AIMS Mathematics, 11(3): 7980–8013.
DOI: 10.3934/math.2026329
Received: 30 September 2025
Revised: 03 March 2026
Accepted: 05 March 2026
Published: 25 March 2026

http://www.aimspress.com/journal/Math

Research article

Numerical solution of the Hibler model of sea-ice dynamics: Comparing backward Euler and Crank-Nicolson methods

Salim Bensassi[1], **Boualem Khouider**[2,*], **Clint Seinen**[2] **and M'hamed Kesri**[1]

[1] Dynamical Systems Laboratory, Department of Analysis, Faculty of Mathematics, University of Science and Technology Houari Boumediene, BP 32 Bab Ezzouar, 16111 - Algiers, Algeria
[2] Mathematics and Statistics, University of Victoria, P.O. Box 1700 STN CSC, Victoria, Canada

* **Correspondence:** Email: khouider@uvic.ca; Tel: 1(250)7217439.

Abstract: The viscous-plastic momentum equations for sea-ice dynamics are a model for the time evolution of the horizontal velocity field of a vertically homogeneous column of ice-pack continuum whose thickness and ice concentration vary in time and space. They constitute a highly nonlinear system of partial differential equations, and as such, both the mathematical analysis and numerical solution of its solution remain a challenge. To shed some light on this problem, we compared the performance of two known time discretization methods: the backward Euler (BE) and the Crank-Nicolson (CN) methods. Both methods are in theory unconditionally stable, but Euler's method is only first-order accurate while Crank-Nicolson is second-order. Centered finite differences were used for the spatial derivatives. This led to a nonlinear system of algebraic equations which was then solved using a Jacobian-free Newton-Krylov approach. First, the two methods were compared in terms of their ability to reproduce a synthetic solution to the viscous-plastic momentum equations when an artificial forcing is applied. The convergence of the two methods is assessed for a short time integration period of 12 hours and grid resolutions of $dx = 5, 10, 20$ km and $dt = 20, 10, 5$ minutes. While both methods showed an overall second-order convergence in space, only the BE method displayed the expected first-order convergence in timestep refinement. The CN's errors didn't decrease under time refinement, although they are consistently twice as small as the BE's errors. This behaviour persisted in longer 4-day runs, where the BE method displayed a more stable solution overall, especially at the coarse resolution with $dt = 80$ minutes, where the CN failed to converge. The two methods are then compared in a 1-year climatic simulation with realistic wind and ocean forcings, using a grid resolution of 20 km where both methods performed fairly well. However, after 1 year the two solutions diverged somewhat significantly from one-another, providing hard evidence that climate models are sensitive to the underlying numerical methods.

Keywords: viscous-plastic sea-ice equations; Jacobian-free Newton-Krylov; numerical solutions; Crank-Nicolson; backward Euler; numerical stability

Mathematics Subject Classification: 35Q74, 35Q86, 65M06

1. Introduction

Sea ice is a material composed of ice floes of various sizes and shapes. It plays a significant role in the climate-earth system, particularly in the Arctic and Antarctic regions. Sea ice albedo affects how much sunlight is absorbed by the ocean, forming a partially insulating layer that decouples the thermodynamic exchange between the ocean and atmosphere. Additionally, sea ice melt in the summer months constitutes an important source of freshwater in the upper ocean, which in turn influences the global overturning circulation of the ocean [19]. Furthermore, as sea ice cover diminishes, the ocean surface temperature rises due to the diminished albedo, which may in turn accelerate the melting process. In this sense, sea ice is known as a positive feedback parameter of global warming, which is not only an indicator of the effect on anthropogenic climatic forcing but also part of the equation towards understanding this forcing and its consequences for the earth system.

The earliest mathematical models of sea ice internal forces emerged in the early 1960s, presented as a two-dimensional, thin-layer model with linear-viscous dynamics. For example, in 1965 Campbell [3] presented a steady circulation model of ice and water for the whole Arctic Basin based on this concept. However, these ideas were quickly abandoned in favour of more complex, nonlinear processes. Sea ice is now viewed more as a granular material, similar to the flow of some clay fields and rocks. This major leap was due to the findings of the Arctic Ice Dynamics Joint Experiment (AIDJEX) in the 1970s. This international consortium led the first comprehensive field experiment in the Arctic to understand sea ice physics and dynamics. They proposed an elastic-plastic rheology for sea ice internal deformations and the concept of thickness distribution, as outlined in [7].

The numerical complexity of elastic-plastic rheology, coupled with the difficulty in determining the exact shape of the yield curve, led Hibler to propose a viscous-plastic (VP) rheology based on an elliptic yield curve in 1979 as a compromise [12]. The mushiness of sea ice on large horizontal scales of 100 km and larger, which can contain many ice floes, and especially during the melt season, was used to justify the viscous behaviour when the internal stress is below the plastic yield threshold. The VP model of sea ice has become one of the most successful ones in terms of its applicability today, as it has been adopted by most of the existing operational earth-system models [9]. However, limitations of the Hilber model in accurate representation on linear kinematic features that appear to characterize sea-ice dynamics have been reported in the literature, and brittle rheologies have been proposed as an alternative. See [29, 30] and references therein.

Despite the sea ice momentum equations (SIME), with a VP formulation remaining intrinsically difficult to solve numerically due to their nonlinear nature, significant research has been dedicated to developing efficient numerical methods over the past decades. Hibler et al. [12, 36] used a multi-time level method to solve the VP momentum equations. Their method is based on a modified Euler time step, where the solution is first advanced to the middle of the time step by solving the linearized system implicitly via a successive over relaxation (SOR) solver. Zheng and Hibler [36] found that the underlying pseudo time stepping failed to produce a realistic VP solution and Hunke and Zhang [16] showed that this approach resulted in a very slow transient response to forcing unless small time steps,

relative to the time-scale of the changing forcing, were used.

An implicit time stepping is often used for stability reasons because an explicit scheme for the SIME with VP rheology requires a very small time step, similar to the heat equation with a large heat conductivity parameter. To avoid the difficulty of inverting the highly nonlinear rheology functions associated with implicit time stepping, Hunke and collaborators introduced the elastic-viscous-plastic (EVP) model in [15, 17]. This model includes a relaxation equation for the internal stress tensor, allowing for an explicit scheme with large time steps. In essence, this adds an artificial elastic term to the VP constitutive law to become the EVP model. However, recently, Losch and Danilov [24] showed that the EVP model can also result in notably different solutions from the fully VP approach, which may be deemed unphysical. Additionally, the EVP model is known to produce artificial elastic waves that may pollute the numerical solution. Nonetheless, approximations that provide stabilized approximations for the Newton solver have been introduced and used in [25, 26, 35].

For a better solution maintaining the original VP rheology, Lemieux et al. [21, 22] proposed a Jacobian-free Newton-Krylov (JFNK) solver for the discretized SIME with VP formulation, utilizing the backward Euler (BE) time stepping. This method comprises two distinct stages. Initially, Newton's method is formulated to solve the nonlinear system. Subsequently, a Krylov subspace method is employed to solve the linear system without explicitly computing the Jacobian matrix. A first-order approximation of the Gateaux derivative, based on a small parameter ϵ, is utilized to approximate the action of the Jacobian matrix on the Newton iterations' increments. Seinen and Khouider [32] focused on Lemieux and collaborators' work [21, 22] and proposed the Crank-Nicolson (CN) scheme, specifically the trapezoidal or the second Adams-Moulton scheme, instead of the BE method. In addition, they improved the Gateaux derivative approximation of the Jacobian matrix by combining a closed-form differentiation for all linear and nearly linear terms with a second-order approximation for the remaining nonlinear terms. Numerical simulations of a synthetic test problem demonstrated that this improved approximation, with an error value between 10^{-8} and 10^{-4}, significantly improved the performance of the JFNK solver. They also empirically determined an optimal tolerance for the conditional termination parameter of JFNK based on efficiency and accuracy. Building on this work, we propose and test the first-order BE scheme in comparison to the second-order CN scheme on the same synthetic problem while maintaining the improved approximation of the Jacobian matrix. As in the work of Lemieux and collaborators, Seinen and Khouider used second-order centered finite differences based on the Arakawa C-grid for the spatial discretization of the SIME-VP model.

The scientific motivation behind this work stems from a practical consideration. While both the BE and CN schemes are unconditionally stable and well-suited for the SIME-VP, the former has a formal first-order accuracy, while the second is second-order. Additionally, the BE method is known to have better A-stability properties than the trapezoidal method, making it more adaptable for stiff problems. Therefore, it is desirable to determine whether the second-order accuracy of the CN scheme outweighs the better stability properties of the BE scheme. Stable and accurate time-stepping schemes for nonlinear parabolic problems, some of which include stochastic elements, have been successfully used in several studies [1, 2, 18, 27].

The remainder of this paper is divided into eight sections. Section 2 describes the viscous-plastic sea ice model. Section 3 introduces the Arakawa C-grid and the numerical methods used to discretize and solve the SIME equations. Section 4 presents the synthetic test problem, and Sections 5, 6, and 7 summarize the comparison of the performance of the two schemes, respectively, in terms of their

behaviour under grid refinement, their performance at coarse resolution, and the overall fidelity for climatic scale simulation under prescribed realistic wind and ocean current forcings. Finally, Section 8 concludes the paper.

2. The viscous-plastic sea ice model

The Hibler model of sea ice dynamics consists of two main components: the viscous-plastic sea ice momentum equations (Eq (2.1)) and the continuity equations (2.2) and (2.3) for the ice thickness h and the area concentration A of sea ice [13]:

$$\rho h\frac{D\mathbf{u}}{Dt} = -\rho h f(\mathbf{k}\times\mathbf{u}) + \tau_a - \tau_w + \nabla.\sigma - \rho h g\nabla H_d, \tag{2.1}$$

$$\frac{\partial h}{\partial t} + \nabla.(h\mathbf{u}) = S_h, \tag{2.2}$$

$$\frac{\partial A}{\partial t} + \nabla.(A\mathbf{u}) = S_A. \tag{2.3}$$

Here, ρ is the density of the ice, h the sea ice thickness, f the Coriolis parameter, $\frac{D}{Dt} \equiv \frac{\partial}{\partial t} + u\frac{\partial}{\partial x} + v\frac{\partial}{\partial y}$ is the advective derivative (although the nonlinear advection terms are neglected in practice), $\mathbf{u} = u\mathbf{i}+v\mathbf{j}$ the horizontal sea ice velocity, and $\mathbf{i}$, $\mathbf{j}$, and $\mathbf{k}$ are the Cartesian unit vectors. We denote by x and y the coordinates in the east, west, or zonal direction $\mathbf{i}$ and in the north, south, or meridional direction $\mathbf{j}$, respectively, while t is time. τ_a is the wind stress, τ_w the water drag, σ the internal ice stress tensor, and $\nabla.\sigma$ is known as the rheology term. H_d is the sea surface height, and g is the acceleration due to gravity. Further, the sea surface tilt is given in terms of the geostrophic ocean current [34]:

$$g\nabla H_d = -f\mathbf{k}\times\mathbf{u}_w^g, \tag{2.4}$$

where $\mathbf{u}_w^g$ is the geostrophic ocean current. Also, following [22], the air and water drag terms are expressed via an empirical quadratic law with constant turning angles and constant drag coefficients [21]:

$$\tau_a = \rho_a C_{da} \mid \mathbf{u}_a^g \mid (\mathbf{u}_a^g\cos\theta_a + (\mathbf{k}\times\mathbf{u}_a^g)\sin\theta_a), \tag{2.5}$$

$$\tau_w = C_w((\mathbf{u}-\mathbf{u}_a^g)\cos\theta_w + (\mathbf{k}\times(\mathbf{u}-\mathbf{u}_a^g)\sin\theta_w), \tag{2.6}$$

$$C_w = \rho_w C_{dw} \mid \mathbf{u}-\mathbf{u}_w^g \mid, \tag{2.7}$$

where $\mathbf{u}_a^g$ is the geostrophic wind and ρ_a and ρ_w represent the air and water densities, respectively, while C_{da} and C_{dw} are the air and water drag coefficients. Because $\mathbf{u}$ is much smaller than $\mathbf{u}_a^g$, $\mathbf{u}$ is neglected in the expression for the wind stress [21]. The VP constitutive law, relating internal stresses and strain rates reads [12]

$$\sigma_{ij} = 2\eta\dot{\epsilon}_{ij} + [\zeta-\eta]\dot{\epsilon}_{kk}\delta_{ij} - P\delta_{ij}/2, \quad i,j = 1,2, \tag{2.8}$$

where σ_{ij} is the i, j component of the internal stress tensor, ζ and η are the bulk and shear viscosities, respectively, and δ_{ij} is the Kronecker delta. The strain rates, $\dot{\epsilon}_{ij}$, are defined as [21, 31]

$$\dot{\epsilon}_{11} = \frac{\partial u}{\partial x}, \quad \dot{\epsilon}_{22} = \frac{\partial v}{\partial y}, \quad \dot{\epsilon}_{12} = \dot{\epsilon}_{21} = \frac{1}{2}(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}), \quad \dot{\epsilon}_{kk} = \dot{\epsilon}_{11}+\dot{\epsilon}_{22}. \tag{2.9}$$

The parameterization of the ice strength or pressure, P, is given by [13]

$$P = P^* h \exp[-C(1 - A)], \tag{2.10}$$

where $P^* = 27.5 \times 10^3$ N m^{-2} is the ice strength parameter, and $C = 20$ is the ice concentration parameter. Lemieux et al. [20] define the bulk and shear viscosities by these formulas:

$$\zeta = KP \tanh(\frac{\mathbf{1}}{2\triangle K}), \tag{2.11}$$

$$\eta = \zeta e^{-2}, \tag{2.12}$$

where $K = 2.5 \times 10^8$ s, $e = 2$ is the principal axis ratio of the elliptical yield curve, and

$$\Delta = [(\dot{\epsilon}_{11}^2 + \dot{\epsilon}_{22}^2)(1 + e^{-2}) + 4e^{-2}\dot{\epsilon}_{12}^2 + 2\dot{\epsilon}_{11}^2\dot{\epsilon}_{22}^2(1 - e^{-2})]^{1/2}. \tag{2.13}$$

