L} & 0 & \dfrac{\beta}{L} & -\dfrac{\alpha}{L} & 1 \end{bmatrix} \begin{bmatrix} \Delta u_i \\ \Delta v_i \\ r_i \\ \Delta u_j \\ \Delta v_j \\ r_j \end{bmatrix} \tag{5}$$

where, as shown in Figure 1 and Figure 2, L is the initial element length, ΔL is elongation, θ_i and θ_j are deflection angles at each end respectively, r_i and r_j are the total rotation at each end respectively. While, α and β are the components of cosine vector.
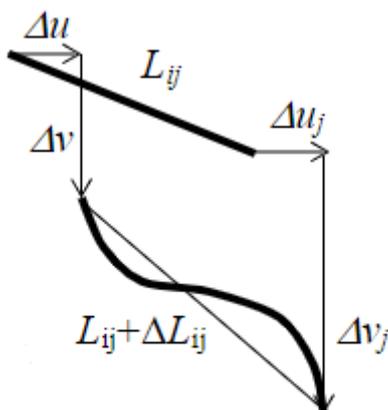
Figure 1. Displacement and element deformation
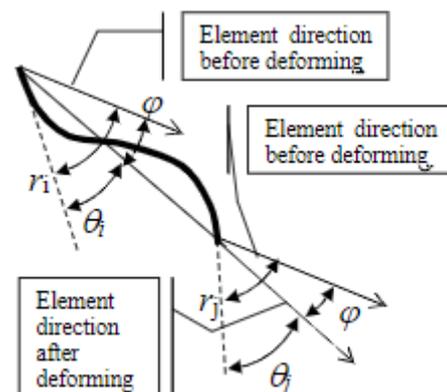
Figure 2. Rigid body rotation

As for Eq.(4) geometrical stiffness matrix $\mathbf{K_G}$, in the case of plane frame structures is given as follows:

$$\mathbf{K_G} = \begin{bmatrix} \boldsymbol{\mu} & -\boldsymbol{\mu} \\ -\boldsymbol{\mu} & \boldsymbol{\mu} \end{bmatrix} \tag{6}$$

Where:

$$\boldsymbol{\mu} = \begin{bmatrix} \mu\beta^2 + 2\gamma\alpha\beta & -\mu\alpha\beta - \gamma(\alpha^2 - \beta) & 0 \\ -\mu\alpha\beta - \gamma(\alpha^2 - \beta) & \mu\beta^2 + 2\gamma\alpha\beta & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{7}$$

$$\mu = \frac{N}{L}, \qquad \gamma = \frac{Q}{L} = -\frac{M_i + M_j}{L^2} \tag{8, 9}$$

Numerical example

As a model with ultra-large deformation, is proposed a cantilever with a moment acting on the free edge and is deformed to a circular shape to create initial equilibrium state as shown in Fig.3. The circular shape is achieved after solving and renewing with the static analysis by the tangent stiffness equation iteratively until the unbalanced force converges.

Initial state   After loading

Basic conditions
L = 1.0 (m)
Cross-section $A = 1.33 \times 10^{-5} m^2$
Moment of Inertia $I = 5.43 \times 10^{-13} m^4$
Young's modulus $E = 200$ GPa
Density $\rho = 7850 \frac{Kg}{m^3}$
Moment $M = 2EI\pi$
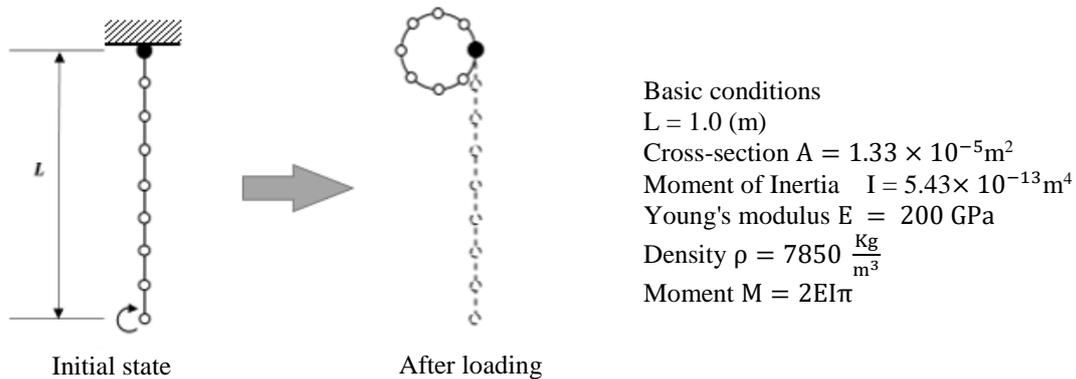
Figure 3. A model for numerical example

As shown in Fig.4, the convergence is achieved after 15 iterations, and it has undoubtedly converged while going through the state of a hundred million times the first time.
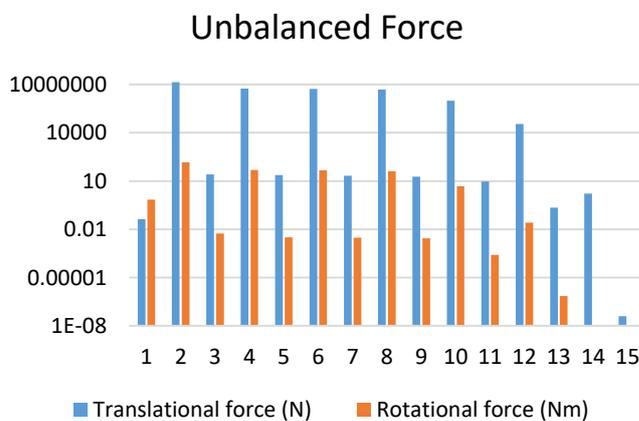
Figure 4. Convergence process of unbalanced force
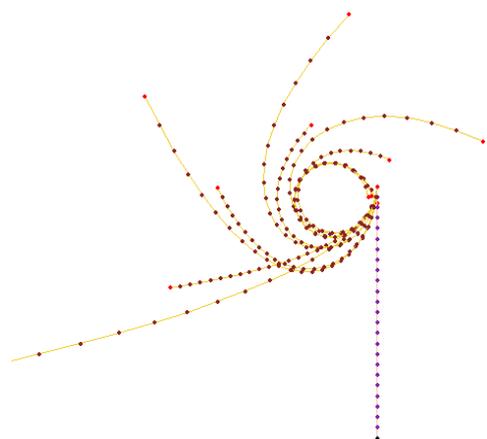
Figure 5. Shapes of unbalanced state toward converging

The shape of the structure during these iterations is shown in Fig.5 where the purple line represents the initial state, and the yellow lines represent the state during each iteration respectively.

After the external force is removed, the undamped free vibration dynamic analysis is performed, and the energy conservation is investigated.

Here, Newmark beta process is adopted. The iteration for geometrical nonlinear analysis to find the displacement to the time step of *i*+1 from *i* can be expressed as follows.
If the constant external force can be expressed as:

$$\mathbf{F} = \mathbf{P}(t) + \mathbf{M}\left\{\left(\frac{1}{2\beta}-1\right)\ddot{\mathbf{u}}_i + \frac{4}{\beta\Delta t}\dot{\mathbf{u}}_i\right\} + \mathbf{C}\left\{\left(\frac{1}{4\beta}-1\right)\ddot{\mathbf{u}}_i\Delta t + \left(\frac{1}{4\beta}-1\right)\dot{\mathbf{u}}_i\right\} \qquad (10)$$

Here, $\mathbf{P}(t)$ is external force depending on time, and $\mathbf{M}$ and $\mathbf{C}$ are mass matrix and damping matrix respectively. In this paper, we didn't apply damping so $\mathbf{C}=\mathbf{0}$. Further, the unbalanced force of jth iteration can be:

$$\Delta \mathbf{U}_j = \mathbf{F} - \left(\frac{\mathbf{M}}{\beta\Delta t^2} + \frac{\mathbf{C}}{2\beta\Delta t}\right)\Delta \mathbf{u}_{i+1,j} - \mathbf{J}(\Delta \mathbf{u}_{i+1,j})\mathbf{S}(\Delta \mathbf{u}_{i+1,j}) \qquad (11)$$

therefore, we can use the tangent stiffness equation for each iteration step will be:

$$\Delta \mathbf{u}_{i+1,j+1} = \mathbf{K}_{i+1,j}^{-1}\Delta \mathbf{U}_j, \quad \mathbf{K} = \mathbf{K}_G + \mathbf{K}_0 \qquad (12), (13)$$

Japan Meteorological Agency publishes earthquake acceleration data in 1/100 second increment. Therefore, assuming that the response to the external force of an earthquake is examined, analysis by time increments of 1/10000 second level shown in some literatures [3][4] so far is not realistic.

In this study, 1/100 second of time increments is adopted, and the influence of Newmark β value on the length of time keeping the energy conservation is examined. As for the value of β, 25 data were verified every 0.0125 from 0.2 to 0.5. Also here, if the energy at a certain time step is double as the initial energy, the energy conservation is considered to be lost.
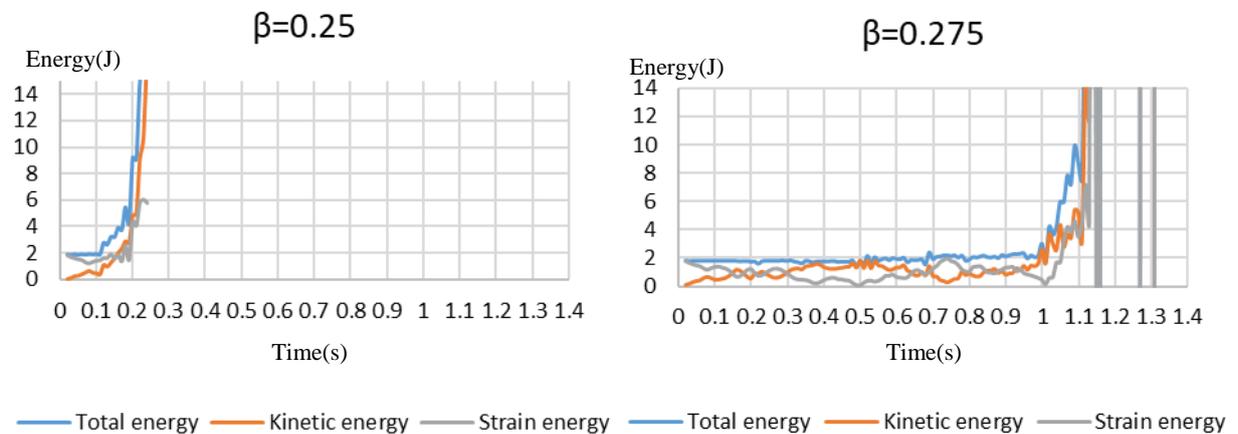
Figure 6. Energy history Δt=0.01, β=0.25   **Figure 7.** Energy history Δt=0.01, β=0.275

First of all, two significant cases are shown in Fig.6 and 7 that are for β=0.25 and β=0.275, respectively. In case of β=0.25 the energy conservation is lost and diverged after only 0.16seconds. When using a β value slightly larger for example β=0.275, a longer time for energy conservation is noticed.

On the other hand, if β=0.4375 is applied, the energy conservation is lost at t=98.69s and quite long-time energy conservation has been observed in this case (Fig.8).
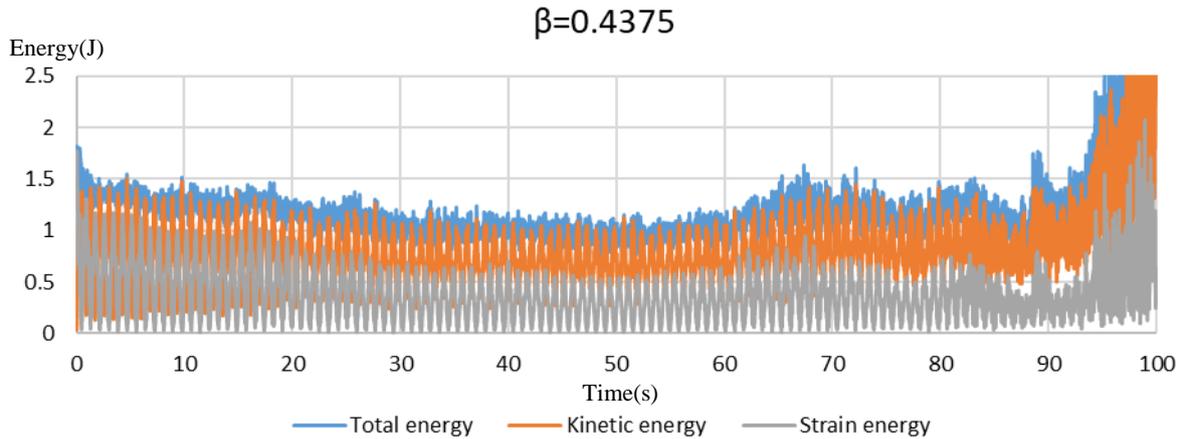
Figure 8. Energy history Δt=0.01, β=0.4375

As we can see, using a higher value of β usually a longer energy conservation time is observed. But, after investigating many cases of different β values was found that this phenomenon is not consistently true. In some cases, even if a higher value of β is taken there is a chance that the energy conversation will be lost earlier compared to the smaller β value as shown in Figure 9, where for β=0.475 the energy conservation is lost at t=24.03s.
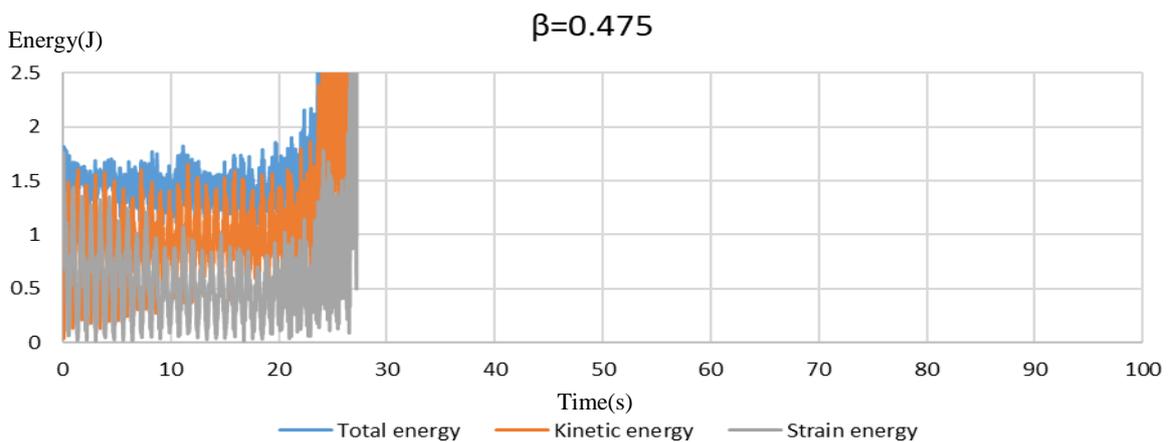
Figure 9. Energy history Δt=0.01, β=0.475

After examining different cases for β values between β=1/5 and β=1/2, it is found that despite the non-continuity of having better results at each time higher β is used there is a tendency that with a higher value of β longer time of energy conservation will be obtained, as shown in Fig. 10.
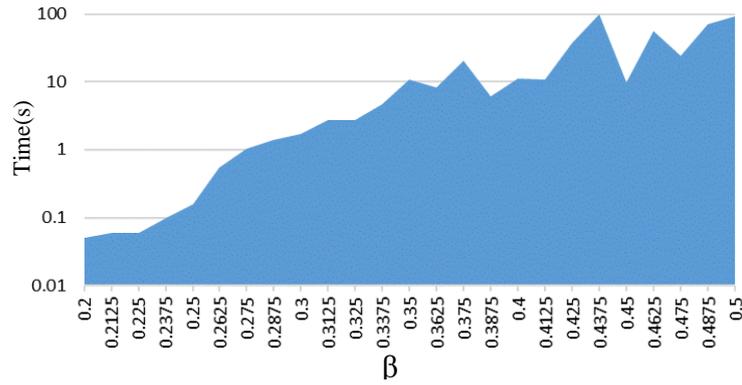
Figure 10. Relation between conservation time

Furthermore, for the case of β=1/2 with time increment Δt=0.01s (Fig.11) it is found to be tolerable stable energy.

Δt=0.00025; β=1/2
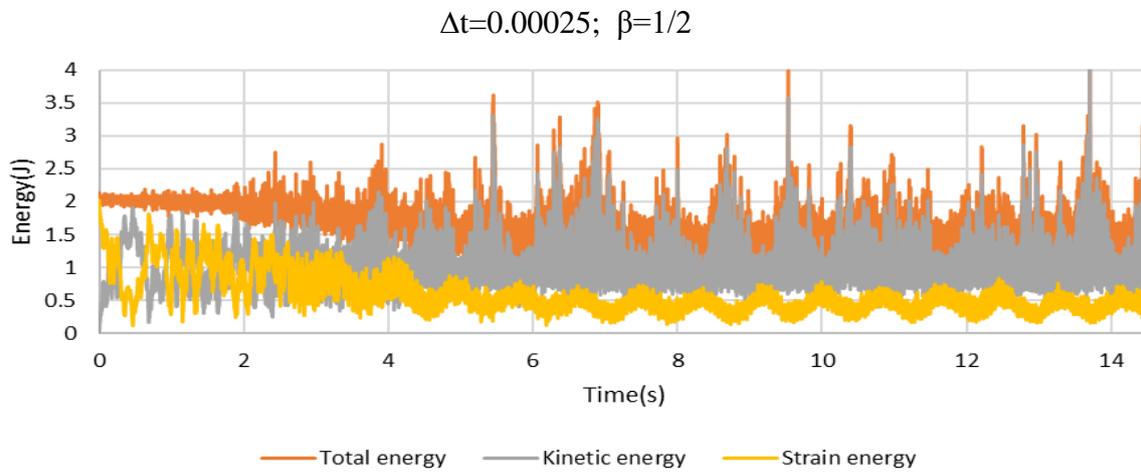
Figure 12. Energy history Δt=0.00025, β=0.5

On the other hand, for the same β=1/2 with finer time increment, in this case Δt =0.00025s (Fig.12), despite what is expected, it is noticed a high level of numerical damping, which causes the energy to fluctuate irregularly.

Δt=0.01; β=1/2
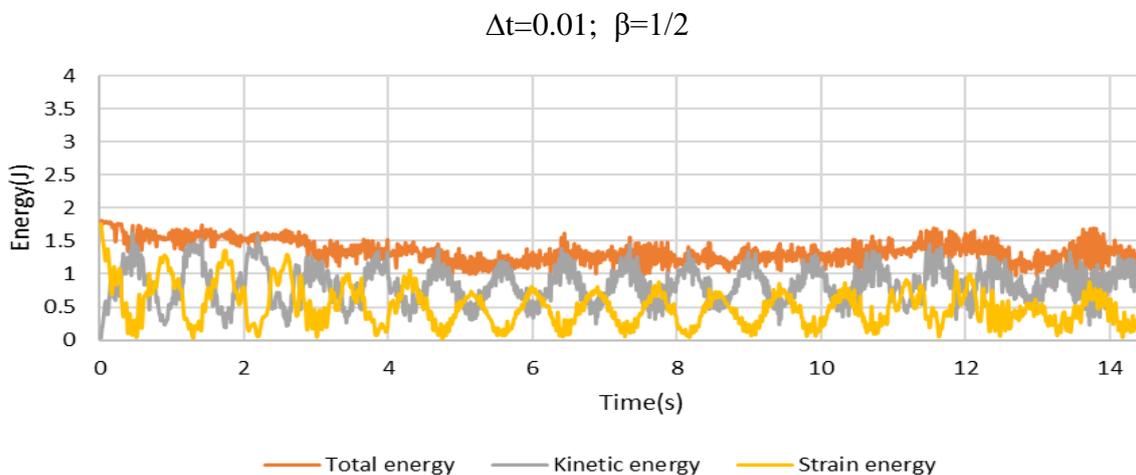
Figure 11. Energy history Δt=0.01, β=0.5

129

Moreover, by comparison of vertical displacement at the edge of the cantilever, in the case of Δt=0.00025s, the influence of numerical damping appears significantly(Fig.13).
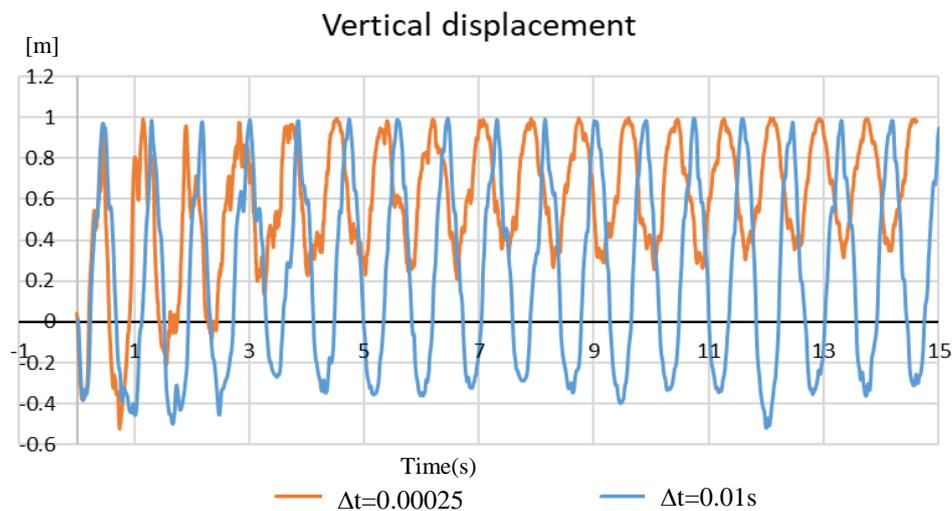
Figure 13. Comparison of displacement in Δt=0.01s and Δt=0.00025

Conclusion

In this study, the influence of Newmark β and time increment on energy conservation duration was analyzed. It is found that by using a higher value for β the energy conservation duration is significantly extended. In the case when β=1/2 was applied, it is noticed that with a rougher time increment Δt=0.01s stable energy in respect of time was acquired, while when fine time increment Δt=0.00025s is used notable numerical damping is detected. In the latter case, the numerical damping is also detectable when vertical displacement is compared to the former case. The fact that rough time increment brings a more stable energy response is convenient in the computational aspect.

References

[1]   Tien Long Nguyen, Carlo Sansour, Mohammed Hjiaj, (2017) "Long-term stable time integration scheme for dynamic analysis of planar geometrically exact Timoshenko beams" Journal of Sound and Vibration 396 144–171.
[2]   U. Galvaneito, M. A. Crisfield, "An Energy-Conserving Co-Rotational Procedure for The Dynamics of Planar Beam Structures"   International Journal for Numerical Methods in Engineering, Vol. 39,2265-2282
[3]   J. C. Simo, N. Tarnow "The discrete energy-momentum method. Conserving algorithms for nonlinear elastodynamics", Zeitschrift für angewandte Mathematik und Physik ZAMP volume 43, pages757–792 (1992)
[4]   Sophy Chhang, Carlo Sansour, Mohammed Hjiaj, Jean-marc Battini, "An energy-momentum co-rotational formulation for nonlinear dynamics of planar beams, Computers and Structures, vol. 187, pp. 50-63 (2017).
[5]   Tod A. Laursen: Computational Contact and Impact Mechanics: Fundamentals of Modeling Interfacial Phenomena in nonlinear Finite Element Analysis, Springrer Verlag, 2002.
[6]   N. M. Newmark, "A Method of computation for Structural Dynamics", Proc.ASCE, Vol. 85, No. EM3, 67-94 (1959).
[7]   H. Obiya, "A study on accuracy and versatile of the tangent stiffness method by separation of element stiffness from geometrical stiffness (In Japanese)," Saga University, Japan (1998).
[8]   https://www.data.jma.go.jp/svd/eqev/data/bulletin/data/format/datfmt_e.html

An ANN-BCMO Approach for Material Distribution Optimization of Bi-directional Functionally Graded Nanocomposite Plates with Geometrically Nonlinear Behaviors

Paowpat Pensupa[*], **Toan Minh Le, and Jaroon Rungamornrat**[†]

Center of Excellence in Applied Mechanics and Structures, Department of Civil Engineering, Faculty of Engineering, Chulalongkorn University, Thailand

[*]Presenting author: paowpat_p@hotmail.com
[†]Corresponding author: Jaroon.R@chula.ac.th

Abstract

This study commences with the application of an efficient artificial neural network (ANN)-balancing composite motion optimization (BCMO) approach for finding the optimal material distribution of bi-directional functional graded nanocomposite (FGN) thin plates considering geometrically nonlinear behavior. The method integrates ANN into the framework of BCMO to improve the computational efficiency of the iteration-based optimization process. In this strategy, the ANN-based surrogate model is used to predict the high-fidelity structural responses obtained by a geometrically nonlinear NURBS-based isogeometric analysis based on the Kirchhoff-Love plate model and the Von Kármán nonlinearity. Whilst BCMO is employed as an optimizer for solving optimization problems without complex sensitivity analyses. To enhance the possibility to explore the complex distribution of optimal material profiles, the optimal in-plane volume fraction function of FGN plates is modeled by a two-dimensional non-uniform rational B-spline (NURBS) basis function which is separate from the NURBS analysis meshes. Accordingly, its unknown control point values are selected as the continuous design variables. The effectiveness and accuracy of the proposed algorithm are illustrated via selected numerical examples. Results show a significant reduction in the computational effort over the conventional approach which is based on BCMO and IGA direct analysis.

Keywords: Material distribution optimization, Artificial neural network, Balancing composite motion optimization, Bi-directional functionally graded plates, Geometrical nonlinearity.

1. Introduction

Structural optimization has gained popularity in various engineering scenarios. In general, it can be divided into several categories such as shape, topology, sizing, and material distribution optimizations. In contrast to the first three types of optimizations which have a long history of development, the material distribution optimization has just recently received great attention from many researchers due to the emergence of functionally graded materials (FGMs) [1][2] in which the spatial variation of mechanical properties in certain directions can be optimally designed to achieve desirable structural performances. Recently, carbon nanotubes (CNTs) were introduced with remarkable mechanical properties such as very high strength and low density. With their attractive properties, CNTs have been used broadly as reinforcing constituents in other materials, especially FGMs [3], as a better choice for structural engineering and designs.

During past decades, the optimal design of structures made of FGMs has been extensively studied using both gradient-based and gradient-free algorithms. For instance, Moita et al. [4] used the feasible arc interior point algorithm to find the best material profile of functionally graded (FG) plate and shell structures for free vibration and linear buckling analyses. Qin and Dong [5] used the sequential quadratic method for shape and material distribution optimization

of the curvilinearly stiffened plates. Goupee and Vel [6] employed a real coded genetic algorithm for optimizing the material distribution of FG beams in the steady-state free and forced vibration problems. Roque and Martins [7] and Roque et al. [8] employed the differential evolution to obtain the optimal material distribution for improving the dynamic performance of FG macro-scale and micro-scale beams, respectively. In addition, several other metaheuristic algorithms have also been employed to solve various optimization problems for FG plate- and shell-type structures such as golden-section search [9], particle swarm optimizations [10][11], and genetic algorithm [12][13]. Results from those studies have indicated that metaheuristic-based algorithms can overcome the main drawbacks of the gradient-based optimization algorithms (such as the need of explicit information on the gradients of the objective function and the constraint concerning the design variables, and the sensitivity to assumed initial solutions) in optimizing the material distribution for structures made of FGMs. However, there still exist two challenges that prevent the broad applications of such metaheuristic algorithms to handling the optimization problems. Firstly, most of the algorithms employed in the aforementioned studies rely on the algorithms that depend on the algorithmic control parameters which need to be selected a priori. Thus, it requires highly experienced users to perform the parameter tuning to improve the performance of the algorithm. Secondly, despite their simplicity, the requirement of a great number of populations and generations in optimization loops makes those algorithms cumbersome especially for large-scale and nonlinear optimization problems.

Due to the growing application of FG plates, especially ones with nanocomposite reinforcement and in-plane material inhomogeneity, the development of efficient and effective tools for optimizing the material variation under various loading conditions and nonlinearities plays an important role in the analysis and design of such important structures. Hence, the present study attempts to devote the novel application of a surrogate-assisted optimization approach based on a recently developed parameter-free metaheuristic algorithm named balancing composite motion optimization (BCMO) [14] and artificial neural network (ANN) [15] to tackle the problem of finding optimal material distribution for geometrically nonlinear FG nanocomposite plates. This proposed framework will not only preserve the attractive features of BCMO in solving optimization problems, but also utilize the capacity of an ANN-based surrogate model to improve the efficiency of the BCMO framework for solving time-consuming optimization problems. Specifically, the BCMO algorithm does not require specifying algorithmic control parameters and, more importantly, it possesses a robust ability to balance the exploration and exploitation which is a key factor to ensure a global convergence. This algorithm has been applied to various engineering problems such as the ultimate load estimation of rectangular concrete-filled steel tubes [16] and the optimization of rectangular concrete-filled steel tube short columns [17], and demonstrated its high potential for the optimal design of engineering structures. Whilst the use of ANN to construct a surrogate model to replace the evaluation of optimization objectives and constraints using the expensive numerical analyses can result in a significant decrease in computation cost has been achieved, the quality of the solution has still been ensured at an appropriate accuracy level.

The rest of this article is outlined as follows. Section 2 briefly presents the fundamental formulation of the geometrically nonlinear thin plate model and the numerical solution methodology based on isogeometric analysis. The proposed ANN-BCMO is provided in Section 3 whereas selected numerical examples and obtained results are presented and discussed in Section 4. Finally, Section 5 ends the paper with noteworthy conclusions.

2. Theoretical Formulation

2.1. *Modeling of in-plane bi-directional FG nanocomposite plates*

This study focuses on bi-directional FG nanocomposite (BFGN) thin plates consisting of the polymer as a matrix and multi-walled carbon nanotubes as the reinforcement, as shown in Figure 1. The material properties are varied in the in-plane directions (i.e., x and y directions) while being homogeneous across the thickness. Due to the material inhomogeneity, the analysis of BFGN plates depends on two main aspects: the volume fraction variation through the grading direction and the homogenization technique used to evaluate the effective properties.
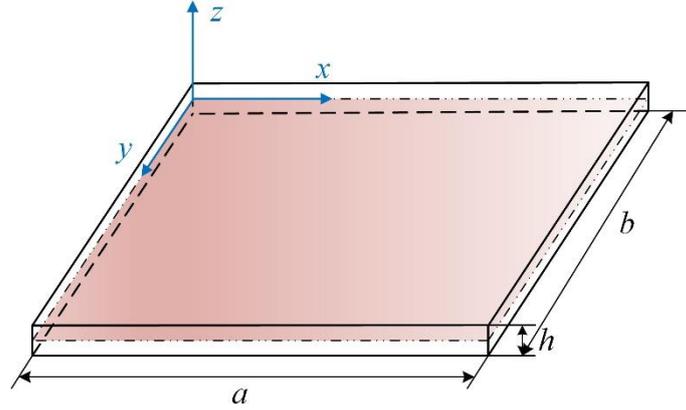
Figure 1. Schematic of BFGN thin plates

The volume fraction functions of carbon nanotubes (CNTs) and polymer are denoted by $V_{CNT}(x, y)$ and $V_p(x, y)$, respectively, and confined with the following rule:

$$V_{CNT}(x,y) + V_p(x,y) = 1 \ \ with \ \ 0 \le V_{CNT}, V_p \le 1 \tag{1}$$

These volume fraction functions are commonly defined using predefined mathematical functions in terms of the in-plane curvilinear coordinates (x, y). However, using the defined mathematic functions cannot describe the complex material property variation [18]. Hence, in the present study, the NURBS surface functions are utilized to represent the volume fraction of the CNT as

$$V_{CNT}(\xi, \eta) = \left[\sum_{j=1}^{n} R_i(\xi, \eta) V_{CNT}^i \right] V_{CNT}^* \tag{2}$$

where ξ and η are the parametric coordinates; n is the number of control points; $V_{CNT}^i \in [0,1]$ denotes the ith control value; and $R_i(\xi, \eta)$ are the NURBS basis functions whose detailed description can be referred to Cottrell et al. [19]. In addition, V_{CNT}^* is the upper bound of the CNT volume fraction and can be defined, based on the desired mass fraction of the CNT in the composite plate, as [20]

$$V_{CNT}^* = \frac{\rho_p W_{CNT}}{\rho_p W_{CNT} + \rho_{CNT}(1 - W_{CNT})} \tag{3}$$

where ρ_{CNT} and ρ_p are the density of CNTs and polymer matrix, respectively, and W_{CNT} is the desired mass fraction of CNT which can be chosen upon the design requirement.

Once V_{CNT} is known, the volume fraction of the polymer V_p is determined by

$$V_p(\xi, \eta) = 1 - V_{CNT}(\xi, \eta) \tag{4}$$

Accordingly, the effective Young's modulus E_{eff} and Poisson's ratio ν_{eff} are obtained via a homogenization scheme based on the modified rule of mixture [20]

$$E_{eff} = (k_l k_0 k_w E_{CNT} - E_p)V_{CNT}e^{\zeta V_{CNT}} + E_p \tag{5}$$

$$\nu_{eff} = \nu_{CNT}V_{CNT} + \nu_p V_p \tag{6}$$

where E_{CNT} and E_p are Young's modulus of CNTs and polymer matrix, respectively; and ν_{CNT} and ν_p denote the Poisson's ratio of CNTs and polymer matrix, respectively. In addition, the calculation of the parameters $\{k_l, k_0, k_w\}$ and ζ are referred to El-Ashmawy et al. [20].

2.2. Kirchhoff-Love plate formulation

The kinematic formulation of the thin plate considered in this study is based on Kirchhoff-Love plate theory, in which segments normal to the plate midsurface are assumed to remain straight and perpendicular to the midsurface during the deformation. This results in the following displacement field:

$$\begin{bmatrix} u(x,y,z) \\ v(x,y,z) \\ w(x,y,z) \end{bmatrix} = \begin{bmatrix} 1 & 0 & -z\dfrac{\partial(\cdot)}{\partial x} \\ 0 & 1 & -z\dfrac{\partial(\cdot)}{\partial y} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_0(x,y) \\ v_0(x,y) \\ w_0(x,y) \end{bmatrix} \tag{7}$$

where $u_0(x,y)$, $v_0(x,y)$, and $w_0(x,y)$ are the midsurface displacements in the x-, y- and z-directions.

By adopting Von Kármán theory, the nonlinear strain-displacement relations are given by [21]

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_{xx} \\ \varepsilon_{xx} \\ 2\varepsilon_{xy} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial u}{\partial x} + \dfrac{1}{2}\left(\dfrac{\partial w}{\partial x}\right)^2 \\ \dfrac{\partial v}{\partial y} + \dfrac{1}{2}\left(\dfrac{\partial w}{\partial y}\right)^2 \\ \dfrac{\partial u}{\partial y} + \dfrac{\partial v}{\partial x} + \dfrac{\partial w}{\partial x}\dfrac{\partial w}{\partial y} \end{bmatrix} \tag{8}$$

Substituting the displacement components in Eq. (7) into Eq. (8) yields

$$\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}_m^l + z\boldsymbol{\varepsilon}_b^l + \boldsymbol{\varepsilon}_m^{nl} \tag{9}$$

where $\{\boldsymbol{\varepsilon}_m^l, \boldsymbol{\varepsilon}_b^l\}$ denote the linear membrane and bending strains, respectively, whereas $\boldsymbol{\varepsilon}_m^{nl}$ is the nonlinear strain. These strains are given by

$$\boldsymbol{\varepsilon}_m^l = \begin{bmatrix} \dfrac{\partial u_0}{\partial x} \\ \dfrac{\partial v_0}{\partial y} \\ \dfrac{\partial u_0}{\partial y} + \dfrac{\partial v_0}{\partial x} \end{bmatrix}, \quad \boldsymbol{\varepsilon}_b^l = \begin{bmatrix} -\dfrac{\partial^2 w_0}{\partial x^2} \\ -\dfrac{\partial^2 w_0}{\partial y^2} \\ -2\dfrac{\partial^2 w_0}{\partial x \partial y} \end{bmatrix}, \quad \boldsymbol{\varepsilon}_m^{nl} = \dfrac{1}{2}\begin{bmatrix} \dfrac{\partial w_0}{\partial x} & 0 \\ 0 & \dfrac{\partial w_0}{\partial y} \\ \dfrac{\partial w_0}{\partial y} & \dfrac{\partial w_0}{\partial x} \end{bmatrix}\begin{bmatrix} \dfrac{\partial w_0}{\partial x} \\ \dfrac{\partial w_0}{\partial y} \end{bmatrix} = \dfrac{\mathbf{A}}{2}\begin{bmatrix} \dfrac{\partial w_0}{\partial x} \\ \dfrac{\partial w_0}{\partial y} \end{bmatrix} \tag{10}$$

The constitutive relation is expressed as

$$\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\varepsilon} \tag{11}$$

where $\boldsymbol{\sigma}$ is the generalized stress vector and $\mathbf{D}$ is the moduli matrix defined by

$$\boldsymbol{\sigma} = [\sigma_{xx} \quad \sigma_{yy} \quad \sigma_{xy}]^T, \quad \mathbf{D} = \frac{E(x,y)}{1-v^2(x,y)} \begin{bmatrix} 1 & v(x,y) & 0 \\ v(x,y) & 1 & 0 \\ 0 & 0 & \frac{1-v(x,y)}{2} \end{bmatrix} \quad (12)$$

with $E(x,y)$ and $v(x,y)$ denoting Young's modulus and Poison's ratio of bi-directional FG nanocomposite materials, respectively. The principle of virtual work can be written as

$$\int_V \boldsymbol{\sigma}\delta\boldsymbol{\varepsilon}d\Omega = \int_\Omega q(x,y)\delta w_0 d\Omega + P\varsigma(\mathbf{X}-\bar{\mathbf{X}})\delta w_0 \quad (13)$$

where $q(x,y)$ and P are the distributed and concentrated transverse loads, respectively. In addition, $\varsigma(\cdot)$ denotes the Dirac-delta function whereas $\mathbf{X}$ and $\bar{\mathbf{X}}$ are the position vector and the loading location, respectively.

2.3. IGA for geometrically nonlinear analysis

Within the framework of IGA [19], the plate geometry is described by bivariate NURBS functions as

$$\mathbf{X} = \sum_{J=1}^N R_I^a(\xi,\eta)\mathbf{X}^I \quad (14)$$

where $R_I^a(\xi,\eta)$ are NURBS basis functions used in the analysis and N is the number of control points. Hence, the displacements can be approximated by

$$\mathbf{u}_0 = \sum_{J=1}^N R_I^a(\xi,\eta)\mathbf{u}_0^I \quad (15)$$

where $\mathbf{u}_0^I = [u_0^I \quad v_0^I \quad w_0^I]^T$. Substituting Eq. (15) into Eq. (10) leads to

$$\boldsymbol{\varepsilon} = \mathbf{B}\mathbf{u}_0 \quad (16)$$

where $\mathbf{u}_0 = [(\mathbf{u}_0^1)^T \quad (\mathbf{u}_0^2)^T \quad ... \quad (\mathbf{u}_0^N)^T]^T$ and the matrix $\mathbf{B}$ can be calculated from

$$\mathbf{B} = \mathbf{B}_m^l + z\mathbf{B}_b^l + \frac{1}{2}\mathbf{A}\mathbf{B}_m^{nl} \quad (17)$$

with

$$\mathbf{B}_m^l = [\mathbf{B}_{m1}^l \quad \mathbf{B}_{m2}^l \quad ... \quad \mathbf{B}_{mN}^l], \quad \mathbf{B}_{mI}^l = \begin{bmatrix} \frac{\partial R_I}{\partial x} & 0 & 0 \\ 0 & \frac{\partial R_I}{\partial y} & 0 \\ \frac{\partial R_I}{\partial y} & \frac{\partial R_I}{\partial x} & 0 \end{bmatrix} \quad (18)$$

$$\mathbf{B}_b^l = [\mathbf{B}_{b1}^l \quad \mathbf{B}_{b2}^l \quad ... \quad \mathbf{B}_{bN}^l], \quad \mathbf{B}_{bI}^l = \begin{bmatrix} 0 & 0 & -\frac{\partial^2 R_I}{\partial x^2} \\ 0 & 0 & -\frac{\partial^2 R_I}{\partial x^2} \\ 0 & 0 & -\frac{\partial^2 R_I}{\partial x^2} \end{bmatrix} \quad (19)$$

$$\mathbf{B}_m^{nl} = [\mathbf{B}_{m1}^{nl} \quad \mathbf{B}_{m2}^{nl} \quad ... \quad \mathbf{B}_{mN}^{nl}], \quad \mathbf{B}_{mI}^{nl} = \begin{bmatrix} 0 & 0 & \frac{\partial R_I}{\partial x} \\ 0 & 0 & \frac{\partial R_I}{\partial y} \end{bmatrix} \quad (20)$$

According to the principle of virtual work (i.e., Eq. (13)), a system of nonlinear algebraic equations is obtained as follows:

$$L(u) = K(u)u - F = 0 \tag{21}$$

where $\mathbf{L}$ is termed the residual vector; $\mathbf{F}$ is the force vector; and $\mathbf{K}$ is the stiffness matrix defined by

$$\mathbf{K} = \sum_{I=1}^{N} \sum_{J=1}^{N} \left(\int_V \overline{\mathbf{B}}_I^T \mathbf{D} \mathbf{B}_J dV \right) \tag{22}$$

with $\overline{\mathbf{B}}_I = \mathbf{B}_{mI}^l + z\mathbf{B}_{bI}^l + A\mathbf{B}_{mI}^{nl}$. The nonlinear system Eq. (21) can be solved efficiently by Newton-Raphson iterative method. Specifically, the displacement at the $(n+1)^{th}$ iteration, denoted by $\mathbf{u}^{n+1}$, is updated via the following rule:

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \Delta\mathbf{u}^n \tag{23}$$

$$\Delta\mathbf{u}^n = -\mathbf{K}_t^{-1}\mathbf{L}^n \tag{24}$$

where $\mathbf{u}^n$ denotes the displacement at the nth iteration; $\mathbf{L}^n = \mathbf{L}(\mathbf{u}^n)$; and $\mathbf{K}_t$ is the tangent stiffness matrix obtained from the differentiation of the internal force vector as [21]

$$\mathbf{K}_t = \sum_{I=1}^{N} \sum_{J=1}^{N} \int_V \left[\overline{\mathbf{B}}_I^T \mathbf{D} \overline{\mathbf{B}}_J + \left(\mathbf{B}_{mI}^{nl} \right)^T \boldsymbol{\sigma} \mathbf{B}_{mJ}^{nl} \right] dV \tag{25}$$

with $\boldsymbol{\sigma} = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} \\ \sigma_{xy} & \sigma_{yy} \end{bmatrix}$ denoting the stress matrix. The iterative process is terminated if the following criterion is satisfied:

$$\sqrt{\frac{|\mathbf{u}^{n+1} - \mathbf{u}^n|^2}{|\mathbf{u}^{n+1}|^2}} \leq \epsilon \tag{26}$$

where ϵ is a specified tolerance. In the present study, $\epsilon = 10^{-6}$ is employed.

