An unstructured C-grid type variational formulation for the sea ice dynamics

Giacomo Capodaglio¹, Mark R. Petersen², Adrian K. Turner³, Andrew F. Roberts³

Abstract

Historically, B-grid formulations of sea ice dynamics have been dominant because they have matched the grid type used by ocean models. The reason for the grid match is simple – it facilitates penetration of the curl of ice-ocean stress into the deep ocean with minimal numerical diffusivity because sea ice and ocean velocity are co-located. In recent years, as ocean models have increasingly progressed to C-grids, sea ice models have followed suit on quadrilateral meshes, but few if any implementations of unstructured C-grid sea ice models have been developed. We present an unstructured C-grid formulation of the elastic-viscous-plastic rheology, where the velocity unknowns are located at the edges rather than at the vertices, as in the B-grid. The mesh cells in our analysis have $n$ sides, with $n$ greater than or equal to four. Numerical results are also included to investigate the features of the proposed method. Our framework of choice is the Model for Prediction Across Scales (MPAS) within E3SM, the climate model of the U.S. Department of Energy, although our approach is general and could be applied to other models as well. While MPAS-Seaice is currently defined on a B-grid, MPAS-Ocean runs on a C-grid, hence interpolation operators are heavily used when coupled simulations are performed. The discretization introduced here aims at transitioning the dynamics of MPAS-Seaice to a C-grid, in order to ultimately facilitate the coupling with MPAS-Ocean and reduce numerical errors associated with this communication.

¹Computational Physics and Methods Group and Center For Nonlinear Studies, Los Alamos National Laboratory, e-mail: gcapodaglio@lanl.gov
²Computational Physics and Methods Group, Los Alamos National Laboratory
³Fluid Dynamics and Solid Mechanics Group, Los Alamos National Laboratory
⁴Document approved for unlimited release LA-UR-21-32250.
1. Introduction

Sea ice, saline ice buoyed to the surface of the ocean, plays an important role in the equilibrium of global climate. For instance, its production stimulates ocean overturning, creates a platform for snow cover that in turn greatly increases the planetary albedo of Earth, and forms a marine thermal blanket against the frigid winter polar atmosphere [5] [17]. The ability to model sea ice and predict its state is therefore an important task for climate modelers, and computer simulation is an invaluable tool for this purpose. Many numerical models of sea ice have been developed since the 1960s, with CICE [14] being perhaps the most widely used in this century owing to its ability to readily exploit parallel computing architectures. CICE is built on a quadrilateral structured mesh and a variational approach is used for the discretization of the divergence of internal ice stress. Within the Model for Prediction Across Scales (MPAS) framework [24] [23], such an approach has been generalized to unstructured grids in the MPAS-Seaice model [27], using meshes obtained from Voronoi tessellations [15]. Both MPAS-Seaice and CICE are built on an Arakawa B-grid [1], where the velocity components are discretized at the vertices of the computational mesh cells. Many existing sea ice models use this kind of staggered grid. Focusing only on the velocity components, other models that place them at the vertices are for instance FESIM [7], the sea ice component of FESOM [25] [28], and LIM [26], although a C-grid placement, where velocity components are discretized at the edge locations, is also available in the latter [3]. On the other hand, the ICON-O model features a triangular mesh with a C-grid type staggering and a finite element discretization [16], where the normal velocity component is defined at the edges of the computational cells. Interest in C-grid type of methods has grown, in part thanks to a shift in ocean model discretizations from a B-grid to a C-grid. Recent works on C-grid type implementations of sea ice dynamics include for instance [20] [19] [6]. Here, we aim at presenting the mathematical formulation of a variational C-grid type of approach for unstructured grids with polygonal cells having \( n \) sides, with \( n \geq 4 \). Our focus is on sea ice dynamics, hence we do not discuss the placement of scalar quantities, and assume that both components of the velocity are discretized at the same edge locations of the mesh. We observe that, traditionally, an Arakawa C-grid [1] would not have the velocity components co-located. Nevertheless, with a slight abuse of notation, we refer to our methodology as a C-grid approach because the velocity components are discretized at the edges of the mesh cells. The C-grid method proposed here is based on the variational strategy for the
elastic-viscous-plastic (EVP) rheology [12], laid out in [13] for a B-grid, and extended to unstructured meshes for the same type of grid in [27]. Although our method has originated with the MPAS framework in mind, it is general enough to be applied for instance to structured quadrilateral meshes as well.

The paper is organized as follows: in Section 2 we lay out the mathematical formulation, highlighting its applicability to a general class of polygonal meshes. Next, in Section 3, we present a series of test cases in planar and spherical domains to investigate the accuracy and convergence of the proposed method, comparing it with the B-grid formulation currently available in MPAS-Seaice. Finally, we summarize our findings and discuss future work in Section 4.

2. Variational formulation on a C-grid

Let us consider the sea ice momentum equation

\[
m \frac{\partial \mathbf{u}}{\partial t} = \nabla \cdot \mathbf{\sigma} + \mathbf{\tau}_a + \mathbf{\tau}_w - \mathbf{k} \times m \mathbf{f} \mathbf{u} - mg \nabla H_0.
\]  

(1)

The left hand side represents the inertial term, with \( m \) being the mass of snow and ice per unit area and \( \mathbf{u} \) the sea-ice velocity. On the right hand side, the first term is the divergence of the ice internal stress \( \mathbf{\sigma} \), \( \mathbf{\tau}_a \) and \( \mathbf{\tau}_w \) are the horizontal stresses due to atmospheric winds and ocean currents respectively, the next term is the Coriolis force and the last takes into account the force coming from the ocean surface tilt. The unit vector \( \mathbf{k} \) is normal to the Earth surface, \( f \) is the Coriolis parameter, \( g \) is the gravitational acceleration, and \( H_0 \) is the ocean surface height. The aim of this section is to discuss how to compute the divergence of the internal stress \( \mathbf{\sigma} \), assuming the components of the velocity vector are discretized at the edges of the mesh cells, i.e. on a C-grid, rather than at the vertices, i.e. on a B-grid. Our framework of choice is MPAS, of which the ocean and sea ice components are part of the Energy Exascale Earth System Model (E3SM), developed by the U.S. Department of Energy, which runs full climate simulations on variable-resolution meshes [9, 21, 4]. The present analysis aims at facilitating the coupling of MPAS-Seaice, which currently runs on a B-grid, with MPAS-Ocean, which is on a C-grid instead. Having the velocities co-located would reduce numerical diffusivity during communication between the two models. The MPAS codes run on unstructured polygonal meshes obtained from a Voronoi tessellation [22], normally referred to as the primal mesh, to which is associated a Delaunay triangulation, the
dual mesh. For a recent paper on MPAS type meshes see [10]. In MPAS, the discretization points on the edges are located at the intersection between line segments joining dual cell centers with primal cell centers. In MPAS-Ocean, an orthogonal reference frame is placed at every edge of the mesh with the tangential axis oriented as the edge. Moreover, only one component of the velocity vector is prognostic, namely the one that is normal to the edge according to this reference frame. Unlike MPAS-Ocean, here we make the assumption that all the orthogonal reference frames at the edge locations are oriented in the same way according to global eastwards and northwards directions, as is currently in the B-grid formulation of MPAS-Seaice, see Figure 1. Moreover, for MPAS-Seaice both components of the velocity vector are prognostic, because they are needed for computing the divergence of the internal stress. Therefore, all the vector fields \( \mathbf{f} \) discussed from now on are expressed according to a reference frame whose components are referred to as eastwards and northwards, i.e. \( \mathbf{f} = (f_\alpha, f_\beta) \), with \( f_\alpha \) being the component directed eastwards and \( f_\beta \) the one directed northwards. This framework for MPAS-Seaice requires that the velocity fields coming from MPAS-Ocean have to be rotated first, before they can be used as input for the sea ice model.

We focus on the discretization of the term \( \mathbf{F} := \nabla \cdot \mathbf{\sigma} \), because the other terms in Eq. 1 can be handled in a fairly straightforward way on a C-grid. Following the variational approach described in [27], we observe that over the entire domain, the total work done by the internal stress is equal to the dissipation of mechanical energy (ignoring boundary effects):

\[
\int_{\Omega}(\mathbf{u} \cdot \mathbf{F})dA = -\int_{\Omega}(\sigma_{11}\epsilon_{11} + 2\sigma_{12}\epsilon_{12} + \sigma_{22}\epsilon_{22})dA, \tag{2}
\]
where $\sigma$ is the internal stress tensor, as already mentioned, and $\epsilon$ is the strain tensor. For ease of notation, let

$$
D_1 := -\int_\Omega \sigma_{11} \epsilon_{11} dA, \quad D_2 := -\int_\Omega 2\sigma_{12} \epsilon_{12} dA, \quad D_3 := -\int_\Omega \sigma_{22} \epsilon_{22} dA, \quad \text{and} \quad D := D_1 + D_2 + D_3. \quad (3)
$$

Now, the domain $\Omega$ appearing in the above integrals is intended to be the computational domain, i.e. $\Omega = \bigcup_{c=1}^{N_c} V_c$, where $V_c$ is the $c$-th cell of the computational grid and $N_c$ is the total number of cells.

**Remark 2.1.** To avoid tedious notation, in this analysis we assume we are dealing with an aquaplanet, i.e. a spherical domain with no continents, hence no boundaries are present on the computational grid. The results of the analysis do not change in case continents or boundaries are present, and the treatment of coastal boundary conditions follows either a Dirichlet or a Neumann approach.