The standard approach for solving the sea ice dynamics model (Eqs (2.1)–(2.3)) is a time splitting method. In this part, we focus only on solving Eq (2.1); we neglected the momentum advection term in (2.1); because this term is very small compared to the other ones in the SIME, according to [36] and [17], we write the u and v components of (2.1) as;

$$-\rho h\frac{\partial u}{\partial t} + \rho h f v + \tau_{au} - \tau_{wu} + \frac{\partial \sigma_{11}}{\partial x} + \frac{\partial \sigma_{12}}{\partial y} - \rho h g\frac{\partial H_d}{\partial x} = 0, \tag{2.14}$$

$$-\rho h\frac{\partial v}{\partial t} - \rho h f u + \tau_{av} - \tau_{wv} + \frac{\partial \sigma_{21}}{\partial x} + \frac{\partial \sigma_{22}}{\partial y} - \rho h g\frac{\partial H_d}{\partial y} = 0, \tag{2.15}$$

where τ_{au} and τ_{wu} represent the u-components of the air and water drag, likewise for the v-components, and the rheology terms are given by

$$\frac{\partial \sigma_{11}}{\partial x} + \frac{\partial \sigma_{12}}{\partial y} = \frac{\partial}{\partial x}[(\eta + \zeta)\frac{\partial u}{\partial x}] + \frac{\partial}{\partial x}[(\eta - \zeta)\frac{\partial v}{\partial y}] + \frac{\partial}{\partial y}[\eta\frac{\partial u}{\partial y}] + \frac{\partial}{\partial y}[\eta\frac{\partial v}{\partial x}] - \frac{1}{2}\frac{\partial P}{\partial x}, \tag{2.16}$$

and

$$\frac{\partial \sigma_{21}}{\partial x} + \frac{\partial \sigma_{22}}{\partial y} = \frac{\partial}{\partial y}[(\eta + \zeta)\frac{\partial u}{\partial y}] + \frac{\partial}{\partial y}[(\eta - \zeta)\frac{\partial v}{\partial x}] + \frac{\partial}{\partial x}[\eta\frac{\partial u}{\partial y}] + \frac{\partial}{\partial x}[\eta\frac{\partial v}{\partial x}] - \frac{1}{2}\frac{\partial P}{\partial y}. \tag{2.17}$$

Additionally, as done in [32], to get $\partial_*\eta$ and $\partial_*\zeta$, we differentiate (3.3) to produce

$$\partial_*\zeta = k\tanh(\frac{1}{2\Delta k})\partial_* P - \frac{P}{2\Delta^2} \times [1 - \tanh^2(\frac{1}{2\Delta k})]\partial_*\Delta, \tag{2.18}$$

and for mathematical convenience, we use $\partial_*\Delta = \frac{\partial_*(\Delta^2)}{2\Delta}$ to get our final form for the viscosity parameter

$$\partial_*\zeta = k\tanh(\frac{1}{2\Delta k})\partial_* P - \frac{P}{2\Delta^3} \times [1 - \tanh^2(\frac{1}{2\Delta k})]\partial_*(\Delta^2), \tag{2.19}$$

and calculate $\partial_*(\Delta^2)$ according to

$$\partial_*(\Delta^2) = (1 - e^{-2})[2\dot{\epsilon}_{11}\partial_*\dot{\epsilon}_{11} + 2\dot{\epsilon}_{22}\partial_*\dot{\epsilon}_{22}] + 8e^{-2}\dot{\epsilon}_{12}\partial_*\dot{\epsilon}_{12} + 2(1 - e^{-2})[\dot{\epsilon}_{22}\partial_*\dot{\epsilon}_{11} + \dot{\epsilon}_{11}\partial_*\dot{\epsilon}_{22}]. \tag{2.20}$$

3. The numerical method

The VP-SIMEs are discretized on the Arakawa C-grid. In this grid, the cell centres are known as tracer points, while the corner points are the node points. The ice velocities, u and v, are anchored at the centres of the vertical (along the y-coordinate) and horizontal (along the x-coordinate) edges. To evaluate the x- and y-derivatives, the middle of the horizontal and vertical interfaces are used. This results in a second-order approximation of mixed derivatives at the cell centres, which are also used as anchor points for the thickness and ice area tracers that define the pressure or ice strength field P. The u and v derivatives are linearly interpolated in the perpendicular direction to obtain the viscosity coefficients ζ and η at the cell centres. For more details on the spatial discretization using the C-grid, refer to [32].

For convenience, we denote by U-grid the set of grid points at which the velocity component u is defined and by V-grid for points at which the v-velocity is defined.

Many considerations towards the minimization of apparent numerical errors have been taken care of in [32]. For instance, to limit the level of numerical truncations, the rheology terms in (2.16) and (2.17) and the water drag terms in (2.14) and (2.15), have been rearranged accordingly, prior to their discretization. In the case of the implementation of the BE scheme, for example the time-discretized SIME, are first written in component-wise form as:

$$\begin{aligned} &-\rho h\frac{u^n}{dt}+\rho h f v^n_{avg}+\frac{\partial}{\partial x}\Big[(\eta+\zeta)\frac{\partial u}{\partial x}\Big]^n+\frac{\partial}{\partial y}\Big[\eta\frac{\partial u}{\partial y}\Big]^n+\frac{\partial}{\partial x}\Big[(\zeta-\eta)\frac{\partial v}{\partial y}\Big]^n\\ &+\frac{\partial}{\partial y}\Big[\eta\frac{\partial v}{\partial x}\Big]^n-C^n_{wu}(u^n\cos\theta_w-v^n_{avg}\sin\theta_w)\\ =&-\rho h\frac{u^{n-1}}{dt}-C^n_{wu}(u^g_w\cos\theta_w-v^g_{wavg}\sin\theta_w)+r_u, \end{aligned} \tag{3.1}$$

and

$$\begin{aligned} &-\rho h\frac{v^n}{dt}-\rho h f u^n_{avg}+\frac{\partial}{\partial y}\Big[(\eta+\zeta)\frac{\partial v}{\partial y}\Big]^n+\frac{\partial}{\partial x}\Big[\eta\frac{\partial v}{\partial x}\Big]^n+\frac{\partial}{\partial y}\Big[(\zeta-\eta)\frac{\partial u}{\partial x}\Big]^n\\ &+\frac{\partial}{\partial x}\Big[\eta\frac{\partial u}{\partial y}\Big]^n-C^n_{wv}(v^n\cos\theta_w-u^n_{avg}\sin\theta_w)\\ =&-\rho h\frac{v^{n-1}}{dt}-C^n_{wv}(v^g_w\cos\theta_w-u^g_{wavg}\sin\theta_w)+r_v, \end{aligned} \tag{3.2}$$

on the U- and V-grids, respectively. Here, r_u and r_v include all the terms that do not depend on u and v, i.e., the pressure gradient term, the sea surface tilt and the wind stress. Moreover v_{avg} is the average of the four v components of velocity surrounding a u location on the C-grid (same idea for u_{avg}). The same idea applies for u^g_{wavg} and v^g_{wavg} for the ocean currents. For boundary conditions, we apply homogeneous Dirichlet conditions at land boundaries, $\mathbf{u}=\mathbf{0}$.

The spatial discretization of Eqs (3.1) and (3.2) uses centred finite differences along the U- and V-grids, respectively. The details are found in [32]. This leads to a system of N nonlinear equations with N unknowns, which can be concisely written as

$$\mathbf{A}(\mathbf{u}^n)\mathbf{u}^n=\mathbf{b}(\mathbf{u}^n), \tag{3.3}$$

where $\mathbf{u}^n$ is the N-dimensional solution vector. This vector is created by first stacking all U points in our computational domain into an array and then all the V points. $\mathbf{A}(\mathbf{u}^n)$ is an $N \times N$ matrix containing all the necessary coefficients (both linear and nonlinear) from the left-hand side of Eqs (3.1) and (3.2). $\mathbf{b}(\mathbf{u}^n)$ is an N-dimensional vector containing the right-hand side of Eqs (3.1) and (3.2). The dependency of $\mathbf{b}$ and $\mathbf{u}^n$ accounts for the presence of the nonlinear coefficient C_w^n.

To solve (3.3), we use a Jacobian free Newton–Krylov (JFNK) iterative-approach. Letting $\mathbf{u}^k$ be the kth iterate, its associated residual is:

$$\mathbf{F}(\mathbf{u}^k) = \mathbf{A}(\mathbf{u}^k)\mathbf{u}^k - \mathbf{b}(\mathbf{u}^k). \tag{3.4}$$

Taking a first order multivariate Taylor expansion about $\mathbf{F}(\mathbf{u}^k)$ and setting the left hand side to zero gives us the linear system:

$$\mathbf{J}(\mathbf{u}^k)\delta\mathbf{u}^k = -\mathbf{F}(\mathbf{u}^k) \quad where \quad \mathbf{u}^{k+1} = \mathbf{u}^k + \delta\mathbf{u}^k,$$

which is solved, for the increment $\delta\mathbf{u}^k$ using the generalized minimum residual (GMRES) method. This solver does not require the calculation of the Jacobian matrix. It needs only the calculation of the its action on a given vector.

Lemieux et al. [22] use the first-order Gateaux derivative to estimate this action, based on the following approximation:

$$\mathbf{J}(\mathbf{u}^k)\mathbf{w} \approx \frac{\mathbf{F}(\mathbf{u}^k + \epsilon\mathbf{w}) - \mathbf{F}(\mathbf{u}^k)}{\epsilon}. \tag{3.5}$$

Seinen and Khouider [32] improved this approximation of the Jacobian by combining an exact calculation of all derivatives associated with term $\mathbf{b}(\mathbf{u})$ and centered finite differences to approximate the Gateaux derivate of the highly nonlinear part of $\mathbf{J}$, involving the rheology terms. They get the following form of the Jacobian approximation:

$$\mathbf{J}(\mathbf{u}^k)\mathbf{w} \approx \frac{\mathbf{A}(\mathbf{u}^k + \epsilon\mathbf{w})\mathbf{u}^k - \mathbf{A}(\mathbf{u}^k - \epsilon\mathbf{w})\mathbf{u}^k}{2\epsilon} + \mathbf{A}(\mathbf{u}^k)\mathbf{w} - D\mathbf{b}(\mathbf{u}^k)\mathbf{w}. \tag{3.6}$$

Here, D denotes the Jacobian differential operator of $\mathbf{b}(\mathbf{u})$ evaluated at $\mathbf{u} = \mathbf{u}^k$, as described in [32].

The final step is to provide stopping criteria for both the nonlinear and linear solvers. We follow a procedure introduced and used in [32]. For the linear solver, the stopping criterion is given by

$$\|J(\mathbf{u}^{k-1})\delta\mathbf{u}^{k-1} + \mathbf{F}(\mathbf{u}^{k-1})\| < \gamma(k-1)\|\mathbf{F}(\mathbf{u}^{k-1})\|.$$

Here, $\delta\mathbf{u}^{k-1}$ is the approximate solution to (3.24), $\|.\|$ represents the $\mathbf{L}_2$ functional norm, and $\gamma(k-1)$ is given by

$$\gamma(k-1) = \begin{cases} \gamma_{ini} & if \ \ \|\mathbf{F}(\mathbf{u}^{k-1})\| \geq res_t, \\ \min(\gamma_{ini}, \|\mathbf{F}(\mathbf{u}^{k-1})\|/\|\mathbf{F}(\mathbf{u}^{k-2})\|) & if \ \ \|\mathbf{F}(\mathbf{u}^{k-1})\| < res_t, \end{cases} \tag{3.7}$$

where $\gamma_{ini} = 0.99$ and $res_t = 0.625$.

For the nonlinear solver (Newton iterations), the stopping criterion used is given by

$$\| \mathbf{F}(\mathbf{u}^{k-1}) \| \leq \rho h_0 f u_0 \epsilon_{tol}, \tag{3.8}$$

where u_0 is a typical sea ice velocity, chosen to be 0.1 m/s, $h_0 = 1$ m is a reference ice thickness, the ice density is set to $\rho = 900$ kg m^{-3}, and

$$\epsilon_{tol} = \gamma_{nl}(\frac{dx}{L_x})^2, \tag{3.9}$$

where L_x is the length scale of our domain, chosen to be the domain extent in the x-direction, and γ_{nl} is a tolerance coefficient parameter whose default value is $\gamma_{nl} = 10$.

In the case of the CN time discretization, the equations in (3.1) and (3.2) are replaced by their equivalent when the trapezoidal method is used for the time derivative. Simply put, this amounts to multiplying all terms of the left hand sides of these equations, other than those divided by dt, by half and then adding the exact same terms but evaluated at the current time step, i.e, $n - 1$, to the right of the equations.

4. Set up and synthetic solution

The ideal situation for validating and assessing the performance the numerical scheme is to test it on a known (exact) solution. Unfortunately, the SIME does not have any known analytical solutions. Nevertheless, we can slightly modify the governing equations by adding a prescribed forcing on the right hand side of (2.1) so the resulting system has an exact solution, and then test the solver's ability to approximate that solution. We call this solution a synthetic solution. Specifically, we consider the forced SIME system

$$-\rho h\frac{\partial \mathbf{u}}{\partial t} - \rho h f(\mathbf{k}\times\mathbf{u}) + \tau_a - \tau_w + \nabla.\sigma - \rho h g\nabla H_d = \mathcal{L}(\mathbf{w}), \tag{4.1}$$

where $\mathcal{L}(\mathbf{w})$ is produced by replacing $\mathbf{u}$ in (2.1) with the 2-D propagating sine wave [32]:

$$\mathbf{w} = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{10}\sin\left((\frac{4x}{L_x}-2)^2 + (\frac{4y}{L_y}-2)^2 + ct\right) \\ \frac{1}{10}\sin\left((\frac{4x}{L_x}-2)^2 + (\frac{4y}{L_y}-2)^2 + ct\right) \end{bmatrix}, \tag{4.2}$$

where $L_x = L_y = 2000$ km are the extents of our domain in the x- and y-directions, and $c = 5\times 10^{-6} s^{-1}$ is a phase parameter.