3. Optimization problem

3.1. Problem Statement

The problem statement is to optimize the distribution of material properties or, equivalently, to find the optimal volume fraction distribution of the CNT, i.e., $V_{CNT}(\xi, \eta)$, of the BFGN plate under a static load so that its weight becomes minimum. The optimization problem under the displacement constraint is described as follows:

$$
\begin{aligned}
\text{Minimize} \quad & W(\boldsymbol{V}_{CNT}) = \int_V \left(\rho_{CNT} V_{CNT} + \rho_p V_p \right) dV \\
& = \int_V \left[(\rho_{CNT} - \rho_p) V_{CNT} + \rho_p \right] dV \\
\text{Subjected to} \quad & \mathbf{K}(\mathbf{u})\mathbf{u} = \mathbf{F} \\
& u_{max} \leq u_0 \\
& 0 \leq V_{CNT}^i \leq 1, \quad i = 1, \dots, n
\end{aligned}
\tag{27}
$$

where $\boldsymbol{V}_{CNT} = (V_{CNT}^1, V_{CNT}^2, \dots, V_{CNT}^n)$ is the vector of design variables defined based on the approximation Eq. (2); $W(\boldsymbol{V}_{CNT})$ is the total weight of the BFGN plate; and u_{max} and u_0 are the maximum displacement and the allowable displacement, respectively.

Eq. (27) is a constraint optimization problem. To apply metaheuristic algorithms, a penalty method is adopted to transform Eq. (27) into a corresponding unconstrained problem. Consequently, the above objective function is modified to handle the constraint violation as [14]

$$W_p(\boldsymbol{V}_{CNT}) = (1 + \varepsilon_1 \vartheta)^{\varepsilon_2} W(\boldsymbol{V}_{CNT}) \tag{28}$$

where $W_p(\boldsymbol{V}_{CNT})$ is the penalized objective function; ε_1 and ε_2 are the coefficients of the penalty function; and $\vartheta = max\left[0,\left(\frac{u_{max}}{u_0}-1\right)\right]$. The choices of ε_1 and ε_2 can be referred to the work of Le-Duc et al. [14] and [22].

3.2. Balancing Composite Motion Optimization

In this section, the key features of the BCMO algorithm are recalled. At the beginning of the optimization process, the BCMO creates randomly an initial population including NP candidate solutions in a given search domain. First, a population of design variables x_i is initialized by the uniform distribution within the solution space as

$$x_i = x_i^{LB} + \left(x_i^{UB} - x_i^{LB}\right)rand[0,1], \quad i = 1, \dots, d \tag{29}$$

where x_i^{LB} and x_i^{UB} are the lower and upper boundaries of the ith individual and d is the number of design variables. Accordingly, all the individuals are sorted based on their value of the objective function as $x = argsort(f(x))$ where $f(x)$ is the objective function of the population.

Throughout the optimization iterations, the position of each individual can be updated via the following mechanism:

$$x_i^{t+1} = x_i^t + v_i \tag{30}$$

where x_i^{t+1} and x_i^t are the positions of the ith individual in the current t^{th} and next $(t+1)^{th}$ generations, respectively, whereas v_i is its composite motion in the solution space. Generally, v_i can be defined by a composite of two motion components including the relative movement of the current i^{th} individual with respect to the better j^{th} individual ($j > i$ or $j = i = 1$), denoted by $v_{i/j}$, and the relative motion between the jth individual to the global instant optimal point x_{Oin}, denoted by v_j. The general form of v_i, $v_{i/j}$ and v_j are defined as

$$v_i = v_{i/j} + v_j \tag{31}$$

$$v_j = L_{GS} \times dv_j \times \left(x_{Oin} - x_j\right) \tag{32}$$

$$v_{i/j} = dv_{i/j} \times \left(x_j - x_i\right) \tag{33}$$

where L_{GS} and the two parameters $dv_{i/j}$ and dv_j are chosen via the following selection rule corresponding to a random number TV as

$$L_{GS} = \begin{cases} e^{\frac{-1}{d}\frac{j}{NP}\left(x_j - x_{Oin}\right)^2} & ; \quad TV > 0.5 \\ e^{\frac{-1}{d}\left(1-\frac{j}{NP}\right)\left(x_j - x_{Oin}\right)^2} & ; \quad otherwise \end{cases} \tag{34}$$

$$\{dv_{i/j}, dv_j\} = \begin{cases} rand(1,d) & ; \quad TV > 0.5 \\ -rand(1,d) & ; \quad otherwise \end{cases} \tag{35}$$

In the above mechanism, x_{Oin} is assigned via the selection between the previous best (i.e., x_1^{t-1}) and a trial individual u_1^t based on their values of the objective function as follows:

$$x_{Oin}^t = \begin{cases} u_1^t & ; \quad f(u_1^t) < f(x_1^{t-1}) \\ x_1^{t-1} & ; \quad otherwise \end{cases} \tag{36}$$

where the trial individual u_1^t is determined by using the information from the previous generation as

$$u_1^t = \frac{LB+UB}{2} + v_{k1/k2}^t + v_{k2/1}^t \tag{37}$$

with $[LB, UB]$ being the lower and upper bounds of the design space whereas $v_{k1/k2}^t$ and $v_{k2/1}^t$ denoting the pseudo relative movements of the k_1^{th} individual with respect to the k_2^{th} individual and the k_2^{th} individual with respect to the previous best one computed from Eq. (33), respectively. In general, k_1 is randomly chosen in a range of [2, NP] and $k_2 < k_1$. More details about the BCMO can be found in the original work of Le-Duc et al. [14].

3.3. Artificial Neural Network

The most widely used artificial neural network type for problem approximations, which is adopted herein, is the multi-layer perceptron network (MLP) [15]. The MLP network is built with one input layer, one output layer, and an arbitrary number of hidden layers. In the input and output layers, one neuron is assigned to each input parameter or output parameter whereas the number of neurons in hidden layers can be arbitrary. The neurons of one layer relate to each neuron of the previous layer, but information only flows in the forward direction which is from the input towards the output layers. Each connection has its connection weight and bias as adjusting parameters. Finally, the output values of the neurons in the current layer can be obtained with the activation function of a sum of output values of the neurons in the previous layer multiplied by corresponding connection weights between layers and the biases of the current layers as

$$a_i^{[l]} = g\left(\sum_{j=1}^{n_{l-1}} W_{ij}^{[l-1]} a_j^{[l-1]} + b_i^{[l-1]}\right) \tag{38}$$

where $W_{ij}^{[l-1]}$ denotes the connection weight between the ith neuron in the lth layer and the jth neuron in the $(l-1)^{th}$ layer whereas $b_i^{[l-1]}$ is the bias between the ith neuron in the lth layer and any neuron in the $(l-1)^{th}$ layer. In addition, $a_i^{[l]}$ is the output value of the ith neuron in the lth layer and g is the selected activation function. More details on ANN can be found to in the works of Haykin and Lippmann [23] and Truong et al. [15].

To assess the accuracy of output results from the ANN architecture, the mean square error (MSE)-based loss function is used in this work and it can be expressed explicitly as

$$\mathcal{L}_{MSE} = \frac{1}{m}\sum_{i=1}^{m}\left(a_i^{[L]} - a_i^{exact}\right)^2 \tag{39}$$

where m is the number of data used for training the ANN model; $a_i^{[L]}$ is the predicted output value obtained from the ANN model; and a_i^{exact} is the exact output value obtained from the data.

3.4. Artificial Neural Network - Balancing Composite Motion Optimization Approach

In the optimization problem under consideration, if values of the objective function and corresponding constraints required in the BCMO iterations are evaluated directly by IGA, the high computational cost induced by the geometrically nonlinear analyses will render the optimization procedure cumbersome. Thus, an approach based on the combination of ANN and BCMO is alternatively employed to solve the optimization problem. In the proposed method, the ANN-based surrogate model of both the objective function and constraint is properly constructed first and then integrated directly into the BCMO framework. Consequently, this strategy not only dramatically saves the computational cost during the optimization process but also ensures the possibility of global convergence of the optimal solution. The effectiveness

and efficiency of the proposed optimization framework are examined via the selected numerical examples presented in the following section.

4. Numerical Examples

In this section, two numerical examples for the material distribution optimization of BFGN square and circular plates considering geometrical nonlinearity are presented. The efficiency of the ANN-BCMO approach is demonstrated by comparing the results directly with those obtained from the BCMO together with the direct isogeometric analysis (IGA-BCMO). For the optimization process, the number of individuals is chosen as $NP = 30$ and the iterative process is terminated when either the following stopping criterion $|f_{mean}/f_{best} - 1| \leq 10^{-6}$ [24] is met or the number of iterations reaches 50,000. In the numerical study, the following material properties are used: $E_{CNT} = 900\ GPa$, $E_p = 1.9\ GPa$, $v_{CNT} = 0.28$, $v_p = 0.34$, $\rho_{CNT} = 2100\ kg/m^3$, $\rho_p = 1050\ kg/m^3$, $k_0 = 0.2$, $k_w = 0.1$, $k_l = 0.5163$, $\zeta = 3.7029$, $W_{CNT} = 100\%$. All the analyses are implemented in MATLAB on a laptop computer Core i7-1165G7 CPU @ 2.8 GHz with 16 GB RAM of memory.

4.1. Square Plate

This example concerns the minimization of the weight of a simply supported square BFGN plate with $L = 1.2\ m$ and $h = 0.02L$. The plate is subjected to a uniformly distributed load $q = 10^6\ N/m$ and the allowable displacement at the plate center is given by $u_0 = 0.1\ m$. In this example, a mesh using quartic NURBS basis functions with 6×6 control points is employed in the response analysis and a mesh with only 3×3 control points is utilized to represent the optimal material distribution.

The effect of various factors such as the size of dataset, optimizers, activation functions, and neural network architecture on the accuracy of the surrogate model is first investigated. In Table 1, the RMSE errors and computational time associated with different sizes of the dataset and a fixed ANN architecture 9-50-50-50-2 are presented and compared. It is apparent that when the number of samples is larger than 1200, the error decreases insignificantly, but the construction time increases dramatically. Thus, the dataset of 1200 samples is employed in the subsequent analyses. Tables 2 and 3 present the MSE and RMSE, respectively, for the training and testing processes with different optimizers and activation functions. Among all options considered, the combination of Adam and ReLU yields the minimum MSE and RMSE and is, therefore, utilized for further calculations. Next, an investigation to find the optimal ANN architecture is conducted by using the grid search method with different numbers of hidden layers and the number of neurons per layer. As can be seen from the results presented in Table 4, the best optimal model architecture is the one with 3 hidden layers with 50 neurons in each hidden layer. All obtained information is then employed to construct the ANN surrogate model in the optimization scheme.

Table 1: RMSE errors and constructing time for the test set of each dataset

Number of samples	Architecture	RMSE (%)	Constructing time (sec)
400	9-50-50-50-2	0.85	3,914.42
1,200	9-50-50-50-2	0.62	11,611.41
2,400	9-50-50-50-2	0.61	23,414.86
4,000	9-50-50-50-2	0.59	39,910.05

Table 2: Comparisons of MSE of the surrogate model trained by different optimizers and activation functions

Optimizer	Activation					
	ReLU		Sigmoid		Tanh	
	Training	Test	Training	Test	Training	Test
SGDM	6.64	7.31	91.33	109.66	1.62	1.86
RMSprop	0.11	0.29	3.57	5.63	0.61	0.92
Adam	0.09	0.19	1.28	2.19	0.55	0.88

Table 3: Comparisons of RMSE of the surrogate model trained by different optimizers and activation functions

Optimizer	Activation					
	ReLU		Sigmoid		Tanh	
	Training	Test	Training	Test	Training	Test
SGDM	3.65	3.82	13.52	14.81	1.80	1.93
RMSprop	0.48	0.76	2.67	3.36	1.10	1.36
Adam	0.42	0.62	1.60	2.09	1.05	1.33

Table 4: Comparisons of MSE, RMSE, and training time with different neural network architecture.

Hidden Layers	Neurons	MSE ($\times 10^{-4}$)		RMSE (%)		Training time (sec)
		Training	Validation	Training	Validation	
3	50	0.09	0.19	0.42	0.62	269.14
	100	0.12	0.20	0.49	0.63	386.84
	150	0.18	0.20	0.60	0.64	531.00
	200	0.20	0.20	0.63	0.63	702.31
	250	0.13	0.20	0.52	0.64	888.64
	300	0.11	0.25	0.47	0.71	1,165.13
4	50	0.13	0.23	0.50	0.68	313.03
	100	0.18	0.24	0.60	0.70	515.66
	150	0.24	0.23	0.69	0.68	665.70
	200	0.17	0.21	0.58	0.65	988.23
	250	0.18	0.24	0.60	0.70	1,316.91
	300	0.22	0.26	0.67	0.72	1,685.72
5	50	0.23	0.24	0.67	0.70	371.89
	100	0.15	0.26	0.54	0.72	584.17
	150	0.21	0.27	0.64	0.73	918.16
	200	0.14	0.24	0.52	0.70	1,283.08
	250	0.26	0.26	0.73	0.72	1,419.55
	300	0.18	0.28	0.60	0.75	2,056.61
Total time						16,061.77

Once the surrogate model is obtained, it is then used in the optimization process for calculating the objective function and constraint. Due to the stochastic nature of the metaheuristic algorithm, 20 runs of ANN-BCMO are performed, but only one run of IGA-BCMO is tested due to its extremely high computation cost required. The comparison between the best result

of the ANN-BCMO runs and the one by IGA-BCMO is presented in Table 5 whereas the convergence of the penalized objective and the optimal volume fraction of CNT are reported in Figures 2 and 3, respectively. It is noted that the computational time of ANN-BCMO includes the time for generating data, the training time by using grid search, and the optimization time by using BCMO. As can be seen from those results, the ANN-BCMO outperforms the conventional strategy (i.e., IGA-BCMO) for the optimization problem considered.

Table 5: **Comparison of the objective function and computational time obtained by ANN-BCMO and IGA-BCMO for the BFGN square plate**

	ANN-BCMO	IGA-BCMO	Difference (%)
Weight	7,877.873	7,868.858	0.11
Computational time (sec)	27,455.59	240,263.06	
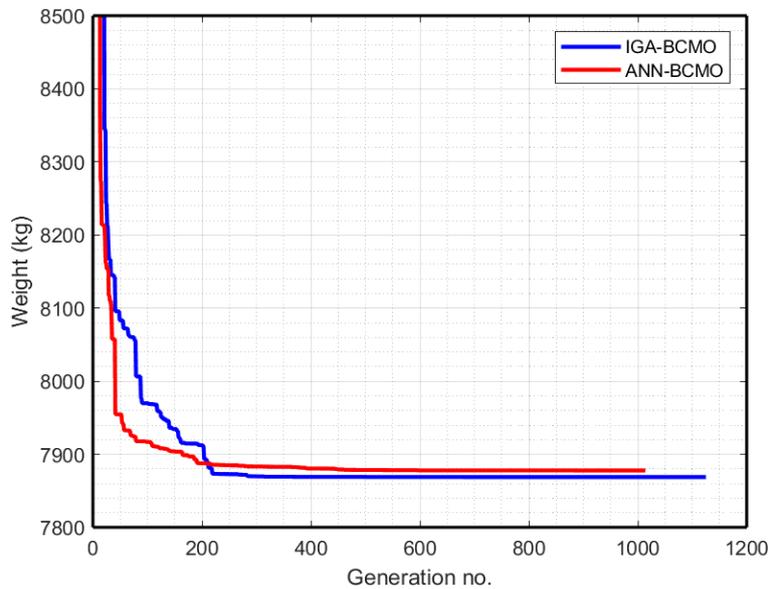
Figure 2. Convergence history of IGA-BCMO and ANN-BCMO for the BFGN square plate.
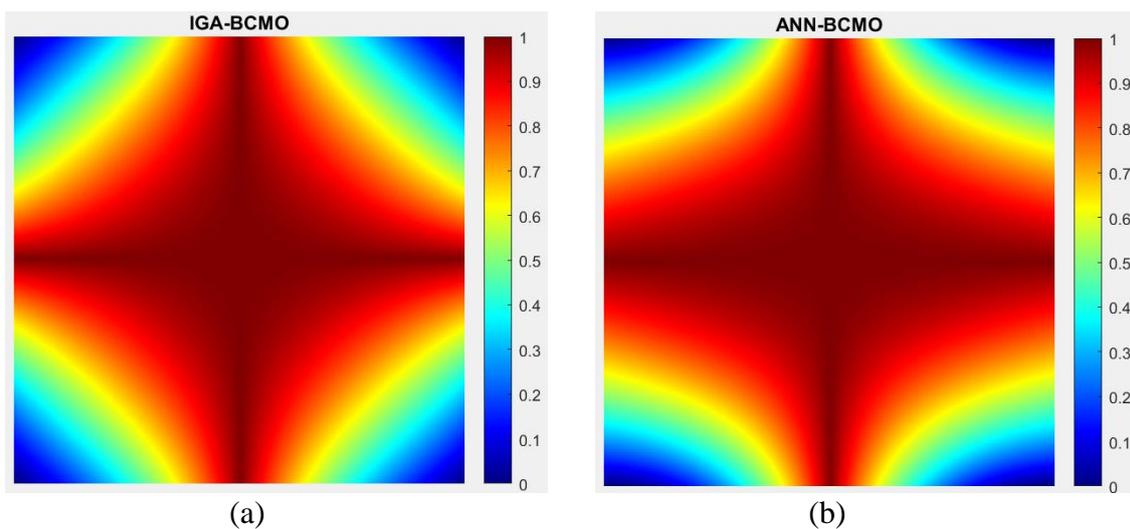
Figure 3. Optimal distribution of CNT for the BGFN square plate using (a) IGA-BCMO, and (b) ANN-BCMO.

4.2. Circular Plate

Finally, the problem of finding optimal CNT volume fraction distribution of a clamped circular plate with the radius $R = 100\ m$ and the thickness $t = 2\ m$ is considered. The plate is loaded by a concentrated point load $P = 1 \times 10^6\ N$ at its center. For this particular case, it is aimed to minimize the total weight of the structure while limiting the center displacement to $u_0 = 0.1\ m$. In addition, the response analysis is performed with a mesh using quartic NURBS basis functions with 8×8 control points whereas the optimal CNT volume fraction is approximated by a mesh with only 3×3 control points.

The investigation to find the best ANN architecture for this particular problem is carried out first and results are reported in Table 6. Based on such information, the ANN architecture with 3 hidden layers with 100 neurons in each hidden layer is employed for solving the optimization problem. Again, 20 runs of ANN-BCMO and one run of IGA-BCMO are performed. The comparisons between the optimal results for the BFGN circular plate obtained by the best ANN-BCMO run and IGA-BCMO are shown in Table 7 whereas the convergence of the objective function and the schematic of optimal distribution of CNT are illustrated in Figures 4 and 5, respectively. Again, although the minimum weights obtained by both methods are similar, the computational cost required by the proposed ANN-BCMO approach is much less than that required by the IGA-BCMO method.

Table 6: The corresponding MSE, RMSE, and training time of the trained model with respect to different neural network architectures.

Hidden Layers	Neurons	MSE (x 10^{-6})		RMSE (%)		Training time (sec)
		Training	Validation	Training	Validation	
3	50	4.69	6.54	0.31	0.36	274.33
	100	4.45	6.33	0.30	0.36	366.59
	150	4.64	6.65	0.30	0.36	495.48
	200	9.01	6.41	0.42	0.36	676.22
	250	9.59	7.69	0.44	0.39	885.52
	300	4.45	6.49	0.30	0.36	1,062.55
4	50	9.49	7.21	0.44	0.38	307.02
	100	8.50	7.44	0.41	0.39	494.59
	150	21.59	6.99	0.66	0.37	682.41
	200	7.43	8.87	0.39	0.42	1,006.72
	250	6.34	10.12	0.36	0.45	1,127.20
	300	7.27	8.37	0.38	0.41	1,485.39
5	50	6.16	8.08	0.35	0.40	386.05
	100	12.05	10.45	0.49	0.46	612.56
	150	8.18	9.89	0.40	0.44	870.34
	200	9.61	7.25	0.44	0.38	1,181.27
	250	14.60	8.02	0.54	0.40	1,459.20
	300	10.62	6.89	0.46	0.37	1,950.38
Total time						15,323.81

Table 7: Comparison of the objective function and computational time obtained by ANN-BCMO and IGA-BCMO for the BFGN circular plate

	ANN-BCMO	IGA-BCMO	Difference (%)
Weight	68,113,432	68,011,980	0.15
Computational time (sec)	28,118.86	220,341.80	
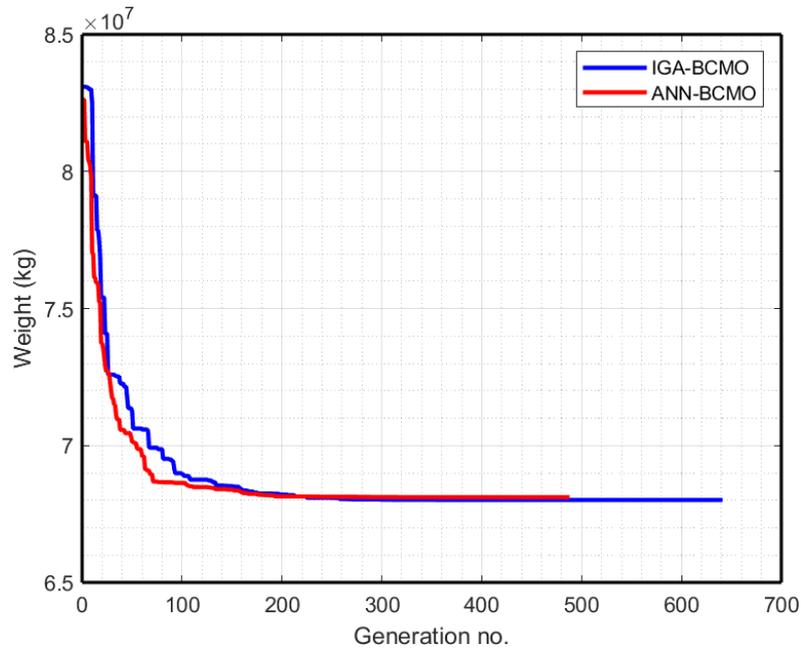
Figure 4. Convergence history of IGA-BCMO and ANN-BCMO for the BFGN circular plate.
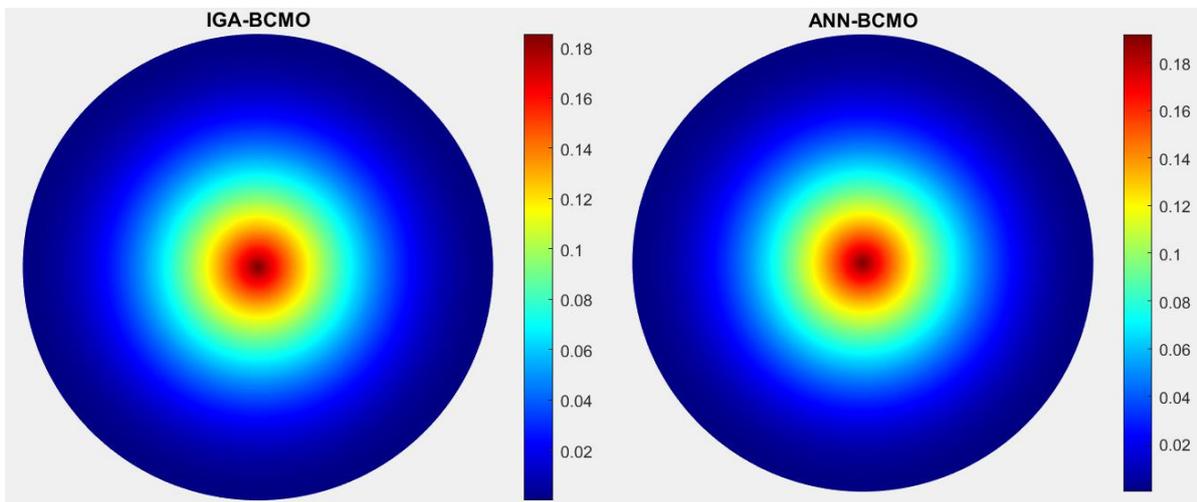
(a) (b)

Figure 5. Optimal distribution of CNT for the BGFN circular plate using (a) IGA-BCMO, and (b) ANN-BCMO.

5. Conclusion

This study has employed, for the first time, an integrated approach based on the ANN and BCMO to tackle the material distribution optimization problem of in-plane BFGN plates. The core of the proposed framework is based on the replacement of the high-cost geometrically nonlinear isogeometric analysis with Kirchhoff-Love plate model and Von Kármán nonlinearity by an effective ANN-based surrogate model in the framework of BCMO optimization loops. The optimization of interests is to minimize the weight of the structure under the displacement-based constraints. To enhance the possibility of exploring optimal solutions, the NURBS-based representation of the material volume fraction distribution has been introduced in which the control values of the volume fraction of CNT are chosen as the primary design variables. Selected numerical examples of BFGN square and circular plates have been considered to demonstrate the performance of the ANN-BCMO scheme. By comparing results obtained from the ANN-BCMO and a conventional strategy based on IGA-BCMO, the superior effectiveness and efficiency of the proposed scheme are revealed.

Acknowledgment

References

[1] Koizumi, M. (1993) The concept of FGM, *Ceramic transactions* **34**, 3-10.
[2] Yamanouchi, M., Koizumi, M., Hirai, T. and Shiota, I. (1990) *Proc. First Int. Symp. on Functionally Gradient Materials*, FGM Forum Tokyo, Japan.
[3] Kinloch Ian, A., Suhr, J., Lou, J., Young Robert, J. and Ajayan Pulickel, M. (2018) Composites with carbon nanotubes and graphene: An outlook, *Science* **362**(6414), 547-553.
[4] Moita, J. S., Araújo, A. L., Correia, V. F., Mota Soares, C. M. and Herskovits, J. (2018) Material distribution and sizing optimization of functionally graded plate-shell structures, *Composites Part B: Engineering* **142**, 263-272.
[5] Qin, X. C. and Dong, C. Y. (2021) NURBS-based isogeometric shape and material optimization of curvilinearly stiffened plates with FGMs, *Thin-Walled Structures* **162**, 107601.
[6] Goupee, A. J. and Vel, S. S. (2006) Optimization of natural frequencies of bidirectional functionally graded beams, *Structural and Multidisciplinary Optimization* **32**(6), 473-484.
[7] Roque, C. M. C. and Martins, P. A. L. S. (2015) Differential evolution for optimization of functionally graded beams, *Composite Structures* **133**, 1191-1197.
[8] Roque, C. M. C., Martins, P. A. L. S., Ferreira, A. J. M., and Jorge, R. M. N. (2016) Differential evolution for free vibration optimization of functionally graded nano beams, *Composite Structures* **156**, 29-34.
[9] Cho, J. R. and Ha, D. Y. (2002) Volume fraction optimization for minimizing thermal stress in Ni-Al2O3 functionally graded materials, *Materials Science and Engineering A* **334**(1-2), 147-155.
[10] Kou, X. Y., Parks, G. T. and Tan, S. T. (2012) Optimal design of Functionally Graded Materials using a procedural model and Particle Swarm Optimization, *CAD Computer Aided Design* **44**(4), 300-310.
[11] Fereidoon, A., Sadri, F. and Hemmatian, H. (2012) Functionally graded materials optimization using particle swarm-based algorithms, *Journal of Thermal Stresses* **35**(4), 377-392.
[12] Alshabatat, N. T., Myers, K. and Naghshineh, K. (2016) Design of in-plane functionally graded material plates for optimal vibration performance, *Noise Control Engineering Journal* **64**(2), 268-278.
[13] Nabian, M. and Ahmadian, M. T (2011) Multi-objective optimization of functionally graded hollow cylinders, *ASME 2011 International Mechanical Engineering Congress and Exposition, IMECE 2011* **8**, 583-590.
[14] Le-Duc, T., Nguyen, Q. H. and Nguyen-Xuan, H. (2020) Balancing composite motion optimization, *Information Sciences* **520**, 250-270.
[15] Truong, T. T., Lee, S. and Lee, J. (2020) An artificial neural network-differential evolution approach for optimization of bidirectional functionally graded beams, *Composite Structures* **233**, 111517.

[16] Asteris, P. G., Lemonis, M. E., Nguyen, T. A., Van Le, H. and Pham, B. T. (2021) Soft computing-based estimation of ultimate axial load of rectangular concrete-filled steel tubes, *Steel and Composite Structures* **39**(4), 471-491.

[17] Thanh Duong, H., Chi Phan, H., Le, T. T. and Duc Bui, N. (2020) Optimization design of rectangular concrete-filled steel tube short columns with Balancing Composite Motion Optimization and data-driven model, *Structures* **28**, 757-765.

[18] Huynh, T. A., Lieu, X. Q. and Lee, J. (2017) NURBS-based modeling of bidirectional functionally graded Timoshenko beams for free vibration problem, *Composite Structures* **160**, 1178-1190.

[19] Cottrell, J. A., Hughes, T. J. and Bazilevs, Y. (2009) *Isogeometric analysis: toward integration of CAD and FEA*, John Wiley & Sons.

[20] El-Ashmawy, A. M., Xu, Y. and El-Mahdy, L. A. (2021) Mechanical properties improvement of bi-directional functionally graded laminated MWCNT reinforced composite beams using an integrated tailoring–optimization approach, *Microporous and Mesoporous Materials* **314**, 110875.

[21] Yin, S., Yu, T., Bui, T. Q. and Nguyen, M. N. (2015) Geometrically nonlinear analysis of functionally graded plates using isogeometric analysis, *Engineering Computations* **32**(2), 519-558.

[22] Xu, Z., Cui, Y. and Li, B. (2021) A quantum-inspired particle swarm optimization for sizing optimization of truss structures, *Journal of Physics: Conference Series* **1865**(4), 042127.

[23] Haykin, S. and Lippmann, R. (1994) Neural networks, a comprehensive foundation, *International journal of neural systems* **5**(4), 363-364.

[24] Pham, H. A., Huong, T. Q. and Dang, H. X (2022) Optimal Design of Functionally Graded Sandwich Porous Beams for Maximum Fundamental Frequency Using Metaheuristics, *Lecture Notes in Mechanical Engineering* 229-239.

Particle Swarm Optimization for Minimum Connection Placement in Prefabricated Modular Housing Design

*Thamonwan Suwannasri, Arnut Sutha, Thu Huynh Van, and †Sawekchai Tangaramvong

Applied Mechanics and Structures Research Unit, Department of Civil Engineering, Chulalongkorn University, Bangkok 10330, Thailand.

*Presenting author: thamonwan.ss@hotmail.com
†Corresponding author: sawekchai.t@chula.ac.th

Abstract

The modular housing has increasingly gained the popularity among communities in views of its fast construction and minimum site preparation. The prefabrication technology makes it possible for the easy storage and mass logistics to construction sites. At variance with typical frames, the modular house composes a series of prefabricated lightweight steel panels that are assembled through the precisely designed connections (nuts and bolts and/or welding). The total cost and assembly time are proportional to the number of connections predefined. This paper, therefore, proposes a so-called Comprehensive Learning Particle Swarm Optimization (CLPSO) method to determine the minimum placement of connections necessarily required for the assembly of semi-detached modular houses under applied external forces. The limit state design criteria comply with AISC-LRFD (2016) specifications. The connections adopt the combination of five specially designed nuts-and-bolts patterns, where their possible locations are predefined. Their behaviors are described by the compatibility conditions of displacements at some specific degrees of freedom (in 3D space) associated with interface nodes of steel panels. The proposed CLPSO approach assigns the binary variables to all connection locations and efficiently determines their optimal placement leading to the minimum construction cost. The applications of the proposed CLPSO method are illustrated through the realistic design of public residential houses managed by National Housing Authority of Thailand.

Keywords: Comprehensive learning particle swarm optimization; Minimum connection placement; Prefabrication; Steel lightweight structures; Modular housing.

Introduction

The key criteria underlying national housing development that well serve high population density consider rapid construction, cost optimization, energy efficiency, environmental friendliness and responsiveness to residents' needs. The approach often applies the principle modular design system adopting standard sizes of material cross-sections and a principal member length to minimize material wastes. The prefabricating construction method [1-6] involves material production in the factory and member assembly on site. This advantageously enhances the high precision and minimizes constructional periods [2, 3, 7-9].

One of the challenges underlying the development of prefabrication technology is the design of joints that are typical but sufficiently strong to connect all required structural members. It generally requires the large number of joints but incurs expensive resources over the assembling procedures. The minimum placement of connections [2, 10-12] is essential for the cost-effective management and construction. This paper therefore proposes the comprehensive learning particle swarm optimization (CLPSO) method [13] to minimize the total number of connections employed in the prefabricated modular housing design. More explicitly, five main connection

types with five combinations from these principles are considered as the standard joints employed at the pre-defined possible locations in steel frame panels. The connections are modeled using the zero-length line finite elements, which can describe the compatibility of co-nodal displacements at the specified degrees of freedom between steel panels. The CLPSO approach provides the optimal distribution of required connections, such that the assembled house complies with the limit state design AISC-LRFD (2016) specifications.

Connection system and model

The connections between structure members required for the housing fabrication consider three main categories, namely inter-module connection, intra-module connection and module to foundation connection. The five typical connection Types A to E in Fig. 1 employed as the principal joints and their functionality are summarized in Table 1.
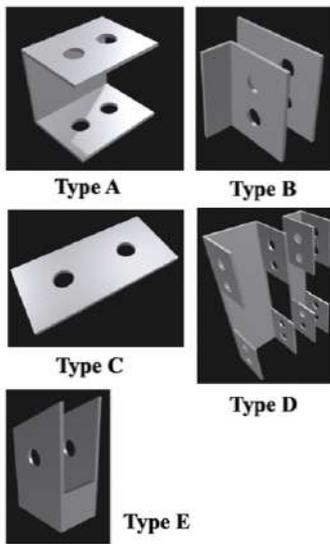
Table 1. Principle connections and functionality

Connection Types	Usability
Type A	Connect the wall panels in the horizontal plane
Type B	Connect the wall panels at right angles
Type C	Connect the wall panels to the concrete beams
Type D	Connect the upper and lower wall panels Connect the roof panel
Type E	Connect the wall panel to the roof panel

Figure 1. Principle connections

In addition, five combinations, depicted in Fig. 2, selected from these five principal connections are assigned to the predefined nodal locations of steel panels, see Fig. 3. Each of which determines different connectivity conditions describing the compatibility of degrees of freedom (DOFs) at the interfacing nodes.

The connections are modeled using zero-length line elements in Fig. 4 that explicitly define the connectivity conditions associated with the five principle and five combined connection types described in Table2. The zero length members are first allocated to all possible connecting locations at nodes of individual steel panels, and later are minimized using the optimization process.
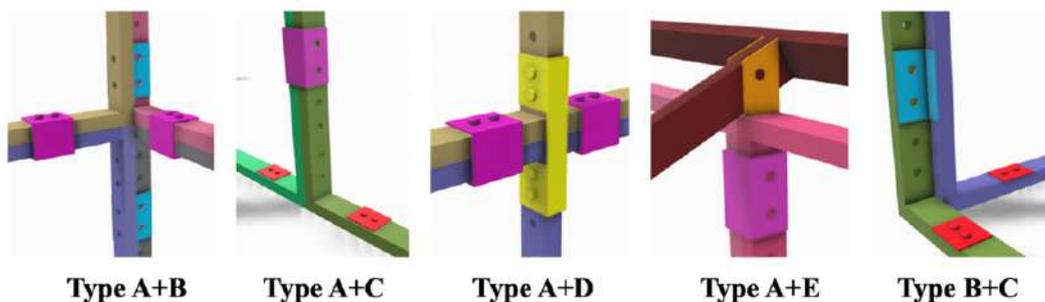
Figure 2. Combination from five principal connection Types A to E

Table 2. Connectivity conditions at interface nodal DOFs

TYPE	DOF					
	U_1	U_2	U_3	R_1	R_2	R_3
A	F	F	F	T	F	F
B	F	F	F	T	F	T
C	T	T	T	F	T	T
D	F	F	F	T	F	F
E	F	F	F	F	T	T
A+B	F	F	F	T	F	T
A+C	T	T	T	T	T	T
A+D	F	F	F	T	F	F
A+E	F	F	F	T	T	T
B+C	T	T	T	T	T	T
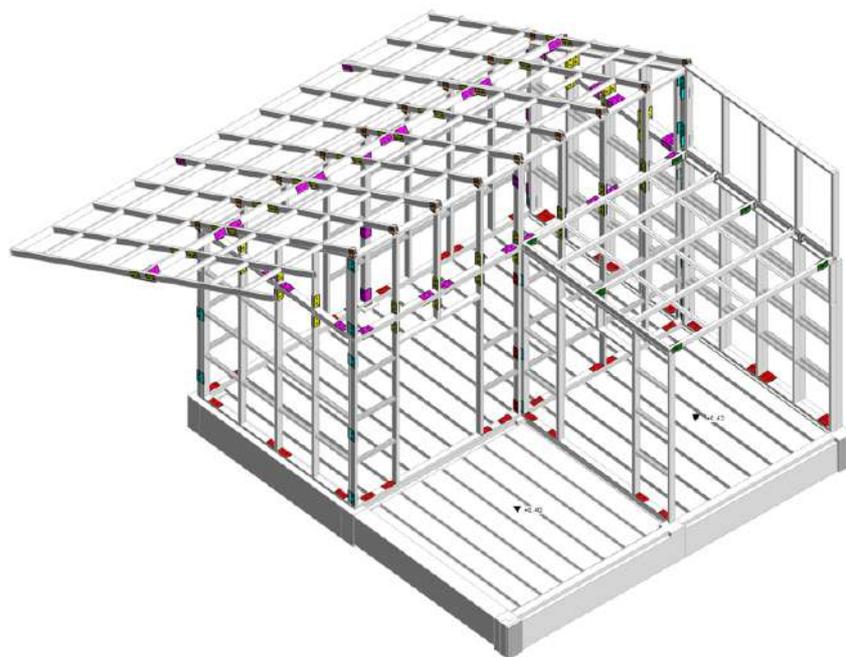
Figure 3. Initial placement of connections in the prefabricated house
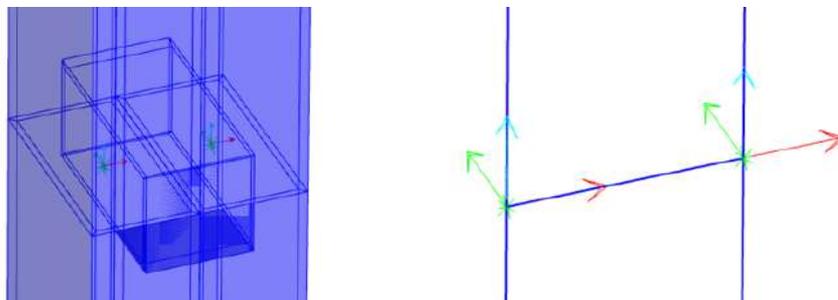
Figure 4. Zero-length line element modelling nodal connectivity

State problem

The optimization problem minimizes the total number of bolts-and-nuts connection placements necessarily for the assembly of galvanized steel structural panels and hence prefabricated house depicted in Fig. 5. The individual steel panels with a typical dimension of 2.4×2.4 m^2 are drawn in Fig. 6. Three following optimal connection placement cases are performed: 1) determination of the active connections at predefined nodal locations; 2) limiting the maximum number of connection locations and types; and 3) limiting only the maximum number of connection locations. The minimum connection placement design complies with ultimate strength and serviceability AISC-LRFD (2016) specifications.