Given that the integrals are over $\Omega$, we decompose each of them as the sum of integrals over diamond shaped polygons $P_e$ that only intersect at their boundaries, each of which is associated with an edge of the mesh, as in Figure 2. Such polygons are obtained from the computational grid by joining the vertices that are the end points of an edge $e$ with the cell centers that share $e$, see the green shapes in Figure 2. Note that the $P_e$ form a cover of $\Omega$, i.e. if $N_e$ is the total number of edges of the mesh, then

$$
\Omega = \bigcup_{e=1}^{N_e} P_e. \quad (4)
$$

Hence Eq. (2) becomes

$$
\sum_{e=1}^{N_e} \int_{P_e} (u \cdot F) dA = D(u_{\alpha,1}, u_{\alpha,2}, \ldots, u_{\alpha,N_e}, u_{\beta,1}, u_{\beta,2}, \ldots, u_{\beta,N_e}), \quad (5)
$$

where $(u_{\alpha,e}, u_{\beta,e}) = u(x_e, y_e)$, with $(x_e, y_e)$ being the location of the discretization point on edge $e$.

We have

$$
\sum_{e=1}^{N_e} \int_{P_e} (u_\alpha F_\alpha + u_\beta F_\beta) dA = D(u_{\alpha,1}, u_{\alpha,2}, \ldots, u_{\alpha,N_e}, u_{\beta,1}, u_{\beta,2}, \ldots, u_{\beta,N_e}). \quad (6)
$$

We assume that, on $P_e$, $u$ and $F$ are constant and equal to the value at the edge location $(x_e, y_e)$, i.e. $u|_{P_e} =$
Figure 2: Decomposition of the computational domain as in Eq. (4) with diamond shaped polygons (delimited by dashed lines), each of which is uniquely associated with an edge location (red triangle). Left: quadrilateral mesh. Right: Voronoi tessellation.

\[ u(x_e, y_e), \] where \( u(x_e, y_e) = (u_{\alpha,e}, u_{\beta,e}) \) and \( F|_{P_e} = F(x_e, y_e) \), where \( F(x_e, y_e) = (F_{\alpha,e}, F_{\beta,e}) \). Therefore, the integral in Eq. (6) becomes

\[ \int_{P_e} (u_{\alpha} F_{\alpha} + u_{\beta} F_{\beta}) \, dA = (u_{\alpha,e} F_{\alpha,e} + u_{\beta,e} F_{\beta,e}) A_{P_e}, \] (7)

where \( A_{P_e} \) denotes the area of \( P_e \). With the above equation, we obtain

\[ \sum_{e=1}^{N_e} (u_{\alpha,e} F_{\alpha,e} + u_{\beta,e} F_{\beta,e}) A_{P_e} = D(u_{\alpha,1}, u_{\alpha,2}, \ldots, u_{\alpha,N_e}, u_{\beta,1}, u_{\beta,2}, \ldots, u_{\beta,N_e}). \] (8)

Differentiation of the above equation with respect to \( u_{\alpha,e'} \) and \( u_{\beta,e'} \) gives, respectively

\[ \frac{\partial}{\partial u_{\alpha,e'}} \sum_{e=1}^{N_e} (u_{\alpha,e} F_{\alpha,e} + u_{\beta,e} F_{\beta,e}) A_{P_e} = \frac{\partial}{\partial u_{\alpha,e'}} D(u_{\alpha,1}, u_{\alpha,2}, \ldots, u_{\alpha,N_e}, u_{\beta,1}, u_{\beta,2}, \ldots, u_{\beta,N_e}), \] \[ \frac{\partial}{\partial u_{\beta,e'}} \sum_{e=1}^{N_e} (u_{\alpha,e} F_{\alpha,e} + u_{\beta,e} F_{\beta,e}) A_{P_e} = \frac{\partial}{\partial u_{\beta,e'}} D(u_{\alpha,1}, u_{\alpha,2}, \ldots, u_{\alpha,N_e}, u_{\beta,1}, u_{\beta,2}, \ldots, u_{\beta,N_e}). \] (9)

Under the same assumption in [13, 27], i.e. that \( F_{\alpha,e} \) and \( F_{\beta,e} \) do not depend on any of the \( u_{\alpha,e} \) or \( u_{\beta,e} \),
we have

\[ F_{\alpha,e} = \frac{1}{A_P} \frac{\partial}{\partial u_{\alpha,e}} D(u_{\alpha,1}, u_{\alpha,2}, \ldots, u_{\alpha,N_e}, u_{\beta,1}, u_{\beta,2}, \ldots, u_{\beta,N_e}), \]  

\[ F_{\beta,e} = \frac{1}{A_P} \frac{\partial}{\partial u_{\beta,e}} D(u_{\alpha,1}, u_{\alpha,2}, \ldots, u_{\alpha,N_e}, u_{\beta,1}, u_{\beta,2}, \ldots, u_{\beta,N_e}). \]  

(10)

Making the terms \(D_1\), \(D_2\) and \(D_3\) explicit, we have:

\[ D_1 = -\int_{\Omega} \sigma_{11} \left[ \frac{\partial u_{\alpha}}{\partial x} - u_{\beta} C_1(r) \tan(\lambda) \right] dA, \]  

(11)

\[ D_2 = -\int_{\Omega} \sigma_{12} \left[ \frac{\partial u_{\alpha}}{\partial y} + \frac{\partial u_{\beta}}{\partial x} + u_{\alpha} C_2(r) \tan(\lambda) \right] dA, \]  

(12)

\[ D_3 = -\int_{\Omega} \sigma_{22} \left[ \frac{\partial u_{\beta}}{\partial y} + u_{\beta} C_3(r) \tan(\lambda) \right] dA, \]  

(13)

where \(C_i(r), i = 1, 2, 3\) is either identically zero, if the domain is planar, or equal to \(b_i/r\) if the domain is spherical, where \(r\) is the radius of the spherical domain \(r\) and \(b_i\) is a non-negative number that does not depend on \(r\). Above, \(\lambda\) represents the latitude. The terms that multiply \(\tan(\lambda)\) are usually called metric terms [13].

Next, we consider a different cover of \(\Omega\), see Figure 3: let \(N_v\) be the total number of vertices of the mesh and recall that \(N_c\) represent the total number of cells. Then a cover of \(\Omega\) is given by

\[ \Omega = \left( \bigcup_{c=1}^{N_c} \hat{V}_c \right) \cup \left( \bigcup_{v=1}^{N_v} T_v \right), \]  

(14)

where, for a given edge \(e\), the \(\hat{V}_i\) are polygons obtained by joining the edge points of the two cells \(V_i\) and \(V_j\) that own \(e\) (see the orange shapes in Figure 3) and the polygons \(T_v\) are obtained by joining with the point on \(e\) the edge points on those edges \(e'\) that share a vertex with \(e\) and that belong to one of the cells that owns \(e\) (see the blue shapes in Figure 3). Note that the polygons \(\hat{V}_i\) and \(T_v\) overlap only on their boundaries.

**Remark 2.2.** In the case of a quadrilateral mesh as in Figure 3 left, both \(\hat{V}_i\) and \(T_v\) are quadrilaterals. In case of a Voronoi tessellation as in Figure 3 right, \(\hat{V}_i\) can be a, for instance, a hexagon or a pentagon, and \(T_v\)
Based on the decomposition in Eq. (14), we express any of the functions $u_\alpha$, $u_\beta$ or $\sigma_{ij}$ on either $\hat{V}_c$ or $T_v$ according to a basis expansion, where the specific choice of basis depends on whether we are in $\hat{V}_c$ or $T_v$. Namely, for any of the functions $u_\alpha$, $u_\beta$ or $\sigma_{ij}$, we approximate their value at any point $(x, y)$ in $T_v$ as the

\textit{is a triangle.}
linear combination of basis functions centered at the vertices of \( T_v \)

\[
f(x, y) \approx \sum_{i=1}^{n_v} f_{v_i} \mathcal{L}_{v_i}(x, y),
\]

where \( n_v \) is the number of vertices of \( T_v \), \( f \) is any of the functions just mentioned, \( f_{v_i} \) is the value of the function at the \( i \)-th vertex of \( T_v \) (which is an edge location) and \( \mathcal{L}_{v_i} \) is the basis function associated with vertex \( v_i \). We choose basis functions \( \mathcal{L}_{v_i} \) for which

\[
\begin{align*}
\sum_{i=1}^{n_v} \mathcal{L}_{v_i}(x, y) &= 1 \text{ for any } (x, y) \in T_v, \\
\mathcal{L}_{v_i}(x_{v_j}, y_{v_j}) &= \delta_{ij}
\end{align*}
\]

where \((x_{v_j}, y_{v_j})\) is the location of the \( j \)-th vertex of \( T_v \) and \( \delta_{ij} \) is Kronecker’s delta. Because of these features, the function \( f \) is approximated with a function similar to a finite element nodal interpolator. The same approach is used for any polygon \( \hat{V}_c \), with basis functions centered at the vertices of \( \hat{V}_c \), which are the edge points of the Voronoi cell \( V_c \). Therefore, we can write the value of \( u_\alpha, u_\beta \) and \( \sigma_{ij} \) at any point \((x, y)\) in \( \hat{V}_c \) as the linear combination of basis functions as

\[
f(x, y) \approx \sum_{i=1}^{n_c} f_{c_i} B_{c_i}(x, y),
\]

where \( n_c \) is the number of vertices of polygon \( \hat{V}_c \), \( f \) is again the field, \( f_{c_i} \) is the value of the field at the \( i \)-th vertex of \( \hat{V}_c \) (which is an edge location of \( V_c \)) and \( B_{c_i} \) is the basis function associated with the polygon vertex \( c_i \). We also require \( \{B_{c_i}\} \) to have the same properties in (16).

**Remark 2.3.** The choice of basis functions for the implementation will be either Wachspress [8, 27] or piece-wise linear (PWL) [2, 27] basis functions, as these two options are those currently available in MPAS-Seaice for the B-grid. Both options guarantee the properties in (16). Moreover, globally, the fields will be continuous over the computational domain because both \( \{\mathcal{L}_{v_i}\} \) and \( \{B_{c_i}\} \) are linear at the edges of their respective domains of definition for either the Wachspress choice or the PWL.