Here, we focus on comparing the accuracy of the two methods in reproducing this synthetic solution. In terms of efficiency, the two methods are expected to behave similarly because they share the same overall structure. They both amount to solving the same nonlinear equations at every time step, although the BE method is expected to be a little faster because it has a less complicated right-hand side.

To add some realism, the computational domain is divided into two ice-covered regions, consisting of two 750×750 km^2-squares located at two opposite corners, and a contiguous region of open water. This is illustrated in Figure 1. The open water region is masked during the numerical simulations.

Figure 2a and 2b show the u and v velocity components of this solution at time $t = 0$ on the u-grid and v-grid, respectively, over the entire domain. Figure 2c and 2d overlay the velocity field (arrows) on the u-velocity contours at times $t = 0$ and $t = 4$ days in part of the ice-covered domain as per Figure 1. The flow structure in Figure 2c and 2d, particularly, depicts large-scale features such as strong shearing

and vorticity that are relevant to sea ice flows in nature. We note that the time variation merely induces a phase shift of this basic structure.

From Figure 1, land boundaries are located along the north, south, east, and west edges of the grid. Inside the domain, the blue cells represent water, while the white cells represent ice. Two ice patches are placed within the water region as shown in the figure. Dirichlet boundary conditions are applied on land boundaries, while Neumann conditions are applied on water boundaries. To obtain the correct test solution, we set

$$\mathbf{u} = \alpha(\mathbf{x}, t) \quad \text{at land boundaries,} \tag{4.3}$$

and

$$\frac{\partial \mathbf{u}}{\partial \mathbf{n}} = \beta(\mathbf{x}, t) \quad \text{at water boundaries,} \tag{4.4}$$

where α and β are given respectively by the known solution and its derivatives determined from (4.2).

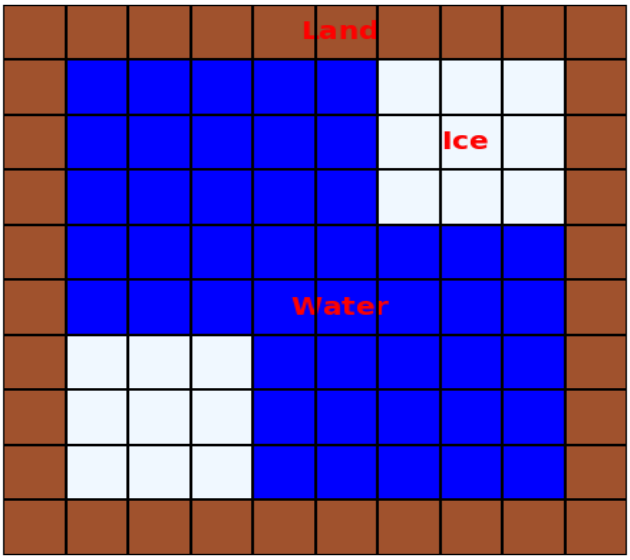
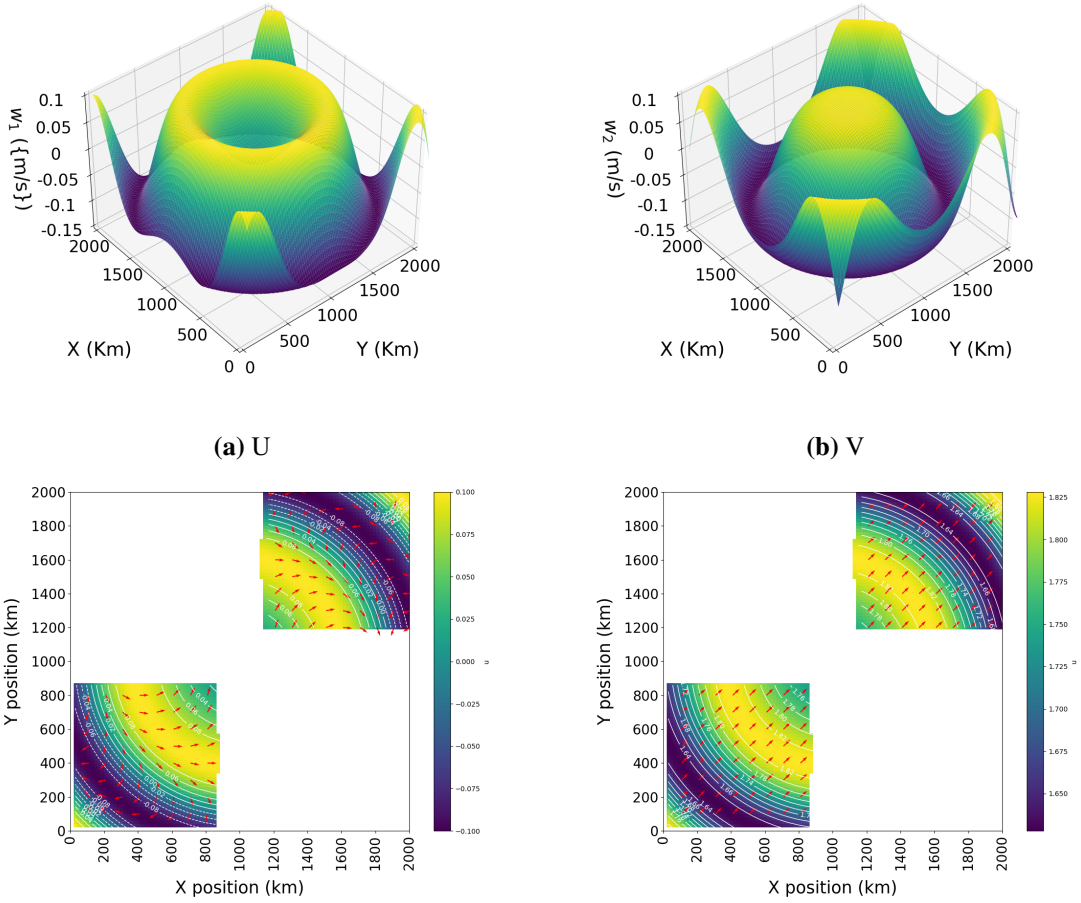


Figure 1. Schematic representation of the computational domain with land, water, and ice regions.



(a) U **(b)** V

(c) t=0 s **(d)** t=345600 s

Figure 2. Combined visualization showing: (a) and (b) the structure of the synthetic test solution on the U- and V–grids; (c) and (d): contours of the u-velocity component and flow field arrows, in the ice-covered part of the domain as per Figure 1, at $t = 0$ s and $t = 4$ days, respectively.

It remains to define the forcing terms due to wind stress and water drag, although these terms may not significantly impact the numerical scheme's performance. For completeness, we follow [17, 32] and set $\mathbf{u}_w^g$ and $\mathbf{u}_a^g$ to

$$\mathbf{u}_w^g = \begin{bmatrix} u_w^g \\ v_w^g \end{bmatrix} = \begin{bmatrix} 0.1(2y - L_y)/L_y \\ -0.1(2x - L_x)/L_x \end{bmatrix} \tag{4.5}$$

and

$$\mathbf{u}_a^g = \begin{bmatrix} u_a^g \\ v_a^g \end{bmatrix} = \begin{bmatrix} 5 + \left(\sin(\frac{2\pi t}{\theta}) - 3\right)\sin(\frac{2\pi x}{L_x})\sin(\frac{\pi y}{L_y}) \\ 5 + \left(\sin(\frac{2\pi t}{\theta}) - 3\right)\sin(\frac{\pi x}{L_x})\sin(\frac{2\pi y}{L_y}) \end{bmatrix}, \tag{4.6}$$

where θ is the wind forcing period, set to 4 days.

5. Performance under grid refinement

In this section, we present our numerical results and assess the performance of the two methods for the synthetic solution presented in the previous section. To streamline the analysis, we summarize the physical parameters used in those simulations in Table 1.

Table 1. Numerical parameters for validation simulation.

Symbol	Definition	Value
dx	spatial resolution	20 (km)
dt	Temporal resolution	10 (min)
L_x	X-extent	2000 (km)
L_y	Y-extent	2000 (km)
T	Final time	12 (hours)
ϵ	Gateaux derivative approximation parameter	10^{-6}
res_t	Residual transition	0.625
K_{max}	Maximum number of nonlinear iterations	200
f	Coriolis parameter	1.46×10^{-4} sec^{-1}

For the sake of simplicity, we chose an integration time of twelve hours to test our solver. However, it's important to note that Earth system models are typically run on much longer time scales, ranging from a few months to many years.

Using the default time step of $dt = 10$ minutes and default spatial resolution $dx = 20$ km, as listed in Table 1, we plot in Figure 3a the number of nonlinear iterations at each time step throughout the simulation period. Figure 3b shows the typical evolution of the residual associated with the JFNK (nonlinear) solver at the last time step for both the CN and BE methods. As can be seen, both methods converged in a reasonable number of iterations (8 to 10) based on the imposed tolerance error, as per (3.8). Note that this corresponds to a numerical value of $\epsilon_{tol} = 0.0602$ in the non-dimensional units of the nonlinear functional **F**. The average number of nonlinear iterations for the CN method was 9.65, while for the BE method, it was 9.75. Although the CN method suggests faster convergence, its gain isn't significant. In terms of computational complexity, the CN method has a slightly more complex right-hand side. Figure 3b illustrates the residual evolution throughout the iterative process, confirming that the CN method has a smaller residual from the beginning, giving it an advantage. This could be explained by the fact that the CN method is second-order accurate in time, providing a better starting point for the JFNK iteration (which is set to the numerical solution at the previous time step). The numerical solution is saved every two hours during the simulation. The error with respect to the synthetic solution in Figure 2 is evaluated and simultaneously outputted at both the U-grid and V-grid points. The spatial distribution of the absolute error at the end of the simulation is shown in Figure 4 for both the CN and BE methods, using the default time step and spatial grid size of $dt = 10$ minutes and $dx = 20$ km. Compared to the exact solution in Figure 2, the maximum absolute error for both methods varies between 1% to 5%, which is deemed acceptable given the high complexity of the system of equations. We also note that the error maxima are located in regions on large gradients away from the boundaries, which indicates that the boundary conditions are adequately implemented. The maximum error for the CN scheme is smaller, but the two errors remain on the same order of

magnitude. This is somewhat expected since the two methods share the same grid stencil in the spatial coordinates. For sufficiently small time steps, the error will be dominated by the spatial resolution. However, the main interest here is to determine whether the method with higher-order time accuracy will be beneficial when the time step is kept relatively large, potentially leading to overall efficiency during long-climatic simulations.

To explore this issue, we start by examining the behaviour of the numerical error, which is the difference between the exact and numerical solutions, evaluated at all spatial grid points at a given time t during the integration period. As we vary the resolution parameters dt and dx, we assess the actual rate of convergence, which is the rate at which this error decreases.

We report in Tables 2 through 3 the L_2 and L_∞ norm errors between the numerical and exact velocity vectors, evaluated on both the U-grid and the V-grid, for different time steps and spatial resolutions. The associated effective convergence rates are estimated based on Eqs (5.1) and (5.2).

Assuming that for a given dx and dt, the error norm (L_2 or L_∞, for example) takes the form $||\mathbf{E}(dx, dt)|| = O(dx^p) + O(dt^q)$, we can estimate the effective rate of convergence in the corresponding discretization (time or space) by dividing the grid parameters, dx and dt, in half and comparing the successive errors.

The formula for the effective rate of convergence in the time discretization is given by:

$$q(dx, dt) = \log\left(\frac{||\mathbf{E}(dx, dt)||}{||\mathbf{E}(dx, dt/2)||}\right) / \log(2), \tag{5.1}$$

and for the space discretization, it is:

$$p(dx, dt) = \log\left(\frac{||\mathbf{E}(dx, dt)||}{||\mathbf{E}(dx/2, dt)||}\right) / \log(2). \tag{5.2}$$

Here, $\mathbf{E} = |\mathbf{u}_{exact} - \mathbf{u}|$, where $\mathbf{u}_{exact}$ and $\mathbf{u}$ are the exact and numerical solutions, respectively. As can be seen from Tables 2 and 3, while the L_2 errors are significantly smaller than those associated with the L_∞ norm, they both decrease with decreasing spatial grid size (dx). This suggests that both methods are effective for the test case. However, there are differences in their behaviour under time step and spatial grid refinements. For instance, for all grid and time step configurations, the CN method consistently produces smaller errors than the BE method, except in rare cases. For example, on the V-grid at 10 hours with $dx = 10$ km and $dt = 5$ minutes, the BE method achieves an L_∞ norm error of 1.81×10^{-4}, while the CN method achieves 2.08×10^{-4}. Furthermore, the error discrepancies between the BE and CN methods are particularly high at coarse resolution in the time step, reaching an order of magnitude difference, especially when $dt = 20$ minutes. However, these discrepancies diminish significantly and become negligible when the time resolution is the highest ($dt = 5$ minutes). This is because, typically for numerical solutions of partial differential equations, for a fixed spatial resolution, the time discretization errors dominate when the time step is relatively large. However, when the time resolution is refined enough, the opposite happens, i.e., the spatial discretization dominates. The resolution threshold at which this transition occurs depends on the method used. In our case, while for the BE method, the time resolution error diminishes with the time step, as shown in Table 4, when the time step is varied from $dt = 20$ minutes to $dt = 5$ minutes, the error almost linearly decreases. This is expected for a first-order method. The near-linear convergence with dt of the BE can be indeed inferred from just looking at the results in Table 2. However, the equivalent errors reported in Table 3,

corresponding to the CN method, do not seem to significantly change with the time step. The time resolution errors seem to rather stagnate. See also Table 5. This error stagnation is due to the fact that even with $\Delta x = 5$ km, the error is dominated by the spatial contribution because unlike the BE method, the CN method is second-order accurate in time. The second-order convergence in space for both BE and CN is confirmed by the overall spatial convergence rates reported in Tables 6 and 7, respectively. However, the convergence rate varies non-monotonically throughout the grid refinement process, for both methods. For example, large error gains, made during the first grid refinement (going from coarse to moderate resolution), with $p \gtrsim 7$, are often compensated by slower convergence and even stagnation of errors ($p \lesssim 0$) in the subsequent grid refinement (going from moderate to fine resolution), and vice versa. To confirm this trend, we report in Tables 8 to 11 the mean rate of convergence over the three successive grids. With average rates varying between 2.80 and up 4. these tables basically confirm an overall convergence rate that exceeds the theoretically expected second order rate.

