Domain decomposition methods and acceleration techniques for the phase field fracture staggered solver

Johann Rannou, Christophe Bovet

▶ **To cite this version:**

Johann Rannou, Christophe Bovet. Domain decomposition methods and acceleration techniques for the phase field fracture staggered solver. 2023. hal-03938084

HAL Id: hal-03938084
https://hal.science/hal-03938084

Preprint submitted on 13 Jan 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Received: Added at production | Revised: Added at production | Accepted: Added at production

DOI: xxx/xxxx

RESEARCH ARTICLE

Domain decomposition methods and acceleration techniques for the phase field fracture staggered solver

Johann Rannou | Christophe Bovet

[1]Université Paris-Saclay, ONERA, Matériaux et Structures, 92322 Châtillon, France

Correspondence
Johann Rannou Email: johann.rannou@onera.fr
Christophe Bovet. Email: christophe.bovet@onera.fr

Summary

The phase field modeling of fracture is able to simulate the nucleation and the propagation of complex crack patterns. However, the relatively small internal lengths that are required usually lead to very fine meshes and high computational costs, especially for three-dimensional applications. Unstable crack propagations are regularized through an implicit dynamics framework potentially leading to a very large variation of time steps switching from a quasi-static regime to a dynamic one. Since the fracture is a non-conservative and path dependent phenomenon, this strategy is well adapted. It also provides a better control on the dissipated energy than adding an artificial viscosity. It however increases the computational cost of the method.
To reduce the time to solution and exploit modern supercomputers, we propose a domain decomposition framework and acceleration techniques for the phase field fracture staggered solver. The displacement subproblem and the phase field one are solved with parallel domain decomposition solvers. Dual domain decomposition methods provide low cost preconditioner well adapted to the phase field subproblem. For displacement subproblems undergoing unstable crack propagations, primal domain decomposition methods are preferred to be less sensitive to the treatment of floating substructures.
Preconditioners performances are assessed and scalability studies over academic test cases, up to 324 subdomains, are presented. Finally, the robustness of the approach is illustrated on two semi-industrial simulations.

KEYWORDS:
phase field fracture, damage, domain decomposition, FETI, BDD

1 | INTRODUCTION

The phase field modeling of fracture, also known as variational approach of fracture mechanics, has been widely developed during the last decade. This approach appears to be a very promising tool to bridge damage mechanics (that can be used to

determine the onset of failure) and fracture mechanics of quasi-brittle materials (which is used to predict the crack evolution). Variational approach of fracture mechanics has been initially introduced as a novel technique to handle Griffith theory [1,2]. More recently, strong links to damage mechanics have been established. While phase field models are not constitutive laws in the sense of a strict local relationship between strain and stress, the concept of stress threshold can be exhibited [3,4]. It thus provides the essential ingredient of most classical damage models that allows the prediction of crack nucleation. A dimensional analysis as well as more detailed analysis of specific phase field models establish a link between the energy propagation criterion of the Griffith theory, the stress threshold and the phase field internal length. Depending on the kind of analysis, this internal length can be viewed either as a numerical parameter that has to tend to 0 within the framework of brittle fracture and Γ-convergence, or as a material parameter that is related to either a microstructural feature, the Irwin length or the characteristic size of a process zone.

Either way, apart from a few exceptions such as geomaterials or woven fibers composites who can have a relatively large microstructures compared to the size of the structure of interest, the internal length is very small. Since this length influences the gradient of the phase field (i.e. the damage variable), the discretization generally has to be very small, thus leading to very expensive numerical models. In this work, phase field problems are solved using a staggered iteration scheme [2], which is known to be relatively robust. However, this staggered scheme is also quite expensive since it does not use tangent descent directions like Newton-Raphson solvers [5].

A last source of numerical cost in the present work is the way we choose to handle instabilities. Instabilities occur when the structure, while dissipating the cracking surface energy, can not store the remaining energy coming from the work of external forces as strain energy. This additional energy is thus transformed into kinetic energy and the response becomes dynamic. Several techniques can be used to handle this. The first technique is to add some kind of viscous regularization with the risk to artificially introduce a too large amount of dissipation. The second one is to use a continuation solver to find a quasi-static response [6], but it will be further impossible to use time-dependent constitutive laws. Finally, one can solve the problem in a transient dynamics framework. This latter solution is the one selected in this work. This requires the solver to be able to adapt its time stepping from a quasi-static regime to a “fast” dynamic one, thus requiring several decades between the maximum and minimum time steps. Thus, it possibly increases by a large amount the cost of the numerical simulation. Acceleration strategies and parallel frameworks consequently become essential.

To improve the overall computational efficiency, the first strategy is to reduce the number of subproblems to be solved by optimizing the minimization algorithm [7,8] or the time integration scheme [9]. The second one is to accelerate the construction and resolution of the tangent systems. Several authors have been investigating the parallelization of phase field simulations with staggered scheme. Most of them consider explicit schemes that remove the need for an efficient linear solver. The parallelization for explicit schemes has been investigated using Abaqus [10,11] with shared memory parallelism. A massively parallel GPU

implementation with time step adaptation is proposed [12]. Regarding implicit schemes, a Krylov solver associated with an algebraic multigrid preconditioner with near nullspace correction using Petsc is investigated [7]. The nullspace correction is needed to improve the interpolation operators for elasticity problems [13]. A parallel implementation based on the Gradient Projection Conjugate Gradient of Petsc associated with an incomplete Cholesky factorization as preconditioner was also proposed [14].

Non overlapping domain decomposition methods, such as the Balancing Domain Decomposition (BDD) [15] or the Finite Element Tearing and Interconnecting (FETI) [16], their constrained variants FETI-DP [17] and BDDC [18] or multipreconditioned one AMPFETI [19,20,21], provide a good alternative to multigrid methods. Thanks to the combination of direct and iterative solvers, domain decomposition methods are usually more robust than traditional iterative solvers. Stiffness scaling [22] and multipreconditioned variants [19,20,21] provide robustness against material heterogeneity misplaced with respect to the domain decomposition interface. This material heterogeneity is characterized by large variation of diagonal coefficients of the stiffness matrix. This robustness is essential since the degradation function of the phase field approach produces similar effects. Also, one crucial point of FETI and BDD methods is that the local operators nullspaces provide a coarse problem essential for the scalability. In this context, the transition between quasi-static regime to a highly dynamic one makes the detection of local operators nullspace complexe. This point requires a specific attention even if recent progress have been made in this field [23].

The aim of this paper is to present the parallel framework based on domain decomposition methods, used to accelerate the staggered resolution algorithm of the phase field fracture formulation. The article is organized as follows: Section 2 gives a short presentation of the phase field fracture formulation as well as the specificities of the dynamics resolution; then Section 3 recalls the FETI and BDD methods and highlights the specificities for both the phase field and the displacement subproblems as well as the issues caused by the dynamic/quasi-static transition. Acceleration techniques are discussed in Section 4. Scalability results on academic benchmarks are provided in Section 5 and Section 6 presents two semi-industrial simulations. Finally, Section 7 concludes the paper.

2 | PHASE FIELD FRACTURE AND STAGGERED RESOLUTION ALGORITHM

In this section, the basics of the phase field approach for brittle and quasi-brittle fracture are first reminded. In this work, we focus on the so-called AT2 model (named after the work of Ambrosio and Tortorelli on Γ-convergence [24]) which has been historically used in the seminal work of Bourdin *et al.* [2]. The extension to AT1 should however be straightforward. Secondly, we give some insights on the staggered solution strategy and its related code-coupling implementation. We also emphasize the way we use an implicit dynamics integration scheme to regularize the fracture process during unstable phases.

2.1 | The basics of phase field models for quasi-brittle fracture

Phase field models for fracture mechanics are closely related to the *variational approach to brittle fracture*[1]. They aim at representing the crack path with a continuous scalar field (the so-called phase field). Both displacement field and phase field have to fulfill the Hamilton's principle[25] in accordance with the Griffith theory[26]. In this framework, the total potential energy formally writes as the following functional:

$$\mathcal{E}(u,\Gamma) = \underbrace{\int_{\Omega/\Gamma} \psi_0\left(\varepsilon\left(u\right)\right) dV}_{\Psi_s} + \underbrace{G_c \int_{\Gamma} dS}_{\Psi_c} \tag{1}$$

where Ψ_s is the strain energy which is computed over the whole domain Ω from the elastic energy density ψ_0 which depends on the linearized strain tensor $\varepsilon(u)$, u being the unknown displacement field. The dissipated surface energy corresponding to the cracking process is denoted Ψ_c. It is integrated over the crack surface Γ which is a parameter of the functional in this formal context. The so-called critical energy release rate G_c is supposed to be a material property in the quasi-brittle fracture theory. The kinetic energy writes:

$$\mathcal{K}(\dot{u}) = \int_{\Omega} \frac{1}{2}\rho \dot{u} \cdot \dot{u}\, dV \tag{2}$$

where ρ is the material density and $\dot{u}$ is the velocity field.

Because the functional parameter Γ appears as the integral domain in the expression of Ψ_c, its evaluation in the context of numerical resolutions through discretization techniques such as finite elements can become complex if not possible. The main idea of the *variational approach to brittle fracture*[2] is to approximate the surface integral in Ψ_c by a volume integral:

$$\Psi_c = G_c \int_{\Gamma} dS \approx \underbrace{G_c \int_{\Omega} \gamma_l(\phi, \nabla\phi) dV}_{\Gamma_l} \tag{3}$$

The unknown measure of the surface Γ is replaced by a surface density functional Γ_l which depends on the continuous scalar field $\phi \in [0,1]$ defined on Ω and its gradient. When $\phi = 0$, the material is locally unbroken while the material is locally broken when $\phi = 1$. The Γ-convergence concept is used to enforce the approximation (3)[2,24]. In particular in the context of the so-called AT2 model, $\gamma_l(\phi, \nabla\phi)$ is defined as:

$$\gamma_l(\phi, \nabla\phi) = \frac{1}{2}\left(\frac{1}{l}\phi^2 + l(\nabla\phi)^2\right) \tag{4}$$

where l is a length parameter. In this situation, it can be proven that[2], if $\phi = 1$ on Γ and $\nabla\phi \cdot n = 0$ on $\partial\Omega$, then:

$$\lim_{l\to 0} \int_{\Omega} \gamma_l(\phi, \nabla\phi) dV = \int_{\Gamma} dS \tag{5}$$

The following functional $\mathcal{E}_l(u,\Gamma)$ will now be used in place of $\mathcal{E}(u,\Gamma)$:

$$\mathcal{E}_l(u,\phi) = \int_\Omega \psi(\varepsilon(u),\phi)dV + G_c \int_\Omega \gamma_l(\phi,\nabla\phi)dS \tag{6}$$

It is worth noting that $\psi_0(\varepsilon(u))$ as been replaced by $\psi(\varepsilon(u),\phi)$ which accounts for the presence of the crack through the field ϕ. These two quantities are related by:

$$\psi(\varepsilon(u),\phi) = g(\phi)\psi_0^+ + \psi_0^- \tag{7}$$

The degradation function $g(\phi)$ accounts for the effect of the ϕ field on the elastic response of the material. The elastic energy ψ_0 is split into a positive part ψ_0^+ and a negative one ψ_0^- which are related respectively to the positive and negative principal strain components. This split has been proposed[27] to account for the fact that in some materials, cracks are not likely to propagate in a compression state. In the AT2 model $g(\phi)$ is defined as $g(\phi) = (1-\phi)^2$.

The whole problem can now be written following the Hamilton's principle which requires to define an action-integral over the time interval $[t_0, t_1]$:

$$\mathcal{A} = \int_{t_0}^{t_1} \left(\mathcal{E}_l - \mathcal{K} - \mathcal{P}\right) dt \tag{8}$$

where $\mathcal{P}$ is the work of the external forces. The Hamilton's principle $\delta\mathcal{A} = 0$ yields the strong form of the model:

$$\nabla\sigma = \rho\ddot{u} \tag{9}$$

$$\frac{G_c}{2}\left(\frac{1}{l}\phi - l\Delta\phi\right) = (1-\phi)\psi_0^+ \tag{10}$$

with the following boundary conditions:

$$\sigma\cdot n = t \quad \text{on } \partial\Omega_t \tag{11}$$

$$u = u_d \quad \text{on } \partial\Omega_d \tag{12}$$

$$\nabla\phi\cdot n = 0 \quad \text{on } \partial\Omega \tag{13}$$

where t and u_d are the prescribed surfacic traction density and prescribed displacement. It is also worth mentioning that if σ were computed from equation (7) as $\sigma = \partial\Psi/\partial\varepsilon = g(\phi)$, equation (9) would become non-linear. We thus use the hybrid formulation[28] in which the stress in computed as:

$$\sigma = g(\phi)\frac{\partial\Psi_0}{\partial\varepsilon} \tag{14}$$

For a comprehensive derivation of the strong form, the reader is referred to[27] or[29]. The main advantage of this formulation is that both unknowns u and ϕ and their gradients are continuous fields defined over the whole volume. They can then be handled by classical discretization techniques such as the finite elements method.

The structure of equation (10) as well as the effect of ϕ on the elastic response expressed in (7) suggest that phase field models are related to gradient enhanced damage models used to circumvent the issues arising from softening constitutive behaviors [30,31]. In [32], strong relations between these two families of models are established. Actually, their differences lie more in their origin than in the structure of their equations. As mentioned before, phase field models and the variational approach to brittle fracture rely on the Griffith theory of the linear elastic fracture mechanics where the material is considered as elastic and the crack is viewed as a macroscopic feature in the structure. On the opposite, damage models are fundamentally inelastic constitutive models. They are designed to account for the effect of microscopic defects at the macroscopic scale. When these defects grow above a certain amount, damage models naturally exhibit a *softening* regime which goes up to total failure (null stress and null stiffness). These models then suffer from the so-called *loss of ellipticity* (or *localization*) issue [33] and need to be regularized by techniques such as viscous regularization [?] or non-local damage [34,30] models which use an internal length parameter.

Damage models are designed as inelastic constitutive laws (i.e. satisfying the local state principle [35]). They usually rely on stress (or strain) criteria to define the onset of damage and on the definition of a damage kinetics. The required parameters can be identified on specific tests as long as the samples exhibit a homogeneous response [36]. On the other hand, the phase field models are not designed as constitutive laws. The apparent softening behavior induced by equation (7) is not related to any experiment led on a homogeneous sample since the Γ-convergence tends to make this apparent softening to occur on a vanishing zone surrounding the crack surface. The evolution of ϕ is not considered as a damage kinetics driven by the material local state but as the consequence of the global minimization of $\mathcal{E}_l$. While stress thresholds or stress peeks can be explicitly defined as more or less complex formulations in damage models, in phase field models they are only the indirect consequences of a global minimization problem. The only material parameters that are required by brittle fracture phase field models are the elastic moduli and the critical energy release rate. The internal length is considered to be a numerical parameter that has not to be as small as possible. The internal length can however recover a sort of physical meaning when one is interested in quasi-brittle material when the notion of stress threshold has to be accounted for. In this situation, the link that exists between the internal length, the toughness, the young modulus, and the stress threshold can be exhibited [3].

In non-local damage models, toughness is not formally controlled since the Γ-convergence concept has not been introduced. There is therefore no direct link between the dissipated volumetric energy density defined by the constitutive behavior, the internal length and the critical energy release rate (which is an energy per unit area).