With the decomposition in Eq. (14) in mind, we continue by considering \( D_1 \) and therefore the integral in
which can be rewritten as

\[ D_1 = - \sum_{e=1}^{N_e} \int_{T_e} \sigma_{11} \left[ \frac{\partial u_\alpha}{\partial x} - u_\beta C_1(r) \tan(\lambda) \right] dA - \sum_{c=1}^{N_c} \int_{\hat{V}_c} \sigma_{11} \left[ \frac{\partial u_\alpha}{\partial x} - u_\beta C_1(r) \tan(\lambda) \right] dA. \]  

(18)

Substituting Eq. (15) and (17) in Eq. (18) we get

\[ D_1 = - \sum_{e=1}^{N_e} \int_{T_e} \sum_{j=1}^{n_e} \sigma_{11} \left[ \sum_{j=1}^{n_e} u_{avj} \frac{\partial \mathcal{L}_{ej}}{\partial x} - \sum_{j=1}^{n_e} u_{vaj} \mathcal{L}_{ej} C_1(r) \tan(\lambda) \right] dA 
- \sum_{c=1}^{N_c} \int_{\hat{V}_c} \sum_{j=1}^{n_c} \sigma_{11} \left[ \sum_{j=1}^{n_c} u_{acj} \frac{\partial \mathcal{B}_{cj}}{\partial x} - \sum_{j=1}^{n_c} u_{bcj} \mathcal{B}_{cj} C_1(r) \tan(\lambda) \right] dA. \]  

(19)

Under the assumption that the value of \( \lambda \) changes slowly within \( T_e \) or \( \hat{V}_c \), the above equation becomes

\[ D_1 = - \sum_{e=1}^{N_e} \sum_{i=1}^{n_e} \sum_{j=1}^{n_e} \sigma_{11} \left( u_{avj} \int_{T_e} \mathcal{L}_{ej} \frac{\partial \mathcal{L}_{ej}}{\partial x} dA - u_{vaj} \int_{T_e} \mathcal{L}_{ej} \mathcal{L}_{ej} dA \right) 
- \sum_{c=1}^{N_c} \sum_{i=1}^{n_c} \sum_{j=1}^{n_c} \sigma_{11} \left( u_{acj} \int_{\hat{V}_c} \mathcal{B}_{cj} \frac{\partial \mathcal{B}_{cj}}{\partial x} dA - u_{bcj} \int_{\hat{V}_c} \mathcal{B}_{cj} \mathcal{B}_{cj} dA \right). \]  

(20)

with \( \lambda_e \) begin the latitude at edge \( e \). Now, there exist \( \hat{e} \) and \( \hat{e'} \) such that \( e \) is a vertex of \( T_{\hat{e}} \) and \( T_{\hat{e'}} \). Also, there exist \( \hat{c} \) and \( \hat{c'} \) such that \( e \) is a vertex of \( \hat{V}_{\hat{c}} \) and \( \hat{V}_{\hat{c'}} \). Hence a given edge point receives contributions for the divergence of the stress from two adjacent polygons \( \hat{V}_{\hat{e}} \) and two adjacent polygons \( T_{\hat{e}}, \) as shown also in Figure 3 and Figure 4. In the existing formulation on MPAS-Seaice that relies on a B-grid, a given vertex receives contributions from the four cells that own that vertex in case of a quadrilateral mesh. The cells are three in case of a Voronoi tessellation, see Figure 4. With these considerations, we take the derivative of Eq. (20) with respect to \( u_{\alpha,e} \) and \( u_{\beta,e} \) to obtain

\[ \frac{\partial D_1}{\partial u_{\alpha,e}} = - \left( \sum_{i=1}^{n_e} \left( \sigma_{11} \psi_i (N_e^e)_{i,j} + \sigma_{11} \psi'_i (N_e^e)_{i,j} \right) + \sum_{i=1}^{n_e} \left( \sigma_{11} \psi_i (M_e^e)_{i,j} + \sigma_{11} \psi'_i (M_e^e)_{i,j} \right) \right), \]  

\[ \frac{\partial D_1}{\partial u_{\beta,e}} = C_1(r) \tan(\lambda_e) \left( \sum_{i=1}^{n_e} \left( \sigma_{11} \psi_i (M_e^e)_{i,j} + \sigma_{11} \psi'_i (M_e^e)_{i,j} \right) + \sum_{i=1}^{n_c} \left( \sigma_{11} \psi_i (N_e^e)_{i,j} + \sigma_{11} \psi'_i (N_e^e)_{i,j} \right) \right), \]  

(21)

where \( j \) in each case refers to the local index that corresponds to the global index of \( e \), and \( M_e \) and \( M_c \) are
the matrices defined as

\[(M_v)_{i,j} = \int_{\mathcal{V}_v} \mathcal{L}_{v_i} \mathcal{L}_{v_j} \, dA, \quad i, j = \{1, \ldots, n_v\}, \quad v = 1, \ldots, N_v,\]

\[(M_c)_{i,j} = \int_{\mathcal{V}_c} B_{c_i} B_{c_j} \, dA, \quad i, j = \{1, \ldots, n_c\}, \quad c = 1, \ldots, N_c,\]

\[(N^v_{\sigma})_{i,j} = \int_{\mathcal{V}_v} \frac{\partial \mathcal{L}_{v_i}}{\partial \sigma} \, dA, \quad i, j = \{1, \ldots, n_v\}, \quad v = 1, \ldots, N_v,\]

\[(N^c_{\sigma})_{i,j} = \int_{\mathcal{V}_c} \frac{\partial B_{c_i}}{\partial \sigma} \, dA, \quad i, j = \{1, \ldots, n_c\}, \quad c = 1, \ldots, N_c.\]  

(22)

Note that in Eq. (22), \(M_v\) and \(M_c\) are symmetric matrices, whereas \(N^v_{\sigma}\) and \(N^c_{\sigma}\) are not. It follows from Eq. (10) that

\[
(\nabla \cdot \sigma)_{\alpha,e}^{D_1} = \frac{1}{A_{P_e}} \frac{\partial D_1}{\partial u_{\alpha,e}} = -\frac{1}{A_{P_e}} \left( \sum_{i=1}^{n_v} (\sigma_{11} \tilde{e}_i (N^v_{\sigma})_{i,j} + \sigma_{12} \tilde{e}_i (N^v_{\sigma})_{i,j}) + \sum_{i=1}^{n_c} (\sigma_{11} \tilde{e}_i (N^c_{\sigma})_{i,j} + \sigma_{12} \tilde{e}_i (N^c_{\sigma})_{i,j}) \right),
\]

\[
(\nabla \cdot \sigma)_{\beta,e}^{D_1} = \frac{1}{A_{P_e}} \frac{\partial D_1}{\partial u_{\beta,e}} = \frac{C_1(r)}{A_{P_e}} \tan(\lambda_e) \left( \sum_{i=1}^{n_v} (\sigma_{11} \tilde{e}_i (M_v)_{i,j} + \sigma_{12} \tilde{e}_i (M_v)_{i,j}) + \sum_{i=1}^{n_c} (\sigma_{11} \tilde{e}_i (M_c)_{i,j} + \sigma_{12} \tilde{e}_i (M_c)_{i,j}) \right). \tag{23}
\]

We continue with \(D_2\) and therefore the integral in (12), which can be expressed as

\[
D_2 = -\sum_{v=1}^{N_v} \int_{\mathcal{V}_v} \sigma_{12, v} \mathcal{L}_{v_i} \left[ \frac{\partial u_\alpha}{\partial y} + \frac{\partial u_\beta}{\partial x} + u_\alpha C_2(r) \tan(\lambda) \right] dA - \sum_{c=1}^{N_c} \int_{\mathcal{V}_c} \sigma_{12, c} \left[ \frac{\partial u_\alpha}{\partial y} + \frac{\partial u_\beta}{\partial x} + u_\alpha C_2(r) \tan(\lambda) \right] dA. \tag{24}
\]

Substituting the basis expansions, the above equation becomes

\[
D_2 = -\sum_{v=1}^{N_v} \sum_{i=1}^{n_v} \sigma_{12, v} \mathcal{L}_{v_i} \left[ \sum_{j=1}^{n_v} u_{\alpha v} \frac{\partial \mathcal{L}_{v_j}}{\partial y} + \sum_{j=1}^{n_v} u_{\beta v} \frac{\partial \mathcal{L}_{v_j}}{\partial x} + C_2(r) \tan(\lambda) \sum_{j=1}^{n_v} u_{\alpha v} \mathcal{L}_{v_j} \right] dA
\]

\[-\sum_{c=1}^{N_c} \sum_{i=1}^{n_c} \sigma_{12, c} B_{c_i} \left[ \sum_{j=1}^{n_c} u_{\alpha c} \frac{\partial B_{c_j}}{\partial y} + \sum_{j=1}^{n_c} u_{\beta c} \frac{\partial B_{c_j}}{\partial x} + C_2(r) \tan(\lambda) \sum_{j=1}^{n_c} u_{\alpha c} B_{c_j} \right] dA. \tag{25}
\]
Under the same assumption made for $D_1$ of a slowly varying $\lambda$ within $T_e$ and $\hat{V}_c$ we have

\[ D_2 = -\sum_{v=1}^{N_v} \sum_{i=1}^{n_v} \sum_{j=1}^{n_v} \sigma_{12v} u_{\alpha v} \int_{T_v} \mathcal{L}_{v_i} \frac{\partial \mathcal{L}_{v_j}}{\partial y} dA + u_{\beta v} \int_{T_v} \mathcal{L}_{v_i} \frac{\partial \mathcal{L}_{v_j}}{\partial x} dA + C_2(r) \tan(\lambda_c) u_{\alpha v} \int_{T_v} \mathcal{L}_{v_i} \mathcal{L}_{v_j} dA \]

\[ -\sum_{c=1}^{N_c} \sum_{i=1}^{n_c} \sum_{j=1}^{n_c} \sigma_{12c} u_{\alpha c} \int_{\hat{V}_c} \mathcal{B}_{c_i} \frac{\partial \mathcal{B}_{c_j}}{\partial y} dA + u_{\beta c} \int_{\hat{V}_c} \mathcal{B}_{c_i} \frac{\partial \mathcal{B}_{c_j}}{\partial x} dA + C_2(r) \tan(\lambda_c) u_{\alpha c} \int_{\hat{V}_c} \mathcal{B}_{c_i} \mathcal{B}_{c_j} dA. \] 

