J. Appl. Comput. Mech., 6(3) (2020) 373-382
DOI: 10.22055/JACM.2019.30242.1699

ISSN: 2383-4536
jacm.scu.ac.ir

M&MoCS

Shahid Chamran University of Ahvaz

Journal of Applied and Computational Mechanics

Research Paper

An Efficient Implementation of Phase Field Method with Explicit Time Integration

Wenlong Zhang, Ala Tabiei

Department of Civil Engineering, University of Cincinnati, Cincinnati, OH, 45220, USA

Received July 09 2019; Revised August 29 2019; Accepted for publication September 06 2019.
Corresponding author: Wenlong Zhang (zhang2wl@mail.uc.edu)

Abstract: The phase field method integrates the Griffith theory and damage mechanics approach to predict crack initiation, propagation, and branching within one framework. No crack tracking topology is needed, and complex crack shapes can be captures without user intervention. In this paper, a detailed description of how the phase field method is implemented with explicit dynamics into LS-DYNA is provided. The displacement field and the damage field are solved in a staggered approach and the phase field equation is solved every N^{th} time step (N is refered to as calculation cycle) to save computational time. An N value smaller than $1/400$ of the total time step numbers is suggested. Several simulations are presented to demonstrate the feasibility of this solving scheme.

Keywords: Phase field method, Calculation cycle, LS-DYNA, Explicit time integration.

1. Introduction

Fracture as a primary failure mechanism of engineering components has been studied extensively since the mid of the last century. Although a solid theoretical foundation has been built for fracture mechanics, crack initiation and propagation prediction remains a challenging topic for engineers and researchers. Since experiments can be both expensive and challenging, numerical models serve as efficient approaches to study this area. Various numerical methods have been proposed to handle the crack propagation behavior, for example, rezoning method for plates and shell [Areias 2013], meshfree cracking particle method [1], splitting particle method [2], and the dual-horizon peridynamics method [3], [4]. This paper focuses on phase field method, a smeared approach to handle crack propagation.

The phase field method is a valuable tool to describe state change and has thus been widely used on solidification problems [5]. Combined and inspired by the gradient damage model and variational approach [6]–[8], it was introduced to fracture analysis and keeps gaining popularity [9]. Developed based on the foundation of Γ-convergence, the phase field method creatively combines damage mechanics and classical Griffith energy theory and integrate the crack initiation, propagation, and branching into the same framework. Instead of describing the crack with sharp discontinuities [10], the phase field method regularizes the sharp crack surface topology into the solid by a diffusive phase field variable, thus avoiding complex and difficult crack topologies used in remeshing [11] and XFEM [12]scheme.

Miehe (2010) [11] used a thermodynamically consistent framework for developing the phase field method for quasi-static brittle fracture problems. He proposed a concept of *surface energy density function* $\gamma(d, \nabla d)$ and separated the compression and tension energy-driven fracture. These two concepts deeply influenced the following phase field method development [13]–[15]. Borden (2012) [15] and Hofacker (2012) [14] extended the phase field method to dynamic brittle fracture, and they both used a *staggered scheme* to update the phase field and displacement field successively. Hesch (2014) [16] incorporated nonlinear material behavior and finite deformation into the phase field method and used an energy-

Published online: September 07 2019

momentum consistent integrator to handle the non-linear behavior. Verhoosel (2014) [17] extended the phase field method to cohesive fracture and applied it to adhesive interfaces. In terms of numerical implementation, the staggered scheme mentioned above is regarded as a more efficient and robust procedure. Shuwei extended the phase field method for poroelastic media [18]. Msekh (2018) applied the phase field to composite and considers both matrix and interphase fracture [19]. User-defined element and material has been implemented into commercial finite element software Abaqus [20], [21]. The concept of phase field method has also been adopted into other numerical frameworks to create new possibilities. For example, Kakouris (2017) [22] adopted the phase field method for material point method (MPM) to handle large deformation and high nonlinearity crack propagation problems.

Although lots of research has been done in implementing the phase filed method for crack propagation prediction, implementation into explicit dynamic problems has just started [3]. For highly nonlinear, fast crack propagation, complex contact condition, and severe structure softening scenarios, explicit time integration is necessary as it can capture the high dynamic effect and does not have convergence issue. LS-DYNA is a well-known finite element software for its explicit time integration and its flexibility of user-defined subroutines, in which a user can not only define material properties in an element-level but also gather the information of the whole model and do manipulations on the assembly level. Thus the phase field implementation into LS-DYNA is described in this paper. Essential details of the implementation are elaborated and can serve as documents for future researchers.

The structure of this paper is as follows: In Section 2 the concept of phase field is described; in Section 3 a detailed description of the implementation into LS-DYNA is given, and a concept of calculation cycle is proposed; in Section 4, two numerical models are shown to demonstrate the robustness of our proposed concept and implementation.