2.2 | Finite element discretization

The finite element discretization of the strong form (9)-(10) with boundary conditions (11)-(13) is easily obtained through a classical variational analysis:

$$\int_\Omega \sigma : \varepsilon(\delta u)\, d\Omega + \int_\Omega \rho \ddot{u} \cdot \delta u\, d\Omega - \int_{\partial\Omega} f^d \cdot \delta u\, dS = 0 \quad \forall \delta u \in \mathcal{U}_0 \tag{15}$$

$$\int_\Omega \left(-2(1-\phi)\psi_0 + \frac{G_c \phi}{l}\right) \delta\phi + l G_c \nabla\phi \cdot \nabla\delta\phi\, d\Omega = 0 \quad \forall \delta\phi \in \Phi_0 \tag{16}$$

where δu and $\delta\phi$ are the displacement and phase field test functions while $\mathcal{U}_0$ and Φ_0 are the admissible spaces. In this work, we use first order finite elements to interpolate these weak forms. We write $N(x)$ the shape functions and $B(x)$ and $B^s(x)$ their gradient and symmetrizing gradient so that :

$$\begin{aligned} u_i(x) &= \sum_{I\in\mathcal{N}} N_I(x) U_{Ii} \quad \phi(x) = \sum_{I\in\mathcal{N}} N_I(x)\Phi_I \\ \varepsilon_{ij}(x) &= \sum_{I\in\mathcal{N}} B^s_{Ij}(x) U_{Ii} \quad \frac{\partial\phi(x)}{\partial x_j} = \sum_{I\in\mathcal{N}} B_{Ij}(x)\Phi_I \end{aligned} \tag{17}$$

where U_{Ii} and Φ_I are the degrees of freedom related to node I, and space direction i. In the following, the summation convention of duplicated indices will be used. The semi-discretized (*i.e.* not discretized in time) weak formulation then writes:

$$\int_\Omega \sigma_{ij} B^s_{Ij} \delta U_{Ii}\, d\Omega + \int_\Omega \rho \ddot{U}_{Ii} N_J N_I \delta U_{Ji}\, d\Omega - \int_{\partial\Omega} f^d_i N_I \delta U_{Ii}\, dS = 0 \quad \forall \delta u \in \mathcal{U} \tag{18}$$

$$\int_\Omega \left(-2(1-N_J\Phi_J)\psi_0 + \frac{G_c}{l} N_J \Phi_J\right) N_I \delta\Phi_I + l G_c B_{Ii}\phi_I B_{Ji}\delta\Phi_J\, d\Omega = 0 \quad \forall \delta\phi \in \Phi_0 \tag{19}$$

where $\ddot{U}_{Ii}$ are acceleration degrees of freedom. The semi-discretized residuals write:

$$R^u_{Ii} = \int_\Omega \sigma_{ij} B^s_{Ij}\, d\Omega + \int_\Omega \rho N_J N_I \ddot{U}_{Ji}\, d\Omega - \int_{\partial\Omega} f^d_i N_I\, dS \tag{20}$$

$$R^\phi_I = \int_\Omega \left(-2(1-N_J\Phi_J)\psi_0 + \frac{G_c}{l} N_J\Phi_J\right) N_I + l G_c B_{Ii}\phi_I B_{Ji}\, d\Omega \tag{21}$$

2.3 | Numerical solution using a staggered scheme

The finite element discretization of the phase field model leads to a non-linear two-fields problem that can be classically solved using either a monolithic solver[30,37] or an alternate minimization scheme as initially proposed in[2]. According to the current literature, it is generally admitted that monolithic schemes associated to Newton solvers perform poorly. This is due to the non-convexity of the energy functional $\mathcal{E}_l$ with respect to the displacement and damage unknowns. Several strategies can however improve the robustness of this approach as emphasized in[5,38,39]. In this work, we have selected an alternate minimization solver which consists in solving alternatively the mechanical equation (9) and the phase field equation (10) as illustrated in Figure 1. While the convergence can be slow to achieve, this strategy is known to be more robust since each subproblem taken separately is convex. Also, the domain decomposition methods used in this work take advantage of this staggered scheme. Separate tangent

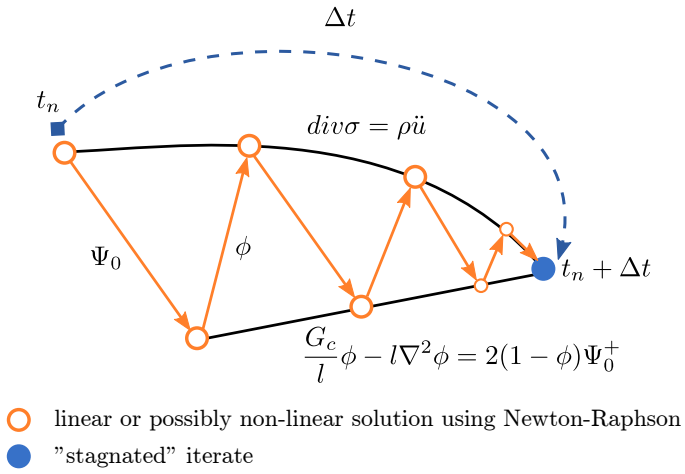


Figure 1 Schematic of the alternate minimization solver for one loading increment (from time t_n to t_{n+1})

systems are symmetric positive definite allowing the use of the conjugate gradient iterative solver (and its multipreconditioned variants).

Figure 1 illustrates the iterative process of the staggered scheme over a time increment from t_n to $t_{n+1} = t_n + \Delta t$. A solution of the equilibrium subproblem is first performed considering the field ϕ fixed. This subproblem can be non-linear and be solved locally with Newton-like methods. This is useful if one need to account for nonlinearities arising from other features in the whole model such as cohesive zones or contact for instance. In this work however, all applications deal with linear equilibrium problems. Then the field Ψ_0^+ is post-processed and prescribed to the phase field subproblem (which is linear in the case of AT2 variant). The process continues until a stagnation criterion is achieved as detailed in Section 2.4 and algorithm 1.

To integrate equation (18) over time, we use the classical α-method[40] which uses the Newmark relations to link the displacement dofs $U_{Ii}^{t_n}$ with the approximation of the velocity $\dot{U}_{Ii}^{t_n}$ and the acceleration $\ddot{U}_{Ii}^{t_n}$ degrees of freedom:

$$U_{Ii}^{t_n+\Delta t} = \underbrace{U_{Ii}^{t_n} + \Delta t \dot{U}_{Ii}^{t_n} + \frac{\Delta t^2}{2}(1-2\beta)\ddot{U}_{Ii}^{t_n}}_{\widetilde{U}_{Ii}^{t_n}} + \beta \Delta t^2 \ddot{U}_{Ii}^{t_n+\Delta t} \tag{22}$$

$$\dot{U}_{Ii}^{t_n+\Delta t} = \dot{U}_{Ii}^{t_n} + \Delta t(1-\gamma)\ddot{U}_{Ii}^{t_n} + \Delta t \gamma \ddot{U}_{Ii}^{t_n+\Delta t} \tag{23}$$

where superscripts t_n and $t_n + \Delta t$ relate the nodal quantities to their values at the beginning and at the end of the current time step respectively. $\widetilde{U}_{Ii}^{t_n}$ is the predictor which is completely defined from the converged previous increment. The α-method also slightly modifies the residual (20) into:

$$R_{Ii}^u = \int_\Omega \left((1-\alpha)\sigma_{ij}^{t_n+\Delta t} + \alpha\sigma_{ij}^{t_n}\right) B_{Ij}^s \, d\Omega + \int_\Omega \rho N_J N_I \frac{1}{\beta \Delta t^2}(U_{Ji}^{t_n+\Delta t} - \widetilde{U}_{Ji}^{t_n}) \, d\Omega - \underbrace{\int_{\partial\Omega} \left((1-\alpha) f_i^{d\,t_n+\Delta t} + \alpha f_i^{d\,t_n}\right) N_I \, dS}_{F_{\text{ext}}^{t_n+\alpha\Delta t}} \tag{24}$$

The integrator parameters β and γ in (22) and (23) are linked to the parameter $\alpha \in [0, 0.3]$ of equation (24) such that $\beta = (1+\alpha)^2/4$ and $\gamma = 1/2 + \alpha$. Note that σ^{t_n} is known from the previous converged increment.

The full tangent operator of the full problem is obtained through a first order Taylor expansion of the residuals (24) and (21) with respect to U_{Ii} and ϕ_I. Since the staggered scheme does not require the computation of the coupling terms $\partial R^u/\partial\phi$ and $\partial R^\phi/\partial u$, the two required operators can thus be written as:

$$K^{uu}_{IiKk} = \frac{\partial R^u_{Ii}}{\partial U_{Kk}} = \int_\Omega B^s_{Ij} \frac{\partial \sigma_{ij}}{\partial \varepsilon_{kl}} B^s_{Kl}\, d\Omega \tag{25}$$

$$K^{\phi\phi}_{IK} = \frac{\partial R^\phi_I}{\partial \phi_K} = \int_\Omega \left(2\psi_0 + \frac{G_c}{l}\right) N_K N_I + l G_c B_{Ii} B_{Ki}\, d\Omega \tag{26}$$

Following expression (14), we have:

$$\frac{\partial \sigma_{ij}}{\partial \varepsilon_{kl}} = (1-\phi)^2 C_{ijkl} \tag{27}$$

where C_{ijkl} is the Hooke operator. The mass matrix finally writes:

$$M_{IiKk} = \int_\Omega \rho N_I N_K \delta_{ik}\, d\Omega \tag{28}$$

where δ_{ij} stands for the Kronecker delta. With all of this, we can write the two linear systems we have to solve at each iteration of the staggered algorithm:

$$\left(\frac{1}{\beta \Delta t^2} M + (1-\alpha) K^{uu}\right) U^{t_n+\Delta t} = F_{\text{ext}}^{t_n+\alpha\Delta t} + \frac{1}{\beta\Delta t^2} M \widetilde{U}^{t_n} - \alpha K^{uu} U^{t_n} \tag{29}$$

$$K^{\phi\phi} \Phi^{t_n+\Delta t} = F^\phi_{\text{ext}} \tag{30}$$

where:

$$F^\phi_{\text{ext}\,I} = \int_\Omega 2\psi_0^+ N_I\, d\Omega \tag{31}$$

2.4 | Dynamics to regularize unstable fracture

In the applications discussed further in this paper, the overall loading is applied slowly. The structure response is then most of the time quasi-static. However, the fracture process may be unstable and the crack propagation may occur in one or several short dynamic jerks. Instabilities occur when the structure, submitted to an increasing work of external forces, can not store as strain energy all the remaining additional energy available after crack propagation. Because of the energy conservation principle, this additional energy is thus inevitably transformed into kinetic energy. The response becomes dynamic and a sudden drop in the force-displacement curve occurs as illustrated in Figure 2. At a large time scale the solution thus exhibits a lack of smoothness that is difficult to handle with Newton-based or staggered implicit solvers. At a low enough time scale however, the slope of the response recovers its smoothness thanks to inertia and is thus easier to be captured with a Newton-Raphson solver or a staggered

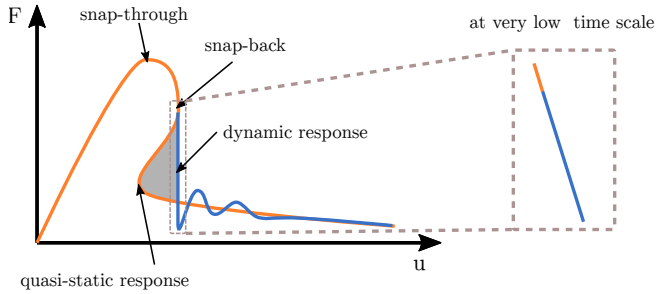


Figure 2 Schematic of a dynamic response arising after a structural instability. The orange curve corresponds to a quasi-static regime (*i.e.* kinetic energy is negligible) while the blue curve corresponds to a dynamic regime. A quasi-static response after the snap-back may however exist and can then be captured by a continuation algorithm.

scheme. Figure 2 also shows the quasi-static response that would be captured with a continuation algorithm thus exhibiting a snap-back. This quasi-static response corresponds to a crack growing under a decreasing work of external forces.

As mentioned previously in the introduction, these instabilities are regularized using an implicit dynamics solver. The main difficulty in this approach holds in that the precise timing of instabilities is not known a priori. In practice, their occurrence is characterized by a very large number of staggered iterations during a load increment, or even by a non convergence. To work around this difficulty, the time-stepping adaptation rules have to be tuned so that a substancial time step decrease, usually of several order of magnitude, is allowed. Thus, the algorithm adapts its time steps to work at a proper time scale where the solution can be regularized by the inertia as illustrated in Figure 3.

When the geometry is simple enough, it can happen that when letting the staggered solver iterate a lot in the instability neighborhood, a balanced solution can be found within a single load increment. In such situations, this solution often corresponds to the complete failure of the whole structure or of a large region between two geometrical features (a ligament between two holes for instance). Because the fracture is a non-conservative and path dependent phenomenon, and despite the satisfaction of the final equilibrium, that solution may not correspond to the one that would have been obtained with more progressive time-stepping. To avoid this unwanted behavior, one enhance the time stepping heuristics with a rule that constraints, at every integration point, the relative growth of ϕ to be bounded by $\Delta\phi_{\max}$ in a single load increment. As described at line 27 of Algorithm 1, if this can not be achieved the load step is invalidated and restarted with a smaller time step decreased by the user defined factor β_t. In other words, an integration point can not switch from a pristine state to a fully damaged one during a single increment. Despite its apparent high cost, this methodology has proven to be very robust and tends to guaranty a good objectivity in terms of time stepping. In order to recover a quasi-static regime, when convergence is achieved the time step is increased by a user defined factor $\gamma_t > 1$ close to 1 and up to a maximum value $\Delta t_{\max}$ (line 31 of Algorithm 1).

In order to avoid oscillations, we use the numerical damping of the α-method by setting $\alpha = 0.3$ so that kinetic energy is dissipated and the quasi-static regime is recovered as soon as possible after the occurrence of instabilities. Ideally, this dissipated

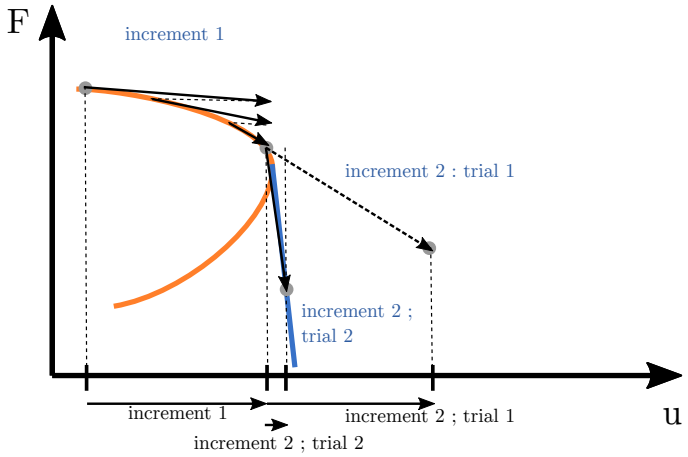


Figure 3 Schematic of the time-stepping adaptation of a Newton-Raphson or a staggered algorithm in the instability vicinity.

energy would correspond to the difference between the dynamic response and the quasi-static response that would have been obtained with a continuation algorithm (gray shaded area in Figure 2). With quasi-static solvers, such instabilities can also be circumvented by introducing some sort of viscosity that regularizes the solution but has, as side effect, to dissipate energy into a possibly less controlled way. We believe that the energy dissipated by the numerical damping if more consistent to the one that is physically dissipated during dynamics events than the viscous dissipation is.

2.5 | Parallel framework and remarks on the implementation

Staggered solution and code coupling

The alternate minimization is implemented in the implicit finite element software Z-set co-developed by Mines ParisTech and Onera[41]. Z-set has a Python binding that has been intensively used for multiphysics applications through a code coupling framework[42,43]. The alternate minimization solver has been implemented using this tenet. Thus, the equilibrium equation (9) is solved with one instance of Z-set while the phase field equation (10) is solved with an other one. These instances are running alternatively as depicted in Figure 1 and detailed in Algorithm 1.

A dedicated finite element has been designed to handle equation (10) according to expression (26). This approach is weakly intrusive since the rest of the finite element software remains unchanged. In the alternate minimization scheme, the phase field is viewed as a fixed known quantity at each displacement solution step while the displacement related ψ_0 field (the source term of equation (10)) is fixed at each phase field solution step. This process ends when a stagnation criterion of the phase field is met (line 13 in Algorithm 1). The displacement subproblem at fixed phase field can possibly be nonlinear, a Newton-Raphson solver is then used (line 11 in Algorithm 1). This allows to integrate into the mechanical model nonlinearities such as cohesive zone models (which is however not done in the following examples).

Algorithm 1 Staggered algorithm. Nodal vectors are noted as U_i^j and Φ_i^j where i stands for the load increment number and j is the number of the current staggered iteration.

$\Delta t \leftarrow \Delta t_{\text{init}}$
$t_0 \leftarrow 0$
$U_0^0, \dot{U}_0^0, \ddot{U}_0^0 \leftarrow U(0), \dot{U}(0), \ddot{U}(0)$
$\Phi_0^0 \leftarrow 0$
$i \leftarrow 1$ *(load increment counter)*
while $t_i < t_{\text{end}}$ **do** ▷ loading loop
  $\Phi_i^0 \leftarrow \Phi_{i-1}^{\text{last}}$
  $j \leftarrow 0$ *(staggered iteration counter)*
  continue $\leftarrow$ True
  while continue **do** ▷ staggered loop
    solve for U_i^j with fixed ϕ at $t = t_i$ (equations (9), (11), (12))
    solve Φ_i^j with fixed ψ_0 at $t = t_i$ (equations (10), (13))
    if $||\Phi_i^j - \Phi_i^{j-1}||_\infty < \epsilon$ **then**
      continue $\leftarrow$ False
      convergence $\leftarrow$ True
    else
      if $j > j_{\max}$ **then**
        continue $\leftarrow$ False
        convergence $\leftarrow$ False
      end if
      $j \leftarrow j + 1$
    end if
  end while
  if not convergence **then**
    $\Delta t \leftarrow \Delta t / \beta_t$
    $t_i \leftarrow t_{i-1} + \Delta t$
  else if $||\Phi_i^j - \Phi_i^0||_\infty > \Delta\phi_{\max}$ **then** ▷ if ϕ grows too fast
    $\Delta t \leftarrow \Delta t / \beta_t$
    $t_i \leftarrow t_{i-1} + \Delta t$
  else ▷ convergence and growth controlled : move on to next load increment
    $\Delta t \leftarrow \min(\Delta t \times \gamma_t, \Delta t_{\max})$
    $t_{i+1} \leftarrow t_i + \Delta t$
    $i \leftarrow i + 1$
  end if
end while

Staggered solution and parallel framework

Even though Z-set is used for both the displacement and the phase field subproblems, the parallel framework should be seen as a combination of a Multiple-Program-Multiple-Data model (MPMD) with a Single-Program-Multiple-Data model (SPMD). Indeed, to be able to exploit the domain decomposition methods presented in the next Section, two sets of Z-set instances are run. The first set is dedicated to the displacement subproblem, it handles in parallel the construction of finite element operators, tangent system resolution, input, output, etc. Similarly, the second set is dedicated to the damage subproblem. The communication between all processes of Z-set are handled with the MPI protocol. The communication scheme is illustrated in Figure 4.

