



Enforcing local boundary conditions in peridynamic models of diffusion with singularities and on arbitrary domains

Jiangming Zhao¹ · Siavash Jafarzadeh¹ · Ziguang Chen^{2,3} · Florin Bobaru¹ 

Received: 25 October 2023 / Accepted: 19 April 2024

© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2024

Abstract

Imposing local boundary conditions and mitigating the surface effect at free surfaces in peridynamic (PD) models are often desired. The fictitious nodes method (FNM) “extends” the domain with a thin fictitious layer of thickness equal to the PD horizon size, and is a commonly used technique for these purposes. The FNM, however, is limited, in general, to domains with simple geometries. Here we introduce an algorithm for the mirror-based FNM that can be applied to arbitrary domain geometries. The algorithm automatically determines mirror nodes (in the given domain) of all fictitious nodes based on approximating, at each fictitious node, the “generalized” (or nonlocal) normal vector to the domain boundary. We tested the new algorithm for a peridynamic model of a classical diffusion problem with a flux singularity on the boundary. We show that other types of FNMs exhibit “pollution” of the solution far from the singularity point, while the mirror-based FNM does not and, in addition, shows a significantly faster rate of convergence to the classical solution in the limit of the horizon going to zero. The new algorithm is then used for mirror-based FNM solutions of diffusion problems in domains with curvilinear boundaries and with intersecting cracks. The proposed algorithm significantly improves the accuracy near boundaries of domains of arbitrary shapes, including those with corners, notches, and crack tips.

✉ Florin Bobaru
fbobaru2@unl.edu

¹ Department of Mechanical & Materials Engineering,
University of Nebraska-Lincoln, Lincoln, NE 68588-0526,
USA

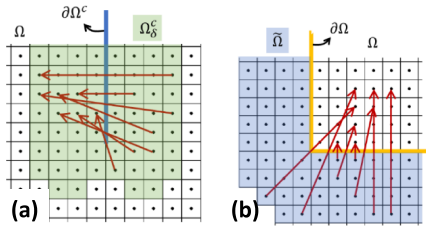
² Department of Mechanics, Huazhong University of Science
and Technology, Wuhan 430074, China

³ Hubei Key Laboratory of Engineering Structural Analysis
and Safety Assessment, 1037 Luoyu Road, Wuhan 430074,
China

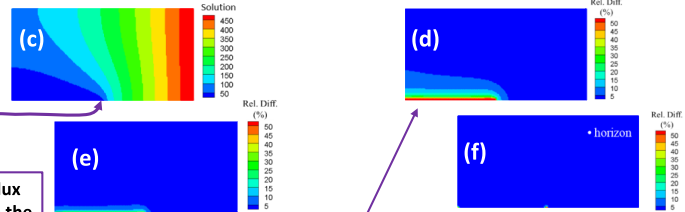
Graphical Abstract

Enforcing local boundary conditions in peridynamic models of diffusion with singularities and on arbitrary domains

Automatic determination of mirror nodes: **(a)** near a crack tip; **(b)** near a domain corner



New algorithm eliminates “pollution” of the PD solution far from the **classical singularity location**. In **(c)**: classical sol. for Motz problem; Rel. diff. from classical sol. for PD model with BCs imposed with the: **(d)** naive method, **(e)** Taylor method, and **(f)** the new method (mirror Fictitious Nodes Method).

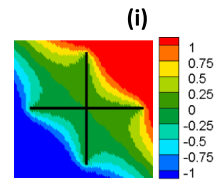
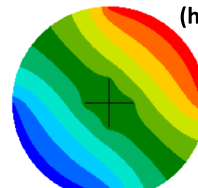
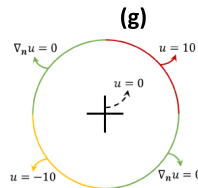


Location of flux singularity in the classical model

Existing methods pollute the solution far from singularity

New method avoids polluting the solution

New algorithm finds PD solution for diffusion in a domain with **curved boundaries** and **intersecting insulated cracks**; **(g)** boundary conditions; **(h)** PD solution, **(i)** zoom-in near crack tips.



Keywords Peridynamics · Diffusion · Singularities · Surface effect · Fictitious nodes method

List of symbol

A	The constant in the PD kernel function
D_i	Generalized flux intensity factors
f	An arbitrary scalar function that specifies the Robin boundary condition
s	Source/sink term in diffusion equation
t, T	Time
u	Scalar quantity of interest, e.g., temperature and concentration
$\mathbb{R}^d$	Real coordinate space
$\mathbb{R}^+, \mathbb{R}^-$	The sets of positive and negative real numbers
ρ, θ	Polar coordinates
ν	Diffusivity
Ω	Problem domain
Ω_δ	Subdomain within Ω and within δ distance to the domain boundary
$\partial\Omega$	Domain boundary
$\partial\Omega_D$	Boundary subjected to Dirichlet condition
$\partial\Omega_N$	Boundary subjected to Neumann condition
$\partial\Omega_{NR}$	Boundary subjected to Neumann or Robin condition
$\tilde{\Omega}$	Fictitious region

$\tilde{\Omega}_D$

Fictitious region subjected to Dirichlet volume-constraint

$\tilde{\Omega}_{NR}$

Fictitious region subjected to Neumann or Robin volume-constraint

$\hat{\Omega}$

A regular-shaped region covering Ω and $\tilde{\Omega}$

$\mathbf{x}$

A point of interest

$\tilde{\mathbf{x}}$

A point within the fictitious domain

$\mathbf{x}_b$

A point on domain boundary

$\mathbf{y}$

A point within the horizon of $\mathbf{x}$

d

Distance between $\mathbf{x} \in \Omega$ and the boundary

$\tilde{d}$

Distance between $\mathbf{y} \in \tilde{\Omega}$ and the boundary

$\mathcal{L}_\omega$

PD Laplacian operator

ω

PD kernel function

$\mathcal{H}_\mathbf{x}$

Horizon region for an arbitrary point $\mathbf{x}$

$\tilde{\mathcal{H}}_{\tilde{\mathbf{x}}}$

$\mathcal{H}_{\tilde{\mathbf{x}}} \cap \Omega$ For an arbitrary point $\tilde{\mathbf{x}} \in \tilde{\Omega}$

δ

Horizon size

Δx

Grid size

m

The ratio of horizon size and grid size

$\xi(\mathbf{x}, \mathbf{y})$

Vector pointing from $\mathbf{x}$ to $\mathbf{y}$

$\text{dist}(\mathbf{x}, \mathbf{y})$

Distance between points $\mathbf{x}$ and $\mathbf{y}$

$\text{dist}(\mathbf{x}, \partial\Omega)$	Distance between points $\mathbf{x}$ and the boundary $\partial\Omega$
$\text{dist}(\tilde{\mathbf{n}}, \mathbf{y})$	Perpendicular distance between point $\mathbf{y}$ and the vector $\tilde{\mathbf{n}}$
u_D	Dirichlet BC
q	Flux
$\tilde{u}$	Value of u approximated at a point in the fictitious region
u_b	Value of u at a point on the domain boundary
$\mathbf{x}^R$, or $\text{Ref}_{\partial\Omega}(\tilde{\mathbf{x}})$	The reflection, or mirror point, of $\tilde{\mathbf{x}}$ across the boundary
$\mathbf{x}^P$, or $\text{OProj}_{\partial\Omega}(\tilde{\mathbf{x}})$	The orthogonal projection of $\tilde{\mathbf{x}}$ onto the boundary
$\mathbf{n}$	Outward unit normal vector
$\mu(\mathbf{x}, \mathbf{y})$	Binary function to track the status of bond between $\mathbf{x}$ and $\mathbf{y}$
$\hat{\mathbf{n}}$	“Generalized” normal vector at a point in the problem domain
$\tilde{\mathbf{n}}$	“Generalized” normal vector at a point in the fictitious region

1 Introduction

The peridynamic (PD) theory [1], as a nonlocal extension of classical continuum mechanics, allows for the natural treatment of discontinuities/singularities (such as cracks/damages [2–5]) by employing integration, over a nonlocal region called the horizon region, rather than differentiation. While the PD method has been primarily used to deal with mechanical behaviors [2, 4, 6–9], it has also been employed in diffusion-type problems involving cracks and damage, including thermal diffusion [10–13] and mass transport (e.g. corrosion) [14–20]. For a nonlocal formulation, associated BCs are of nonlocal type as well, and they are sometimes referred to as “nonlocal volume constraints” [21, 22]. In reality, however, conditions to be imposed (on values and/or derivatives of the unknown function) are known (measurable) only at the surface of a body, not through a finite thickness layer at the surface. The natural representation of such measurement-based conditions is via local boundary conditions. Therefore, imposing local BCs in nonlocal/peridynamic models is often desired/needed. Another issue caused by nonlocality is the “peridynamic surface effect” [23] which appears because, unlike in the bulk, points near the free surface/boundary do not have a full horizon region. The PD surface effect leads to the slightly different behavior of material points near the surface compared with those in the bulk.

A few strategies have been introduced to tackle these issues in the literature. One is to get rid of the nonlocality at boundaries, either by decomposing the domain into

local and nonlocal subdomains where the former is placed in the neighborhood of the boundary [24] or via using a variable horizon which decreases from a constant value in the interior of the domain to zero at the boundary [25]. These approaches, however, do not work for problems in which nonlocality/discontinuity plays an important role near the boundary (e.g., in problems with interfaces), increase complexity since the coupling of local and nonlocal models is not trivial, or lead to imbalances in bond force/flux between material points caused by using variable-size horizons [26]. A different concept has been recently introduced in [27] to improve results in dynamic brittle/quasi-brittle fracture in samples with complex geometries, like perforated plates, by using non-uniform grids near a boundary. Another strategy is explored by Aksoylu et al. [28–31] who proposed new operators that agree with the original PD operator in the bulk of the domain and simultaneously enforce local BCs. However, this approach, to the best of our understanding, appears to be limited to rectangular/box domains.

Another popular strategy for reducing the PD surface effect and to impose local BCs is to extend the solution domain by a fictitious layer with the thickness of the PD horizon so that each point in the solution domain has a full horizon region. Then the local BCs (including traction-free BCs) are converted into nonlocal volume constraints to be imposed on this fictitious region. This strategy is called the fictitious nodes method (FNM), or the extended domain method (EDM), and can be further classified into several categories based on the rule of conversion [11, 12, 32–36]. Similar ideas have been used in other nonlocal numerical models [37, 38]. Some of these FNMs require the reformulation of governing equations for each type of problem [33, 36] and thus may not be suitable for general applications. Others do not involve modifying the governing equations and have been proven to work efficiently for problems set on domains with simple geometry (e.g. rectangular) [39, 40], but applying them to problems with irregular geometries (e.g. curved boundaries, kinks, corners, and cracks) is not possible. Specifically, for the mirror-based FNM which determines the volume constraint at each fictitious node based on the value of its mirror node in the solution domain [34, 35, 41], the absence of a general algorithm that can find the mirror nodes (of all fictitious nodes) required by the method makes it inapplicable to problems defined on arbitrary-shape domains.

In this work, we introduce a new algorithm that automatically finds mirror nodes for the mirror-based FNM for domains of arbitrary geometries, including those with cracks. The algorithm approximates, at each fictitious node, a “generalized” normal vector (the “nonlocal normal”) to the boundary of the given domain if the boundary satisfies 1st-order smoothness conditions. We select the PD diffusion model to test the new algorithm. With small modifications,

the algorithm presented here can easily be applied to PD models solving other types of problems, such as fracture and corrosion damage. We also investigate two other types of FNMs and compare their results with those from the mirror-based FNM: the “naive” version [42], and the Taylor FNM [32, 37, 43]. We compare the performance of different FNMs in enforcing local boundary conditions using two problems: one set on a simple domain with no singularities, and the other with a flux singularity (in the corresponding local model) along a boundary where Dirichlet and Neumann boundary conditions meet, the Motz problem [44, 45]. We show that the new algorithm can let us use the mirror-based FNM to solve diffusion problems in domains with cracks and curved boundaries.

It should be noted that the main focus of this work is methodology and algorithm development, accompanied by verification via several important and relatively difficult examples. A practical algorithm is described in detail and possible extensions are discussed. While numerical experiments may offer insight into the well-posedness of the problems we investigated here, proofs on existence, uniqueness, and well-posedness are outside of the scope of our manuscript.

This paper is organized as follows: in Sect. 2 we review the PD method for diffusion-type problems; in Sect. 3 we discuss the fictitious nodes method and introduce the autonomous algorithm to generalize the mirror-based FNM for arbitrary geometries; in Sect. 4 we compare the performance of three different types of FNMs using examples with and without local singularities, then test the generality and capability of the algorithm introduced here on the mirror-based FNM for more complicated problems with cracks and in domains with curved boundaries; conclusions are given in Sect. 5.

2 The peridynamic model for diffusion

Consider the diffusion of a scalar quantity u (e.g., temperature, mass, concentration, etc.) in a homogeneous and isotropic body occupying the domain $\Omega \in \mathbb{R}^d$, $d = 1, 2$, or 3, with constant diffusivity ν . The classical local model describes the diffusion by using the following PDE-based formulation:

$$\begin{cases} \frac{\partial u(\mathbf{x}, t)}{\partial t} = \nu \nabla^2 u(\mathbf{x}, t) + s(\mathbf{x}, t) & \forall (\mathbf{x}, t) \in \Omega \times [0, T] \\ G(u(\mathbf{x}, t)) = 0 & \forall (\mathbf{x}, t) \in \partial\Omega \times [0, T] \end{cases} \quad (1)$$

where $s \in \mathbb{R}$ is the source/sink term and $G(u(\mathbf{x}, t))$ defines the BCs (which could be Dirichlet, Neumann, Robin, or mixed).