The design process is encoded within a PYTHON environment, and the resulting design is analyzed using the commercial finite element analysis and design package, called SAP 2000.
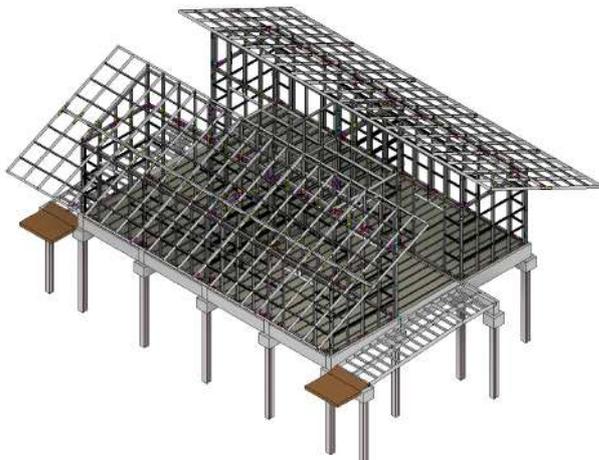
Figure 5. Prefabricated steel house
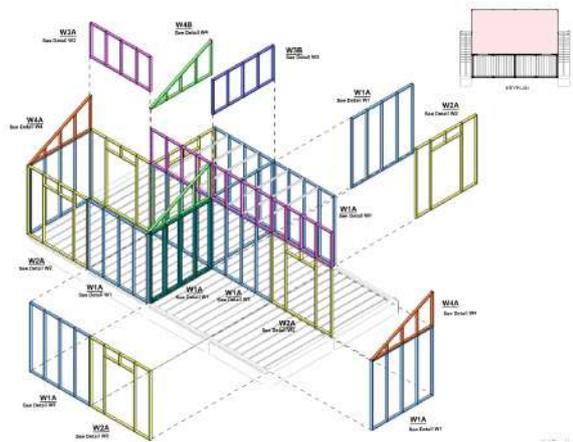
Figure 6. Unassembled steel panels

The prefabricated house in Fig. 5 consists of four main steel panels, depicted in Fig. 6. Due to its symmetry in both axis directions, only a quarter of the house can be modeled using the standard line finite element model in Fig. 7

The foundation system adopts the precast reinforced concrete foundations and ground beams. These are not considered as the design variables in the problem. The steel panels are connected on the ground beams using Type C connections providing the restraints in both translations and moment resistances in a perpendicular direction to the wall plane.
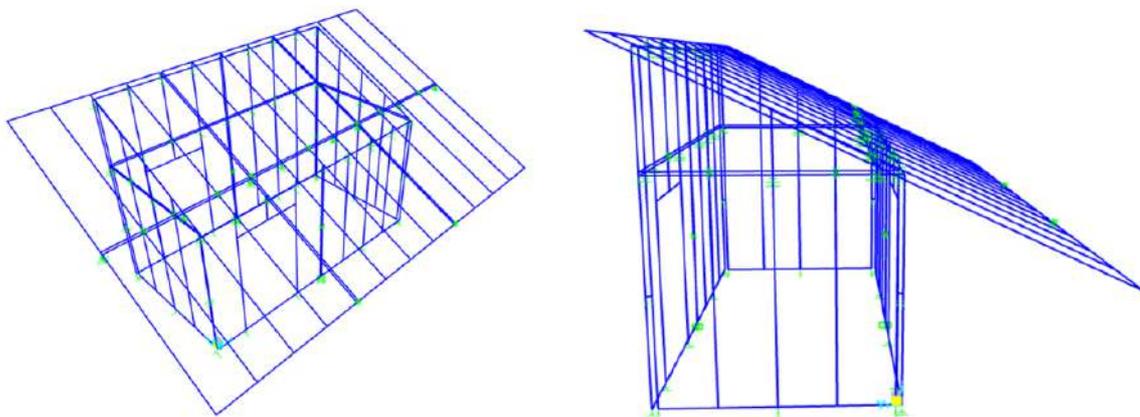
Figure 7. Finite element model of a quarter of prefabricated modular house

The structure system applies the load bearing wall panels [14], in which all steel members of the wall adopt the standard rectangular hollow section (RHS) strut members. The material properties are detailed in Table 3.

Table 3. Material properties

Properties	Symbol	Value	UNIT
Yield Strength of Cold-formed Steel (Grade SGC490)	F_y	300	MPa
Modulus of Elasticity	E	210000	MPa
Shear Modulus	G	80000	MPa
Density	ρ	7850	kg/m^3
Poisson's Ratio	ν	0.3	-

The applied loads and their combinations are given in Tables 4 and 5.

Table 4. Applied loads

Type	Loading	Unit
Dead Load	Self-weight	kN
Live Load	0.5	kN/m^2
Roof Live Load	0.88	kN/m^2
Lateral Load	0.5	kN/m^2

The structure analysis and design are based on the standard of load combinations from AISC using LRFD method as shown in Table 5, where D is a dead load, L live load, L_r roof live load and W wind load.

Table 5. Load combination

Case	Combination
1	1.4D
2	$1.2D + 1.6L + 0.5L_r$
3	$1.2D + 1.6L_r + 0.5W$
4	$1.2D + 1.0L + 0.5L_r + 1.0W$

Comprehensive learning particle swarm optimization

Based on the study of various Particle Swarm Optimization (PSO) problems [15-17], there is the difficulty of premature convergence for multimodal problems. Limiting learning to only Gbest may be an irrational decision. Even the most recent values can cause the answer to diverge from the global optimum. In addition, particles may be attracted and answered at a local optimum.

Comprehensive Learning Particle Swarm Optimization (CLPSO) algorithm, developed by Liang and Huang [13], particles learn only the Pbest value in each dimension. In all iterations of the particle group learning method, particles in all dimensions learn from different Pbest values of other dimensions for a couple of iterations. The Pbest value of particles can be used as a model to guide the way to optimum solution of other particles and each dimension of each particle can be learned from the Pbest value of the other particle's corresponding dimension.

During the search process, it is unable to determine whether the Pbest value of a particle's dimension is excellent or poor. Therefore, each particle dimension has an equal chance of

picking up new information from other particles. Each particle is updated in Eq. (1) with different velocity in Eq. (2):

$$x_{id}^{(k+1)} = x_{id}^{(k)} + v_{id}^{(k+1)} \tag{1}$$

$$v_{id}^{(k+1)} = w \times v_{id}^{(k)} + rand \times (Pbest_{(fi)d} - x_{id}^{(k)}) \tag{2}$$

$$P_{ci} = 0.05 + \frac{0.45\left(\exp\left(\frac{10(i-1)}{N_p-1}\right)\right)}{\exp(10)-1} \tag{3}$$

$$w_k = w_{max} - k \times \frac{w_{max}-w_{min}}{iter_{max}} \tag{4}$$

$x_{id}^{(k)}$ and $v_{id}^{(k)}$ denote the position and velocity of the i-th particle at k-th time, respectively. $Pbest_{(fi)d}$ is the best position value of the i-th particle for each dimension, d. It can be the value of the particle itself, or another particle. For each particle of some dimensions, the Pbest value of the other particle is randomly, *rand* within the $[0,1]$ interval chosen based on learning probability, Pc_i value in Eq. (3) by defined as the social learning probability to learn from while other dimensions, where N_p is the total number of particle populations. Weight function, w_k is defined by a collection of steps that result in a uniform search in the initial stage and a local search. w_k in Eq. (4) is the weighting function for k iterations during the velocity solution process of each particle, where w_{max} and w_{min} are equal to 0.9 and 0.4, respectively.

For iterations k times during the answer process, the velocity of each particle corresponds to Eq. (5):

$$v_{id}^{(k)} = \min(v_d^{max}, \max(-v_d^{max}, v_{id})) \tag{5}$$

where,
$$v_{max} = 0.2(x_{max} - x_{min})$$

To find a suitable solution to optimize number of connections of this modular structure problem, the default variable is the activeness of restraint at each DOF in each node. The position variables x_{min} and x_{max} have value of 'zero' and 'one' for inactive (False) and active (True) for the DOF restraints, respectively.

Optimization formulation

Design Variables

The design of the rigid frame structure defines a galvanized steel material RHS cross-section of typically 38-76mm. depth and in steel thickness of 2.3mm., which is the standard cross-sectional size according to the building materials market. In terms of proper connection placement, there is a discontinuity. The proper design of the positioning of the connection is discrete structural optimization, which is considered a Non-Smooth Problem. The combinations of forces are defined in the design and analysis of the prefabricated modular housing structure.

Objective Function

To analyze the location of the appropriate connection and analyze the appropriate connection type. It is necessary to consider the sum of the numbers for which the connection is active, and then used the result to analyze the design of various types of connections. Therefore, the objective function is minimum number of active restraints at each DOFs of connection as follows:

$$Minimize\ C(x) = \sum_{i=1}^{nv} x_i \quad : \quad x_i = \begin{cases} 1 & : \ when\ restraint\ is\ active \\ 0 & : \ when\ restraint\ is\ not\ active \end{cases} \tag{6}$$

where C(x) is the total active restraints of all considering DOF of connections; *nv* is the number of variables that define different connection conditions in each case.

Constraint

In order to properly design the connection of structures, the material strength constraints for the stability of the structure and the serviceability constraints of the structure must be determined.

1. Strength Constraints
   Since the structure used in the connection placement optimization is light steel framing consists of galvanized steel RHS-sections of typically 38-76mm depth and in steel thickness of 2.3mm. For each member subjects to axial force, shear force and bending moment, strength constraints of members are subject to AISI S100-2016 & ANSI/AISC 360-16 specifications shown in the following.

$$\lambda_i^\sigma = \frac{|\sigma_i|}{\sigma_i^a} - 1 \leq 0 \tag{7}$$

   where σ_i are stresses of *i*-th element.

2. Serviceability Constraints
   In defining serviceability constraints are presented as displacement constraints considering the critical part of the structure. The constraints set out include the frame drift ratio *Δ/H* and the inter-story drift ratio *d/H* under the service loads using the load combination [18].

$$\begin{cases} \lambda_i^d = \frac{|\Delta^*|}{\Delta_u} - 1 \leq 0 \\ \lambda_j^d = \frac{|d^*|}{d_u} - 1 \leq 0 \end{cases} \tag{8}$$

   The frame drift ratio Δ^* must comply with the allowable horizontal displacement Δ_u defined by the code, which is equal to H/120. The maximum inter-story displacement d^* must meet the allowable inter-storey displacement d_u established by the code, adopted equal to H/360, where H is height of frame.

Optimal results of minimum connection placement

The optimization analyzed the model results with SAP2000 and encoded within the Spyder having a direct interface, through Application Programming Interface (API) and use the data to analyze with the Comprehensive Learning Particle Swarm Optimization (CLPSO) using the features of the connection and loading under the load of the AISC-LRFD 2016 Specifications design standard. The CLPSO algorithm takes the random value of the variables and compares the results of the structural analysis of such situations and develops the solution to convergence to the appropriate answer. The answer can be compared with the connection type defined at each connection location.

Conditions for various case studies and algorithms are defined by coding in Python. The result of structure analysis is render in the SAP2000 software. The results of each appropriate analysis may give different types of connections or locations of connections due to proper design can provide several options that pass the constraints imposed by the standard requirements. However, analysis of the results showed that the number of connections was lower in all cases, as shown in Table 6.

All results were re-analyzed to recheck the displacement and stresses of each member, in which the results match the specified conditions. The results of the structural analysis for the optimization solution sample are shown in Table 7. Therefore, it can be seen that the number of connections of the structure can be reduced. This connection placement optimization design

can be taken into account in conjunction with other factors that affect construction costs, such as the shape or thickness of the steel plate used in the connection design as well as the number of bolts and nuts. Consideration with other factors will help to come up with the most suitable model for the project.

Table 6. Optimal design solutions

Case	Result	Remark				
	SUM	Type A	Type B	Type C	Type D	Type E
original	117	38	16	24	25	14
1	66	22	14	20	4	6
2	66	24	6	16	14	6
3	87	30	6	24	13	14

Table 7. Structural responses of designed solutions

Case	Result								
	Max U_1 (mm.)	Max U_2 (mm.)	Max U_3 (mm.)	Max Axial (kN)	Max Shear2 (kN)	Max Shear3 (kN)	Max Torsional (kN-m)	Max Moment2 (kN-m)	Max Moment3 (kN-m)
original	5.26	5.59	4.27	3.84	1.13	1.49	0.104	0.645	0.332
1	7.87	23.7	20.4	4.82	1.94	2.86	0.247	0.998	0.542
2	8.00	22.8	9.42	6.31	5.93	2.62	0.309	1.102	0.901
3	8.74	19.6	7.26	5.56	1.94	2.73	0.177	1.078	0.537

Concluding remarks

This paper presents a method for determining the optimal connection placement with the CLPSO algorithm by locating the connectable nodes and assigning different connectivity conditions as the compatibility of degrees of freedom at the nodes. An example of the structure being analyzed is the load bearing wall panels system with the five principle and five combined connection types, where all steel members of wall panels have the standard rectangular hollow cross-section (RHS). The algorithm randomly searched for suitable solutions for arranging different types of connections at various locations of the structure. As a result of the optimization, the number of connections was reduced in all cases. There are different proper answers according to the design of the connection conditions. Verify the results by re-analyzing the structure with each connection pattern, found that the structure analysis results meet the specification for strength constraints and serviceability constraints.

To reduce the number of connections in the structure, different conditions can be set according to the design requirements. The CLPSO algorithm is the method that can provide an optimal minimum connection placement for assembling modular houses within the design criteria of AISC-LRFD (2016) requirements.

Acknowledgement

References

[1] Lacey, A.W., et al., Structural response of modular buildings–an overview. Journal of building engineering, 2018. 16: p. 45-56.
[2] Gatheeshgar, P., et al. Development of affordable steel-framed modular buildings for emergency situations (Covid-19). in Structures. 2021. Elsevier.

[3] Lawson, R. and R. Ogden, Developments in pre-fabricated systems in light steel and modular construction. Transport, 2005. 35(15): p. 15.

[4] Kravets, O.J., et al., Development of algorithms for complex numerical optimization of objects with modular structure. International Journal on Information Technologies and Security, 2018. 10(4): p. 45-56.

[5] Liew, J., Y. Chua, and Z. Dai. Steel concrete composite systems for modular construction of high-rise buildings. in Structures. 2019. Elsevier.

[6] Chua, Y., J.R. Liew, and S. Pang, Modelling of connections and lateral behavior of high-rise modular steel buildings. Journal of Constructional Steel Research, 2020. 166: p. 105901.

[7] Lawson, R., et al., Pre-fabricated Systems in housing using light steel and modular Construction. Steel Structures, 2005. 5: p. 477-48.

[8] Generalova, E.M., V.P. Generalov, and A.A. Kuznetsova, Modular buildings in modern construction. Procedia engineering, 2016. 153: p. 167-172.

[9] Kamali, M. and K. Hewage, Life cycle performance of modular buildings: A critical review. Renewable and sustainable energy reviews, 2016. 62: p. 1171-1183.

[10] Baghdadi, A., Heristchian, M., & Kloft, H. (2021). Connections placement optimization approach toward new prefabricated building systems. Engineering Structures, 233, 111648.

[11] Nadeem, G., et al. Connection design in modular steel construction: A review. in Structures. 2021. Elsevier.

[12] Lacey, A.W., W. Chen, and H. Hao, Experimental methods for inter-module joints in modular building structures–A state-of-the-art review. Journal of Building Engineering, 2022. 46: p. 103792.

[13] Huang, V.L., P.N. Suganthan, and J.J. Liang, Comprehensive learning particle swarm optimizer for solving multiobjective optimization problems. International Journal of Intelligent Systems, 2006. 21(2): p. 209-226.

[14] Lawson, R. and R. Ogden, 'Hybrid'light steel panel and modular systems. Thin-Walled Structures, 2008. 46(7-9): p. 720-730.

[15] Kaveh, A. and A. Zolghadr, Comparison of nine meta-heuristic algorithms for optimal design of truss structures with frequency constraints. Advances in Engineering Software, 2014. 76: p. 9-30.

[16] Jafari, M., E. Salajegheh, and J. Salajegheh. Optimal design of truss structures using a hybrid method based on particle swarm optimizer and cultural algorithm. in Structures. 2021. Elsevier.

[17] Kaveh, A. and S. Talatahari, A hybrid CSS and PSO algorithm for optimal design of structures. Structural Engineering and Mechanics, 2012. 42(6): p. 783-797.

[18] Chen, W.-F. and S.-E. Kim, LRFD steel design using advanced analysis. Vol. 13. 1997: CRC press.

Analytical Solutions for Geometric Non-Linear Beam-Reinforced Thin Plates Using the Methodology of Groebner Bases

*†Y. Jane Liu[1], John Peddieson[2], Stephen Idem[2]

[1]Department of Civil and Environmental Engineering, **[2]**Department of Mechanical Engineering,
Tennessee Tech University, USA.

**Presenting author: jliu@tntech.edu
†Corresponding author: jliu@tntech.edu

Abstract

This paper illustrates the utility of the methodology of Groebner bases computations combined with the energy method in the analysis of geometric non-linear beam-reinforced thin rectangular isotropic plates (BRP) for modeling rectangular duct systems under internal pressure. The governing integro-partial differential equation is derived based on Kirchhoff/von Karman plate theory. With Rayleigh/Ritz methodology, a system of coupled polynomial algebraic equations is generated by using the exact solution of the BRP plate from linear analysis as a shape function. Then Groebner basis methodology is employed to decouple these equations. Under certain conditions, an analytical expression for the lateral displacements of the BRP plate under pressure can be obtained in a fully symbolic form, in terms of such parameters as geometric and material properties of the beams and panels. The analytical solutions have been compared with the results using the finite element software, ANSYS. The comparative study indicates that for commonly used duct panels with the aspect ratio (L:W) less than 1:4, the analytical solutions for displacements are in very close agreement. Finally, the study is found to be a unique alternative, worthy of further investigation, and potentially effective in the analysis of similar problems occurring in a variety of engineering applications.

Keywords: Groebner bases, geometrically non-linear analysis, beam-reinforce plates, rectangular duct system, computational algebraic geometry

Introduction

Groebner bases were introduced in 1965 by Bruno Buchberger, an Austrian mathematician [1], who included an algorithm to determine them. A more detailed mathematical background underlying the methodology can be found in books such as Cox et al. [2]. With the increasing capability of symbolic computation in recent decades, considerable progress has been made in the area of advanced computational algebraic geometry. Buchberger's algorithm for the determination of Groebner bases has been implemented in many mathematical symbolic computational systems such as the most popular commercial systems, e.g., Mathematica and Maple. Because of its availability, the use of the Groebner basis methodology has now become a feasible option for many scientific and engineering applications. Groebner basis methodology has been used to determine analytical solutions for non-linear problems in structural engineering [3]-[10], such as for a thin elastic composite plate with large deflection on the Pasternak foundation, calculation of the natural frequency of geometrically non-linear composite plates, evaluation of large deflections of a plate

with functional graded materials, and assessment of equilibrium equations for snapping through buckling of shallow arches with geometrical imperfection, etc.

This paper describes the utility of the methodology of Groebner bases in the analysis of geometric non-linear beam-reinforced thin plates (BRP) for modeling rectangular duct systems under internal pressure. The focus of this study is to illustrate how the method can be developed to solve non-linear engineering problems which are usually solved using numerical approaches. In the proposed procedure for the non-linear analysis, the governing integro-partial differential equation is derived based on Kirchhoff/von Karman plate theory, which is a functional of the total potential energy in terms of displacement functions for the system. Appling the variational principle to the functional of the total potential energy combined with the Ritz method, and using the exact solution of the BRP plate from linear analysis as a shape function, the governing equation can be transformed into a set of coupled non-linear algebraic equations. With the use of the Groebner basis package in Maple [11], analytical expressions for the lateral displacements of the BRP plate under internal pressure can be obtained in a fully symbolic form. These solutions can be expressed in terms of such parameters as geometric and material properties of the beams and plate. Thus, the displacement expression in terms of pressure, beam stiffness, and panel dimensions can be used as a convenient tool for preliminary design predictions for engineering practice. The analytical solutions have been compared with the results obtained from the finite element software ANSYS [12]. The comparison study indicates that for commonly used duct panels with the aspect ratio (L:W) less than 1:4, the analytical solutions for displacements are in very close agreement for three commonly used duct wall or panel thicknesses. Finally, the study is found to be a unique alternative, worthy of further investigation, and potentially effective in the analysis of similar problems occurring in a variety of engineering applications.

Motivation

Recently completed investigations [13][14] of the effectiveness of the SMACNA Industrial Duct Construction Standard (1st and 2nd Editions) [15][16] governing rectangular ductwork design indicates that: i) the concept of using an individual plate to predict the performance of the whole rectangular duct system can be traced back to 1970s, which has proven to be acceptable and cost-effective, and ii) one of the assumptions of the theoretical formulations for its plate models is to assume the stiffness of reinforcements to be infinite, which obviously is not consistent with the real duct behavior. In [17][18], a new beam-reinforced plate (BRP) finite element model including finite stiffness for the reinforcements and membrane effects was introduced and compared to duct models and plate models from existing industrial standards and literature [19]. Figure 1. (a) depicts deformation predicted by ANSYS for a 24-panel duct system under a uniform internal pressure. It demonstrates that as the number of the panels increased, the maximum deflection value for each top/bottom panel of the system approached a constant value throughout the entire duct system, which confirms that the concept of using an individual panel to predict the behavior of the duct system is a valid approximation. Moreover, the simulation also captured the deformation of the reinforcements (beams) in Figure 1.(b). Table 5.2 in [17] provides comparisons of plate models in their configurations, boundary conditions, and analysis types. The improvements and limitations of the introduced plate model were defined and discussed in [17] as well. These studies have motivated the current study of finding analytical solutions to the BRP plate model as a convenient

tool for preliminary design in engineering practice. The BRP plates will be defined in detail in next section.
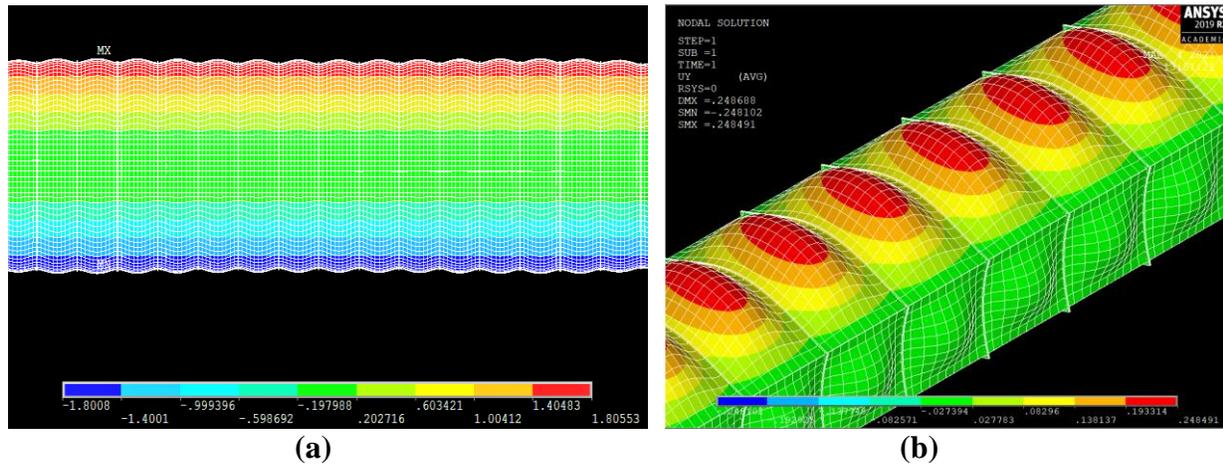
Figure 1. In finite model (a) deflection of 24-panel air duct, 100:1 scale, with p = 14 in. w.g. (b) beam deformation, 75:1 scale [17]

Problem Formulation

The primary objective of this section is to develop a general methodology for obtaining an analytical solution of non-linear deflection for beam-reinforced thin rectangular isotropic plates (BRP). The procedure of the methodology is as follows: i) generating the total strain energy of a BRP plate based on Kirchhoff/von Karman plate theory [20], ii) solving the exact linear solution of deflection for the BRP plate by Lévy's assumption [21], iii) forming the governing integro-partial differential equation which is a functional of the total potential energy in terms of displacement functions for the system, iv) applying the principle of minimum total potential energy with Rayleigh/Ritz method by using the exact linear solution of deflection as one of assumed shape functions for deflection to produce a system of coupled algebraic equations, and v) employing the methodology of Groebner bases to decouple the algebraic equations to obtain the analytical non-linear deflection solution of the BRP plate.

Strain Energy of BRP Plates

Figure 2.(a) defines a beam-reinforced thin rectangular plate (BRP) with two opposite edges simply supported (SS) and the other two edges supported by beams elastically; in Figure 2.(b), the plate coordinate system and dimension a and b are defined. It is assumed that the plate is composed of a homogeneous, isotropic material with a uniform thickness t, while the elastic beams on each side can have different material and geometrical properties. It is further supposed that the plate is subjected to a distributed pressure $q(x, y)$. According to the geometrically non-linear Kirchhoff/von Kármán plate theory, the total strain energy U can be expressed in terms of displacement functions u, v, and w as a sum of the plate and beam bending strain energy U_b and the membrane strain energy U_m as follows:

$$U_b = \frac{Et^3}{24(1-v^2)} \int_A \left\{ \left(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} \right)^2 - 2(1-v) \frac{\partial^2 w}{\partial x^2} \frac{\partial^2 w}{\partial y^2} - \left(\frac{\partial^2 w}{\partial x \partial y} \right)^2 \right\} dA$$
$$+ \frac{(E_1 I_1)_b}{2} \int_L \left(\frac{\partial^2 w}{\partial x^2} \right)^2 dA + \frac{(E_2 I_2)_b}{2} \int_L \left(\frac{\partial^2 w}{\partial x^2} \right)^2 dA \tag{1}$$

and:

$$U_m = \frac{Et}{2(1-v^2)} \int_A \left\{ \left(\frac{\partial u}{\partial x} \right)^2 + \frac{\partial u}{\partial x} \left(\frac{\partial w}{\partial x} \right)^2 + \left(\frac{\partial v}{\partial y} \right)^2 + \frac{\partial v}{\partial y} \left(\frac{\partial w}{\partial y} \right)^2 + \frac{1}{4} \left[\left(\frac{\partial w}{\partial x} \right)^2 + \left(\frac{\partial w}{\partial y} \right)^2 \right]^2 \right.$$
$$+ 2v \left[\frac{\partial u}{\partial x} \frac{\partial v}{\partial y} + \frac{1}{2} \frac{\partial v}{\partial y} \left(\frac{\partial w}{\partial x} \right)^2 + \frac{1}{2} \frac{\partial u}{\partial x} \left(\frac{\partial w}{\partial y} \right)^2 \right]$$
$$+ \frac{1-v}{2} \left[\left(\frac{\partial u}{\partial y} \right)^2 + 2 \frac{\partial u}{\partial y} \frac{\partial v}{\partial x} + \left(\frac{\partial v}{\partial x} \right)^2 + 2 \frac{\partial u}{\partial y} \frac{\partial w}{\partial x} \frac{\partial w}{\partial y} + 2 \frac{\partial v}{\partial x} \frac{\partial w}{\partial x} \frac{\partial w}{\partial y} \right] \right\} dA \tag{2}$$

where A is the area of the plate, L denotes the length of the elastic beams. Furthermore, $(E_1 I_1)_b$ and $(E_2 I_2)_b$ are the bending rigidities of the elastic beams, E indicates the modulus of elasticity and, v represents the Poisson's ratio of the plate. In addition, u and v are the in-plane displacement functions, and w is the lateral displacement function of the plate.
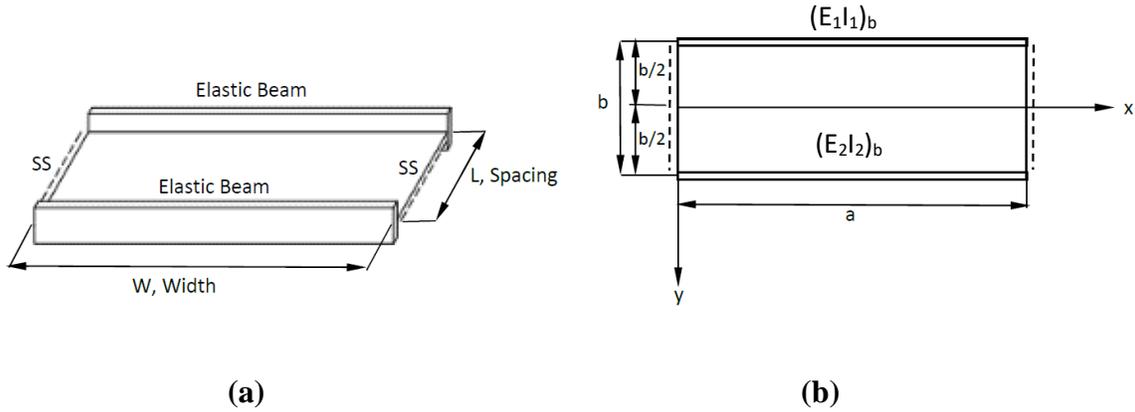
Figure 2. (a) A BRP plate configuration; (b) plate in Cartesian coordinates

Lévy's Solution of BRP Plates

For a rectangular plate with two opposite edges simply supported as shown in Figure 2. (a), the linear solution for deflection of the plate proposed by M. Lévy [22] is a semi-inverse solution in the form of a series:

$$w(x, y) = \sum_{m=1}^{\infty} y_m(y) \sin\left(\frac{m\pi x}{a}\right) \tag{3}$$

where y_m are arbitrary functions of y, and m are integers. It is obvious that the solution in equation (3) satisfies the plate governing differential equation (4) and the boundary conditions $w|_{x=0,x=a} = 0$ and $\frac{\partial^2 w}{\partial x^2}|_{x=0,x=a} = 0$ at the simply supported (SS) edges. The functions y_m can be determined by satisfying equation (4) and the boundary conditions on the edges $y = \frac{-b}{2}$ and $y = \frac{b}{2}$:

$$D\left(\frac{\partial^4 w(x,y)}{\partial x^4} + 2\frac{\partial^4 w(x,y)}{\partial x^2 \partial y^2} + \frac{\partial^4 w(x,y)}{\partial y^4}\right) = q(x,y) \tag{4}$$

where $D = \frac{Et^3}{12(1-v^2)}$ is called the flexural rigidity of the plate with a dimension of [force][length].

By substituting Equation (3) into Equation (4) and expanding the loading function $q(x,y)$ as a Fourier series, an ordinary differential equation can be found in the form of equation (5) for any value of m.

$$\frac{d^4 y_m}{dy^4} - 2\left(\frac{m\pi}{a}\right)^2 \frac{d^2 y_m}{dy^2} + \left(\frac{m\pi}{a}\right)^4 y_m = \frac{2}{aD}\int_0^a q(x,y)\sin\left(\frac{m\pi x}{a}\right)dx \tag{5}$$

The solution to the differential equation in equation (5) is shown as

$$\begin{aligned}y_m = A_m \cosh\left(\frac{m\pi y}{a}\right) + B_m\left(\frac{m\pi y}{a}\right)\sinh\left(\frac{m\pi y}{a}\right) + C_m \sinh\left(\frac{m\pi y}{a}\right) \\ + D_m\left(\frac{m\pi y}{a}\right)\cosh\left(\frac{m\pi y}{a}\right) + f_m(y)\end{aligned} \tag{6}$$

where $f_m(y)$ is the particular solution and A_m, B_m, C_m, and D_m are arbitrary constants. Substituting Equation (6) into Equation (3) yields the complete solution of deflection as:

$$\begin{aligned}w(x,y) = \sum_{m=1}^{\infty}[A_m \cosh\left(\frac{m\pi y}{a}\right) + B_m\left(\frac{m\pi y}{a}\right)\sinh\left(\frac{m\pi y}{a}\right) + C_m \sinh\left(\frac{m\pi y}{a}\right) \\ + D_m\left(\frac{m\pi y}{a}\right)\cosh\left(\frac{m\pi y}{a}\right) + f_m(y)]\sin(\frac{m\pi x}{a})\end{aligned} \tag{7}$$

The boundary conditions listed in equation (8) are the displacement boundary conditions based on the proposed BRP plate model specifically. The model assumes that the elastic beams attached to the plate along the edges $y = \frac{-b}{2}$ and $y = \frac{b}{2}$ are constrained from twisting and in-plane displacement. Equation (8) defines mathematically: i) the twisting constraint on the beams by setting the slope in the y-direction to zero; ii) the bending deflection of the elastic beams caused by the distributed force acting on the beams at the edges $y = \frac{-b}{2}$ and $y = \frac{b}{2}$ that is equal and opposite to the internal shear force acting on the plate. The four boundary conditions in (8) can be used to solve for the four arbitrary constants in (7).

$$\frac{\partial w}{\partial y}\Big|_{y=\pm\frac{b}{2}} = 0$$

$$D\left[\frac{\partial^3 w}{\partial y^3} + (2-v)\frac{\partial^3 w}{\partial x^2 \partial y}\right]\Big|_{y=\frac{b}{2}} = \left((E_1 I_1)_b \frac{\partial^4 w}{\partial x^4}\right)\Big|_{y=\frac{b}{2}}$$

$$-D\left[\frac{\partial^3 w}{\partial y^3} + (2-v)\frac{\partial^3 w}{\partial x^2 \partial y}\right]\Big|_{y=-\frac{b}{2}} = \left((E_2 I_2)_b \frac{\partial^4 w}{\partial x^4}\right)\Big|_{y=-\frac{b}{2}}$$

(8)

In symmetric cases, i.e., where the plate's geometric and material properties, boundary conditions, and loading are all symmetric with respect to the coordinate system, C_m and D_m have to be equal to zero, and $(E_1 I_1)_b = (E_2 I_2)_b$. Thus, the lateral deflection $w(x, y)$ in equation (7) can be reduced to equation (9), and the boundary conditions on the edges $y = \pm\frac{b}{2}$ in (8) can be expressed in terms of equation (10):

$$w(x, y) = \sum_{m=1}^{\infty}\left[A_m \cosh\left(\frac{m\pi y}{a}\right) + B_m\left(\frac{m\pi y}{a}\right)\sinh\left(\frac{m\pi y}{a}\right) + f_m(y)\right]\sin\left(\frac{m\pi x}{a}\right)$$

(9)

$$\frac{\partial w}{\partial y}\Big|_{y=\frac{b}{2}} = 0$$

$$D\left[\frac{\partial^3 w}{\partial y^3} + (2-v)\frac{\partial^3 w}{\partial x^2 \partial y}\right]\Big|_{y=\frac{b}{2}} = \left((E_1 I_1)_b \frac{\partial^4 w}{\partial x^4}\right)\Big|_{y=\frac{b}{2}}$$

(10)

When the BRP plate is subjected to a uniformly distributed load $q(x, y) = q_o$, the particular solution $f_m(y)$ can be calculated as in (11):

$$f_m(y) = \sum_{m=1}^{\infty}\frac{2q_o a^4}{Dm^5\pi^5}(1 - \cos(m\pi)) \ \ or$$

$$f_m(y) = \sum_{m=1,3,5\ldots}^{\infty}\frac{4q_o a^4}{Dm^5\pi^5}$$

(11)

With the symmetric boundary conditions in equation (10), the constants A_m and B_m in equation (9) can be determined. The complete analytical linear solution of deflection for a Lévy plate with reinforced elastic beams is shown in equation (12), which will be used for the non-linear analysis in this study.

$$w(x, y) = \sum_{m=1}^{\infty}\frac{1}{D}\left[q_o a^4\left(\frac{4}{m^5\pi^5} + A_m \cosh\left(\frac{m\pi y}{a}\right)\right.\right.$$

$$\left.\left. + B_m\left(\frac{m\pi y}{a}\right)\sinh\left(\frac{m\pi y}{a}\right)\right)\sin\left(\frac{m\pi x}{a}\right)\right]$$

(12)

where:

$$A_m = -\frac{4\lambda\left(\frac{1}{2}\pi bm\cosh(\alpha_m)+a\sinh(\alpha_m)\right)}{\pi^4 m^4(\pi a\lambda m\cosh(\alpha_m)\sinh(\alpha_m)+\frac{1}{2}\pi^2 m^2 b\lambda+2a\cosh(\alpha_m)^2-2a)} \tag{13a}$$

$$B_m = \frac{8a\lambda\sinh(\alpha_m)}{\pi^4 m^4(2\pi a\lambda m\cosh(\alpha_m)\sinh(\alpha_m)+\pi^2 m^2 b\lambda+4a\cosh(\alpha_m)^2-4a)} \tag{13b}$$

$$\lambda = \frac{(E_1 I_1)_b}{aD}, \quad \alpha_m = \frac{m\pi b}{2a} \tag{13c}$$

The Principle of Minimum Total Potential Energy

The total work done W_q performed by the distributed load acting on the plate can be calculated as:

$$W_q = \int_A q(x,y)w(x,y)dA \tag{14}$$

Thus, the total potential energy Π of the system can be expressed as:

$$\Pi = U_b + U_m - W_q \tag{15}$$

The shape functions chosen for the in-plane displacements, u_s and v_s, respectively, and the lateral displacement w_s, are the functions that satisfy all the displacement boundary conditions for the BRP plate with symmetric conditions. For simplicity, the displacement functions can be assumed to have the form of (16a), (16b), and (16c).

$$u = u_s A_0$$

$$u_s = \sin\left(\frac{2\pi x}{a}\right)\sin\left(\frac{\pi(y-\frac{b}{2})}{b}\right) \tag{16a}$$

$$v = v_s B_0$$

$$v_s = \sin\left(\frac{\pi x}{a}\right)\sin\left(\frac{2\pi y}{b}\right) \tag{16b}$$

$$w = w_s C_0$$

$$w_s = \sum_{m=1}^{\infty}\frac{1}{D}\left[q_o a^4\left(\frac{4}{m^5\pi^5}+A_m\cosh\left(\frac{m\pi y}{a}\right)\right.\right. \tag{16c}$$
$$\left.\left.+B_m\left(\frac{m\pi y}{a}\right)\sinh\left(\frac{m\pi y}{a}\right)\right)\sin\left(\frac{m\pi x}{a}\right)\right]$$

The shape function w_s is the linear solution of the plate in equation (12), and A_0, B_0, and C_0 are unknown coefficients to be computed. Also, note that choosing only three unknowns A_0, B_0, and

C_0 in the assumed displacement functions, besides offering simplicity, has been proven to be sufficient to maintain the accuracy of the deflections for the cases investigated in the current study. Substituting the assumed displacement functions u, v, and w in (16a), (16b), and (16c) into the total potential energy equation (15), and applying the principle of minimum potential energy or extrema of the functional, implies that the first variation of the functional is equal to zero in (17). This yields a system of three coupled non-linear algebraic equations in terms of the unknowns A_0, B_0, and C_0 that can be generated by imposing the conditions in (18) for the case of the symmetric BRP plates:

$$\delta\Pi = 0 \tag{17}$$

$$\frac{\partial\Pi}{\partial A_0} = 0 \ , \frac{\partial\Pi}{\partial B_0} = 0 \ , \text{and} \ \frac{\partial\Pi}{\partial C_0} = 0 \tag{18}$$

The Groebner basis package in Maple 2020 can then be employed for computing the reduced Groebner bases, which comprise a set of decoupled algebraic equations based on the set of coupled non-linear algebraic equations generated from (18). Since the determination of the lateral displacement w is the focus the present investigation, the decoupled Groebner basis equations can be manipulated to have a basis equation that contains only one unknown coefficient C$_0$ which can be solved directly. Then, all other unknown coefficients can be determined by substituting C$_0$ into the other basis equations. With all constants determined, the analytical solutions for all the displacements are solved, and detailed parametric studies can be performed.

Results and Discussion

Symbolic solutions have been obtained for an isotropic rectangular BRP plate using the methodology presented in the previous section. The lateral displacement function $w(x, y)$ is expressed in terms of the constant coefficient C$_0$ with the assumed shape function. C$_0$ is a function of the plate properties E, v, t, a, and b, the properties of the elastic beams E_1 and I_1, and the lateral pressure $q(x, y) = q_o$. The fully symbolic solution is too long in length to include in this paper. However, a complete set of calculations for C$_0$ expressed as a function of pressure for a given BRP plate is presented as an example in Appendix A.

For demonstration purposes, and to show the effectiveness and accuracy of the calculated deflections for the BRP plates derived from symbolic solutions, the deflections w versus pressure load q_0 are presented below for rectangular plates having 10, 16, 22-gauge duct wall thicknesses and three types of beam reinforcements. In every instance E $= 29,500 \ ksi, v = 0.3$. The pressure loads q_0 are systematically varied from 1 to 36 inch w.g. for all the cases. Figures 3(a), (b) present a comparison of the lateral displacements obtained using linear and non-linear analysis of plates having dimensions of 30 x 30 (in. $\times$ in.) and 15 x 30 (in. $\times$ in.), respectively. The non-linear solutions are carried out using 1-term, 3-term, and 5-term series. The resulting solutions show that the 1-term solution is sufficient to predict the non-linear deflection of the plates. It can likewise be

observed that significant differences occur between the linear and non-linear solutions as the pressure load increases.