Two MPI communicators are exploited, one for each subproblem. To simplify the communication pattern, the mesh is split with the same domain decomposition for both subproblems, $(\Omega_u^i)_i = (\Omega_\Phi^i)_i$. With this hypothesis, the only communications between the two MPI communicators are per subdomain point-to-point communications $\Omega_u^i \leftrightarrow \Omega_\Phi^i$ to exchange the distributed vectors ψ_0 and Φ. As it is, the hypothesis leads to a strong load unbalance between the displacement subproblem and the damage subproblem instances of Z-set. We choose to oversubscribe processes in order to attribute Ω_u^i and Ω_Φ^i to the same physical core to overcome this. Moreover, a perfect data locality is obtained for these point-to-point communications. Although useful, one can release this hypothesis handling non conform decomposition and interface, with the CWIPI library for instance[44]. The other type of communications are internal to their respective communicator and classically handled by domain decomposition solvers. In this work, we use the domain decomposition solvers natively implemented in Z-set[20,21].

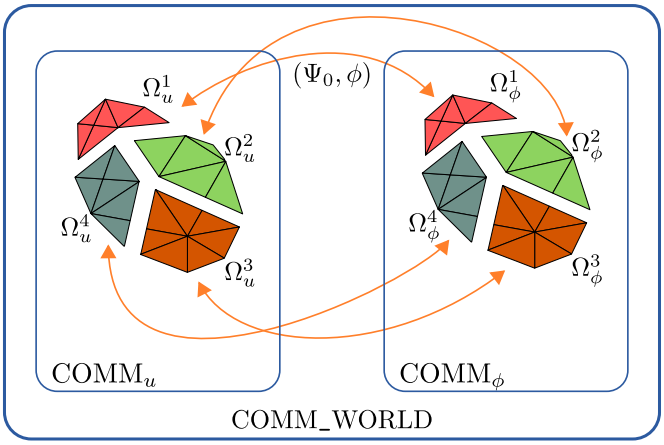


Figure 4 Schematic representation of the communication scheme.

3 | FETI AND BDD METHODS APPLIED TO THE PHASE FIELD FRACTURE PROBLEM

FETI[16] and BDD[15] share the same construction process. They are mainly the combination of three ingredients; the reformulation of the global problem into an equivalent set of substructured problems, the static condensation of these problems onto the interface and the use of a Krylov solver to solve the interface problem.

Although these methods are not limited to linear problems, we restrain to this framework for simplicity. If the problem considered is nonlinear, the method is classically applied to the tangent system induced by the Newton process. Another promising strategy is to swap the Newton and the Krylov loop in order to reduce the number of global communication[45,46]. This "swapped" strategy is not considered in the present work.

In the following, FETI and BDD are reminded using an abstract operator $\boldsymbol{\mathcal{A}}$. This abstract operator is assumed to be symmetric positive definite since both the phase field operator and the effective stiffness operator satisfy these properties. Some important points of these methods are highlighted, then the method is applied to the displacement subproblem and to the damage one. More details can be found in the review paper[47] and references herein.

3.1 | Substructured formulation

Let us consider the linear system of equation $\boldsymbol{\mathcal{A}x} = \boldsymbol{b}$ arising from the finite element discretization. The sparse matrix is given by (25) or (29) for the displacement subproblem, and by (26) for the damage one. Let $\boldsymbol{x}$ be the vector of unknowns and $\boldsymbol{b}$ the right-hand-side. Let $(\Omega^s)_{1\leqslant s\leqslant N_d}$ be a non overlapping partition of Ω such that: $\bar{\Omega} = \bigcup_{s=1}^{N_d} \bar{\Omega}^s$ and $\Omega^s \bigcap \Omega^p = \varnothing, \forall s \neq p$. The interface between two subdomains p and q is denoted by $\Upsilon^{sp} = \bar{\Omega}^s \bigcap \bar{\Omega}^p$ and the union of all interfaces of the subdomain s is denoted Υ^s. The set gathering all interfaces is denoted Υ. In the substructured formulation, only local quantities (e.g. restricted to one subdomain) are assembled such as the matrices $\boldsymbol{\mathcal{A}}^s$ and the right-hand-side $\boldsymbol{b}^s$. The global system is equivalent to the substructured formulation

$$\boldsymbol{\mathcal{A}}^s \boldsymbol{x}^s = \boldsymbol{b}^s + \lambda^s \quad \forall\, 1 \leqslant s \leqslant N_d \tag{32}$$

$$\sum_{s=1}^{N_d} \underline{\boldsymbol{A}}^s \boldsymbol{T}^s \boldsymbol{x}^s = \boldsymbol{0} \tag{33}$$

$$\sum_{s=1}^{N_d} \boldsymbol{A}^s \boldsymbol{T}^s \lambda^s = \boldsymbol{0} \tag{34}$$

where $\boldsymbol{T}^s : \Omega^s \to \Upsilon^s$ are trace operators, $\boldsymbol{A}^s$ and $\underline{\boldsymbol{A}}^s$ are primal and dual assembly operators respectively (see[47] for their definition). The Lagrange multiplier field λ^s enforces the continuity of the primal unknown across the subdomains interfaces. Equations (32) traduce the equilibrium of all subdomains, (33) corresponds to the continuity of the primal unknown across the interfaces (displacement, damage) and (34) expresses the equilibrium of the interface (action-reaction principle, flux equality).

All unknowns can be separated between internal unknowns (denoted with subscript i) and boundary ones (denoted with subscript b). Internal degrees of freedom can be eliminated in order to express (32)-(34) only in terms of boundary unknowns

$$\boldsymbol{S}^s \boldsymbol{x}_b^s = \boldsymbol{b}_p^s + \lambda_b^s \quad \forall\, 1 \leqslant s \leqslant N_d \tag{35}$$

$$\sum_{s=1}^{N_d} \underline{\boldsymbol{A}}^s \boldsymbol{x}_b^s = \boldsymbol{0} \tag{36}$$

$$\sum_{s=1}^{N_d} \boldsymbol{A}^s \lambda_b^s = \boldsymbol{0} \tag{37}$$

where $\boldsymbol{S}^s$ and $\boldsymbol{b}_p^s$ are primal Schur complements and condensed right-hand-sides.

$$\boldsymbol{S}^s = \boldsymbol{\mathcal{A}}_{bb}^s - \boldsymbol{\mathcal{A}}_{bi}^s \boldsymbol{\mathcal{A}}_{ii}^{s^{-1}} \boldsymbol{\mathcal{A}}_{ib}^s \tag{38}$$

$$\boldsymbol{b}_p^s = \boldsymbol{b}_b^s - \boldsymbol{\mathcal{A}}_{bi}^s \boldsymbol{\mathcal{A}}_{ii}^{s^{-1}} \boldsymbol{b}_i^s \tag{39}$$

Equations (35–37) are the starting point of both FETI and BDD methods.

Finally, we would like to point out that assembly operators are orthogonal in the following sense:

$$\sum_{s=1}^{N_d} \underline{\boldsymbol{A}}^s \boldsymbol{A}^{s\top} = \boldsymbol{0} \tag{40}$$

which means that any local interface vector $\boldsymbol{x}_b^s$ can be uniquely defined as a combination of a balanced vector and a continuous one $\boldsymbol{x}_b^s = \underline{\boldsymbol{A}}^{s\top} \boldsymbol{y} + \boldsymbol{A}^{s\top} \boldsymbol{z}$.

3.2 | A reminder of the Balancing Domain Decomposition method[15]

The Balancing domain decomposition (BDD) method writes the interface problem in terms of one unique primal global unknown $\boldsymbol{x}_b$. Local interface vectors are given by $\boldsymbol{x}_b^s = \boldsymbol{A}^{s\top} \boldsymbol{x}_b$ and (36) is satisfied by construction thanks to the orthogonality property of assembly operators (40). Few algebraic manipulations lead to the primal formulation:

$$\sum_{s=1}^{N_d} \boldsymbol{A}^s \boldsymbol{S}^s \boldsymbol{A}^{s\top} \boldsymbol{x}_b - \sum_{s=1}^{N_d} \boldsymbol{A}^s \boldsymbol{b}_p^s = \sum_{s=1}^{N_d} \boldsymbol{A}^s \boldsymbol{\lambda}_b^s = \boldsymbol{0} \tag{41}$$

$$\boldsymbol{S} \boldsymbol{x}_b - \boldsymbol{b}_p = \boldsymbol{0} \tag{42}$$

The global primal Schur complement $\boldsymbol{S} = \sum_{s=1}^{N_d} \boldsymbol{A}^s \boldsymbol{S}^s \boldsymbol{A}^{s\top}$ is never built explicitly. Since this system is solved using a Krylov iterative solver, only the result of a multiplication by $\boldsymbol{S}$ is needed. This computation is well suited to parallel computers since $\boldsymbol{S}$ is a sum of local contributions. Also, in the present implementation, no Schur complement are explicitly computed, the action of these Schur operators are evaluated implicitly.

Preconditioner

The BDD preconditioner $\boldsymbol{M}_{BDD}^{-1}$ mimics the additive structure of $\boldsymbol{S}$, it is chosen as a scaled sum of generalized inverse of primal Schur complements defined by

$$\boldsymbol{M}_{BDD}^{-1} = \sum_{s=1}^{N_d} \tilde{\boldsymbol{A}}^s \boldsymbol{S}^{s\dagger} \tilde{\boldsymbol{A}}^{s\top} \tag{43}$$

where $\tilde{\boldsymbol{A}}^s$ are scaled primal assembly operators and the superscript $\boldsymbol{S}^{s\dagger}$ denotes for a generalized inverse of $\boldsymbol{S}^s$. The generalized inverse is needed since depending on the original PDE and on the natural boundary conditions of the problem, $\boldsymbol{S}^s$ may be singular. Corresponding subdomains are commonly qualified as "floating subdomains"[48]. The action of $\boldsymbol{S}^{s\dagger}$ is obtained by solving a

local problem with Neumann boundary conditions. Regarding scaling operators, classical choices are multiplicity scaling and stiffness scaling (often called **k**-scaling) [22].

Coarse problem

The BDD preconditioner is applied to the residual of the Krylov solver $\boldsymbol{z} = \boldsymbol{M}_{BDD}^{-1}\boldsymbol{r}$. For floating subdomains, local right-hand-sides must lie inside the image of $\boldsymbol{S}^s$ which leads to the optimality conditions:

$$\boldsymbol{R}_{\boldsymbol{b}}^{s\top}\tilde{\boldsymbol{A}}^{s\top}\boldsymbol{r} = \boldsymbol{0} \quad \forall s \tag{44}$$

where $\boldsymbol{R}_{\boldsymbol{b}}^s$ is the nullspace of $\boldsymbol{S}^s$. We rewrite this condition as $\boldsymbol{C}^\top\boldsymbol{r} = \boldsymbol{0}$ with

$$\boldsymbol{C} = \left(\tilde{\boldsymbol{A}}^1\boldsymbol{R}_{\boldsymbol{b}}^1 \mid \ldots \mid \tilde{\boldsymbol{A}}^{N_d}\boldsymbol{R}_{\boldsymbol{b}}^{N_d}\right) \tag{45}$$

These optimality conditions provide an additional coarse problem which is enforced using an augmented Krylov solver. An augmentation projector $\boldsymbol{\Pi}_C$ such that $\boldsymbol{C}^\top\boldsymbol{S}\boldsymbol{\Pi}_C = \boldsymbol{0}$ is defined, and the solution is sought as:

$$\boldsymbol{x}_b = \boldsymbol{x}_0 + \boldsymbol{\Pi}_C\tilde{\boldsymbol{x}} \tag{46}$$

$$\boldsymbol{x}_0 = \boldsymbol{C}(\boldsymbol{C}^\top\boldsymbol{S}\boldsymbol{C})^{-1}\boldsymbol{C}^\top\boldsymbol{b}_p \tag{47}$$

$$\boldsymbol{\Pi}_C = \boldsymbol{I} - \boldsymbol{C}(\boldsymbol{C}^\top\boldsymbol{S}\boldsymbol{C})^{-1}\boldsymbol{C}^\top\boldsymbol{S} \tag{48}$$

The system solved by the Krylov solver is finally

$$\boldsymbol{S}\boldsymbol{\Pi}_C\tilde{\boldsymbol{x}} = \left(\boldsymbol{b}_p - \boldsymbol{S}\boldsymbol{x}_0\right) \tag{49}$$

If the initial system is symmetric positive definite, a Conjugate gradient is classically used.

3.3 | A reminder of FETI method [16]

The FETI method writes the interface problem in terms of one unique dual global unknown λ_b. The Lagrange multiplier field λ_b enforces the continuity of the primal unknown across the subdomains interfaces. Local interface vectors are given by $\lambda_b^s = \underline{\boldsymbol{A}}^{s\top}\lambda_b$ and (37) is satisfied by construction. The remaining equations are:

$$\boldsymbol{S}^s\boldsymbol{x}_b^s = \boldsymbol{b}_p^s + \underline{\boldsymbol{A}}^{s\top}\lambda_b \quad \forall\, 1 \leqslant s \leqslant N_d \tag{50}$$

$$\sum_{s=1}^{N_d}\underline{\boldsymbol{A}}^s\boldsymbol{x}_b^s = \boldsymbol{0} \tag{51}$$

In order to eliminate primal unknowns, local systems have to be solved. As for BDD, the use of generalized inverses are needed for floating subdomains. Compatibility conditions arise and the solution $\boldsymbol{x}_b^s$ is not unique defined. The magnitude of the nullspace

components are gathered in local vectors $\boldsymbol{\beta}^s$.

$$\boldsymbol{x}_b^s = \boldsymbol{S}^{s\dagger}\left(\boldsymbol{b}_p^s + \underline{\boldsymbol{A}}^{s\top}\lambda_b\right) + \boldsymbol{R}_b^s\boldsymbol{\beta}^s \qquad \forall\, 1 \leqslant s \leqslant N_d \tag{52}$$

$$\boldsymbol{0} = \boldsymbol{R}_b^{s\top}\left(\boldsymbol{b}_p^s + \underline{\boldsymbol{A}}^{s\top}\lambda_b\right) \qquad \forall\, 1 \leqslant s \leqslant N_d \tag{53}$$

After some algebraic manipulations, the classical FETI system is formed:

$$\begin{pmatrix} \boldsymbol{F} & \boldsymbol{G} \\ \boldsymbol{G}^\top & \boldsymbol{0} \end{pmatrix}\begin{pmatrix} \lambda_b \\ \boldsymbol{\beta} \end{pmatrix} = \begin{pmatrix} \boldsymbol{d} \\ \boldsymbol{e} \end{pmatrix} \tag{54}$$

where

$$\boldsymbol{e} = -\left(\boldsymbol{b}_p^{1\top}\boldsymbol{R}_{\boldsymbol{b}}^1 | \ldots | \boldsymbol{b}_p^{N_d\top}\boldsymbol{R}_{\boldsymbol{b}}^{N_d}\right)^\top \quad ; \quad \boldsymbol{d} = -\sum_{s=1}^{N_d}\underline{\boldsymbol{A}}^s\boldsymbol{T}^s\boldsymbol{\mathcal{A}}^{s\dagger}\boldsymbol{b}_p^s$$

$$\boldsymbol{F} = \sum_{s=1}^{N_d}\underline{\boldsymbol{A}}^s\boldsymbol{S}^{s\dagger}\underline{\boldsymbol{A}}^{s\top} \quad ; \quad \boldsymbol{G} = \left(\underline{\boldsymbol{A}}^1\boldsymbol{R}_{\boldsymbol{b}}^1 | \ldots | \underline{\boldsymbol{A}}^{N_d}\boldsymbol{R}_{\boldsymbol{b}}^{N_d}\right)$$

As before, the presence of nontrivial local kernels depends on the underlying partial differential equation and on the prescribed Dirichlet boundary conditions. The second row, $\boldsymbol{G}^\top\lambda = \boldsymbol{e}$, is associated with the constraint that the right-hand-side, once restricted to subdomain interfaces, should lie in the image of local operators.

Coarse problem

Instead of solving a saddle point system, an initialization–projection strategy is applied and λ_b is sought as:

$$\lambda_b = \lambda_0 + \boldsymbol{\Pi}\tilde{\lambda} \quad ; \quad \boldsymbol{G}^\top\lambda_0 = \boldsymbol{e} \quad ; \quad \boldsymbol{G}^\top\boldsymbol{\Pi} = \boldsymbol{0}$$

The coarse problem projector $\boldsymbol{\Pi}$ is classically of the form (55) where operator $\boldsymbol{Q}$ is a symmetric positive definite matrix.

$$\boldsymbol{\Pi} = \boldsymbol{I} - \boldsymbol{Q}\boldsymbol{G}(\boldsymbol{G}^\top\boldsymbol{Q}\boldsymbol{G})^{-1}\boldsymbol{G}^\top \tag{55}$$

Substituting this form into (54), and pre-multiplying by $\boldsymbol{\Pi}^\top$ leads to the final linear system:

$$\boldsymbol{\Pi}^\top\boldsymbol{F}\boldsymbol{\Pi}\tilde{\lambda} = \boldsymbol{\Pi}^\top(\boldsymbol{d} - \boldsymbol{F}\lambda_0) \tag{56}$$

This system is solved with a Krylov iterative solver. As for BDD, a Conjugate gradient is classically used.