(26)

With the same reasoning as for $D_1$, given an edge $e$ we have

\[ \frac{\partial D_2}{\partial u_{\alpha,e}} = -\sum_{i=1}^{n_v} \left[ \sigma_{12v} \left( (N^v_{y})_{i,j} + C_2(r) \tan(\lambda_e) (M^v_{\alpha})_{i,j} \right) + \sigma_{12v} \left( (N^v_{y})_{i,j} + C_2(r) \tan(\lambda_e) (M^v_{\alpha})_{i,j} \right) \right] \]

\[-\sum_{i=1}^{n_v} \left[ \sigma_{12v} \left( (N^v_{y})_{i,j} + C_2(r) \tan(\lambda_e) (M^v_{\alpha})_{i,j} \right) + \sigma_{12v} \left( (N^v_{y})_{i,j} + C_2(r) \tan(\lambda_e) (M^v_{\alpha})_{i,j} \right) \right], \]

(27)

\[ \frac{\partial D_2}{\partial u_{\beta,e}} = -\sum_{i=1}^{n_v} (\sigma_{12v} (N^v_{y})_{i,j} + \sigma_{12v} (N^v_{y})_{i,j}) - \sum_{i=1}^{n_v} (\sigma_{12v} (N^v_{y})_{i,j} + \sigma_{12v} (N^v_{y})_{i,j}). \]

(28)

where $j$ in each case refers again to the local index that corresponds to the global index of $e$, and

\[ (N^v_{y})_{i,j} = \int_{T_v} \mathcal{L}_{v_i} \frac{\partial \mathcal{L}_{v_j}}{\partial y} dA, \quad i, j = \{1, \ldots, n_v\}, \quad v = 1, \ldots, N_v, \]

\[ (N^v_{y})_{i,j} = \int_{\hat{V}_c} \mathcal{B}_{c_i} \frac{\partial \mathcal{B}_{c_j}}{\partial y} dA, \quad i, j = \{1, \ldots, n_c\}, \quad c = 1, \ldots, N_c, \]

are non-symmetric matrices.
It follows from Eq. (10) that

\[
(\nabla \cdot \sigma)^{D_2}_{\alpha,e} = \frac{1}{A_{\beta,e}} \frac{\partial D_2}{\partial u_{\alpha,e}}
\]

\[
= -\frac{1}{A_{\beta,e}} \left( \sum_{i=1}^{n_v} \left[ \sigma_{12\beta_i} \left( (N^\beta_{ei})_{i,j} + C_2(r) \tan(\lambda_e)(M^\beta_{ei})_{i,j} \right) + \sigma_{12\beta_i} \left( (N^\beta_{ei})_{i,j} + C_2(r) \tan(\lambda_e)(M^\beta_{ei})_{i,j} \right) \right] + \sum_{i=1}^{n_v} \left[ \sigma_{12\beta_i} \left( (N^\beta_{ei})_{i,j} + C_2(r) \tan(\lambda_e)(M^\beta_{ei})_{i,j} \right) + \sigma_{12\beta_i} \left( (N^\beta_{ei})_{i,j} + C_2(r) \tan(\lambda_e)(M^\beta_{ei})_{i,j} \right) \right],
\]

\[
(\nabla \cdot \sigma)^{D_2}_{\beta,e} = \frac{1}{A_{\beta,e}} \frac{\partial D_2}{\partial u_{\beta,e}}
\]

\[
= -\frac{1}{A_{\beta,e}} \left( \sum_{i=1}^{n_v} \left( \sigma_{12\beta_i} (N^\beta_{ei})_{i,j} + \sigma_{12\beta_i} (N^\beta_{ei})_{i,j} \right) + \sum_{i=1}^{n_v} \left( \sigma_{12\beta_i} (N^\beta_{ei})_{i,j} + \sigma_{12\beta_i} (N^\beta_{ei})_{i,j} \right) \right).
\]

(29)

Last, let's consider \( D_3 \), hence the integral in Eq. (13), which becomes

\[
D_3 = -\sum_{v=1}^{N_v} \int_{T_v} \sigma_{22} \left[ \frac{\partial u_{\beta}}{\partial y} + u_{\beta} C_3(r) \tan(\lambda) \right] dA - \sum_{e=1}^{N_e} \int_{V_e} \sigma_{22} \left[ \frac{\partial u_{\beta}}{\partial y} + u_{\beta} C_3(r) \tan(\lambda) \right] dA.
\]

(30)

Substituting the basis expansion we have

\[
D_3 = -\sum_{v=1}^{N_v} \int_{T_v} \sum_{i=1}^{n_v} \sigma_{22\beta_i} \left[ \sum_{j=1}^{m_v} u_{\beta v_j} \frac{\partial L_{v_j}}{\partial y} + C_3(r) \tan(\lambda) \sum_{j=1}^{n_v} u_{\beta v_j} L_{v_j} \right] dA
\]

\[
- \sum_{e=1}^{N_e} \int_{V_e} \sum_{i=1}^{n_e} \sigma_{22\beta_i} \left[ \sum_{j=1}^{m_e} u_{\beta c_j} \frac{\partial B_{c_j}}{\partial y} + C_3(r) \tan(\lambda) \sum_{j=1}^{n_e} u_{\beta c_j} B_{c_j} \right] dA,
\]

(31)

which then gives

\[
D_3 = -\sum_{v=1}^{N_v} \sum_{i=1}^{n_v} \sum_{j=1}^{n_v} \sigma_{22\beta_i} \left( u_{\beta v_j} \int_{T_v} L_{v_j} \frac{\partial L_{v_j}}{\partial y} dA + C_3(r) \tan(\lambda) u_{\beta v_j} \int_{T_v} L_{v_j} L_{v_j} dA \right)
\]

\[
- \sum_{e=1}^{N_e} \sum_{i=1}^{n_e} \sum_{j=1}^{n_e} \sigma_{22\beta_i} \left( u_{\beta c_j} \int_{V_e} B_{c_j} \frac{\partial B_{c_j}}{\partial y} dA + C_3(r) \tan(\lambda) u_{\beta c_j} \int_{V_e} B_{c_j} B_{c_j} dA \right).
\]

(32)
Hence, we have

\[
\frac{\partial D_3}{\partial u_{\alpha,e}} = 0,
\]

\[
\frac{\partial D_3}{\partial u_{\beta,e}} = -\sum_{i=1}^{n_v} \left[ \sigma_{22\hat{v}_i} \left( (N^p_v)_{i,j} + C_3(r) \tan(\lambda_e)(M_{\hat{v}_e})_{i,j} \right) + \sigma_{22\hat{v}_i'} \left( (N^p_{v'})_{i,j} + C_3(r) \tan(\lambda_e)(M_{\hat{v'}_e})_{i,j} \right) \right] - \sum_{i=1}^{n_c} \left[ \sigma_{22\hat{c}_i} \left( (N^p_c)_{i,j} + C_3(r) \tan(\lambda_e)(M_{\hat{c}_e})_{i,j} \right) + \sigma_{22\hat{c}_i'} \left( (N^p_{c'})_{i,j} + C_3(r) \tan(\lambda_e)(M_{\hat{c'}_e})_{i,j} \right) \right].
\]

(33)

It follows again from Eq. (10) that

\[
(\nabla \cdot \sigma)^{D_3}_{\alpha,e} = 0,
\]

\[
(\nabla \cdot \sigma)^{D_3}_{\beta,e} = \frac{1}{A_p} \frac{\partial D_3}{\partial u_{\beta,e}}
\]

\[
= -\frac{1}{A_p} \left( \sum_{i=1}^{n_v} \left[ \sigma_{22\hat{v}_i} \left( (N^p_v)_{i,j} + C_3(r) \tan(\lambda_e)(M_{\hat{v}_e})_{i,j} \right) + \sigma_{22\hat{v}_i'} \left( (N^p_{v'})_{i,j} + C_3(r) \tan(\lambda_e)(M_{\hat{v'}_e})_{i,j} \right) \right] + \sum_{i=1}^{n_c} \left[ \sigma_{22\hat{c}_i} \left( (N^p_c)_{i,j} + C_3(r) \tan(\lambda_e)(M_{\hat{c}_e})_{i,j} \right) + \sigma_{22\hat{c}_i'} \left( (N^p_{c'})_{i,j} + C_3(r) \tan(\lambda_e)(M_{\hat{c'}_e})_{i,j} \right) \right] \right).
\]

(34)

**Remark 2.4.** We observe that the proposed approach applied to a structured quadrilateral grid is equivalent to discretizing the momentum equation on a rotated grid, to relocate the velocity components from the vertices to the edges, as it is clear from Figure 4 (left).