2. Phase Field Method Formulation

2.1. Solving the displacement and damage as an energy minimization approach

Consider a domain $\Omega \subset \mathbb{R}^D$ $(D = 1,2,3)$, with $\partial\Omega_u$ the Dirichlet boundary, $\partial\Omega_t$ the Neuman boundary, and Γ the interior crack surface with a dimension of $D-1$, the total variational potential energy $\mathcal{E}$ (Equation (2)) is then a sum of the fracture energy Ψ_c, elastic strain energy $\Psi_e(\boldsymbol{u}, d)$, kinetic energy $K(\boldsymbol{u})$ and external energy $\mathcal{P}(\boldsymbol{u})$ for a quasi-static problem, in which $\boldsymbol{u}$ is the displacement field and d is the damage field. According to the Griffith theory of brittle fracture, the energy required to open a unit area of crack surface equals the critical energy release rate G_c. Thus the fracture energy in the whole body equals the total crack surface area multiplied by G_c, (Equation (3)). The elastic strain energy can be calculated by integrating the elastic strain energy density over the whole domain, multiplied by an energy degradation function, a function of damage d (Equation (4)). After substituting all the components, the total potential energy $\mathcal{E}$ can be expressed in Equation (6).

$$\mathcal{E} = \Psi_c + \Psi_e(\boldsymbol{u}, d) + K(\boldsymbol{u}) - \mathcal{P}(\boldsymbol{u}) \tag{1}$$

$$\Psi_c = G_c \int_\Gamma dA \tag{2}$$

$$\Psi_e(\boldsymbol{u}, d) = \int_\Omega g(d)\psi_e(\boldsymbol{\varepsilon}(\boldsymbol{u}))dV \tag{3}$$

$$K(\boldsymbol{u}) = \int_\Omega \frac{1}{2}\rho\dot{\boldsymbol{u}}\dot{\boldsymbol{u}}dV \tag{4}$$

$$\mathcal{P}(\boldsymbol{u}) = \int_\Omega \boldsymbol{b}\cdot\boldsymbol{u}dV + \int_{\partial\Omega} \boldsymbol{t}\cdot\boldsymbol{u}dA \tag{5}$$

$$\mathcal{E} = \int_\Omega \left[g(d)\psi_e(\boldsymbol{\varepsilon}(\boldsymbol{u})) + \frac{1}{2}\rho\dot{\boldsymbol{u}}\dot{\boldsymbol{u}} - \boldsymbol{b}\cdot\boldsymbol{u}\right]dV - \int_{\partial\Omega} \boldsymbol{t}\cdot\boldsymbol{u}dA + G_c\int_\Gamma dA \tag{6}$$

The energetic degradation function $g(d)$ describes the degradation of the stiffness as the damage increases. It should satisfy the following conditions:

$$\begin{cases} g(0) = 1 \\ g(1) = 0 \\ g'(1) = 0 \end{cases} \tag{7}$$

In Miehe's formulation [5], the energetic degradation function has a quadratic form $g(d) = (1-d)^2$, and solving for the displacement field is a process of minimizing the total potential energy $\mathcal{E}$. It is not easy and straightforward to manipulate the crack surface integral, and that is where the phase field approximation of the crack surface comes into the picture.

2.2. Phase field approximation of a crack surface

The key idea of the phase field approximation of a crack length or area is to transform an integral over the surface to an integral over the whole domain. To better illustrate the idea, we use a $(0,1) \times (0,1)$ domain Ω as an example (Figure (1)). A crack line Γ exists at the middle of the domain, and other region is intact. Set the damage value as one on the crack surface Γ and let it degrade to 0 over a short distance controlled by a characteristic length l_0. The governing equation of damage over the whole domain is represented as Equation (8). Moreover, the approximated crack surface can be represented as an integral of a *crack surface density function* $\gamma(d, \nabla d)$ (Equation (9)), and a widely used form of $\gamma(d, \nabla d)$ is by Miehe (Equation (10)) [23]. A contour plot of phase field damage approximating crack length is presented in Figure (2). Γ-convergence can be proved by numerical examples: as l_0 decreases the approximated crack length approaches its real value (Figure (3)).

$$d(\boldsymbol{x}) - l_0^2 \Delta d(\boldsymbol{x}) = 0 \ \ in\ \Omega, \qquad d = 1\ on\ \Gamma \tag{8}$$

$$\int_{\Gamma} ds \cong \int_{\Omega} \gamma(d, \nabla d) dV \tag{9}$$

$$\gamma(d, \nabla d) = \frac{1}{2}\left(\frac{1}{l_0} d^2 + l_0 |\nabla d|^2\right) \tag{10}$$

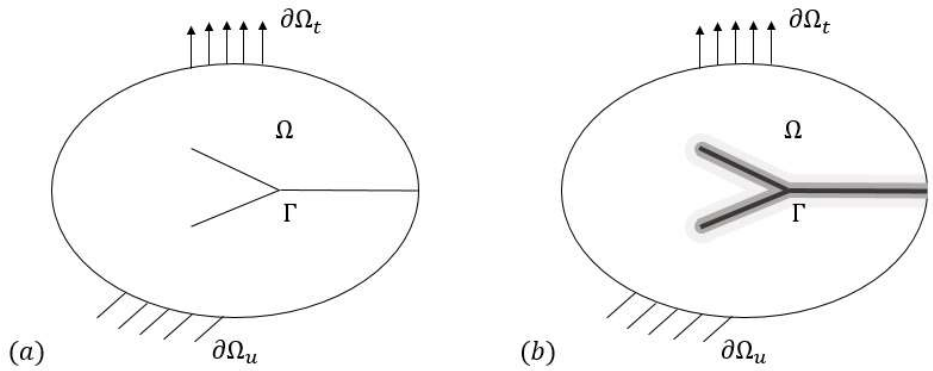

Fig. 1. (a) Discrete representation of a crack (2) Diffusive representation of a crack