Preconditioner

The FETI operator being a scaled sum of dual Schur complement, the usual preconditioner is chosen as a scaled sum of (approximation of) primal Schur complements:

$$\boldsymbol{M}_{FETI}^{-1} = \sum_{s=1}^{N_d}\tilde{\underline{\boldsymbol{A}}}^s\tilde{\boldsymbol{S}}^s\tilde{\underline{\boldsymbol{A}}}^{s\top} \tag{57}$$

where $(\underline{\tilde{\boldsymbol{A}}}^s)$ are scaled assembly operators such that:

$$\left(\sum_{s=1}^{N_d} \underline{\boldsymbol{A}}^s \underline{\tilde{\boldsymbol{A}}}^{s\top}\right) \underline{\boldsymbol{A}}^j = \underline{\boldsymbol{A}}^j, \quad \forall\, 1 \leqslant j \leqslant N_d$$

One of the strength of the FETI method is the diversity of available preconditioners. Indeed, several approximations of $\tilde{\boldsymbol{S}}^s$ may be used. Common choices are:

- $\tilde{\boldsymbol{S}}^s = \boldsymbol{S}^s$ leading to the Dirichlet (or full) preconditioner which is costly but optimal;
- $\tilde{\boldsymbol{S}}^s = \boldsymbol{\mathcal{A}}^s_{bb}$ (lumped preconditioner);
- $\tilde{\boldsymbol{S}}^s = \mathrm{diag}(\boldsymbol{\mathcal{A}}^s_{bb})$ (super lumped preconditioner).

The Dirichlet preconditioner provides the better convergence rate, but the lumped one may surpass it in terms of time to solution. This choice is often guided by numerical experiments (see Section 5). Regarding scaling operators, classical choices are the same as for BDD: multiplicity scaling and stiffness scaling[22].

3.4 | Local kernels and generalized inverse

The FETI method requires a certain care in the computation of generalized inverse, especially in the case of ill conditioned systems where the difference between null and small pivots becomes unclear. A misdetection of these kernels leads to a divergence of the Krylov solver and the FETI system (56) is not equivalent anymore to the initial one. Also, ill conditioned generalized inverses may slow down its convergence. A misdetection is less critical for BDD since it comes into play at the preconditioner level. Several techniques aiming to provide a robust way to compute them have been proposed based on explicit construction, partial factorization or condensation[49,50,51]. The graph centrality approach proposed in[23] is used in this work.

Also, the presence of local kernels remains an essential ingredient to the parallel scalability of the BDD and FETI method. Indeed, it provides a coarse problem that quickly propagates the "loading information" across all subdomains. If a floating subdomain does not induce a singular local operator, these methods fall back to the Neumann-Neumann and Dirichlet-Dirichlet method which usually scale with difficulty. An auxiliary coarse problem may be added using an augmentation strategy (see Section 4).

3.5 | Application to the displacement subproblem

Quasi-static simulations

For stable cracks propagations, the simulations can be performed assuming quasi-static regime. Here, local operators $\boldsymbol{\mathcal{A}}^s$ are simply local stiffness matrices given by (25). To simplify the notations, the *uu* exponent is removed, thus the stiffness operator

is denoted $\boldsymbol{K}^s$ in this section. These local operators have a nontrivial kernel in the case of floating subdomains which provides natural coarse problem for both FETI and BDD.

Implicit dynamics simulations

As detailed in Section 2.4, unstable crack propagation are handled using implicit dynamics with automatic time step adaptation. The automatic step adaptation process fulfills two main objectives. When entering into a dynamic regime, the time step is automatically reduced to regularize the solution and to accurately describe the unstable phase of the crack propagation. When recovering a quasi-static regime, the time step is increased in order to reduce the computational cost of the overall simulation. The time step reduction may be "brutal" because of numerous rejected steps at the static–dynamic transition as it will be illustrated in the Section 6. The time step increase is however limited by a growth factor γ_t. Whatever the time integrator used (Newmark, HHT, generalized-α) and its implementation (displacement, velocity and acceleration form), the variation of the time step makes the detection of floating subdomains complex. Local operators are of the form:

$$\boldsymbol{\mathcal{A}}^s = \left(\frac{1}{\beta \Delta t^2} \boldsymbol{M}^s + (1-\alpha)\boldsymbol{K}^s \right) \tag{58}$$

In the dynamic regime Δt is small so that the local operator is dominated by the mass matrix $\boldsymbol{M}^s$ whereas $\boldsymbol{\mathcal{A}}^s$ tends to $(1-\alpha)\boldsymbol{K}^s$ in the quasi-static regime where Δt is relatively high. Thanks to the regularization provided by the mass matrix, $\boldsymbol{\mathcal{A}}^s$ is mathematically invertible even in the absence of Dirichlet boundary conditions. This consideration is however only valid in exact arithmetic. In the quasi-static regime, the mass matrix becomes negligible and $\boldsymbol{\mathcal{A}}^s$ becomes singular if no Dirichlet boundary condition are present.

All automatic kernel detection strategies cited in Section 3.4 rely on a user defined threshold to make the distinction between small and null singular values (or pivots according to the method used). At the dynamic–static transition, the time step slowly increases so that the smallest singular values progressively tends to zero. The automatic kernel detection may fail at these intermediate time steps despite our recent progress in this domain[23]. Thus, only the BDD method is used for dynamics since it is less affected by a misdetection of the local kernels.

Please note that the FETI-DP method[17] only partially resolves this point. The coarse problem is no more based on the nullspace operators, but the propagation of the cracks may split the subdomain into several pieces leading extra rigid body motions. The crack path being unknown a priori, the FETI-DP corner nodes need to be adapted on the fly. Also, new pieces completely inside a subdomain lead to internal mechanisms.

3.6 | Application to the damage subproblem

The properties of the finite element operator associated with the damage subproblem (30) has some consequences on the domain decomposition methods. First, the additive structure of $\boldsymbol{K}^{\phi\phi}$, which looks like the sum of a "mass" matrix and a discrete Laplace operator, leads to a reasonably well conditioned matrix. Thus, we expect a good convergence rate of the interface Krylov solvers. The FETI method has the advantage here, since cheap preconditioners are available.

Also, thanks to the regularization of the "mass" matrix, the damage local operator is always invertible even in the absence of Dirichlet boundary condition. Floating subdomains do not induce singular local operators which removes the need for the coarse grid mechanism of the FETI and BDD methods. As in the case of implicit dynamics, a degraded scalability may be expected without this coarse problem. However, contrarily to implicit dynamics, the right-hand-side of the AT2 damage subproblem is dense which compensates the lack of coarse problem. Auxiliary coarse spaces could be used, but the numerical experiments have shown good convergence without this extra feature. In the following, only the FETI method is considered for the damage subproblem in order to exploit the available preconditioners.

4 | ACCELERATION STRATEGIES OF THE KRYLOV SOLVER

The staggered minimization scheme requires numerous tangent system resolutions. To improve the overall computational efficiency, the first strategy is to reduce the number of subproblem to be solved by optimizing the minimization algorithm[7]. The second one is to accelerate the resolution of the tangent systems. Both strategies are compatible, but we only consider acceleration techniques for the resolution of the linear systems in the following. In the domain decomposition framework, the two classical approaches are to add an auxiliary coarse problem and/or to improve the preconditioner in order to speed up the convergence of the iterative solver.

4.1 | Additional coarse problem

This approach consists in augmenting the resolution of the Krylov solver by an additional constraint of the form $\boldsymbol{C}_a^\top \boldsymbol{r} = 0$ where the matrix $\boldsymbol{C}_a$ is a basis of a chosen subspace and $\boldsymbol{r}$ is the residual vector of the iterative solver. Thus, the solution solves the problem exactly in the range of $\boldsymbol{C}_a$, at every iteration. If $\boldsymbol{C}_a$ is of small dimension and well-chosen, the convergence can be greatly accelerated.

Additional coarse problem for the FETI method

The FETI variant of this approach, so-called FETI-2, was initially introduced to efficiently solve plate and shell problems by imposing the displacement continuity at corner nodes through a second level of initialization and projection[52,53]. With the

FETI-2 method, the solution is sought as (59) and only $\lambda^\star$ remains unknown.

$$\lambda_b = \lambda_0 + \mathbf{\Pi}(\tilde{\lambda}_0 + \mathbf{\Pi}_{C_a}\lambda^\star) \tag{59}$$

$$\tilde{\lambda}_0 = \boldsymbol{C}_a(\boldsymbol{C}_a^\top \boldsymbol{F}\boldsymbol{C}_a)^{-1}\boldsymbol{C}_a^\top(\boldsymbol{d} - \boldsymbol{F}\lambda_0) \tag{60}$$

$$\mathbf{\Pi}_{C_a} = \boldsymbol{I} - \boldsymbol{C}_a(\boldsymbol{C}_a^\top \boldsymbol{F}\boldsymbol{C}_a)^{-1}\boldsymbol{C}_a^\top \boldsymbol{F} \tag{61}$$

The system to be solved by the Krylov solver is:

$$\mathbf{\Pi}^\top \boldsymbol{F}\mathbf{\Pi}\mathbf{\Pi}_{C_a}\lambda^\star = \mathbf{\Pi}^\top(\boldsymbol{d} - \boldsymbol{F}(\lambda_0 + \mathbf{\Pi}\tilde{\lambda}_0)) \tag{62}$$

Other choices of $\boldsymbol{C}_a$ have been proposed in order to overcome the lack of a natural coarse problem (in the case of dynamic studies for instance[54]); to reuse the search space in the context of repeated or similar studies [55,56] or to "catch some arduous components" of the solution using local generalized eigenvalue problem[57].

In this work, we have investigated the Selective Reuse of Krylov Subspace algorithm (SRKS)[56] in order to exploit the similarity of successive linear resolutions. The SRKS algorithm is based on the analysis of the Ritz values which are the eigenvalues of the Hessenberg matrix generated by the Krylov iterative solver. When converged, these Ritz values accurately approximate the eigenvalues of the initial matrix. The associated Ritz vectors are then good approximations of the corresponding eigenvectors. The SRKS algorithm puts the converged Ritz vectors of a current linear solve in the additional coarse space $\boldsymbol{C}_a$ of the subsequent ones. The identification of converged Ritz values relies on a stagnation criterion which depends on a user defined threshold ϵ_{srks}.

In order to limit the memory requirement of the process, the maximal size of $\boldsymbol{C}_a$ is prescribed by the user. When $\boldsymbol{C}_a$ reaches the maximal allowed size, its update follows a first-in first-out mechanism: the oldest recycled Ritz vectors are popped out and new ones are added.

Additional coarse problem for the BDD method

The BDD coarse problem already takes the form of an augmented resolution. If it is possible to enlarge this coarse problem using the $\boldsymbol{C}_a$ described previously, it turns out to be less practical because the concatenated matrix $(\boldsymbol{C}\ \boldsymbol{C}_a)$ has to be full column rank. A singular value decomposition of $(\boldsymbol{C}\ \boldsymbol{C}_a)$ is needed to get a robust process. The overhead associated with the singular value decomposition limits the efficiency of the approach. Additional coarse problem for the BDD method is not considered further in this work.

4.2 | Multipreconditioning

Both FETI and BDD suffer from a loss of performance when strong material heterogeneities are misplaced with respect to the interface. The damage field weaken the strain energy what induces such a heterogeneity. Thus, the condition number of the

tangent systems degrades with the crack propagation. The absence of the natural coarse space in the case of implicit dynamics simulation also hinders the convergence of the solvers. The multipreconditioned variants AMPFETI and AMPBDD are efficient alternatives here [19,58,20,21] since they are robust with respect to material heterogeneities and provide a "non local" mechanism that compensates the lack of natural coarse problem. In this work, a new implementation of AMPBDD has been tested to solve the unstable propagation test cases in Section 5.3.1. The AMPBDD implementation uses the concept of multipreconditioning aggregates first presented for AMPFETI [21] and relies on the global adaptive test [58].

5 | NUMERICAL RESULTS

The parallel performances of the proposed parallel framework are presented in this section. First, some details about the implementation are provided. Then, a three dimensional plate with multiple cracks is used in Section 5.2 to analyze the scalability of the method. The test case is derived in two variants in order to analyze stable and unstable cracks propagation.

5.1 | Remarks on the implementation and hardware computing resources

All the computations presented in this section have been performed using the Part 1 of the Sator supercomputer at ONERA. Part 1 of Sator is made of Intel Xeon Broadwell (E5-2680v4@2.4GHz) with 2×14 cores per node. All compute nodes are interconnected with an Intel Omnipath 100Gbps network. In all configurations, MUMPS solver [59] (version 5.1.2) is used in association with the BLAS library provided by Intel 19.0.5 MKL for local solves. The Pardiso direct solver is used to solve the coarse problem. Eigen library [60] is used for dense linear algebra. Communications are handled by the Intel MPI library (version 19.0.5).

5.2 | Weak scalability study

To study the weak scalability one need to design a test case that presents a reproducible solution pattern and exhibits a significant crack propagation. We thus propose to generate problems of increasing sizes on the basis of the "four cracks pattern" depicted in Figure 5. The figure also represents the underlying finite element mesh. The initial crack is explicitly discretized, and the elements are refined in the crack propagation area. Two $N \times N$ patterns are illustrated in Figure 6. A homogeneous vertical displacement field is applied on the upper and lower faces.

In this condition, the propagation of the cracks would be rapidly unstable and the solution would become inhomogeneous thus preventing the study of the weak scalability in a quasi-static regime. To circumvent this difficulty, we add a non-damageable substrate (as depicted in Figure 5) which acts as an elastic stabilizer by storing the energy that would be turned into kinetic energy. The damageable cracked plate and the substrate meshes are both extruded from the same surface mesh with one element

in the thickness. These two parts are mechanically linked at their interface by using a continuous finite element discretization. However, to keep the phase field localized in the damageable plate, the mesh used to solve the damage subproblem is split at the interface as depicted in Figure 5. The physical meaning of the resulting model is debatable, but it offers a proper way to expand the size of the problem while guaranteeing a stable propagation of an almost repeatable crack propagation.

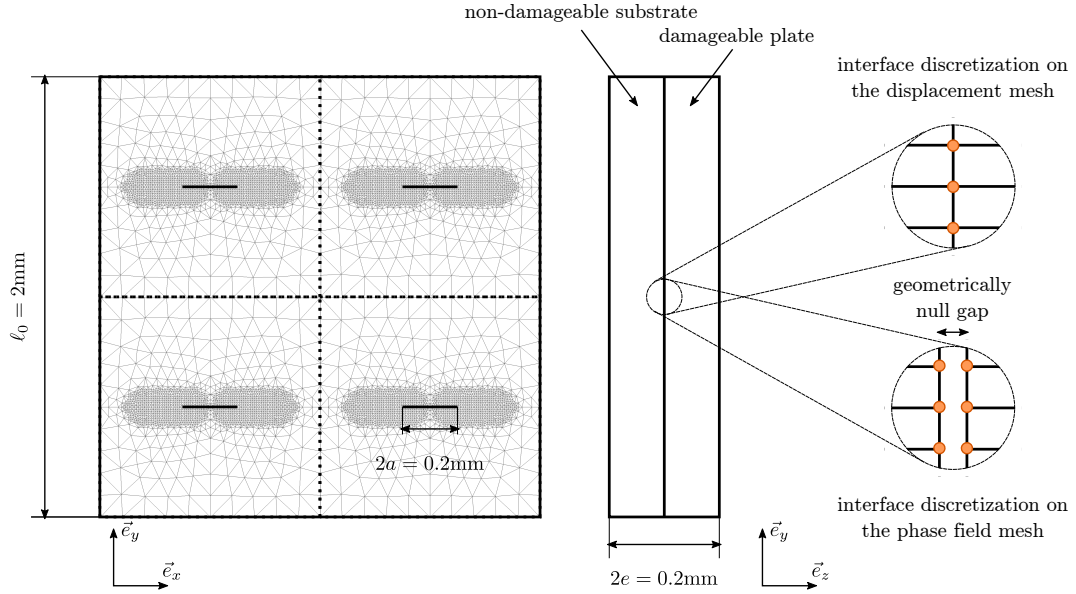


Figure 5 Base pattern of the weak scalability study with a representation of the pre-cracks. The split of nodes at the interface in the phase field mesh is schematized with a gap which is null in practice.