(a)

(b)
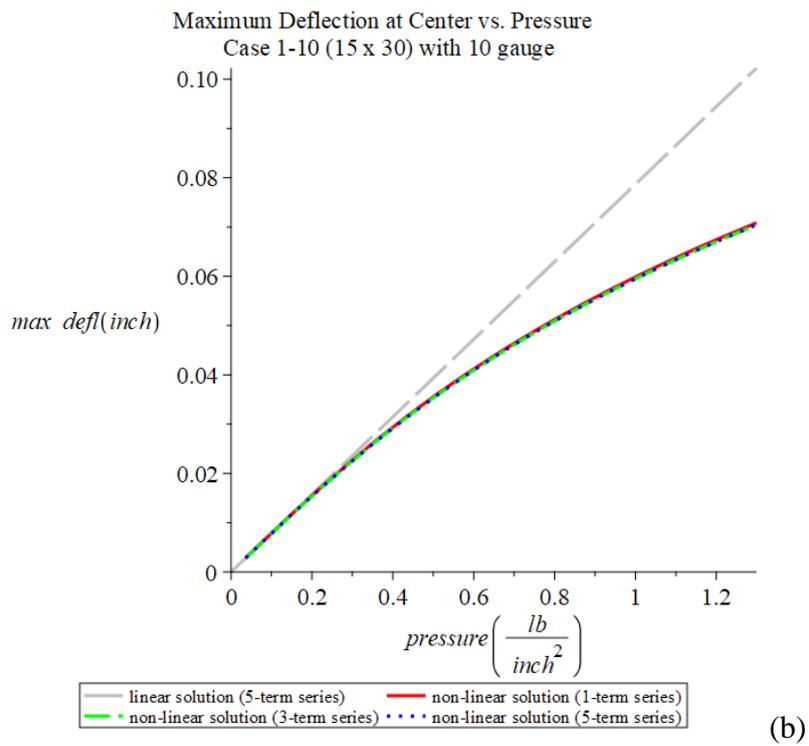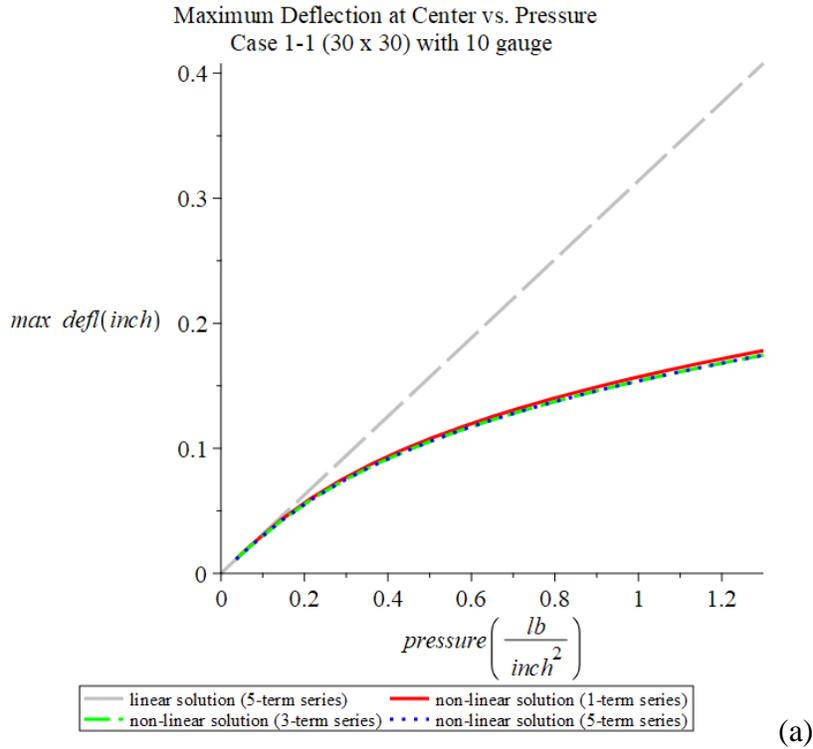
Figure 3. Comparison of linear and non-linear behaviors of the BRP plates for 10-gauge duct wall thickness with various dimensions (W x L)

Figures 4.(a)-(d) show a comparison of the lateral displacements calculated using non-linear analytical and numerical analyses of 10-gauge plates with various aspect ratios. The non-linear analytical solutions with 1-term, 3-term, and 5-term series exhibit close agreement with the numerical results calculated by ANSYS. The percent errors for deflection are in the range of 0.12% to 3.5%. Figures 5 and 6 demonstrate similar conclusions for the 16-gauge and 22-gauge plates. The maximum percent error for the lateral displacement is less than 5.6%, which occurred in the 22-gauge plates.

(a)

(b)

Maximum Deflection at Center vs. Pressure
Case 1-5 (150 x 30) with 10 gauge

max defl(inch)

pressure $\left(\dfrac{lb}{inch^2}\right)$

— non-linear solution (1-term series) — non-linear solution (3-term series)
···· non-linear solution (5-term series) ◇ ANSYS results

(c)

Maximum Deflection at Center vs. Pressure
Case 1-11 (22.5 x 30) with 10 gauge

max defl(inch)

pressure $\left(\dfrac{lb}{inch^2}\right)$

— non-linear solution (1-term series) — non-linear solution (3-term series)
···· non-linear solution (5-term series) ◇ ANSYS results

(d)

**Figure 4. Comparison of non-linear analytical and ANSYS solutions of the BRP plates
for 10-gauge with various dimensions (W x L)**
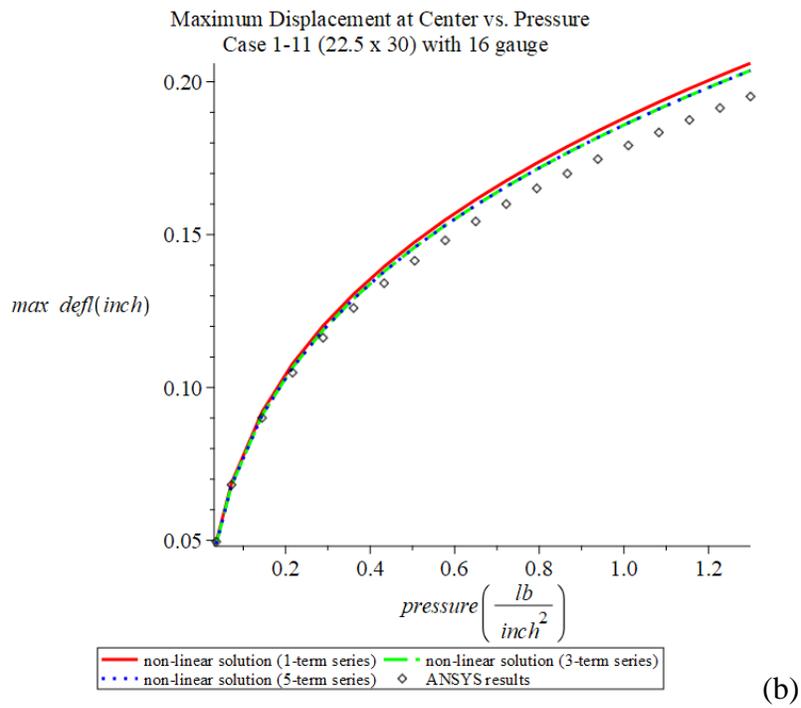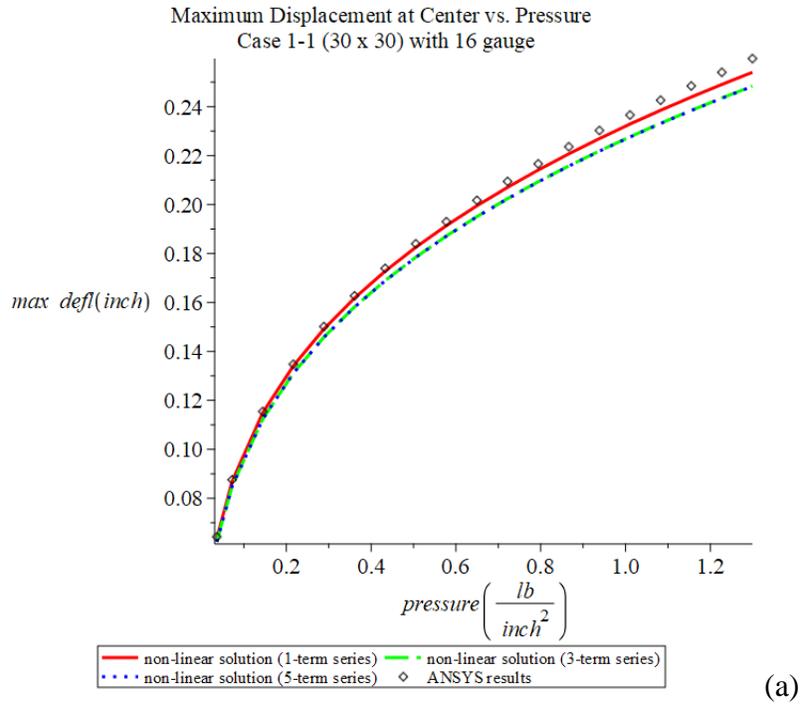
Figure 5. Comparison of non-linear analytical and ANSYS solutions of the BRP plates for 16-gauge with various dimensions (W x L)

(a)

(b)

Figure 6. Comparison of analytical non-linear and ANSYS solutions of the BRP plates for 22-gauge with various dimensions (W x L)

Conclusions and Recommendations

Conclusions

A general procedure of using Groebner basis methodology combined with the energy method has been developed for non-linear analysis of thin rectangular beam-reinforced plates (BRP) with opposite edges simply supported and the other two edges supported by elastic beams. The procedure is fully imp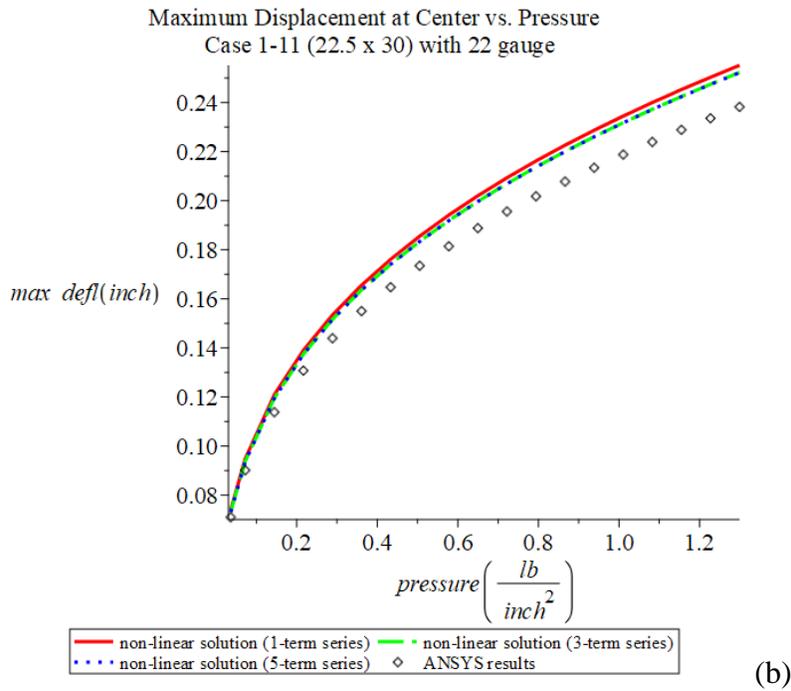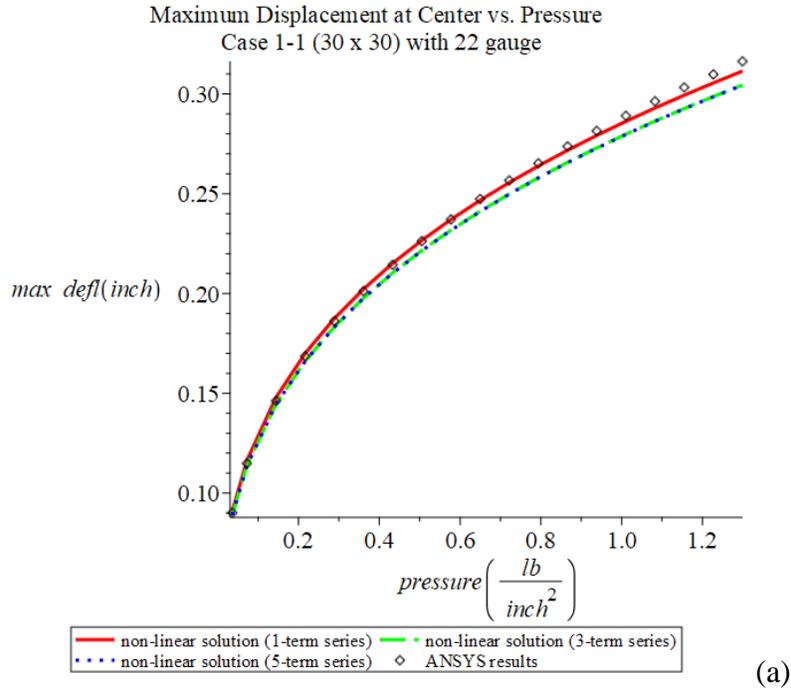lemented in Maple 2020. An analytical expression was obtained for the large transverse deflection of the plate derived by the proposed methodology. Parametric studies were carried out for the cases with varying geometrical properties of the plate and beams. The predicted maximum lateral deflections versus distributed pressure loads obtained with the analytical expression were compared with numerical results generated by using ANSYS. The results were found to be in very good agreement for non-linear analysis, with percent errors ranging from 0.12% to 5.6% for the cases examined. It should be noted that the lateral displacement function given in equation (16c) contained only one parameter, i.e., C_0. All unknown coefficients in the assumed in-plane and out-of-plane displacement functions can be represented fully symbolically in terms of E, v, t, a, b, I_b, and E_b. Since these fully symbolic expressions for u, v, and w can be prohibitively lengthy, to illustrate basic concepts the example shown in Appendix A provides an analytical expression of plate deflection in terms of pressure load for one particular case. It also shows the advantage of the analytical solutions as a convenient design tool to obtain preliminary deformation predictions effectively. Moreover, the results obtained from the current study indicated that the computational method of Groebner bases provides a unique alternative, worthy of further investigation, and is potentially effective for analyzing similar problems occurring in various engineering applications.

Recommendations

It is recommended that the following work be carried out: i) the procedure developed in this study can be extended to various loading cases including temperature or moisture effects, ii) BRP plates supported by different elastic foundations should be considered, and iii) it would be beneficial to add elastic torsional boundary conditions to the current BRP plate model along the simply supported edges to account for the effects of vertical panels to improve the prediction for the performance of whole duct systems.

Appendix A.

Presented below is the coefficient C_0, the maximum displacement at the center of the BRP plates as defined in equation (16c), with material and geometrical properties of the given BRP plate as follows:

$$E = 29,500 \ ksi, v = 0.3, \tag{Aa}$$

$$a = 30 \ in., b = 30 \ in., t = 0.1265 \ in., I = 0.1374 \ in.^4 \tag{Ab}$$

The calculated C_0 below is for non-linear analysis as defined in (Aa) and (Ab). As shown, C_0 is an analytical formula of uniformly distributed pressure load q_0:

C_0 = 6.277237608*10^(-103)*(1.342763994*10^282*sqrt(1.207055258*10^47*q0^2 + 4.126884867*10^45) + 4.665124767*10^305*q0)^(1/3)/q0 - 1.225488228*10^101/(q0*(1.342763994*10^282*sqrt(1.207055258*10^47*q0^2 + 4.126884867*10^45) + 4.665124767*10^305*q0)^(1/3))

Figure A1(a) demonstrates the deflection from analytical solution of the plate at a given load q_0 = 10 w.g. in three dimensional, and Figure A1(b) shows the deflection as a function of x along the center line of the plate at y = 0 using a 1-term series and a 5-term series. It can be seen that 1-term and 5-term solutions are fairly close.

(a)   (b)
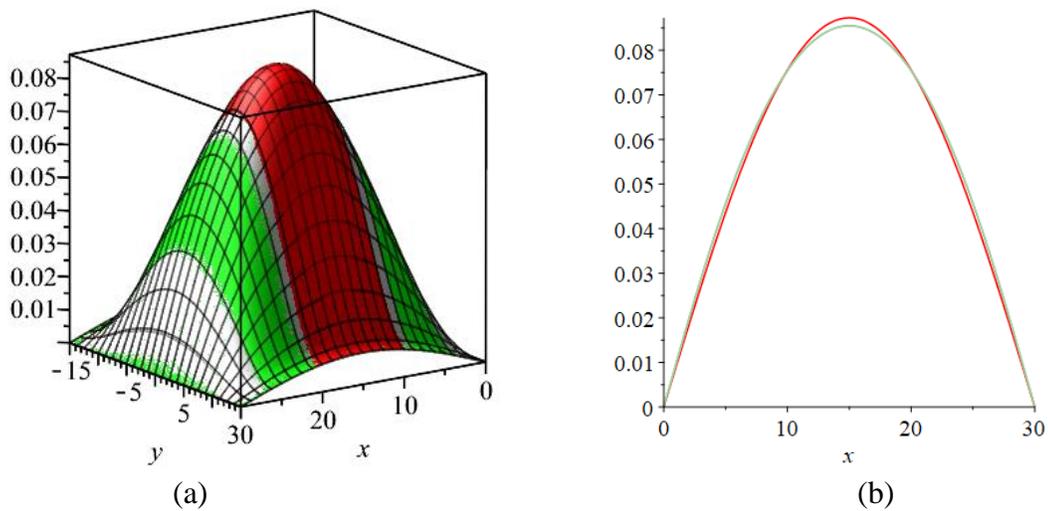
**Figure A1 The deformation of the BRP plate (30 x 30) under q_0 = 10 w.g.
(a) in 3-D; (b) along the center line y = 0 with 1-term and 5-terms series**

Figure A2 shows comparisons among analytical linear and non-linear, and Ansys solutions for the maximum deflections in terms of pressure load q_0 from 1 to 36 inch w.g. The analytical solution in (16c) is a solution in series, therefore, 1-term, 3-term, and 5-term analytical non-linear solutions are also presented in Figure A2 for the specified BRP plate. It is observed that the percent error for the analytical solution compared to ANSYS results is within a range from 0.12% to 2.10%.
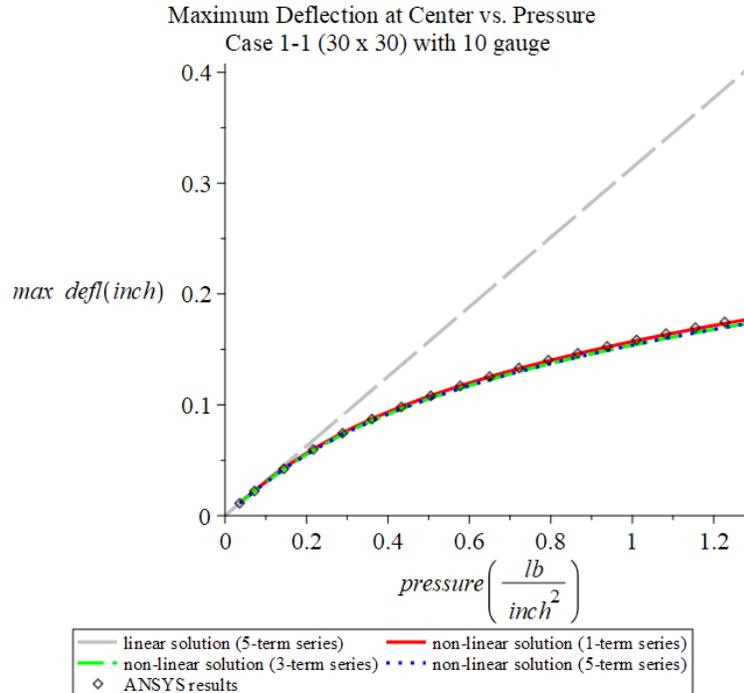
Figure A2. Comparisons among linear, non-linear analytical, and ANSYS solutions

As an example, the data files and the calculated percent error are included as follows:

1. The maximum deflections calculated from the formula (1-term series) in (16c) for the pressures q_0 ranging from 1 to 36 in. w.g.

Wmax:=[0.0115, 0.0227, 0.0429, 0.0602, 0.0749, 0.0875, 0.0985, 0.1084, 0.1172, 0.1253, 0.1328, 0.1397, 0.1461, 0.1522, 0.1579, 0.1634, 0.1685, 0.1735, 0.1782]

2. The center deflections computed by ANSYS for the pressures q_0 ranging from 1 to 36 in. w.g.

FE data:= [0.0113, 0.0222, 0.0422, 0.0597, 0.0744, 0.0872, 0.0981, 0.1081, 0.1171, 0.1255, 0.1332, 0.1403, 0.1465, 0.1527, 0.1586, 0.1642, 0.1696, 0.1747, 0.1797]

3. The percent error for the center deflection for the pressure q_0 from 1 to 36 in. w.g.

Error%:= [2.10, 1.99, 1.67, 0.92, 0.71, 0.37, 0.42, 0.27, 0.12, -0.14, -0.32, -0.41, -0.22, -0.32, -0.42, -0.51, -0.62, -0.72, -0.83]

References

[1] B. Buchberger, *An algorithm for finding a basis for the residue class ring of a zero-dimensional ideal*, Ph.D. thesis, University of Innsbruck, Innsbruck, Austria, 1965.

[2] Cox, D., Little, J. and O'Shea, D. (2007) *Ideals,Varieties, and Algorithms* 3rd ed., Springer Science+Business Media, New York, NY.

[3] Liu, J. and Peddieson, J. Evaluation of Groebner basis methodology as an aid to harmonic balance, *Journal of Vibration and Acoustics*, vol. 136, 2014.

[4] Liu, Y. J. and Peddieson, J. Application of Groebner bases to non-linear mechanics problems, ICMS, vol. 8592, pp. 398-405, Springer, 2014.

[5] Liu, Y. J., Buchanan, G. R. and J. Peddieson, Application of Groebner basis methodology to non-linear static cable analysis, *Journal of Offshore Mechanics and Arctic Engineering*, vol. 135, 2013.

[6] VanDervort, R. *Geometrically non-linear Analysis of Rectangular Orthotropic Plates Using Groebner Bases*, M.S. Thesis, Tennessee Tech University, 2009.

[7] Gaw, D. *Geometrically non-linear Analysis of Isotropic and Laminated Composite Plates Subjected to Thermal Loading Using Groebner Bases*, M.S. Thesis, Tennessee Tech University, 2013.

[8] Harrell, T. M. *Application of Groebner bases to geometrically non-linear analysis of axisymmetric circular isotropic plates*, M.S. thesis, Tennessee Technological University, 2014.

[9] Perry C. G. *Application of Groebner bases to geometrically non-linear analysis of rectangular composite plates resting on a Pasternak Foundation*, M.S. thesis, Tennessee Technological University, 2018.

[10] Perry C. G. and Liu Y. J. Geometrically non-linear Analysis of Thin Rectangular Plates on a Pasternak Foundation Using Groebner Bases, *the 6th Annual International Conference on Architecture and Civil Engineering (ACE)*, May 14-15, 2018 in Singapore, GSTF.

[11] Maplesoft, *Maple 2020, Maple 2020 Help*.

[12] Ansys® Academic Research Mechanical, Release 22.1, Help System, Theory Reference Manual, ANSYS, Inc.

[13] Liu, J. and Idem, S. (2019) *Understanding the Expected Deformation of Rectangular Ductwork*, Final Report: DMI.

[14] Liu, J. and Idem, S. (2020) *Final Report-Development of Unified Duct Design Equations and Improvements to the Current FEA Models*. DMI Technical Report.

[15] *Rectangular Industrial Duct Construction Standards*. 1980, 1st Edition. Chantilly, VA: SMACNA.

[16] *Rectangular Industrial Duct Construction Standards*. 2004, 2nd Edition. Chantilly, VA: SMACNA.

[17] Crispi, M. *Numerical non-linear Analysis of Beam-Reinforced Thin Plates for Modeling of Duct System*. M.S. Thesis, Tennessee Tech University, 2021.

[18] Crispi, M., Liu, J., Peddieso, J., Idem, S. (2022). Non-linear analysis of beam-reinforced thin plates for modeling rectanguar duc systems, IMECE2022-90109, *Proceedings of ASME IMECE, IMECE 2022 Conference*, Oct. 30 to Nov. 3, 2022, Columbus, OH.

[19] Thanga, T., Sivakumaran, K.S., Halabieh, B., Strength of plates of rectangular industrial ducts. *Procedia Engineering*. vol. 14, pp. 622-629, 2011.

[20] von Kármán, T. Festigkeitsprobleme in Maschinenbau, *Encyklopadie der Mathematischen Wissenschaften,* vol. 4, pp. 311-385, 1910.

[21] S. Timoshenko and S. Woinowsky-Krieger (1959) *Theory of Plates and Shells.* McGraw-Hill Book Company, Inc., New York, Toronto, London,

[22] Lévy, M. Sur L'equilibrie d'une Plaque Rectangulaire. *Comptes Rendues*, vol. 129, pp. 535-539, 1899.

Application of PDS-FEM to simulate high-power LASER induced cracking

*†Maddegedara Lalith[1], Muhammad Naveed Akram[2], Mahendra Kumar Pal[3], Elia Nicolin[2], Yosuke Kawahito[4], Toshihiro Kameda[5] and Muneo Hori[4]

[1]Earthquake Research Inst., Univ. of Tokyo, Japan
[2]Dept. of Civil Eng., The Univ. of Tokyo, Japan
[3]IIT BHU Varanasi, Uttar Pradesh, India
[4]Japan Agency for Marine-Earth Science and Technology, Yokohama, Japan
[5]Faculty of Eng., Univ. of Tsukuba, Ibaraki, Japan
*Presenting author: lalith@eri.u-tokyo.ac.jp
†Corresponding author: lalith@eri.u-tokyo.ac.jp

Abstract

We extended PDS-FEM for simulating thermal conduction, thermal induced mechanical deformation and cracking of brittle elastic materials with the long-term objective of simulating high-power LASER induced cracking in large concrete bodies. Taking a variational approach, we derived the governing matrix equations for simulating the above phenomena using higher-order PDS-FEM (HO-PDS-FEM). Due to the involvement of extreme temperatures in the target long-term application, both the radiative and convective boundary conditions are taken into account. Implementation of static and transient thermal conduction are verified comparing with analytical solutions, and thermal induced cracking is validated by comparing with experimental observations of quenching induced cracking in a thin plate. As a preliminary assessment, cutting of concrete block using a high-power LASER is simulated using a high-resolution mesh, and the resulting crack patterns on surfaces are qualitatively compared. The comparison produced a satisfactory agreement indicating the developed numerical model can be used study cracking induced by concentrated high energy sources like LASERS.

Keywords: thermal induced cracking, higher order particle discretization scheme, LASER cutting, verification and validation

Introduction

Advancement of technologies in the demolishing industry is necessary to address the problems like lack of environmentally safe technologies to demolish large structures such as troubled nuclear power plants, high-rises in congested commercial districts of modern cities[1, 2]. Demolishing massive containment buildings of troubled nuclear power plants using conventional technologies can release a large amount of radioactive dust to the air causing environmental problems and further deteriorating public trust in nuclear power. Also, the demolishing industry requires new technologies usable in limited spaces to remove high-rise buildings in congested commercial centers of metropolis. High-power LASERs are promising ultra-low-level dust and sound producing alternative to progressively cut and remove thick concrete walls of NPPs and reinforced concrete (RC) structures in congested cities[1]. High-power LASERS have already been considered promising method for the removal of the skin of concrete components exposed to radioactive materials, when demolishing NPPs [2]. However, the removal of 1∼2 mm top layer of concrete surfaces can produce a large amount of dust causing environmental concerns. A much lesser radioactive dust producing and convenient approach is to cut the structure into large chunks using high-power LASERs, and to transport them to a safe location for safe recycling[1, 3]. Also, cutting into large chunks is a very convenient approach to demolish

high-rises in congested cities. Ability to transmit long distance via fiber optics provides high flexibility, which is very attractive when working in congested cities and harmful environments. Extensive studies have to be conducted to access the viability and safety of using high-power LASERs to demolish large structures. While extensive experiments have to be conducted to explore how to use LASERs to cut several meters thick concrete bodies, the objective of this paper is to develop a numerical model to simulate extensive cracks in concrete induced by the concentrated energy from high-power LASERs.

Research on thermal shock induced failure of brittle materials has been of interest to the researchers for many decades for a range of problems like quenching of ceramics to the fire safety of concrete structure. Geyer *et al.*[4] have done an experimental study on thermally induced parallel-edge cracks due to contraction of boundary layers during the cooling of a brittle material. Bourdin [5, 6] has studied the variational approach to model fracture and extended the study to model crack propagation during glass quenching. Jiang *et al.*[7] have performed a series of experimental and numerical studies on cracking pattern in ceramics subjected to thermal shocks. Several researchers have reproduced Jiang *et al.*'s experiment[8, 9, 10]. Chu *et al.* [11] have proposed an evolution equation using elastic energy density functions and studied the dynamic crack propagation in brittle materials using a thermo-mechanical coupled phase field model. The primary objective of this study has been to enhance the thermal stress analysis techniques conventionally used to model cracking phenomena, which are not efficient in modelling cracking of bulk 3D bodies. Recent advancements in the classical finite element method such as enriched/extended FEM [12, 13, 14] have made it possible to accurately evaluate the crack-tip stress field. However, the techniques introduced by these advancements are substantially tedious and introduce a substantial computation cost.

Since our long-term objective of this study is simulating high-power LASER induced cracking in large concrete bodies, as the base numerical method we choose Particle Discretization Scheme FEM (PDS-FEM) which provides numerically efficient treatments for modeling crack propagation in large scale models. PDS-FEM[15, 16, 18, 19] is designed to exploit FEM's accuracy in modeling continuum and particle based method's efficiency in modeling discontinuities. A unique feature of PDS is the use of conjugate domain tessellations in the FEM framework; one tessellation (e.g., Voronoi) is used for approximating functions, and the other tessellation (e.g., Delaunay) for approximating the corresponding derivatives. Within a tessellation element, corresponding variables (i.e., function or derivatives) are approximated as a linear combinations of a suitable base functions with compact support in the tessellation element. As explained in the next section, this introduces numerous discontinuities which are exploited by PDS-FEM to numerically efficiently model propagating cracks. PDS-FEM has been applied to simulate various crack propagation phenomena including quasi-static, dynamic, high strain rates, frictional interfaces and material non-linearity [17, 18, 20, 21].

We present the implementation of HO-PDS-FEM to analyze heat conduction, thermal induced deformations and cracking in brittle elastic solids with their verification and validation. Section 2 briefly outlines the mathematical background of HO-PDS and its implementation into the FEM framework to simulate heat conduction, thermal induced mechanical deformations and cracking. The verification of static and transient heat conduction, and the validation of the thermal induced cracking are presented in the section 3. For the validation, we numerically reproduced a quenching experiment by Jiang *et al.*[7] and compared numerically obtained crack patterns with those from the experiment. Section 4 compares numerically and experimentally obtained crack patterns in a 50cm×10cm×10cm concrete bar as a preliminary evaluation of the developed numerical model. The article is finally wrapped up with a summary in section 5.

Mathematical Background

Higher Order PDS

As mentioned in the introduction, PDS uses conjugate tessellation pair Voronoi and Delaunay to approximate a function and its derivatives. HO-PDS-FEM approximates the corresponding variable (i.e., functions or derivatives) within a given tessellation element as a linear combination of suitable base functions which have compact support in their corresponding tessellation elements. As an example, let $\{\boldsymbol{x}^\alpha\}$ and $\{\boldsymbol{x}^\beta\}$ be mother points of a set of Voronoi and Delaunay tessellation elements $\{\Phi^\alpha\}$ and $\left\{\Psi^\beta\right\}$, respectively. Then, a function $f(\boldsymbol{x})$ and its derivatives $f_{,i} = \frac{\partial f}{\partial x_i}$ can be approximated as

$$
\begin{aligned}
f(\boldsymbol{x}) &\approx f^d(\boldsymbol{x}) = \sum_{\alpha,n} f^{\alpha n} P^{\alpha n} \quad &(1) \\
\nabla f(\boldsymbol{x}) &\approx g^d(\boldsymbol{x}) = \sum_{\beta,m} g^{\beta m} Q^{\beta m}, \quad &(2)
\end{aligned}
$$

respectively. Here, $\{P^{\alpha n}\} = \{1, .., (\boldsymbol{x} - \boldsymbol{x}^\alpha)^{n-1}\}\phi^\alpha(\boldsymbol{x})$, $\{Q^{\beta m}\} = \left\{1, .., (\boldsymbol{x} - \boldsymbol{x}^\beta)^{m-1}\right\}\psi^\beta(\boldsymbol{x})$. By minimizing the L^2-norm of the errors $E^f = \int_\Omega (f(\boldsymbol{x}) - f^d(\boldsymbol{x}))^2 \, \mathrm{d}v$, and $E^g = \int_\Omega (g(\boldsymbol{x}) - f_{,i}^d(\boldsymbol{x}))^2 \, \mathrm{d}v$, the set of coefficients $\{f^{\alpha n}\}$s' and $\{g^{\beta n}\}$s' can be determined as

$$
\begin{aligned}
f^{\alpha n} &= \sum_{n=0}^{|P^\alpha|} (I^{\alpha n n'})^{-1} \int_{\Phi^\alpha} P^{\alpha n'} f(\boldsymbol{x}) \mathrm{d}v \quad &(3) \\
g_i^{\beta m} &= \sum_{m=0}^{|Q^\beta|} w^{\beta m m'} \sum_{\alpha=1}^{N^\alpha} \sum_{n=1}^{|P^\alpha|} f^{\alpha n} \int_{\Psi^\beta} (P^{\alpha n})_{,i} Q^{\beta m} \mathrm{d}v \\
&= w^{\beta m m'} f^{\alpha n} h_i^{\beta \alpha m' n}, \quad &(4)
\end{aligned}
$$

where $I^{\alpha n n'} = \int_{\Phi^\alpha} P^{\alpha n} P^{\alpha n'} \mathrm{d}v$, $J^{\beta m m'} = \int_{\Psi^\beta} Q^{\beta m} Q^{\beta m'} \mathrm{d}v$, $w^{\beta m m'} = \left(J^{\beta m m'}\right)^{-1}$, and $h_i^{\beta \alpha m' n} = \int_{\Psi^\beta} Q^{\beta m'} P_{,i}^{\alpha n} \mathrm{d}v$. Note that, in the rest of the paper, we use the Einstein's summation notation with respect to repeated subscripts or superscripts.

Modeling heat conduction

Consider a domain Ω occupied by a thermally isotropic solid, subjected to suitable boundary conditions along the boundary $\partial\Omega$. Thermal conduction of this domain is expressed by the following strong form

$$
\rho c \dot{T} - \nabla . \boldsymbol{q} - H = 0, \quad (5)
$$

where $\boldsymbol{q}$ is the thermal flux vector, c is the specific heat of the material, $T(x_i, t)$ is the temperature at a point $x_i \in \Omega$, t is the time, and H is the heat generated per unit volume at x_i. According to the Fourier's model, the thermal flux vector for thermally isotropic material with thermal conductivity κ is $\boldsymbol{q} = \kappa \nabla \boldsymbol{T}$.

Our PDS-FEM formulation for heat conduction is based on the following classical weak form for heat conduction, where $g(\boldsymbol{x})$ is an arbitrary function satisfying $g(\bar{\boldsymbol{x}}) = 0$ at $\bar{\boldsymbol{x}} \in \partial\Omega$, and $\boldsymbol{n}$

is the unit outward normal to $\partial\Omega$.

$$
\begin{aligned}
W &= \int_{\Omega}\left(\rho c\frac{\mathrm{d}T}{\mathrm{d}t} + \nabla.\boldsymbol{q} - H\right) g(\boldsymbol{x})\mathrm{d}v \\
&= \int_{\Omega}\left(\rho c\dot{T} - H - \boldsymbol{q}.\boldsymbol{\nabla}g\right) g(\boldsymbol{x})\,\mathrm{d}v + \int_{\partial\Omega} \boldsymbol{n}.\boldsymbol{q}\, g(\boldsymbol{x})\,\mathrm{d}.s \quad (6)
\end{aligned}
$$

In the above, the last term, which represents boundary conditions, can be expressed it in terms of radiative and convective boundary conditions as

$$
\int_{\partial\Omega} \boldsymbol{n}.\boldsymbol{q}\, g(\boldsymbol{x})\,\mathrm{d}s = \int_{\partial\Omega_q} g\bar{q}\,\mathrm{d}s + \int_{\partial\Omega_c} gh_c\,(T - T_\infty)\,\mathrm{d}s + \int_{\partial\Omega_r} g\sigma\varepsilon\left(T^4 - T_\infty^4\right)\,\mathrm{d}s, \quad (7)
$$

where $\bar{q}$ is the prescribed heat flux on $\partial\Omega_q$, T_∞ is the temperature of the surrounding medium, h_c is the convective heat transfer coefficient, ε is the surface emissivity, and the Stefan-Boltzmann constant $\sigma = 5.67032 \times 10^{-8}\,Wm^{-2}K^{-4}$. The last three integrals represents the thermal, convective and radiative boundary conditions on $\partial\Omega_q$, $\partial\Omega_c$, and $\partial\Omega_r$, respectively. Additionally, the essential boundary conditions $T(\bar{\boldsymbol{x}}, t_0) = \bar{T}$ are prescribed on $\partial\Omega_T$.

We approximate T using Voronoi tessellation as $T \approx \sum_{\alpha,n} T^{\alpha n} P^{\alpha n}$ and its derivative using Delaunay tessellation as $T_{,i} \approx \sum_{\beta,m} T_i^{\beta m} Q^{\beta m}$. According to Eq. 4, we can express the relation between $T^{\alpha n}$ and $T_i^{\beta m}$ as

$$
T_i^{\beta m} = w^{\beta m m'} T^{\alpha n} h_i^{\alpha\beta m'n}. \quad (8)
$$

Further, the variables H, q, and the parameters ρ, c are approximated using Voronoi tessellation, while κ is approximated using Delaunay tessellation. Substituting these approximations and $g = g^{\alpha n} P^{\alpha n}$ to Eq. 6, we can obtain the HO-PDS-FEM approximation of the governing equation as

$$
\begin{aligned}
0 &= \left(\rho^\alpha c^\alpha I^{\alpha n n'}\right)\dot{T}^{\alpha n'} + \hat{K}^{\beta\alpha\alpha'nn'}T^{\alpha'n'} - I^{\alpha n n'}H^{\alpha n'} \\
&\quad + I_{\partial\Omega_q}^{\alpha n n'}\bar{q}^{\alpha n'} + I_{\partial\Omega_c}^{\alpha n n'}h_c^\alpha\left(T^{\alpha n'} - T_\infty^{\alpha n'}\right) \\
&\quad + I_{\partial\Omega_r}^{\alpha 11}g^{\alpha 1}\sigma\varepsilon^\alpha\left(\left(T^{\alpha 1}\right)^4 - \left(T_\infty^{\alpha 1}\right)^4\right), \quad (9)
\end{aligned}
$$

where the thermal conductivity matrix $\hat{K}^{\beta\alpha\alpha'nn'} = w^{\beta m m'}h_i^{\alpha\beta\bar{m}n}\kappa^\beta\delta_{ij}h_i^{\alpha'\beta\hat{m}n}$. For the sake of simplicity, we approximated $T^4 \approx (T^{\alpha 1}P^{\alpha 1})^4$.

Modeling mechanical and thermal induced deformation

When modeling the thermo-mechanical interactions, we consider only one-way coupling assuming the rate of mechanical deformation is too slow to produce any heat energy. Assume that the material occupying the above considered domain Ω is linear elastic and subjected to externally applied mechanical and thermal loading along the boundary $\partial\Omega$. Let the initial temperature of the domain be $T(x_i, t = 0) = T_0$, and $\Delta T = T - T_0$. We can express the thermal induced strain ϵ_{ij}^θ as

$$\epsilon_{ij}^{\theta} = \alpha \Delta T \delta_{ij}, \tag{10}$$

where α is the coefficients of thermal expansion. For the sake of simplicity, we assume α to be isotropic. The total strain ϵ_{ij} consists of mechanical strain ϵ_{ij}^{M} and thermal strain ϵ_{ij}^{θ}.

$$\epsilon_{ij} = \frac{1}{2} \left(u_{i,j} + u_{j,i} \right) = \epsilon_{ij}^{M} + \epsilon_{ij}^{\theta}. \tag{11}$$

Since only ϵ_{ij}^{M} generates stress

$$\sigma_{ij} = c_{ijkl} \left(\epsilon_{ij} - \epsilon_{ij}^{\theta} \right). \tag{12}$$

Considering Lagrangian

$$\mathcal{L} = \frac{1}{2} \int \rho \dot{u}_i \dot{u}_i - \epsilon_{ij} c_{ijkl} \left(\epsilon_{kl} - 2\alpha \Delta T \delta_{kl} \right) \, \mathrm{d}v, \tag{13}$$

and applying the Hamilton's principle (i.e. $\delta \int \mathcal{L} dt = 0$), we can obtain

$$\int \delta u_i c_{ijkl} \left(\epsilon_{kl} - \alpha \Delta T \delta_{kl} \right) n_j \, \mathrm{d}s \, \mathrm{d}t + \int \delta u_i \left(\rho \ddot{u}_i - c_{ijkl} \left(\epsilon_{kl} - \alpha \Delta T \delta_{kl} \right)_{,j} \right) \, \mathrm{d}v \, \mathrm{d}t = 0 \tag{14}$$

In the above, the first term contains the natural boundary conditions, and the last contains the desired strong form of the governing equations for deformation under thermal and mechanical loading.