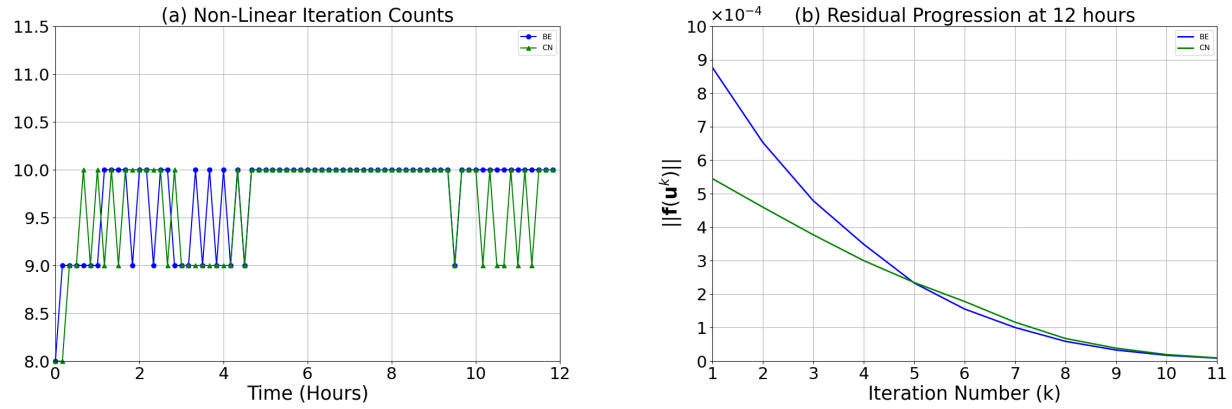


Figure 3. (a) Number of non-linear iterations associated with the JFNK solver for Crank-Nicolson and Backward Euler methods. (b) Corresponding evolution with the progression of the iteration at the last time step. The residual is measured using the L2 norm, and its units are kg·m/(m^2·s^2). The residual norm is expressed in units of the quantity $\rho h_0 f u_0$.

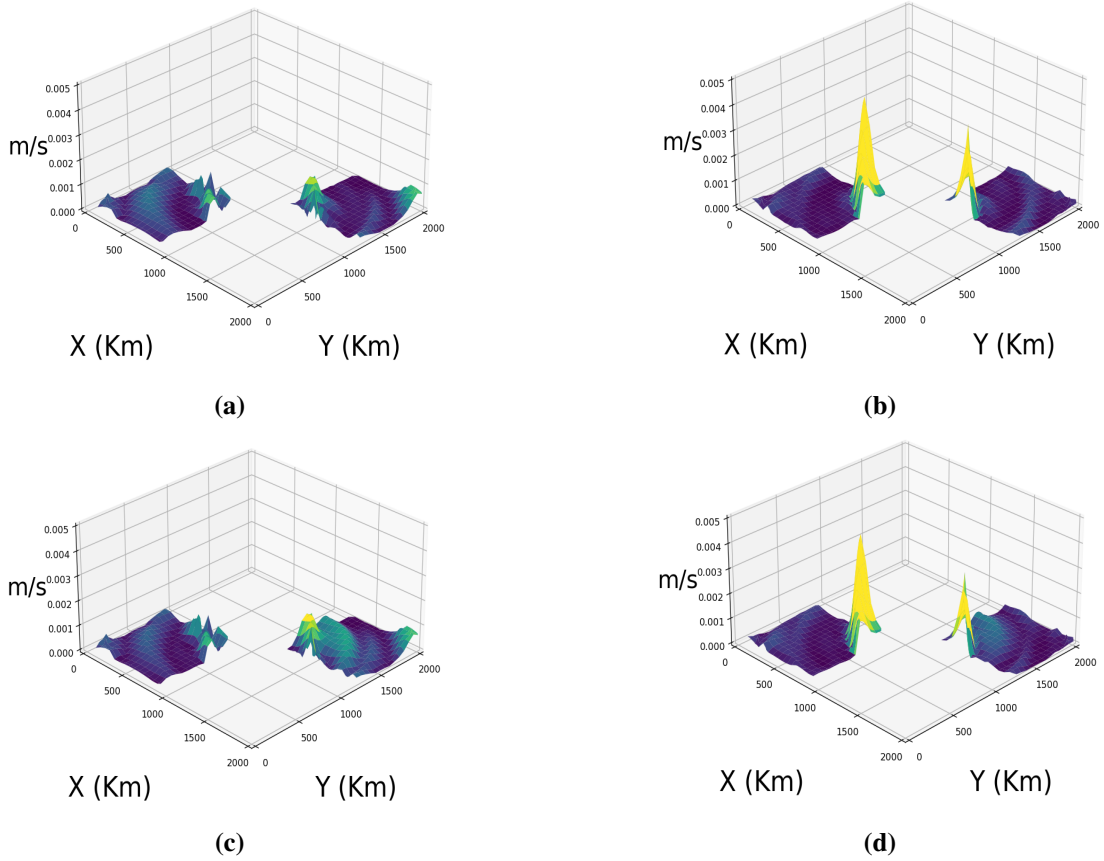


Figure 4. Absolute error for Crank-Nicolson (a and b) and the BE (c and d) methods at the end of simulation (i.e a 12 hours) for the u (a and c) and v (b and d) velocity components, as computed on the U-grid and V-grid, respectively. dx = 20 km, dt = 10 minutes.

Table 2. L_2 and L_∞ nrom errors for the u and v velocities obtained with the BE method, with respect to both the U-grid and V-grid.

	U-grid		V-grid	
Time	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$
dx=5 km and dt= 5 min				
2 hours	5.61×10^{-5}	3.38×10^{-4}	4.31×10^{-5}	3.27×10^{-4}
4 hours	5.80×10^{-5}	3.61×10^{-4}	4.05×10^{-5}	3.13×10^{-4}
6 hours	5.71×10^{-5}	3.51×10^{-4}	3.86×10^{-5}	2.96×10^{-4}
8 hours	5.48×10^{-5}	3.20×10^{-4}	3.67×10^{-5}	2.70×10^{-4}
10 hours	5.24×10^{-5}	3.16×10^{-4}	3.44×10^{-5}	2.45×10^{-4}
12 hours	5.05×10^{-5}	3.16×10^{-4}	3.17×10^{-5}	2.17×10^{-4}
dx=5 km and dt= 10 min				
2 hours	9.49×10^{-5}	7.27×10^{-4}	7.88×10^{-5}	6.49×10^{-4}
4 hours	1.01×10^{-4}	7.95×10^{-4}	7.29×10^{-5}	6.24×10^{-4}
6 hours	9.78×10^{-5}	7.53×10^{-4}	6.94×10^{-5}	5.87×10^{-4}
8 hours	9.36×10^{-5}	7.18×10^{-4}	6.49×10^{-5}	5.47×10^{-4}
10 hours	8.71×10^{-5}	6.54×10^{-4}	5.96×10^{-5}	4.91×10^{-4}
12 hours	7.89×10^{-5}	5.70×10^{-4}	5.40×10^{-5}	4.31×10^{-4}
dx= 5 km and dt=20 min				
2 hours	1.82×10^{-4}	1.84×10^{-3}	1.50×10^{-4}	1.28×10^{-4}
4 hours	2.00×10^{-4}	1.65×10^{-3}	1.41×10^{-4}	1.25×10^{-3}
6 hours	1.93×10^{-4}	1.58×10^{-3}	1.33×10^{-4}	1.18×10^{-3}
8 hours	1.81×10^{-4}	1.48×10^{-3}	1.25×10^{-4}	1.09×10^{-3}
10 hours	1.67×10^{-4}	1.35×10^{-3}	1.15×10^{-4}	9.91×10^{-4}
12 hours	1.50×10^{-4}	1.20×10^{-3}	1.03×10^{-4}	8.71×10^{-4}
dx=10 km and dt= 5 min				
2 hours	1.55×10^{-4}	1.21×10^{-3}	7.88×10^{-5}	5.60×10^{-4}
4 hours	5.89×10^{-4}	1.68×10^{-2}	7.53×10^{-4}	6.62×10^{-2}
6 hours	1.28×10^{-4}	2.57×10^{-2}	9.58×10^{-4}	2.72×10^{-2}
8 hours	1.93×10^{-3}	4.41×10^{-2}	1.29×10^{-3}	2.83×10^{-2}
10 hours	1.92×10^{-3}	4.41×10^{-2}	1.24×10^{-3}	2.39×10^{-2}
12 hours	2.94×10^{-3}	5.20×10^{-2}	1.36×10^{-3}	3.03×10^{-2}
dx=20 km and dt= 5 min				
2 hours	5.12×10^{-3}	6.98×10^{-2}	3.01×10^{-3}	4.12×10^{-2}
4 hours	4.77×10^{-3}	6.39×10^{-2}	3.76×10^{-3}	4.58×10^{-2}
6 hours	3.44×10^{-3}	4.71×10^{-2}	4.17×10^{-3}	4.99×10^{-2}
8 hours	3.37×10^{-3}	4.04×10^{-2}	5.18×10^{-3}	5.64×10^{-2}
10 hours	3.07×10^{-3}	3.81×10^{-2}	5.67×10^{-3}	6.15×10^{-2}
12 hours	3.53×10^{-3}	3.83×10^{-2}	5.35×10^{-3}	5.93×10^{-2}

Table 3. Same as Table 2, but for the CN method.

Time	U-grid		V-grid	
	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$
dx=20 km and dt= 5 min				
2 hours	2.74×10^{-3}	3.25×10^{-2}	2.35×10^{-3}	3.39×10^{-2}
4 hours	4.00×10^{-3}	4.13×10^{-2}	3.50×10^{-3}	5.16×10^{-2}
6 hours	3.29×10^{-3}	3.70×10^{-2}	3.66×10^{-3}	5.39×10^{-2}
8 hours	3.42×10^{-3}	4.01×10^{-2}	4.73×10^{-3}	5.95×10^{-2}
10 hours	3.44×10^{-3}	4.05×10^{-2}	5.07×10^{-3}	6.19×10^{-2}
12 hours	3.79×10^{-3}	4.29×10^{-2}	5.25×10^{-3}	6.14×10^{-2}
dx=10 km and dt=5 min				
2 hours	1.61×10^{-4}	1.21×10^{-3}	6.48×10^{-5}	5.54×10^{-4}
4 hours	1.02×10^{-3}	2.16×10^{-2}	1.30×10^{-3}	3.56×10^{-2}
6 hours	2.94×10^{-3}	5.79×10^{-2}	1.09×10^{-3}	2.85×10^{-2}
8 hours	3.21×10^{-3}	6.33×10^{-2}	1.05×10^{-3}	2.50×10^{-2}
10 hours	3.50×10^{-3}	6.47×10^{-2}	1.44×10^{-3}	2.12×10^{-2}
12 hours	2.88×10^{-3}	5.82×10^{-2}	1.11×10^{-3}	2.17×10^{-2}
dx=5 km and dt=5 min				
2 hours	5.09×10^{-5}	3.26×10^{-4}	2.09×10^{-5}	1.78×10^{-4}
4 hours	5.12×10^{-5}	3.26×10^{-3}	2.12×10^{-5}	1.89×10^{-4}
6 hours	5.14×10^{-5}	3.24×10^{-4}	2.13×10^{-5}	1.98×10^{-4}
8 hours	5.17×10^{-5}	3.23×10^{-4}	2.15×10^{-5}	2.06×10^{-4}
10 hours	5.20×10^{-5}	3.19×10^{-4}	2.15×10^{-5}	2.11×10^{-4}
12 hours	5.22×10^{-5}	3.18×10^{-4}	2.17×10^{-5}	2.15×10^{-4}
dx=5 km and dt=20 min				
2 hours	4.19×10^{-5}	3.30×10^{-4}	1.70×10^{-5}	1.80×10^{-4}
4 hours	4.25×10^{-5}	3.28×10^{-4}	1.73×10^{-5}	1.89×10^{-4}
6 hours	4.27×10^{-5}	3.26×10^{-4}	1.74×10^{-5}	1.97×10^{-4}
8 hours	4.24×10^{-5}	3.25×10^{-4}	1.76×10^{-5}	2.04×10^{-4}
10 hours	4.27×10^{-5}	3.20×10^{-4}	1.76×10^{-5}	2.10×10^{-4}
12 hours	4.32×10^{-5}	3.18×10^{-4}	1.76×10^{-5}	2.10×10^{-4}
dx=5 km and dt=10 min				
2 hours	4.21×10^{-5}	3.27×10^{-4}	1.73×10^{-5}	1.79×10^{-4}
4 hours	4.25×10^{-5}	3.27×10^{-4}	1.75×10^{-5}	1.89×10^{-4}
6 hours	4.25×10^{-5}	3.24×10^{-4}	1.76×10^{-5}	1.98×10^{-4}
8 hours	4.28×10^{-5}	3.23×10^{-4}	1.78×10^{-5}	2.06×10^{-4}
10 hours	4.29×10^{-5}	3.19×10^{-4}	1.78×10^{-5}	2.10×10^{-4}
12 hours	4.32×10^{-5}	3.18×10^{-4}	1.78×10^{-5}	2.15×10^{-4}

Table 4. Estimated convergence rates with respect to time step refinement in the L_2 and L_∞ norm errors in Table 2, for the BE method, with dx = 5 km.