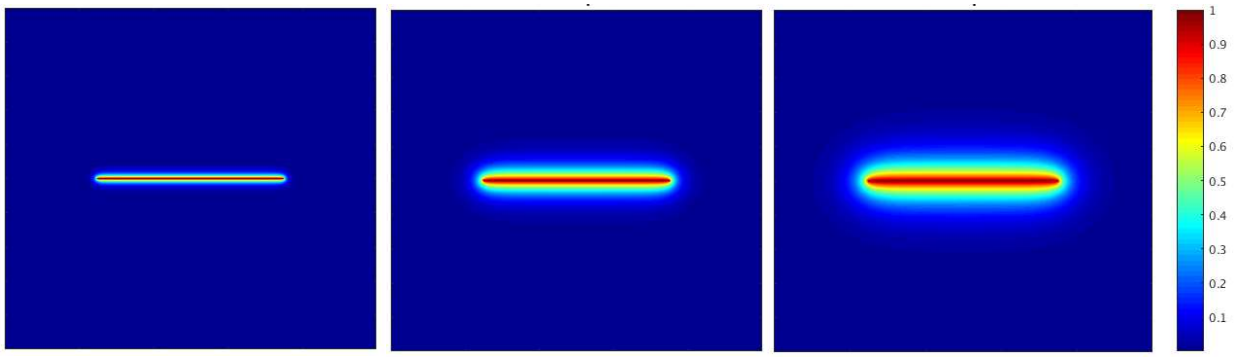

Fig. 2. Phase field damage to approximate a crack: l_0 increases from left to right (0 means intact and 1 means crack)

Replacing the phase field approximation of crack surface in ψ_c, the total potential energy becomes:

$$\mathcal{E} = \int_{\Omega} \left[g(d)\psi_e(\boldsymbol{\varepsilon}(\boldsymbol{u})) + \frac{1}{2}\rho \dot{\boldsymbol{u}}^2 - \boldsymbol{b} \cdot \boldsymbol{u} + G_c \gamma(d, \nabla d) \right] dV - \int_{\partial\Omega} \boldsymbol{t} \cdot \boldsymbol{u} dA \tag{11}$$

To solve for damage and displacement fields by minimizing the total potential energy, we take a staggered scheme, which updates the displacement field and the phase field in an alternative manner [14]. To solve for the displacement $\boldsymbol{u}$, we keep the damage field constant and optimize the displacement field (Equation (13)), and explicit dynamics is used to update displacement with time. After displacement field is obtained, an optimization over the damage field is carried out. The elastic strain energy density ψ_e is calculated and its maximum value in history $\mathcal{H}$ is used to avoid damage decreasing (Equation (14~15)) [24].

$$\boldsymbol{u} = ArgInf_u(\mathcal{E}(\boldsymbol{u}; d)) \tag{12}$$

$$d = ArgInf_d(\mathcal{E}(d; u, \mathcal{H})) \tag{13}$$

$$\mathcal{H} = max(\psi_e^+, \mathcal{H}) \tag{14}$$

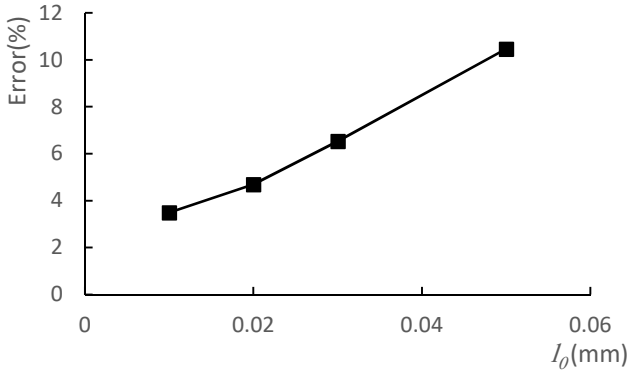

Fig. 3. Crack length relative error vs. characteristic length l_0

2.3. Asymmetric degradation of elastic strain energy

Our research focuses on the elasticity theory with no plastic deformation. Equation (11) considers an isotropic damage accumulation on the bulk energy. However, as Miehe [24] pointed out, for brittle materials, the damage accumulation in tension and compression are entirely different, for example, the concrete material has a compressive strength around 10 times of the tensile strength. There have been many models proposed to consider that anisotropic damage accumulation [14], and we adopt the one proposed by [14], in which the total elastic strain energy is separated into a compressive and tensile part through a spectral decomposition of strain tensor (Equation 17~18).

$$\psi_e = \psi_e^+ + \psi_e^- \tag{15}$$

where ψ_e^+ is the tensile part and ψ_e^- is the compressive part. Damage accumulation is only applied to the tensile part to degrade the tensile stiffness of the material:

$$\mathcal{E} = \int_\Omega \left[g(d)\psi_e^+(\boldsymbol{\varepsilon}) + \psi_e^-(\boldsymbol{\varepsilon}) + \frac{1}{2}\rho\dot{\boldsymbol{u}}^2 - \boldsymbol{b}\cdot\boldsymbol{u} + G_c\gamma(d, \nabla d) \right] dV - \int_{\partial\Omega} \boldsymbol{t}\cdot\boldsymbol{u} dA \tag{16}$$

Spectral decomposition of the strain tensor is used to calculate the positive and negative bulk energy (Equation (18)):

$$\varepsilon = \sum_{i=1}^{3} \varepsilon_i \boldsymbol{n}_i \otimes \boldsymbol{n}_i \tag{17}$$

$$\psi_e^\pm = \lambda\langle tr(\boldsymbol{\varepsilon})\rangle_\pm^2 + \mu \sum_{i=1}^{3} \langle\varepsilon_i\rangle_\pm^2 \tag{18}$$

where $\langle\cdot\rangle_+ = \frac{1}{2}(\cdot + |\cdot|)$ and $\langle\cdot\rangle_- = \frac{1}{2}(\cdot - |\cdot|)$. Then the stress field is obtained by taking the derivative of $\psi_e^\pm$ against strain $\boldsymbol{\varepsilon}$. In Equation (19~20), $\boldsymbol{\sigma}^+$ is the tensile part of the stress and $\boldsymbol{\sigma}^-$ is the compressive part of the stress.