Substituting $u_i \approx \sum_{\alpha} u_i^{\alpha n} P^{\alpha n}$ and $u_{i,j} \approx \sum_{\beta} u_{ij}^{\beta m} Q^{\beta m}$ to $\epsilon_{ij} = \frac{1}{2} \left(u_{i,j} + u_{j,i} \right)$, $\epsilon_{ij}^{\beta m}$ and $u_i^{\alpha n}$ can be related as

$$\begin{aligned} \epsilon_{ij}^{\beta m} &= \frac{1}{2} w^{\beta m m'} \left(h_j^{\beta \alpha m' n} u_i^{\alpha n} + h_i^{\beta \alpha m' n} u_j^{\alpha n} \right) \\ &= w^{\beta m m'} b_j^{\beta \alpha m' n} u_i^{\alpha n}. \end{aligned} \tag{15}$$

Substituting the above approximated field variable into Eq. 14, it is straight forward to obtain the governing linear set of equations for thermal induced mechanical deformations as;

$$0 = \left(I^{\alpha n n'} \rho^{\alpha} \right) \ddot{u}_i^{\alpha n'} - K_{ik}^{\beta \alpha n \alpha' n'} u_k^{\alpha n'} + w^{\beta m m'} b_j^{\beta \alpha m' n} c_{ijkl}^{\beta} \epsilon_{kl}^{\theta \alpha' n'} I^{\beta m \alpha' n'} \tag{16}$$

where $I^{\beta m \alpha' n'} = \int_{\Psi} P^{\alpha' n'} Q^{\beta m} \mathrm{d}v$, $K_{ik}^{\beta \alpha n \alpha' n'} = w^{\beta m m'} b_j^{\beta \alpha m n} c_{ijkl}^{\beta} b_l^{\beta \alpha m' n'}$ is the element stiffness matrix. Note that repeated superscript β does not imply summation.

Modeling crack growth

The use of base functions with compact support within each Voronoi element produces numerous discontinuities in any function approximation. It is the use of conjugate tessellation for approximating derivatives which enables us to define bounded approximations for the derivatives of this discontinuous function approximations. HO-PDS-FEM uses these discontinuties in function approximation to numerically efficiently model propagating cracks or other discontinuties. To elaborate, lets consider the Delaunay triangle with an infinitesimally thin neighborhood
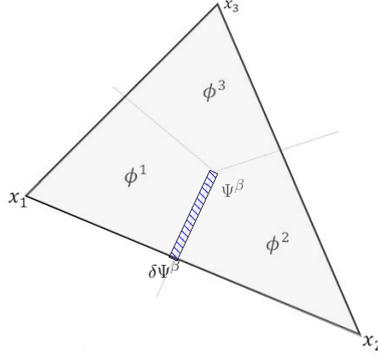
Figure 1: Infinitesimally thin neighborhood of any Voronoi boundary contains a discontinuity.

$\delta\Psi^\beta$ around the boundary of two neighboring Voronoi elements shown in **Fig. 1**. The term $h_j^{\beta\alpha m'n}$ of Eq. 4 establishes the relation between the function and its derivatives, connecting the discontinuous local function approximations over neighboring Voronoi elements. The evaluation of $h_j^{\beta\alpha m'n}$ involves integrations over Ψ^β and $\delta\Psi^\beta$ as

$$h_j^{\beta\alpha m'n} = \int_{\Psi^\beta} Q^{\beta m} P_{,j}^{\alpha n'} \phi^\alpha \mathrm{d}v + \int_{\delta\Psi^\beta} Q^{\beta m} P^{\alpha n'} \phi^\alpha{}_{,j} \, \mathrm{d}v. \tag{17}$$

In the above, the last term represents the contribution to the derivatives from the discontinuities of the function approximation along Voronoi boundaries. Using a conjugate tessellation (e.g., Delaunay) and utilizing the divergence theorem, PDS manages to find bounded contributions from this discontinuity to the derivatives. Hence, one can model a discontinuity forming along the boundary $\delta\Psi^\beta$ by nullifying this last term, which is equivalent to dropping the contributions to the derivative from the corresponding Voronoi boundary. Thus, all that is required to model a propagating discontinuity in displacement and thermal fields is to drop the contributions from any desired Voronoi boundaries and recalculate the corresponding element matrices. This PDS-FEM's numerical treatment to introduce a discontinuity does not require introduction of new degrees of freedoms like in ordinary FEM, or complicated numerical treatments like in extended FEM. The cost of this numerically light treatment for discontinuities is the lower accuracy of crack tip stress or thermal gradient fields. However, as demonstrated in [16, 18], PDS-FEM provides sufficient crack-tip accuracy for applications. This numerically light treatment enables one to simulate growth of discontinuties in large scale 3D domains such as high-power LASER induced cracking in large conceete structures.

Verification and Validation

This section presents the verification tests to check the correctness of the computer implementation, and a validation test to check the developed numerical scheme's ability to reproduce thermal induced cracking observed in the real-world. From here onward, PDS-FEM with the sets of polynomial bases $\{P^{\alpha n}\} = \{1\}\phi^\alpha(\boldsymbol{x})$ and that with the sets $\{Q^{\beta m}\} = \{1\}\psi^\beta(\boldsymbol{x})$ is referred as the 0th-order PDS-FEM, and $\{P^{\alpha n}\} = \{1, \boldsymbol{x} - \boldsymbol{x}^\alpha\}\phi^\alpha(\boldsymbol{x})$, $\{Q^{\beta m}\} = \{1, .., \boldsymbol{x} - \boldsymbol{x}^\beta, (\boldsymbol{x} - \boldsymbol{x}^\beta)^2\}\psi^\beta(\boldsymbol{x})$ is referred as the 1st-order PDS-FEM.
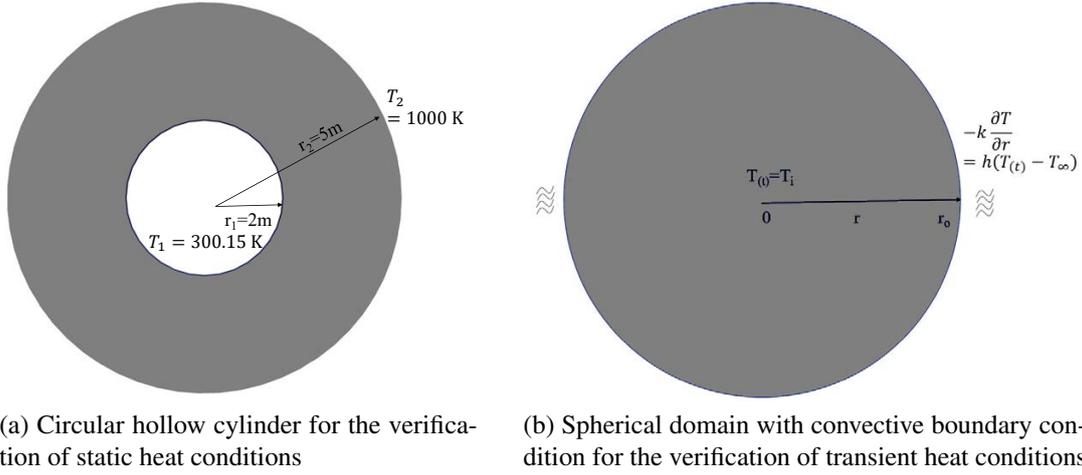
(a) Circular hollow cylinder for the verification of static heat conditions

(b) Spherical domain with convective boundary condition for the verification of transient heat conditions

Figure 2: Problem settings for the verifications of the static and transient heat conduction.
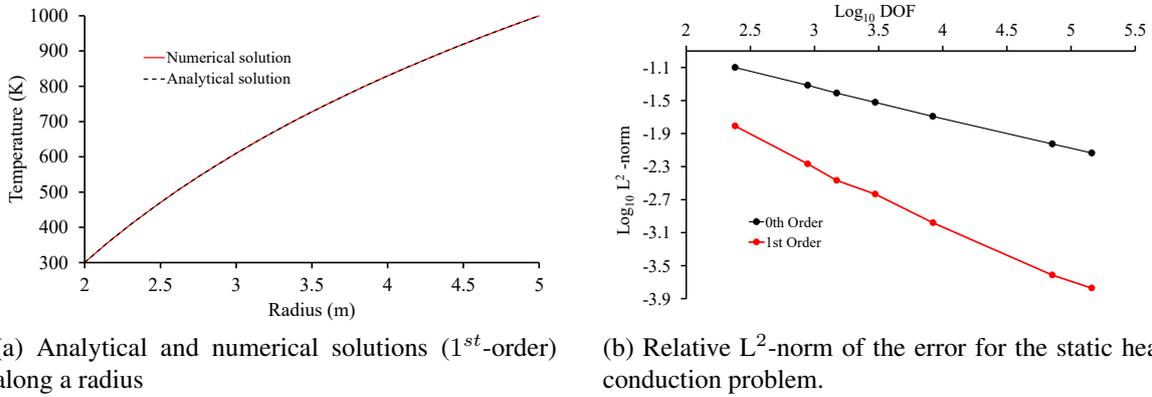
(a) Analytical and numerical solutions (1st-order) along a radius

(b) Relative L^2-norm of the error for the static heat conduction problem.

Figure 3: Accuracy and the rate of convergence for the static heat conduction test.

Verification of static heat conduction

We consider the circular hollow cylinder with inner radius 2m and outer radius 5m shown in Fig. 2a for the verification of static heat conduction. The temperature at the inner and outer surfaces of the cylinder are set to 300.15K and 1000K. The values of Young's Modulus, Poisson's ratio and thermal conductivity are 35 GPa, 0.20, 2.5 $\text{Wm}^{-1}\text{K}^{-1}$, respectively. The domain is modeled with 2101201 tetrahedral elements and 375263 Voronoi elements.

Analytical and numerical solutions along radii shown in Fig. 3a indicate that the two solutions are in good agreement, thus verifying the proposed scheme in quasi-static settings. Equation 18 shows the analytical solution for this problem from [23], where r is the distance from the center. Further, the relative L^2-norm of the error $\dfrac{\sqrt{\int (T_{analytical} - T_{numerical})^2 \, dv}}{\sqrt{\int (T_{analytical})^2 \, dv}}$ shown in Fig. 3 confirms that the numerical solutions produce expected rates of convergence. Specifically, the 0th-order PDS-FEM produces the rate of convergence 1 and the 1st-order PDS-FEM produces that of 2.

$$(T_1 - T) \, ln(\frac{r_2}{r_1}) = (T_1 - T_2) \, ln(\frac{r}{r_1}) \tag{18}$$

(a) The analytical solution is evaluated using first 770 eigen roots.

(b) Variation of relative L^2-norm of the error with respect to time.
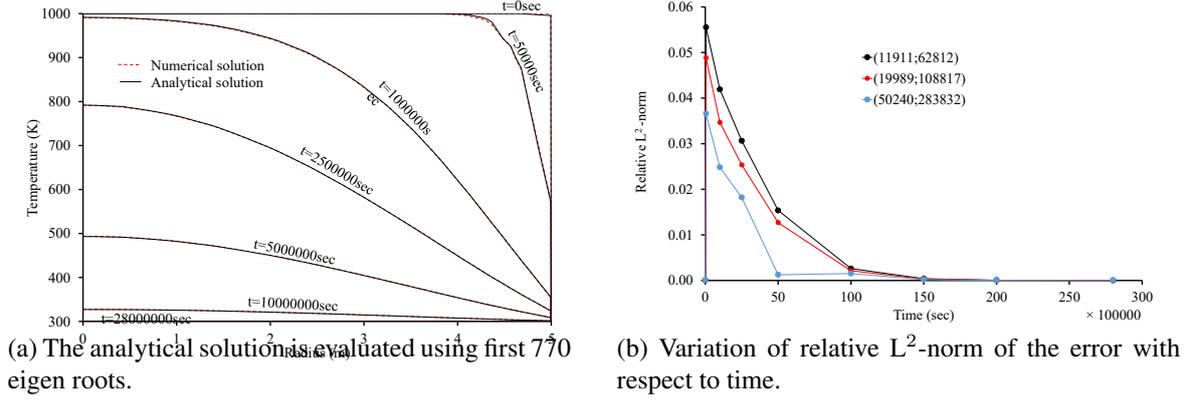
Figure 4: Accuracy and convergence of the transient thermal conduction test.

Verification of Transient Heat Conduction

For the verification of transient heat conduction, we use the spherical domain shown in Fig. 2b with 5 cm radius. The Young's modulus, Poisson's ratio and density of the material are 35 GPa, 0.20, 2400 kgm^{-3}, respectively. Thermal conductivity(k), convective heat transfer coefficient (h_c) and specific heat capacity(c) are 2.5 ($\mathrm{Wm}^{-1}\mathrm{K}^{-1}$), 12 ($\mathrm{Wm}^{-2}\ \mathrm{K}^{-1}$) and 960 ($\mathrm{Jkg}^{-1}\mathrm{K}^{-1}$), respectively. Convective boundary conditions are imposed using relation $-\kappa\frac{\partial T}{\partial r} = h(T_{r,t}-T_\infty)$, the initial temperature is assumed to be 1000K, and the far-field temperature T_∞ is set to be 300.15K. The domain is modeled with 92853 tetrahedral elements and 26186 Voronoi elements. A forward Euler scheme has been employed for time integration with the time increment of 3.125 sec.

Our numerical results at different time steps are compared with the analytical solution given by the following equation [22].

$$\frac{T(r,t) - T_\infty}{T_i - T_\infty} = \sum_{n=1}^{\infty} A_n \frac{sin(\lambda_n \frac{r}{r_o})}{\lambda_n \frac{r}{r_o}} e^{-\left(\frac{\lambda_n}{r}\right)^2 \alpha t}, \qquad (19)$$

where $A_n = \frac{4(sin\lambda_n - \lambda_n cos\lambda_n)}{2\lambda_n - sin(2\lambda_n)}$, r is any point along a radius, and λ_n denotes the roots of the eigen function $1 - \lambda_n cot\lambda_n = \frac{hr}{k}$. Figure 4a compares analytical and numerical solutions along a radial direction, at a several time steps. The close agreement of the two solutions demonstrates that our numerical implementation is accurate and maintains a high accuracy even after 8,960,000 time steps at intervals of 3.125 seconds. Further, the relative L^2-norm of the error shown in Fig. 4b indicates that the error is negligible and diminishes with time. Further, it shows that the relative L^2-norm of the error reduces when the domain is tessellated with finer elements. The pair of number in the legend of Fig. 4b correspond to number of Voronoi and tetrahedral elements, respectively.

Validation of thermal induced cracking

To validate the crack propagation functionality of the developed code, we numerically reproduced Jiang *et al.*'s quenching experiment[7] and compared the numerically and experimentally obtained crack patterns, both qualitatively and quantitatively. In their experiment, Jiang *et al.* heated a 1mm thick 99% $\mathrm{Al_2O_3}$ plates of dimension 50 mm×10 mm×1 mm to different temperatures and suddenly immersed it in a water bath at 20 °C (see Fig. 5). They repeated this experiment 4 times with samples heated to $T_0 = 300°\mathrm{C}$, $400°\mathrm{C}$, $500°\mathrm{C}$ and $600°\mathrm{C}$, and recorded
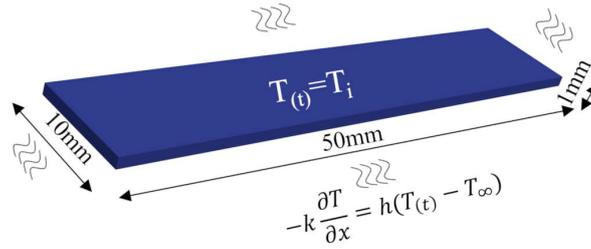
Figure 5: 1mm thick Al$_2$O$_3$ plate subjected to convective boundary conditions.

the resulting crack patterns. In all their experiments, the top and bottom faces of dimensions 50×10 mm^2 had been insulated such that convective heat loss occurred only from the surfaces with 1 mm width.

We modeled the thin plate with 409900 tetrahedral elements and 136160 Voronoi elements, and assigned convective boundary conditions on all the 1mm thick surfaces to mimic the loss of heat to the water bath. Young's modulus, Poisson's ratio, density and surface energy density of 99% Al$_2$O$_3$ are 370 GPa, 0.22, 3980 kgm^{-3} and 12.16 Jm^{-2}, respectively [7]. Convective heat transfer coefficient, specific heat capacity, thermal conductivity and coefficient of thermal expansion are assumed to be 54500 Wm^{-2}K^{-1}, 880 Jkg^{-1}K^{-1}, 31 Wm^{-1}K^{-1} and 7.5 $\times 10^{-6}$ K^{-1}, respectively[7, 8].

Compared to the speed of stress waves, the propagation of heat is several orders of magnitudes slower. Further, stress waves emitted by propagating thermal cracks carry a negligible amount of energy. Hence, only the thermal diffusion (i.e., Eq. 9) is simulated as dynamic process, while the mechanical deformation is simulated as a quasi-static process (i.e., the inertia term in Eq. 16 was neglected). A forward Euler scheme with time increment 0.1 μs was used for simulating the thermal diffusion.

Though J-integral based criterion is preferred in simulating crack propagation, we opted for maximum principle stress based criterion due to its simplicity and lower numerical overhead. Further, our use of parallel computing to reduce the computational time makes it somewhat complicated to implement J-integral based criterion. Hence, all the crack propagation simulations presented in this paper use principle stress based failure criterion (i.e., if the maximum principle stress of an element is larger than the tensile strength of the material, the Voronoi boundary with the normal closest to the maximum principle direction is broken as explained in the previous section).

Figure 6 compares the numerically obtained crack patterns with those from Jiang *et al.*'s experiment. The cracks grow perpendicular to the boundaries. It is clearly visible that higher initial temperatures produces longer cracks and closer crack spacing. Qualitative comparisons of length and spacing of the cracks show that the numerically obtained crack patterns are in reasonably good agreement with those from the experiment.

To quantitatively compare the results from different samples, Jiang *et al.* have classified the cracks into three groups as shorter cracks, middle cracks and longer cracks, according to the criterion given in Table 1. Further, they have ignored the cracks shorter than 0.25 mm in the observations of the $T_0 = 300°C$ sample, and ignored the cracks shorter than 0.125 mm in the remaining samples. We used the same classification and compared the numerically obtained crack patterns with those from Jiang *et al.*'s experiment.

Statistics of the quantitative comparison shown in Fig. 7 show that the numerically obtained

T	Shorter	Middle	Longer
300°C	$l_{cr} \leq 1.1$	$1.1 < l_{cr} < 3.4$	$l_{cr} \geq 3.4$
400°C	$l_{cr} \leq 0.55$	$0.55 < l_{cr} < 1.75$	$l_{cr} \geq 3.9$
500°C	$l_{cr} \leq 0.5$	$0.50 < l_{cr} < 1.65$	$l_{cr} \geq 4.05$
600°C	$l_{cr} \leq 0.4$	$0.40 < l_{cr} < 1.4$	$l_{cr} \geq 4.1$

Table 1: Jiang *et al.*'s classification of cracks[7]. l_{cr} is crack length.

(a) $T = 300°C$

(b) $T = 400°C$

(c) $T = 500°C$

(d) $T = 600°C$
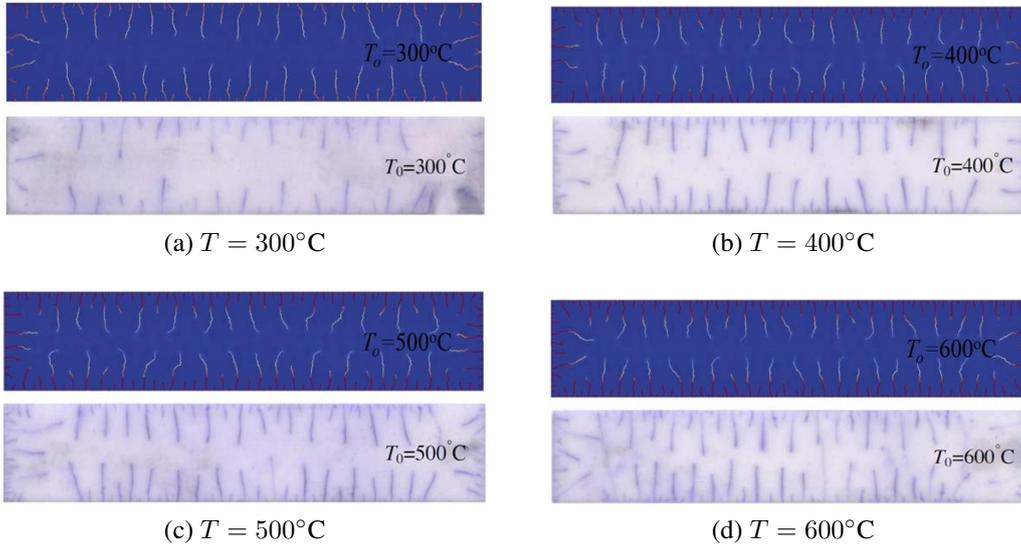
Figure 6: Crack patterns after thermal shocks. (Upper figure is from simulation and lower is from experiment.)

crack patterns are in reasonably good agreement with the experimental observations. This reasonable agreement of both qualitative and quantitative comparisons demonstrate that the developed numerical model is capable of simulating thermal induced cracking of brittle materials.

Thermal Cracking of Concrete

As a preliminary assessment of the developed system's fitness for simulating high-power LASER induced cracking in bulk concrete, we numerically reproduced a LASER cutting experiment we conducted at an industrial facility. Unlike the experiment reproduced in the previous section, this experiment involves highly concentrated heat sources, radiative and convective boundary conditions and heterogeneous materials. Further, it involves diffusion of water, evaporation and condensation inside concrete capillaries. However, we did not include any of these water related phenomena in the current study.

In this preliminary assessment, we consider an experiment conducted on an RC block with dimensions $50\,\text{cm} \times 10\,\text{cm} \times 10\,\text{cm}$, and two reinforcement bars of diameter 36mm (see Fig. 8). The numerical model consists of three materials; cement mortar, aggregates, and steel. The total number of aggregates are determined according to their volume fraction in ordinary concrete. Aggregates are assumed to be spherical inclusions of diameter 20mm and distributed randomly. We used a highly refined model with 12 million Delaunay and 2 million of Voronoi elements. These numbers are chosen such that a resulting Voronoi elements have a size of 3mm such that a volume of a Voronoi element is closer to 3~4 sand particles. We assume all the three materials to be linear elastic, and Table 2 provides their mechanical and thermal properties. The surface emissivity ε, surface energy and the convective heat transfer coefficient h_c of the RC block are

(a) $T = 300°C$

(b) $T = 400°C$

(c) $T = 500°C$

(d) $T = 600°C$
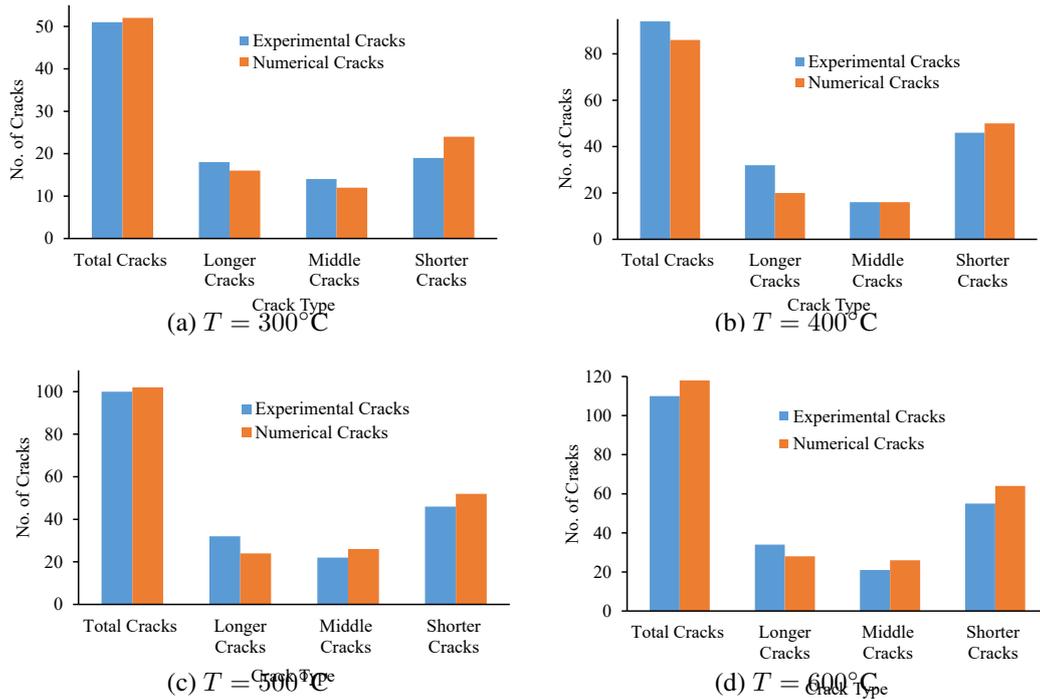
Figure 7: Statistics of the transient problem.

set to 0.9, 97.5 Jm^{-2} and 19.73 $Wm^{-2}K^{-1}$, respectively.

In the experiment, the high-power LASER melted the concrete in the vicinity of the LASER beam, creating a 10~15 mm wide cut through concrete block. The melted concrete flowed as a highly viscous fluid. Due to the extreme conditions (above 2000°C), it was impossible to take any measurements of conditions inside the cut. The melting process makes it difficult to use thermal flux boundary conditions. We made use of the fact that the concrete LASER contact region always consists of melts of the material, and model the LASER beam as a straight thin beam of 2 mm radius with temperature equal to the melting temperature of concrete. The red line in Fig. 8 shows the position of the LASER beam at the start (i.e. t=0 s). This high temperature zone is moved vertically through the sample at a speed of 6 mms^{-1} to mimic the target LASER speed.

(a) different volume inclusions in the mesh

(b) imposed boundary conditions
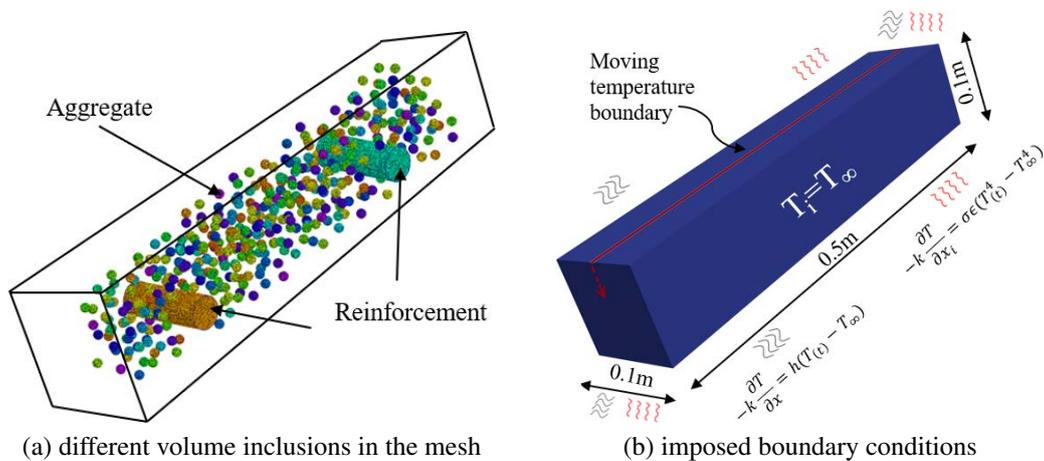
Figure 8: Geometric and boundary condition details for numerical simulation.

	Cement paste	Aggregate	Reinforcement
Young's modulus E (GPa)	17.5	50	200
Poisson's ratio ν	0.1	0.35	0.30
Density ρ (kgm^{-3})	2201	1076	8050
tensile strength σ_t (MPa)	2.25	3	400
Thermal conductivity k (Wm^{-1}K^{-1})	2.43	2.75	50.2
Specific heat capacity c (Jkg^{-1}K^{-1})	1040	850	510
Coefficient of thermal expansion α (K^{-1})	15	7	11

**Table 2: Properties of concrete constituents**

Results and Discussion

Figure 9 shows the simulated crack patterns, while Fig. 10 shows crack patterns obtained from a set of experiments. Overall, the numerical crack pattern is in a reasonable agreement with the experiment. The large cracks extending vertically through the sample up to the top surface are clearly visible in both the numerical and experimental results (see Fig. 9(a), Fig. 10(a) and (b)). In most of the experiments, we observed that these vertical cracks nearly split the sample into two. Several large cracks, some of which even when reaching the top surface, are clearly visible around the reinforcement bars both in the experiment and the simulation. These crack around the reinforcements are induced due to the high thermal conductivity of reinforcement bars and uneven thermal expansions of reinforcement bars and concrete. In the simulation, a significant amount of cracks have appeared in the vicinity of the LASER beam due to the constant presence of the heat source inducing high thermal stresses. Though not that many, a number of similar cracks of hairline openings were observed in the experiment. It is further observed that concrete has cracked along the horizontal line joining both reinforcement bars. However, such horizontal cracks were observed only in the experiments with 4 reinforcement bars. Though fully grown horizontal cracks were not observed in the samples with 2 reinforcement bars, initiation of such cracks was visible near the reinforcement bars (see the vicinity of the left reinforcement of Fig. 10(b)).

The above is only a qualitative comparison based on naked eye observations of the sample surface. Naked eye observations of surface cannot uncover any cracks within the sample and even our observations may have failed to identify hairline cracks on the surface. Hence, in the future, reliable quantitative comparisons have to be conducted by comparing the numerical results with CT scans of the samples from the experiments.

Concluding Remarks

We presented an extension of HO-PDS-FEM to simulate heat conduction, thermal induced mechanical deformation of brittle elastic solids and cracking. Convective and radiative boundary conditions are included in the formulation since the samples can be heated to extreme temperatures (e.g. above 2000°C) during the target high-power LASER cutting of concrete. Our verification tests for static and transient heat conduction produced excellent agreement with analytical solutions, demonstrating the accuracy of the implemented code. Simulations with 0th-order and 1st-order PDS-FEM produced the expected 1st-order and 2nd-order convergence rates, respectively. Further, reproducing an experiment by Jiang *et al.*, we validated that the developed model could reproduce the real-world observations.

Crack patterns induced in an RC block by a moving high-power LASER beam was simulated as a preliminary application of the developed numerical scheme. A high fidelity mesh (total of 12

| (a) cracks on top and left faces | (b) cracks on the bottom and right faces |
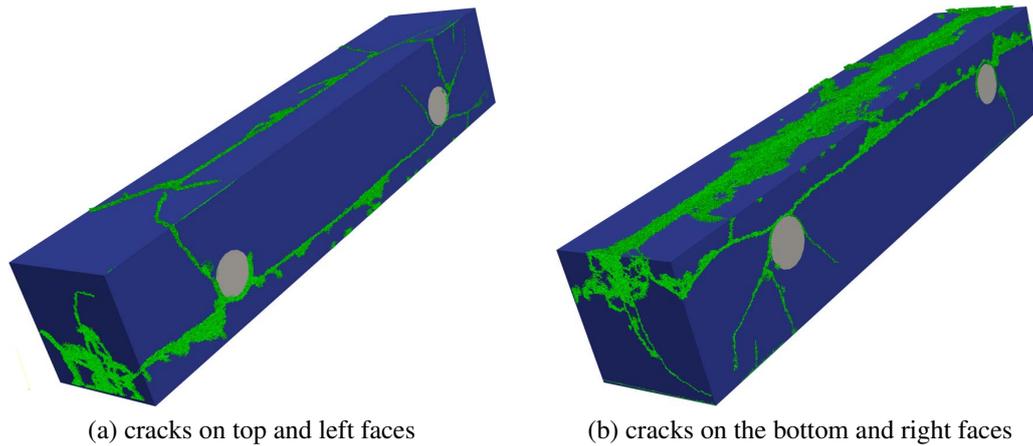
Figure 9: Simulated crack patterns in a RC sample due to a moving source of high temperature.

| (a) cracks on top surface | (b) cracks around the reinforcements |
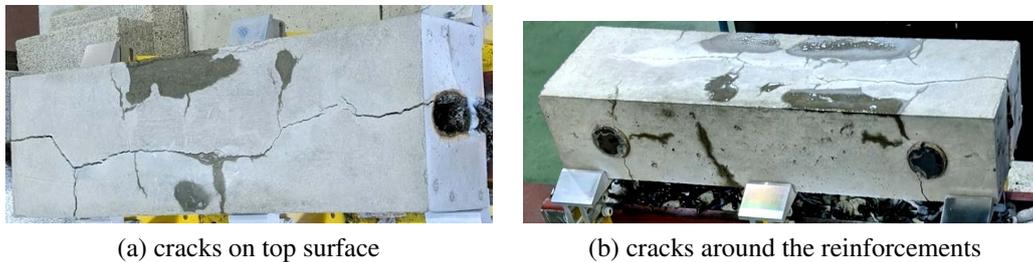
Figure 10: Crack patterns observed in an experiment.

million elements) consisting of randomly distributed aggregates, steel reinforcement bars and cement matrix is used with the aim of accurately capturing the crack patterns. Naked eye comparison shows that the simulated crack patterns (directions and locations) are in good agreement with those of experiment. We plan to conduct quantitative evaluation of the numerical results by comparing with the CT scans of the experimental samples.

Acknowledgments

References

[1] Long, N. P., Daido, H., Yamada, T., Nishimura, A., Hasegawa, N., and Kawachi, T. (2017) Experimental characterization of concrete removal by high-power quasi-continuous wave fiber laser irradiation. *Journal of Laser Applications*, **29**(4), 041501-1-11.

[2] Hilton, P., and Walters, C. (2010) The Potential of High Power Lasers in Nuclear Decommissioning - 10092. *WM2010 Conference*, March 7-11, Phoenix, AZ.

[3] Seo, Y., Lee, D. and Pyo, S. (2020) High-Power Fiber Laser Cutting for 50-mm-Thick Cement-Based Materials. *Materials* 13, pp. 1113.

[4] Geyer, J.F. and Nemat, N.S. (1982) Experimental investigation of thermally induced interacting cracks in brittle solids. *Int. J. Solids Struct.* **18**, 349–356.

[5] Bourdin, B., Francfort, G.A. and Marigo, J.J. (2000) Numerical experiments in revisited brittle fracture. *J. Mech. Phys. Solids* **48**, 797–826.

[6] Bourdin, B. (2007) The variational formulation of brittle fracture:numerical implementation and extensions. *IUTAM symposium on discretization methods for evolving discontinuities* New York: Springer.

[7] Jiang, C.P., Wu, X.F., Li, J., Song, F., Shao, Y.F., Xu, X.H. and Yan, P. (2012) A study of the mechanism of formation and numerical simulations of crack patterns in ceramics subjected to thermal shock *Acta-Mater.* **60**, 4540–4550.

[8] Li, J., Song, F. and Jiang, C.P. (2013) Direct numerical simulations on crack formation in ceramic materials under thermal shock by using a non-local fracture model. *J. Eur. Ceram Soc.* **33**, 2677–2687.

[9] Tang, S.B., Zhang, H., Tang, C.A. and Liu, H.Y. (2016) Numerical model for the cracking behavior of heterogeneous brittle solids subjected to thermal shock *Int. J. Solids & Structures*, **80**, 520–531.

[10] Yu, F., Zhe, W., Fengyu, R., and Daguo, W. (2017) Numerical model of thermo-mechanical coupling for the tensile failure process of brittle materials. *AIP Advances* **7**, 105023.

[11] Chu, D., Li, X. and Liu, Z. (2017) Study the dynamic crack path in brittle material under thermal shock loading by phase field modeling. *Int. J. Fract*, **208**, 115-130.

[12] Moes, N., Nicolas, D.J. and Belytschko, T. (1999) A finite element method for crack growth without remeshing. *Int. J. for Numerical Methods in Engineering*, **46**(1), 131–150.

[13] Sukumar, N., Moes, N., Moran, B. and Belytschko, T. (2000) Extended finite element method for three-dimensional crack modelling. *Int J for Numerical Methods in Engineering*, **48** (11), 1549–1570.

[14] Moes, N. and Belytschko, T. (2002) Extended finite element method for cohesive crack growth. *Engineering Fracture Mechanics*, **69** (7), 813–833.

[15] Hori, M., Oguni, K. and Sakaguchi, H. (2005) Proposal of FEM implemented with particle discretization scheme for analysis of failure phenomena. *J. of Mech. and Phys. of Solids*, **53**, 681–703.

[16] Wijerathne, M.L.L., Oguni, K. and Hori, M. (2009) Numerical analysis of growing crack problem using particle discretization scheme. *Int. J. for Numerical Methods in Engineering*, **80**, 46–73.

[17] Wijerathne, M.L.L., Hori, M., Sakaguchi, H. and Oguni,K. (2010) 3D dynamic simulation of crack propagation in extra corporeal shock wave lithotripsy. *IOP Conference Series: Materials Science and Engineering*, **10**(1).

[18] Mahendra K.P., Wijerathne, M.L.L. and Hori, M. (2019) Numerical modeling of brittle cracks using Higher Order Particle Discretization Scheme-FEM. *Int J. Comput. Methods*, **16**(4), 1843006.

[19] Mahendra K.P., Wijerathne, M.L.L., Hori, M., Ichimura, T. and Tanaka,S. (2014) Implementation of Finite Element Method with higher order Particle Discretization Scheme. *J. of Japan Society of Civil Engineers, Ser.A2*, **70**(2), 297–305.

[20] Quaranta, L., Maddegedara, L., Okinaka, T. and Hori, M. (2020) Application of PDS-FEM to simulate dynamic crack propagation and supershear rupture. *Computational Mech.* **65**, 1289–1304.

[21] Quaranta, L., Maddegedara, L., Hori, M., (2019), Interaction of horizontally aligned coplanar 3D penny cracks under compression. *J. of the Mechanics and Physics of Solids*,**131**, 180–203.

[22] Yunus, A. Cengel, A.J. and Ghajar. (2015) *Heat and Mass Transfer; Fundamentals & Applications*, 5th Edition, McGraw-Hill.

[23] Mills, A.F. (1999) Heat Transfer, Second Edition, Prentice-Hall, New Jersey.

Formulation of a Novel Implicit Stress Integration Algorithm based on Plastic Consistency Parameter and its Verification Using von Mises Plasticity

*M. A. K. M. Dharmasiri[1], †Maddegedara Lalith[2], Kohei Fujita[2], Tsuyoshi Ichimura[2]

and Muneo Hori[3]

[1]Dept. of Civil Eng., The Univ. of Tokyo, Japan
[2]Earthquake Research Inst., The Univ. of Tokyo, Japan
[3]Japan Agency for Marine-Earth Science and Technology, Yokohama, Japan
*Presenting author: kasun@eri.u-tokyo.ac.jp
†Corresponding author: lalith@eri.u-tokyo.ac.jp

Abstract

We propose a novel algorithm for integrating the standard rate form of plasticity in which the state variables are gradually returned onto the yield surface by a series of implicit plastic correction stages. Its features are discussed in relation to the Closest Point Projection Method (CPPM) and the Cutting Plane Method (CPM). As in CPPM, it is straightforward to derive a consistent tangent operator for the proposed method. Like in CPM, it uses the successive linearization of the yield function about the current state to evaluate the state variables. The proposed integration method can be easily implemented in existing finite element analysis frameworks since the required first and second order derivatives are similar to those required in CPPM. Several numerical tests are performed using von Mises plasticity and linear hardening rules. Single material point tests reveal that the proposed algorithm provides near identical stress remapping to that of CPM and CPPM. For the classical perforated sheet benchmark with both linear isotropic hardening and linear kinematic hardening, CPM, CPPM and the proposed methods produce near identical results. For the combined hardening, a slight disparity between the results from CPPM with the other two methods is observed. Further, the multi-element tests demonstrate that the consistent tangent operator of the proposed method is on par with that of CPPM.

Keywords: plasticity; integrating the rate form; consistent tangent operator; von Mises;

1 Introduction

Recent advances in computing has made large scale simulations involving nonlinear analysis a reality. Search for novel algorithms for nonlinear problems can contribute to optimally utilizing the available computational resources by choosing suitable algorithm according to the simulated phenomena and the computer hardware. One such nonlinear problem is classical plasticity. Dependig on the required accuracy, involved phenomena, stability, parallel computing model, etc. we can choose a suitable algorithm for integrating the rate-form of classical plasticity from a number of algorithms available. Scalet and Auricchio [1] provides an excellent summary of such classical and less classical methods used for stress integration.