	U-grid				V-grid			
	20 min → 10 min		10 min → 5 min		20 min → 10 min		10 min → 5 min	
Time	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$
2 hours	0.94	1.00	0.75	1.10	0.94	0.98	0.86	0.98
4 hours	0.97	1.05	0.80	1.13	0.95	1.00	0.84	0.99
6 hours	0.98	1.07	0.77	1.10	0.94	1.00	0.84	0.98
8 hours	0.95	1.04	0.77	1.16	0.95	1.00	0.82	1.00
10 hours	0.94	1.05	0.73	1.04	0.95	1.00	0.78	1.00
12 hours	0.93	1.07	0.65	0.85	0.93	1.00	0.76	0.98

Table 5. Estimated convergence rates with respect to time step refinement in the L_2 and L_∞ norm errors in Table 3, for the CN method, with dx = 5 km.

	U-grid				V-grid			
	20 min → 10 min		10 min → 5 min		20 min → 10 min		10 min → 5 min	
Time	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$
2 hours	-0.006	0.009	-0.012	0.008	-0.020	0.003	-0.010	0.008
4 hours	0.001	0.006	-0.007	0.001	-0.020	0.003	-0.015	-0.005
6 hours	0.008	0.009	-0.011	0.00009	-0.016	-0.003	-0.010	-0.006
8 hours	-0.01	0.006	-0.009	0.001	-0.017	-0.012	-0.008	0.0008
10 hours	-0.007	0.005	-0.010	-0.001	-0.020	0.005	-0.007	-0.006
12 hours	0.0004	0.019	-0.010	0.0001	-0.010	-0.003	-0.020	0.0001

Table 6. Same as Table 4 but for the spatial grid refinement and dt = 5 minutes.

Methods	U-grid				V-grid			
and	20 km → 10 km		10 km → 5 km		20 km → 10 km		10 km → 5 km	
Time	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$
2 hours	5.03	5.84	1.21	1.84	5.26	6.20	0.60	0.77
4 hours	3.01	1.92	3.01	5.53	2.32	0.80	3.93	6.39
6 hours	1.41	0.87	4.23	6.19	2.12	0.87	4.36	6.52
8 hours	0.80	-0.12	4.87	7.10	2.00	1.00	4.88	6.71
10 hours	0.67	-0.21	4.93	7.12	2.18	1.35	4.91	6.61
12 hours	0.26	-0.43	5.61	7.36	1.96	0.94	5.16	7.14

Table 7. Estimated convergence rates for spatial grid refinement associated with the L_2 and L_∞ norm errors in Table 3 for Crank-Nicolson method: dt = 5 minutes.

Methods	U-grid				V-grid			
and	20 km → 10 km		10 km → 5 km		20 km → 10 km		10 km → 5 minkm	
Time	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$
2 hours	4.08	4.74	1.66	1.89	5.18	5.93	1.62	1.63
4 hours	1.97	0.93	4.31	6.05	1.40	0.53	5.95	7.55
6 hours	0.16	-0.64	5.83	7.48	1.73	0.91	5.68	7.16
8 hours	0.09	-0.65	5.95	7.61	2.16	1.25	5.62	6.92
10 hours	-0.02	-0.67	6.07	7.66	1.81	1.54	6.06	6.65
12 hours	0.39	-0.43	5.78	7.51	2.23	1.79	5.67	6.36

Table 8. Average convergence rates for spatial grid refinement based on the L_2 and L_∞ norm errors reported in Table 2 for the backward Euler method (Δt = 5 minutes) on the U-grid.

Methods	U-grid				Averages	
and	20 km → 10 km		10 km → 5 km		Mean $\|\|\mathbf{E}\|\|_{L_2}$	Mean $\|\|\mathbf{E}\|\|_{L_\infty}$
Time	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$		
2 hours	5.03	5.84	1.21	1.84	3.12	3.84
4 hours	3.01	1.92	3.01	5.53	3.01	3.72
6 hours	1.41	0.87	4.23	6.19	2.82	3.53
8 hours	0.80	-0.12	4.87	7.10	2.83	3.49
10 hours	0.67	-0.21	4.93	7.12	2.80	3.45
12 hours	0.26	-0.43	5.61	7.36	2.93	3.46

Table 9. Average convergence rates for spatial grid refinement based on the L_2 and L_∞ norm errors reported in Table 2 for the backward Euler method (Δt = 5 minutes) on the V-grid.

Methods	V-grid				Averages	
and	20 km → 10 km		10 km → 5 km		Mean $\|\|\mathbf{E}\|\|_{L_2}$	Mean $\|\|\mathbf{E}\|\|_{L_\infty}$
Time	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$		
2 hours	5.26	6.20	0.60	0.77	2.93	3.48
4 hours	2.32	0.80	3.93	6.39	3.12	3.59
6 hours	2.12	0.87	4.36	6.52	3.24	3.69
8 hours	2.00	1.00	4.88	6.71	3.44	3.85
10 hours	2.18	1.35	4.91	6.61	3.54	3.98
12 hours	1.96	0.94	5.16	7.14	3.56	4.04

Table 10. Average convergence rates for spatial grid refinement based on the L_2 and L_∞ norm errors reported in Table 3 for the Crank–Nicolson method (Δt = 5 minutes) on the U-grid.

Methods	U-grid				Averages	
and	20 km → 10 km		10 km → 5 km		Mean $\|\|\mathbf{E}\|\|_{L_2}$	Mean $\|\|\mathbf{E}\|\|_{L_\infty}$
Time	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$		
2 hours	4.08	4.74	1.66	1.89	2.87	3.31
4 hours	1.97	0.93	4.31	6.05	3.14	3.49
6 hours	0.16	-0.64	5.83	7.48	3.99	3.42
8 hours	0.09	-0.65	5.95	7.61	3.02	3.48
10 hours	-0.02	-0.67	6.07	7.66	3.03	3.50
12 hours	0.39	-0.43	5.78	7.51	3.08	3.54

Table 11. Average convergence rates for spatial grid refinement based on the L_2 and L_∞ norm errors reported in Table 3 for the Crank–Nicolson method (Δt = 5 minutes) on the V-grid.

Methods	V-grid				Averages	
and	20 km → 10 km		10 km → 5 km		Mean $\|\|\mathbf{E}\|\|_{L_2}$	Mean $\|\|\mathbf{E}\|\|_{L_\infty}$
Time	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$	$\|\|\mathbf{E}\|\|_{L_2}$	$\|\|\mathbf{E}\|\|_{L_\infty}$		
2 hours	5.18	5.93	1.62	1.63	3.40	3.78
4 hours	1.40	0.53	5.95	7.55	3.68	4.04
6 hours	1.73	0.91	5.68	7.16	3.71	4.04
8 hours	2.16	1.25	5.62	6.92	3.89	4.09
10 hours	1.81	1.54	6.06	6.65	3.94	4.10
12 hours	2.23	1.79	5.67	6.36	3.95	4.08

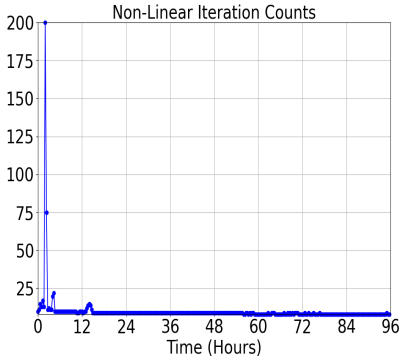
6. Behaviour at coarse time resolution and long time integration

In this section, we fix the spatial discretization at dx = 10 km and run both methods for up to 4 days, with increasingly larger and larger time steps. We consider the cases with dt = 20 minutes, 40 minutes, and 80 minutes, that are in the range of time steps used in climate simulations. We note that because the two methods are in theory conditionally stable, the expectation is that our codes could be run with an arbitrarily large time step. However, this is only true for linear equations such as the heat equation. The nonlinearity may lead to surprises as the the Newton iterations may not always converge.

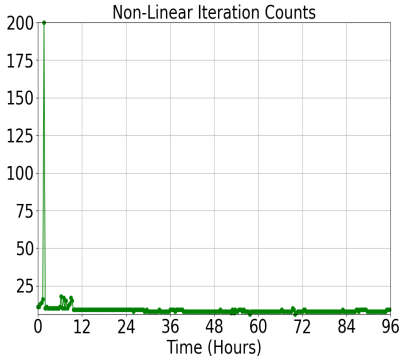
We plot in Figure 5 the number of nonlinear iterations per time step as a function of the integration time for the aforementioned grid configurations.

Figure 5 shows that both methods run until the end of the simulation (i.e., for 4 days), despite the fact that the Newton iteration did not always converge, as the maximum iteration number, set to 200 iterations, is reached multiple times. For the backward Euler (BE) method, we observe the following: With a time step of 20 minutes, the method failed to converge only at the seventh time step; with dt = 40 minutes, it failed to converge at time steps 3, 4, 5, and 6; and with dt = 80 minutes, the method did not converge at the first and second time steps, resumed convergence until time step 14, and then failed to

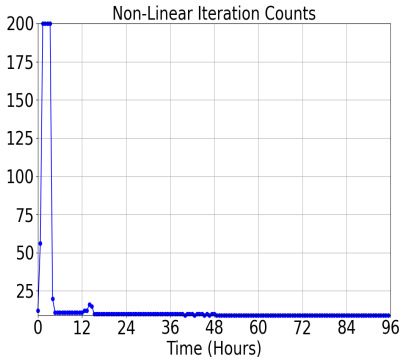
converge again until the end of the simulation. For the CN method, at the finest resolution (i.e., $dt = 20$ minutes), the Newton iteration diverges only at the sixth time step. At the moderate resolution (i.e., $dt = 40$ minutes), divergence starts at the third time step and continues until step 15, then convergence resumes until step 24. The method diverges again at step 25, converges at step 26, and then diverges once more until the end of the simulation. At the coarse resolution (i.e., $dt = 80$ minutes), the Newton iteration starts diverging from the fifth time step and continues to diverge until the end. This suggests that to guarantee convergence of the JFNK solver, a time step smaller than 20 minutes is needed. To gain further insight, we plot in Figure 6 the solution at time $t = 4$ days corresponding to the six cases in Figure 6. Compared to the exact solution in Figure 2d, we observe that both the BE and CN methods provide fairly accurate approximations at fine time steps $dt = 20, 40$ minutes (Figure 6a-d). However, at $dt = 80$ minutes, the numerical solutions in Figure 6e and 6f show significant degradation, with the CN solution being particularly poor. In fact, the CN solution exhibits excessive grid-scale oscillation, a hallmark of numerical instability. Interestingly, the fact that the BE method is L-stable helped maintain the stability of the numerical solution at coarse time steps.



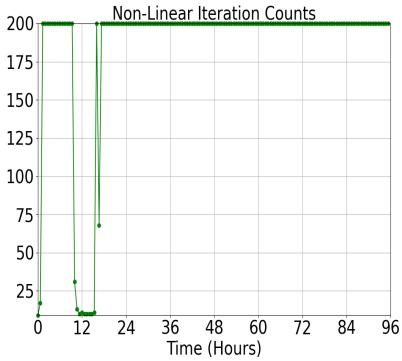
(a) BE: dt = 20 minutes



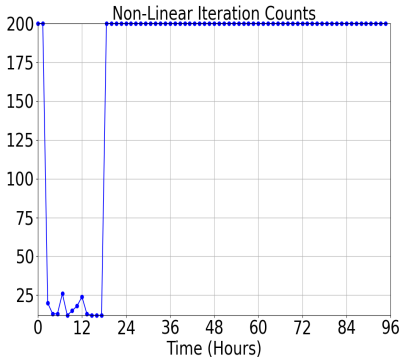
(b) CN: dt = 20 minutes



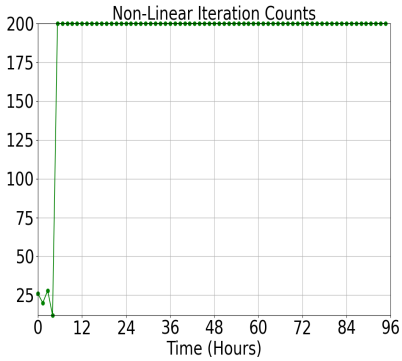
(c) BE: dt = 40 minutes



(d) CN: dt = 40 minutes

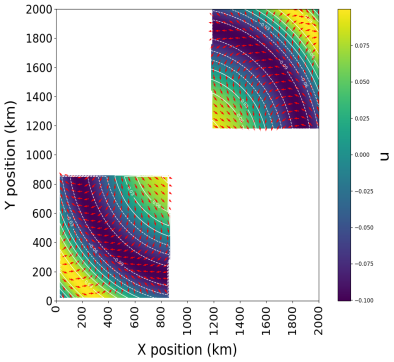


(e) BE: dt = 80 minutes

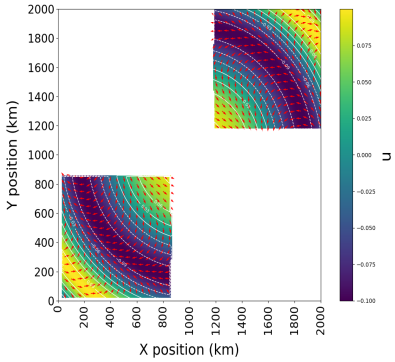


(f) CN: dt = 80 minutes

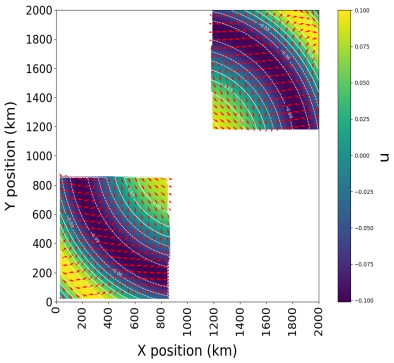
Figure 5. Time evolution of nonlinear iteration count for BE (a, c, and e) and CN (b, d, and f) methods when $dx = 10$ km and $dt = 20, 40, 80$ minutes, demonstration integration time until divergence.