$$\boldsymbol{\sigma} = \frac{\partial\psi_e}{\partial\boldsymbol{\varepsilon}} = g(d)\boldsymbol{\sigma}^+ + \boldsymbol{\sigma}^- \tag{19}$$

$$\boldsymbol{\sigma}^\pm = \lambda\langle tr(\boldsymbol{\varepsilon})\rangle_\pm + 2\mu \sum_{i=1}^{3} \langle\varepsilon_i\rangle_\pm \boldsymbol{n}_i \otimes \boldsymbol{n}_i \tag{20}$$

3. Implementation of the Phase Field Method into LS-DYNA

3.1. Matrix form of the equations

The implementation of phase filed method is done into LS-DYNA. This large-scale finite element software is chosen because it has a flexible user-defined element and user-defined material subroutines that allow a user to go above an

element level and do manipulations in the assembly level. In a dynamic problem, a kinematic term is added to the total potential energy (Equation (21)). By taking $\delta_{\boldsymbol{u}}\mathcal{E}=0$ we can obtain the equation of motion (Equation (22)):

$$\mathcal{E}=\int_{\Omega}\left[g(d)\psi_e\left(\boldsymbol{\varepsilon}(\boldsymbol{u})\right)+\frac{1}{2}\rho\dot{\boldsymbol{u}}^2-\boldsymbol{b}\cdot\boldsymbol{u}+G_c\gamma(d,\nabla d)\right]dV-\int_{\partial\Omega}\boldsymbol{t}\cdot\boldsymbol{u}dA \tag{21}$$

$$\int_{\Omega}\rho\ddot{\boldsymbol{u}}+g(d)K\boldsymbol{u}\,dV=\int_{\Omega}\boldsymbol{b}\cdot\boldsymbol{u}\,dV+\int_{\partial\Omega}\boldsymbol{t}\cdot\boldsymbol{u}\,dA \tag{22}$$

In matrix form, Equation (22) can be expressed in Equation (23) where the central difference method is used to discretize the time domain, and Δt is the time step. A lumped mass matrix is used (Equation (25)), and instead of assembling the stiffness matrix over the domain, the internal force vector is calculated (Equation (26)), where $[B_u]$ matrix transforms displacement to strain, and the stress vector has already considered the damage field.

$$[M]\{\ddot{u}\}+[\bar{K}]\{u\}=\{R^{ext}\} \tag{23}$$

$$[M]\frac{u_{n+1}-2u_n+u_{n-1}}{\Delta t^2}+[\bar{K}]u_n=\{R_n^{ext}\} \tag{24}$$

$$[M]=\begin{bmatrix}m_1 & & \\ & \ddots & \\ & & m_n\end{bmatrix} \tag{25}$$

$$[\bar{K}]\{u_n\}=\sum_{i=1}^{Ne}\int_{\Omega_i}[B_u]\{\bar{\sigma}_n\}\,dV=\{R_n^{int}\} \tag{26}$$

$$\{\bar{\sigma}_n\}=\{g(d)\sigma_n^{+}\}+\{\sigma_n^{-}\} \tag{27}$$

In the above equations, $[M]$ is the mass matrix, $[\bar{K}]$ represents the stiffness matrix, $\{R^{ext}\}$ is the external force vector, $\{R_n^{int}\}$ is the internal force vector, and Δt is the time step size. Combining all these pieces together, we can get the displacement for the next time step $\{u_{n+1}\}$ (Equation (28)), where M_i is the lumped mass at each node.

$$\{u_{n+1}\}=\frac{\Delta t^2}{M_i}(\{R_n^{ext}\}-\{R_n^{int}\})+2\{u_n\}-\{u_{n-1}\} \tag{28}$$

After the displacement field is solved, the damage field can be solved using $\delta_d\mathcal{E}=0$ and $\gamma(d,\nabla d)=\frac{1}{2}\left(\frac{1}{l}d^2+l|\nabla d|^2\right)$:

$$\delta_d\mathcal{E}=\int_{\Omega}[g'(d)\mathcal{H}+G_c\delta_d\gamma(d,\nabla d)]dV=0 \tag{29}$$

Replace the energy degradation function $g(d)$ by $(1-d)^2$:

$$\int_{\Omega}-2(1-d)\mathcal{H}+G_c\left(\frac{d}{l}-l\Delta d\right)dV=0 \tag{30}$$

Equation (31) shows the variational form of Equation (30):

$$\int_{\Omega}\left[\left(\frac{G_c}{l}+2\mathcal{H}\right)d\delta d+G_c l\nabla d\nabla\delta d\right]dV=\int_{\Omega}2\mathcal{H}\delta d\,dV \tag{31}$$

In matrix form, Equation (31) is replaced by Equation (32), where $[N]$ (Equation (34)) is the shape function for the damage field, $[B]$ (Equation (35)) calculates the gradients of the damage field, and $\mathcal{H}_i$ is the energy density value at each element:

$$\sum_{i=1}^{Ne}\int_{\Omega_i}[N]^T[N]\left(\frac{G_c}{l}+2\mathcal{H}_i\right)\{d\}+G_c l[B]^T[B]\{d\}\,dV=\int_{\Omega_i}2[N]^T\mathcal{H}_i\,dV \tag{32}$$

In a short form, Equation (32) can be expressed as Equation (33):

$$[K^d]\{d\}=\{F^d\} \tag{33}$$

Where $[K^d]$ and $\{F^d\}$ can be viewed as a stiffness and force matrix for the damage field and require assembly on the global level, and because of the linear property of the phase field damage equation, this equation can be directly solved without iterations.