The elastic parameters of both the cracked plate and the substrate are $E = 210\,\text{GPa}$ and $\nu = 0.3$. The AT2 model is selected on the cracked plate with $l = 0.04\,\text{mm}$ and $G_c = 0.1\text{J.mm}^{-2}$

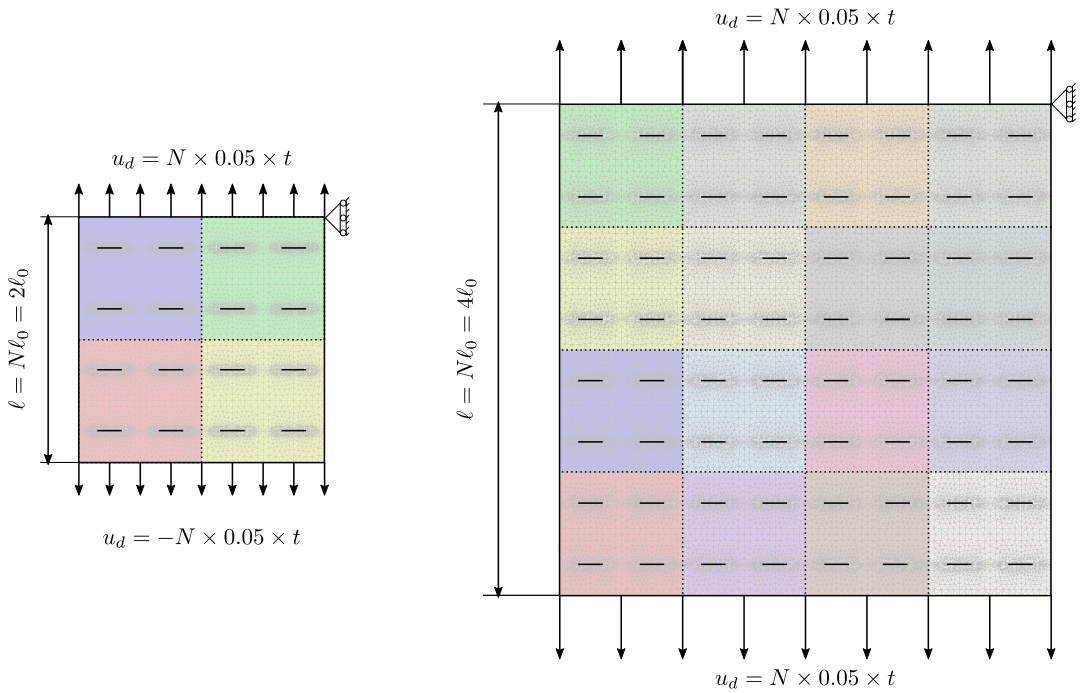


Figure 6 Illustration of $N \times N$ patterns problems and their corresponding domain decompositions (regular decomposition) with $N = 2$ and $N = 4$. Each color represents a subdomain. Vertical displacements are imposed on the top and bottom sides and horizontal displacement is blocked in the upper-right corner.

5.2.1 | Influence of the preconditioner for the damage subproblem

As recalled in Section 3.3, the FETI method has the advantage of providing several combinations for the preconditioner. Each of these combinations is defined by the choice of a scaling operator and an approximation of the Schur complement. If the Dirichlet preconditioner provides the best convergence rate, it may be outperformed in terms of time to solution. This assessment depends on the problem to be solved and on the software implementation.

The FETI method is commonly applied to mechanical problems. For instance, scalability results and robustness assessment of several combinations are provided in [20] for the FETI and AMPFETI methods. The cases studied in [20] were highly heterogeneous elasticity problems.

The application of the FETI method to the phase field subproblem is new and needs some numerical tests. In this Section, several preconditioners are compared for the 10×10 replicated pattern test case. Table 1 defines the three different combinations of scaling and operator used. Only the results of the phase field subproblem are shown, the overall performance of the method will be detailed in the next section. The convergence is triggered when $\|r_i\|/\|r_0\| \leq 10^{-6}$. At the end of the simulation, cracks have largely propagated in the plate (similarly to Figure 14). The results are grouped in Table 2 where all tangent system solves are taken into account. As expected, the Dirichlet preconditioner provides the best convergence rate. However, the Lumped variant is the fastest in terms of computational time. Despite its simplicity, the diagonal preconditioner of Super lumped variant

Preconditioner			
Variant	Dirichlet	Lumped	Super lumped
Operator	Dirichlet	Lumped	Super lumped
Scaling	Stiffness	Stiffness	Stiffness

Table 1 Considered preconditioner combinations for the damage problem.

is also faster than the Dirichlet one. The convergence curves of all tangent systems are given in Figure 7. Whatever the variant

Variant	Dirichlet	Lumped	Super lumped
Total number of iter.	367	800	1442
Average number of iter.	7.06	15.38	27.73
Number of tangent systems	52	52	52
ϕ-solver (s)	28.71	17.48	24.91

Table 2 Three dimensional plate with multiple cracks undergoing stable propagation: influence of the preconditioner for the damage subproblem for the case 10×10 with $N_d = 100$.

is, we observe that the convergence of the Krylov solver is almost the same for all tangent systems. So, the evolution of the vector Ψ_0 has little impact on the condition number of the preconditioned system. The same tendencies are observed for different number of replicated patterns. From these conclusions, the Lumped variant will be used from now on.

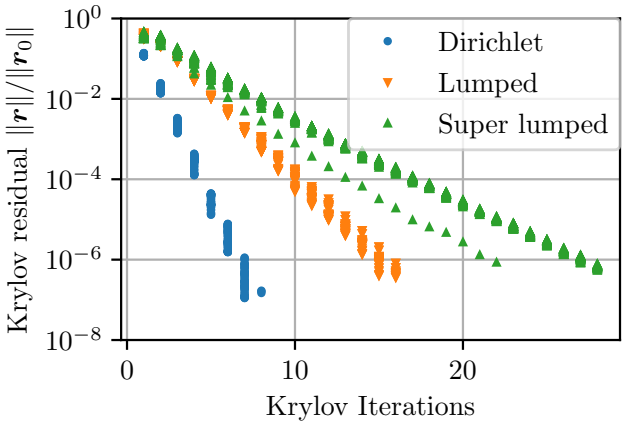


Figure 7 Three dimensional plate with multiple cracks undergoing stable propagation: residual curves of the Krylov solvers for the damage subproblem for the case 10×10 with $N_d = 100$. Convergence curves of all tangent systems are superimposed.

5.2.2 | Three dimensional plate with multiple cracks undergoing stable propagation

This section is dedicated to the study of the weak scalability of the proposed parallel framework. The test case is the plate presented previously with the elastic stabilizer. The study starts from 4 subdomains and goes up to 324 subdomains which corresponds to 17.8 millions unknowns for the mechanical problem and 5.9 millions unknowns for the damage one. The studied configurations are gathered in Table 3. Two domain decompositions are considered. As shown in Figure 6, the first one is

Test case	N_d	#DOFs-$\boldsymbol{u}$	#DOFs-$\boldsymbol{\phi}$	#cores
2×2	4	221481	73827	4
4×4	16	884457	294819	16
6×6	36	1988937	662979	36
8×8	64	3534921	1178307	64
10×10	100	5522409	1840803	100
12×12	144	7951401	2650467	144
14×14	196	10821897	3607299	196
16×16	256	14133897	4711299	256
18×18	324	17887401	5962467	324

Table 3 Three dimensional plate with multiple cracks undergoing stable propagation: studied configurations.

regular which leads to perfect load balancing and straight interfaces. This decomposition is well adapted to weak scalability studies since all local problems are identical and the condition number of the interface problem grows slowly with the number of subdomains. However, this regular decomposition has little practical interest when looking at engineering applications. So, another decomposition is studied (see Figure 8). This decomposition is automatically generated using the graph partitioning software METIS. This decomposition is well-balanced, but the interface between subdomains is more and more jagged which is known to penalize the convergence of domain decomposition methods. Also, the placement of the interface is not controlled which means that it can cross or go along the cracks. A perfect scalability is not expected with this kind of decomposition.

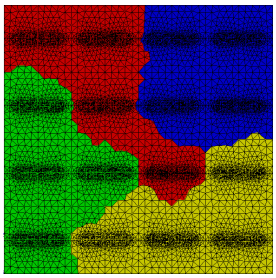


Figure 8 Illustration of the mesh and decomposition used for weak scalability (two by two pattern, automatic domain decomposition with $N_d = 4$).

For the phase field subproblem, the FETI solver uses the Lumped variant defined in Table 1. The preconditioner of the FETI solver used for the displacement subproblem is the combination of Primal Schur complement and stiffness scaling. The projector is orthogonal ($\boldsymbol{Q}$ is set to the identity matrix in (55)). For both solvers, the convergence is triggered when $\|r_i\|/\|r_0\| \leq \varepsilon$. The convergence criterion ε is set to 10^{-6}. Unless otherwise specified, the Krylov solver is a conjugate gradient. Regarding the staggered scheme, the stagnation threshold is $\epsilon = 0.05$, and the maximum number of fixed point iterations before reducing the time step is 5.

Two timers have been defined in order to analyze the parallel performances:

- The "Time to solution" timer corresponds to the complete duration of the simulation, it includes inputs/outputs, finite element operator construction, local operator factorizations, resolutions, staggered algorithm communications, etc.
- The "Solver time" timer regroups, for the corresponding physics, the time spent in the Krylov iterative solver. It includes search directions orthogonalization, applications of FETI (or BDD) operator and preconditioner, projection, etc.

Few counters are also provided: the cumulated number of Krylov iterations of all solves (Total number of iter.), the total number of tangent systems (Number of tangent systems) and an average number of Krylov iterations per solve (Average number of iter.).

Scalability results with the regular decomposition

The results are gathered in Table 4. In terms of Krylov iterations, the weak scalability of both solvers is excellent since a plateau is observed as soon as $N_d \geq 36$. Despite the absence of a coarse problem for the damage solver, its number of iterations does not increase with the number of subdomains. The time spent in the displacement FETI subproblem slightly increases with the number of subdomains. This tendency naturally affects the overall time to solution. Due to its better condition number and lower number of unknowns, the computational cost of the damage subproblem is negligible compared to mechanical one.

The convergence curves of all tangent systems of the mechanical solver are shown in Figure 7. All superimposed curves are well clustered which means that convergence rate of the mechanical solver is stable with respect to the crack propagation.

The evolution of the time step is shown in Figure 10. Whatever the size of the problem, the time step evolution is the same meaning that we have provided a reproducible pattern.

Scalability results with the automatic decomposition

The results obtained with the automatic decomposition are gathered in Table 5. Here the jagged interface penalizes the weak scalability of both solvers. A significant increase of the number of Krylov iterations is observed. The convergence curves of all tangent systems are shown in Figure 11 (only the case $N_d \in \{100, 144, 196, 256, 324\}$ are shown for readability). For the damage solver, all curves are well clustered which means that convergence rate of the Krylov solver is stable with respect to the crack propagation. For the mechanical one, the curves gradually shift to the right indicating that the convergence of the Krylov

N_d	4	16	36	64	100	144	196	256	324
Overall performances									
Time to solution (s)	702.5	784.8	854.3	892.8	905.9	926.1	955.9	975.8	979.4
Displacement subproblem									
Total number of iter.	560	952	1064	1064	1064	1066	1064	1064	1064
Average number of iter.	10.00	17.00	19.00	19.00	19.00	19.04	19.00	19.00	19.00
Number of tangent systems	56	56	56	56	56	56	56	56	56
Solver time (s)	236.6	298.5	355.7	385.4	382.0	388.1	396.1	403.4	390.3
Damage subproblem									
Total number of iter.	200	208	208	208	208	208	208	208	208
Average number of iter.	3.85	4.00	4.00	4.00	4.00	4.00	4.00	4.00	4.00
Number of tangent systems	52	52	52	52	52	52	52	52	52
Solver time (s)	7.319	8.43	9.178	9.193	9.264	9.59	10.56	9.796	9.867

Table 4 Three dimensional plate with multiple cracks undergoing stable propagation: weak scalability study with regular decomposition. FETI with a conjugate gradient is used for both physics.

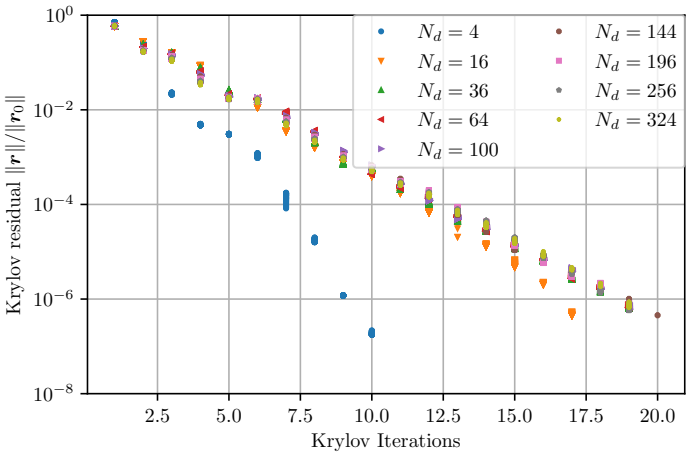
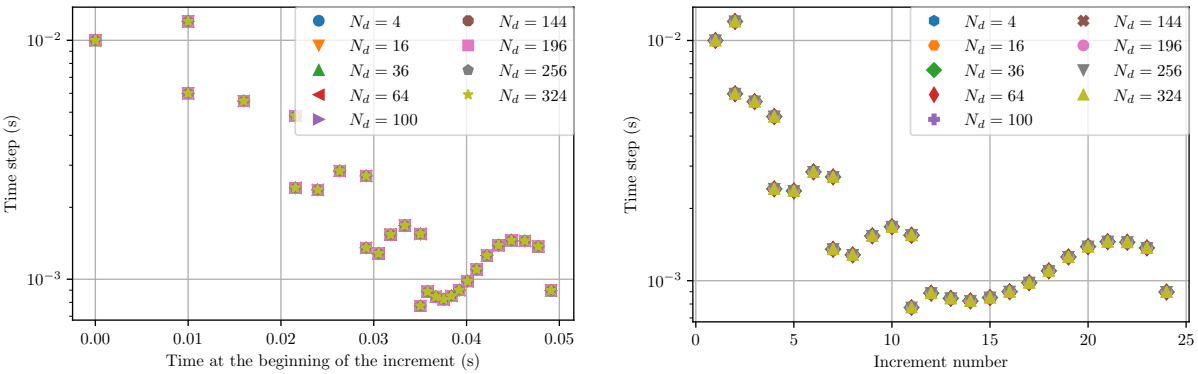


Figure 9 Three dimensional plate with multiple cracks undergoing stable propagation: residual curves of the conjugate gradient solvers for the displacement subproblem with regular decomposition. Convergence curves of all tangent systems are superimposed.

solver is slightly slowed down by the crack propagation. This trend is visible regardless of the number of subdomains. Again, the computational cost of the damage subproblem is negligible compared with to mechanical one.

Improving the convergence with Krylov subspace recycling

As shown in the previous section, the scalability with automatic decomposition may be improved, especially for the displacement subproblem. To this end, the Selective recycling of Krylov subspaces (SRKS) introduced in Section 4 has been tested. The stagnation threshold for the Ritz values is set to 10^{-6} and the maximal size of the augmentation subspace $\boldsymbol{C}_a$ is 100. As explained previously, a FIFO mechanism is used when the maximal size is reached. This situation only happened with $N_d \geq 196$. The



(a) Time step evolution w.r.t time.

(b) Time step evolution w.r.t increment number.

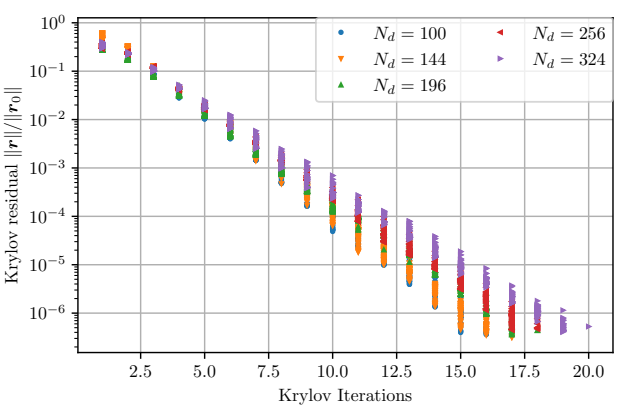
Figure 10 Three dimensional plate with multiple cracks undergoing stable propagation: evolution of the time step.

N_d	4	16	36	64	100	144	196	256	324
Overall performances									
Time to solution (s)	737.0	834.9	944.9	1066.0	1163.0	1224.0	1493.0	1558.0	1727.0
Displacement subproblem									
Total number of iter.	929	1365	1736	2256	2655	2951	4529	4819	5366
Average number of iter.	16.59	24.38	31.00	40.29	47.41	52.70	80.88	86.05	95.82
Number of tangent systems	56	56	56	56	56	56	56	56	56
Solver time (s)	269.0	341.6	441.6	551.9	621.1	675.0	920.7	976.5	1113.0
Damage subproblem									
Total number of iter.	260	317	278	642	800	816	868	896	976
Average number of iter.	5.00	6.10	5.35	12.35	15.38	15.69	16.69	17.23	18.77
Number of tangent systems	52	52	52	52	52	52	52	52	52
Solver time (s)	8.065	9.789	10.05	15.42	17.56	18.33	20.27	20.21	21.86

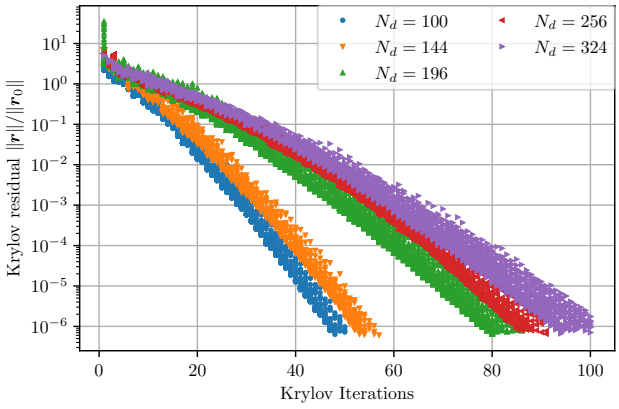
Table 5 Three dimensional plate with multiple cracks undergoing stable propagation: weak scalability study with automatic decomposition. FETI with a conjugate gradient is used for both physics.

results obtained with the automatic decomposition and Krylov subspace recycling are gathered in Table 6. A comparison with Table 5, shows that the total number of Krylov iterations is significantly reduced with SRKS for all values of N_d. For instance, the reduction is about 40% for $N_d = 324$. However, the reduction of the computational time is neither systemic nor significant. The best reduction is about 15% for $N_d \in (64, 144)$. For $N_d \in (196, 256)$, SRKS increases the time to solution by approximately 5%.