Closest Point Projection Method (CPPM)[2] and the Cutting Plane Method (CPM)[3] are the most popular classical algorithms used to predict the evolution of state variables such as stresses and plastic internal variables. The earliest ideas about CPPM were set forth by Wilkins [2] and subsequent contributions by several others have made this implicit algorithm, a very capable, accurate, albeit a relatively computationally costly numerical integration scheme. The CPM on the other hand, which was introduced by Ortiz and Simo [3], is an incomplete implicit algorithm which follows the path of the steepest descent [4] to arrive at estimates for the state

variables during plastic deformations. Summarily, it could be said that for state variable remapping, CPPM is computationally costly due to its reliance on the second order derivatives for stress integration procedure, and CPM, in this regard, is computationally inexpensive. CPPM uses a residual based approach to estimate the remapped stresses while CPM utilizes successive linearization of the yield functions at the current state to estimate the plastic consistency parameter, thereby updating the state variables.Though computationally expensive CPPM is unconditionally stable, while computationally light CPM is not.

In this study, having observed these classical methods, an attempt has been made to introduce an implicit numerical integration scheme for rate independent plasticity, in which the state variables are gradually returned onto the yield surface by a series of implicit plastic correction stages. Similar to CPM, in the proposed method, the state variables are evaluated by the successive linearization of the yield function about the current state. For state variables remapping, the proposed scheme also require second order derivatives, like in CPPM. However, the solution strategy is marginally lower in terms of computational cost to that of CPPM, per iteration basis, for state variables remapping excluding the evaluation of the consistent tangent operator. On the other hand, the proposed framework requires iterative update of elasto-plastic tangent operator during material point iterations unlike in CPPM, which is evaluated only once during material point iterations. However, the overall computational efficiency depends not only on the computational cost for each of the material point iterations, but also on the accuracy of the elasto-plastic tangent operator and how accurate a prediction can be made regarding the evolution of state variables during material point iterations.

A brief summary of the classical theory of plasticity [5, 6] which is the basis for all of the three stress integration algorithms, CPPM, CPM and the proposed method is presented in section 2. The stress integration algorithms and the elasto-plastic tangent operators of CPPM, CPM and the proposed method and a comparison of their features is presented in section 3. In this paper, von Misses plasticity is used to study the accuracy and the performance of the newly minted proposed stress integration algorithm with its consistent tangent operator, in relation to CPPM and CPM methods. The verification problems considered and the results with a comparison are presented in section 4.

We use $\dot{f}$ to denote the time derivative of the quantity f and when quantities are represented with two superscripts separated by a comma, i.e. $(.)^{p,k}$, the first superscript p denotes the state whereas the second superscript k denotes the material point iteration.

2   Classical Flow Theory of Plasticity

We consider classical flow theory [5, 6] based rate independent infinitesimal elasto-plastic deformations of an isotropic continuum subjected to suitable Dirichlet and Neumann boundary conditions prescribed as a function of time $t \in \mathbb{R}^+$. The linearized Green strain tensor for the induced infinitesimal deformation field $\boldsymbol{u}$ is defined as

$$\boldsymbol{\varepsilon} = \nabla^{sym}\boldsymbol{u} \tag{2.1}$$

Following Coleman [7], the history dependence of stress is quantified as

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}\left(\boldsymbol{\varepsilon}, \boldsymbol{\kappa}\right), \tag{2.2}$$

where the internal plastic variable $\boldsymbol{\kappa}$ consists of hardening parameters such as the size of the yield surface (isotropic hardening) and translation direction of the yield surface (back stress

in kinematic hardening). Further, we assume that that following assumptions of the classical plasticity theory hold true.

1. Additive decomposition $\varepsilon = \varepsilon^e + \varepsilon^p$, where ε^e and ε^p are elastic and plastic contributions.

2. $\boldsymbol{\sigma} = \mathbf{C} : \varepsilon^e$, where $\mathbf{C}$ is the fourth order elastic tangent tensor. This implies

$$\dot{\boldsymbol{\sigma}} = \mathbf{C} : \left(\dot{\varepsilon} - \dot{\varepsilon}^p\right). \tag{2.3}$$

3. The stress should always satisfy $\phi\left(\boldsymbol{\sigma}, \boldsymbol{\kappa}\right) \leq 0$, where $\phi\left(\boldsymbol{\sigma}, \boldsymbol{\kappa}\right)$ is a suitable convex scalar function known as the yield criterion. Material behaves elastically when $\phi\left(\boldsymbol{\sigma}, \boldsymbol{\kappa}\right) < 0$ and plastically when $\phi\left(\boldsymbol{\sigma}, \boldsymbol{\kappa}\right) = 0$.

4. The evolution of $\dot{\varepsilon}^p$ and $\dot{\boldsymbol{\kappa}}$ are defined by

$$\dot{\varepsilon}^p = \dot{\lambda}\boldsymbol{m}$$
$$\dot{\boldsymbol{\kappa}} = \dot{\lambda}\mathbf{A}(\boldsymbol{\sigma}, \boldsymbol{\kappa}, \varepsilon^p) \tag{2.4}$$

where $\dot{\lambda}\,(\geq 0)$ is the plastic consistency parameter, $\boldsymbol{m} = \frac{\partial \psi}{\partial \boldsymbol{\sigma}}$ is the flow vector specifying the direction of the plastic flow (where ψ is the plastic potential surface and for associative flow rules $\psi = \phi$), and $\mathbf{A}(\boldsymbol{\sigma}, \boldsymbol{\kappa}, \varepsilon^p)$ is the generalized form of hardening modulus. Furthermore, we can impose the consistency condition that $\dot{\lambda}\dot{\phi} = 0$ and Kuhn–Tucker complementary conditions that $\dot{\lambda} \geq 0, \quad \phi \leq 0, \quad \dot{\lambda}\phi = 0$ on the consistency parameter, $\dot{\lambda}$ and yield criterion, ϕ.

3 Stress Integration Algorithms and Elasto-plastic Tangent Operators

Due to their non-linear nature, numerical schemes are required to integrate the governing rate forms of plasticity given by the Eqs. 2.3, and 2.4. Most of the available numerical methods make different approximations for these rate forms to obtain numerical schemes with different properties. In this section, we present the formulations of the widely used CPM and CPPM, and a novel fully implicit return mapping stress integration algorithm in which the state variables are gradually returned onto the yield surface by a series of implicit plastic correction stages.

For nonlinear finite element analysis using Newton-Raphson method, a material tangent operator is required to compute the element stiffness matrix. There are two tangent operators, such as the continuum tangent operator and the consistent tangent operator. The continuum tangent operator is constructed by making use of the satisfaction of the plastic consistency condition stated in section 2, while the consistent tangent operator is consistent with the algorithm that is used to compute the state variables. While the continuum tangent operator can be used as the elastoplastic tangent modulus / operator in any numerical integration scheme, the consistent tangent operator may not be available for some integration schemes. In this study, we also present the formulation of a consistent tangent operator for the proposed algorithm. .

The non-linear nature of plastic deformation problems requires two levels of iterative solving using suitable numerical schemes; global-level iterations using a scheme such as Newton-Raphson to determine displacement field of the domain, and material-level iterations to determine the resulting state of stress using a scheme such as CPPM. In the following discussion, we assume that we are at the $(k+1)^{th}$ material-level iteration of the $(n+1)^{th}$ load step (global-level iteration).

3.1 Cutting Plane Method (CPM)

The main characteristic of CPM is that it express all the state variable as a function of the plastic consistency parameter $\Delta\lambda$ (i.e., $\boldsymbol{\sigma} = \boldsymbol{\sigma}(\Delta\lambda)$ and $\boldsymbol{\kappa} = \boldsymbol{\kappa}(\Delta\lambda)$) and linearizes ϕ with respect to $\Delta\lambda$ around the current state (i.e. $\boldsymbol{\sigma}^k$ and $\boldsymbol{\kappa}^k$).

CPM approximates $\boldsymbol{\sigma}^{k+1}$ and $\boldsymbol{\kappa}^{k+1}$ as

$$
\boldsymbol{\sigma}^{k+1} = \begin{cases} \boldsymbol{\sigma}_e = \boldsymbol{\sigma}_n + \mathbf{C} : \boldsymbol{\Delta\varepsilon} & \text{if } k = 0 \\ \boldsymbol{\sigma}^k - \Delta\lambda\mathbf{C} : \boldsymbol{m}^k & \text{if } k > 0 \end{cases}
$$
$$
\boldsymbol{\kappa}^{k+1} = \begin{cases} \boldsymbol{\kappa}_n & \text{if } k = 0 \\ \boldsymbol{\kappa}^k + \left.\frac{\partial\boldsymbol{\kappa}}{\partial\Delta\lambda}\right|^k \delta\lambda & \text{if } k > 0, \end{cases}
\tag{3.1}
$$

and expresses ϕ as a function of state variables at unknown state as $\phi\left(\boldsymbol{\sigma}^{k+1}\left(\Delta\lambda\right), \boldsymbol{\kappa}^{k+1}\left(\Delta\lambda\right)\right) = 0$. To iteratively solve this nonlinear function, CPM linearizes ϕ^{k+1} around the the current state $\left(\boldsymbol{\sigma}^k, \boldsymbol{\kappa}^k\right)$ as

$$
\phi^{k+1} \approx \phi^k + \left(\left.\frac{\partial\phi}{\partial\boldsymbol{\sigma}}\right|^k : \left.\frac{\partial\boldsymbol{\sigma}}{\partial\Delta\lambda}\right|^k + \left.\frac{\partial\phi}{\partial\boldsymbol{\kappa}}\right|^k : \left.\frac{\partial\boldsymbol{\kappa}}{\partial\Delta\lambda}\right|^k\right) \delta\lambda = 0.
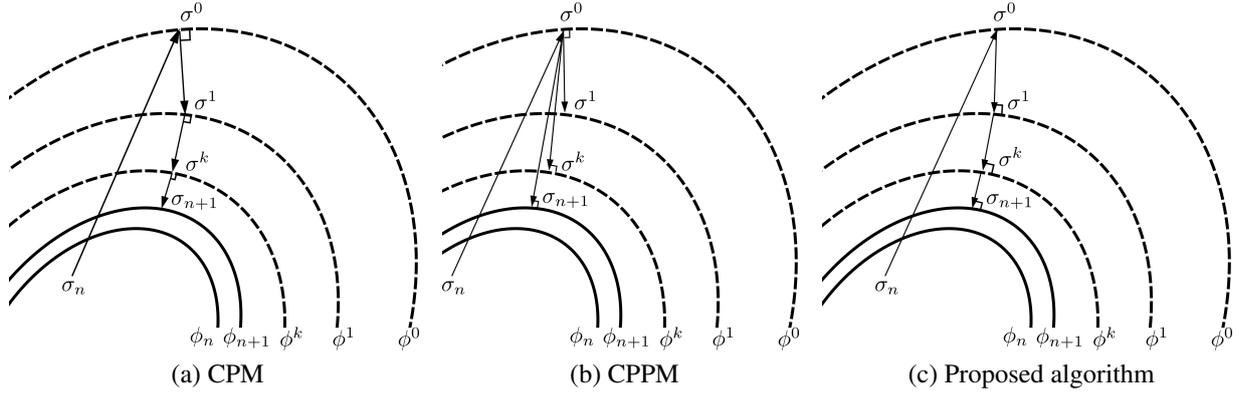\tag{3.2}
$$

Accordingly, $\delta\lambda$ is determined as

$$
\delta\lambda = \left[\left.\frac{\partial\phi}{\partial\boldsymbol{\sigma}}\right|^k : \mathbf{C} : \boldsymbol{m}^k - \left.\frac{\partial\phi}{\partial\boldsymbol{\kappa}}\right|^k : \left.\frac{\partial\boldsymbol{\kappa}}{\partial\Delta\lambda}\right|^k\right]^{-1} \phi^k
\tag{3.3}
$$

and the state variables are updated as

$$
\boldsymbol{\sigma}^{k+1} = \boldsymbol{\sigma}^k + \left.\frac{\partial\boldsymbol{\sigma}}{\partial\Delta\lambda}\right|^k \delta\lambda
$$
$$
\boldsymbol{\kappa}^{k+1} = \boldsymbol{\kappa}^k + \left.\frac{\partial\boldsymbol{\kappa}}{\partial\Delta\lambda}\right|^k \delta\lambda.
\tag{3.4}
$$

The above two steps are repeated until a suitable convergence criteria are met. Figure 3.1a depicts the stress return (mapping) during local (material level) iterations when CPM is used for numerical integration.

(a) CPM	(b) CPPM	(c) Proposed algorithm

Figure 3.1: Stress return mapping during CPM, CPPM and the numerical integration

The continuum tangent operator, which is generally used with CPM [4], is given by,

$$\mathbf{C}^{ep} = \mathbf{C} - \frac{(\mathbf{C} : \boldsymbol{m}) \otimes (\boldsymbol{n} : \mathbf{C})}{\boldsymbol{n} : \mathbf{C} : \boldsymbol{m} + \mathrm{K}^{\mathrm{p}}}, \tag{3.5}$$

where $\boldsymbol{n} = \frac{\partial \phi}{\partial \boldsymbol{\sigma}}$ and $\mathrm{K}^{\mathrm{p}} = -\frac{\partial \phi}{\partial \boldsymbol{\kappa}} : \frac{\partial \boldsymbol{\kappa}}{\partial \Delta \lambda}$. A consistent tangent operator was introduced later to CPM by Starmen et al. [8] which is not considered in this paper.

3.2 Closest Point Projection Method (CPPM)

The earliest ideas pertaining to CPPM were suggested by Wilkins [2] for von Misses plasticity. Since then, various extensions such as application to linear isotropic and kinematic hardening [9], nonlinear hardening [10] have been added to CPPM. CPPM is well reputed for its accuracy, robustness and stability [4]. In contrast to CPM, CPPM regards the $\boldsymbol{\sigma}$, $\boldsymbol{\kappa}$ and $\Delta \lambda$ as independent variables.

CPPM is based on the following approximations for the rate forms given by Eqs. 2.3, and 2.4.

$$\boldsymbol{\sigma}^{k+1} = \begin{cases} \boldsymbol{\sigma}_e = \boldsymbol{\sigma}_n + \mathbf{C} : \Delta \boldsymbol{\varepsilon} & \text{if } k = 0 \\ \boldsymbol{\sigma}_e - \Delta \lambda^{k+1} \mathbf{C} : \boldsymbol{m}^{k+1} & \text{if } k > 0 \end{cases}$$

$$\boldsymbol{\kappa}^{k+1} = \begin{cases} \boldsymbol{\kappa}_n & \text{if } k = 0 \\ \boldsymbol{\kappa}_n + \mathbf{A} \left(\boldsymbol{\sigma}^{k+1}, \boldsymbol{\kappa}^{k+1}, \Delta \lambda^{k+1} \right) & \text{if } k > 0 \end{cases} \tag{3.6}$$

The above expressions are nonlinear since the right hand sides are expressed in terms of the unknown state variables $\boldsymbol{\sigma}^{k+1}$, $\boldsymbol{\kappa}^{k+1}$ and $\Delta \lambda^{k+1}$. CPPM obtains an iterative scheme to solve these nonlinear equations based on the following residuals.

$$\begin{aligned} \mathbf{r}_{\boldsymbol{\sigma}}^{k+1} &= \boldsymbol{\sigma}^{k+1} - \left[\boldsymbol{\sigma}_e - \Delta \lambda^{k+1} \mathbf{C} : \boldsymbol{m}^{k+1} \right] \\ \boldsymbol{r}_{\boldsymbol{\kappa}}{}^{k+1} &= \boldsymbol{\kappa}^{k+1} - \left[\boldsymbol{\kappa}_n + \mathbf{A} \left(\boldsymbol{\sigma}^{k+1}, \boldsymbol{\kappa}^{k+1}, \Delta \lambda^{k+1} \right) \right] \\ r_{\phi}^{k+1} &= \phi \left(\boldsymbol{\sigma}^{k+1}, \boldsymbol{\kappa}^{k+1}, \Delta \lambda^{k+1} \right) \end{aligned} \tag{3.7}$$

Taking Taylor expansion about the solution at k^{th} iteration, ignoring higher order terms, and setting the residuals $\mathbf{r}_{(.)} \left(\boldsymbol{\sigma}^{k+1}, \boldsymbol{\kappa}^{k+1}, \Delta \lambda^{k+1} \right) = \mathbf{0}$, we can obtain the following linear set of equations for $\boldsymbol{\delta\sigma}$, $\boldsymbol{\delta\kappa}$, and $\delta\lambda$, which are the incremental updates of $\boldsymbol{\sigma}$, $\boldsymbol{\kappa}$, and $\Delta \lambda$, respectively.

Note that we drop the superscripts and subscripts for convenience, and all the terms in the right hand sides are evaluated at the solution of kth iteration. Here, $\mathbf{I}^{sym}$ is the fourth order major symmetric identity tensor.

$$
\left\{
\begin{array}{c}
\boldsymbol{\delta\sigma} \\
\boldsymbol{\delta\kappa} \\
\delta\lambda
\end{array}
\right\}
= -
\left[
\begin{array}{ccc}
\mathbf{I}^{sym} + \Delta\lambda\mathbf{C} : \frac{\partial\boldsymbol{m}}{\partial\boldsymbol{\sigma}} & \Delta\lambda\mathbf{C} : \frac{\partial\boldsymbol{m}}{\partial\boldsymbol{\kappa}} & \mathbf{C} : \boldsymbol{m} \\
-\frac{\partial\mathbf{A}}{\partial\boldsymbol{\sigma}} & \mathbf{I}^{sym} - \frac{\partial\mathbf{A}}{\partial\boldsymbol{\kappa}} & -\frac{\partial\mathbf{A}}{\partial\Delta\lambda} \\
\frac{\partial\phi}{\partial\boldsymbol{\sigma}} & \frac{\partial\phi}{\partial\boldsymbol{\kappa}} & 0
\end{array}
\right]^{-1}
\left\{
\begin{array}{c}
\mathbf{r}_{\boldsymbol{\sigma}}^{k} \\
\mathbf{r}_{\boldsymbol{\kappa}}^{k} \\
r_{\phi}^{k}
\end{array}
\right\}
\tag{3.8}
$$

Solving the above, we can incrementally update the state variables $\boldsymbol{\sigma}^{k+1}$, $\boldsymbol{\kappa}^{k+1}$ and $\Delta\lambda^{k+1}$ as follows until requisite convergence criteria are met. Figure 3.1b depicts how the stress is updated by the CPPM's return mapping iterations.

$$
\begin{aligned}
\boldsymbol{\sigma}^{k+1} &= \boldsymbol{\sigma}^{k} + \boldsymbol{\delta\sigma} \\
\boldsymbol{\kappa}^{k+1} &= \boldsymbol{\kappa}^{k} + \boldsymbol{\delta\kappa} \\
\Delta\lambda^{k+1} &= \Delta\lambda^{k} + \delta\lambda
\end{aligned}
\tag{3.9}
$$

Differentiating Eq. (3.6) and the yield criterion with respect to $\boldsymbol{\Delta\varepsilon}$, we can obtain

$$
\left[
\begin{array}{c}
\frac{\partial\boldsymbol{\sigma}}{\partial\boldsymbol{\Delta\varepsilon}} \\
\frac{\partial\boldsymbol{\kappa}}{\partial\boldsymbol{\Delta\varepsilon}} \\
\frac{\partial\Delta\lambda}{\partial\boldsymbol{\Delta\varepsilon}}
\end{array}
\right]
=
\left[
\begin{array}{ccc}
\mathbf{I}^{sym} + \Delta\lambda\mathbf{C} : \frac{\partial\boldsymbol{m}}{\partial\boldsymbol{\sigma}} & \Delta\lambda\mathbf{C} : \frac{\partial\boldsymbol{m}}{\partial\boldsymbol{\kappa}} & \mathbf{C} : \boldsymbol{m} \\
-\frac{\partial\mathbf{A}}{\partial\boldsymbol{\sigma}} & \mathbf{I}^{sym} - \frac{\partial\mathbf{A}}{\partial\boldsymbol{\kappa}} & -\frac{\partial\mathbf{A}}{\partial\Delta\lambda} \\
\frac{\partial\phi}{\partial\boldsymbol{\sigma}} & \frac{\partial\phi}{\partial\boldsymbol{\kappa}} & 0
\end{array}
\right]^{-1}
\left[
\begin{array}{c}
\mathbf{C} \\
\mathbf{0} \\
\mathbf{0}
\end{array}
\right] .
\tag{3.10}
$$

The consistent tangent operator $\frac{\partial\boldsymbol{\sigma}}{\partial\boldsymbol{\Delta\varepsilon}}$ for CPPM can be found by solving the above at the converged state variables.

3.3 *Proposed Method*

As explained above, CPPM treats $\boldsymbol{\sigma}$, $\boldsymbol{\kappa}$ and $\Delta\lambda$ as independent variables, while CPM treats only $\Delta\lambda$ as the independent variable. Both the methods express ϕ as a function of the corresponding independent state variables at the $(k+1)^{th}$ iteration, which are unknown. To solve the resulting nonlinear equations, both the methods linearize sufficient number of fundamental expressions. CPPM consisting of several independent variables, obtains three sets of linear equations by linearizing the residues $\boldsymbol{r}_{\boldsymbol{\sigma}}$ and $\boldsymbol{r}_{\boldsymbol{\kappa}}$ given by Eq. 3.7 and ϕ, about $\boldsymbol{\sigma}^{k}$, $\boldsymbol{\kappa}^{k}$. On the other hand, since CPM has only one variable, $\Delta\lambda$, linearization of only ϕ about $\boldsymbol{\sigma}^{k}$, $\boldsymbol{\kappa}^{k}$ produces sufficient number of equations. CPM is known to be less stable, compared to the unconditionally stable CPPM. While CPM uses a single constraint (equation) in the stress integration, the proposed method imposes equal number of constraints to that of CPPM on the state variables and the yield criterion by way of evaluating $\frac{\partial\boldsymbol{\sigma}}{\partial\Delta\lambda}$ and $\frac{\partial\boldsymbol{\kappa}}{\partial\Delta\lambda}$ of Eq. 3.2 at the unknown $(k+1)^{th}$ state, in addition to $\frac{\partial\phi}{\partial\Delta\lambda}$, thereby increasing the number of quantities evaluated at the unknown $(k+1)^{th}$ state to that of CPM.

The proposed algorithm relies on the fact that the rates of stress ($\dot{\boldsymbol{\sigma}}$), back stress ($\dot{\boldsymbol{\alpha}}$) and plastic strain ($\dot{\boldsymbol{\varepsilon}}^{p}$) can be expressed as a function of $\dot{\lambda}$ during plastic deformation. Expressing $\dot{\boldsymbol{\sigma}}$, $\dot{\boldsymbol{\alpha}}$ and $\dot{\boldsymbol{\varepsilon}}^{p}$ as a function of $\dot{\lambda}$, we can obtain the following incremental forms of Eqs. (2.3), and (2.4).

$$\boldsymbol{\sigma}^{k+1} = \begin{cases} \boldsymbol{\sigma}_e = \boldsymbol{\sigma}_n + \mathbf{C} : \Delta\boldsymbol{\varepsilon} & \text{if } k = 0 \\ \boldsymbol{\sigma}^k - \Delta\lambda\mathbf{C} : \boldsymbol{m}^{k+1} & \text{if } k > 0 \end{cases}$$

$$\boldsymbol{\kappa}^{k+1} = \begin{cases} \boldsymbol{\kappa}_n & \text{if } k = 0 \\ \boldsymbol{\kappa}^k + \mathbf{A}\left(\boldsymbol{\sigma}^{k+1}\left(\Delta\lambda\right), \boldsymbol{\kappa}^{k+1}\left(\Delta\lambda\right)\right) & \text{if } k > 0 \end{cases}$$

(3.11)

Using Taylor expansion, we obtain a first order approximation about the current state, $(.)^k$ for the yield function $\phi\left(\boldsymbol{\sigma}^{k+1}\left(\Delta\lambda\right), \boldsymbol{\kappa}^{k+1}\left(\Delta\lambda\right)\right) = 0$ as

$$\phi^{k+1} \approx \phi^k + \frac{\partial\phi}{\partial\Delta\lambda}\delta\lambda$$

$$0 = \phi^k + \left(\left.\frac{\partial\phi}{\partial\boldsymbol{\sigma}}\right|^k : \frac{\partial\boldsymbol{\sigma}}{\partial\Delta\lambda} + \left.\frac{\partial\phi}{\partial\boldsymbol{\kappa}}\right|^k : \frac{\partial\boldsymbol{\kappa}}{\partial\Delta\lambda}\right)\delta\lambda.$$

(3.12)

The above expression for $\frac{\partial\phi}{\partial\Delta\lambda}$ and the partial differentiation of Eqs. (3.11) and (3.12) with respect to $\Delta\lambda$ provide the following linear set of equations which can be solved for the unknowns $\frac{\partial\boldsymbol{\sigma}}{\partial\Delta\lambda}$, $\frac{\partial\boldsymbol{\kappa}}{\partial\Delta\lambda}$ and $\frac{\partial\phi}{\partial\Delta\lambda}$.

$$\begin{bmatrix} \frac{\partial\boldsymbol{\sigma}}{\partial\Delta\lambda} \\ \frac{\partial\boldsymbol{\kappa}}{\partial\Delta\lambda} \\ \frac{\partial\phi}{\partial\Delta\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{I}^{sym} + \Delta\lambda\mathbf{C} : \frac{\partial\boldsymbol{m}}{\partial\boldsymbol{\sigma}} & \Delta\lambda\mathbf{C} : \frac{\partial\boldsymbol{m}}{\partial\boldsymbol{\kappa}} & 0 \\ -\frac{\partial\mathbf{A}}{\partial\boldsymbol{\sigma}} & \mathbf{I}^{sym} - \frac{\partial\mathbf{A}}{\partial\boldsymbol{\kappa}} & 0 \\ \frac{\partial\phi}{\partial\boldsymbol{\sigma}} & \frac{\partial\phi}{\partial\boldsymbol{\kappa}} & -1 \end{bmatrix}^{-1} \begin{bmatrix} -\mathbf{C} : \boldsymbol{m} \\ \frac{\partial\mathbf{A}}{\partial\Delta\lambda} \\ 0 \end{bmatrix}$$

(3.13)

$$\delta\lambda = -\left(\frac{\partial\phi}{\partial\Delta\lambda}\right)^{-1}\phi^k$$

(3.14)

Once $\frac{\partial\phi}{\partial\Delta\lambda}$ is found, $\delta\lambda$ can be found using Eq. (3.14), and the state variables can be updated as

$$\boldsymbol{\sigma}^{k+1} = \boldsymbol{\sigma}^k + \frac{\partial\boldsymbol{\sigma}}{\partial\Delta\lambda}\delta\lambda$$

$$\boldsymbol{\kappa}^{k+1} = \boldsymbol{\kappa}^k + \frac{\partial\boldsymbol{\kappa}}{\partial\Delta\lambda}\delta\lambda$$

$$\Delta\lambda = 0 + \delta\lambda = \delta\lambda,$$

(3.15)

until suitable convergence criteria are met.

Comparison of Eqs. 3.8 and 3.13 shows that the components in their right hand sides are identical, except the CPPM's residuals. In that respect, each material-level iteration of CPPM and the proposed method requires identical computational effort. The properties of the last column and row of the Eq. (3.13) allow us to uncouple and solve the linear system as two independent systems for $\left\{\begin{array}{cc} \frac{\partial\boldsymbol{\sigma}}{\partial\Delta\lambda} & \frac{\partial\boldsymbol{\kappa}}{\partial\Delta\lambda} \end{array}\right\}^T$ and $\frac{\partial\phi}{\partial\Delta\lambda}$, which slightly reduces the computational effort compared to CPPM.

The novel proposed algorithm preserves the characteristics of CPM that it successively linearizes the yield function at the current state to first estimate plastic consistency parameter using the derivatives of the state variables with respect to the plastic consistency parameter and then update the state variables. A pseudo code for the proposed algorithm is given in Algorithm 1, and Fig. 3.1c depicts the updating of stress during the return mapping iterations of the proposed algorithm.

Algorithm 1: A pseudo code of the proposed algorithm. $\eta_{(.)}$ is a suitable small number to check the convergence of the quantity $(.)$.

---

input : $\boldsymbol{\Delta\varepsilon}_{n+1}$

output: $\boldsymbol{\Delta\varepsilon}_{n+1}^e$: elastic portion of $\boldsymbol{\Delta\varepsilon}_{n+1}$

$\quad\quad\quad$ $\boldsymbol{\kappa}_{n+1}$: plastic internal variables

$\quad\quad\quad$ $\boldsymbol{\sigma}_{n+1}$: state of stress after $(n+1)^{th}$ global-iteration

$\quad\quad\quad$ $\left.\frac{\partial\boldsymbol{\sigma}}{\partial\boldsymbol{\Delta\varepsilon}}\right|_{n+1}$: consistent tangent operator

// Elastic predictor
$\boldsymbol{\Delta\varepsilon}_{n+1}^e = \boldsymbol{\Delta\varepsilon}_{n+1}; \quad \boldsymbol{\sigma}^0 = \boldsymbol{\sigma}_n + \mathbf{C} : \boldsymbol{\Delta\varepsilon}_{n+1}^e; \quad \boldsymbol{\kappa}^0 = \boldsymbol{\kappa}_n;$

if $(\phi\left(\boldsymbol{\sigma}^0, \boldsymbol{\kappa}^0\right) < 0)$ **then** // Check whether yielded
$\quad$ $\boldsymbol{\sigma}_{n+1} = \boldsymbol{\sigma}^0; \quad \boldsymbol{\kappa}_{n+1} = \boldsymbol{\kappa}^0; \quad \boldsymbol{\Delta\varepsilon}_{n+1}^e = \boldsymbol{\Delta\varepsilon}_{n+1};$
$\quad$ **return**;
else
$\quad$ // Initialize state variable for the iteration $k = 1$
$\quad$ $\frac{\partial\boldsymbol{\sigma}}{\partial\Delta\lambda} = -\mathbf{C} : \boldsymbol{m}^0; \quad\quad \frac{\partial\boldsymbol{\kappa}}{\partial\Delta\lambda} = \left.\frac{\partial\mathbf{A}}{\partial\Delta\lambda}\right|^0;$
$\quad$ $\delta\lambda = \left(\left.\frac{\partial\phi}{\partial\boldsymbol{\sigma}}\right|^0 : \frac{\partial\boldsymbol{\sigma}}{\partial\Delta\lambda} - \left.\frac{\partial\phi}{\partial\boldsymbol{\kappa}}\right|^0 : \frac{\partial\mathbf{A}}{\partial\Delta\lambda}\right)^{-1} \phi\left(\boldsymbol{\sigma}^0, \boldsymbol{\kappa}^0\right);$
$\quad$ $\boldsymbol{\sigma}^1 = \boldsymbol{\sigma}^0 + \frac{\partial\boldsymbol{\sigma}}{\partial\Delta\lambda}\delta\lambda; \quad \boldsymbol{\kappa}^1 = \boldsymbol{\kappa}^0 + \frac{\partial\boldsymbol{\kappa}}{\partial\Delta\lambda}\delta\lambda; \quad \Delta\lambda = 0 + \delta\lambda = \delta\lambda;$
$\quad$ $k = 1;$

$\quad$ // Successively linearize $\phi^{k+1} = \phi(\boldsymbol{\sigma}^{k+1}, \boldsymbol{\kappa}^{k+1})$ and update the state variables
$\quad$ **do**
$\quad\quad$ $\begin{bmatrix} \frac{\partial\boldsymbol{\sigma}}{\partial\Delta\lambda} \\ \frac{\partial\boldsymbol{\kappa}}{\partial\Delta\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{I}^{sym} + \Delta\lambda\mathbf{C} : \frac{\partial\boldsymbol{m}}{\partial\boldsymbol{\sigma}} & \Delta\lambda\mathbf{C} : \frac{\partial\boldsymbol{m}}{\partial\boldsymbol{\kappa}} \\ -\frac{\partial\mathbf{A}}{\partial\boldsymbol{\sigma}} & \mathbf{I}^{sym} - \frac{\partial\mathbf{A}}{\partial\boldsymbol{\kappa}} \end{bmatrix}^{-1} \begin{bmatrix} -\mathbf{C} : \boldsymbol{m} \\ \frac{\partial\mathbf{A}}{\partial\Delta\lambda} \end{bmatrix};$
$\quad\quad$ $\delta\lambda = -\left(\left.\frac{\partial\phi}{\partial\boldsymbol{\sigma}}\right|^k : \frac{\partial\boldsymbol{\sigma}}{\partial\Delta\lambda} + \left.\frac{\partial\phi}{\partial\boldsymbol{\kappa}}\right|^k : \frac{\partial\boldsymbol{\kappa}}{\partial\Delta\lambda}\right)^{-1} \phi^k;$

$\quad\quad$ // Update the state variables
$\quad\quad$ $\boldsymbol{\sigma}^{k+1} = \boldsymbol{\sigma}^k + \delta\boldsymbol{\sigma}$; where $\delta\boldsymbol{\sigma} = \frac{\partial\boldsymbol{\sigma}}{\partial\Delta\lambda}\delta\lambda;$
$\quad\quad$ $\boldsymbol{\kappa}^{k+1} = \boldsymbol{\kappa}^k + \delta\boldsymbol{\kappa}$; where $\delta\boldsymbol{\kappa} = \frac{\partial\boldsymbol{\kappa}}{\partial\Delta\lambda}\delta\lambda;$
$\quad\quad$ $\Delta\lambda = \delta\lambda;$
$\quad\quad$ $k = k + 1;$ // increment the iteration index
$\quad$ **while** $\left(\left(\phi^k \leq \eta_\phi\right) \textbf{ or } (\delta\boldsymbol{\sigma} < \eta_{\boldsymbol{\sigma}}) \textbf{ or } (\delta\boldsymbol{\kappa} < \eta_{\boldsymbol{\kappa}})\right)$

$\quad$ $\boldsymbol{\sigma}_{n+1} = \boldsymbol{\sigma}^k; \quad \boldsymbol{\kappa}_{n+1} = \boldsymbol{\kappa}^k;$
$\quad$ **return**;

Consistent Tangent Operator

By differentiating the set of equations given in Eq. (3.11) and the yield criterion with respect to $\Delta\varepsilon$, we can obtain,

$$
\begin{bmatrix}
\frac{\partial\boldsymbol{\sigma}}{\partial\boldsymbol{\Delta\varepsilon}}\Big|^{k+1} \\
\frac{\partial\boldsymbol{\kappa}}{\partial\boldsymbol{\Delta\varepsilon}}\Big|^{k+1} \\
\frac{\partial\Delta\lambda}{\partial\boldsymbol{\Delta\varepsilon}}\Big|^{k+1}
\end{bmatrix}
=
\begin{bmatrix}
\mathbf{I}^{sym} + \Delta\lambda\mathbf{C}:\frac{\partial\boldsymbol{m}}{\partial\boldsymbol{\sigma}} & \Delta\lambda\mathbf{C}:\frac{\partial\boldsymbol{m}}{\partial\boldsymbol{\kappa}} & \mathbf{C}:\boldsymbol{m} \\
-\frac{\partial\mathbf{A}}{\partial\boldsymbol{\sigma}} & \mathbf{I}^{sym} - \frac{\partial\mathbf{A}}{\partial\boldsymbol{\kappa}} & -\frac{\partial\mathbf{A}}{\partial\Delta\lambda} \\
\frac{\partial\phi}{\partial\boldsymbol{\sigma}} & \frac{\partial\phi}{\partial\boldsymbol{\kappa}} & 0
\end{bmatrix}^{-1}
\begin{bmatrix}
\frac{\partial\boldsymbol{\sigma}}{\partial\boldsymbol{\Delta\varepsilon}}\Big|^{k} \\
\frac{\partial\boldsymbol{\kappa}}{\partial\boldsymbol{\Delta\varepsilon}}\Big|^{k} \\
\mathbf{0}
\end{bmatrix}
\tag{3.16}
$$

The components of the inverted matrix in the right hand side are also evaluated using the state variables obtained at the end of the $(k+1)^{th}$ iteration (i.e., the latest completed material-level iteration). The above recursive relation is repeatedly applied at the end of each material-level iteration, and the $\frac{\partial\boldsymbol{\sigma}}{\partial\boldsymbol{\Delta\varepsilon}}$ obtained at the end of converged material-level iteration is the consistent tangent operator of the proposed algorithm.

While CPPM requires solving Eq. (3.10) only once at end of converged material-level iterations, the proposed requires recursively solving Eq. (3.16) at the end of each material-level iteration. This extra amount of computational effort is one major disadvantage of the proposed method compared to CPPM. As it will be demonstrated in the next section, both the CPPM and the proposed methods require the same number of global-level iterations indicating that the consistent tangent operators of both the methods perform equally.

3.4   Comparison of the Proposed Algorithm with CPPM and CPM

From the formulation of the proposed integration scheme, we can establish the main features of the proposed integration scheme as,

- All three stress integration algorithms are implicit in the sense that in all three algorithms the unknown variables are evaluated at the unknown state.

- Uses satisfaction of implicit constitutive relations (Eq. (3.11) and $\phi^{k+1}=0$) to arrive at an estimate for the plastic consistency parameter in contrast to the residual based approach used in CPPM. Successive linearization of the yield function around the current state is used to estimate the plastic consistency parameter using the derivatives of the state variables with respect to the plastic consistency parameter.

- Unlike in CPPM, the plastic consistency parameter is not continuously updated in the proposed method. Like in CPM, the plastic consistency parameter is found at each iteration separately and is not carried to the next iteration by additive updates.

- Like in CPPM, the first and second order derivatives of the yield surface and the plastic potential surface are used during the stress integration whereas in CPM only the first order derivatives are used.

- A consistent tangent operator is available for global iterations which has to be updated iteratively unlike in CPPM where the consistent tangent operator is evaluated explicitly at the end of successful convergence of the state variables. Therefore, the evaluation of the consistent tangent operator in the proposed method at the end of each material level iteration adds additional computational cost in comparison to CPPM.

It is evident that there are key distinguishable features that separate the proposed integration scheme from the veteran CPPM and CPM integration schemes. Further investigations are necessary to establish the numerical stability and usability of the proposed scheme for generalized plasticity models. In this paper, we consider the application of the proposed integration scheme for limited use in the von Misses model.

4 Verification tests

In this section, the accuracy and convergence behaviour of the algorithm is assessed and compared against CPPM and CPM, with two tests conducted using von Mises yield criterion. The following form of the von Mises yield function is used in all of the numerical simulations presented in this section. Note that the Frobenius norm ($\|.\|_{\mathcal{F}}$) of a second order arbitrary tensor, $\mathbf{A}$, is $\|\mathbf{A}\|_{\mathcal{F}} = \sqrt{\mathbf{A} : \mathbf{A}}$.

$$\phi = \sqrt{\frac{3}{2}} \|\boldsymbol{s} - \boldsymbol{\alpha}\|_{\mathcal{F}} - (\sigma_{y,0} + A_I e^p) \tag{4.1}$$

Here, $\boldsymbol{\alpha}$ is the back stress defined by Eq. (4.2) following Ziegler's rule [11, 12],

$$\dot{\boldsymbol{\alpha}} = A_k (\boldsymbol{\sigma}, \boldsymbol{\alpha}) \dot{e}^p (\boldsymbol{\sigma} - \boldsymbol{\alpha}) \tag{4.2}$$

$A_K, \sigma_{y,0}$, and A_I are kinematic hardening modulus (constant), initial yield stress, and isotropic hardening modulus (constant). By virtue of setting different values for A_K and A_I, linear kinematic hardening ($A_I = 0$), linear isotropic hardening ($A_K = 0$), and combined hardening ($A_I, A_K \neq 0$) phenomena could be simulated. e^p is the effective plastic strain which is a stress integration algorithm dependent quantity and is defined as follows (here, $\boldsymbol{e}^p$ is the deviatoric part of plastic strain),

$$e^p = \int_0^t \sqrt{\frac{2}{3}} \|\dot{\boldsymbol{e}}^p\|_{\mathcal{F}} \, dt \tag{4.3}$$

For the proposed integration method, it follows from the incremental form representation of plastic strain, $\boldsymbol{\varepsilon}^{p,k+1} = \boldsymbol{\varepsilon}^p + \Delta\lambda \boldsymbol{m}^{k+1}$, that (Here, $\boldsymbol{d}$ is the deviatoric part of $\boldsymbol{m}$),

$$e^{p,k+1} = \begin{cases} e_n^p & \text{if } k = 0 \\ e_n^p + \sqrt{\frac{2}{3}} \left\| \boldsymbol{e}^{p,k} + \Delta\lambda \boldsymbol{d}^{k+1} \right\|_{\mathcal{F}} & \text{if } k > 0 \end{cases} \tag{4.4}$$

Furthermore, we use the following incremental forms for back stress in each of the stress integration methods considered,

$$\boldsymbol{\alpha}^{k+1} = \begin{cases} \boldsymbol{\alpha}_n & \text{if } k = 0 \\ \boldsymbol{\alpha}^k + \frac{\Delta\lambda A_K}{\sigma_e} \sqrt{\frac{2}{3}} \left\| \boldsymbol{d}^k \right\|_{\mathcal{F}} \mathbf{P}^{sd} : \left(\boldsymbol{\sigma}^k - \boldsymbol{\alpha}^k \right) & \text{if } k > 0, \end{cases} \quad \text{CPM} \tag{4.5}$$

$$\boldsymbol{\alpha}^{k+1} = \begin{cases} \boldsymbol{\alpha}_n & \text{if } k = 0 \\ \boldsymbol{\alpha}^k + \frac{\Delta\lambda^{k+1} A_K}{\sigma_e} \sqrt{\frac{2}{3}} \left\| \boldsymbol{d}^{k+1} \right\|_{\mathcal{F}} \mathbf{P}^{sd} : \left(\boldsymbol{\sigma}^{k+1} - \boldsymbol{\alpha}^{k+1} \right) & \text{if } k > 0, \end{cases} \quad \text{CPPM} \tag{4.6}$$

$$\boldsymbol{\alpha}^{k+1} = \begin{cases} \boldsymbol{\alpha}_n & \text{if } k = 0 \\ \frac{A_K}{\sigma_e} \sqrt{\frac{2}{3}} \left\| \boldsymbol{e}^{p,k} + \Delta\lambda \boldsymbol{d}^{k+1} \right\|_{\mathcal{F}} \mathbf{P}^{sd} : \left(\boldsymbol{\sigma}^{k+1} - \boldsymbol{\alpha}^{k+1} \right) & \text{if } k > 0, \end{cases} \quad \text{Proposed}$$

$$\tag{4.7}$$

Here, $\mathbf{P}^{sd} = \mathbf{I}^{sym} - \frac{1}{3}\mathbf{I} \otimes \mathbf{I}$ is the fourth order isotropic tensor that converts any second-order tensor into its symmetric-deviator form [13] and $\sigma_e = \sqrt{\frac{3}{2}} \|\boldsymbol{s} - \boldsymbol{\alpha}\|_{\mathcal{F}}$.