$$[N] = [N_1 \quad \cdots \quad N_8] \tag{34}$$

$$[B] = \begin{bmatrix} B_{1,x} & \cdots & B_{8,x} \\ B_{1,y} & \cdots & B_{8,y} \end{bmatrix} \tag{35}$$

3.2. User-defined element and material subroutines in LS-DYNA

These equations in the previous section are implemented into LS-DYNA through a user-defined solid element subroutine (hexahedron element is used) and a user-defined material subroutine. The description of this implementation can serve as good instruction to future researchers.
Common blocks are used to save essential data for solving the damage of all the elements. Lots of the simulation related variable can be found in common blocks, for example, simulation time, the current coordinate of a node, and so on. Some aspects of the user-defined subroutines are described below:

Element connectivity data

This information is essential in finding connectivity data in an element to construct a Jacobian matrix. However, to the best of our knowledge, this information is not directly available. Thus we read in this information from a mesh file at the beginning of the simulation and save it to a common block. Since the read data action only needs to be done once, it does not impair the simulation time. One can read in any pre-processed data at the beginning of the simulation save it in a common block for future use. For example, in Discontinuous Galerkin method, one needs the surrounding element id of an interface element to calculate the element bounding force [25], and that information can be obtained through a MATLAB or python program and read into the user-subroutine.

Strains

In "usersld" subroutine, which is a user-defined solid element subroutine, strain values are not directly accessible, but strain rates at each time step for each integration point are provided. Thus, to obtain the strain values for energy calculation $\mathcal{H}$, we initiated a strain vector at the beginning of the simulation and saved it to a common block. At each time step, the strain rate is multiplied by the time step and added to the strain vector.

Setting up the global $[K^d]$ and $\{F^d\}$ matrices

In LS-DYNA explicit dynamics analysis and its user-defined subroutines, data are passed into the user subroutine in vectors. For example, if there are 1000 nodes in the model, and each pass only gives 136 nodal data, there will be a total of 8 pass of vectors with the last pass only has 48 nodal information. At each pass, we can only access the information within this pass. Thus, in order to create a global $[K^d]$ and $\{F^d\}$ matrices, we save the $[K^d]^e$ and $\{F^d\}^e$ of each element into a common block until we have these data of all the elements available. To calculate $[K^d]^e$ and $\{F^d\}^e$, Equation (32~35) are used respectively at full Gauss integration (8 integration points).

Solving the linear system of equations

This research uses the LAPACK and BLAS library in the Intel Fortran Math Kernal Library (MKL) to solve the linear system of equations. "dgetrf" and "dgetrs" subroutines are used to solve the linear system of equations using LU decomposition. "dsytrd", "dormtr" and "dsteqr" subroutines are used to calculate eigenvalues and eigenvectors calculation for separating the internal energy.

Calculation cycle

Although MKL is a very optimized library, it is still computationally expensive if we solve a large linear system of equations every time step. Thus, we propose a practical implementation that only solves the damage every N^{th} time steps. The number of the current time step number can be obtained from the sub routine. It is a parameter "ncycle" located in the common block "bk06". Note that a proper N value needs to be justified: although a larger N value saves more computational time, it should not be too large to cause inaccuracy.
The compilation is done using Intel compiler on the Linux system of Ohio Super Computer center system, and LS-DYNA 971_d_R11 is used. For Windows system, Intel Fortran compiler integrated with Visual Studio 2010 can be used. For each pass of data, the operation flow chart is shown in Figure 4.

4. Verifications

This section shows two numerical examples to verify the phase field method implementation with explicit dynamic analysis. Simulations with different calculation cycles are tests, and their accuracy is compared.

4.1. A bar under tensile load

This example simulates a bar under tensile loading. The setup is a bar under tensile loading. The bar has a length of 10mm and a width of 0.5mm, with one end fixed and one end under tensile loading (Figure 5). The displacement is placed

at a quasi-static rate of 3 mm/s. A section reduction of 2% is introduced to the midpoint of the bar so that damage can concentrate there. The simulation is run with a fixed characteristic length $l = 0.1mm$ and different N values. The mesh size around the mid-span is 0.05mm so that the l is twice the mesh size, as suggested by Miehe [5]. The material properties used are $E = 41000MPa, v = 0.17, G_c = 0.05\,N/mm$. The contour plot of damage is shown in Figure 6, and damage value is close to 1 at the mid-span and degrades to 0.24 outside the diffusive length. Figure 7 compares the force vs. time curve at different N values. It can be observed that when the for global behavior (force vs. time curve) that the result converges to a perfectly brittle failure as N approaches 1000, which is about $1/2000$ of the total time steps as it takes around 2 million time steps to reach the failure point. When 5% of error is allowed, $N = 5000$, $1/400$ of the total time steps may also be considered.