These results are due to extra computations induced by the recycling process. First, $\boldsymbol{C}_a$ has to satisfy $\boldsymbol{G}^\top \boldsymbol{C}_a = 0$, so the augmentation subspace is projected ($\boldsymbol{C}_a \leftarrow \boldsymbol{\Pi} \boldsymbol{C}_a$) at every resolution. Also, a singular value decomposition is done to ensure $\boldsymbol{C}_a$ is full rank. Then, the duration of one Krylov iteration is slightly higher with the augmentation projector $\boldsymbol{\Pi}_{\boldsymbol{C}_a}$.



(a) Convergence of the damage solver.



(b) Convergence of the mechanical solver.

Figure 11 Three dimensional plate with multiple cracks undergoing stable propagation: residual curves of the Conjugate Gradient solvers for the automatic decomposition. Convergence curves of all tangent systems are superimposed.

Moreover, subdomains are really small in order to limit the computational cost of the overall numerical campaign. Even for $N_d = 324$, the duration of one Krylov iteration is less than $0.2s$. Here, the decrease in Krylov iterations barely compensates these extra computational cost. Better results are expected with larger subdomains since one Krylov iteration involves forward and backward substitutions to compute the action of $\boldsymbol{F}$ and $\boldsymbol{M}^{-1}_{FETI}$.

N_d	4	16	36	64	100	144	196	256	324
Overall performances									
Time to solution (s)	725.4	816.2	899.6	990.3	1149.0	1120.0	1545.0	1616.0	1741.0
Displacement subproblem									
Total number of iter.	567	1015	1186	1343	1702	1753	2436	2465	3250
Average number of iter.	10.12	18.12	21.18	23.98	30.39	31.30	43.50	44.02	58.04
Number of tangent systems	56	56	56	56	56	56	56	56	56
Solver time (s)	258.8	325.5	396.9	469.8	616.1	578.4	970.8	1028.0	1125.0
Damage subproblem									
Total number of iter.	260	317	278	642	800	816	866	896	976
Average number of iter.	5.00	6.10	5.35	12.35	15.38	15.69	16.65	17.23	18.77
Number of tangent systems	52	52	52	52	52	52	52	52	52
Solver time (s)	8.475	9.771	10.06	15.23	17.59	18.27	20.41	21.05	22.28

Table 6 Three dimensional plate with multiple cracks undergoing stable propagation: weak scalability study with automatic decomposition. FETI with Krylov subspace recycling is used for the displacement subproblem.

5.3 | Strong scalability study

Despite our efforts, we were not able to design a weak scalability study for unstable crack propagation. Just removing the elastic stabilizer of the plate with multiple cracks is not satisfactory. Indeed, the prescribed displacement is proportional to the number of patterns in order to get the same local strain, but the strain rate increases with the number of patterns leading to dynamic effects such as waves propagation. Chercher une phrase pour compléter The evolution of the time step is not reproducible at all what prevents any meaningful comparison. So, only a strong scalability study is proposed to evaluate the performances in the case of unstable propagation.

5.3.1 | Three dimensional plate with multiple cracks undergoing unstable propagation

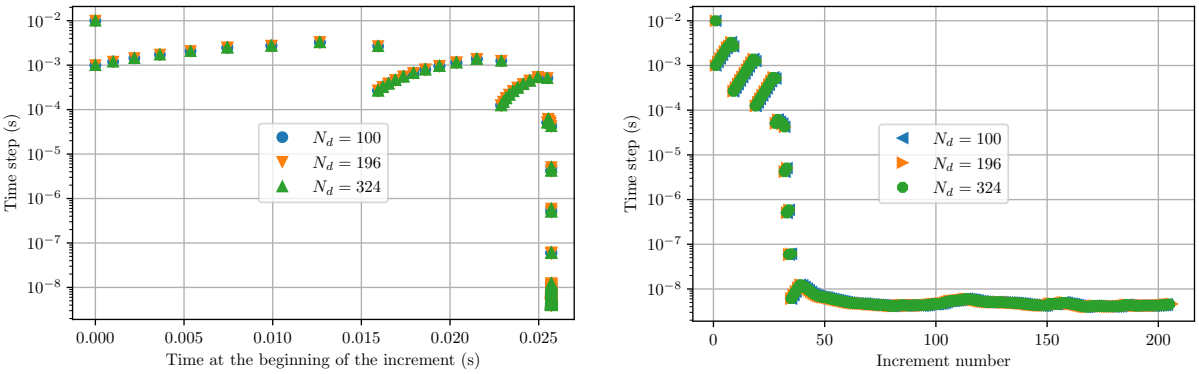
The test case used for the strong scalability study is the 18×18 plate pattern without the elastic stabilizer. The number of degrees of freedom is 17.8 millions for the mechanical problem and 5.9 millions for the damage one. The study starts from 100 subdomains and goes up to 324 subdomains. We were not able to use less than 100 subdomains due to the memory available on the computational nodes. Only the automatic domain decompositions are considered. There is approximately 18 000 (resp. 59 000) nodes per subdomain with 324 (resp. 100) subdomains.

For the damage subproblem, the FETI solver uses the Lumped variant defined in Table 1. For the displacement subproblem, the BDD method with a Conjugate gradient was not robust enough to converge for all tangent systems. It led to artificial time step reductions due to the stagnation or the divergence of the Krylov solver. The AMPBDD solver is used here to be robust enough. The underlying AMPCG solver uses 32 multipreconditioning aggregates and a global adaptive test with $\tau = 10^{-2}$. For both solvers, the convergence criterion is 10^{-6}. Regarding the staggered scheme, the stagnation threshold is $\epsilon = 0.05$, and the maximum number of fixed point iteration before reducing the time step is $j_{\max} = 5$.

The evolution of the time step is shown in Figure 12, a significant reduction of the time step is visible when the crack propagation becomes unstable. The results are gathered in Table 7. The number of Krylov iterations of the mechanical solver slightly increases with the number of subdomains. As seen in Table 7, the total number of iterations for the phase field subproblem with 196 subdomains is rather high which seems associated with a bad domain decomposition. This tendency is not visible for the displacement subproblem since the adaptive process of AMPCG compensates this with a larger search space. The overall efficiency defined as $(T_{100} \times 100)/(T_{N_d} \times N_d)$, is 88% for 196 subdomains and 81% for 324 subdomains.

As shown in Figure 13a, the convergence of the damage subproblem is stable with respect to the crack propagation and the number of subdomains. Regarding the mechanical solver (see Figure 13b), two trends are visible. The solver typically converges in less than 20 iterations when the time step is tiny whereas the convergence is slower for intermediate time steps.

The crack phase field at the end of the computation is shown in Figure 14. Some side effects are present close to the boundary. In the central part of the plate, all cracks propagation are similar.



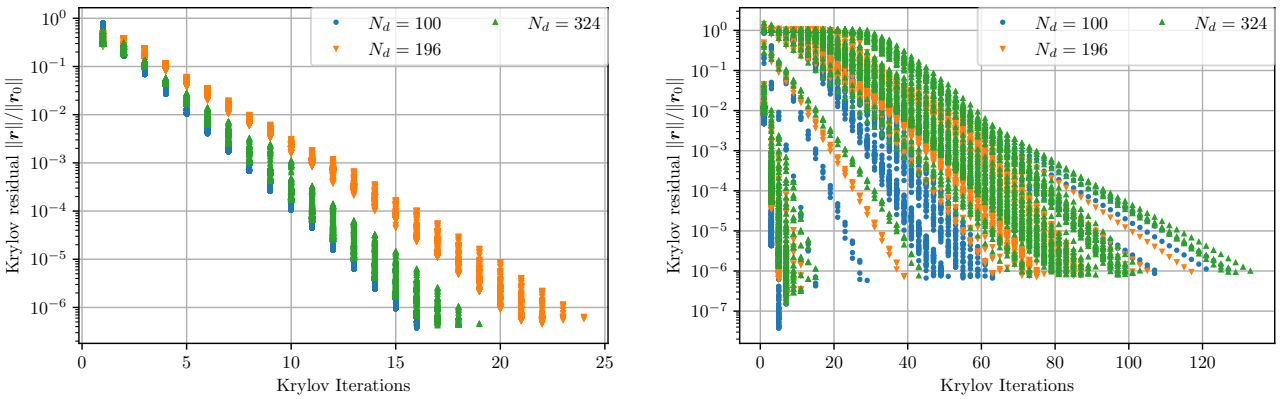
(a) Time step evolution w.r.t time.

(b) Time step evolution w.r.t increment number.

Figure 12 Three dimensional plate with multiple cracks undergoing stable propagation: evolution of the time

N_d	100	196	324
Overall performances			
Time to solution (s)	17080.0	9875.0	6511.0
Displacement subproblem			
Total number of iter.	5625	8254	9402
Average number of iter.	13.42	19.51	22.33
Number of tangent systems	419	423	421
Solver time (s)	4186.0	3083.0	2362.0
Damage subproblem			
Total number of iter.	6415	8944	6886
Average number of iter.	15.61	21.55	16.67
Number of tangent systems	411	415	413
Solver time (s)	322.5	279.2	132.5

Table 7 Three dimensional plate with multiple cracks undergoing unstable propagation: strong scalability study with automatic decomposition. The displacement subproblem uses AMPBDD and the damage one FETI.



(a) Convergence of the damage solver.

(b) Convergence of the mechanical solver.

Figure 13 Three dimensional plate with multiple cracks undergoing unstable propagation: residual curves of the Krylov solvers for the automatic decomposition. Convergence curves of all tangent systems are superimposed.

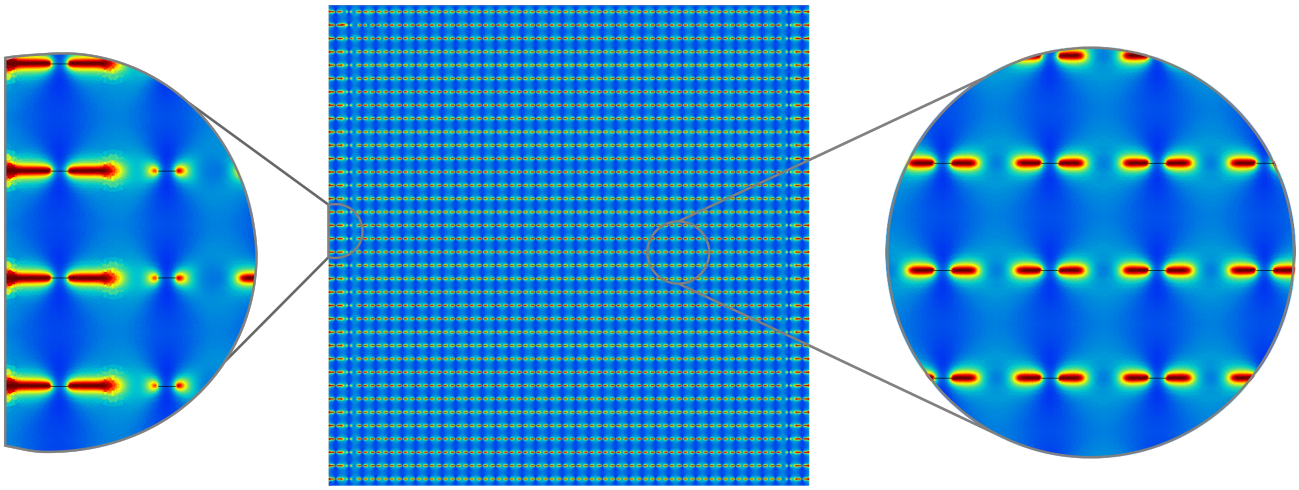


Figure 14 Three dimensional plate with multiple cracks undergoing unstable propagation: crack phase field at the final time.

6 | SEMI-INDUSTRIAL APPLICATIONS

As an example of industrial applications, we provide two test cases. We first focus on the two-dimensional case of matrix cracking in a transversely loaded CFRP at microscale, and we highlight the consequences of the a priori severe time step refinement strategy. The second application deals with a larger three dimensional problem.

6.1 | Cracking in CFRP at microscale level

This section investigates the case of matrix cracking in carbon fibers reinforced plastics (CFRP) loaded in the transverse (90°) direction. The simulation is led at the microscopic scale on a 570μm x 260μm rectangular sample as illustrated in Figure 15. The 90° ply is made of 1800 individual fibers and is surrounded by two domains standing for the adjacent 0° plies. Horizontal displacement at the left boundary and vertical displacement at the bottom left corner are blocked whereas a horizontal displacement is applied on the right boundary from 0 to 0.05 mm in a 50 s time range. We are interested in predicting both initiation and propagation of cracks in the matrix. An AT2 model is thus selected to model the matrix bulk with $E = 3.1$ GPa and $\nu = 0.3$ as elastic parameters and $G_c = 3.5\text{J.m}^{-2}$ and $l = 2.5\mu$m.

Fibers are supposed to be very tough in comparison to the matrix so that a linear elastic behavior is affected to both the individual fibers and the homogenized 0° plies. Carbon fibers exhibit a strong intrinsic anisotropy and elastic parameters in the transverse direction of individual fibers are taken to be $E = 26$ GPa and $\nu = 0.04$ whereas they are taken to be $E = 139$GPa and $\nu = 0.33$ in the top and bottom homogenized plies.

The meshes used to solve the displacement and the phase field subproblems are almost the same. That have exactly the same elements but as described in Section 5.2 and illustrated in Figure 5, the nodes are split at the fiber/matrix and matrix/ 0° plies interfaces in the phase field mesh whereas they are not in the displacement one. This allows the cracks to remain into the matrix and to not affect the fibers and surrounding plies while keeping a continuous description of the displacement field. The zoom in Figure 16 provides a representative view of the mesh which is composed of quadrilateral elements of characteristic size $h = 0.5\mu$m.

The whole model contains 4 160 000 and 1 170 000 dofs for the displacement and the phase field subproblems respectively. The overall mesh in decomposed into 42 domains as illustrated in Figure 15. The final result is shown in Figure 16.

Within the staggered scheme, the displacement subproblem is solved using a BDD solver because dynamic instabilities are likely to occur and a FETI solver is used to solve the phase field subproblem. The staggered scheme stagnation criterion is set at $\epsilon = 0.05$, and the time step is refined by a factor $\beta_t = 4$ if stagnation is not achieved within $j_{\max} = 6$ iterations and increased by a factor $\gamma_t = 1.025$ in the other case. Moreover, to prevent any bifurcation on a wrong instable branch and to ensure a proper

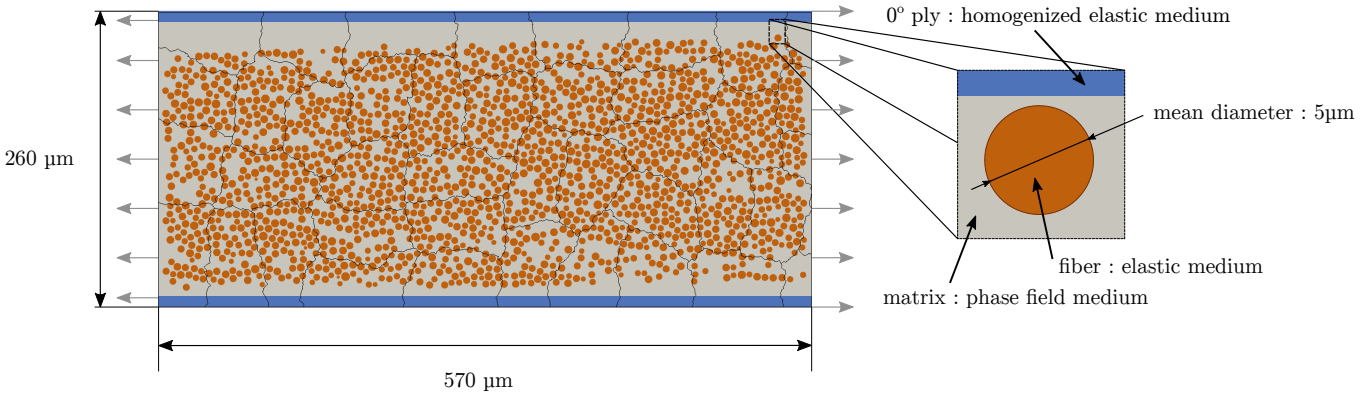


Figure 15 Schematic of the microstructure. Fibers are modeled with a linear elastic behavior while an AT2 phase field model is applied to the matrix

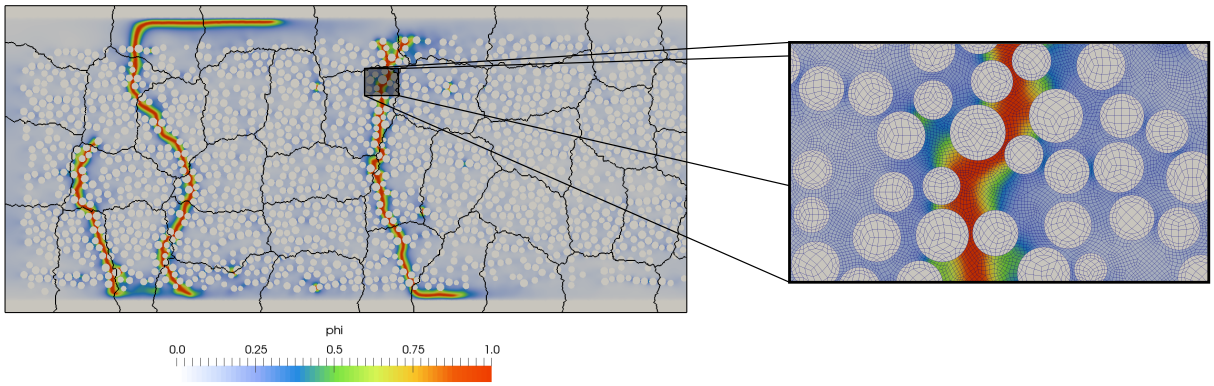


Figure 16 Crack pattern obtained after two dynamic events. The left image exhibits the domain decomposition and the zoom on the right side shows the mesh.

inertial regularization and a progressive propagation, the time step is also refined if any integration point undergoes an increase in ϕ of more than $\Delta\phi_{\max} = 0.9$ (see Algorithm 1).