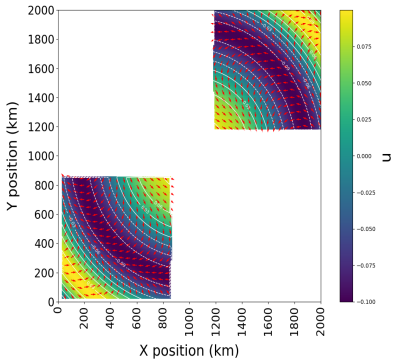
(a) BE on U-grid: dt = 20 minutes



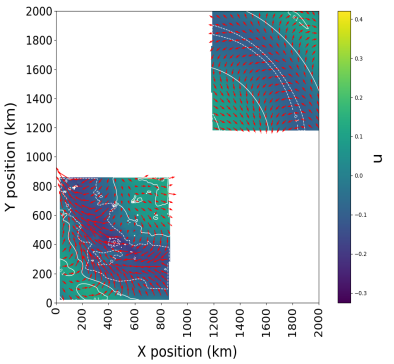
(b) CN on U-grid: dt = 20 minutes



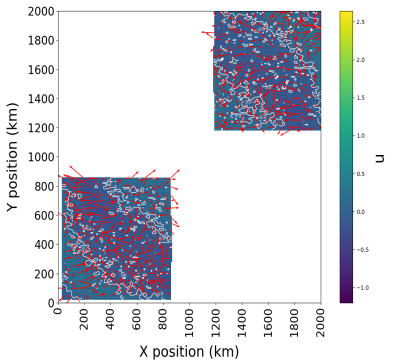
(c) BE on U-grid: dt = 40 minutes



(d) CN on U-grid: dt = 40 minutes



(e) BE on U-grid: dt = 80 minutes



(f) CN on U-grid: dt = 80 minutes

Figure 6. Approximate solutions of the u and v fields computed using the BE and CN methods for a fixed spatial step dx = 10 km and time steps dt = 20, 40, and 80 minutes at the end of day 4.

7. Climatic simulation with the coupled system

In this section, we couple the sea ice momentum equations (2.1) with the continuity equations (2.2) and (2.3), which govern the ice thickness h and the ice compactness (or area fraction) A. For the sake of simplicity, the thermodynamic effects of freezing and melting are ignored. A meaningful approach

for including these effect would require the inclusion of a heat budget equation.

The continuity equations (2.2) and (2.3) are discretized using a nonoscillatory central differencing scheme [28], where h and A are forced to satisfy $h \geq 0$ and $0 \leq A \leq 1$ throughout the simulation. Homogeneous Dirichlet boundary conditions are used at both the open-water and land boundaries for h and A ($h = A = 0$). Moreover, the same validation domain in Figure 1, is used, but now the ice is allowed to move.

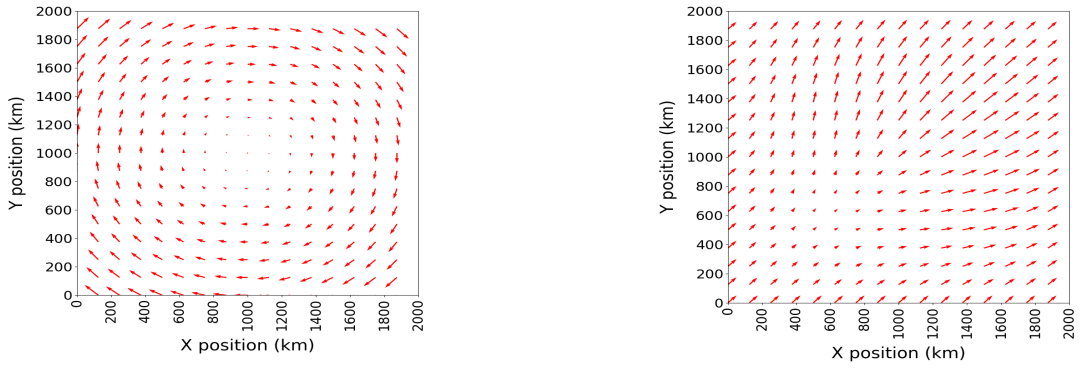
A level-set technique, where a distance function $\phi(x, y, t)$ that delimits the ice region, is employed at every time step:

$$
\begin{aligned}
&|\nabla\phi| = 1, (x, y, t) \in \Omega_t U \Omega_t^C, \\
&\phi(x) = 1, (x, y, t) \in \Omega_t - \partial\Omega_t, \\
&\phi(x, y, t) = 0, (x, y) \in \partial\Omega_t, \\
&\phi(x, y, t) = 0 (x, y) \in \Omega_t^c - \partial\Omega_t,
\end{aligned}
$$

where Ω_t is the region where ice is present. The distance function ϕ defines our computational domain with a moving boundary. The details of this technique, including a method of solving it using time marching to steady steady [33], are found in [31].

In a nutshell, Ω_t is defined as the region where both h and A are positive and can be thought of as the mask over the region covered by ice. Sea ice models used for climate predictions use discrete masks based often based on the ice volume threshold. For instance the Los Alamos Sea Ice Model (CICE) uses a combined mass/area cut-off to set the ice extent region [14]. The distance function allows a more effective definition of the ice extent region at the subgrid level as moving-boundary domain, which is extremely useful given that sea ice models are run at coarse resolutions ranging from 10 km to 100 km. This information is then employed to correct the integration volumes and properly enforce the Neumann boundary condition at the open-water boundary points in (4.2). The level-set method will also be useful in the future when thermodynamics are included.

Initially, at $t = 0$, the ice is assumed to be at rest ($u = 0 \ \ m/s$), the concentration is set to $A = 1$ where ice is present, and the initial thickness field is uniform with $h = 1$ m, over the ice region. The ice field is then allowed to drift and evolve subject to the wind and ocean stresses defined in Eqs (2.5)–(2.7) and (4.6). See also Figure 7.



(a) **(b)**

Figure 7. Geostrophic ocean and wind forcing fields at t = 3 days, showing (a) ocean forcing and (b) wind forcing.

In this simulation, we use the same numerical parameters as those reported in Table 1 for the validation simulation, with some modifications. The simulation is performed over a full year (i.e., $T = 365$ days), with a time resolution of $dt = 1$ hour and a spatial resolution of $dx = 20$ km. Additionally, for the nonlinear iteration process, a maximum of 200 iterations is imposed, while a minimum of 5 iterations is enforced. A new stopping criterion (7.1) is applied because the termination criterion used in the validation case is not suitable for this simulation [31].

$$0.997 < \frac{||F(u^k)||}{||F(u^{k-1})||} \leq 1. \tag{7.1}$$

This criterion forces the iteration to stop where the residual stagnates.

The computational performance of this simulation is lower than that of the validation simulation, mainly due to the additional equations for the conservation of h and A and the distance function. However, the solver demonstrates robust behavior for both the BE and CN methods when accounting for the added complexity of the full system.

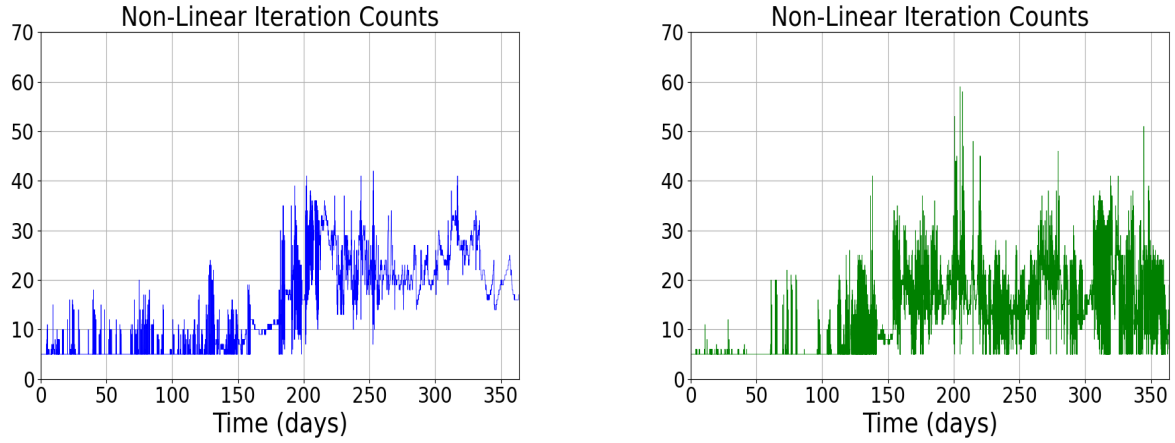
Figure 8 shows the nonlinear iterations for the BE and CN methods. In Figure 8a, the number of nonlinear iterations for the BE method ranges between 5 and 45, while in Figure 8b, for the CN method, it ranges between 5 and 60. The average number of iterations is 14.34 for the BE method and 12.83 for the CN method, with standard deviations of 9.20 and 8.20, respectively. This is consistent with the result in the previous section that showed that overall the CN method has overall less nonlinear iteration, even though Figure 8b shows that the CN methods has outliers to up to 60 iterations in a time step.

Figure 9 shows the residuals computed using the L^2 norm for the BE method Figure 9a and the CN method Figure 9b. This demonstrates that while the BE method seems to be more stable, both methods have overall comparable and, so far, acceptable behaviour.

Our solver performs well for both methods, as it consistently produces smooth velocity solutions at every time step, without any discernible numerical oscillations, as shown in Figure 10. The figure presents the solution after 4 days, after 140 days, and at the end of the year for both the BE (Figure 10a,c,e) and CN (Figure 10b,d,f) methods. While the exact solution is not know, it is worth noting that the two solutions are, to some extent, consistent with one another, and that the velocity field is overall in agreement with a forced drift resulting from a combination of the wind and ocean current velocities in Figure 7. While the ocean current is a standing gyre swirling the whole basin, the wind overall directs the ice towards the upper-right corner of the domain.

Figure 11 shows a 3D-plot of the ice thickness at the end of 4 days (Figure 11a for the BE method and Figure 11b for the CN method), at the end of 140 days (Figure 11c for the BE method and Figure 11d for the CN method), and at the end of the year (Figure 11e for the BE method and Figure 11f for the CN method).

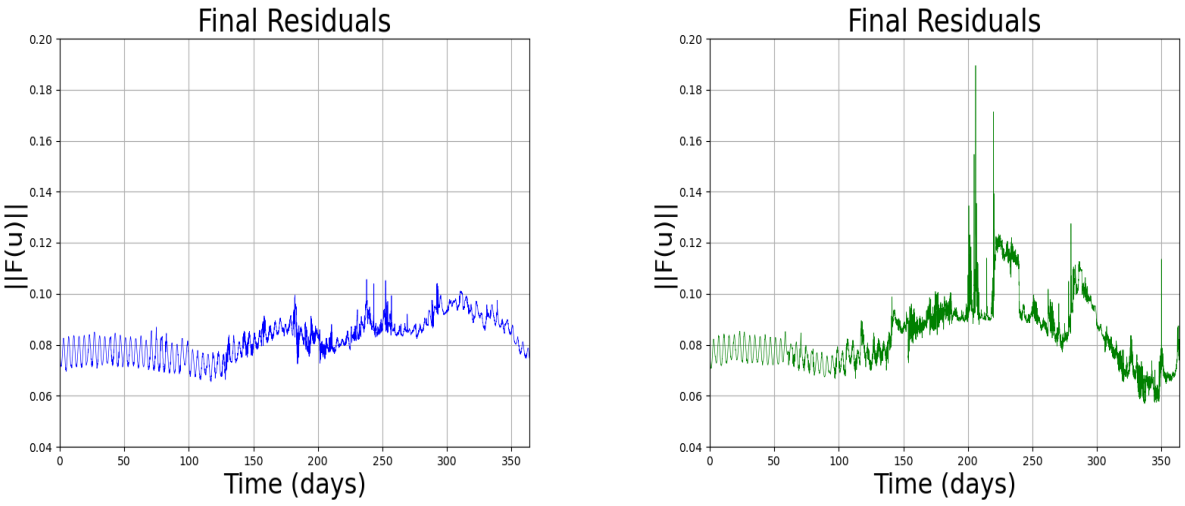
We note that from Figure 11, the two methods remain consistent up to 140 days. At the end of the year, the CN method displays a more smeared ice field while the BE method's solution remain mainly concentrated along the main diagonal. We recall that at the land boundaries, determined by the square basin extent, the ice velocity is set to zero, so no mass is advected past this boundary. In a more realistic setting, one could allow ice to tip-over outside the ocean when the ice thickness becomes sufficiently high near the land boundary.



(a) Nonlinear iterations using the BE method

(b) Nonlinear iterations using the CN method

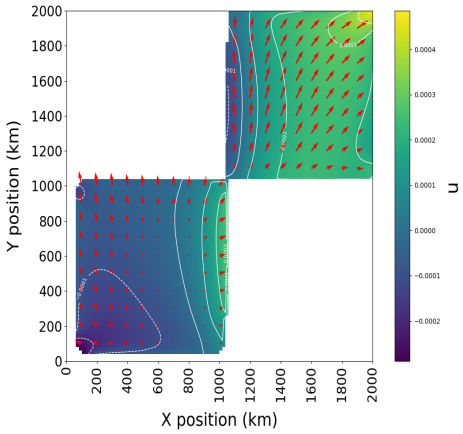
Figure 8. Comparison of nonlinear iterations for the BE and CN methods over the entire year.