The peridynamic (PD) bond-based diffusion model [10, 21, 46], on the other hand, is written as:

$$\frac{\partial u(\mathbf{x}, t)}{\partial t} = \nu \mathcal{L}_\omega u(\mathbf{x}, t) + s(\mathbf{x}, t) \quad \forall (\mathbf{x}, t) \in \Omega \times [0, T] \quad (2)$$

where $\mathcal{L}_\omega$ is the PD Laplacian operator which can be expressed as:

$$\mathcal{L}_\omega u(\mathbf{x}, t) = \int_{\mathcal{H}_x} \omega(\mathbf{x}, \mathbf{y})(u(\mathbf{y}, t) - u(\mathbf{x}, t)) d\mathbf{y}, \quad (3)$$

Here $\mathcal{H}_x$ is the (nonlocal) horizon region of $\mathbf{x}$ and is usually selected to be a disk in 2D (a line segment in 1D or a sphere in 3D) centered at $\mathbf{x}$, with the radius denoted by δ (the horizon size, or simply the “horizon”, when it is clear from the context that we refer to the radius or the region, not the region itself). Figure 1 schematically shows a 2D PD body with a generic point $\mathbf{x}$, its family, and its horizon region. Objects that carry pairwise nonlocal interactions between points are called bonds. In the more generalized state-based formulation, diffusion in a bond is directly influenced by other bonds which connect to the same point [12]. This setting may be beneficial for certain problems but is not considered here.

The kernel function $\omega(\mathbf{x}, \mathbf{y}) : \Omega \times \Omega \rightarrow \mathbb{R}$ in Eq. (3) denotes a nonnegative symmetric mapping, i.e., $\omega(\mathbf{x}, \mathbf{y}) = \omega(\mathbf{y}, \mathbf{x}) \geq 0$. It has been shown that for $u \in C^2(\Omega)$, $\mathcal{L}_\omega u \rightarrow \Delta u$ as $\delta \rightarrow 0$, under certain conditions [42, 47]. A kernel function that leads to good convergence properties with the one-point Gaussian quadrature discretization of the spatial domain is [42]:

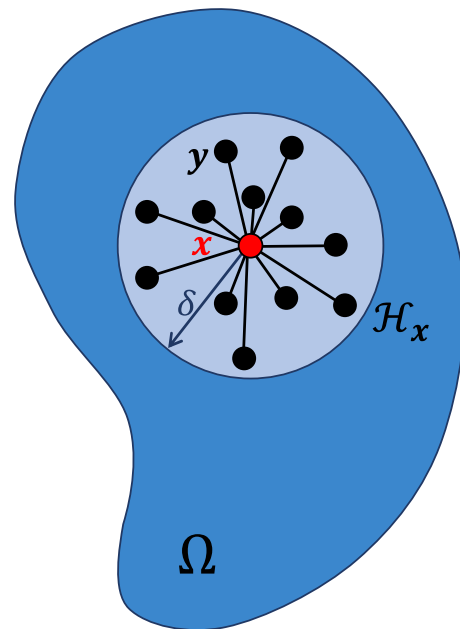


Fig. 1 A peridynamic body with a generic point $\mathbf{x}$ and its horizon region, $\mathcal{H}_x$. Nonlocal interactions exist through the bond between point $\mathbf{x}$ and an arbitrary point $\mathbf{y}$ located in its horizon

$$\omega(\mathbf{x}, \mathbf{y}) = \frac{A}{\|\mathbf{y} - \mathbf{x}\|^2} \quad (4)$$

where A is a constant which can be calibrated by matching PD flux to classical flux for a linearly distributed field, as shown in [11]. One can also determine A by enforcing that Eq. (2) recovers the classical diffusion equation as $\delta \rightarrow 0$ using the approach first provided in [25] for 1D linear elasticity, as shown below for 2D diffusion.

Suppose $u(\mathbf{x})$ is sufficiently smooth, one can write, for all $\mathbf{x} \in \Omega$, and $\mathbf{y} \in \mathcal{H}_x$:

$$\begin{aligned} u(\mathbf{y}) &= u(\mathbf{x}) + \xi_x \frac{\partial u}{\partial x}(\mathbf{x}) + \xi_y \frac{\partial u}{\partial y}(\mathbf{x}) \\ &+ \frac{1}{2!} \left(\xi_x^2 \frac{\partial^2 u}{\partial x^2}(\mathbf{x}) + 2\xi_x \xi_y \frac{\partial^2 u}{\partial x \partial y}(\mathbf{x}) + \xi_y^2 \frac{\partial^2 u}{\partial y^2}(\mathbf{x}) \right) \\ &+ \frac{1}{3!} \left(\xi_x^3 \frac{\partial^3 u}{\partial x^3}(\mathbf{x}) + 3\xi_x^2 \xi_y \frac{\partial^3 u}{\partial x^2 \partial y}(\mathbf{x}) + 3\xi_x \xi_y^2 \frac{\partial^3 u}{\partial x \partial y^2}(\mathbf{x}) \right. \\ &\left. + \xi_y^3 \frac{\partial^3 u}{\partial y^3}(\mathbf{x}) \right) + 0 \left(\left(\xi_x^2 + \xi_y^2 \right)^2 \right) \end{aligned} \quad (5)$$

where $\xi = \mathbf{y} - \mathbf{x} = [\xi_x \ \xi_y]^T$. Substitute Eq. (5) into Eq. (3) and consider symmetry, we get:

$$\begin{aligned} \mathcal{L}_\omega u(\mathbf{x}, t) &= \frac{A}{2} \int_{\mathcal{H}_x} \frac{\xi_x^2 \frac{\partial^2 u}{\partial x^2} + \xi_y^2 \frac{\partial^2 u}{\partial y^2} + O\left(\left(\xi_x^2 + \xi_y^2\right)^2\right)}{\xi_x^2 + \xi_y^2} d\tilde{\mathbf{x}} \\ &= \frac{A}{2} \left(\frac{\partial^2 u}{\partial x^2} \int_0^\delta \rho d\rho \int_0^{2\pi} \cos^2 \theta d\theta + \frac{\partial^2 u}{\partial y^2} \int_0^\delta \rho d\rho \int_0^{2\pi} \sin^2 \theta d\theta + \delta^2 O(\delta^2) \right) \\ &= \frac{\pi \delta^2 A}{4} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + O(\delta^2) \right) \end{aligned} \quad (6)$$

One recovers the 2D classical diffusion equation as $\delta \rightarrow 0$ when we have:

$$A = \frac{4}{\pi \delta^2} \quad (7)$$

Equation (7) agrees with the calibrated values given in [10, 11]. With this value, the PD model converges to the classical model of order two for $\mathbf{x}$ with a full horizon. However, using Eq. (7) for $\mathbf{x}$ which does not have a full horizon region would lead to lower diffusivity (or a lower stiffness for problems in elasticity (see [23, 25])). Moreover, BCs for PD equations (e.g., Eq. (2)) should be nonlocal (sometimes called “volume-constraints” [21, 22]), but usually, only the local BCs are available. In the next section, we discuss the FNMs which allow us to enforce local BCs in PD models.

3 Fictitious nodes methods

Unlike classical local methods, the boundary conditions in peridynamics are nonlocal. However, when solving practical problems, imposing local-type boundary conditions

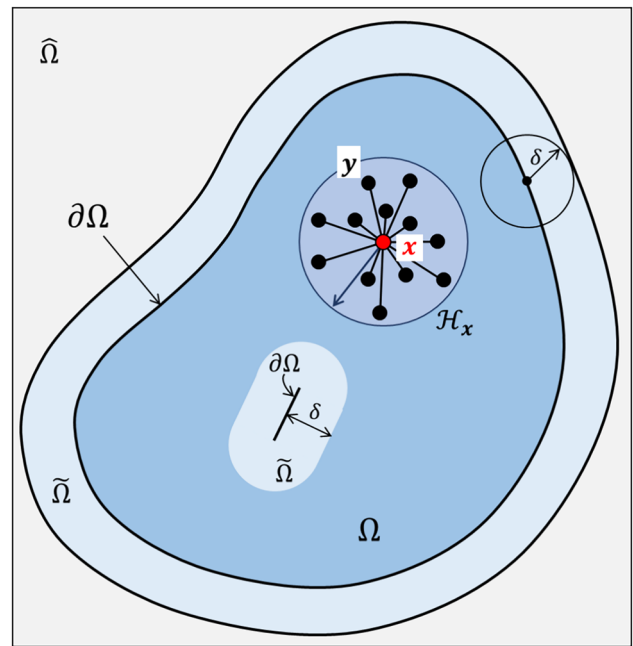


Fig. 2 Schematic of a peridynamic domain (Ω), its boundary ($\partial\Omega$, which includes the crack surface), its fictitious layer ($\tilde{\Omega}$, which includes the region around the crack surface overlayed on Ω), and the regular-shaped region $\hat{\Omega}$ within which $\tilde{\Omega}$ is determined

in nonlocal/peridynamic models is usually desired/needed because, in reality, conditions (on the unknown function values or its flux) are imposed at the surfaces of a body, not through a finite layer near the surface. The natural representation of such conditions (based on measurements) is via local boundary conditions. Various methods to impose local boundary conditions in PD models have been investigated in [12, 22, 48]. One such method is the fictitious nodes method (FNM) [12, 34, 48].

In FNM for peridynamics, certain volume constraints $c(\tilde{u}(\tilde{\mathbf{x}})) = 0$ are specified on the extended fictitious region $\tilde{\Omega} = \{\tilde{\mathbf{x}} \notin \Omega | \text{dist}(\tilde{\mathbf{x}}, \partial\Omega) < \delta\}$ (e.g., the “collar” of the domain Ω shown in Fig. 2), so that desired local boundary conditions on $\partial\Omega$ (which includes crack surfaces) are satisfied or approximately satisfied. The domain consists of material points, while the fictitious domain is the collection of fictitious points, which are not material points but may occupy the same spatial location with material points, for example when constructing the fictitious domain around a crack (see Fig. 2). If fictitious points “coincide” with material points, that only means that they share the same location, but they are independent of each other.

Such volume-constrained PD problems are defined analogously to boundary value problems for PDEs in the local theory according to nonlocal vector calculus [22].

Volume-constrained PD steady-state diffusion (or Laplace) problem can be expressed as [22]:

$$\begin{cases} \mathcal{L}_\omega u(\mathbf{x}) = 0 & \mathbf{x} \in \Omega \\ c(\tilde{u}(\tilde{\mathbf{x}})) = 0 & \tilde{\mathbf{x}} \in \tilde{\Omega} \end{cases} \quad (8)$$

When applied to free boundaries (or traction-free crack surfaces), the FNM mitigates the peridynamic surface effect [34]. Surface effect is a common issue for numerical models using nonlocal formulations [22, 38]. The PD surface effect appears because, unlike in the bulk, material points near the free boundary/surface do not have a full non-local neighborhood. The PD surface effect leads to the slightly different behavior of material points near the surface compared with those in the bulk. This could mean a lower diffusivity near the surface for diffusion problems, and a lower stiffness for problems in elasticity (see [23]). These effects are reduced as one decreases δ , and become “exact” when δ is the same as the physical nonlocal interaction range, which could be atomistically small. In practical modeling, δ is usually set to match observable physical length scales (see discussion in [49]), and not larger than relevant geometrical features of the domain (notch widths, etc.).

Before implementing FNMs, the explicit $\partial\Omega$ at the initial time is needed to determine the initial configuration of discretized Ω and $\tilde{\Omega}$ (see Appendix A). Note that Ω is considered as a closed set, i.e., if a node $\mathbf{x}$ sits on $\partial\Omega$, then $\mathbf{x} \in \Omega$. Then the boundary $\partial\Omega$ will be implicitly tracked by bonds connecting points in Ω and those in $\tilde{\Omega}$, or by broken bonds. A regular-shaped region $\hat{\Omega}$ is usually placed outside Ω in which $\tilde{\Omega}$ is determined, and the $\tilde{\Omega}$ determined after discretization is usually larger than the one before discretization to assure $\mathcal{H}_x$ is complete $\forall \mathbf{x} \in \Omega$ in the discretized configuration. In this section, we review three different types of FNM from the literature.

3.1 Naïve FNM

The “naïve”-type FNM is often used in the literature to impose local Dirichlet and homogeneous Neumann (no flux) BCs because of its simplicity. This method enforces Dirichlet BCs by assigning the same values to all fictitious points corresponding to a boundary point, while homogeneous Neumann BCs (insulated BCs in diffusion problems) are enforced by simply neglecting all corresponding fictitious points [42]. See Fig. 3 for an illustration of how a Dirichlet BC $u(\mathbf{x}_b) = u_b$ at $\mathbf{x}_b \in \partial\Omega_D$ (boundary subjected to Dirichlet BC) is enforced at $\tilde{\mathbf{x}} \in \tilde{\Omega}_D$ (fictitious region subjected to Dirichlet volume-constraint). An even simpler version of “enforcing” local Dirichlet BCs in the nonlocal model is to dispense entirely with the fictitious points and subject only the surface points to the values of local

boundary conditions [10]. Previous work has shown that results by both versions converge to local solutions as the horizon size approaches zero [13, 42].

Remark: The naïve FNM has the advantage of featuring the simplest implementation and is the most efficient (see Sect. 4.1). However, a jump-discontinuity in the solution at the boundary may be generated, leading to possible errors in fluxes near the boundary (see Sect. 4.1).