Following [27], is it possible to define an alternative formulation by using the decomposition of Ω in Eq. (14) from the very beginning, in place of Eq. (4). In this way, Eq. (5) becomes

\[
\sum_{v=1}^{N_v} \int_{T_v} (u \cdot F) dA + \sum_{c=1}^{N_c} \int_{\hat{V}_c} (u \cdot F) dA = D(u_{\alpha,1}, u_{\alpha,2}, \ldots, u_{\alpha,N_e}, u_{\beta,1}, u_{\beta,2}, \ldots, u_{\beta,N_e}),
\]

(35)

which gives

\[
\sum_{v=1}^{N_v} \int_{T_v} (u_\alpha F_{\alpha} + u_\beta F_{\beta}) dA + \sum_{c=1}^{N_c} \int_{\hat{V}_c} (u_\alpha F_{\alpha} + u_\beta F_{\beta}) dA = D(u_{\alpha,1}, u_{\alpha,2}, \ldots, u_{\alpha,N_e}, u_{\beta,1}, u_{\beta,2}, \ldots, u_{\beta,N_e}).
\]

(36)

Now, we assume that within each polygon, $F_{\alpha}$ and $F_{\beta}$ are expanded using the same basis functions we intro-
duced above for internal stress and velocity, hence

\[
\sum_{e=1}^{N_e} \int_{T_e} (u_\alpha F_\alpha + u_\beta F_\beta) dA = \sum_{e=1}^{N_e} \int_{T_e} \left[ \left( \sum_{i=1}^{n_r} u_{\alpha e_i} \mathcal{L}_{e_i} \right) \left( \sum_{j=1}^{n_e} F_{\alpha e_j} \mathcal{L}_{e_j} \right) + \left( \sum_{k=1}^{n_r} u_{\beta e_k} \mathcal{L}_{e_k} \right) \left( \sum_{l=1}^{n_e} F_{\beta e_l} \mathcal{L}_{e_l} \right) \right] dA,
\]

\[
\sum_{c=1}^{N_c} \int_{\tilde{V}_c} (u_\alpha F_\alpha + u_\beta F_\beta) dA = \sum_{c=1}^{N_c} \int_{\tilde{V}_c} \left[ \left( \sum_{i=1}^{n_r} u_{\alpha c_i} \mathcal{B}_{c_i} \right) \left( \sum_{j=1}^{n_e} F_{\alpha e_j} \mathcal{B}_{e_j} \right) + \left( \sum_{k=1}^{n_r} u_{\beta c_k} \mathcal{B}_{c_k} \right) \left( \sum_{l=1}^{n_e} F_{\beta e_l} \mathcal{B}_{e_l} \right) \right] dA.
\]

(37)

We now substitute the above equation in Eq. (36) and take the derivative with respect to \( u_{\alpha,e} \) (doing so with respect to \( u_{\beta,e} \) would produce a similar outcome)

\[
\sum_{j=1}^{n_e} F_{\alpha e_j} \int_{T_{e_j}} \mathcal{L}_{e_j} \mathcal{L}_{e_j} + \sum_{j=1}^{n_e} F_{\alpha e_j} \int_{T_{e_j}} \mathcal{L}_{e_j} \mathcal{L}_{e_j} + \sum_{c=1}^{n_e} \int_{\tilde{V}_c} \mathcal{B}_{c_j} \mathcal{B}_{c_j} + \sum_{j=1}^{n_e} \int_{\tilde{V}_{e_j}} \mathcal{B}_{e_j} \mathcal{B}_{e_j} = \frac{\partial D}{\partial u_{\alpha,e}},
\]

(38)

with \( \hat{i} \) being the local index corresponding to \( e \).

We make the assumption that \( \mathbf{F} \) varies slowly within \( T_{e_j} \cup T_{e_j} \cup \tilde{V}_c \cup \tilde{V}_{e_j} \) and that its value within the closure of this set can be approximated with the value at \( e \), hence the above equation simplifies to

\[
F_{\alpha,e} \left( \sum_{j=1}^{n_e} \int_{T_{e_j}} \mathcal{L}_{e_j} \mathcal{L}_{e_j} + \sum_{j=1}^{n_e} \int_{T_{e_j}} \mathcal{L}_{e_j} \mathcal{L}_{e_j} + \sum_{c=1}^{n_e} \int_{\tilde{V}_c} \mathcal{B}_{c_j} \mathcal{B}_{c_j} + \sum_{j=1}^{n_e} \int_{\tilde{V}_{e_j}} \mathcal{B}_{e_j} \mathcal{B}_{e_j} \right) = \frac{\partial D}{\partial u_{\alpha,e}}.
\]

(39)

Using the first property in Eq. (16), this leads to

\[
F_{\alpha,e} \left( \int_{T_{e}} \mathcal{L}_{e} + \int_{T_{e}} \mathcal{L}_{e} + \int_{\tilde{V}_{e}} \mathcal{B}_{e} + \int_{\tilde{V}_{e}} \mathcal{B}_{e} \right) = \frac{\partial D}{\partial u_{\alpha,e}}.
\]

(40)

Hence, the new value of \( A_{P_z} \) with this alternative formulation would be

\[
A_{P_z} = \int_{T_{e}} \mathcal{L}_{e} + \int_{T_{e}} \mathcal{L}_{e} + \int_{\tilde{V}_{e}} \mathcal{B}_{e} + \int_{\tilde{V}_{e}} \mathcal{B}_{e},
\]

(41)

instead of being the area of the diamond shaped polygon associated with edge \( e \).
3. Numerical Results

To investigate the properties of our formulation, we present results on the plane and sphere focusing on the accuracy and convergence of the proposed discretization using analytical solutions as references.

3.1. Spatial discretization test

We begin with a theoretical analysis to obtain sufficient conditions under which the proposed scheme is expected to be at least a second order approximation of the continuous divergence operator. Placing ourselves in a general setting, let us consider $\Omega \subset \mathbb{R}^2$ to be a closed and bounded set, and let $\sigma$ be a stress tensor defined on $\Omega$ given by

$$\sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{bmatrix},$$

(42)

with $\sigma_{12} = \sigma_{21}$. Note that as before, $\Omega$ denotes the computational domain. We also introduce $\Omega_e \subset \Omega$ to be the set of all edge points of the mesh, i.e. if $e$ is an edge of the mesh, then $(x_e, y_e) \in \Omega_e$. The divergence of the stress is a $1 \times 2$ vector defined as

$$\nabla \cdot \sigma = \begin{bmatrix} \frac{\partial \sigma_{11}}{\partial x} + \frac{\partial \sigma_{21}}{\partial y} \\ \frac{\partial \sigma_{12}}{\partial x} + \frac{\partial \sigma_{22}}{\partial y} \end{bmatrix}^T.$$

(43)

Above, we have defined $F = (F_\alpha, F_\beta) := \nabla \cdot \sigma$, hence

$$F_\alpha = \frac{\partial \sigma_{11}}{\partial x} + \frac{\partial \sigma_{21}}{\partial y}, \quad F_\beta = \frac{\partial \sigma_{12}}{\partial x} + \frac{\partial \sigma_{22}}{\partial y}.$$ 

(44)

Evaluated at an edge location $(x_e, y_e) \in \Omega_e$, the above functions give

$$F_\alpha(x_e, y_e) = \frac{\partial \sigma_{11}}{\partial x}(x_e, y_e) + \frac{\partial \sigma_{21}}{\partial y}(x_e, y_e), \quad F_\beta(x_e, y_e) = \frac{\partial \sigma_{12}}{\partial x}(x_e, y_e) + \frac{\partial \sigma_{22}}{\partial y}(x_e, y_e).$$

(45)
For the same point \((x_e, y_e) \in \Omega_e\), from Eq. (10), we have

\[
F_\alpha(x_e, y_e) := F_{\alpha,e} = \frac{1}{A_{p,e}} \frac{\partial}{\partial u_{\alpha,e}} (D_1 + D_2 + D_3),
\]

\[
F_\beta(x_e, y_e) := F_{\beta,e} = \frac{1}{A_{p,e}} \frac{\partial}{\partial u_{\beta,e}} (D_1 + D_2 + D_3).
\]

Considering a planar case with no metric terms, for \(F_{\alpha,e}\) we have

\[
\frac{1}{A_{p,e}} \frac{\partial D_1}{\partial u_{\alpha,e}} = -\frac{1}{A_{p,e}} \left( \sum_{i=1}^{n_c} (\sigma_{11\hat{e}_i}(N^x_{e})_{i,j} + \sigma_{11\hat{e}'_i}(N^x_{e})_{i,j}) + \sum_{i=1}^{n_c} (\sigma_{11\hat{e}_i}(N^y_{e})_{i,j} + \sigma_{11\hat{e}'_i}(N^y_{e})_{i,j}) \right),
\]

\[
\frac{1}{A_{p,e}} \frac{\partial D_2}{\partial u_{\alpha,e}} = -\frac{1}{A_{p,e}} \left( \sum_{i=1}^{n_c} (\sigma_{12\hat{e}_i}(N^x_{e})_{i,j} + \sigma_{12\hat{e}'_i}(N^x_{e})_{i,j}) + \sum_{i=1}^{n_c} (\sigma_{12\hat{e}_i}(N^y_{e})_{i,j} + \sigma_{12\hat{e}'_i}(N^y_{e})_{i,j}) \right),
\]

\[
\frac{1}{A_{p,e}} \frac{\partial D_3}{\partial u_{\alpha,e}} = 0.
\]

Recall that above \(j\) is the local index associated to the edge \(e\). It follows by comparing (45) and (46) that with our spatial discretization of the divergence of the stress, we are approximating the gradient operator at the edge points \((x_e, y_e) \in \Omega_e\) in the following way:

\[
\frac{\partial \sigma_{11}}{\partial x}(x_e, y_e) \approx -\frac{1}{A_{p,e}} \left( \sum_{i=1}^{n_c} (\sigma_{11\hat{e}_i}(N^x_{e})_{i,j} + \sigma_{11\hat{e}'_i}(N^x_{e})_{i,j}) + \sum_{i=1}^{n_c} (\sigma_{11\hat{e}_i}(N^y_{e})_{i,j} + \sigma_{11\hat{e}'_i}(N^y_{e})_{i,j}) \right),
\]

\[
\frac{\partial \sigma_{21}}{\partial y}(x_e, y_e) \approx -\frac{1}{A_{p,e}} \left( \sum_{i=1}^{n_c} (\sigma_{12\hat{e}_i}(N^x_{e})_{i,j} + \sigma_{12\hat{e}'_i}(N^x_{e})_{i,j}) + \sum_{i=1}^{n_c} (\sigma_{12\hat{e}_i}(N^y_{e})_{i,j} + \sigma_{12\hat{e}'_i}(N^y_{e})_{i,j}) \right).
\]