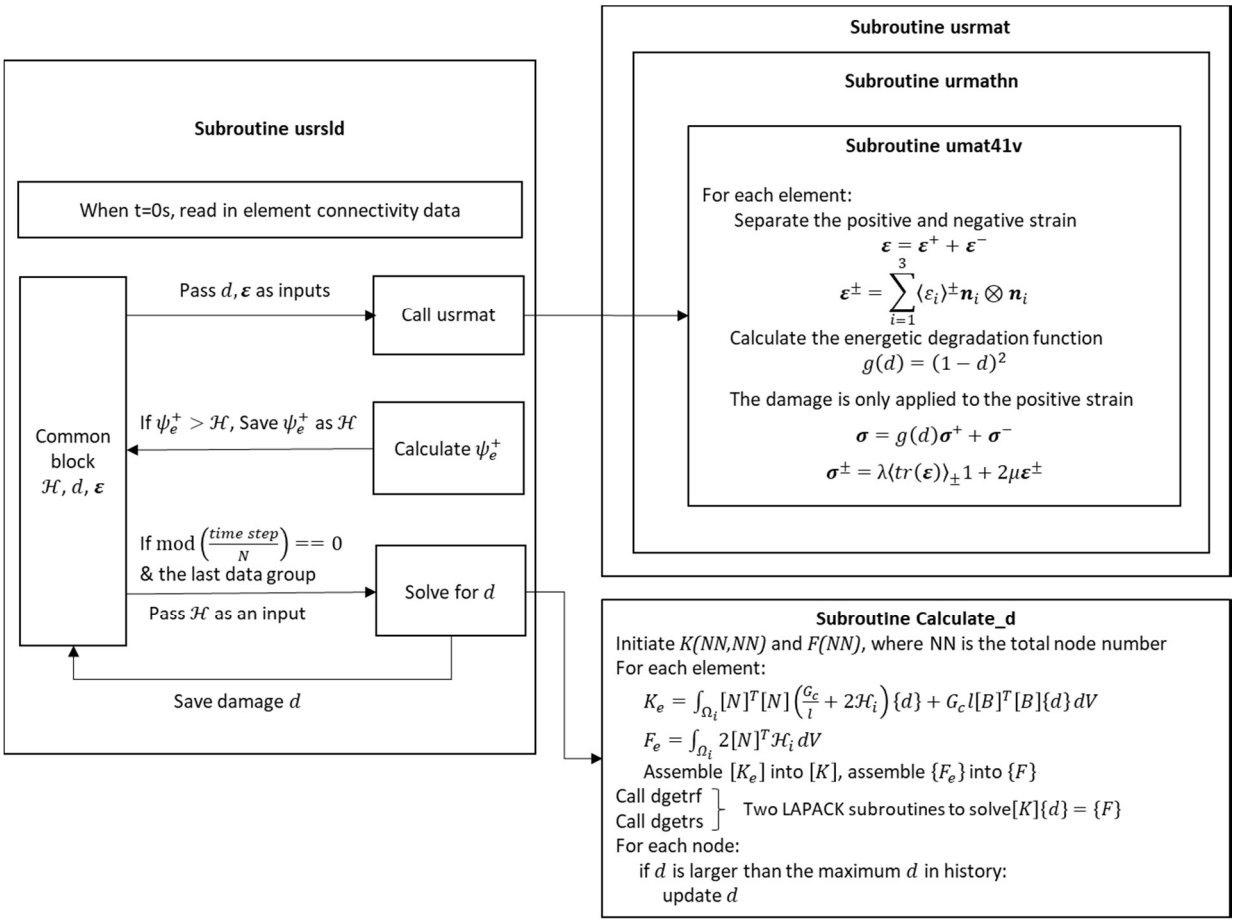

Fig. 4. Algorithm about the phase field method implementation into LS-DYNA

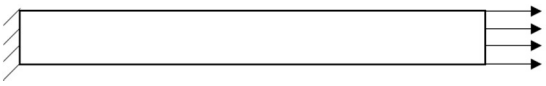

Fig. 5. Illustration of the problem

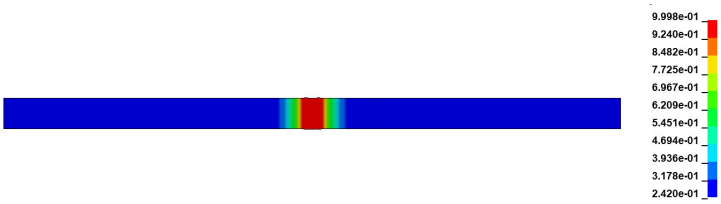

Fig. 6. A contour plot of the damage value in the bar

4.2. Mixed-mode crack propagation in a plate

To further verify the computational time's influence on the phase field's ability in predicting crack propagation, a mixed mode crack propagation simulation is carried out. A shear loading is applied to the top and bottom edge of the plate (Figure 8). The material property used is $E = 1000MPa, v = 0.2$, $\rho = 0.01\,g/mm^3$, $l = 1.0mm$ and $G_c = 0.01\,N/mm$. The loading speed in this case is 20 mm/s. It takes around 15600 time steps to finish the simulation and calculation cycles

of 20, 40, 60, and 80 are used. The resulting crack shapes are plotted in Figure 9 for different calculation cycles. To quantitively compare the effect of calculation cycles, the reaction force vs. time curves are plotted for different calculation cycles and they are summarized in Figure 10. It can be observed that when the calculation cycle N is less than 40, which is around $1/400$ of the total time steps, there is little difference between the force vs. time responses. This agrees with the finding in Section 4.1.