3.2 Taylor-based FNM

The second FNM, used in the PD context first in [32], requires a Taylor expansion (to linear terms) for $\mathbf{x} \in \tilde{\Omega} \cup \Omega_\delta$, in which $\Omega_\delta = \{\mathbf{x} \in \Omega | \text{dist}(\mathbf{x}, \partial\Omega) < \delta\}$. We call it Taylor-based FNM or simply Taylor FNM. To impose the local Dirichlet BC $u(\mathbf{x}) = u_D(\mathbf{x})$ for $\mathbf{x} \in \partial\Omega_D$ using Taylor FNM, for each $\mathbf{x} \in \Omega_\delta$, $\mathbf{y} \in \tilde{\Omega}_D \cap \mathcal{H}_x$ and $\mathbf{x}_b = \partial\Omega_D \cap \overline{\mathbf{x}\mathbf{y}}$, $\tilde{u}(\mathbf{y})$ is extrapolated from $u(\mathbf{x})$ as:

$$\tilde{u} = u + \left(\frac{\partial u}{\partial \xi} \right) \xi + \mathcal{O}(\delta^2) \approx u + \frac{(u_b - u)}{d} (d + \tilde{d}) = u_b + \frac{\tilde{d}}{d} (u_b - u) \quad (9)$$

where $\tilde{u} = \tilde{u}(\mathbf{y})$, $u = u(\mathbf{x})$, $u_b = u_D(\mathbf{x}_b)$, $\xi = \mathbf{y} - \mathbf{x}$, $d = \text{dist}(\mathbf{x}, \mathbf{x}_b)$ and $\tilde{d} = \text{dist}(\mathbf{y}, \mathbf{x}_b)$ (“dist” denotes distance).

Note that when $\mathbf{x}$ approaches the boundary in Eq. (9), a significantly large $\tilde{d}/d$ may occur. This situation could cause numerical errors during the simulation. Therefore, a further modification on Eq. (9) may be required as follows [43] to avoid the above issue:

$$\tilde{u} = u + (u_b - u) \times \min \left\{ \lambda, \left(1 + \frac{\tilde{d}}{d} \right) \right\} \quad (10)$$

where λ is a parameter determined from numerical tests. Normally $\lambda = 1.5$ leads to good results [43].

To impose a local Neumann or Robin BC $\nabla u_b \cdot \mathbf{n} = f(u_b)$ for $\mathbf{x}_b \in \partial\Omega_{NR}$ (boundary subjected to Neumann or Robin BC), where f is a given function, for each $\mathbf{x} \in \Omega_\delta$, $\mathbf{y} \in \tilde{\Omega}_{NR} \cap \mathcal{H}_x$ and $\mathbf{x}_b = \partial\Omega_{NR} \cap \overline{\mathbf{x}\mathbf{y}}$, $\tilde{u}$ can be approximated by u and u_b by the following Taylor expansion [32]:

$$\tilde{u} = u + \xi \left(\frac{\partial u}{\partial \xi} \right) + \mathcal{O}(\delta^2) \approx u + \xi \left(\frac{\partial u_b}{\partial \mathbf{n}} \right) = u + (d + \tilde{d}) f(u_b) \quad (11)$$

in which u_b is not given and needs to be approximated from the following equation:

$$u_b \approx u + df(u_b) \quad (12)$$

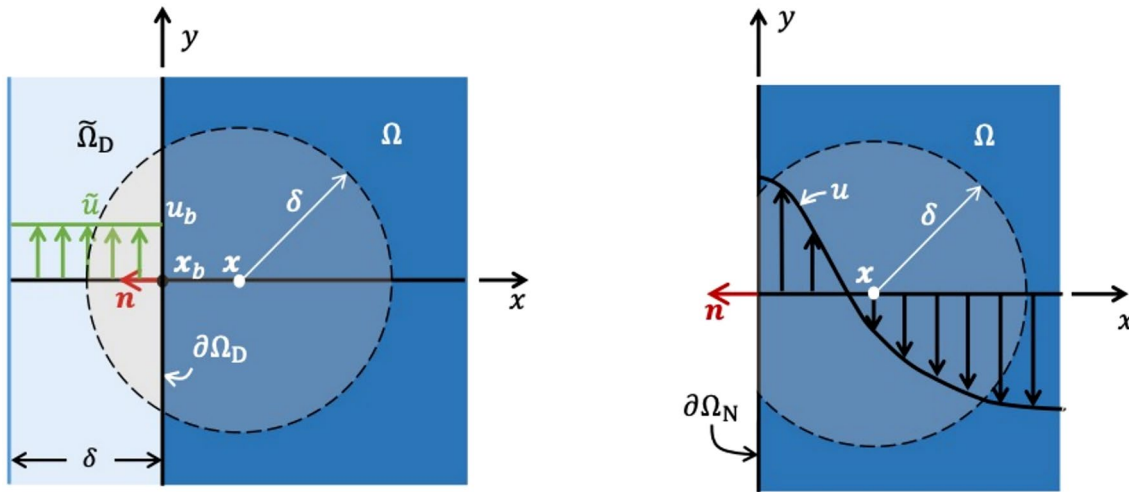


Fig. 3 Illustration of using the naïve FNM to enforce the local Dirichlet (a) and homogeneous Neumann (b) BCs

If $f(u_b)$ is a nonlinear function, a nonlinear equation solver, such as Newton's method, is needed to solve for u_b [50].

Remark: In this Taylor approach, $\tilde{u}$ at each $y \in \tilde{\Omega}$ changes, even in the same solution step, and needs to be computed anew for each $x \in \Omega_\delta \cap \mathcal{H}_y$ at which an integration over $\mathcal{H}_x$ is performed (see Eqs. (9) & (12)). This is illustrated in Fig. 4, where for $x_i \in \Omega_\delta$ and $x_b \in \partial\Omega_D \cap \mathcal{H}_{x_i}$, $\tilde{u}_i$ is the distribution of $\tilde{u}(y) \forall y \in \{y \in \tilde{\Omega}_D | \partial\Omega_D \cap \overline{x_i y} = x_b\}$. The solution step refers to each call to the Conjugate Gradient (CG) solver (see Fig. 24 in Appendix A for the flowchart of the simulation). Moreover, for boundaries with irregular geometries such as corners, those x and y nearby also have variable d and $\tilde{d}$ associated with them, because, for each pair of x and y , $\overline{x y}$ may intersect different parts of the boundary.

3.3 Mirror-based FNM

As shown in Fig. 5, the mirror FNM assigns the constraint $\tilde{u}(\tilde{x})$ at each $\tilde{x} \in \tilde{\Omega}$ based on $u(x^R)$ and $u(x^P)$ in which $x^P = \text{OProj}_{\partial\Omega}(\tilde{x})$ is the orthogonal projection of $\tilde{x}$ onto $\partial\Omega$ and $x^R = \text{Ref}_{\partial\Omega}(\tilde{x}) = \tilde{x} + 2(x^P - \tilde{x})$ is the reflection, or mirror point, of $\tilde{x}$ through/across $\partial\Omega$. For $\tilde{x} \in \tilde{\Omega}$, when $\partial\Omega_{\tilde{x}} = \{\partial\Omega \cap \mathcal{H}_{\tilde{x}}\}$ is continuous and the normal to $\partial\Omega_{\tilde{x}}$ at each point is unique, we have $\tilde{x}x^P = -kn(x^P)$, in which $k \in \mathbb{R}^+$ and $n(x^P)$ is the outward unit normal vector at x^P .

In the mirror FNM, to impose the local Dirichlet BC $u(x) = u_D(x)$ for $x \in \partial\Omega_D$ and the Neumann BC $\nabla_n u(x) = q(x)$ for $x \in \partial\Omega_N$, $\tilde{u}(\tilde{x})$ at $\tilde{x} \in \tilde{\Omega}_D$ is assigned as:

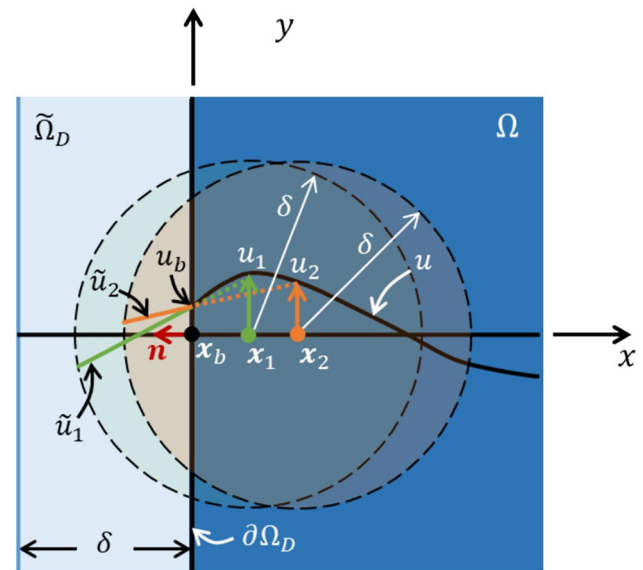


Fig. 4 Illustration of enforcing the local Dirichlet BC by using the Taylor FNM (redrawn from [38])

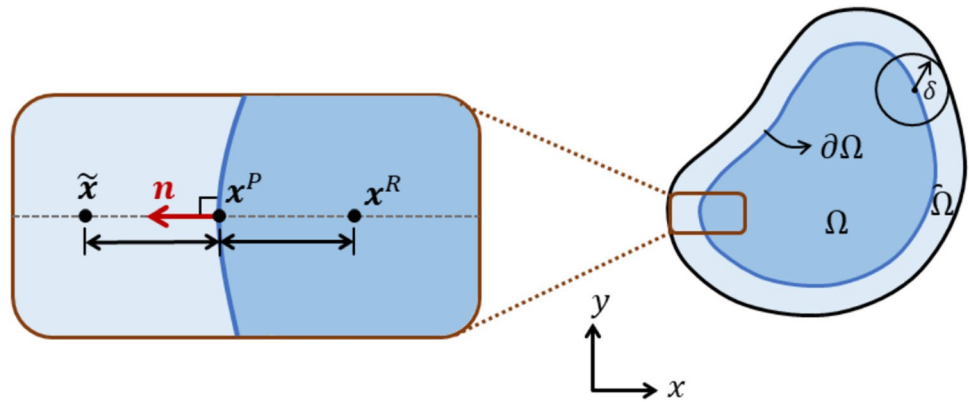
$$\tilde{u}(\tilde{x}) = 2u(x^P) - u(x^R) \quad (13)$$

and $\tilde{u}(\tilde{x})$ at $\tilde{x} \in \tilde{\Omega}_N$ is assigned as:

$$\tilde{u}(\tilde{x}) = u(x^R) + \|\tilde{x} - x^R\|q(x^P) \quad (14)$$

respectively. For the local Robin BC $\nabla_n u(x) = f(u(x))$ for $x \in \partial\Omega_R$, we have, for $\tilde{x} \in \tilde{\Omega}_R$:

Fig. 5 A schematic of mirror points in mirror-based FNM [12]



$$\begin{aligned}\tilde{u}(\tilde{x}) &= u(x^R) + \|\tilde{x} - x^R\| f(u(x^P)) \\ &= u(x^R) + \|\tilde{x} - x^R\| f\left(\frac{u(x^R) + \tilde{u}(\tilde{x})}{2}\right)\end{aligned}\quad (15)$$

in which the approximation $u(x^P) = \frac{u(x^R) + \tilde{u}(\tilde{x})}{2}$ is made because u should be close to linear between $\tilde{x}$ and x^R . Note that $\tilde{u}(\tilde{x})$ requires to be solved using a nonlinear solver if function f is nonlinear.

While the Eq. (9) of Taylor-based FNM with $d = \tilde{d}$ appears to be similar to Eq. (13) of mirror-based FNM, they are different, except when nodes x and y (based on which quantities in Eq. (9) are computed) coincide with nodes x^R and $\tilde{x}$ in Eq. (13), respectively.

Remark: In the mirror FNM, $\tilde{u}(\tilde{x})$ at each $\tilde{x} \in \tilde{\Omega}$ does not change in the same solution step because it only depends on x^R and x^P which can be uniquely determined for each x . See Fig. 6 for illustrations of how local Dirichlet BCs are enforced in the mirror FNM.

Remark: Note that in the Taylor and mirror FNMs described in this section, all the information for $x \in \Omega_\delta \cup \tilde{\Omega}$, such as $\text{dist}(x, \partial\Omega)$ or $\text{Ref}_{\partial\Omega}(x)$, are considered as given, which works for simple geometries. However, for general cases, such information is not straightforward and cannot be provided as input. For example, for shapes with corners of various angles, cusps, crack tips, etc., the tangent line (and normal vector) is not well defined everywhere. While some ad-hoc choices can be made for overcoming this issue, we aim for a general strategy that assures all the information required to implement the FNM can be determined without ambiguity. Moreover, the enforcement of local BCs on surfaces of cracks has not been considered until now, mainly because fictitious regions are not clearly defined for such “inner” surfaces yet. In the next section, we introduce an algorithm for the mirror FNM that automatically enforces local BCs, including those imposed on crack surfaces, in PD models for problems with arbitrary boundary shapes.

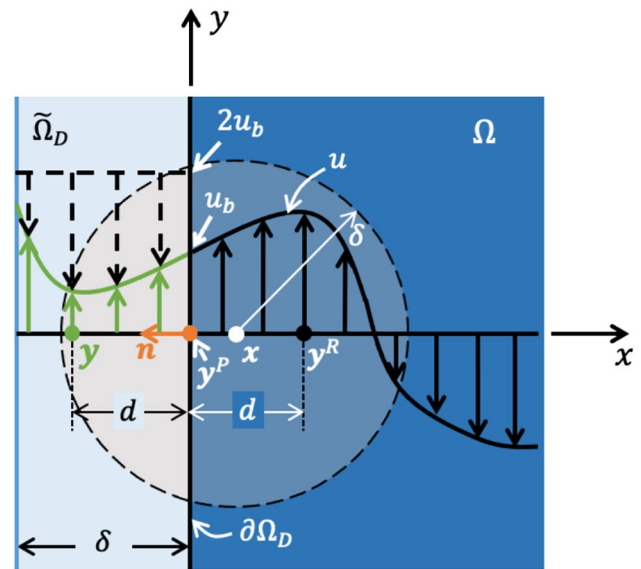


Fig. 6 Illustration of enforcing a local Dirichlet BC in the mirror FNM (redrawn from [12])

3.4 An algorithm to find mirror nodes in mirror-based FNM

Using the mirror FNM for arbitrarily shaped boundaries/surfaces (with corners, cusps, etc.) requires finding both x^P and $x^R \forall \tilde{x} \in \tilde{\Omega}$. Since $x^R = \tilde{x} + 2(x^P - \tilde{x})$, as shown in Fig. 5, once x^P is known, x^R can be determined. Starting from $\tilde{x} \in \tilde{\Omega}$, one could determine x^P as $\arg\min_{y \in \mathcal{H}_{\tilde{x}} \cap \partial\Omega} \text{dist}(y, \tilde{x})$. However, the uniqueness of the solution is not guaranteed. Moreover, for $\tilde{x}$ near corners with an angle smaller than 90° (see regions $\tilde{\Omega}_1$ in Fig. 7), it may lead to $x^R \notin \Omega$.