For \(F_{\beta,e}\), still considering a planar case with no metric terms, we have

\[
\frac{1}{A_{p,e}} \frac{\partial D_1}{\partial u_{\beta,e}} = 0,
\]

\[
\frac{1}{A_{p,e}} \frac{\partial D_2}{\partial u_{\beta,e}} = -\frac{1}{A_{p,e}} \left( \sum_{i=1}^{n_c} (\sigma_{22\hat{e}_i}(N^x_{e})_{i,j} + \sigma_{22\hat{e}'_i}(N^x_{e})_{i,j}) + \sum_{i=1}^{n_c} (\sigma_{22\hat{e}_i}(N^y_{e})_{i,j} + \sigma_{22\hat{e}'_i}(N^y_{e})_{i,j}) \right),
\]

\[
\frac{1}{A_{p,e}} \frac{\partial D_3}{\partial u_{\beta,e}} = -\frac{1}{A_{p,e}} \left( \sum_{i=1}^{n_c} (\sigma_{22\hat{e}_i}(N^x_{e})_{i,j} + \sigma_{22\hat{e}'_i}(N^x_{e})_{i,j}) + \sum_{i=1}^{n_c} (\sigma_{22\hat{e}_i}(N^y_{e})_{i,j} + \sigma_{22\hat{e}'_i}(N^y_{e})_{i,j}) \right).
\]
This implies that

\[
\frac{\partial \sigma_{12}}{\partial x}(x_e, y_e) \approx -\frac{1}{A_{P_e}} \left( \sum_{i=1}^{n_e} \left( \sigma_{12} \varepsilon_i (N_e^x)_{i,j} + \sigma_{12} \varepsilon_i^i (N_e^y)_{i,j} \right) + \sum_{i=1}^{n_e} \left( \sigma_{12} \varepsilon_i (N_e^v)_{i,j} + \sigma_{12} \varepsilon_i (N_e^v)_{i,j} \right) \right),
\]

\[
\frac{\partial \sigma_{22}}{\partial y}(x_e, y_e) \approx -\frac{1}{A_{P_e}} \left( \sum_{i=1}^{n_e} \left( \sigma_{22} \varepsilon_i (N_e^y)_{i,j} + \sigma_{22} \varepsilon_i (N_e^y)_{i,j} \right) + \sum_{i=1}^{n_e} \left( \sigma_{22} \varepsilon_i (N_e^v)_{i,j} + \sigma_{22} \varepsilon_i (N_e^v)_{i,j} \right) \right),
\]

which leads to the same conclusion reached after Eq. (48), i.e. that our spatial discretization approximates

\[
\nabla f(x, y) \approx \mathcal{F}(f)(x, y) := \left[ -\frac{1}{A_{P_e}} \left( \sum_{i=1}^{n_e} (f_{\varepsilon_i} (N_e^x)_{i,j} + f_{\varepsilon_i} (N_e^y)_{i,j}) + \sum_{i=1}^{n_e} (f_{\varepsilon_i} (N_e^v)_{i,j} + f_{\varepsilon_i} (N_e^v)_{i,j}) \right) \right],
\]

where \( C^2(\Omega) \) is the space of twice differentiable functions with continuous derivatives on \( \Omega \), and \( \mathcal{B}(\Omega_e) \) denotes the space of bounded functions on \( \Omega_e \). Let \( y \in \Omega_e \), then to test the accuracy of the approximation in Eq. (51), we consider the Taylor expansion of \( f \in C^2(\Omega_e) \) at \( x \in \Omega_e \) (with \( x \) and \( y \) being edges points of the same element):

\[
f(x) = f(y) + (x_1 - y_1) \frac{\partial}{\partial x_1} f(y) + (x_2 - y_2) \frac{\partial}{\partial x_2} f(y) \]

\[
+ \frac{1}{2} \left[ (x_1 - y_1)^2 \frac{\partial^2}{\partial x_1^2} f(y) + (x_2 - y_2)(x_1 - y_1) \frac{\partial^2}{\partial x_1 \partial x_2} f(y) \right]
\]

\[
+ (x_1 - y_1)(x_2 - y_2) \frac{\partial^2}{\partial x_2^2} f(y) + (x_2 - y_2)^2 \frac{\partial^2}{\partial x_2} f(y) \right] + \mathcal{O}(\|x - y\|^3). \tag{54}
\]

Let us define

\[
g_1(x) := 1, \quad g_2(x) := (x_1 - y_1), \quad g_3(x) := (x_2 - y_2), \quad g_4(x) := (x_1 - y_1)^2, \tag{55}
\]

\[
g_5(x) := (x_2 - y_2)(x_1 - y_1), \quad g_6(x) = g_5(x), \quad g_7(x) = (x_2 - y_2)^2. \tag{56}
\]
which are all functions in $C^2(\Omega)$, and the coefficients

$$c_1 := f(y), \quad c_2 := \frac{\partial}{\partial x_1} f(y), \quad c_3 := \frac{\partial}{\partial x_2} f(y), \quad c_4 := \frac{1}{2} \frac{\partial^2}{\partial x_1^2} f(y),$$

$$c_5 := \frac{1}{2} \frac{\partial^2}{\partial x_1 \partial x_2} f(y), \quad c_6 := c_5, \quad c_7 = \frac{1}{2} \frac{\partial^2}{\partial x_2^2} f(y).$$

(57)

Then, given $y \in \Omega_e$ and neglecting the third order terms, $f$ can be approximated by a truncated Taylor expansion as

$$f \approx \sum_{i=1}^{7} c_i g_i,$$

(59)

with value at $x \in \Omega_e$ approximated by

$$f(x) \approx \sum_{i=1}^{7} c_i g_i(x).$$

(60)

Note that $C^2(\Omega)$ is a linear space and $g_i \in C^2(\Omega)$ for $i = 1, \ldots, 7$ hence due to the linearity of $\mathcal{F}$ we have

$$\mathcal{F}(f)(x) \approx \sum_{i=1}^{7} c_i \mathcal{F}(g_i)(x).$$

(61)

Expanding the above sum we get

$$\mathcal{F}(f)(x) \approx f(y) \mathcal{F}(g_1)(x) + \frac{\partial}{\partial x_1} f(y) \mathcal{F}(g_2)(x) + \frac{\partial}{\partial x_2} f(y) \mathcal{F}(g_3)(x)$$

$$+ \frac{1}{2} \left[ \frac{\partial^2}{\partial x_1^2} f(y) \mathcal{F}(g_4)(x) + \frac{\partial^2}{\partial x_1 \partial x_2} f(y) \mathcal{F}(g_5)(x) + \frac{\partial^2}{\partial x_2^2} f(y) \mathcal{F}(g_7)(x) \right].$$

(62)

(63)

(64)

Because we want $\mathcal{F}$ to be an approximation of the gradient operator (ideally, we would like it to be exactly the gradient operator) on $\Omega_e$, we set the equality

$$\nabla f(x) = \mathcal{F}(f)(x), \quad \forall x \in \Omega_e.$$
The above equality holds if the approximation below holds

\[ \nabla f(x) \approx f(y) \mathcal{F}(g_1)(x) + \frac{\partial}{\partial x_1} f(y) \mathcal{F}(g_2)(x) + \frac{\partial}{\partial x_2} f(y) \mathcal{F}(g_3)(x) \]
\[ + \frac{1}{2} \left[ \frac{\partial^2}{\partial x_1^2} f(y) \mathcal{F}(g_4)(x) + \frac{\partial^2}{\partial x_1 \partial x_2} f(y) \mathcal{F}(g_5)(x) \right. \]
\[ + \left. \frac{\partial^2}{\partial x_2^2} f(y) \mathcal{F}(g_6)(x) + \frac{\partial^2}{\partial x_2^2} f(y) \mathcal{F}(g_7)(x) \right] . \] (66)

Hence, the relation above provides a sufficient condition for \( \mathcal{F} \) to be at least a second order approximation of the derivative operator. In fact, if \( x = y \), then (66) implies

\[ \nabla f(y) \approx f(y) \mathcal{F}(g_1)(y) + \frac{\partial}{\partial x_1} f(y) \mathcal{F}(g_2)(y) + \frac{\partial}{\partial x_2} f(y) \mathcal{F}(g_3)(y) \]
\[ + \frac{1}{2} \left[ \frac{\partial^2}{\partial x_1^2} f(y) \mathcal{F}(g_4)(y) + \frac{\partial^2}{\partial x_1 \partial x_2} f(y) \mathcal{F}(g_5)(y) \right. \]
\[ + \left. \frac{\partial^2}{\partial x_2^2} f(y) \mathcal{F}(g_6)(y) + \frac{\partial^2}{\partial x_2^2} f(y) \mathcal{F}(g_7)(y) \right] . \] (67)

The above relation shows that the following conditions are sufficient for \( \mathcal{F} \) to be at least a second order approximation of \( \nabla f \):

\[
\begin{align*}
\mathcal{F}(g_i)((x_e, y_e)) & = [0, 0]^T, \quad i \neq 2, i \neq 3, \\
\mathcal{F}(g_2)((x_e, y_e)) & = [1, 0]^T, \\
\mathcal{F}(g_3)((x_e, y_e)) & = [0, 1]^T.
\end{align*}
\] (68)

for any point \((x_e, y_e) \in \Omega_e\). Note that if \( \mathcal{F} \) was indeed equal to \( \nabla f \) then the above conditions would be satisfied.