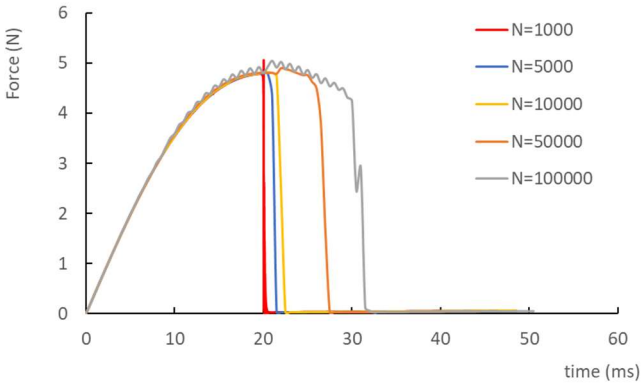

Fig. 7. Force vs. time curve for different calculation cycles

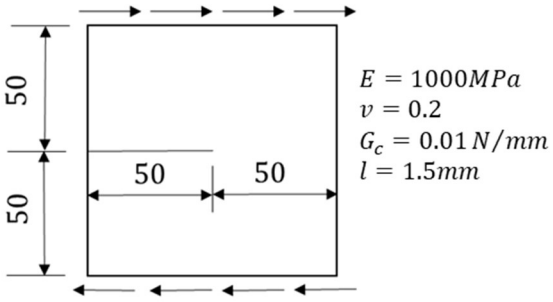

Fig. 8. An illustration of the mixed mode crack propagation simulation

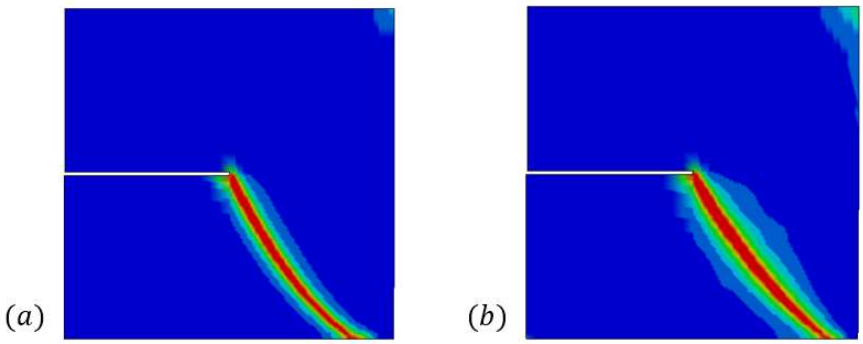

Fig. 9. (a) Crack shape for N=20 (b) Crack shape for N=80

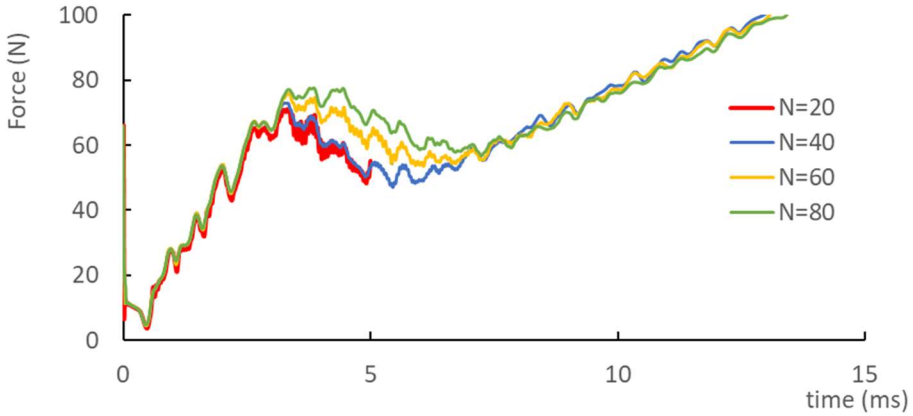

Fig. 10. Force vs. time curve for different damage calculation cycles

5. Conclusion

In this paper, an implementation of the phase field method with explicit time integration into LS-DYNA was elaborated. The process was made possible through a user-defined solid element subroutine and a user-defined material subroutine. The flexibility of the user-defined element subroutine in LS-DYNA allowed solving the linear system of equation for damage in the assembly level. The displacement and damage field were solved in a staggered manner, and the damage field was solved every N^{th} time step to save computational cost. Numerical simulations were presented to show the influence of skipping the damage field calculation for several time steps, and a calculation cycle N less than $1/400$ of the total time steps was suggested to guarantee accuracy.

Acknowledgment

We want to thank the Ohio Super Computer Center (OSC) for its computational resources, without which this research would be impossible.

Conflict of Interest

The authors declared no potential conflicts of interest with respect to the research, authorship and publication of this article.

Funding

The authors received no financial support for the research, authorship and publication of this article.

Nomenclature

$\mathcal{E}$	Total potential energy
Ψ_c, Ψ_e	Fracture energy, Elastic strain energy, respectively
Ψ_e^+, Ψ_e^-	Elastic strain energy due to tensile strain and compressive strain, respectively
K	Kinetic energy
$\mathcal{P}$	External energy
G_c	Critical energy release rate
$\boldsymbol{u}$	Displacement tensor
d	Damage
$g(d)$	Energetic degradation function
$\boldsymbol{b}$	Body force tensor
$\boldsymbol{t}$	Surface traction tensor
l_0	Phase field characteristic length
$\gamma(d, \nabla d)$	Crack density function
$\mathcal{H}$	Maximum Ψ_e^+ in history
$\varepsilon_i, \boldsymbol{n}_i$	i^{th} Eigenvalue and eigenvector of strain matrix, respectivley
λ, μ	Bulk and shear modulus, respectively
$\boldsymbol{\sigma}^+, \boldsymbol{\sigma}^-$	Tensile and compressive part of the stress, respectively
$[M]$	Mass matrix
$[\bar{K}]$	Stiffness matrix
$\{R^{ext}\}, \{R_n^{int}\}$	External and internal force vector, respectivley
$[N]$	Shape function
$[B]$	Matrix that calculates the gradients of damage field
$[B_u]$	Matrix that transforms displacement to strain
N	Calculation cycle
$[K^d]$	Stiffness for the damage field
$\{F^d\}$	Force vector for the damage field

Reference

[1] T. Rabczuk, T. Belytschko, Cracking particles: A simplified meshfree method for arbitrary evolving cracks, *International Journal for Numerical Methods in Engineering*, 61(13), 2004, 2316-2343.
[2] T. Rabczuk, G. Zi, S. Bordas, N.X. Hung, A simple and robust three-dimensional cracking-particle method without enrichment, *Computer Methods in Applied Mechanics and Engineering*, 199(37–40), 2010, 2437-2455.
[3] H.L. Ren, X.Y. Zhuang, C. Anitescu, T. Rabczuk. An explicit phase field method for brittle dynamic fracture, *Computers and Structures*, 217, 2019, 45-56.