We introduce an algorithm that resolves all these issues. First, following the idea in [36], we define the “generalized” normal vector (or the nonlocal normal) to the boundary at some $x \in \Omega$ as:

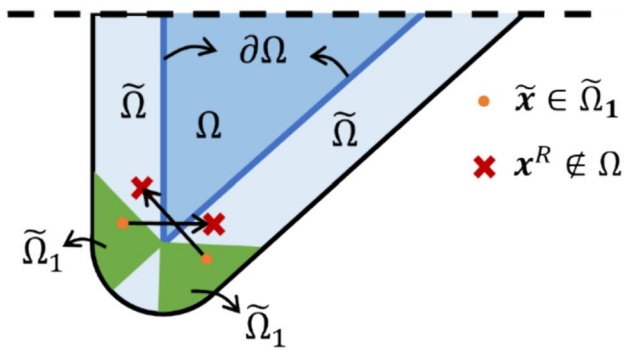


Fig. 7 Schematic of $\tilde{\Omega}_1 \subset \tilde{\Omega}$ near an acute corner where $x^R = \text{Ref}_{\partial\Omega}(\tilde{x}) \notin \Omega$ for $\tilde{x} \in \tilde{\Omega}_1$, when $x^P = \arg\min_{y \in \mathcal{H}_{\tilde{x}} \cap \partial\Omega} \text{dist}(y, \tilde{x})$

$$\hat{n}(x) = \frac{\int_{\mathcal{H}_x} \mu(x, y)(x - y) dy}{\|\int_{\mathcal{H}_x} \mu(x, y)(x - y) dy\|} \quad (16)$$

where μ is the binary function defined as $\mu(x, y) : \Omega \times \Omega \rightarrow \{0, 1\}$. μ takes the value 0 if the bond (x, y) is broken or intersects the boundary (which includes crack surfaces), and 1 otherwise. As we can see in Fig. 8a for a few examples, $\hat{n}$ is nonzero only in the region Ω_δ near the boundary of Ω , i.e., $\|\hat{n}\| = 0$ for $x \notin \Omega_\delta$. $\hat{n}$ is a generalization of the corresponding classical external normal vector $n(x)$ for $x \in \partial\Omega$ if n exists, as shown at the point x_1 in Fig. 8a, where we have $\hat{n}(x_1) \rightarrow n(x_1)$ as $\delta \rightarrow 0$.

The generalized normal vector can be extended to points in the fictitious domain by modifying Eq. (16) to be:

$$\tilde{n}(\tilde{x}) = \frac{\int_{\tilde{\mathcal{H}}_{\tilde{x}}} (1 - \tilde{\mu}(\tilde{x}, y))(y - \tilde{x}) dy}{\|\int_{\tilde{\mathcal{H}}_{\tilde{x}}} (1 - \tilde{\mu}(\tilde{x}, y))(y - \tilde{x}) dy\|} \text{ for } \tilde{x} \in \tilde{\Omega} \quad (17)$$

where $\tilde{\mu}$ is now defined as $\tilde{\mu}(\tilde{x}, y) : \tilde{\Omega} \times \Omega \rightarrow \{0, 1\}$. We also let $\tilde{\mathcal{H}}_{\tilde{x}} = \mathcal{H}_{\tilde{x}} \cap \Omega$, meaning that fictitious nodes are excluded from $\tilde{\mathcal{H}}_{\tilde{x}}$ and there are no bonds between fictitious nodes. Different from the external nonlocal normal, the nonlocal normal in Eq. (17) points towards the inside of the domain Ω (because the mirror nodes are to be found inside Ω). Moreover, in order to find fictitious nodes for regions around cracks, we want to only include bonds that intersect the crack. That is the reason for using the $\tilde{\mu}$ function which is zero if the bond $(\tilde{x}, y)$ intersects the crack. Equation (17) can then be used to find mirror nodes in the solution domain Ω . Since $\tilde{n}(\tilde{x})$ is unique for each $\tilde{x}$, it allows us to locate x^P and x^R uniquely. As we can see in Fig. 8b, this works for both regular/smooth boundaries and boundaries with corners, cracks, or cusps.

Note that for $\tilde{\Omega}$ near sharp (relative to the horizon size) convex corners, Eq. (17) may still lead to $x^R \notin \Omega$ for $\tilde{x} \in \tilde{\Omega}_2$ as shown in Fig. 9, although $\tilde{\Omega}_2$ covers a much smaller area than that of $\tilde{\Omega}_1$ in Fig. 7. This issue can be reduced by using smaller horizon sizes, but not fully resolved. In practice, if $x^R \notin \Omega$ for $\tilde{x} \in \tilde{\Omega}_2$, we let $x^R = \arg\min_{y \in \Omega} \text{dist}(y, x^R)$. Other options include approximating $\tilde{u}(\tilde{x})$ by $u(y)$ at $\arg\min_{y \in \tilde{\Omega}} \text{dist}(y, \tilde{x})$, or using the Naïve FNM, if possible. Since $\tilde{\Omega}_2$ only covers a very small fraction of $\tilde{\Omega}$ around the corner, the error introduced by these approximations should be negligible.

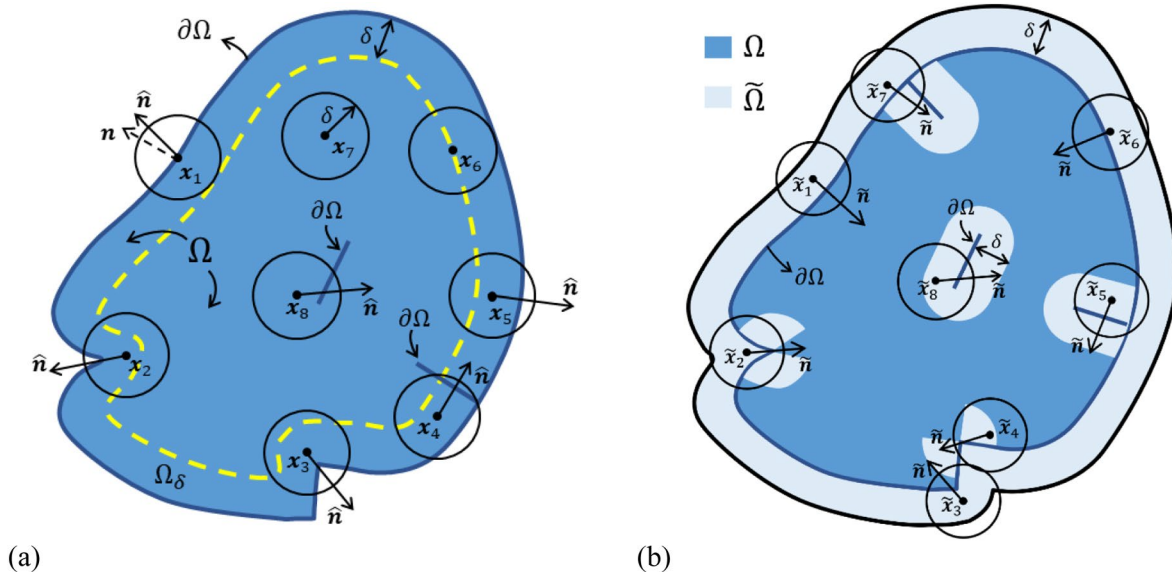


Fig. 8 Schematic for the “generalized” (or nonlocal) normal vector when (a) $x \in \Omega$ and (b) $x \in \tilde{\Omega}$

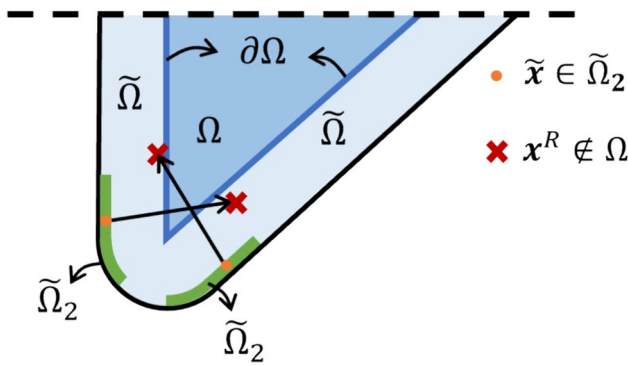


Fig. 9 Schematic of $\tilde{\Omega}_2 \subset \tilde{\Omega}$ near an acute corner for which $x^R = \text{Ref}_{\partial\Omega}(\tilde{x}) \notin \Omega$ for $\tilde{x} \in \tilde{\Omega}_2$, when Eq. (17) is used to determine x^R

Before implementing the algorithm, the explicit $\partial\Omega$ (including crack surfaces) at the initial time is needed to determine the initial configuration of discretized Ω and $\tilde{\Omega}$ (see Appendix A). Note that Ω is a closed set. Then the boundary $\partial\Omega$ will be implicitly tracked by bonds connecting points in Ω and those in $\tilde{\Omega}$ (or by broken bonds). A regular-shaped region $\tilde{\Omega}$ is usually placed outside Ω in which $\tilde{\Omega}$ is determined, and the effective $\tilde{\Omega}$ determined after discretization is usually larger than the one before discretization to ensure $\mathcal{H}_x$ is complete for all $x \in \Omega$ in the discretized configuration. Updating x^P and x^R is required every time the domain Ω evolves (which causes $\tilde{\Omega}$ to evolve as well) in problems such as corrosion or fracture, but only for those $\tilde{x}$ near the evolving portion of the domain.

The algorithm to find mirror nodes for each $\tilde{x} \in \tilde{\Omega}$ is as follows:

- (1) Compute $\tilde{n}(\tilde{x})$ using Eq. (17).
- (2) Search for x^P :
 - a. if $\tilde{x} \in \tilde{\Omega}$, then search in $\tilde{\mathcal{H}}_{\tilde{x}}$ in the direction of $\tilde{n}(\tilde{x})$ for the node closest to $\tilde{x}$, i.e., $x^P = \text{argmindist}(y, \tilde{x})$, subject to $\tilde{\mu}(\tilde{x}, y) = 0$ and $\text{dist}(\tilde{n}(\tilde{x}), y) < \Delta x/2$.
 $\text{dist}(\tilde{n}(\tilde{x}), y) = \sqrt{(y - \tilde{x}) \cdot (y - \tilde{x}) - ((y - \tilde{x}) \cdot \tilde{n}(\tilde{x}))^2}$ is the perpendicular distance between node y and $\tilde{n}(\tilde{x})$;
 - b. if x^P is not found, set $x^P = \text{argmindist}(y, \tilde{x})$, subject to $\text{dist}(\tilde{n}(\tilde{x}), y) < \Delta x/2$;
 - c. if two or more x^P are found, selecting either one of them is acceptable.
- (3) Search for x^R :
 - a. compute $x' = \tilde{x} + (2\|\tilde{x}^P - \tilde{x}\| - \Delta x)\tilde{n}(\tilde{x})$;

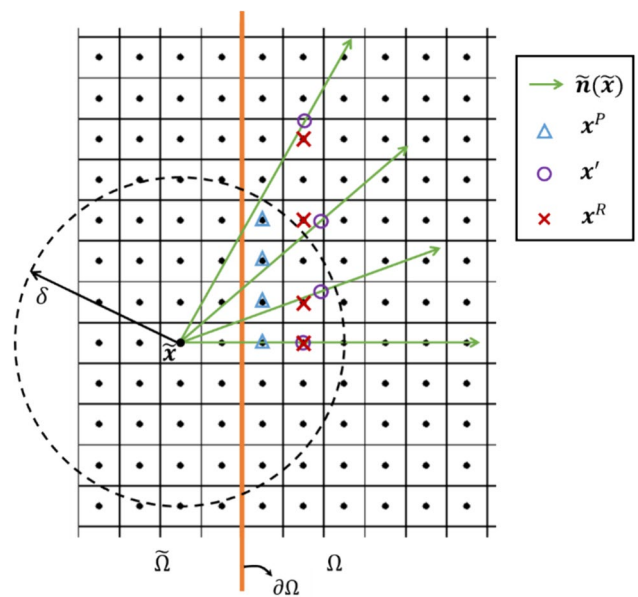


Fig. 10 Schematic diagram of determining x^P and x^R for a generic $\tilde{x} \in \tilde{\Omega}$ given four different $\tilde{n}(\tilde{x})$

- b. since x' may not coincide with a node's coordinates in the domain, we have $x^R = \text{argmindist}(y, x')$,
 $y \in \mathcal{H}_x \cap \Omega$.

Some examples for this search process are shown in Fig. 10.

Remark: By using this algorithm, regardless of the smoothness of $\partial\Omega$, x^P , and x^R will converge, for a certain choice of δ , as the ratio $\delta/\Delta x$ approaches infinity. If the tangent to $\partial\Omega$ is continuous, the converged value will be the analytical one. Until this algorithm, one had to make assumptions (which can vary from one paper to another) about $\tilde{x}x^P$ at sharp corners when using the mirror FNM [12, 34], and the traction-free crack surfaces were usually treated with the naïve FNM. Now, by using this algorithm, x^P and x^R are found automatically and consistently for any $\tilde{x} \in \tilde{\Omega}$. For problems set in domains with complex shapes, the algorithm leads to important time-savings compared to manually determining x^P and x^R , for all $\tilde{x} \in \tilde{\Omega}$. Since the number of fictitious nodes usually only accounts for a small portion of the total number of nodes, the computational cost to locate x^P and $x^R \forall \tilde{x} \in \tilde{\Omega}$ is negligible compared with the cost of a complete simulation.