With respect to the analysis just concluded, we have numerically estimated the values of \( \mathcal{F}(g_i) \) for \( i = \{1, 2, 3, 4, 5, 7\} \), considering \( \mathcal{F} \) to be the operator obtained with the C-grid approach proposed in this paper and also the one obtained with the B-grid approach form [27]. We use a planar mesh with regular hexagonal cells and one with square cells. For the B-grid approach, we select a vertex of the mesh that is not on the boundary of the domain, and neither surrounded by boundary cells. For the C-grid case, we consider the edges that have such a vertex in common. With this setup, for the C-grid case there will be three edges for the hexagonal mesh, each of which will be oriented differently, and four edges for the quadrilateral mesh, with
pairs of edges oriented in the same way. To avoid numerical error, instead of computing the first component of $F(g_2)$, we compute the difference

$$\left| A_{P_e} - \left( - \sum_{i=1}^{n_e} (f_{e_i}N_{e_i}^T)_{i,j} + f_{V'}(N_{V'}^T)_{i,j} \right) - \sum_{i=1}^{n_e} (f_{e_i}'(N_{e_i}^T)_{i,j} + f_{V'}'(N_{V'}^T)_{i,j}) \right|,$$

hence if such a difference is zero, then the first component of $F(g_2)$ is one. Let us denote with $\tilde{F}(g_2)$ the vector $F(g_2)$ whose first entry has been modified as explained. We adopt the same strategy for the second entry of $F(g_3)$, and define $\tilde{F}(g_3)$ in similar way as $\tilde{F}(g_2)$. Hence, if $\tilde{F}(g_i) = 0$ for $i = 2, 3$ and $F(g_j) = 0$ for $j = 1, 4, 5, 7$, then according to the conditions in (68), we can expect the methods to be at least second order accurate. Obviously the equality to the zero vector is intended in machine precision sense. For this test (and all the planar tests) the area $A_{P_e}$ will be the area of the diamond like shapes in Figure 2. We consider both Wachspress and PWL basis functions. For the mesh with square cells, the B-grid returned zero vectors for both Wachspress and PWL basis functions, as did the C-grid, for all four edges considered and both types of basis functions. Hence, on the quadrilateral mesh with square cells, both methods are expected to show second order convergence for the divergence of the stress operator. For the hexagonal mesh, the C-grid returned zero vectors for all three edges and both choices of basis functions, whereas the B-grid did so only for the PWL basis functions. In fact, with the Wachspress choice we had

$$F(g_4) = [0, \gamma_1]^T, \quad F(g_5) = [\gamma_2, 0]^T, \quad F(g_7) = [0, \gamma_3]^T,$$

(69)

with $\gamma_1, \gamma_2$ and $\gamma_3$ being non zero numbers. Hence, the choice of Wachspress basis functions is not expected to be second order with the B-grid approach but only at least first order. This was already observed in [27].

3.2. Convergence rate test

We continue with two tests to assess the accuracy of the proposed discretization in approximating the divergence of the internal stress. Namely, we first consider a unit square domain discretized with the same planar meshes used in the previous section (although with hexagonal cells the domain is not exactly a unit square), and then move to a unit sphere domain discretized with a Voronoi tessellation. In all cases, we consider the values of the internal stress to be prescribed (i.e. given as input) at the edges or vertices, and obtained
analytically through the simplest constitutive relation, $\sigma_{ij} = \epsilon_{ij}$, i.e. we assume the strain and the stress to be equal.

3.2.1. Convergence rate test on a planar mesh

For the planar mesh test case, the strain is obtained from derivatives of an analytical velocity field $\mathbf{u} = (u, u)$ with $u$ given by

$$u(x, y) = \sin(5.12\pi x) \sin(5.12\pi y).$$

(70)

The strain (and hence the stress) is given by

$$\epsilon_{11} = \frac{\partial u}{\partial x}, \quad \epsilon_{22} = \frac{\partial u}{\partial y}, \quad \epsilon_{12} = \frac{1}{2} \left( \frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} \right).$$

(71)

Therefore, the analytical field we use to compute errors is

$$\mathbf{F} = \left[ \frac{\partial \epsilon_{11}}{\partial x} + \frac{\partial \epsilon_{12}}{\partial y}, \frac{\partial \epsilon_{12}}{\partial x} + \frac{\partial \epsilon_{22}}{\partial y} \right].$$

(72)

We are going to compare the B-grid and the C-grid formulations using the following discrete relative $L_2$ norm

$$\sqrt{\sum A_i (F_i - \tilde{F}_i)^2} \sqrt{\sum A_i F_i^2}$$

(73)

where $F$ denotes any component of $\mathbf{F}$ and $\tilde{F}$ any component of the numerical approximation $\tilde{\mathbf{F}}$ of $\mathbf{F}$ computed either with the B-grid or the C-grid method. Note that $\tilde{\mathbf{F}}$ is only available either at the vertices or at the edges, whereas $\mathbf{F}$ being an analytic field can be computed at any spatial location $(x, y)$. In Eq. (73), the summation is taken over the vertices for the B-grid and over the edges for the C-grid. For the planar case, $A_i$ will be the area of the dual triangle centered at the $i$-th vertex for the B-grid or the area of the diamond shaped figure centered at the $i$-th edge for the C-grid. Results for the C-grid approach for both the mesh with square cells and the one with hexagonal cells are shown in Figure 5 (right), considering the eastwards component of the divergence of the stress (results are analogous for the northwards component, hence they’re not shown).

We observe that, as expected from the test in Section 3.1, the method has second order convergence on both
types of mesh and with both types of basis functions, i.e. Wachspress and PWL. Moreover, the choice of basis function does not affect the quality of the approximation, as the associated curves lie on top of each other for a given choice of mesh cells. In Figure 5 (left) we also display the qualitative behavior of the numerical solution obtained on the mesh with hexagonal cells for the C-grid. Next, we compare the B-grid and C-grid approaches on the mesh with square cells and on the one with hexagonal cells, see Figure 6. We only show the behavior of the eastwards component of the divergence of the stress, because the northwards showed an analogous behavior. Once again, as expected by the analysis in Section 3.1, the C-grid approach and the B-grid approach are both second order on the mesh with square cells, with the C-grid showing lower errors than the B-grid. Both methods are insensitive to the choice of basis functions in this case. For the mesh with hexagonal cells, the C-grid approach has again lower errors compared to the B-grid, for which the case of Wachspress basis functions becomes first order as the resolution of the mesh is increased. This behavior for the B-grid was expected from the analysis in the previous section and already reported in [27].
3.2.2. Convergence rate test on a spherical mesh

We continue our analysis considering a spherical Voronoi mesh on a unit sphere. An investigation of the mesh quality revealed that most cells are hexagons, with the exception of a few pentagons. Errors for this case are computed considering only vertices or edges for which their latitude $\lambda$ satisfies $|\lambda| > 20^\circ$. This is because, as explained in [27], the MPAS-Seaice grid is rotated so that the poles of the eastwards and northwards directions are placed at the equator, to avoid a convergence of the northwards components of the velocity at the geographic poles. Hence, with this rotation, the errors due to the metric terms will be prevalent at the equator where no sea ice is present, and therefore leaving these latitudes out of the calculation of the errors is justified.

We are assuming the following constitutive relation $\sigma_{ij} = \epsilon_{ij}$, hence the analytical divergence of the stress field that we use to compute errors is obtained using derivatives of the velocity field $u = (u_1, u_2)$ given by

$$u_1(\lambda, \phi) = Y_5^3(\pi/2 - \lambda, \phi), \quad u_2(\lambda, \phi) = Y_4^2(\pi/2 - \lambda, \phi),$$

(74)

where $(\lambda, \phi)$ are latitude and longitude, and $Y$ is a spherical harmonic function. Considering the metric terms
and geographical coordinates, the strain is given by

\[
\begin{align*}
\epsilon_{11} &= \frac{1}{r \cos(\lambda)} \frac{\partial u_1}{\partial \phi} - \frac{u_2}{r} \tan(\lambda), \\
\epsilon_{22} &= \frac{1}{r} \frac{\partial u_2}{\partial \lambda}, \\
\epsilon_{12} &= \frac{1}{2r} \frac{\partial u_1}{\partial \lambda} + \frac{1}{2r \cos(\lambda)} \frac{\partial u_2}{\partial \phi} + \frac{u_1}{2r} \tan(\lambda) .
\end{align*}
\]

The stress divergence field \( \mathbf{F} = (F_1, F_2) \) we use as analytical solution is also expressed using geographical coordinates, and is the same used in \[27\]

\[
\begin{align*}
F_1 &= \frac{1}{r \cos(\lambda)} \frac{\partial \epsilon_{11}}{\partial \phi} + \frac{1}{r} \frac{\partial \epsilon_{12}}{\partial \lambda} + \frac{2}{r} \tan(\lambda) \epsilon_{12}, \\
F_2 &= \frac{1}{r \cos(\lambda)} \frac{\partial \epsilon_{12}}{\partial \phi} + \frac{1}{r} \frac{\partial \epsilon_{22}}{\partial \lambda} + \frac{1}{r} \tan(\lambda) (\epsilon_{11} - \epsilon_{22}) .
\end{align*}
\]

We observe that, to have the B-grid and the C-grid numerical solutions converge to the analytical expression above, the values of the functions \( C_i(r) \) with \( i = 1, 2, 3 \) have to be different with the two methods. Namely, for the B-grid \( C_1(r) = C_2(r) = 1/r \) and \( C_3(r) = 0 \), choice that is consistent with the definition of the strain in Eq. \(75\). For the C-grid, we have to set \( C_1(r) = C_3(r) = 1/r \) and \( C_2(r) = 2/r \), which means the strain is corrected once substituted into the discrete expression of the divergence of the stress, with extra terms that match those multiplied by \( \tan(\lambda) \) in Eq. \(76\). At the moment, we could not reach a definite conclusion on why this correction is necessary for the C-grid. As shown in \[27\], the B-grid approach can converge on a spherical mesh with \( A_P \) being either the area of the dual triangle centered at a given vertex, or the lumped mass matrix type of quantity defined in \[41\]. Note that for the B-grid, the definition in \[41\] is different in that the integrals are over the three cells that own a given vertex. On the other hand, we found that the C-grid approach on the sphere converges only if the choice in \[41\] is considered for \( A_P \), i.e. the diamond shape option does not provide convergence. Therefore, for both the B-grid and the C-grid we use the choice of \( A_P \) defined in \[41\], hence in the \( L_2 \) norm computation in \[73\] the areas \( A_i \) are equal to \( A_P \) in \[41\]. Another difference between the B-grid and the C-grid approach lies in the way the matrices in Eq. \[22\] and Eq. \[28\] are computed. Namely, for the B-grid, it is sufficient to project the vertex coordinates on a plane tangent to the sphere at the cell center. This means that, in general, for a given vertex, the contributions coming from the three cells that own it (see Figure 1(left) ) would not be computed on the same plane. For the C-grid, on the other hand, to ensure convergence it is necessary to project all the four shapes in Figure 1(right) on the same tangent plane at an edge location,
so for a given edge the contributions coming from the four shapes are all computed on the same plane.