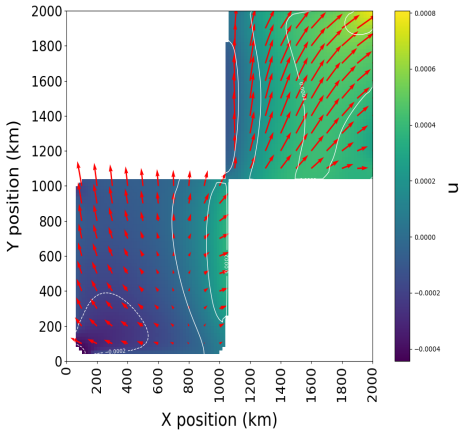
(a) Final nonlinear residuals for the BE method

(b) Final nonlinear residuals for the CN method

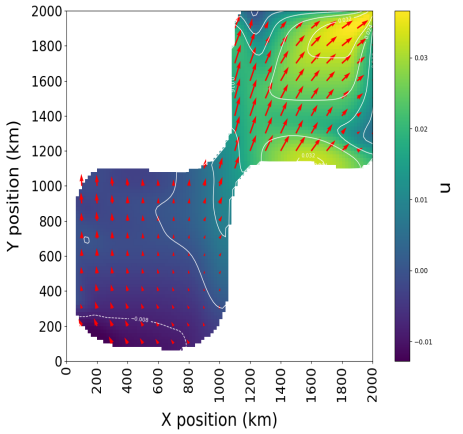
Figure 9. Final nonlinear residual comparison (L^2 norm) for the BE and CN methods over the entire year.



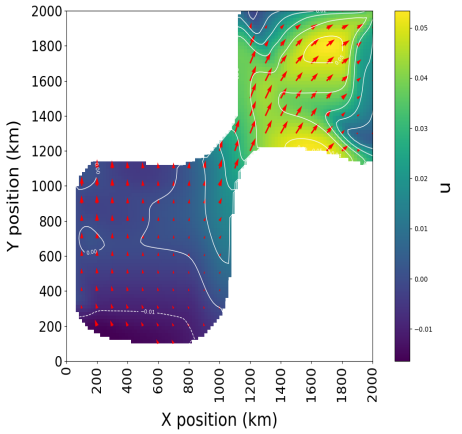
(a) BE velocity at the end of day 4



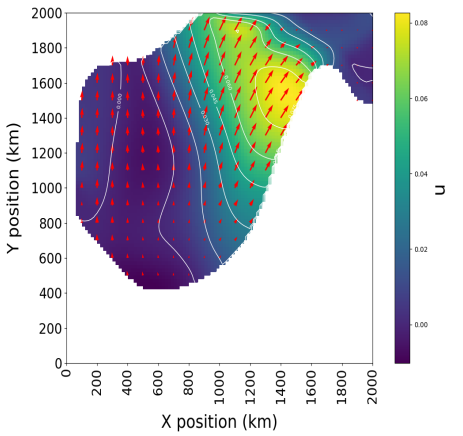
(b) CN velocity at the end of day 4



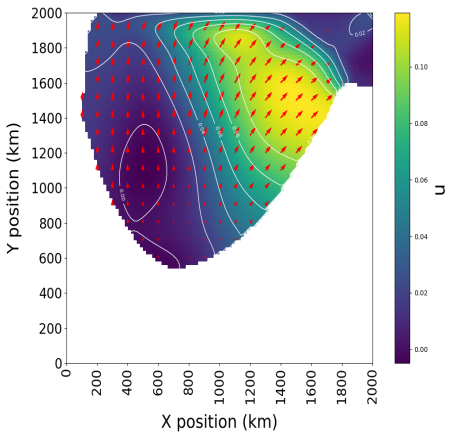
(c) BE velocity at the end of day 140



(d) CN velocity at the end of day 140

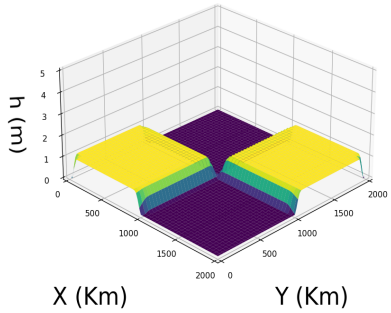


(e) BE velocity at year-end

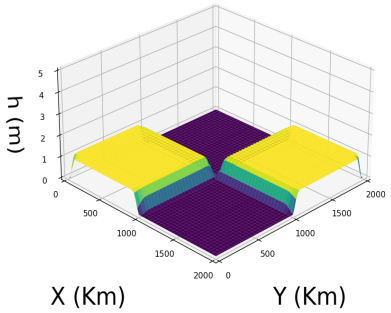


(f) CN velocity at year-end

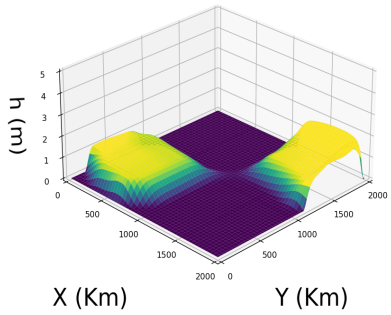
Figure 10. Arrows of velocity (u and v) fields on top of u color contours, computed using the BE and CN methods, with a spatial resolution of $dx = 20$ km and a time step of $dt = 1$ h, shown at the end of day 4, day 140, and at year-end.



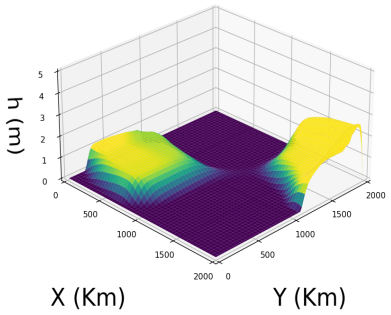
(a) BE thickness at the end of day 4



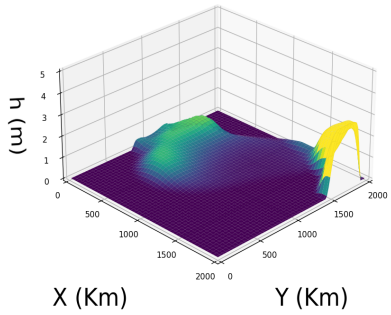
(b) CN thickness at the end of day 4



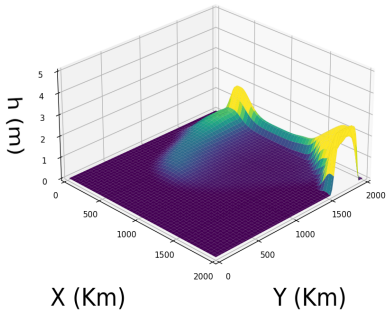
(c) BE thickness at the end of day 140.



(d) CN thickness at the end of day 140.



(e) BE thickness at year-end.



(f) CN thickness at year-end.

Figure 11. Ice thickness computed using the BE and CN methods at the end of day 4, day 140, and year-end.

In the current setting, the ice mass is conserved in principle. However, the numerical scheme employed to advect the ice thickness and ice compactness results in volume loss due in large part to known diffusivity of the central scheme. To assess the magnitude of the loss, we calculated the total volume, $V = \sum_{\text{all grid points },i} A_i h_i$, at each time step. Figure 12 shows its progression in all simulations. While the total loss is substantial, it is worth noting that this occurs over 8,736 time steps, with an average loss per time step of 0.016% for the BE method and 0.019% for the CN method coupled with

the central scheme, i.e.,

$$\frac{V_{BE}^{n+1}}{V_{BE}^{n}} = 0.99984, \quad \frac{V_{CN}^{n+1}}{V_{CN}^{n}} = 0.99981,$$

where V^n is the total volume at time level n.

We note that because loss of mass is mainly caused by the central scheme, which is common to both methods, the total loss remains comparable. However, because the CN method tends to smear the ice field, towards the end of the simulation (Figure 11), the overall volume loss is more significant for this method, as the area over which numerical dissipation occurs is larger. In fact, the two curves in Figure 12 remain close to each other until about 100 days (Figure 10).

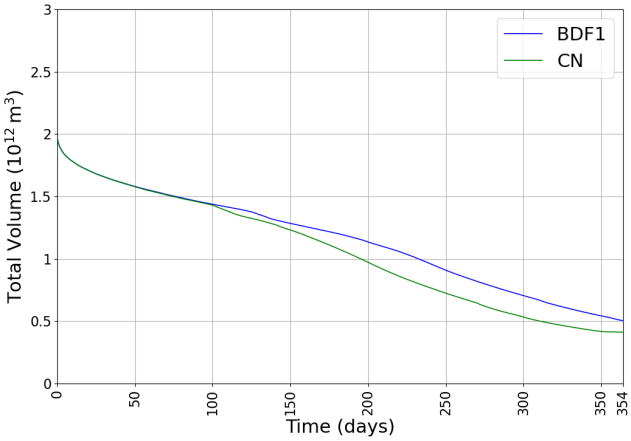


Figure 12. Evolution of the total volume at each time step.

A volume loss of nearly 50% due solely to numerical dissipation is indeed enormous, and this issue must be addressed in sea ice modeling. The transport of tracers in operational models like CICE use upwind schemes [14] that also have significant numerical dissipation [8], and to the authors knowledge, the volume loss issue in these models has not been documented. The loss in volume due to numerical dissipation may have been masked by changes due to melting and freezing. The issue can be addressed through the use of high-order discretely conservative methods [10].

To assess the fidelity of the tanh regularization in representing the viscous-plastic rheology, we can verify to what extent the computed stress invariants remain within the elliptic curve. In the ideal plastic flow regime, the two invariant coordinates must lie on the ellipse while during the purely viscous regime, the stress point is contained within the ellipse interior.

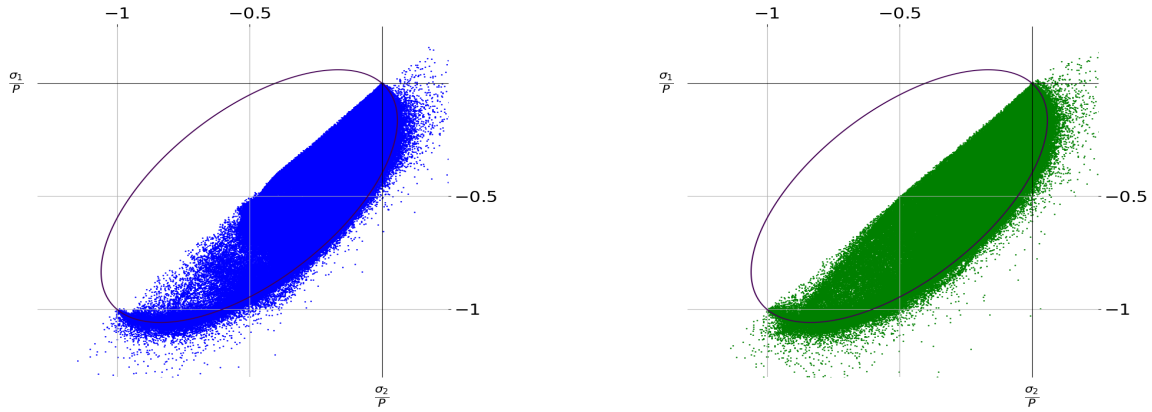
To plot the VP yield curve (Figure 13), we adopted a method commonly used in the sea ice community [17, 20, 31]. In terms of the principle stress, the elliptic yield curve reads as follows:

$$Y(\sigma_1, \sigma_2) = \left(\frac{\sigma_1 + \sigma_2 + P}{P}\right)^2 + \left(\frac{\sigma_2 - \sigma_1}{P} e\right)^2 - 1 = 0, \tag{5.8}$$

where σ_1 and σ_2 are the principal components of the stress tensor $\boldsymbol{\sigma}$. A true VP solution, referred to as VP convergence, is achieved when stress states lie on or within the yield curve. Stress states inside the curve correspond to viscous behavior, while those on the curve indicate plastic behavior. States outside the curve are nonphysical and reflect model inaccuracies [20].

Figure 13 shows a scatter plot of the stress states for the BE method (a) and the CN method (b). These stress states were recorded every 91 time steps, at all points in the ice domain where the ice velocity is non-zero. This is done to avoid points with zero or very weak stresses resulting in round-off errors when computing the principle components. The counting process results in a total of 836,759 data points for the BE method and 828,302 for the CN method.

VP convergence requires $Y(\sigma_1, \sigma_2) \leq \delta$ to account for numerical noise near the yield curve. Following [20], we use $\delta = 0.005$, and convergence is achieved when 99% of the stress states satisfy this condition. In our simulations, the BE method had 786,126 points (93.95%) satisfying the VP criterion, while 50,633 points (6.05%) did not. For the CN method, 776,016 points (93.69%) met the criterion, and 52,286 points (6.31%) failed to satisfy it. Once again, while the BE method seem to present a lower number of stress points outside the yield domain, the two methods remain comparable and both are acceptable in this regard.



(a) **(b)**

Figure 13. Stress states obtained using the BE method (a) and the CN method (b).

8. Conclusions

In almost all climate models, sea ice is represented as a viscous-plastic (VP) flow with a highly nonlinear rheology consisting of an elliptic-yield-curve constitutive law. The dynamics of sea ice remains one of the most uncertain factors in the ability of the Earth system model to address the problem of climate change. The difficulty in solving nonlinear partial differential equations accurately and efficiently is believed to contribute significantly to this uncertainty.

In this study, we investigated the numerical resolution of the sea ice momentum equation. We tested the first-order implicit scheme BE on a synthetic test problem. We chose the implicit method for stability reasons, as explicit schemes for the SIME with VP rheology require very small time steps. The BE methods belongs to the family of backward differentiation formula (BDF) methods. BDF methods are known to have better A-stability properties than Crank-Nicolson and Range-Kutta type methods in general. Therefore, the solver presented and validated here aims to find methods with improved stability compared to the work of [32]. This is particularly important because the sea-ice rheology makes the equations very stiff and solutions highly irregular.

The solver performed quite well. When using the default resolutions of 10 km and 5 minutes, the maximum number of nonlinear iterations per time step did not exceed 11 (the mean is 9.70 with a standard deviation of 0.56) during a 7-day simulation (results not shown). In contrast, the Crank-Nicolson scheme converged in a maximum of 10 nonlinear iterations per time step. The mean iteration count to reach convergence is 9.27, and the standard deviation is 0.45. We note in Figure 4 that the maximum absolute error is concentrated in regions of large gradients, as expected, contrary to what was seen in [32], where larger errors occur in the middle of the domain due to the use of ice-water boundaries.

In Table 4, we observe approximately first-order convergence in time. However, in Tables 6 and 7, the results are less conclusive, suggesting that second-order convergence is not consistently achieved, although it remains close to that observed in Tables 8 to 11. This issue is also found in [32] and is due to the nonlinearity of the SIMEs.