Some examples of x^R found with this algorithm are shown in Fig. 11, with arrows starting from $\tilde{x} \in \tilde{\Omega}$ and ending at $x^R \in \Omega$. Note that $\tilde{x}x^R \approx k\tilde{n}(\tilde{x})$ (with $k \in \mathbb{R}^+$) and there is a transition of direction for $\tilde{x}x^R$ between that of the normal

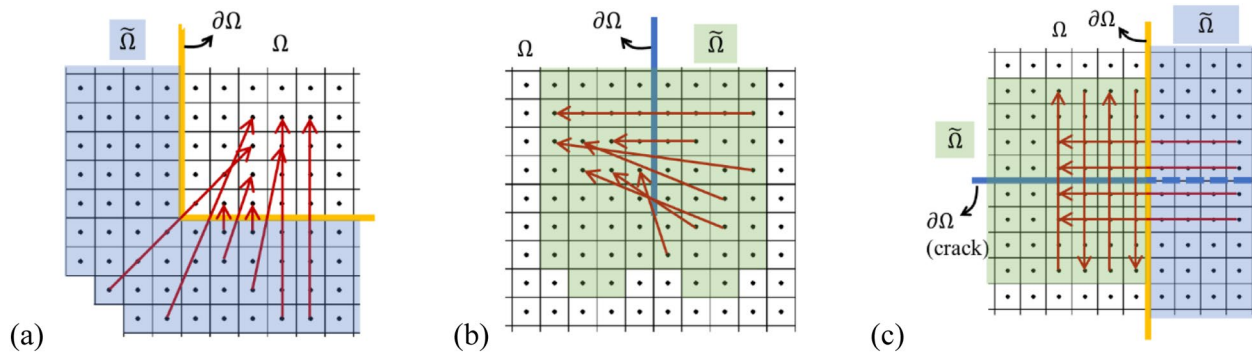


Fig. 11 Examples of some $\tilde{x} \in \tilde{\Omega}$ and their corresponding $x^R \in \Omega$, connected by arrows, computed by the new algorithm: (a) at the cor-

ner; (b) at the inner crack tip; (c) at the edge crack tip. Note that the arrow direction for each $\tilde{x}$ is just an approximation of $\tilde{n}(\tilde{x})$ computed by Eq. (16), due to the discretization

vectors on two edges of the corner (or inner crack tip). The transition zone will narrow down as the horizon δ shrinks (δ -convergence) and the transition will be smoother as $m = \delta/\Delta x$ increases (m -convergence). It may be possible that this transition of the $\tilde{x}x^R$ is used to detect/track crack tips, on the fly, but this idea is not pursued here. Although only the 2D implementation using a uniform grid is considered in this work, extending this algorithm to non-uniform grids and the 3D case is straightforward.

Remark: In this paper, we only treat problems on a fixed domain. For problems in which the crack surfaces open due to deformation, one can use the undeformed configuration to implement the algorithm. For extensions to problems with evolving domains, such as when a crack grows, one would need to update the set of fictitious nodes and create new ones for the newly created surface. Whether the crack stays closed, or it opens, should not affect this procedure. The current algorithm does not cover the growing crack case, but we are planning in the future to address this. One way to easily extend our algorithm to cover such cases could be “detecting” nodes on the growing crack surface based on the information of nodal damage and/or nodal velocity vectors, allowing for the use of existing fictitious nodes and on-the-fly introduction of new mirror nodes.

In the following section, the performance of the mirror FNM will be compared with the naïve and the Taylor FNMs. We will also show the capability of the autonomous mirror FNM using problems with curved geometry and cracks.

4 Results and discussion

In this section, we first show how the previously discussed three different FNMs perform in enforcing local BCs through the steady-state diffusion in a square domain. Then we will show how these FNMs perform in a problem with a singular point on its boundary. Finally, we are going to show the capability of the autonomous mirror FNM in problems with curved geometry and cracks.

4.1 The performance of three different FNMs in enforcing local BCs in peridynamics for diffusion problems without singularities

In the first problem, we consider diffusion in a square domain with its side equal to 0.1, subject to the following local boundary conditions:

$$\begin{cases} u(x = -0.05, y) = 0.1 \\ u(x = 0.05, y) = 0.6 \\ \nabla_n u(x, y = \pm 0.05) = 0 \end{cases} \quad (18)$$

as shown in Fig. 12. The classical solution to this problem is a linear function:

$$u(x, y) = 5x + 0.35 \quad (19)$$

In the PD simulations for this problem, we first take $\delta = 0.004$ and $\Delta x = 0.001$. This leads to about 10,000 nodes for a uniform discretization.

The solution, obtained with the different FNMs, along the dashed mid-line in Fig. 12 is shown in Fig. 13. All three results match the classical solution very well. However, if we zoom in near the boundary, we can see clearly that the result obtained with the mirror FNM matches the classical analytical solution much better than the results obtained with the other two FNMs. The simulation times for the three

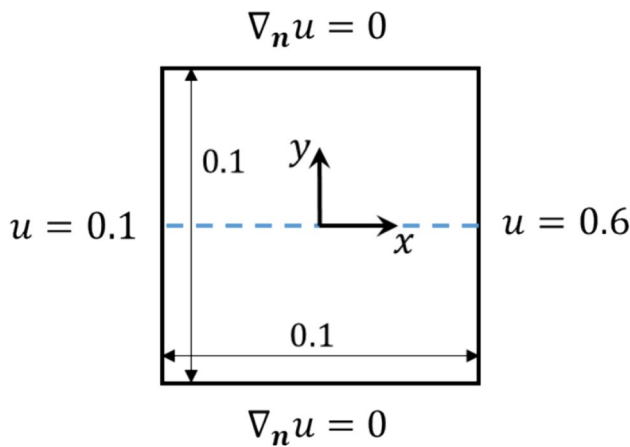


Fig. 12 The geometry and boundary conditions for diffusion in a square domain used to compare different FNMs' capabilities in enforcing local BCs in the PD formulation

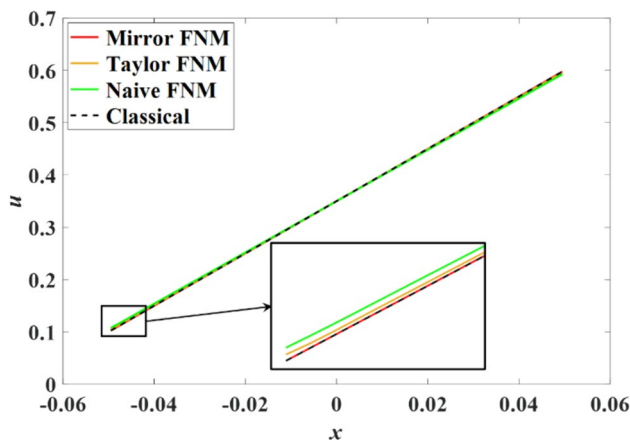


Fig. 13 The classical analytical solution and PD solutions with different FNMs for the steady-state diffusion problem seen in Fig. 12, along the dashed line shown there

FNMs are given in Table 1. The simulation with the naïve FNM is at least 10 times faster than the other two types of FNMs, while the efficiency of Taylor and the mirror FNMs are similar to one another. Therefore, when the accuracy of the solution near the boundary is not critical, the naïve FNM could be the “best” option. If accuracy is needed, then the mirror FNM works better. Certainly, one can use finer grids with the naïve FNM to increase the accuracy near the boundary, but to reach the same accuracy as the mirror FNM, a much finer grid will be required. Considering that the computational cost of PD simulations increases quadratically with grid size (except for the recently introduced convolution-based approximations, see, [35, 51]), this may not be a good option.

Table 1 Computational time of the simulation using the three different FNMs, when $\delta = 0.004$ and $\Delta x = 0.001$

	FNM		
	Naïve	Taylor-based	Mirror-based
Time (s)	2.0	20.1	20.6

Although the authors in [32] were able to obtain good results using Taylor FNM without Eq. (10), we noticed that the convergence of results is not guaranteed for all the simulations in this work unless we implemented Eq. (10). As pointed out in [43], the discrete arrangement of nodes may permit a node to closely approach the boundary. This could potentially lead to significantly large $\tilde{d}/d$ in Eq. (9), which may cause numerical issues. Therefore, we recommend consistently employing Eq. (10) to ensure convergence when utilizing Taylor FNM.

The next example has a singular point along a straight boundary, and we will use it to further test the performance of these different FNMs.

4.2 The performance of the three FNMs for the Motz problem

To test the capability of the different FNMs in handling local singularities (singularities present in the classical/local model of diffusion) along the boundary, we choose the Motz problem [44, 45] (see Fig. 14 for the domain geometry). The classical solution (see below) of the Motz problem has a strong singularity $O(\rho^{-1/2})$ in terms of fluxes at the origin O . Here ρ is the polar distance to the origin.

The Motz problem is a classical steady-state diffusion problem with the following boundary conditions:

$$\begin{cases} \nabla_n u(x = -1, y) = 0 \\ \nabla_n u(x, y = 1) = 0 \\ \nabla_n u(x > 0, y = 0) = 0 \\ u(x < 0, y = 0) = 0 \\ u(x = 1, y) = 500 \end{cases} \quad (20)$$

and its classical solution can be written as [45]:

$$u(\rho, \theta) = \sum_{i=0}^{\infty} D_i \rho^{i+1/2} \cos\left[\left(i + \frac{1}{2}\right)\theta\right] \quad (21)$$

where D_i 's are analogous to the stress intensity factors in linear elastic fracture mechanics, sometimes called “generalized flux intensity factors” [52]. Here we choose the first 34 terms in the series to plot our results, as was done in [53].

In Fig. 15a we show the contours for the classical solution. The point-wise relative differences between the PD solutions and the classical solution are given in Fig. 15b–d.

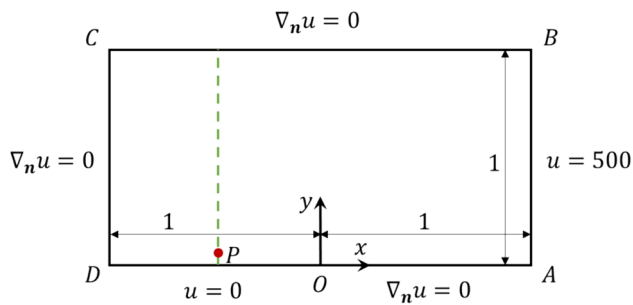


Fig. 14 Domain and boundary conditions for Motz problem

For the PD simulations, we used $\delta = 0.04$ and $\Delta x = 0.01$. Notice that for the mirror FNM, the relative difference larger than 5% is restricted to the horizon region of the singular point (point O) and the left-bottom corner (point D). As the horizon size approaches zero (with $\delta/\Delta x$ not decreasing), these areas also converge to zero. For the other two types of FNMs, the relative difference is large not only near the point of singularity in fluxes, or near the corner but also at locations far away from them. A quantitative comparison of the solutions along the vertical dash line shown in Fig. 14 between the three types of FNMs is shown in Fig. 16. As we can see in the zoomed-in image, the PD solution obtained with the mirror FNM matches the classical solution significantly better than the PD solutions obtained with the other two FNMs.

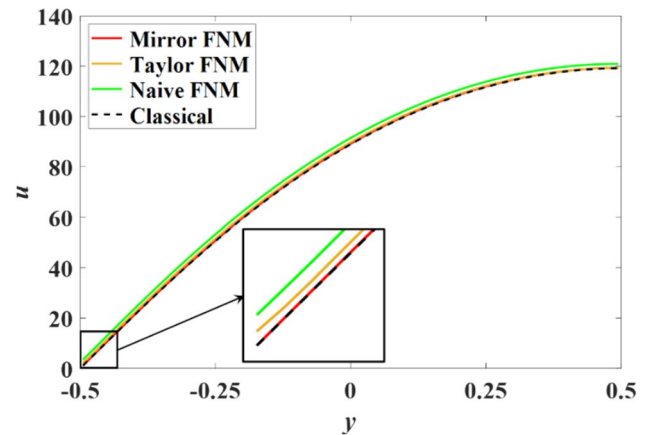


Fig. 16 The classical solution and PD solutions with different FNMs for the Motz problem along the vertical dashed line at $x = -L/2 + \Delta x/2$ shown in Fig. 14

The δ -convergence [54] (with $\delta/\Delta x$ fixed to be 4) to the classical solution for the three PD solutions at point P ($-0.5, -0.48$) (see Fig. 14), is provided in Fig. 17 in log scale. We do not choose a point on the boundary because the analytical solution there is zero (i.e., the relative difference does not exist). Notice that as the horizon size changes, the grid also changes and there may not exist a node at point P . In such cases, we simply obtain the value at point P by averaging the values at the four nearest nodes around P . Figure 17 demonstrates that, as the horizon size decreases ($1/\delta$ increases), solutions obtained with all three types of FNM approach