Figure 17 illustrates the evolution of time stepping over the whole simulation range by plotting the time step as a function of the time. Each red cross corresponds to an increment that has been invalidated either because staggered did not achieve stagnation within 6 iterations either because $\Delta\phi_{\max} > 0.9$. The top graph exhibits two instable crack propagations corresponding to the two main transverse cracks of Figure 16. The bottom graph is a zoom over the first instable event. It clearly shows that a time step of about 10^{-8} s is required to properly capture the crack propagation. The proposed time refinement strategy can a priori appears to be very severe, but the present analysis shows that these low time steps actually correspond to the proper time scale at which inertia regularization operates, and they are not too low if one want to properly describe the continuous propagation of the cracks, thus avoiding spurious branching of the numerical solution. The bottom graph also shows, on top of each increment

marker, the number j of iterations required by the staggered algorithm. It can be seen that rather low number of iterations per increment ($j < j_{\max} = 6$) are sufficient at this time scale, which partially compensate the higher number of time steps. It can be noticed that during the two sudden drops in time steps, a lot of increments are not validated. To avoid this, one could think of significantly increase the factor $\beta_t = 4$. This would however penalize time stepping adaptations that are not due to dynamic instabilities such as the four drops occurring after 30 s which correspond to the sable propagation of the cracks along the 0°/90° plies interfaces.

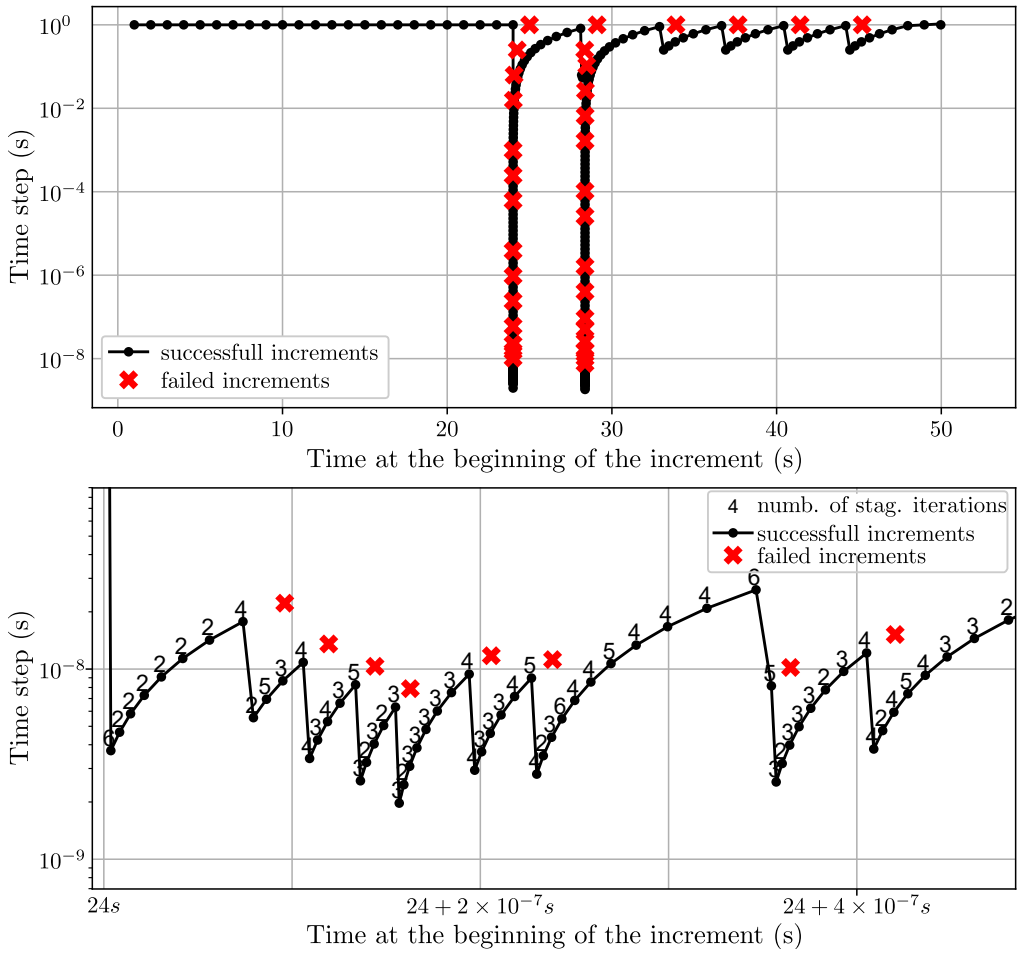


Figure 17 Time stepping evolution. The bottom graph is a zoom over the first dynamic instability.

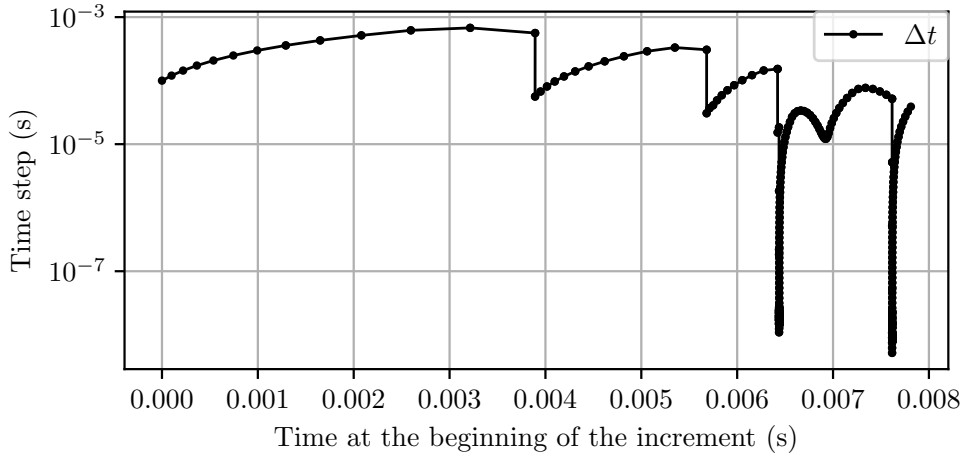
6.2 | Elastic cube with multiple cracks

The propagation of cracks inside an elastic cube submitted to triaxial traction is investigated in this section. The cube occupies a volume of $1mm^3$ and an homogeneous normal displacement is prescribed on its external faces $\boldsymbol{u}_d \cdot \boldsymbol{n} = 0.05 \times t$ where $\boldsymbol{n}$ is the outward normal vector. Sixty-four elliptic precracks are explicitly inserted in the mesh using the Z-cracks module[61] of the Z-set

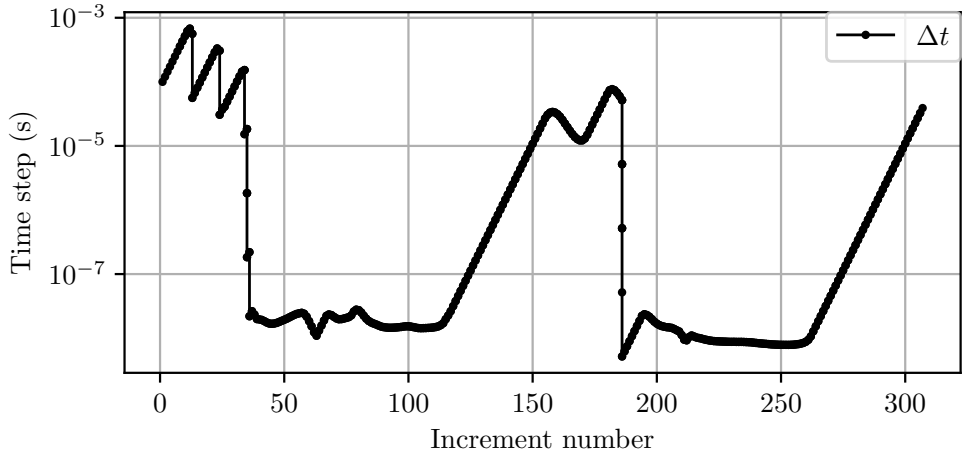
finite element suite. The elastic parameters of the cube are $E = 210$ GPa, $\nu = 0.3$. The AT2 model is used with $l = 0.04mm$ and $G_c = 0.1$ J.mm^{-2}. The location, orientation, size and eccentricity of the ellipses have been randomly selected. The finite element mesh contains 24 764 237 linear tetrahedrons and 4 206 393 nodes. Thus, the number of displacement degrees of freedom is 12 619 179 while the phase field subproblem has 4 206 393 unknowns. The decomposition in 224 subdomains is generated using an automatic graph partitioning library. As before the same decomposition is used for the phase field and the displacement subproblem.

The AMPBDD solver is used for the displacement subproblem, it uses 64 multipreconditioning aggregates and a global adaptive test with $\tau = 10^{-2}$. The solver for the damage subproblem is FETI equipped with a lumped preconditioner and a stiffness scaling. For both solver, the convergence criterion is 10^{-6}. Regarding the staggered scheme, the stagnation threshold is $\epsilon = 0.05$, and the maximum number of fixed point iteration before reducing the time step is $j_{\max} = 5$. The time to solution is about 14 hours, it was composed of 307 increments and 630 tangent linear system resolutions. The time step evolution is presented in Figure 18, the need for an adaptive time step is clearly visible here. During the simulation, two unstable crack propagation phases occur characterized by a significant decrease of the time step.

The evolution of the phase field variable are shown in Figure 19 and Figure 20. The simulation leads to complex crack patterns. We can see the coalescence of cracks around the few precracks, few cracks temporarily stop then propagate again. Finally, a dominant crack is formed that completely cuts the cube.



(a) Time step evolution w.r.t time.



(b) Time step evolution w.r.t increment number.

Figure 18 Elastic cube with multiple cracks: evolution of the time step.

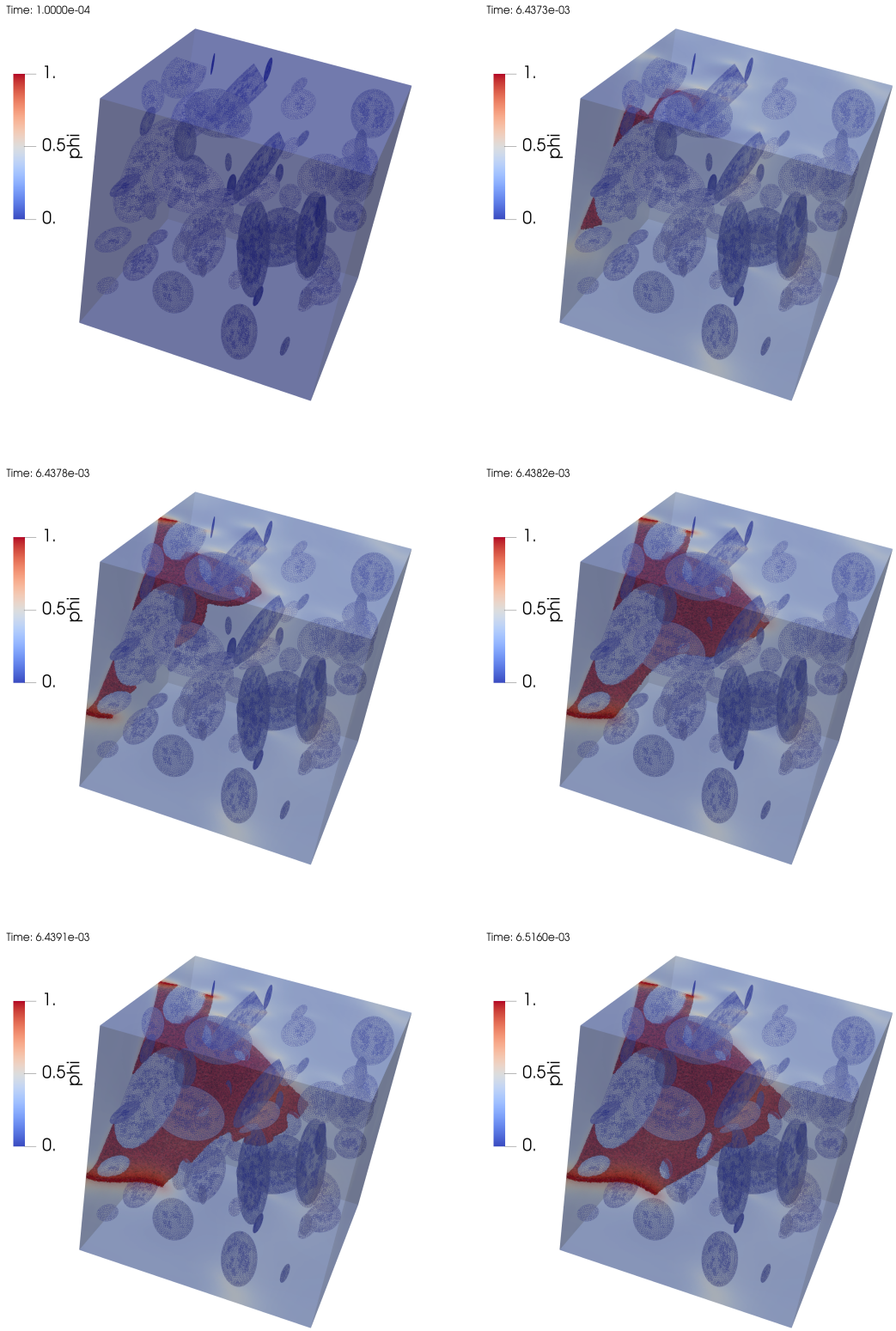


Figure 19 Elastic cube with multiple cracks: evolution of the phase field variable ϕ (part1)

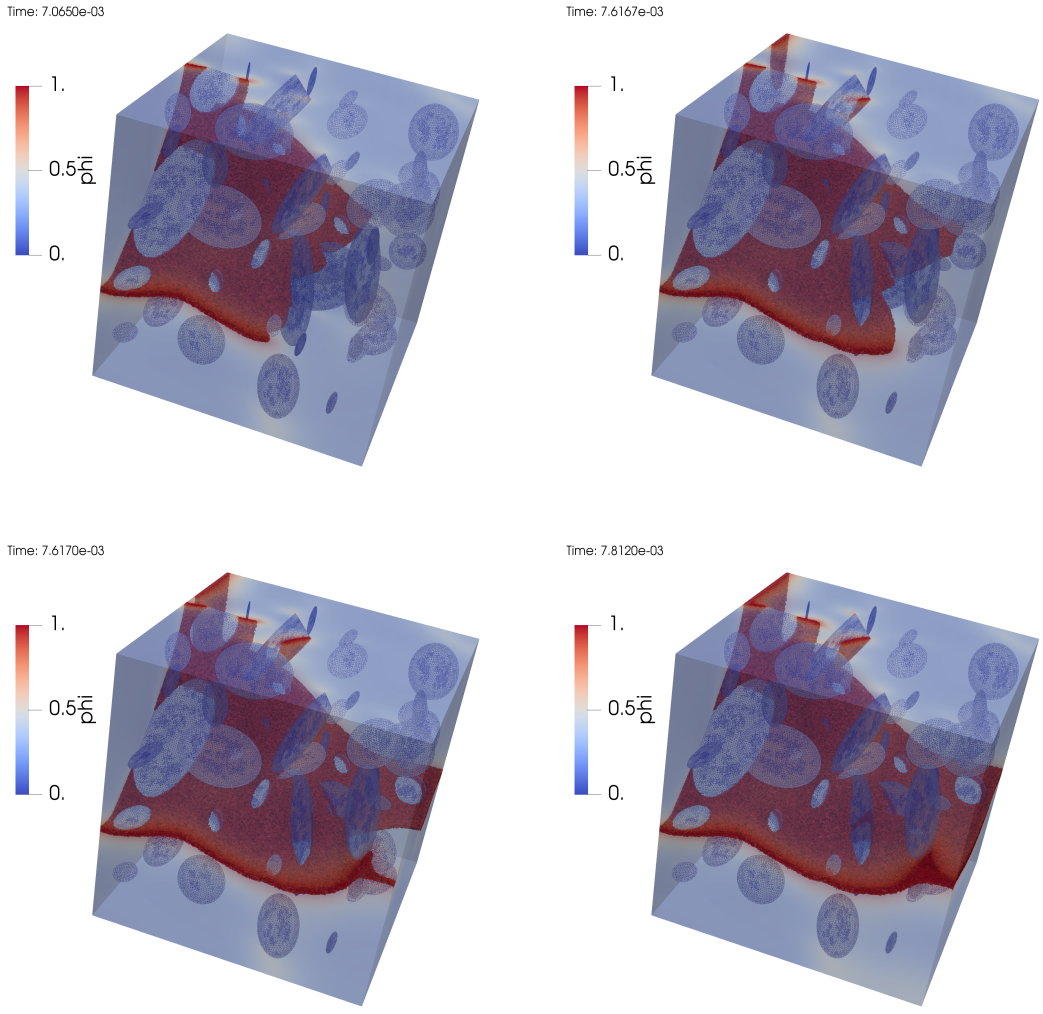


Figure 20 Elastic cube with multiple cracks: evolution of the phase field variable ϕ (part2)

7 | CONCLUSION AND PERSPECTIVES

In this article, we have proposed a domain decomposition framework and acceleration techniques for the phase field fracture staggered solver. The implementation is fully parallel and both the damage and the mechanical problems are solved with a domain decomposition solver. An implicit dynamics scheme with automatic time stepping is used to simulate unstable crack propagation problems. The damage subproblem is solved with a FETI solver, a numerical experiment has shown that cheap preconditioners (lumped and super lumped) still lead to a good convergence rate. Also, the lack of a natural coarse problem does not hinder the parallel scalability of the methods since the right-hand-side of the damage subproblem is dense. The displacement subproblem is solved by a FETI, BDD or adaptive variants. When dealing with unstable crack propagation the primal methods are to be preferred since they are less sensitive to a misdetection of the nullspace of the local stiffness operators. To compensate the absence of natural coarse problem in dynamics, the multipreconditioned variant AMPBDD has been proposed. For stable crack propagation with a quasi-static scheme, all methods can be used.