Numerical tests are conducted for the following cases,

1. Single material point.

2. Uniaxial extension of a perforated sheet.

using all three stress integration algorithms CPPM, CPM, and the proposed method. The algorithms were implemented in C++ with the matrix operations undertaken using the Eigen library[14], a software library written in C++ for matrix computations.

4.1 Single material point

The accuracy of the proposed method is demonstrated using the semi-analytical solutions provided in Anandarajah [4] and Kim [15] for two problems, i.e., linear isotropic hardening and combined hardening respectively. The two sets of material parameters used in the respective problems are (Here, E is the modulus of elasticity and ν is the Poisson's ratio),

1. Material 1: $E = 200$ GPa; $\nu = 0.3$; $A_I = 20$ GPa; $A_K = 0$ MPa; $\sigma_{y,n} = 0.25$ GPa

2. Material 2: $E = 2.4$ GPa; $\nu = 0.2$; $A_I = 70$ MPa; $A_K = 30$ MPa; $\sigma_{y,n} = 300$ MPa

For the linear isotropic hardening problem in Anandarajah [4] with the initial state, $\boldsymbol{\sigma}_n = \left\{ \begin{array}{cccccc} 0.1 & 0.05 & 0.075 & 0 & 0 & 0 \end{array} \right\}^{\mathrm{T}}$ GPa and the applied strain increment, $\boldsymbol{\Delta\varepsilon} = \left\{ \begin{array}{cccc} 0.03 & -0.028 \\ 0.01 & 0 & 00 \end{array} \right\}^{\mathrm{T}}$, Table 1 provides the remapped stresses obtained from the three numerical integration schemes of interest. The results are compared against the semi-analytical solution provided in Anandarajah [4] and the converged results obtained using CPPM by applying subincrements (using 1024 subincrements of the strain increment). The relative error in Table 1, E_R, defined by,

$$E_R = \frac{\|\boldsymbol{\sigma}_{num} - \boldsymbol{\sigma}_{ref}\|}{\|\boldsymbol{\sigma}_{ref}\|} \tag{4.8}$$

where $\boldsymbol{\sigma}_{num}$ is the numerical integration result from the numerical integration without any subincrementation and $\boldsymbol{\sigma}_{ref}$ is the reference converged result obtained by applying CPPM using subincrementation [16]. From the results, it is clear that no significant disparity exists between the numerical integration results obtained from all three numerical integration schemes. Furthermore, all stress integration algorithms took only a single material point iteration to produce the following set of remapped results.

Table 1: Remapped stresses comparison - Linear Isotropic Hardening (Units are in GPa)

Stress components	Integration Scheme				
	Semi-Analytical	Subincrements	CPPM	CPM	Proposed
σ_{11}	2.51333	2.51297	2.51333	2.51333	2.51333
σ_{22}	1.53615	1.53596	1.53615	1.53615	1.53615
σ_{33}	2.17552	2.17607	2.17552	2.17552	2.17552
σ_{12}	0	0	0	0	0
σ_{13}	0	0	0	0	0
σ_{23}	0	0	0	0	0
E_R	0.000185	-	0.000185	0.000185	0.000185

Kim [13] provides a combined hardening problem with $\boldsymbol{\sigma}_n = \left\{ \begin{matrix} 300 & 0 & 0 & 0 & 0 & 0 \end{matrix} \right\}^{\mathrm{T}}$ MPa as the initial state and the applied strain increment, $\boldsymbol{\Delta \varepsilon} = \left\{ \begin{matrix} 0.1 & -0.02 & -0.02 & 0 & 0 & 0 \end{matrix} \right\}^{\mathrm{T}}$. Table 2 shows the remapped stresses obtained from the three stress integration algorithms in relation to the semi-analytical solution provided in Kim [13] and the converged results obtained using CPPM by applying subincrements (using 1024 subincrements of the strain increment). From the results, it is evident that all three stress integration algorithms provide identical estimates for the remapped stresses. As in the previous case, here also only a single material point iteration was utilized by each of the stress integration algorithms to produce the following remapped stresses.

Table 2: Remapped stresses comparison - Combined Hardening (Units are in MPa)

Stress components	Integration Scheme				
	Semi-Analytical	Subincrements	CPPM	CPM	Proposed
σ_{11}	385.16129	385.16129	386.65972	386.65972	386.65972
σ_{22}	77.41935	77.41935	76.67013	76.67013	76.67013
σ_{33}	77.41935	77.41935	76.67013	76.67013	76.67013
σ_{12}	0	0	0	0	0
σ_{13}	0	0	0	0	0
σ_{23}	0	0	0	0	0
E_R	3.37×10^{-14}	-	0.004583	0.004583	0.004583

4.2 *Uniaxial extension of a perforated sheet*

The overall performance of the proposed integration scheme (specifically the consistent tangent operator) is evaluated and compared with CPPM and CPM (with continuum tangent operator) on an elastoplastic homogeneous thin square shaped perforated sheet. The square sheet measures 20 mm a side, a thickness of 1 mm with a central circular hole of radius 1 mm. Considering the symmetry of the sheet, we model only a quarter of the sheet with the appropriate symmetric boundary conditions (Fig. 4.1a). A structured mesh with 1024×3 8-node brick elements and 4356 nodes is used.

The sheet is subjected to a uniform distributed load of magnitude 400 N/mm^2 applied perpendicular to the top edge as shown in Fig. 4.1a according to the cyclic loading history given in

Fig. 4.1b using load control. The total analysis time is 4.0 s with time step increments (Δt) of 0.1 s corresponding to 40 steps. We consider three three-dimensional problems using the same mesh and boundary conditions, i.e., linear isotropic hardening, linear kinematic hardening, and linear combined hardening. The set of material parameters used in the respective problems are as follows,

1. Material 3: $E = 206.9$ GPa; $\nu = 0.29$; $A_I = 10000$ MPa; $A_K = 0$ MPa; $\sigma_{y,0} = 450$ MPa

2. Material 4: $E = 206.9$ GPa; $\nu = 0.29$; $A_I = 0$ MPa; $A_K = 10000$ MPa; $\sigma_{y,0} = 450$ MPa

3. Material 5: $E = 206.9$ GPa; $\nu = 0.29$; $A_I = 5000$ MPa; $A_K = 5000$ MPa; $\sigma_{y,0} = 450$ MPa

(a) Mesh and boundary conditions   (b) Cyclic loading history Vs pseudo time
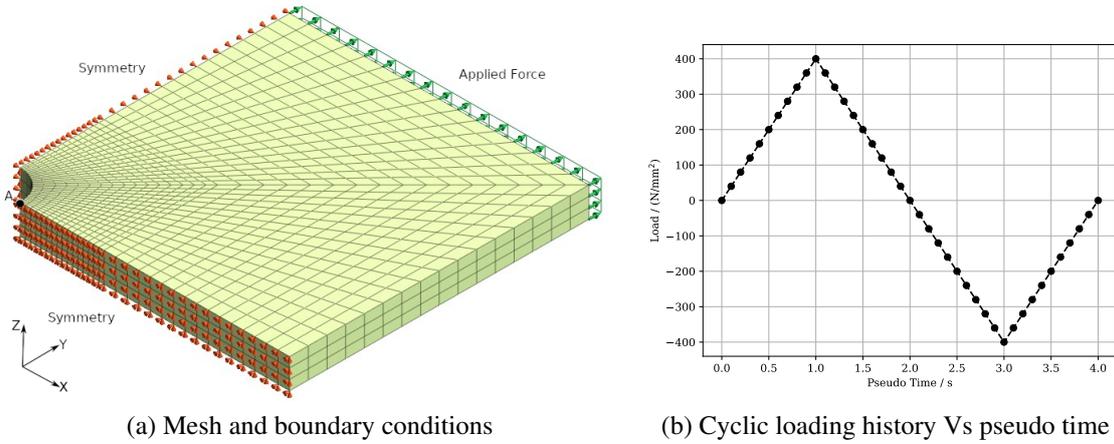
Figure 4.1: Three-dimensional perforated sheet.

We compare the three methods using the displacement measured at node A along the X - axis direction(u_x) during the loading and unloading cycles. We define time-step-wise percentage deviation ($E_{r,dev\%}$) metric as follows, which is used to illustrate the accuracy of a certain algorithm with respect to the results from another reference stress integration algorithm.

$$E_{r,dev\%} = \frac{|d_{method} - d_{ref}|}{|d_{ref}|_{max}} \times 100\% \tag{4.9}$$

Here, d_{method} is the displacement at a particular node obtained using the "method" (method could be CPM, CPPM or Proposed) stress integration algorithm at a particular pseudo time step where as d_{ref} is the displacement at the same node obtained using the reference method of stress integration at a particular pseudo time step. $|d_{ref}|_{max}$ refers to the maximum absolute value of displacement recorded at the same node obtained using the reference method from all the time steps. All three stress integration implementations use the same convergence criteria for global iterations and the local material point iterations.

4.2.1 Linear Isotropic Hardening and Linear Kinematic Hardening

Fig. 4.2a and Fig.4.3a depict the displacement vs load increment for linear isotropic hardening and linear kinematic hardening conducted using material 3 and material 4 set parameters respectively. All three stress integration methods give near identical results for both linear isotropic hardening and linear kinematic hardening cases. This is evident from Fig. 4.2b and Fig. 4.3b which show the percentage deviation with respect to results from CPPM for linear

isotropic hardening and linear kinematic hardening respectively. Fig. 4.2c and Fig.4.3c depict the number of iterations taken by each of the stress integration algorithms during global iterations for linear isotropic hardening and linear kinematic hardening respectively. As you can see, the the proposed scheme and CPPM have consumed the same number of global iterations (102 iterations) for linear isotropic hardening case. For linear kinematic hardening case both the proposed scheme and CPPM have utilized the same number of global iterations (136 iterations). The CPM takes a considerably large number of global iterations for linear isotropic hardening case (256 iterations) as well as for linear kinematic hardening case (351 iterations) to give the same comparable results.
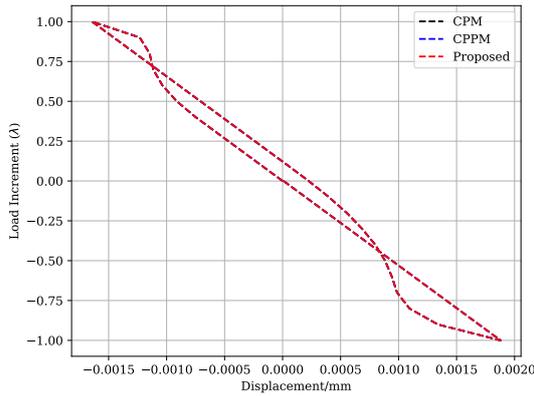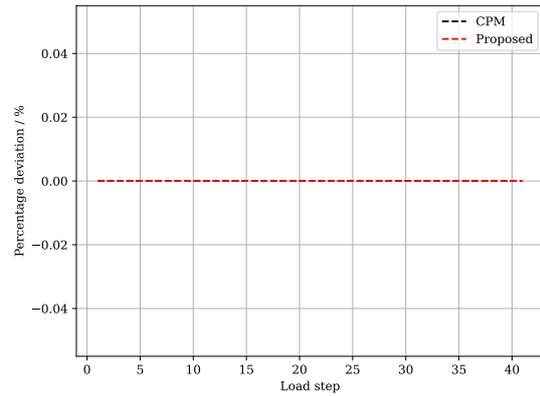
(a) Displacement vs load increment

(b) Percentage deviation with respect to results from CPPM ($\Delta t = 0.1s$)
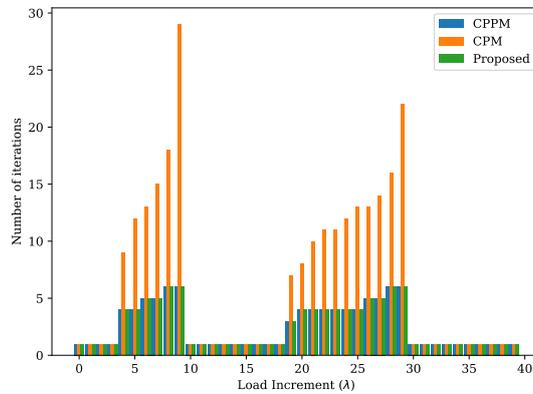
(c) Number of iterations taken by each algorithm at global-level

Figure 4.2: Perforated sheet - Isotropic hardening
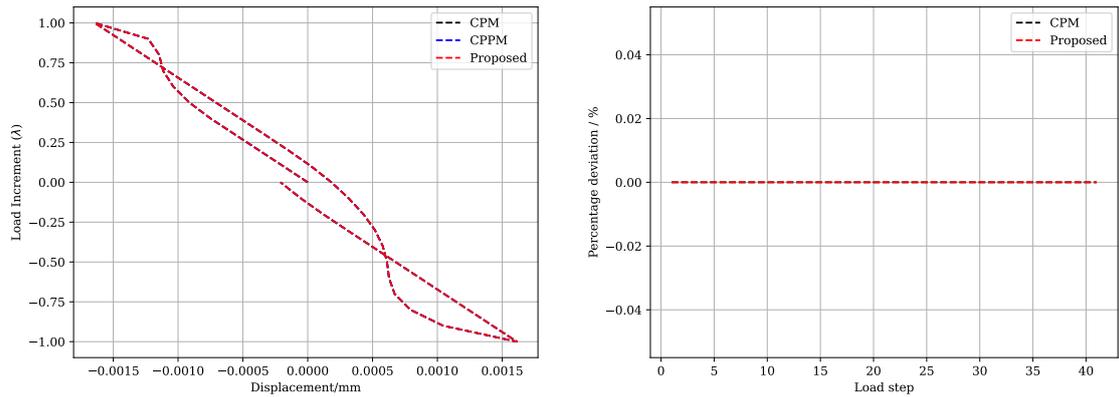
(a) Displacement vs load increment

(b) Percentage deviation with respect to results from CPPM ($\Delta t = 0.1s$)
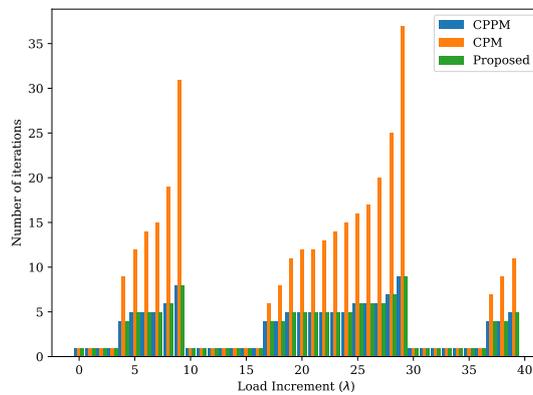
(c) Number of iterations taken by each algorithm at global-level

Figure 4.3: Perforated sheet - Kinematic hardening

4.2.2 Combined Hardening

The displacement vs load increment variation for combined hardening conducted using material 5 set parameters is shown in Fig. 4.4a. Here, we can observe that CPM and the proposed method follow near identical trajectories where as CPPM exhibits a significantly different trajectory after the 19th load step. This is evident from Fig. 4.4b which shows the percentage deviation with respect to results from CPPM. Fig. 4.4c exhibits the number of iterations taken by each of the stress integration algorithms during global iterations. Here, we can observe that the the proposed scheme and CPPM record the same number of global iterations (116 iterations) where as CPM consumes a considerably large number of global iterations (287 iterations) to give the same comparable results.
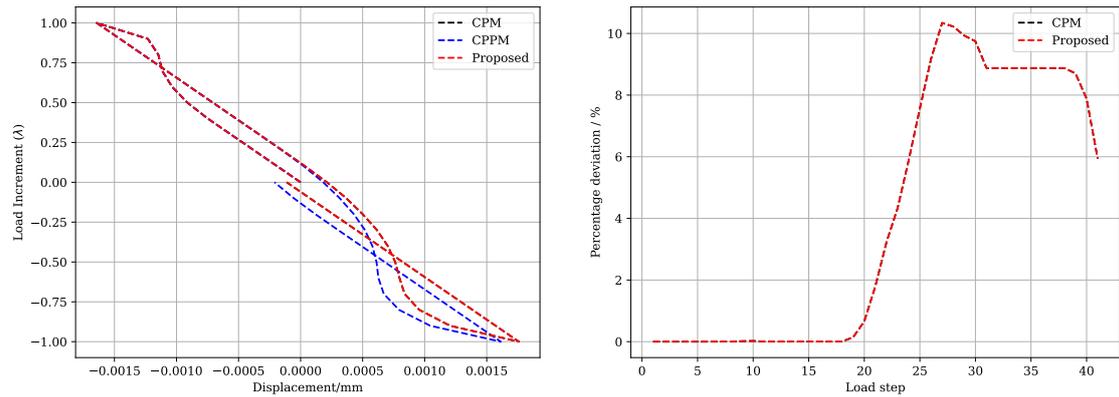
(a) Displacement vs load increment

(b) Percentage deviation with respect to results from CPPM ($\Delta t = 0.1s$)
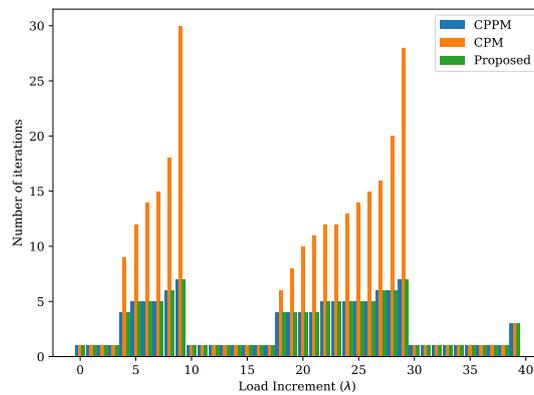
(c) Number of iterations taken by each algorithm at global-level

Figure 4.4: Perforated sheet - Combined hardening

5 Summary and Concluding Remarks

In this paper, a novel implicit stress integration algorithm which consists of some of the properties of CPPM and CPM is presented. In fact, the first and second order derivatives in the proposed stress are the same as that of CPPM (see Eq. (3.7) and Eq. (3.13)). The proposed implicit algorithm, uses satisfaction of implicit constitutive relations and successive linearization of the yield function around the current state to arrive at an estimate for the plastic consistency parameter required to update the state variables, in contrast to to the residual based approach used in CPPM. The successive linearization of the yield function about the current state to evaluate the state variables is a feature that the proposed stress integration scheme shares with CPM. However, the proposed method imposes equal number of constraints (equations) to that of CPPM by way of evaluating the derivatives of the state variables and the yield criterion with respect to the plastic consistency parameter in contrast to the single constraint (equation) used by CPM in the stress integration procedure. Further, unlike in CPM, it is straightforward to derive a consistent tangent operator for the proposed method.

Several verification tests are performed using the von Mises yield criterion to verify the proposed stress integration scheme and compare its performance in relation to CPPM and CPM. Single material point tests are performed to verify the accuracy of the stress integration procedure whereas the multi-element tests are carried out to verify and evaluate the performance of

the consistent tangent operator. In the context of von Misses model for material point iterations, the following can be inferred,

- the stress integration results of the proposed algorithm are on par with the results from CPPM and CPM for linear hardening rules.

- the computational cost associated with material point iterations (per iteration basis, when evaluation of the consistent tangent operator is excluded) is lowest for CPM and highest for CPPM where as the computational cost of the proposed scheme is marginally less than CPPM. Since, only first order derivatives are required for CPM, it has the lowest computational cost. In the proposed scheme the number of linear simultaneous equations that need to be solved per iteration is always one less than the number of linear simultaneous equations that need to be solved for CPPM. This reduces the computational cost of the proposed scheme marginally in comparison to CPPM.

In the context of von Misses model for global level iterations for linear hardening rules, the following can be inferred,

- the accuracy of the global response results are on par with CPPM with the consistent tangent operator and CPM with the continuum tangent operator.

- identical number of global level iterations to that of CPPM are required to obtain the converged solutions. This implies that the consistent tangent operator obtained form the proposed scheme is as good as the CPPM counterpart. CPM reporting the highest number of global level iterations could be attributed to using the continuum tangent operator.

- the total computational cost per material point evaluation which includes the cost associated with the stress integration as well as the consistent tangent operator evaluation, is lowest for CPM and highest for the proposed scheme due to the iterative nature of the consistent tangent operator of the proposed scheme. However, this disadvantage of solving for the small (at most 13×13 for isotropic material) consistent tangent operator several times during a material point iteration dwarfs in comparison to the advantage yielded through the use of a consistent tangent operater at the global iterations due to second order convergence.

From the results, it is evident that the proposed scheme is a viable alternative for elastoplastic stress integration of von Mises plasticity as it provides comparable results to that of CPPM and CPM. As the first and second order derivatives required in the stress integration procedure of the proposed method are the same as that of CPPM, one can easily implement the proposed method in existing finite element analysis frameworks. In the future, we plan to explore proposed methods performance in simulating complex palsticty models like Drucker-Prager and Cam-Clay.

Acknowledgments

References

[1] Scalet, G., Auricchio, F., (2018). Computational Methods for Elastoplasticity: An Overview of Conventional and Less-Conventional Approaches. *Arch Computat Methods Eng*, **25**, 545–589.

[2] Wilkins, M.L. (1964). Calculation of elasto-plastic flow. In Methods of Computational Physics 3, Academic, New York.

[3] Ortiz, M. and Simo, J.C. (1986). Analysis of a new class of integration algorithms for elastoplastic constitutive relations. *International Journal for Numerical Methods in Engineering*, **23**: 353–366.

[4] Anandarajah, A., (2010). Computational methods in elasticity and plasticity: solids and porous media. Springer, New York.

[5] Simo, J. C., and Hughes T. J. R., (1998), *Computational Inelasticity*, Springer, New York.

[6] Neto, E.A. de S., Perić, D., Owen, D.R.J., (2008). Computational methods for plasticity: theory and applications. Wiley, Chichester, West Sussex, UK.

[7] Coleman, B.D. (1964). Thermodynamics of materials with memory. *Archive for Rational Mechanics and Analysis*, **17**: 1.

[8] Starman, B., Halilovič, M., Vrh, M., Štok, B., (2014). Consistent tangent operator for cutting-plane algorithm of elasto-plasticity. *Comput. Methods Appl. Mech. Eng.*, **272**: 214–232.

[9] Krieg, R.D. and Krieg, D.B. (1977). Accuracies of numerical solution methods for the elastic perfectly plastic model. *Transactions of ASME Journal of Pressure Vessel Technology*, **99**: 510–515.

[10] Simo, J.C. and Taylor, R.L. (1985). Consistent tangent operators for rate-independent elastoplasticity. *Computer Methods in Applied Mechanics and Engineering*, **48**: 101–118.

[11] Shield, R. and Ziegler, H. (1958). On Prager's hardening rule. *Journal of Applied Mathematics and Physics (ZAMP)*, 9a: 260.

[12] Ziegler, H. (1959). A modification of Prager's hardening rule. *Quarterly of Applied Mathematics*, 17: 55.

[13] Pedroso, D.M., Sheng, D., Sloan, S.W. (2008). Stress update algorithm for elastoplastic models with nonconvex yield surfaces. *Int. J. Numer. Meth. Engng,* **76**, 2029–2062.

[14] Guennebaud, G., et al. (2010). Eigen V3. http://eigen.tuxfamily.org (Retrieved on 22 April, 2022).

[15] Kim, N. H. (2015). Introduction to Nonlinear Finite Element Analysis. Springer US, New York, NY.

[16] Dodds, R.H., (1987). Numerical techniques for plasticity computations in finite element analysis. *Computers & Structures*, **26**, 767–779.

Analysis for bioheat transfer with thermoelastic effect

†Kuo-Chi Liu

Department of Mechanical Engineering, Far East University, Taiwan.

†Corresponding author: kcliu@mail.feu.edu.tw

Abstract

The model that considers the thermoelastic effect of heat conduction medium with three phase lags was proposed. This work introduces the modified bioheat transfer equation based on the three-phase-lag model and uses it to study the thermal behavior in living tissue. Due to thermoelastic effect, the governing equation of the problem involves the high-order time derivative of temperature. A hybrid numerical scheme based on the Laplace transform is proposed to solve the present problem. The influence of thermoelastic parameters on the behavior of heat transfer in tissue has been investigated.

Keywords: Laplace transform, non-Fourier, three phase lag, thermoelasticity.

Introduction

The experimental data is more accessible and is helpful to establishing the reliable models of thermal behavior in living tissues for the development of medical technology. Various bioheat transfer models have been proposed. The Pennes model is the typical model and shows the infinite and rapid propagation of heat signals. The thermal wave (C-V) and the dual-phase-lag (DPL) models can be regarded as the most commonly used non-Fourier model. The thermal wave model shows the characteristic of finite propagation with a heat flux phase lag. In addition to the heat flux phase lag, the dual phase model defines the temperature gradient phase lag for the effect of micro-structural interactions. The values of the phase lags definitely impact the thermal response and dominate the model of bioheat transfer in tissues [1]. The non-Fourier models have been widely used in the analysis of biological heat transfer problems, such as radiofrequency/microwave ablation [2, 3], magnetic hyperthermia [4-7], laser therapy [8-10].

Recently, Roy Choudhuri [11] introduced the three-phase-lag model of heat conduction in medium with the thermoelastic effect according to one of thermoelectricity models, which was released by Green and Naghdi [12, 13] and involves dissipation in general and admits thermoelastic waves. The basic equation of the three-phase-lag (DPL) model is

$$\vec{q}(\vec{r},\, t + \tau_q) = -\left[\, k\vec{\nabla}T(\vec{r},\, t + \tau_T) + k_v\vec{\nabla}v(r,\, t + \tau_v)\right] \tag{1}$$

which satisfies $\dot{v} = T$. v is the thermal displacement. $\vec{q}$ and T symbol the heat flux vector and temperature, respectively. τ_q, τ_T and τ_v are the phase lags, heat flux vector, temperature gradient and thermal displacement gradient. t is time, $\vec{r}$ is the position vector, k is the thermal conductivity, and k_v is the material constant characteristic of the model. It was assumed that $k_v > 0$ and $\tau_q > \tau_T > \tau_v \geq 0$ [11]. It is able to contain Fourier law, thermal wave model, and dual phase model of heat conduction at the same time.

The relevant scholars [14-16] have also studied the problems of bio-heat transfer based on the DPL model. Kumar and Rai [14] presented the semi-analytical solution of the TPL model of bio-heat transfer. They found the phase lag of thermal displacement gradient has a significant effect on the temperature distribution in the tissue. Zhang et al. [15] presented a modified energy conservation equation and investigated the TPL thermal response in the skin with a constant surface temperature or the constant surface heat flux. Hobiny et al. [16] had the analytical solutions of the TPL model for the thermal responses of biologicals tissue with laser irradiation. The development of thermal therapy always requires more reasonable temperature distribution predictions. Therefore, this paper attempts to investigate the thermal behavior in tissue subjected to constant surface temperature using the TPL bioheat transfer

equation. A hybrid numerical scheme based on the Laplace transform is proposed to solve the present problem.

Mathematical Formulation

The development of the heat transport equations in the tissue was made based on Taylor series of the TPL model [11]. For constant physiological parameters, the first order expansion for q, T and v gets

$$\left(1 + \tau_q \frac{\partial}{\partial t}\right)\vec{q} = -k\vec{\nabla}\left[\left(1 + \tau_T \frac{\partial}{\partial t}\right)T + \left(1 + \tau_v \frac{\partial}{\partial t}\right)\left(\frac{k_v}{k}v\right)\right] \tag{2}$$

For $\dot{v} = T$, Eq. (2) can be rewritten, respectively, as

$$\left(\frac{\partial}{\partial t} + \tau_q \frac{\partial^2}{\partial t^2}\right)\vec{q} = -\left[k_v + (k + k_v\tau_v)\frac{\partial}{\partial t} + k\tau_T \frac{\partial^2}{\partial t^2}\right]\vec{\nabla}T \tag{3}$$

Eqs. (4) and (5) can become to the DPL model of heat conduction as $k_v = 0$ and get into the thermal wave model for $k_v = 0$ and $\tau_T = 0$. And then, the case of $k_v = 0$ and $\tau_q = \tau_T = 0$ would make them to be the Fourier law.

Considering the thermoelastic effect in tissue, the energy conservation equation of bio-heat transfer can be written as [15]

$$\rho c \left(\frac{\partial}{\partial t} + \frac{k_v}{k}\right)\frac{\partial T}{\partial t} = -\nabla \cdot \frac{\partial \vec{q}}{\partial t} + w_b \rho_b c_b \left(\frac{\partial}{\partial t} + \frac{k_v}{k}\right)(T_b - T) + \frac{\partial}{\partial t}(q_m + q_{ext}) \tag{4}$$

where ρ is density, c is specific heat, and w_b is the perfusion rate of blood. The temperature of blood T_b is kept at 37 ^{0}C. q_m is the metabolic heat generation. q_{ext} is the external heating.

The TPL equation of bio-heat transfer for temperature is obtained from Eqs. (3) and (4) by eliminating the heat flux q as

$$\rho c \left(1 + \tau_q \frac{\partial}{\partial t}\right)\left(\frac{\partial}{\partial t} + \frac{k_v}{k}\right)\frac{\partial T}{\partial t}$$

$$= \left[k_v + (k + k_v\tau_v)\frac{\partial}{\partial t} + k\tau_T \frac{\partial^2}{\partial t^2}\right]\nabla^2 T + w_b\rho_b c_b \left(1 + \tau_q \frac{\partial}{\partial t}\right)\left(\frac{\partial}{\partial t} + \frac{k_v}{k}\right)(T_b - T) \tag{5}$$

$$+ \left(\frac{\partial}{\partial t} + \tau_q \frac{\partial^2}{\partial t^2}\right)(q_m + q_{ext})$$

The present problem considers that the skin surface temperature could be kept constant as the skin contacts with a large steel plate at a high temperature. The assumption that tissue temperature approaches T_b deep in tissue $x = L$ was made [17]. The boundary conditions are described as

$$T(0,t) = T_h \quad \text{and} \quad T(L,t) = T_b \tag{6}$$

and the initial conditions are

$$T(x, 0) = T_b, \quad \frac{\partial T(x, 0)}{\partial t} = 0, \text{ and } \frac{\partial T^2(x, 0)}{\partial t^2} = 0 \tag{7}$$

Analytical Method

Eq. (5) is rewritten in the definition of $\theta = T - T_b$ as

$$
k\left[\frac{k_v}{k} + \left(1 + \frac{k_v}{k}\tau_v\right)\frac{\partial}{\partial t} + \tau_T\frac{\partial^2}{\partial t^2}\right]\nabla^2\theta - \rho c\left[\frac{k_v}{k} + \left(1 + \frac{k_v}{k}\tau_q\right)\frac{\partial}{\partial t} + \tau_q\frac{\partial^2}{\partial t^2}\right]\frac{\partial\theta}{\partial t}
$$
$$
- w_b\rho_b c_b\left[\frac{k_v}{k} + \left(1 + \frac{k_v}{k}\tau_q\right)\frac{\partial}{\partial t} + \tau_q\frac{\partial^2}{\partial t^2}\right]\theta \tag{8}
$$
$$
= -\left(\frac{\partial}{\partial t} + \tau_q\frac{\partial^2}{\partial t^2}\right)(q_m + q_{ext})
$$

The initial conditions become

$$\theta(x, 0) = 0, \quad \frac{\partial\theta(x, 0)}{\partial t} = 0, \text{ and } \frac{\partial\theta^2(x, 0)}{\partial t^2} = 0 \tag{9}$$

The Laplace transform technique is used to transform the differential terms in equations (14) and (15) with the initial conditions (16) into

$$\frac{d^2\tilde{\theta}(x,s)}{dx^2} - \lambda^2\tilde{\theta} = -f \tag{10}$$

$\tilde{\theta}$ symbols the Laplace transform of θ. The definitions of λ and f are

$$\lambda^2 = \frac{(\rho cs + w_b\rho_b c_b)\left[\frac{k_v}{k} + \left(1 + \frac{k_v}{k}\tau_q\right)s + \tau_q s^2\right]}{[k_v + (k + k_v\tau_v)s + k\tau_T s^2]} \tag{11}$$

and

$$f = \frac{(1 + \tau_q s)(q_m + q_{ext})}{[k_v + (k + k_v\tau_v)s + k\tau_T s^2]} \tag{12}$$

where s is the Laplace transform parameter of time t.

The boundary conditions (6) in the terms of $\tilde{\theta}$ can be written, respectively, as

$$\tilde{\theta}(0, t) = (T_h\text{-}T_b)/s \text{ and } \tilde{\theta}(L, t) = 0 \tag{13}$$

The current work divides the entire space domain into several sub-space domains. Eq. (17) in the jth sub-space domain can be written as

$$\frac{d^2\tilde{\theta}_j}{dx^2} - \lambda^2\tilde{\theta}_j = -f \qquad \text{for } x_i \leq x \leq x_{i+1}, j = i \tag{14}$$

The analytical solution of Eq. (14) in the interval $[x_i, x_{i+1}]$ with the boundary conditions

$$\tilde{\theta}_j(x_i) = \tilde{\theta}_{i,j} \quad \text{and} \quad \tilde{\theta}_j(x_{i+1}) = \tilde{\theta}_{i+1,j} \tag{15}$$

can be written as

$$\tilde{\theta}_j(x,s) = \frac{1}{sinh\lambda l}\left[\left(\tilde{\theta}_{i,j} - \frac{f}{\lambda^2}\right)sinh\lambda(x_{i+1} - x) + \left(\tilde{\theta}_{i+1,j} - \frac{f}{\lambda^2}\right)sinh\lambda(x - x_i)\right] - \frac{f}{\lambda^2} \tag{16}$$

where l denotes the length of sub-space domain or the distance between two neighboring nodes.

The heat flux and temperature in the entire space domain are continuous is assumed, so

$$\tilde{\theta}_{j-1}(x_i) = \tilde{\theta}_j(x_i) = \tilde{\theta}_i \tag{17}$$

and

$$\frac{d\tilde{\theta}_{j-1}(x_i)}{dx} = \frac{d\tilde{\theta}_j(x_i)}{dx} \tag{18}$$

Substituting Eqs. (16) and (17) into Eq. (18) can obtain the discretized form at the *i*th node as

$$\tilde{\theta}_{i-1} - 2\cosh\lambda l\,\tilde{\theta}_i + \tilde{\theta}_{i+1} = [2 - 2\cosh\lambda l]\frac{f}{\lambda^2} \tag{19}$$

The discretized forms at the internal nodes and the boundary surfaces are rearranged as

$$[M]\{\tilde{\theta}\} = \{F\} \tag{20}$$

where $[M]$ is a matrix with the complex number s. $\{\tilde{\theta}\}$ and $\{F\}$ are a column vector and represent the unknown nodal evaluation temperatures in the Laplace transform form and the forcing term, respectively. The physical value of the nodal temperature can be obtained from Eq. (20) with the Gaussian elimination algorithm and the numerical inversion of the Laplace transform [18].

Results and Discussion

The values of the parameters used in the present work are set as $q_0 = 20$ kW/m^2 , $R_d = 0.0528$, $L = 9$ mm, $\rho = 1190$ kg/m^3 , $c = 3600$ J/kg·K, $\rho_b = 1060$ kg/m^3, $c_b = 3770$ J/kg·K, $w_b = 1.87 \times 10^{-3}$ s^{-1}, $k = 0.235$ W/(m·K), and $k_v = 0.1$ W/(m·K·s) . The values of the three phase lags would be assumed as the different values $\tau_q = 16$ s, $\tau_T = 6$ s, and $\tau_v = 2$ s. The skin surface temperature contacted with a large steel plate is $T_h = 80$ ^{0}C. For comparison and discussion, some parameter values may be adjusted and noted in each figure.

Figure 1 presents the temperature variation at $x = $ L/4 for various values of τ_v. The slight oscillation appears in the curves of temperature variation. It is seen that the amplitude decreases as τ_v increases. This result implies that the thermoelastic effect would create the oscillation phenomenon in heat transfer.
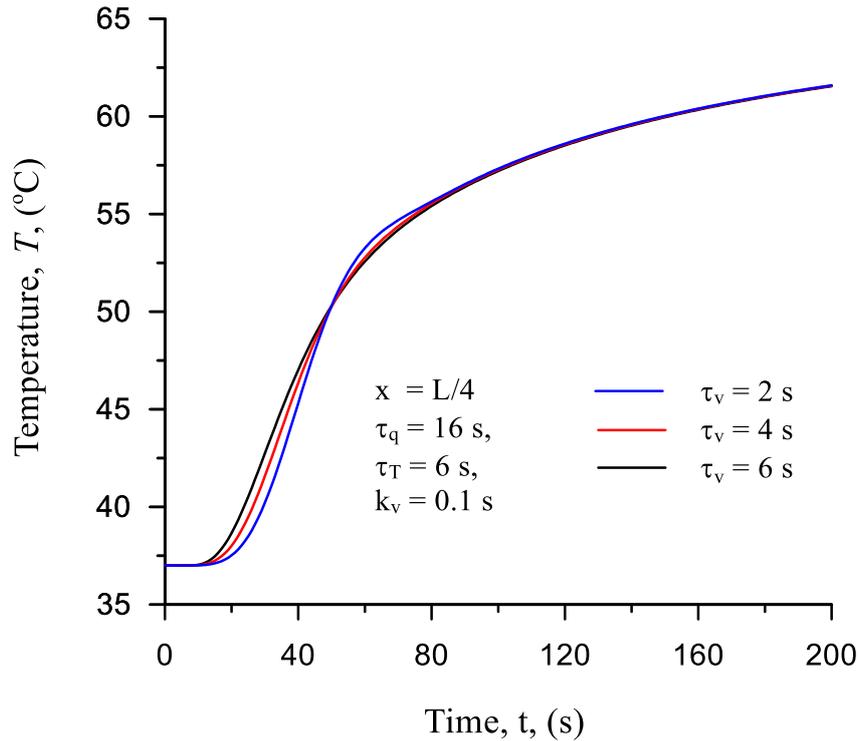
Figure 1. Temperature variation at $x = L/4$ for various values of τ_v.

Figure 2 depicts the temperature variations with $\tau_v = 0$ at $x = L/4$ for various values of k_v. It is observed that the temperature at $x = L/4$ closes to consensus after $t = 100$ s. However, the oscillation phenomenon is more obvious as the k_v value increases for $\tau_v = 0$. At the same time, the oscillation phenomenon is also more obvious than that presented in Fig 1. This result implies that k_v has a greater effect on the behavior of heat transfer than τ_v.
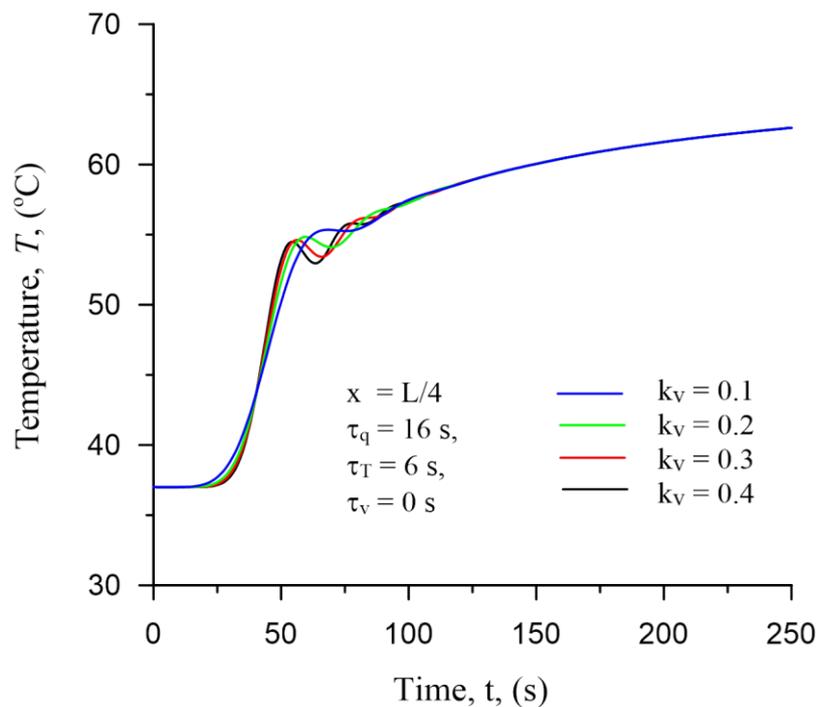
Figure 2. Temperature variations with $\tau_v = 0$ at $x = L/4$ for various values of k_v.