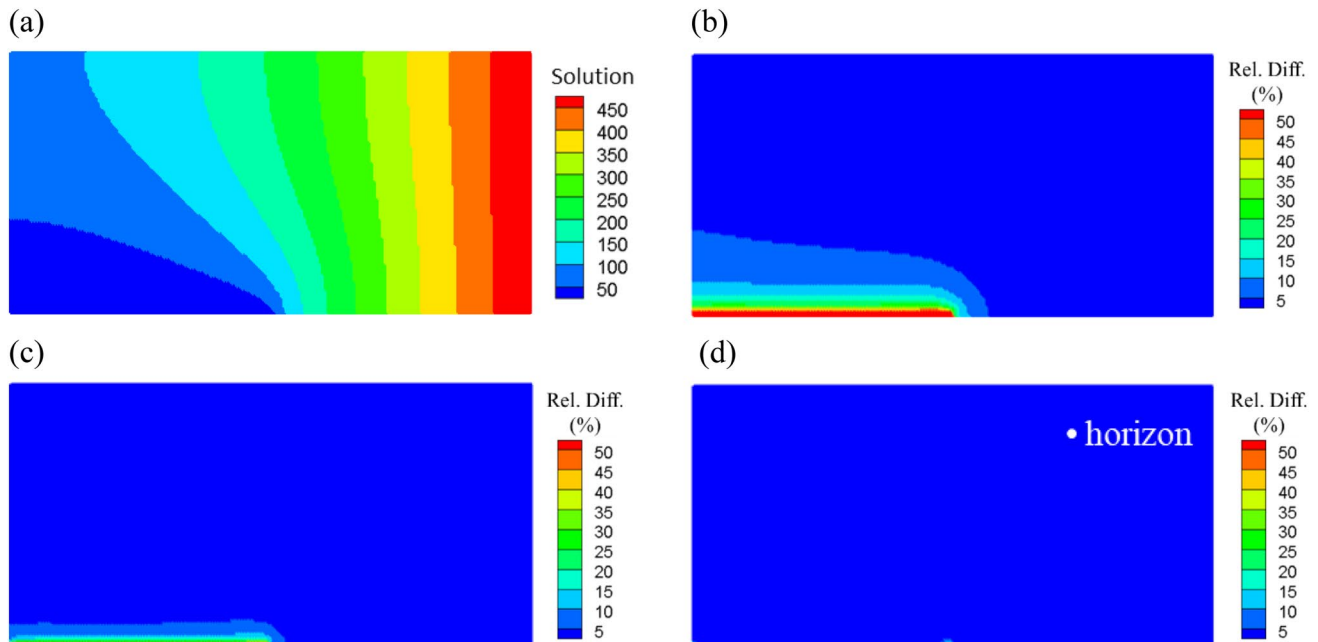


Fig. 15 (a) Contours for the classical solution of the Motz problem; Relative difference (Rel. Diff.) to classical solution of the Motz problem using the PD model with the: (b) naïve, (c) Taylor, and (d) mirror FNMs

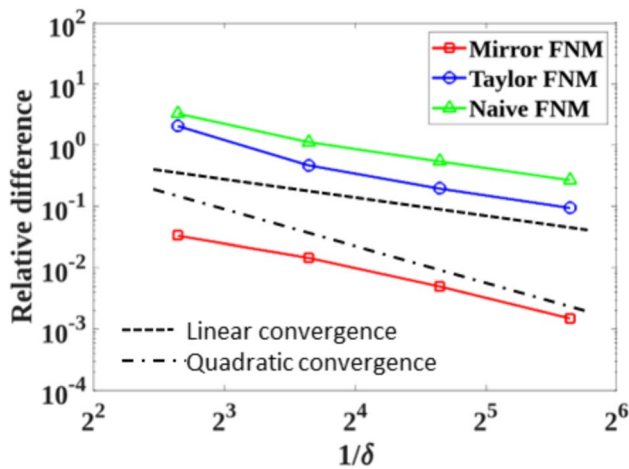


Fig. 17 δ -convergence of PD results (using the different FNMs) to the classical value at point P $(-0.5, -0.48)$ in Fig. 14

the classical solution. However, the mirror FNM produces relative differences from the classical solution that are *two orders of magnitude* smaller than those from the other two FNMs. Moreover, the mirror FNM solution exhibits a convergence rate that is increasing faster than the other two, as the horizon size decreases.

The above two examples show that the mirror FNM works best at accurately enforcing local boundary conditions in PD models, especially for problems with flux singularities (in local models) along the boundary. In the following section, we will test the algorithm developed for the mirror FNM for problems with more complex geometries.

4.3 Steady-state diffusion in circular disks with cracks

In this subsection, we will apply the autonomous mirror FNM to solve the PD formulation for diffusion in circular disks with cracks on which Dirichlet boundary conditions are to be enforced. Using the algorithm developed in Sect. 3.4, the mirror FNM can be easily implemented for curved boundaries and cracks.

4.3.1 Disk with a single crack

We first consider a circular disk of radius 1 with a 0.5-long vertical crack for which the middle point coincides with the center of the disk which serves as the origin of the coordinate system. The geometry and boundary conditions imposed are shown in Fig. 18. For the simulations, a uniform grid with 40,000 nodes is used in the PD meshfree discretization. The horizon size and m-value are 0.04 and 4, respectively. The mirror node for each fictitious node, required by the mirror FNM, is determined by the algorithm

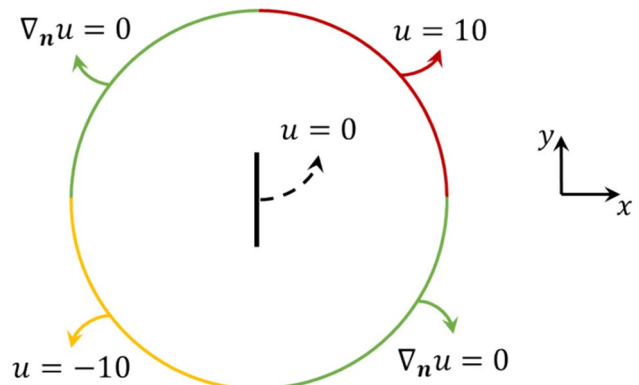


Fig. 18 Local boundary conditions imposed for diffusion in a disk with a pre-crack at the center

described in Sect. 3.4. See Fig. 11 for how mirror-nodes are found for those nodes near the crack tip.

For verification, we also simulate the corresponding PDE-based problem using the steady-state thermal solver in ANSYS Workbench. Details of the ANSYS simulation, including how the “crack” is simulated, can be found in Appendix B. In the PD model, the crack is inserted by cutting all bonds that intersect with the pre-crack segment. The differences between the slightly different approaches in representing the crack in the two models should only have a negligible effect on the results.

Contours of the results obtained by ANSYS and by the PD model, with zoomed-ins around the crack region, are given in Fig. 19. The PD results match the FEM results closely, which shows that the autonomous mirror FNM works very well for problems with a curved boundary and with cracks. The jagged shape in the contours of the PD results appears because no smoothing interpolation technique is used in the visualization, we simply plot the obtained values at each PD node.

In practical simulations, the crack surface may have an angle with respect to the (uniform) discretization grid, whether it is a pre-crack or a new crack formed during the simulation. To demonstrate the generality of our algorithm for the mirror FNM, we solve the same problem but we rotate the uniform grid (using the rotation matrix to transform the coordinates of all nodes) in the counterclockwise direction by 30° and 45° relative to the coordinates shown in Fig. 18, respectively, as shown in Fig. 20. Notice that when the crack is not aligned with the grid, it may intersect with some PD nodes. These nodes are considered to be free nodes (fully damaged) at which the constraint is the average of their family nodes (those not fully damaged). The PD solutions with 30° -rotated and 45° -rotated grids are shown in Fig. 21. They match very well the solution obtained with the original grid shown in Fig. 19. Therefore, the autonomous

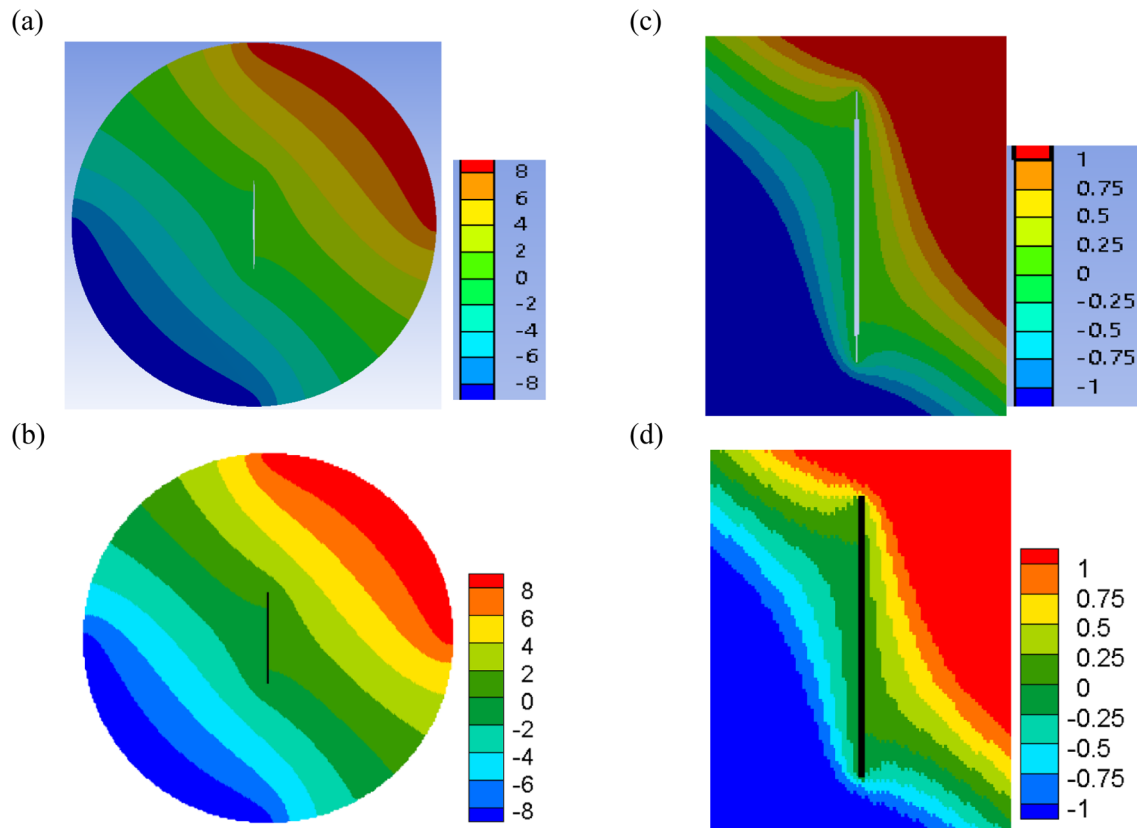


Fig. 19 Contours for the solution to the problem shown in Fig. 18, obtained with (a) ANSYS and (b) PD. In (c, d) we show zoomed-in regions around the crack for the corresponding solutions

algorithm of generating mirror nodes in FNM can handle arbitrarily-oriented uniform grids effectively.

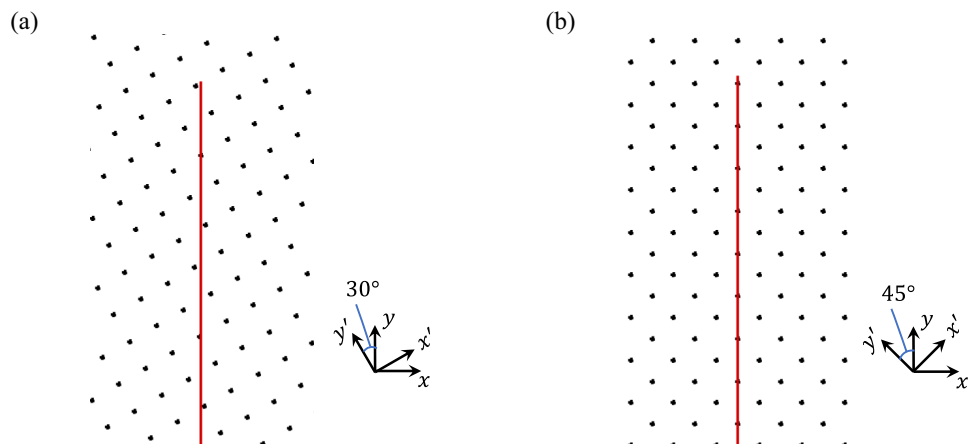
4.3.2 Diffusion in a disk with crossing cracks

To add complexity, we consider the same diffusion problem as the previous one, but with an additional horizontal

crack (of the same length as the vertical one) as shown in Fig. 22. We use the same autonomous algorithm introduced in Sect. 3.4 without any changes to treat this case. A contour plot for the PD solution and the zoomed-in picture around the two cracks with imposed Dirichlet BCs are shown in Fig. 23.

The examples shown in this section demonstrated that the autonomous algorithm for the mirror FNM works very

Fig. 20 Part of the crack (red line) and the PD grid after counterclockwise rotation by (a) 30° and (b) 45°



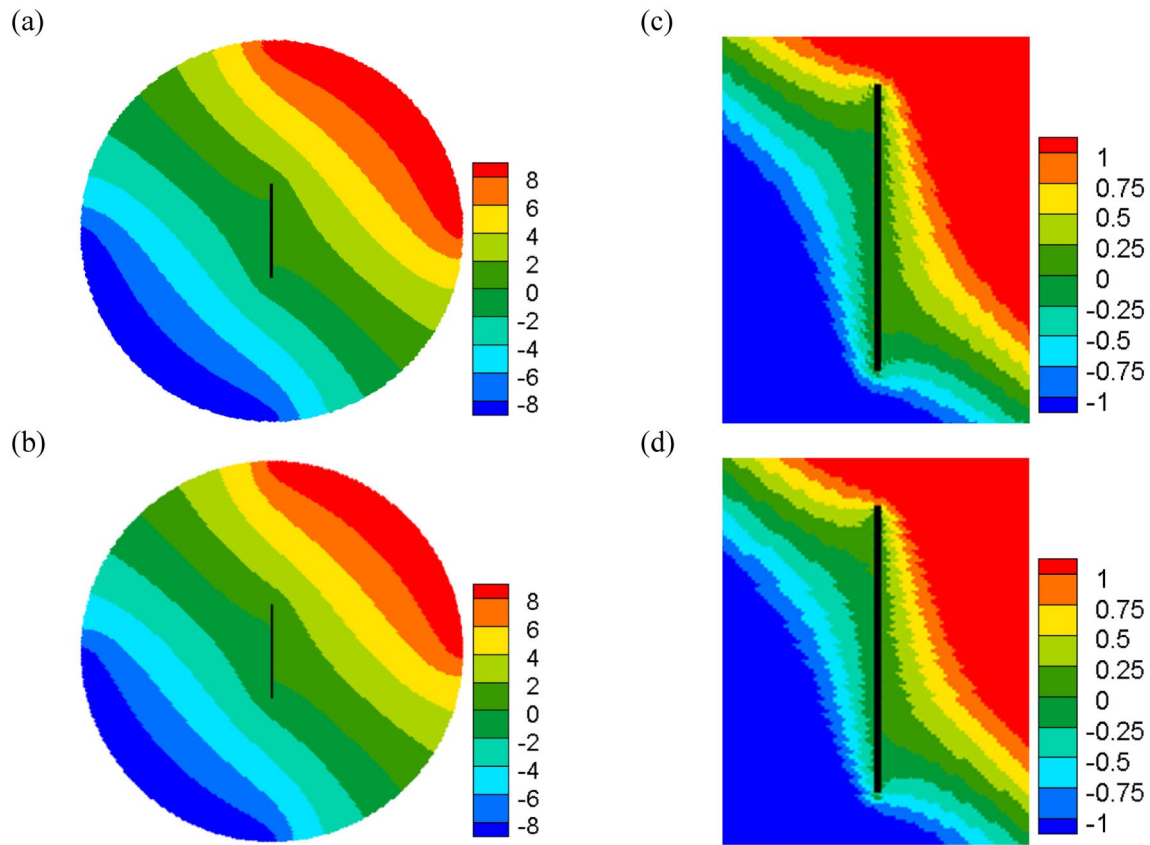


Fig. 21 Contours of PD solutions for diffusion obtained with the mirror FNM for counterclockwise-rotated grids by (a) 30° and (b) 45°. In (c, d) we show zoomed-in views around the crack for the corresponding solutions

well to enforce local BCs for complex geometries, including crack surfaces. The same algorithm could be potentially employed for problems with moving boundaries such as corrosion damage, crack propagation, etc.