[4] H. Ren, X. Zhuang, T. Rabczuk, Dual-horizon peridynamics: A stable solution to varying horizons, *Computer Methods in Applied Mechanics and Engineering*, 318, 2017, 762-782.
[5] W. J. Boettinger, J. A. Warren, C. Beckermann, A. Karma, Phase-Field Simulation of Solidification, *Annual Review of Materials Research*, 32(1), 2002, 163–194.
[6] G. A. Francfort, J.-J. Marigo, Revisiting brittle fracture as an energy minimization problem, *Journal of the Mechanics and Physics of Solids*, 46(8), 1998, 1319-1342.
[7] B. Bourdin, G. A. Francfort, J.-J. Marigo, The Variational Approach to Fracture, *Journal of Elasticity*, 91(1–3), 2008, 5–148.
[8] B. Bourdin, G. Francfort, J.-J. Marigo, Numerical experiments in revisited brittle fracture, *Journal of the Mechanics and Physics of Solids*, 48(4), 2000, 797-826.
[9] C. Miehe, F. Welschinger, M. Hofacker, Thermodynamically consistent phase-field models of fracture: Variational principles and multi-field FE implementations, *International Journal for Numerical Methods in Engineering*, 83(10), 2010, 1273–1311.
[10] A. Tabiei, W. Zhang, Cohesive element approach for dynamic crack propagation: Artificial compliance and mesh dependency, *Engineering Fracture Mechanics*, 180, 2017, 23-42.
[11] A. Tabiei, J. Wu, Development of the DYNA3D simulation code with automated fracture procedure for brick elements, *International Journal for Numerical Methods in Engineering*, 57(14), 2003, 1979–2006.
[12] X. F. Hu, B. Y. Chen, M. Tirvaudey, V. B. C. Tana, T. E.Tay, Integrated XFEM-CE analysis of delamination migration in multi-directional composite laminates, *Composites Part A: Applied Science and Manufacturing*, 90, 2016, 161-173.
[13] H. Ulmer, M. Hofacker, C. Miehe, Phase Field Modeling of Brittle and Ductile Fracture, *Gesellschaft fur Angewandte Mathematik und Mechanik*, 13(1), 2013, 533-536.
[14] M. Hofacker, C. Miehe, A phase field model of dynamic fracture: Robust field updates for the analysis of complex crack patterns, *International Journal for Numerical Methods in Engineering*, 93(3), 2013, 276–301.
[15] M. J. Borden, T. J. R. Hugh, C. M. Landis, C. V. Verhoosel, A higher-order phase-field model for brittle fracture: Formulation and analysis within the isogeometric analysis framework, *Computer Methods in Applied Mechanics and Engineering*, 273, 2014, 100-118.
[16] C. Hesch, K. Weinberg, Thermodynamically consistent algorithms for a finite-deformation phase-field approach to fracture, *International Journal for Numerical Methods in Engineering*, 99(12), 2014, 906-924.
[17] C. V. Verhoosel, R. de Borst, A phase-field model for cohesive fracture, *International Journal for Numerical Methods in Engineering*, 96(1), 2013, 43-62.
[18] S. Zhou, X. Zhuang, T. Rabczuk, A phase-field modeling approach of fracture propagation in poroelastic media, *Engineering Geology*, 240, 2018, 189-203.
[19] M. A. Msekh, N. H. Cuong, G. Zi, P. Areias, X. Zhuang, T. Rabczuk, Fracture properties prediction of clay/epoxy nanocomposites with interphase zones using a phase field model, *Engineering Fracture Mechanics*, 188, 2018, 287-299.
[20] G. Molnár and A. Gravouil, 2D and 3D Abaqus implementation of a robust staggered phase-field solution for modeling brittle fracture, *Finite Elements in Analysis and Design*, 130, 2017, 27-38.
[21] G. Liu, Q. Li, M. A. Msekh, Z. Zuo, Abaqus implementation of monolithic and staggered schemes for quasi-static and dynamic fracture phase-field model, *Computational Materials Science*, 121, 2016, 35–47.
[22] E. G. Kakouris, S. P. Triantafyllou, Phase-field material point method for brittle fracture, *International Journal for Numerical Methods in Engineering*, 112(12), 2017, 1750-1776.
[23] C. Miehe, M. Hofacker, F. Welschinger, A phase field model for rate-independent crack propagation: Robust algorithmic implementation based on operator splits, *Computer Methods in Applied Mechanics and Engineering*, 199(45–48), 2010, 2765-2778.
[24] C. Miehe, F. Welschinger, M. Hofacker, Thermodynamically consistent phase‐field models of fracture: Variational principles and multi-field FE implementations, *International Journal for Numerical Methods in Engineering*, 83(10), 2010, 1273-1311.
[25] W. Zhang, A. Tabiei, D. French, Comparison Between Discontinuous Galerkin Method and Cohesive Element Method: On the Convergence and Dynamic Wave Propagation Issue, *International Journal of Computational Methods in Engineering Science and Mechanics*, 19(5), 2018, 363-373.

ORCID iD

Wenlong Zhang https://orcid.org/0000-0002-1780-5425
Ala Tabiei https://orcid.org/0000-0002-4749-4525