A weak scalability study up to 324 subdomains with numerous crack propagation has been provided for the case of stable crack propagation. For unstable crack propagation, a strong scalability study up to 324 subdomains has shown a parallel efficiency greater than 80%. Domain decomposition automatically generated by a graph partitioning library have been used for these scalability studies, it lower the parallel performances, but it is more representative of industrial applications. Two semi-industrial applications are also presented showing the good performances of the approach.

Regarding the parallel framework of this work, one perspective is to be able to use different domain decompositions for the mechanical and the damage subproblem. It makes more complex the communication pattern of the approach, but it permits to use less cores for the damage subproblem. Another perspective is to combine this approach with a parallel mesh adaptation process in order to reduce the overall cost of the simulation. Finally, this parallel framework may be used for more complex models involving, for example, hyperelasticity[62], finite strain ductile failure[63] or cohesive zone models[64].

ACKNOWLEDGEMENTS

The authors would like to thank F. Laurin for proposing the composite test case and C.P. Przybyla for providing a fiber/matrix mesh generation tool.

References

1. Francfort G, Marigo JJ. Revisiting brittle fracture as an energy minimization problem. *Journal of the Mechanics and Physics of Solids* 1998; 46(8): 1319-1342. doi: https://doi.org/10.1016/S0022-5096(98)00034-9

2. Bourdin B, Francfort G, Marigo JJ. Numerical experiments in revisited brittle fracture. *Journal of the Mechanics and Physics of Solids* 2000; 48(4): 797-826. doi: https://doi.org/10.1016/S0022-5096(99)00028-9

3. Tanné E, Li T, Bourdin B, Marigo JJ, Maurini C. Crack nucleation in variational phase-field models of brittle fracture. *Journal of the Mechanics and Physics of Solids* 2018; 110: 80 - 99. doi: https://doi.org/10.1016/j.jmps.2017.09.006

4. Wu JY, Nguyen VP, Nguyen CT, Sutula D, Bordas S, Sinaie S. Phase field modeling of fracture. *Advances in Applied Mechanics: Multi-scale Theory and Computation* 2019; 52. doi: https://doi.org/10.1016/bs.aams.2019.08.001

5. Gerasimov T, Lorenzis LD. A line search assisted monolithic approach for phase-field computing of brittle fracture. *Computer Methods in Applied Mechanics and Engineering* 2016; 312: 276 - 303. Phase Field Approaches to Fracturedoi: https://doi.org/10.1016/j.cma.2015.12.017

6. Lorentz E, Badel P. A new path-following constraint for strain-softening finite element simulations. *International journal for numerical methods in engineering* 2004; 60(2): 499–526.

7. Farrell P, Maurini C. Linear and nonlinear solvers for variational phase-field models of brittle fracture. *International Journal for Numerical Methods in Engineering* 2017; 109(5): 648–667. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.5300doi: 10.1002/nme.5300

8. Storvik E, Both JW, Sargado JM, Nordbotten JM, Radu FA. An accelerated staggered scheme for variational phase-field models of brittle fracture. *Computer Methods in Applied Mechanics and Engineering* 2021; 381: 113822. doi: 10.1016/j.cma.2021.113822

9. Lu Y, Helfer T, Bary B, Fandeur O. An efficient and robust staggered algorithm applied to the quasi-static description of brittle fracture by a phase-field approach. *Computer Methods in Applied Mechanics and Engineering* 2020; 370: 113218. doi: 10.1016/j.cma.2020.113218

10. Liu G, Li Q, Msekh M, Zuo Z. Abaqus implementation of monolithic and staggered schemes for quasi-static and dynamic fracture phase-field model. *Computational Materials Science* 2016; 121: 35–47. ZSCC: 0000065doi: 10/gcpfcp

11. Zhang P, Yao W, Hu X, Zhuang X. Phase field modelling of progressive failure in composites combined with cohesive element with an explicit scheme. *Composite Structures* 2021; 262: 113353. doi: 10.1016/j.compstruct.2020.113353

12. Ziaei-Rad V, Shen Y. Massive parallelization of the phase field formulation for crack propagation with time adaptivity. *Computer Methods in Applied Mechanics and Engineering* 2016; 312: 224–253. doi: 10.1016/j.cma.2016.04.013

13. Baker AH, Kolev TV, Yang UM. Improving algebraic multigrid interpolation operators for linear elasticity problems: IMPROVING AMG INTERPOLATION FOR ELASTICITY. *Numerical Linear Algebra with Applications* 2010; 17(2-3): 495–517. doi: 10.1002/nla.688

14. Li T, Marigo JJ, Guilbaud D, Potapov S. Gradient damage modeling of brittle fracture in an explicit dynamics context. *International Journal for Numerical Methods in Engineering* 2016; 108(11): 1381–1405. ZSCC: 0000055 _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.5262doi: 10/f9chgx

15. Mandel J. Balancing domain decomposition. *Communications in Numerical Methods in Engineering* 1993; 9(3): 233. doi: 10.1002/cnm.1640090307

16. Farhat C, Roux FX. A method of finite element tearing and interconnecting and its parallel solution algorithm. *International Journal for Numerical Methods in Engineering* 1991; 32(6): 1205. doi: 10.1002/nme.1620320604

17. Farhat C, Lesoinne M, LeTallec P, Pierson K, Rixen D. FETI-DP: a dual–primal unified FETI method—part I: A faster alternative to the two-level FETI method. *International Journal for Numerical Methods in Engineering* 2001; 50(7): 1523–1544. doi: 10/cz9h5t

18. Dohrmann CR. A Preconditioner for Substructuring Based on Constrained Energy Minimization. *SIAM Journal on Scientific Computing* 2003; 25(1): 246–258. Publisher: Society for Industrial and Applied Mathematicsdoi: 10.1137/S1064827502412887

19. Gosselet P, Rixen D, Roux FX, Spillane N. Simultaneous FETI and block FETI: Robust domain decomposition with multiple search directions. *International Journal for Numerical Methods in Engineering* 2015; 104(10): 905–927. nme.4946doi: 10.1002/nme.4946

20. Bovet C, Parret-Fréaud A, Spillane N, Gosselet P. Adaptive multipreconditioned FETI: Scalability results and robustness assessment. *Computers & Structures* 2017: 1–20. doi: 10.1016/j.compstruc.2017.07.010

21. Bovet C, Parret-Fréaud A, Gosselet P. Two-level adaptation for Adaptive Multipreconditioned FETI. *Advances in Engineering Software* 2021; 152: 102952. doi: https://doi.org/10.1016/j.advengsoft.2020.102952

22. Rixen DJ, Farhat C. A simple and efficient extension of a class of substructure based preconditioners to heterogeneous structural mechanics problems. *International Journal for Numerical Methods in Engineering* 1999; 44(4): 489–516. doi: 10.1002/(SICI)1097-0207(19990210)44:4<489::AID-NME514>3.0.CO;2-Z

23. Bovet C. On the use of graph centralities to compute generalized inverse of singular finite element operators: applications to the analysis of floating substructures. working paper or preprint; 2022.

24. Ambrosio L, Tortorelli VM. Approximation of functional depending on jumps by elliptic functional via t-convergence. *Communications on Pure and Applied Mathematics* 1990; 43(8): 999–1036. doi: https://doi.org/10.1002/cpa.3160430805

25. Bedford A. *Hamilton's principle in continuum mechanics*. 139. Springer . 1985.

26. Griffith AA. The Phenomena of Rupture and Flow in Solids. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character* 1921; 221: 163–198. doi: https://doi.org/10.1098/rsta.1921.0006

27. Miehe C, Welschinger F, Hofacker M. Thermodynamically consistent phase-field models of fracture: Variational principles and multi-field FE implementations. *International Journal for Numerical Methods in Engineering* 2010; 83(10): 1273–1311. doi: https://doi.org/10.1002/nme.2861

28. Ambati M, Gerasimov T, De Lorenzis L. A review on phase-field models of brittle fracture and a new fast hybrid formulation. *Computational Mechanics* 2015; 55(2): 383–405.

29. Borden MJ, Verhoosel CV, Scott MA, Hughes TJ, Landis CM. A phase-field description of dynamic brittle fracture. *Computer Methods in Applied Mechanics and Engineering* 2012; 217: 77–95.

30. Peerlings R, Borst dR, Brekelmans W, deVree J. Gradient enhanced damage for quasi-brittle materials. *International Journal for Numerical Methods in Engineering* 1996; 39(19): 3391-3403. doi: 10.1002/(SICI)1097-0207(19961015)39:19<3391::AID-NME7>3.0.CO;2-D

31. Lorentz E, Andrieux S. Analysis of non-local models through energetic formulations. *International Journal of Solids and Structures* 2003; 40(12): 2905–2936. doi: https://doi.org/10.1016/S0020-7683(03)00110-0

32. Borst dR, Verhoosel CV. Gradient damage vs phase-field approaches for fracture: Similarities and differences. *Computer Methods in Applied Mechanics and Engineering* 2016; 312: 78 - 94. Phase Field Approaches to Fracturedoi: http://dx.doi.org/10.1016/j.cma.2016.05.015

33. Borst dR. Damage, Material Instabilities, and Failure. *Encyclopedia of computational mechanics* 2004; volume 2. doi: https://doi.org/10.1002/9781119176817.ecm2035

34. Pijaudier-Cabot G, Bažant Z. Nonlocal damage theory. *Journal of Engineering Mechanics* 1987; 113: 1512–1533. doi: https://doi.org/10.1061/(ASCE)0733-9399(1987)113:10(1512)

35. Lemaitre J, Chaboche JL. *Mechanics of solid materials*. Cambridge university press . 1994

36. Chaboche JL, Maire JF. A new micromechanics based CDM model and its application to CMC's. *Aerospace Science and Technology* 2002; 6(2): 131–145.

37. Germain N, Besson J, Feyel F. Simulation of laminate composites degradation using mesoscopic non-local damage model and non-local layered shell element. *Modelling and Simulation in Materials Science and Engineering* 2007; 15(4): S425. doi: 10.1088/0965-0393/15/4/S08

38. Wu JY, Huang Y, Nguyen VP. On the BFGS monolithic algorithm for the unified phase field damage theory. *Computer Methods in Applied Mechanics and Engineering* 2020; 360: 112704.

39. Kristensen PK, Martínez-Pañeda E. Phase field fracture modelling using quasi-Newton methods and a new adaptive step scheme. *Theoretical and Applied Fracture Mechanics* 2020; 107: 102446.

40. Hughes T. *The finite element method: linear static and dynamic finite element analysis*. Prentice-hall . 1987.

41. Transvalor S.A.*Z-set 9.1 user manual*. 2022.

42. Chantrait T, Rannou J, Gravouil A. Low intrusive coupling of implicit and explicit time integration schemes for structural dynamics: Application to low energy impacts on composite structures. *Finite Elements in Analysis and Design* 2014; 86: 23-33. doi: https://doi.org/10.1016/j.finel.2014.03.011

43. Rannou J, Huchette C, Hurmane A, Tholin F. An analysis of ligthning direct effects on CFRP panels. Part II, the electro-thermo-mechanical simulation strategy. In: TODO. ; 2017.

44. Coupling With Interpolation Parallel Interface. (https://w3.onera.fr/cwipi/bibliotheque-couplage-cwipi); . Accessed: 2022-11-01.

45. Klawonn A, Lanser M, Rheinbach O. Nonlinear FETI-DP and BDDC Methods. *SIAM Journal on Scientific Computing* 2014; 36(2): A737–A765.

46. Negrello C, Gosselet P, Rey C, Pebrel J. Substructured formulations of nonlinear structure problems — Influence of the interface condition. *International Journal for Numerical Methods in Engineering* 2016; 107(13): 1083–1105. doi: 10.1002/nme.5195

47. Gosselet P, Rey C. Non-overlapping domain decomposition methods in structural mechanics. *Archives of Computational Methods in Engineering* 2006; 13(4): 515–572.

48. Farhat C, Géradin M. On the general solution by a direct method of a large-scale singular system of linear equations: application to the analysis of floating structures. *International Journal for Numerical Methods in Engineering* 1998; 41(4): 675–696. doi: 10.1002/(SICI)1097-0207(19980228)41:4<675::AID-NME305>3.0.CO;2-8

49. Farhat C, Géradin M. On the general solution by a direct method of a large-scale singular system of linear equations: application to the analysis of floating structures. *International Journal for Numerical Methods in Engineering* 1998; 41(4): 675–696. doi: 10.1002/(SICI)1097-0207(19980228)41:4<675::AID-NME305>3.0.CO;2-8

50. Brzobohatý T, Dostál Z, Kozubek T, Kovář P, Markopoulos A. Cholesky decomposition with fixing nodes to stable computation of a generalized inverse of the stiffness matrix of a floating structure. *International Journal for Numerical Methods in Engineering* 2011; 88(5): 493–509. doi: 10.1002/nme.3187

51. Dostál Z, Kozubek T, Markopoulos A, Menšík M. Cholesky decomposition of a positive semidefinite matrix with known kernel. *Applied Mathematics and Computation* 2011; 217(13): 6067–6077.

52. Farhat C, Mandel J. The two-level FETI Method for Static and Dynamic Plate Problems - Part I: An Optimal Iterative Solver for Biharmonic systems. *Computer Methods in Applied Mechanics and Engineering* 1998; 155: 129–152.

53. Farhat C, Chen PS, Roux FX. The two-level FETI Method - Part II: Extension to Shell Problems. Parallel Implementation and Performance Results. *Computer Methods in Applied Mechanics and Engineering* 1998; 155: 153–180.

54. Farhat C, Roux FX. Implicit parallel processing in structural mechanics. *Computational Mechanics Advances* 1994; 2(1): 1–124. North-Hollanddoi: 10.1002/nme.1620371111

55. Farhat C, Crivelli L, Roux FX. Extending substructure based iterative solvers to multiple load and repeated analyses. *Computer Methods in Applied Mechanics and Engineering* 1994; 117(1): 195–209.

56. Gosselet P, Rey C, Pebrel J. Total and selective reuse of Krylov subspaces for the solution to a sequence of non-linear structural problems. *International Journal for Numerical Methods in Engineering* 2013; 94(1): 60–83. doi: 10.1002/nme.4441

57. Spillane N, Rixen DJ. Automatic spectral coarse spaces for robust FETI and BDD algorithms. *International Journal for Numerical Methods in Engineering* 2013; 95(11): 953–990. doi: 10.1002/nme.4534

58. Spillane N. An Adaptive Multipreconditioned Conjugate Gradient Algorithm. *SIAM J. Sci. Comput.* 2016; 38(3): A1896–A1918. doi: 10.1137/15M1028534

59. MUMPS: MUltifrontal Massively Parallel sparse direct Solver version 5.1.2. https://mumps-solver.org/; . Accessed: 2022-11-01.

60. Eigen library. https://eigen.tuxfamily.org; . Accessed: 2022-11-01.

61. Chiaruttini V, Riolo V, Feyel F. Advanced remeshing techniques for complex 3D crack propagation. In: 13th International Conference on Fracture. ; 2013: 16–21.

62. Russ J, Slesarenko V, Rudykh S, Waisman H. Rupture of 3D-printed hyperelastic composites: Experiments and phase field fracture modeling. *Journal of the Mechanics and Physics of Solids* 2020; 140: 103941. doi: 10.1016/j.jmps.2020.103941

63. Ambati M, Kruse R, De Lorenzis L. A phase-field model for ductile fracture at finite strains and its experimental verification. *Computational Mechanics* 2016; 57(1): 149–167. doi: 10.1007/s00466-015-1225-3

64. Tarafder P, Dan S, Ghosh S. Finite deformation cohesive zone phase field model for crack propagation in multi-phase microstructures. *Computational Mechanics* 2020; 66(3): 723–743. doi: 10.1007/s00466-020-01874-6

How to cite this article: J. Rannou, C. Bovet (2022), Domain decomposition methods and acceleration techniques for the phase field fracture staggered solver, *Int. J. Numer. Meth. Engng.*, *2022;00:1–6*.