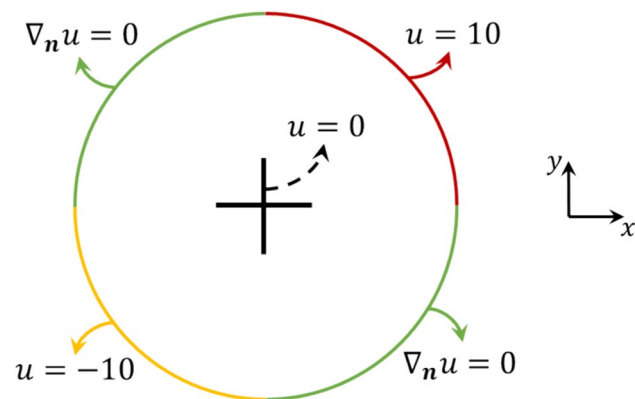


Fig. 22 Diffusion in a disk with two intersecting cracks and associated local boundary conditions

5 Conclusions

We introduced a new algorithm for the mirror-based fictitious nodes method (FNM) in peridynamic (PD) models for diffusion problems set in arbitrary geometries, including domains with cracks. Starting from computing, at each fictitious node, the peridynamic “generalized” normal vector to the domain boundary, the algorithm autonomously finds mirror nodes for all fictitious nodes. The algorithm generates the necessary data for mirror-based FNM to correctly impose the desired local BCs in the nonlocal model and reduce/eliminate the surface effect caused by the incomplete nonlocal region near the free boundary/surface in PD diffusion models. The same algorithm, with minor changes, should also work for other types of PD models, including those with moving boundaries and growing cracks.

We compared the mirror-based FNM with the naïve and the Taylor-based FNMs for problems in which the corresponding local models of diffusion have singularities along the boundary and showed that the PD solution with the mirror-based FNM agrees with the classical solution best among the three approaches. The other two methods

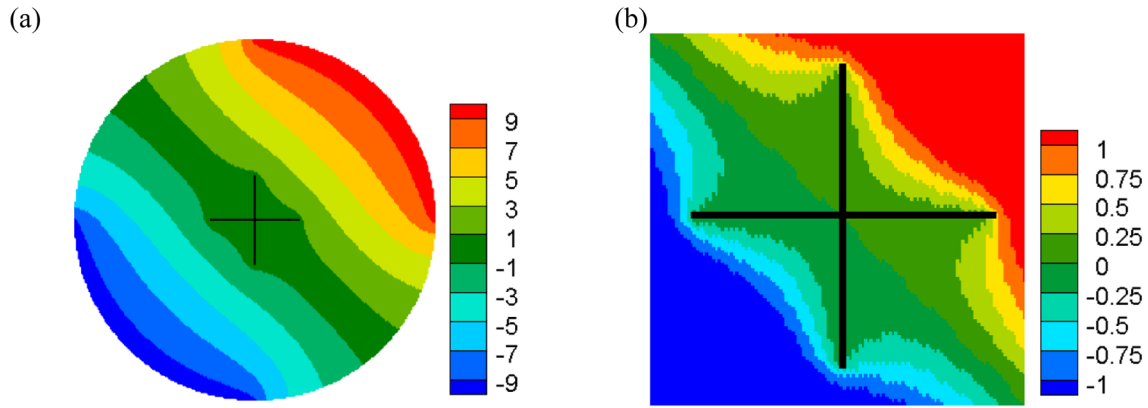


Fig. 23 Contour plot for the PD solution obtained with the new mirror FNM algorithm (a) over the disk and (b) over the zoomed-in area around the intersecting cracks. Crack lines are shown superposed on top of the nodal “temperature” plots

showed significant “pollution” of the solution far from the location of the singularity. We applied the new algorithm to diffusion problems in domains with curved boundaries and internal cracks. In these cases, the peridynamic solution obtained with the new method matched well with the FE solution obtained in ANSYS.

The new algorithm enables FNM imposition of local BCs in PD models on arbitrary domains and leads to more accurate solutions near the boundaries. High accuracy near arbitrarily-shaped boundaries and material interfaces is crucial in, for example, problems that involve crack initiation and propagation, or evolution of corrosion fronts. In such problems, the new method introduced here will have a great impact.

Future work on the subject includes extending the algorithm to 3D problems as well as problems using irregular grids.

Appendix A Numerical implementation of peridynamic diffusion model with the fictitious nodes method

For spatial discretization, we discretize the whole PD interaction region $\Omega \cup \tilde{\Omega}$ uniformly [2] into cells with nodes in the center of those cells. Figure 24 shows a 2D uniform discretization with grid spacing Δx around a node x_i . Non-uniform grids are also possible [23, 55, 56], which may conform better to shapes with, for example, rounded boundaries [57], but this is not pursued in this work. Although only 2D problems are considered here, the extension to 3D cases should be straightforward.

To approximate the peridynamic integral operator, we use a meshfree method with one-point Gaussian quadrature [2], with a modification to account for partial nodal volumes covered by the horizon region of a node [4, 58, 59]. Faster

numerical methods such as boundary-adapted spectral methods, [35, 51] can be alternative options.

The discretized PD Laplace’s equation (see Eqs. (3) and (4)) for each $x_i \in \Omega$ at n^{th} load step becomes:

$$\sum_{\substack{j \in \mathcal{H}_i \\ j \neq i}} \frac{u_j^n - u_i^n}{\xi_{ij}^2} \Delta A_{ij} = 0 \quad (22)$$

where the superscript n means n^{th} load step; the subscripts i and j denote the current node x_i and its family node x_j respectively, in the discretized domain; $\mathcal{H}_i$ is the horizon

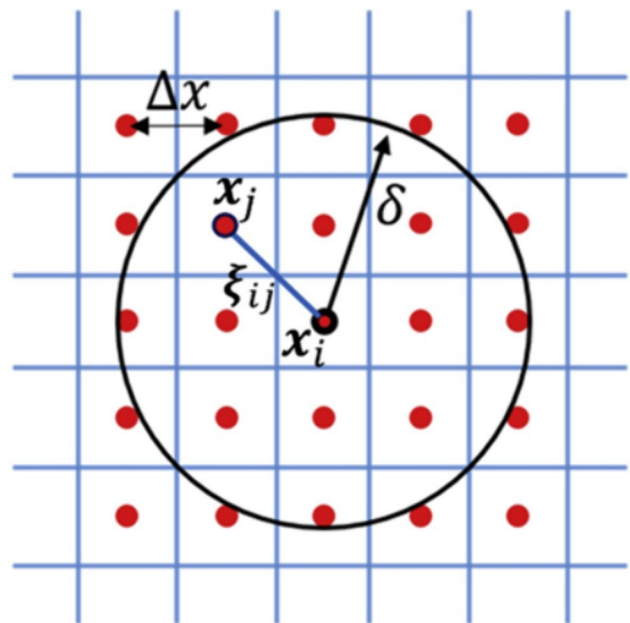


Fig. 24 Uniform discretization for a peridynamic model. The circular region is the horizon region of node x_i

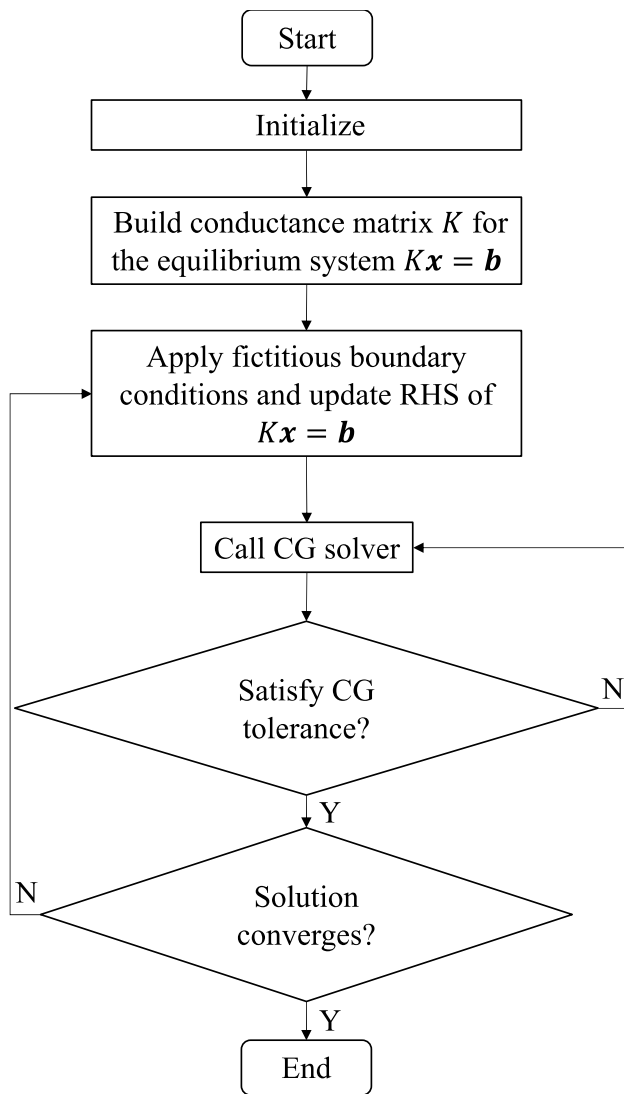


Fig. 25 Workflow for the peridynamic simulation with Taylor/mirror FNM

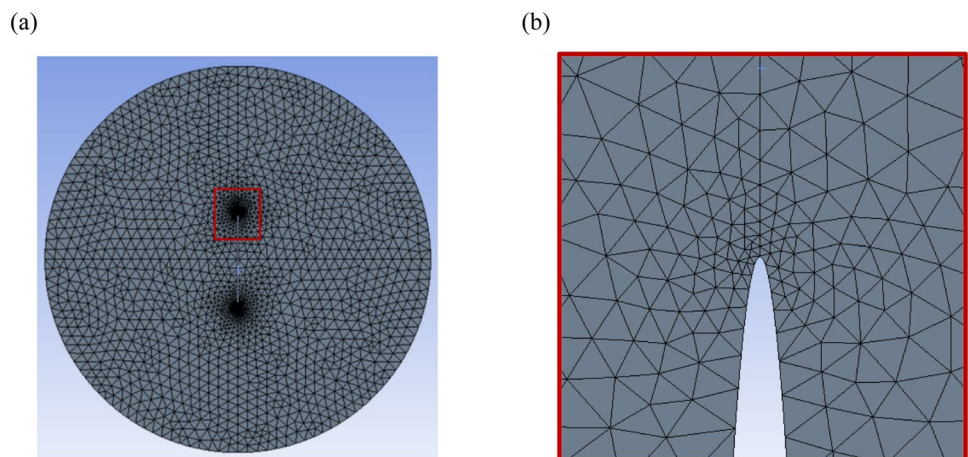
region of node x_i and $j \in \mathcal{H}_i$ includes all the nodes covered by $\mathcal{H}_i$ (fully or partially); $\xi_{ij} = \|x_j - x_i\|$ and ΔA_{ij} is the area of node x_j covered by $\mathcal{H}_i$. The discretized versions for other equations are similar to Eq. (22).

In Taylor-based and mirror-based fictitious nodes methods, the equilibrium system can be solved iteratively using the linear Conjugate Gradient (CG) solver combined with additional criteria to check for the convergence of the solution. At each iteration or solution step, the CG solver is called and the solution in the domain and fictitious region is updated. To minimize the overall computational cost, the tolerance in the CG solver is set to be 10^{-2} at first and then decreased with solution steps by a factor until it reaches 10^{-6} . This treatment can lead to simulations that are 50% more efficient than fixing the tolerance in the CG solver to be 10^{-6} from the start of the simulation. The system converges when the solutions in the domain obtained for two sequential solution steps, differ, in terms of norm-2 of their relative difference, by less than a given tolerance (10^{-6} was used in this work). The detailed workflow for a complete simulation is shown in Fig. 25.

Appendix B FEM modeling of the steady-state diffusion problem in a disk with a crack

To obtain the classical FEM-based solution for the problem shown in Fig. 18, the ANSYS Workbench Steady-State Thermal solver is used. In the FE model, the two crack surfaces are generated by two arcs with the same small curvature and the maximum space between them is 0.01, which equals the grid size in the corresponding PD model. For the mesh, as shown in Fig. 26, the element order is selected to be program-controlled and all elements are quadratic triangles with a maximum size equal to 0.05. The total number

Fig. 26 FEM mesh (a) over the whole disk; (b) near the crack tip



of nodes and elements are 9312 and 4548, respectively, and reduced integration is selected. All other options in the solver are set to their default values.