The effect of the thermal displacement gradient lag τ_v on the behavior of bioheat transfer is further explored. Figure 3 shows the temperature variations at $x = L/4$ with $\tau_v = 2$ s or $\tau_v = 6$ s for various values of k_v. As $\tau_v = 6$, the curves of temperature variation for $k_v = 0.1$, 0.2, 0.3, and 0.4 have overlapped. However, minor discrepancies and oscillations can also be seen among the variation curves for $\tau_v = 2$. This result expresses that the main factor causing the oscillation phenomenon of heat transfer in tissue with thermoelastic effect should be the value of k_v. Conversely, τ_v suppresses the generation of oscillations.
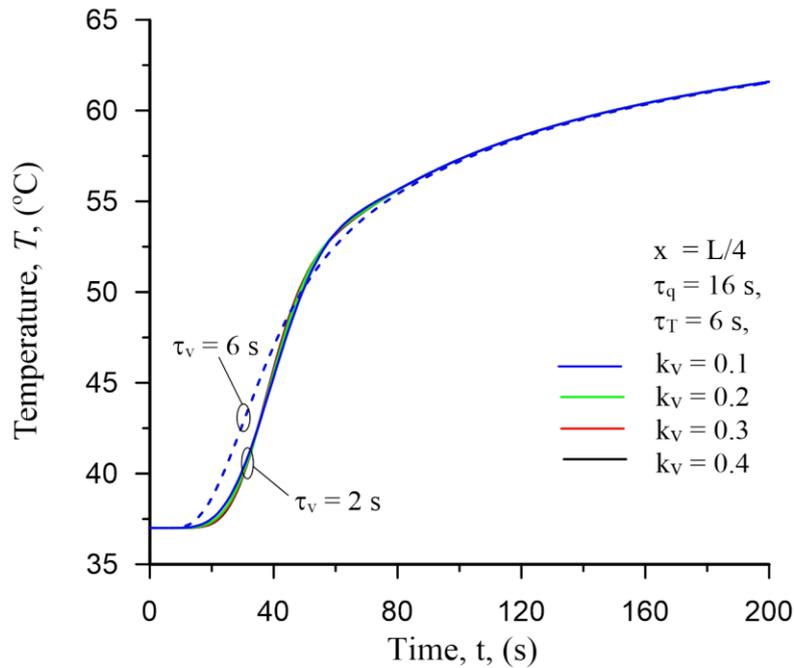
Figure 3. Temperature variations with $\tau_v = 2$ s or $\tau_v = 6$ s at $x = L/4$ for various values of k_v.
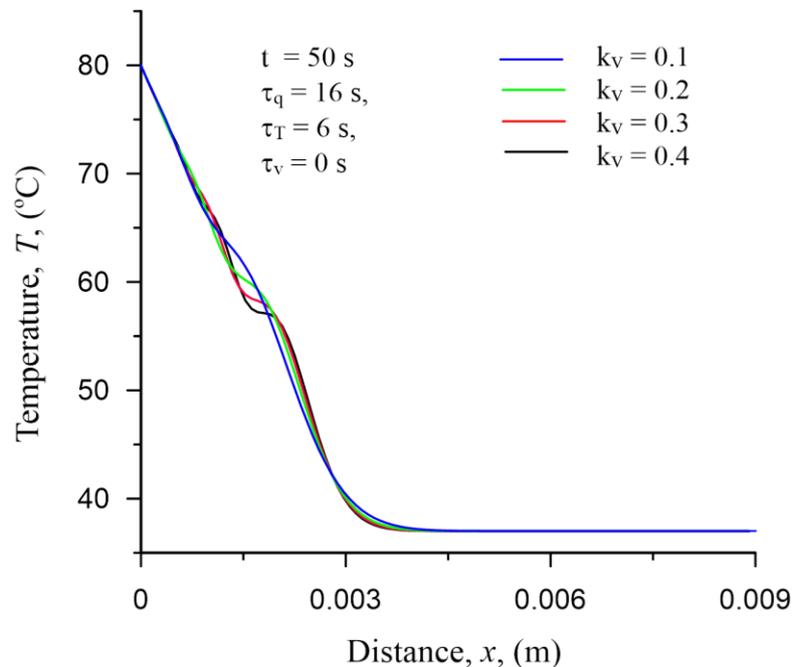
Figure 4. Temperature distribution at $t = 50$ s for various values of k_v with $\tau_q = 16$ s, $\tau_T = 6$ s, and $\tau_v = 0$ s.

The effect of k_v on the temperature distribution at $t = 50$ s is explored as shown in Fig. 4. The figure displays the effect of k_v enhances the oscillation and reduces the propagation speed of thermal signal. Reduction of k_v would raise the temperature.

Figure 5 depicts the variation of temperature distribution over time. The thermal signal has not reached the boundary end $x = L$ before $t = 50$ s. The place where the oscillation occurs moves forward in time.
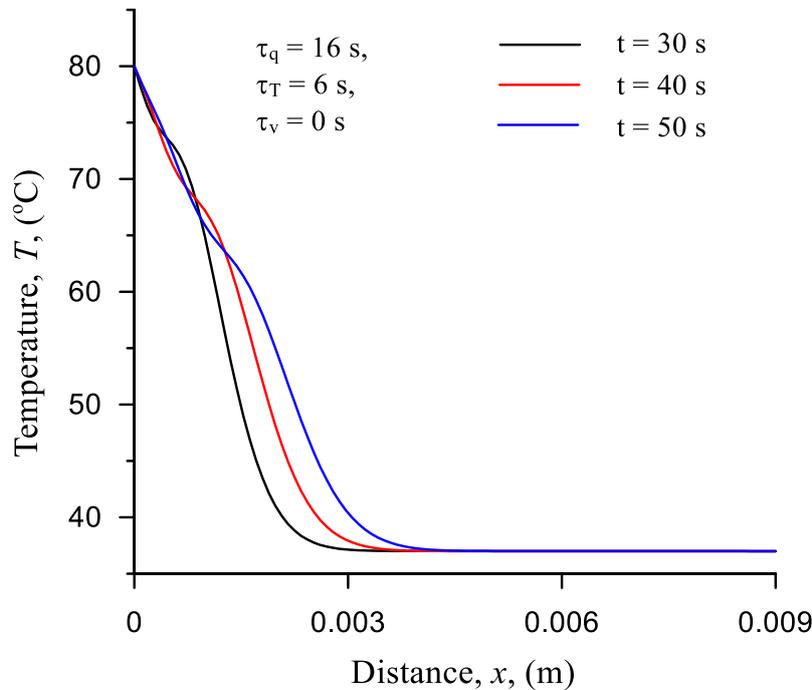
Figure 5. Temperature distribution at various times for $\tau_q = 16$ s, $\tau_T = 6$ s, and $\tau_v = 0$ s.

Conclusions

A numerical scheme based on the Laplace Transform method is proposed for solving the three-phase-lag bioheat transfer equation. The present results indicate that the thermoelastic effect would create the oscillation phenomenon in heat transfer. The temperature oscillation is depressed with increasing the value of τ_v. The effect of k_v enhances the oscillation and reduces the propagation speed of thermal signal. Reduction of k_v would raise the temperature.

References

[1] Liu, K.C., Wang, Y.N. & Chen, Y.S. (2012). Investigation on the bio-heat transfer with the dual-phase-lag effect. *International Journal of Thermal Sciences* **58**, 29-35.
[2] Singh, S. & Melnik, R. (2019). Coupled thermo-electro-mechanical models for thermal ablation of biological tissues and heat relaxation time effects. *Physics in Medicine & Biology* **64**, 245008.
[3] Singh, S. & Melnik, R. (2020). Thermal ablation of biological tissues in disease treatment: A review of computational models and future directions. *Electromagnetic biology and medicine* **39**, 49-88.
[4] Liu, K.C. & Chen, T.M. (2018). Analysis of the thermal response and requirement for power dissipation in magnetic hyperthermia with the effect of blood temperature. *International Journal of Heat and Mass Transfer* **126**, 1048-1056.
[5] Kumar, R., Vashishth, A.K. & Ghangas, S. (2019). Nonlocal heat conduction approach in a bi-layer tissue during magnetic fluid hyperthermia with dual phase lag model. *Bio-medical materials and engineering* **30**, 387-402.
[6] Liu, K.C. & Cheng, P.J. (2019). Numerical analysis of power dissipation requirement in magnetic hyperthermia problems. *Journal of Thermal Biology* **86**, 102430.
[7] Raouf, I., Khalid, S., Khan, A., Lee, J., Kim, H.S. & Kim, M.H. (2020). A review on numerical modeling for magnetic nanoparticle hyperthermia: Progress and challenges. *Journal of Thermal Biology* **91**, 102644.

[8] Hooshmand, P., Moradi, A., Khezry, B. (2015). Bioheat transfer analysis of biological tissues induced by laser irradiation. *International Journal of Thermal Sciences* **90**, 214-223.

[9] Nóbrega, S. & Coelho, P.J. (2017). A parametric study of thermal therapy of skin tissue. *Journal of thermal biology* **63**, 92-103.

[10] Falahatkar, S., Nouri-Borujerdi, A., Najafi, M. & Mohammadzadeh, A. (2017). Numerical solution of non-Fourier heat transfer during laser irradiation on tooth layers. *Journal of Mechanical Science and Technology* **31**, 6085-6092.

[11] Roy Choudhuri, S.K. (2010). On thermoelastic three phase lag model. *J. Therm. Stress.* **30**, 231–238.

[12] Green, A.E. & Naghdi, P.M. (1992). On undamped heat waves in an elastic solid. *J. Therm. Stress.* **15**, 253–264.

[13] Green, A.E. & Naghdi, P.M. (1993). Thermoelasticity without energy dissipation. *J. Elast.* **31**, 189–208.

[14] Kumar, D. & Rai, K. N. (2020). Three-phase-lag bioheat transfer model and its validation with experimental data. *Mechanics Based Design of Structures and Machines* **online**, DOI: 10.1080/15397734.2020.1779741

[15] Zhang, Q., Sun, Y., Yang, J. (2020). Bio-heat response of skin tissue based on three-phase-lag model. *Scientific Reports* **10**, 16421.

[16] Hobiny, A., Alzahrani, F., & Abbas, I. (2020). Analytical estimation of temperature in living tissues using the TPL bioheat model with experimental verification. *Mathematics* **8**, 1188.

[17] Liu, K.C.(2008). Thermal propagation analysis for living tissue with surface heating. International Journal of Thermal Sciences 47, 507-513.

[18] Honig, G., Hirdes, U., (1984) A method for the numerical inversion of Laplace transforms, *J. Comp. Appl. Math.* **10**, 113-132.

Numerical study of effects of wind on the vertical fire spread with vertical/horizontal spandrel

*Zefeng Huang[1], †Zhao Tian[1], and †Xiao Chen[2]

[1]School of Mechanical Engineering, University of Adelaide, Australia.

[2]Sotera Fire Engineering, Australia.

*Presenting author: a1732341@student.adelaide.edu.au

†Corresponding author: xiao.chen@sotera.com.au; zhao.tian@adelaide.edu.au

Abstract

Due to the importance of preventing the vertical fire spreading along with buildings, two fire inhibition methods were raised by the National Construction Code clause [1] C2.6(a) including vertical spandrel of at least 900 mm high or horizontal spandrel of at least 1100 mm deep. This project aims to answer the research question of whether vertical spandrel of 900 mm and horizontal construction of 1100 mm are equivalent in performance in inhibiting the vertical fire spreading under the effects of wind using fire dynamics simulator (FDS) simulations. The geometry of the simulations is modified from experimental works conducted by Oleszkiewicz [2] by adding an air opening on the back wall. The preliminary results show that by slightly increasing the front wind (the wind direction is normal to the front opening of the building) from 0 m/s to 0.5 m/s, the radiation heat transfer from the flame to the above floor is increased slightly, however, further increasing the front wind speed will reduce the radiative heat flux on the above floor, due to the blocking effect of the front wind. When the front wind speed increases to above 4 m/s, the flame is blocked within the room. When the side wind (the wind direction is parallel to the front opening of the building) is introduced, there is a slight increase in heat flux for the wind speed of 1~4 m/s. Based on the preliminary simulation results, it is found that for the wind conditions, fire load and building structure investigated in the paper, the performance of the 900 mm vertical spandrel is lower than the horizontal spandrels even for the horizontal spandrel of 500 mm.

Introduction

During numerous fire accidents within constructed buildings, hot smoke that was emitted from openings such as windows/doors of the incident floor can cause above floors to be inflamed as well. When the room on fire is under-ventilated, the fire caused by the unburned fuel will destroy the windows openings and spread along the exterior walls, and once the heat flux is sufficient enough, the flame height could exceed floor heights causing subsequent fire to spread between floors which will lead to significant damage to properties and loss of lives [4]. Therefore, preventing the vertical spreading of fire between floors via openings has been a major aspect of fire safety engineering [5]. As described in National Construction Code (NCC) clause C2.6(a) [1], there would be two possible approaches to preventing vertical fire spread. One of these two approaches is using a vertical spandrel of more than 900 mm in height. And the second approach is adding a horizontal construction which needs to project at least 450 mm

beyond the openings with a minimum depth of 1100 mm. NCC implies that those two options are alternatives to each other in terms of their efficacy in inhibiting fire spread. The previous study on the clauses using the computational fluid dynamics (CFD) code FDS simulations from [6] suggests that horizontal projection has better fire inhibition performance than vertical spandrel. However, the authors of [6] only conducted the investigation based on a single defined opening geometry and three different heat release rates without consideration of the effect of other parameters, such as outdoor winds. Other similar papers analysing the performance of protection methods such as [7-11] also have significant limitations in their scope. Consequently, it becomes crucial for this project to understand the effect of outdoor winds on the fire inhibition performance difference between horizontal construction and vertical spandrel. This conclusion on the performance of fire inhibition methods could then potentially be used to verify the consistency of existing building code and even act as a guideline for future fire inhibition methods applied for multistorey buildings.

This paper reports the preliminary results of the research on the performance of the vertical spandrel and the horizontal construction based on the Australia NCC requirement in inhibiting the vertical fire spreading under outdoor winds using fire dynamics simulator (FDS) simulations.

Numerical methods and processes

For this project, FDS version 6.7.7 [12] with Pyrosim pre-processor [13] is employed. According to the technical guide from [14], FDS solves the Naiver-Stokes equations to compute flow fields. Additionally, large eddy simulation (LES) is utilised to handle turbulence.

The computation domain, which is modified from the experimental geometry described in [2] and simulation geometry elaborated in [11] is modeled in Pyrosim as shown in Figure 1. Please note that different from the geometry in [2], in this project, an additional air intake with dimensions of 5.4 m x 0.8 m is included at the back of the ground floor to ensure that there would be sufficient air supply throughout the simulation. The domain has an overall size of 7.2 m x 11.2 m x 10.8 m whilst the construction locates in the middle of the domain has the dimensions of 6.4 m x 4.8 m x 10.4 m. The dimension of the domain was made sufficient for the plume to spread beyond the openings. The opening is located in the negative y-axis direction. The front opening for the ground floor has a dimension of 1.0 m x 2.7 m whilst regions above the front opening are sealed off for measurement. An additional air intake with dimensions of 5.4 m x 0.8 m is included at the back of the ground floor to ensure that there would be sufficient air supply throughout the simulation. The thickness of the walls is defined as 0.1 m. The fire source is placed at the center of the room with a surface area of 9 m^2. The height of the vertical spandrel is defined as 0.9 m. The depth of the horizontal construction is varied between 0.5 m and 1.3 m for parametric analysis. The full range of horizontal construction dimensions included for this project are described in table 1.
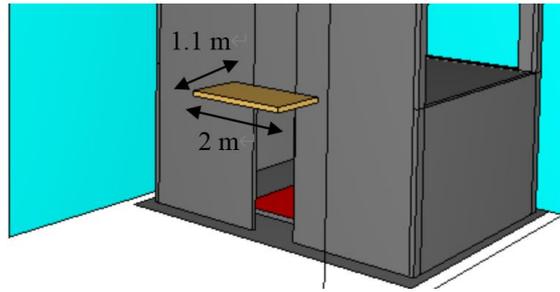
In terms of meshing, a mesh size of 0.1 m is applied for the entire domain which leads to fine mesh resolution according to [13]. The mesh has a D*/dx ratio of approximately 22 which is slightly larger than the suggested value of $\geq$20 from Pyrosim [13].
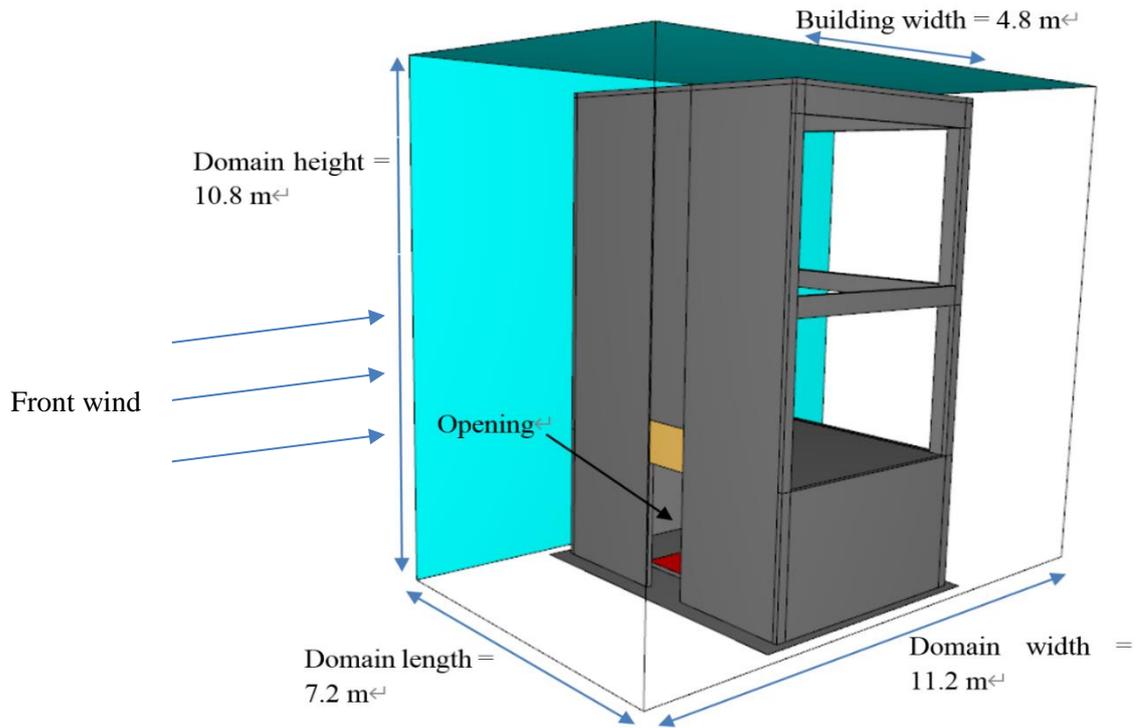
Figure 1 (a): Overall domain for FDS simulation
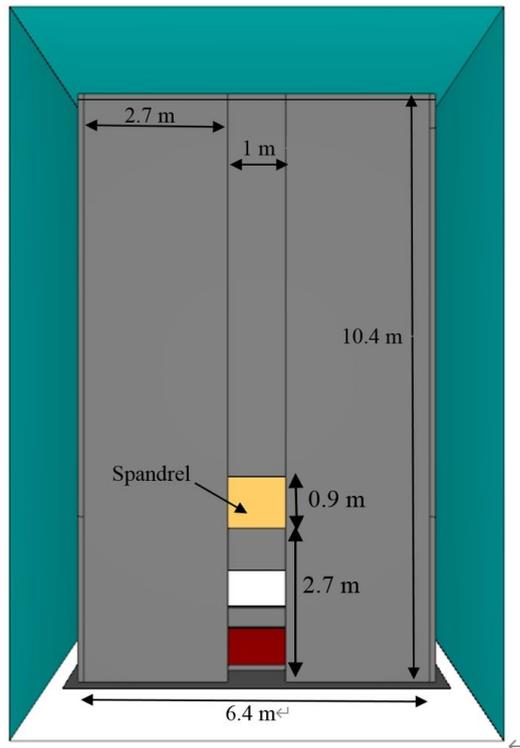
Figure 1 (b): Front view of computational domain with 0.9 m vertical spandrel

Figure 1 (c): Computational domain with 1.1 m horizontal spandrel

For the boundary conditions, the six boundary surfaces of the overall mesh domain are modeled as vents which essentially act as openings to allow air to flow in and out of the domain. For the cases that include outdoor wind, the wind inlet is modelled as an air supply with a designated wind speed. The inert boundary condition is then applied for the spandrels included in the geometry which represents a smooth wall with fixed ambient temperature and emissivity of 0.9. The rest of the walls are defined as concrete walls. For the heat release rate of the burner, a constant total release rate per unit area of 900 kW/m^2 is used. This results in an average heat release rate of 8.1 MW which is roughly maintained after the fire is fully developed. The fire will last for 1800s.

Since this project aims to explore the effect of wind parameters, supply surface boundary conditions are applied at the inlet surface with designated flow speed. The variation of wind parameters included is described in Table 1. In Table 1, for wind direction, 'Front' means the wind direction is normal to the front opening of the building, while 'Side' means the wind direction is parallel to the front opening of the building.

Table 1: Parameters investigated in the simulations.

Horizontal Spandrel Size (m)	Wind direction	Wind speed (m/s)
1.3, 1.1, 0.9, 0.7, 0.5	Front, Side	0.25, 0.5, 0.75, 1, 2, 3, 4, 5

Results and discussion

From Figure 2, it is evident that under the front wind of 1 m/s, the 0.9 m vertical spandrel has higher incident heat flux values on the above wall than those of the horizontal cases, indicating a lower performance in terms of preventing the vertical fire spread. Additionally, increasing the horizontal construction length would lead to better protection performance as well. As shown in Figure 3, by slightly increasing the front wind from 0 s/m to 0.5 m/s, the radiation heat transfer from the flame to the above floor is increased slightly. In other words, low front wind speeds (up to 0.5 m/s in this case) would reduce the effectiveness of the protection methods. However, for front winds with a speed over 1 m/s, increasing the wind speed would be beneficial in preventing vertical fire spread instead, as the heat flux values on the above wall decrease with increasing wind speed until it reaches a constant value of 0.43 kW/m^2. Among the analysed protection methods shown in Figure 3, for low wind speeds up to 3 m/s, the 1.1 m

horizontal projection is more efficient in preventing fire spread than the other protection methods due to its lowest predicted heat flux received at the external wall of the above level.
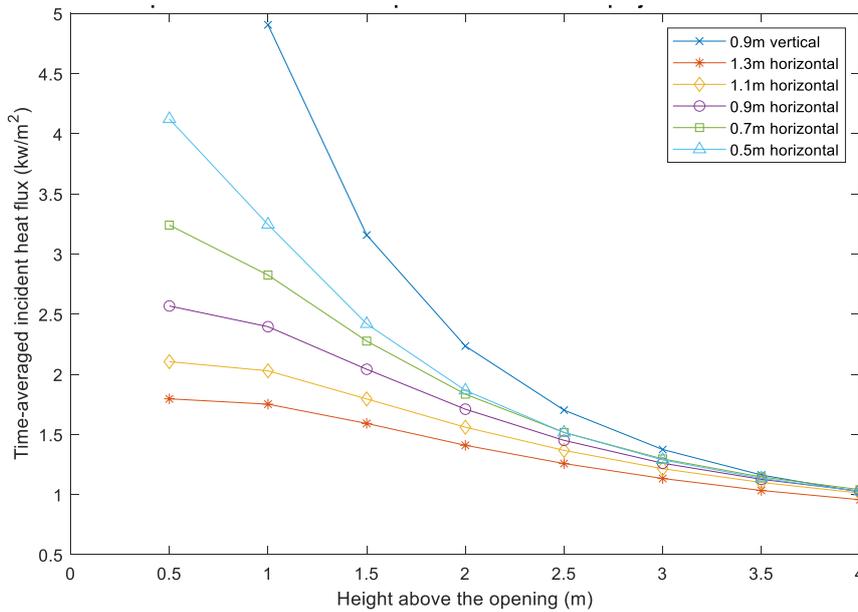

Figure 2: Performance comparison between horizontal construction and vertical spandrel for the 1 m/s front wind
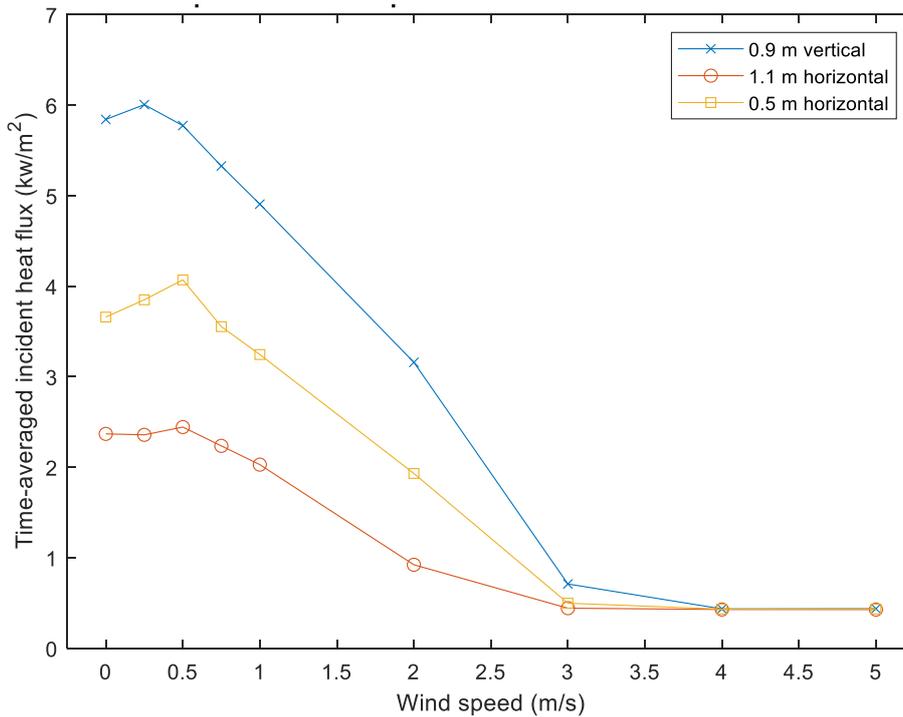
Figure 3: Performance comparison between horizontal construction and vertical spandrel for the front wind with various speeds

Temperature plots for 1.1 m horizontal construction are employed to explore the effect of increasing front wind speed on the performance of protection methods. As shown in Figure 4, by slightly increasing the front wind up to 0.5 m/s, the front wind (from the left-hand side of the figure) pushes the flame (which is indicated by the high-temperature zone) above the

horizontal construction toward the right-hand side, leading to radiation heat transfer from the flame to the above floor increased slightly. As the wind speed increases further to 1 m/s, the flame is pushed back toward the opening below the horizontal construction, decreasing the radiation on the above wall as shown in Figure 3. When the outdoor wind speed increase to 3 m/s, the high-temperature region of over 500 °C is nearly blocked under the horizontal construction. Eventually, when the front wind increases to 5 m/s, the flame is completed blocked within the room, which justifies the lowest heat flux recorded in such conditions as shown in Figure 3.
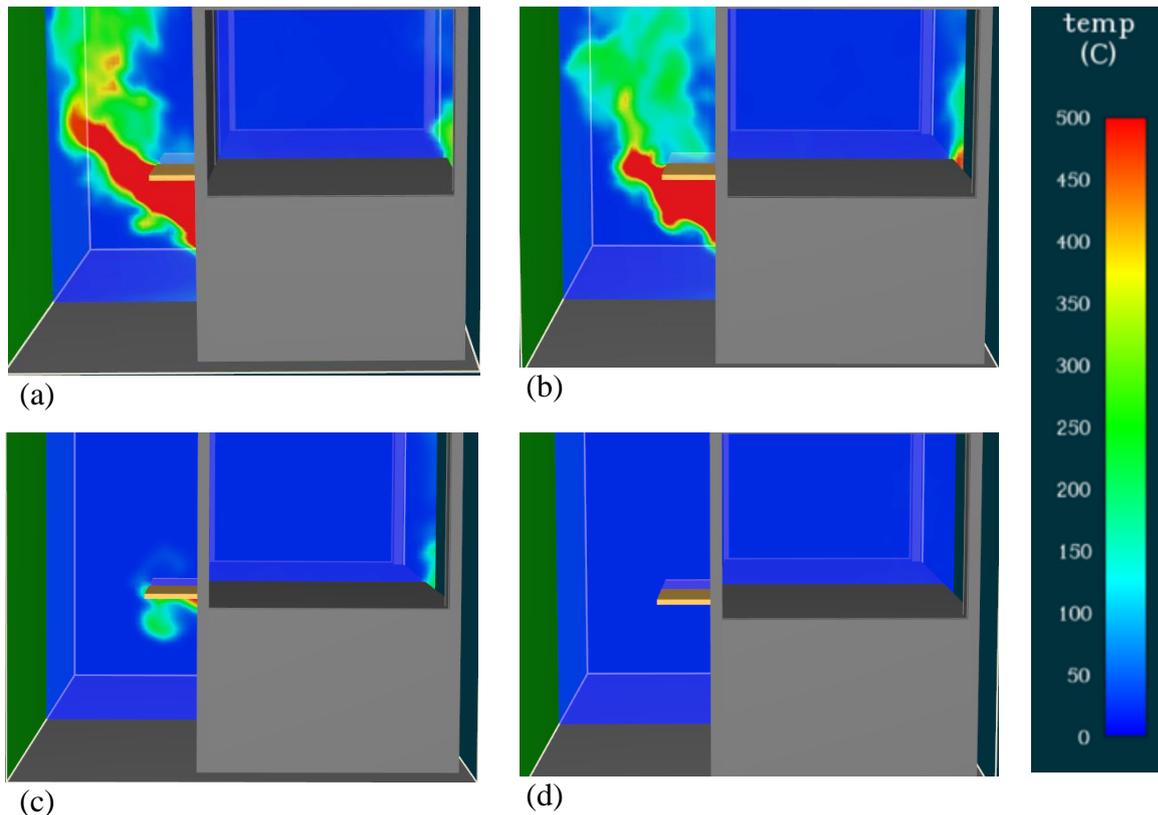
Figure 4: External Temperature plots for 1.1 m horizontal construction with different front wind speed (a) 0.5 m/s, (b) 1 m/s, (c) 3 m/s, (d) 5 m/s.

Temperature contours for spaces inside the structure are plotted to further develop the conjecture. As shown in Figure 5, at a lower front wind speed of 0.5 m/s, the flame travels vertically and then circulates around the ceiling. As the front wind speed increases to 2 m/s – 5 m/s, high-temperature zones near the ground are pushed away towards the rear ventilation slot (the right-hand side of the figure), leading to the lower temperature being experienced at the front wall. Additionally, more outdoor air at 20 °C is introduced to the room at a higher wind speed, cooling down the room.

In terms of the cases with a side wind, as shown in Figure 6, similar to the normal wind cases, the side wind with lower speeds would compromise the performance of protection methods as the heat flux values on the above wall are higher than those of heat flux with 0 m/s side wall. Additionally, increasing side wind speed from 3 m/s to 5 m/s, in this case, is beneficial for inhibiting vertical fire spread. However, unlike the previous cases, there is not a significant performance difference between the cases with 5 m/s side wind and the cases with 0 m/s side

wind. In terms of differences between protection methods, fluctuation in heat flux for vertical spandrel is the most severe out of all protection methods which indicates its protection capability is the most affected by the side wind out of all protection methods investigated. Interestingly, there are abrupt changes in the heat flux for the vertical spandrel case of 1 m/s shown in Figure 6. The causes of the high heat flux on the above wall are still unknown and are under investigation. Nevertheless, it does not affect the general trends and findings of the side wind cases reported in the paper.
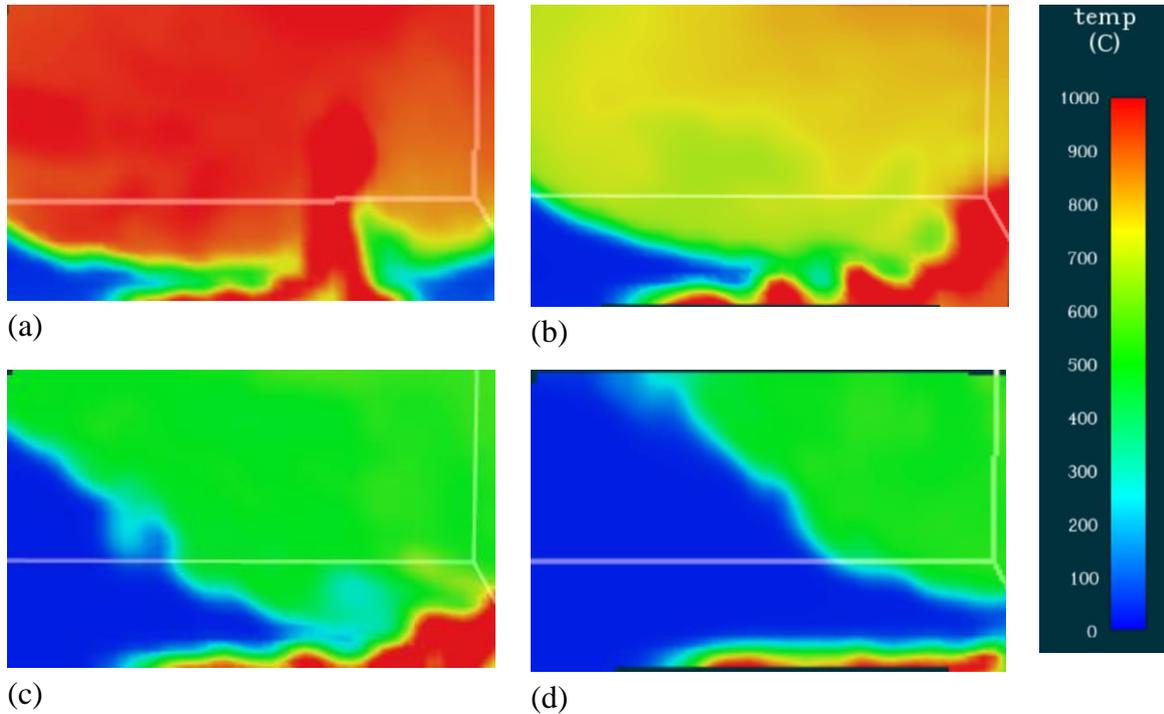
Figure 5: Internal temperature plots for 1.1 m horizontal construction with different front wind speed (a) 0.5 m/s, (b) 2 m/s, (c) 3 m/s, (d) 5 m/s
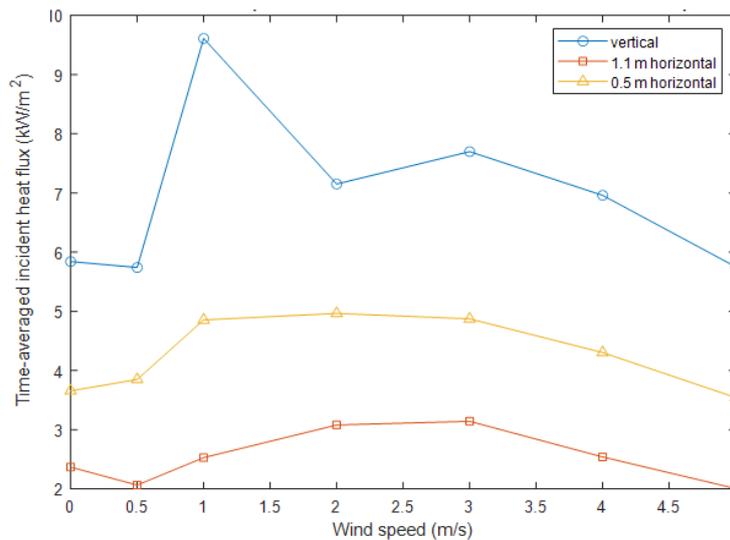
Figure 6: Performance comparison between horizontal construction and vertical spandrel for side winds with various speeds

Conclusions & Future work

Based on the preliminary results, it is found that for the wind conditions (both the front wind and side wind conditions), fire load and building structure investigated in the paper, the performance of the 0.9 m vertical spandrel in preventing vertical fire spread is lower than the horizontal spandrels even for the horizontal spandrel of 0.5 m.

By increasing the front wind from 0 m/s to 0.5 m/s, the radiation heat transfer from the flame to the outdoor wall surface above the opening is increased slightly, however, further increasing the front wind speed will reduce the radiative heat flux on the above floor, due to the blocking effect of the front wind. When the front wind speed increases to above 4 m/s, the flame is blocked within the room. Meanwhile, the changes in heat flux on the outdoor wall above the opening are much less significant for the various side wind speeds from 0 m/s to 5 m/s, partially due to the less significant blocking effects. Nevertheless, these results are only applicable to the investigated conditions.

As part of future work, more simulations need to be run to establish systematic conclusions on performance comparison based on the exact minimal dimensions suggested by NCC clauses. Further efforts will be put into parametric studies where the effect of fire size, opening size, wind speed, and other environmental parameters will be accounted for. Eventually, as the final objective of the project, the simulation model will be applied to real-life scenarios.

Acknowledgments

References

[1] Commonwealth of Australia and the States and Territories, The National Construction Code, Australian Building Codes Board, 2019.
[2] I. Oleszkiewicz, Heat transfer from a window fire plume to a building facade, Paper (National Research Council of Canada. Institute for Research in Construction); no. IRC-P-1662 (1989).
[3] M.A. Delichatsios, J. Ryan, N. Tian, J. Zhang, Vertical safe separation distance between openings in multi-story buildings having a fire-resistant spandrel, MATEC web of Conferences, EDP Sciences, 2016, p. 04003.
[4] M.A. Delichatsios, J. Ryan, N. Tian, J. Zhang, Vertical safe separation distance between openings in multi-storey buildings having a fire resistant spandrel, MATEC Web of Conferences 46 (2016) 04003.
[5] G. Hu, Research on the Fire of High-rise Residential Building Based on Pyrosim Numerical Simulation, IOP Conference Series: Earth and Environmental Science 455 (2020) 012059.
[6] D. Weinert, W. Poh, Performance of horizontal projections in vertical separation of openings in external walls–comparison with BCA solutions, Proceedings of the International Conference on Fire Safety Engineering, Gold Coast, Australia, Citeseer, 2006.
[7] G. Hadjisophocleous, Q. Jia, Comparison of FDS Prediction of Smoke Movement in a 10-Storey Building with Experimental Data, Fire Technology 45(2) (2009) 163-177.
[8] M. Nilsson, B. Husted, A. Mossberg, J. Anderson, R.J. McNamee, A numerical comparison of protective measures against external fire spread, Fire and Materials 42(5) (2018) 493-507.
[9] A. Čolić, I.B. Pečur, Influence of Horizontal and Vertical Barriers on Fire Development for Ventilated Façades, Fire Technology 56(4) (2020) 1725-1754.
[10] W. An, Q. Meng, R. Pan, H. Zhu, Influence of horizontal projection on upward flame spread over XPS thermal insulation material, Fire and Materials 42(5) (2018) 527-536.
[11] P. McKeen, Z. Liao, The impact of horizontal projections on lateral fire spread in multi-unit residential buildings - comparison of numerical and similarity correlations, Fire Safety Journal 126 (2021) 103441.

[12] K. McGrattan, S. Hostikka, R. McDermott, J. Floyd, C. Weinschenk, K. Overholt, Fire dynamics simulator--Technical reference guide, Sixth ed., National Institute of Standards and Technology, Gaithersburg, MD, [online], 2013.

[13]   Thunderhead   Engineering,   Pyrosim   User   Manual,   2021. https://support.thunderheadeng.com/docs/pyrosim/2021-3/user-manual/. (Accessed 18/10/2021.

[14] K. McGrattan, R. McDermott, M. Vanella, S. Hostikka, J. Floyd, Fire dynamics simulator--Technical reference guide, Sixth ed., National Institute of Standards and Technology, Gaithersburg, MD, [online], 2013.

AUTHOR INDEX