The rate of convergence did not appear to as expected, but it tends to improve with grid resolution, as shown in Tables 2 and 3. Perhaps the problem is that the resolution is still not fine enough to observe the expected convergence rate at the early times of the simulation.

The 4-day simulations with $dx = 10$ km, as reported in Section 6, indicate that the solutions remained stable for the most part throughout the time period; the CN solution significantly deteriorated at the coarse timestep $dt = 80$ minutes. This deterioration is likely caused by numerical oscillation inherent to the second-order scheme, as seen in Figure 6f. The nonlinear solver stopped to converge beyond the first 6 hours, as indicated in Figure 5, but this may not be the sole cause. A similar behaviour is seen in both the BE solution at $dt = 40$ minutes and the CN solution with $dt = 80$ minutes, without the same repercussion on the solution at time $t = 4$ days. While the performance of the two methods may differ beyond 4 days, the present results suggest that the BE method is preferred at coarse resolution.

Tables 2 and 3 also show that the overall error decreases when passing from one coarser resolution to a finer one. However, this peculiar behaviour may be a signature of an ill-posedness of the VP model of sea ice [11]. Despite the arctan regularization used here, the nonlinear system may remain ill-posed [5, 6]. As demonstrated in [5], large gradients may result in loss of parabolicity, which in turn may lead to an ill-conditioned problem. As such, the Jacobian associated with our JFNK solver is potentially ill-conditioned, which may partly explain the results. Being L-stable, the BE method is able to artificially restore stability. However, for high-order methods to effectively work, the JFNK solver many need to be supplemented with a preconditioning step. Unfortunately, attempts of employing preconditionning to the linear solver didn't lead to noticeable improvement, and more elaborate methods of dealing directly with the non-linear solver are needed [4]. The implementation of a nonlinear preconditioning will be the subject of future research.

In realistic sea ice simulations, the momentum equations in (2.1), the focus of this study, are coupled to (2.2) and (2.3), which respectively conserve ice thickness and ice area coverage. The methods discussed here were further compared in terms of their ability to simulate sea ice dynamics when these three equations are coupled using a time splitting strategy, as outlined in [31]. In these coupled simulations, the external forcing is applied through wind and ocean stresses, τ_a and τ_w, respectively. The geostrophic ocean and wind currents, u_w^g and u_g^a, in (2.5) and (2.6), respectively, are prescribed based on observations. The results of these coupled simulations are reported in Section 7.

The two methods are run at a relatively fine resolution of 20 km, for which both methods appeared

to behave well in Section 6.

The conservation equations (2.2) and (2.3) are handled using a conservative second-order finite volume method [23], and a level-set method was used to define the ice extent region. The key advantages of using the level set technique is to be able to define the ice domain at the subgrid level and establish meaningful definition of the normal boundary conditions at ice-water boundaries. The simulations in Section 7 demonstrated that the two methods (BE and CN) both provide a well-behaved sea ice field after a 1 year simulation. The two methods remained consistent with one another up until roughly 140 days but begin to diverge after 100 days. At the end of the simulations (365 days), significant differences are apparent. The most noticeable difference is a significant smearing of the ice extent towards the upper-left corner of the domain for the CN method, which is much less pronounced in the BE method. However, based on the direction of the wind and ocean current velocities in this part of the domain (Figure 7), it is tempting to attach a physical attribute for this smearing and hold the BE method at fault. Further investigation is required to know for sure. Nonetheless, this result is a clear evidence that climate scale simulations are inevitably dependent on the details of the numerical methods and modeling errors can sometimes be surprisingly more significant than expected. Moreover, we demonstrated that numerical dissipation alone can lead to significant ice volume loss over the course of a climate simulation, even if the loss per time step may appear insignificant. The use of conservative methods, at the discrete level [10] are thus warranted. We also assessed the impact of the tanh approximation on the viscous-plastic rheology (3.3). We found that over the course of the 1 year simulations, the ice stresses remained in an acceptable range of the elliptic yield domain: $Y(\sigma_1, \sigma_2 \leq 0.005$. The BE method remained in this domain 93.95% of the time, while this number is 93.69% for the CN method. While both methods failed to satisfy the 99% criterion set forward by [20], they are relatively close.

All computations were performed on a Dell Inc. Latitude 5400 equipped with 8 GB RAM and an Intel Core i5-8350U CPU @ 1.70 GHz x 8, running Ubuntu 22.04.5 LTS. For a 12 hour-integration time, with a resolution of $dt = 20$ km and $dt = 10$ minutes, the CN method required 71.14 s of CPU time, whereas the BDF1 method took 93.55 s.

Author contributions

Salim Bensassi: Conducted the numerical simulations, visualization of the results and drafted the original manuscript, analyzed and interpreted the results; Boualem Khouider: Conceptualized the project and provided financial support and supervision, contributed the original software and helped with code optimization; M'hamed Kesri: Conceptualized the project and provided financial support and supervision, analyzed and interpreted the results; Clint Seinen: Analyzed and interpreted the results, contributed the original software and helped with code optimization. All authors have read and approved the final version of the manuscript for publication.

Use of Generative-AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This work is part of the first author's doctoral thesis project. The research of B.K. is sponsored by a Discovery grant from the Natural Sciences and Engineering Research Council of Canada.

Conflicts of interest

Prof. Boualem Khouider is an editorial board member for AIMS Mathematics and was not involved in the editorial review and/or the decision to publish this article. All authors declare no conflicts of interest in this paper.

References

1. M. S. Arif, K. Abodayeh, Y. Nawaz, Numerical modeling of mixed convective nanofluid flow with fractal stochastic heat and mass transfer using finite differences, *Front. Energy Res.*, **12** (2024), 1373079. https://doi.org/10.3389/fenrg.2024.1373079
2. M. S. Arif, K. Abodayeh, Y. Nawaz, A two-stage computational approach for stochastic Darcy-forchheimer non-Newtonian flows, *Front. Phys.*, **13** (2025), 1533252. https://doi.org/10.3389/fphy.2025.1533252
3. W. J. Campbell, The wind-driven circulation of ice and water in a polar ocean, *J. Geophys. Res.*, **70** (1965), 3279–3301. https://doi.org/10.1029/JZ070i014p03279
4. T. F. Chan, K. R. Jackson, Nonlinearly preconditioned Krylov subspace methods for discrete Newton algorithms, *SIAM J. Sci. Stat. Comput.*, **5** (1984), 533–542. https://doi.org/10.1137/0905039
5. S. Chatta, B. Khouider, Well posedness of the regularized-hibler model of sea-ice dynamics, *ESS Open Archive*, 2025. https://doi.org/10.22541/essoar.175309004.43816040/v1
6. S. Chatta, B. Khouider, M. Kesri, Linear well posedness of regularized equations of sea-ice dynamics, *J. Math. Phys.*, **64** (2023), 051504. https://doi.org/10.1063/5.0152991
7. M. D. Coon, G. A. Maykut, R. S. Pritchard, D. A. Rothrock, A. S. Thorndike, Modeling the pack ice as an elastic–plastic material, *AIDJEX Bulletin*, **24** (1974), 1–105.
8. D. R. Durran, *Numerical methods for fluid dynamics: With applications to geophysics*, Springer, **32** (2010). https://doi.org/10.1007/978-1-4419-6412-0
9. D. L Feltham, Sea ice rheology, *Annu. Rev. Fluid Mech.*, **40** (2008), 91–112. https://doi.org/10.1146/annurev.fluid.40.111406.102151
10. T. C. Fisher, M. H. Carpenter, J. Nordström, N. K. Yamaleev, C. Swanson, Discretely conservative finite-difference formulations for nonlinear conservation laws in split form: Theory and boundary conditions, *J. Comput. Phys.*, **234** (2013), 353–375. https://doi.org/10.1016/j.jcp.2012.09.026
11. O. Guba, J. Lorenz, D. Sulsky, On well-posedness of the viscous–plastic sea ice model, *J. Phys. Oceanogr.*, **43** (2013), 2185–2199. https://doi.org/10.1175/JPO-D-13-014.1
12. W. D. Hibler, A dynamic thermodynamic sea ice model, *J. Phys. Oceanogr.*, **9** (1979), 815–846.

13. W. D. Hibler III, A viscous sea ice law as a stochastic average of plasticity, *J. Geophys. Res.*, **82** (1977), 3932–3938. https://doi.org/10.1029/JC082i027p03932

14. E. C. Hunke, W. H. Lipscomb, A. K. Turner, N. Jeffery, S. Elliott, *CICE: The Los Alamos sea ice model documentation and software user's manual, Version 5.1*, Los Alamos National Laboratory, 2015.

15. E. C. Hunke, J. K. Dukowicz, An elastic-viscous-plastic model for sea ice dynamics, *J. Phys. Oceanogr.*, **27** (1997), 1849–1867.

16. E. C. Hunke, Y. Zhang, A comparison of sea ice dynamics models at high resolution, *Mon. Weather Rev.*, **127** (1999), 396–408.

17. E. C. Hunke, Viscous–plastic sea ice dynamics with the evp model: Linearization issues, *J. Comput. Phys.*, **170** (2001), 18–38. https://doi.org/10.1006/jcph.2001.6710

18. O. B. Johnson, L. I. Oluwaseun, Crank-nicolson and modified crank-nicolson scheme for one dimensional parabolic equation, *Int. J. Appl. Math. Theor. Phys.*, **6** (2020), 35–40. https://doi.org/10.11648/j.ijamtp.20200603.11

19. S. Juricke, T. Jung, Influence of stochastic sea ice parametrization on climate and the role of atmosphere–sea ice–ocean interaction, *Philos. Trans. A Math. Phys. Eng. Sci.*, **372** (2014), 20130283. https://doi.org/10.1098/rsta.2013.0283

20. J. F. Lemieux, B. Tremblay, Numerical convergence of viscous-plastic sea ice models, *J. Geophys. Res. Oceans*, **114** (2009), C05009. https://doi.org/10.1029/2008JC005017

21. J. F. Lemieux, B. Tremblay, J. Sedláček, P. Tupper, S. Thomas, D. Huard, et al., Improving the numerical convergence of viscous-plastic sea ice models with the Jacobian-free Newton–Krylov method, *J. Comput. Phys.*, **229** (2010), 2840–2852. https://doi.org/10.1016/j.jcp.2009.12.011

22. J. F. Lemieux, D. A. Knoll, B. Tremblay, D. M. Holland, M. Losch, A comparison of the Jacobian-free Newton–Krylov method and the EVP model for solving the sea ice momentum equation with a viscous-plastic formulation: A serial algorithm study, *J. Comput. Phys.*, **231** (2012), 5926–5944. https://doi.org/10.1016/j.jcp.2012.05.024

23. R. J. LeVeque, *Finite volume methods for hyperbolic problems*, Cambridge University Press, 2012. https://doi.org/10.1017/CBO9780511791253

24. M. Losch, S. Danilov, On solving the momentum equations of dynamic sea ice models with implicit solvers and the elastic–viscous–plastic technique, *Ocean Model.*, **41** (2012), 42–52. https://doi.org/10.1016/j.ocemod.2011.10.002

25. C. Mehlmann, S. Danilov, M. Losch, J. F. Lemieux, N. Hutter, T. Richter, et al., Simulating linear kinematic features in viscous-plastic sea ice models on quadrilateral and triangular grids with different variable staggering, *J. Adv. Model. Earth Syst.*, **13** (2021), e2021MS002523. https://doi.org/10.1029/2021MS002523

26. C. Mehlmann, T. Richter, A modified global Newton solver for viscous-plastic sea ice models, *Ocean Model.*, **116** (2017), 96–107. https://doi.org/10.1016/j.ocemod.2017.06.001

27. Á. Nagy, I. Omle, H. Kareem, E. Kovács, I. F. Barna, G. Bognar, Stable, explicit, leapfrog-hopscotch algorithms for the diffusion equation, *Computation*, **9** (2021), 92. https://doi.org/10.3390/computation9080092

28. H. Nessyahu, E. Tadmor, Non-oscillatory central differencing for hyperbolic conservation laws, *J. Comput. Phys.*, **87** (1990), 408–463. https://doi.org/10.1016/0021-9991(90)90260-8

29. E. Ólason, G. Boutin, A. Korosov, P. Rampal, T. Williams, M. Kimmritz, et al., A new brittle rheology and numerical framework for large-scale sea-ice models, *J. Adv. Model. Earth Syst.*, **14** (2022), e2021MS002685. https://doi.org/10.1029/2021MS002685 .

30. E. Ólason, G. Boutin, T. Williams, A. Korosov, H. Regan, J. Rheinlænder, et al., The next generation sea-ice model nextsim, version 2, *EGUsphere*, 2025. https://doi.org/10.5194/egusphere-2024-3521

31. C. Seinen, *A fast and efficient solver for viscous-plastic sea ice dynamics*, Master's thesis, Canada: The University of Victoria, 2017.

32. C. Seinen, B. Khouider, Improving the Jacobian free Newton–Krylov method for the viscous–plastic sea ice momentum equation, *Phys. D*, **376-377** (2018), 78–93. https://doi.org/10.1016/j.physd.2017.09.005

33. M. Sussman, P. Smereka, S. Osher, A level set approach for computing solutions to incompressible two-phase flow, *J. Comput. Phys.*, **114** (1994), 146–159. https://doi.org/10.1006/jcph.1994.1155

34. B. Tremblay, L. A. Mysak, Modeling sea ice as a granular material, including the dilatancy effect, *J. Phys. Oceanogr.*, **27** (1997), 2342–2360.

35. M. Yaremchuk, G. Panteleev, On the jacobian approximation in sea ice models with viscous-plastic rheology, *Ocean Model.*, **177** (2022), 102078.

36. J. Zhang, W. D. Hibler III, On an efficient numerical method for modeling sea ice dynamics, *J. Geophys. Res. Oceans*, **102** (1997), 8691–8702. https://doi.org/10.1029/96JC03744

AIMS Press