Acknowledgements This work has been supported in part by the US National Science Foundation CMMI CDS&E Grant no. 1953346 and by a Nebraska System Science award. This work was completed utilizing the Holland Computing Center of the University of Nebraska, which receives support from the Nebraska Research Initiative.

Data Availability All data is included in the manuscript.

Declarations

Conflict of interest The authors have no competing interests to declare that are relevant to the content of this article.

References

- Silling SA (2000) Reformulation of elasticity theory for discontinuities and long-range forces. *J Mech Phys Solids* 48:175–209. [https://doi.org/10.1016/S0022-5096\(99\)00029-0](https://doi.org/10.1016/S0022-5096(99)00029-0)
- Silling SA, Askari E (2005) A meshfree method based on the peridynamic model of solid mechanics. *Comput Struct* 83:1526–1535. <https://doi.org/10.1016/j.compstruc.2004.11.026>
- Madenci E, Oterkus E (2014) *Peridynamic theory and its applications*. Springer, New York
- Bobaru F, Zhang G (2015) Why do cracks branch? A peridynamic investigation of dynamic brittle fracture. *Int J Fract* 196:59–98. <https://doi.org/10.1007/s10704-015-0056-8>
- Bobaru F, Foster JT, Geubelle PH, Silling SA (2016) *Handbook of peridynamic modeling*. CRC Press
- Silling SA, Lehoucq RB (2010) Peridynamic theory of solid mechanics. In: *Advances in applied mechanics* 44:73–168. [https://doi.org/10.1016/S0065-2156\(10\)44002-8](https://doi.org/10.1016/S0065-2156(10)44002-8)
- Ha YD, Bobaru F (2010) Studies of dynamic crack propagation and crack branching with peridynamics. *Int J Fract* 162:229–244. <https://doi.org/10.1007/s10704-010-9442-4>
- Foster JT, Silling SA, Chen WW (2010) Viscoplasticity using peridynamics. *Int J Numer Methods Eng* 81:1242–1258. <https://doi.org/10.1002/nme.2725>
- Hu YL, De Carvalho NV, Madenci E (2015) Peridynamic modeling of delamination growth in composite laminates. *Compos Struct* 132:610–620. <https://doi.org/10.1016/j.compstruct.2015.05.079>
- Bobaru F, Duangpanya M (2010) The peridynamic formulation for transient heat conduction. *Int J Heat Mass Transf* 53:4047–4059. <https://doi.org/10.1016/j.ijheatmasstransfer.2010.05.024>
- Bobaru F, Duangpanya M (2012) A peridynamic formulation for transient heat conduction in bodies with evolving discontinuities. *J Comput Phys* 231:2764–2785. <https://doi.org/10.1016/j.jcp.2011.12.017>
- Oterkus S, Madenci E, Agwai A (2014) Peridynamic thermal diffusion. *J Comput Phys* 265:71–96. <https://doi.org/10.1016/j.jcp.2014.01.027>
- Zhao J, Chen Z, Mehrmashhadi J, Bobaru F (2018) Construction of a peridynamic model for transient advection-diffusion problems. *Int J Heat Mass Transf* 126:1253–1266. <https://doi.org/10.1016/j.ijheatmasstransfer.2018.06.075>
- Chen Z, Bobaru F (2015) Peridynamic modeling of pitting corrosion damage. *J Mech Phys Solids* 78:352–381. <https://doi.org/10.1016/j.jmps.2015.02.015>
- Chen Z, Zhang G, Bobaru F (2016) The influence of passive film damage on pitting corrosion. *J Electrochem Soc* 163:C19–C24. <https://doi.org/10.1149/2.0521602jes>
- Jafarzadeh S, Chen Z, Bobaru F (2018) Peridynamic modeling of repassivation in pitting corrosion of stainless steel. *Corrosion* 74:393–414. <https://doi.org/10.5006/2615>
- Jafarzadeh S, Chen Z, Bobaru F (2018) Peridynamic modeling of intergranular corrosion damage. *J Electrochem Soc* 165:C362–C374. <https://doi.org/10.1149/2.0821807jes>
- Jafarzadeh S, Chen Z, Bobaru F (2019) Computational modeling of pitting corrosion. *Corros Rev* 37:419–439. <https://doi.org/10.1515/corrrev-2019-0049>
- Jafarzadeh S, Chen Z, Li S, Bobaru F (2019) A peridynamic mechano-chemical damage model for stress-assisted corrosion. *Electrochim Acta* 323:134795. <https://doi.org/10.1016/j.electacta.2019.134795>
- Jafarzadeh S, Chen Z, Zhao J, Bobaru F (2019) Pitting, lacy covers, and pit merger in stainless steel: 3D peridynamic models. *Corros Sci* 150:17–31. <https://doi.org/10.1016/j.corsci.2019.01.006>
- Du Q, Gunzburger M, Lehoucq RB, Zhou K (2012) Analysis and approximation of nonlocal diffusion problems with volume constraints. *SIAM Rev* 54:667–696. <https://doi.org/10.1137/110833294>
- Du Q, Gunzburger M, Lehoucq RB, Zhou K (2013) A non-local vector calculus, nonlocal volume-constrained problems, and nonlocal balance laws. *Math Models Methods Appl Sci* 23:493–540. <https://doi.org/10.1142/S0218202512500546>
- Bobaru F, Ha YD (2011) Adaptive refinement and multiscale modeling in 2D peridynamics. *Int J Multiscale Comput Eng* 9:635–659. <https://doi.org/10.1615/IntJMultCompEng.201102793>
- Galvanetto U, Mudric T, Shojaei A, Zaccariotto M (2016) An effective way to couple FEM meshes and peridynamics grids for the solution of static equilibrium problems. *Mech Res Commun* 76:41–47. <https://doi.org/10.1016/j.mechrescom.2016.06.006>
- Prudhomme S, Diehl P (2020) On the treatment of boundary conditions for bond-based peridynamic models. *Comput Methods Appl Mech Eng* 372:113391. <https://doi.org/10.1016/j.cma.2020.113391>
- Ren H, Zhuang X, Cai Y, Rabczuk T (2016) Dual-horizon peridynamics. *Int J Numer Methods Eng* 108:1451–1476. <https://doi.org/10.1002/nme.5257>
- Peng X, Chen Z, Bobaru F (2023) Accurate predictions of dynamic fracture in perforated plates. *Int J Fract* 244:61–84. <https://doi.org/10.1007/s10704-023-00719-6>
- Aksoylu B, Beyer HR, Celiker F (2017) Theoretical foundations of incorporating local boundary conditions into nonlocal problems. *Rep Math Phys* 80:39–71. [https://doi.org/10.1016/S0034-4877\(17\)30061-7](https://doi.org/10.1016/S0034-4877(17)30061-7)
- Aksoylu B, Celiker F (2017) Nonlocal problems with local Dirichlet and Neumann boundary conditions. *J Mech Mater Struct* 12(4):425–437. <https://doi.org/10.2140/jomms.2017.12.425>
- Aksoylu B, Celiker F, Kilicer O (2019) Nonlocal operators with local boundary conditions in higher dimensions. *Adv Comput Math* 45:453–492. <https://doi.org/10.1007/s10444-018-9624-6>
- Aksoylu B, Gazonas GA (2020) On nonlocal problems with inhomogeneous local boundary conditions. *J Peridyn Nonlocal Model* 2:1–25. <https://doi.org/10.1007/S42102-019-00022-W/METRICS>
- Wang J, Hu W, Zhang X, Pan W (2019) Modeling heat transfer subject to inhomogeneous Neumann boundary conditions by smoothed particle hydrodynamics and peridynamics. *Int J Heat Mass Transf* 139:948–962. <https://doi.org/10.1016/j.ijheatmasstransfer.2019.05.054>

33. You H, Lu X, Task N, Yu Y (2020) An asymptotically compatible approach for Neumann-type boundary condition on nonlocal problems. *ESAIM Math Model Numer Anal* 54:1373–1413. <https://doi.org/10.1051/m2an/2019089>
34. Le QV, Bobaru F (2018) Surface corrections for peridynamic models in elasticity and fracture. *Comput Mech* 61:499–518. <https://doi.org/10.1007/s00466-017-1469-1>
35. Jafarzadeh S, Larios A, Bobaru F (2020) Efficient solutions for nonlocal diffusion problems via boundary-adapted spectral methods. *J Peridyn Nonlocal Model* 2:85–110. <https://doi.org/10.1007/s42102-019-00026-6>
36. Yu Y, You H, Trask N (2021) An asymptotically compatible treatment of traction loading in linearly elastic peridynamic fracture. *Comput Methods Appl Mech Eng* 377:113691. <https://doi.org/10.1016/j.cma.2021.113691>
37. Takeda H, Miyama SM, Sekiya M (1994) Numerical simulation of viscous flow by smoothed particle hydrodynamics. *Progr Theor Phys* 92:939–960. <https://doi.org/10.1143/ptp/92.5.939>
38. Macia F, Antuono M, Gonzalez LM, Colagrossi A (2011) Theoretical analysis of the no-slip boundary condition enforcement in SPH methods. *Progr Theor Phys* 125:1091–1121. <https://doi.org/10.1143/PTP.125.1091>
39. Scabbia F, Zaccariotto M, Galvanetto U (2021) A novel and effective way to impose boundary conditions and to mitigate the surface effect in state-based peridynamics. *Int J Numer Methods Eng* 122:5773–5811. <https://doi.org/10.1002/nme.6773>
40. Scabbia F, Zaccariotto M, Galvanetto U (2022) A new method based on Taylor expansion and nearest-node strategy to impose Dirichlet and Neumann boundary conditions in ordinary state-based peridynamics. *Comput Mech* 70:1–27. <https://doi.org/10.1007/s00466-022-02153-2>
41. Oterkus S, Madenci E, Agwai A (2014) Fully coupled peridynamic thermomechanics. *J Mech Phys Solids* 64:1–23. <https://doi.org/10.1016/j.jmps.2013.10.011>
42. Chen Z, Bobaru F (2015) Selecting the kernel in a peridynamic formulation: a study for transient heat diffusion. *Comput Phys Commun* 197:51–60. <https://doi.org/10.1016/j.cpc.2015.08.006>
43. Morris JP, Fox PJ, Zhu Y (1997) Modeling low Reynolds number incompressible flows using SPH. *J Comput Phys* 136:214–226. <https://doi.org/10.1006/jcph.1997.5776>
44. Motz H (1947) The treatment of singularities of partial differential equations by relaxation methods. *Q Appl Math* 4:371–377. <https://doi.org/10.1090/qam/18442>
45. Li ZC, Lu TT (2000) Singularities and treatments of elliptic boundary value problems. *Math Comput Model* 31:97–145. [https://doi.org/10.1016/S0895-7177\(00\)00062-5](https://doi.org/10.1016/S0895-7177(00)00062-5)
46. Radu P, Wells K (2019) A doubly nonlocal laplace operator and its connection to the classical Laplacian. *J Integr Equ Appl* 31:379–409. <https://doi.org/10.1216/JIE-2019-31-3-379>
47. Tian X, Du Q (2014) Asymptotically compatible schemes and applications to robust discretization of nonlocal models. *SIAM J Numer Anal* 52:1641–1665. <https://doi.org/10.1137/130942644>
48. Tao Y, Tian X, Du Q (2017) Nonlocal diffusion and peridynamic models with Neumann type constraints and their numerical approximations. *Appl Math Comput* 305:282–298. <https://doi.org/10.1016/j.amc.2017.01.061>
49. Bobaru F, Hu W (2012) The meaning, selection, and use of the peridynamic horizon and its relation to crack branching in brittle materials. *Int J Fract* 176:215–222. <https://doi.org/10.1007/s10704-012-9725-z>
50. Ascher UM, Greif C (2011) A first course in numerical methods. Society for Industrial and Applied Mathematics, Philadelphia
51. Jafarzadeh S, Wang L, Larios A, Bobaru F (2021) A fast convolution-based method for peridynamic transient diffusion in arbitrary domains. *Comput Methods Appl Mech Eng* 375:113633. <https://doi.org/10.1016/j.cma.2020.113633>
52. Yosibash Z (2012) Singularities in elliptic boundary value problems and elasticity and their connection with failure initiation. Springer, New York
53. Hu HY, Li ZC (2006) Collocation methods for Poisson's equation. *Comput Methods Appl Mech Eng* 195:4139–4160. <https://doi.org/10.1016/j.cma.2005.07.019>
54. Bobaru F, Yang M, Alves LF et al (2009) Convergence, adaptive refinement, and scaling in 1D peridynamics. *Int J Numer Methods Eng* 77:852–877. <https://doi.org/10.1002/nme.2439>
55. Henke SF, Shanbhag S (2014) Mesh sensitivity in peridynamic simulations. *Comput Phys Commun* 185:181–193. <https://doi.org/10.1016/j.cpc.2013.09.010>
56. Gu X, Zhang Q, Xia X (2017) Voronoi-based peridynamics and cracking analysis with adaptive refinement. *Int J Numer Methods Eng* 112:2087–2109. <https://doi.org/10.1002/nme.5596>
57. Zhao J, Chen Z, Mehrmashhadi J, Bobaru F (2020) A stochastic multiscale peridynamic model for corrosion-induced fracture in reinforced concrete. *Eng Fract Mech* 229:106969. <https://doi.org/10.1016/j.engfracmech.2020.106969>
58. Hu W, Ha YD, Bobaru F (2010) Numerical integration in peridynamics, Technical Report. University of Nebraska-Lincoln, Department of Mechanical and Materials Engineering
59. Seleson P (2014) Improved one-point quadrature algorithms for two-dimensional peridynamic models based on analytical calculations. *Comput Methods Appl Mech Eng* 282:184–217. <https://doi.org/10.1016/j.cma.2014.06.016>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.