Besides the $L_2$ norm defined in (73), for the spherical tests we also consider the $L_\infty$ norm defined as

$$\max_i |F_i - \tilde{F}_i|,$$

where $F_i$ and $\tilde{F}_i$ are as in (73) and the maximum is taken over all vertices or edges that take part in the computation, depending on whether a B-grid or a C-grid approach is used.

Results are shown in Figure 7 for the case of the eastwards component of the divergence of the stress and in Figure 8 for the northwards. We observe from Figure 7 that while the B-grid approach shows a first order convergence rate for the $L_2$ norm, the C-grid remains second order as it was on a planar mesh. Moreover, the B-grid does not convergence in the $L_\infty$ norm for neither choice of basis function whereas the C-grid approach shows a convergence rate that is slightly better than linear, with the PWL choice showing lower errors than the Wachspress. For the northwards component, the results in Figure 8 show a similar behavior. In Figure 9 we are displaying the eastwards component of the divergence of the stress for the B-grid and the C-grid.
considering different views. The images refer to the case of PWL basis functions and the lowest resolution considered. In Figure 10 we are showing the errors of the numerical solutions compared to the analytical for the eastwards component of the divergence of the stress. Notice how, as expected, the higher values of the error are in proximity of latitudes that are excluded from the computation of the errors. A similar comparison for the northwards component is in Figure 11.

3.3. Velocity solver in a square domain

We conclude with some qualitative results obtained on a square domain of size 80 km as in [27], considering only the velocity solver and turning off advection and column physics. The setup is based on a similar test from [11]. The constitutive relation for this case is the EVP. No snow is present and ice thickness is fixed at 2 m. Ice concentration increases linearly in the eastwards direction from zero at the western boundary to one at the eastern boundary. Forcing terms originate from atmospheric winds $\mathbf{u}^a = (u_1^a, u_2^a)$ and ocean currents
Figure 9: Numerical solution for the eastwards component of the divergence of the stress. Top: B-grid approach. Bottom: C-grid approach. All pictures consider PWL basis functions and lowest resolution.

\[ \mathbf{u}^o = (u^o_1, u^o_2) \]

of the form

\[ u^o_1 = 5 - 3 \sin(2\pi x/L_x) \sin(\pi y/L_y), \quad u^o_2 = 5 - 3 \sin(2\pi y/L_y) \sin(\pi x/L_x), \]

\[ u^o_1 = 0.1((2y - L_y)/L_y), \quad u^o_2 = -0.1((2x - L_x)/L_x), \]

(78)

with \( L_x \) and \( L_y \) being the domain size in the eastwards and northwards directions respectively. The velocity solver is advanced with four time steps, and \( \Delta t = 60 \text{ min} \). The aim of this test is to show that the results obtained with the C-grid are qualitatively similar to those obtained with the B-grid. For this purpose, we display in Figure 12 and Figure 13 the two velocity components for the two approaches, considering Wachspress basis functions and the final time of the simulation. We observe that both velocity profiles are very similar.
Figure 10: Left: error for the B-grid approach. Right: error for the C-grid approach. Both pictures consider the eastwards component of the divergence of the stress, PWL basis functions and lowest resolution of the grid.

Figure 11: Left: error for the B-grid approach. Right: error for the C-grid approach. Both pictures consider the northwards component of the divergence of the stress, PWL basis functions and lowest resolution of the grid.

qualitatively, with only minimal differences at the top right corner of the domain. For completeness, we also show the two components of the divergence of the stress in Figure 14 and Figure 15.
4. Conclusions

We presented a promising new variational formulation on a C-grid for the sea ice dynamics, focusing our analysis on the accuracy of the proposed method in approximating the divergence of the internal stress, which is arguably the most challenging term to discretize for the sea ice dynamics. Studying the convergence rate,
we have shown that the proposed method is second order accurate on a planar domain as well as in a spherical one and that is capable of reproducing similar results as the current B-grid formulation in MPAS-Seaice when used within a full velocity solver. More investigation on the method is needed to fully understand its inner
workings, especially concerning the need for correction terms on the spherical domain, that seem to be necessary to achieve convergence to an analytical solution. Despite the necessity of further work, the method showed appealing features and proved to be more accurate than the current discretization in MPAS-Seaice, making it a viable alternative to be explored in the future.

Acknowledgments

This work was sponsored by the Center For Nonlinear Studies (CNLS) at Los Alamos Laboratory. The authors would like to thank Sara Calandrini, Darren Engwirda and Elizabeth Hunke for helpful discussions.

References

[1] Akio Arakawa and Vivian R Lamb. Computational design of the basic dynamical processes of the UCLA general circulation model. General circulation models of the atmosphere, 17(Supplement C):173–265, 1977.

[2] Teresa S Bailey, Marvin L Adams, Brian Yang, and Michael R Zika. A piecewise linear finite element discretization of the diffusion equation for arbitrary polyhedral grids. Journal of Computational Physics, 227(8):3738–3757, 2008.

[3] Sylvain Bouillon, Miguel Angel Morales Maqueda, Vincent Legat, and Thierry Fichefet. An elastic–viscous–plastic sea ice model formulated on Arakawa B and C grids. Ocean Modelling, 27(3-4):174–184, 2009.

[4] Peter M Caldwell, Azamat Mametjanov, Qi Tang, Luke P Van Roekel, Jean-Christophe Golaz, Wuyin Lin, David C Bader, Noel D Keen, Yan Feng, Robert Jacob, et al. The DOE E3SM coupled model version 1: Description and results at high resolution. Journal of Advances in Modeling Earth Systems, 11(12):4095–4146, 2019.

[5] Judith A Curry, Julie L Schramm, and Elizabeth E Ebert. Sea ice-albedo climate feedback mechanism. Journal of Climate, 8(2):240–247, 1995.

[6] S Danilov, C Mehlmann, and V Fofonova. On discretizing sea-ice dynamics on triangular meshes using vertex, cell or edge velocities. arXiv preprint arXiv:2106.13641, 2021.
[7] Sergey Danilov, Qiang Wang, Ralph Timmermann, Nikolay Iakovlev, Dmitry Sidorenko, Madlen Kimmritz, Thomas Jung, and Jens Schröter. Finite-element sea ice model (FESIM), version 2. Geoscientific Model Development, 8(6):1747–1761, 2015.

[8] Gautam Dasgupta. Interpolants within convex polygons: Wachspress’ shape functions. Journal of Aerospace Engineering, 16(1):1–8, 2003.

[9] Jean-Christophe Golaz, Peter M Caldwell, Luke P Van Roekel, Mark R Petersen, Qi Tang, Jonathan D Wolfe, Guta Abeshu, Valentine Anantharaj, Xylar S Asay-Davis, David C Bader, et al. The DOE E3SM coupled model version 1: Overview and evaluation at standard resolution. Journal of Advances in Modeling Earth Systems, 11(7):2089–2129, 2019.

[10] Kristin E Hoch, Mark R Petersen, Steven R Brus, Darren Engwirda, Andrew F Roberts, Kevin L Rosa, and Phillip J Wolfram. MPAS-Ocean simulation quality for variable-resolution North American coastal meshes. Journal of Advances in Modeling Earth Systems, 12(3):e2019MS001848, 2020.

[11] Elizabeth C Hunke. Viscous–plastic sea ice dynamics with the EVP model: Linearization issues. Journal of Computational Physics, 170(1):18–38, 2001.

[12] Elizabeth C Hunke and John K Dukowicz. An elastic–viscous–plastic model for sea ice dynamics. Journal of Physical Oceanography, 27(9):1849–1867, 1997.

[13] Elizabeth C Hunke and John K Dukowicz. The elastic–viscous–plastic sea ice dynamics model in general orthogonal curvilinear coordinates on a sphere—Incorporation of metric terms. Monthly Weather Review, 130(7):1848–1865, 2002.

[14] Elizabeth C Hunke, William H Lipscomb, AK Turner, N Jeffery, and Scott Elliott. CICE: the Los Alamos sea ice model documentation and software user’s manual version 4.1. Los Alamos National Laboratory, Los Alamos, NM, pages 1–115, 2010.

[15] Lili Ju, Todd Ringler, and Max Gunzburger. Voronoi tessellations and their application to climate and global modeling. In Numerical techniques for global atmospheric models, pages 313–342. Springer, 2011.
[16] Peter Korn. Formulation of an unstructured grid model for global ocean dynamics. *Journal of Computational Physics*, 339:525–552, 2017.

[17] Ronald Kwok and Norbert Untersteiner. The thinning of arctic sea ice. *Phys. Today*, 64(4):36–41, 2011.

[18] Lawrence E Malvern. *Introduction to the Mechanics of a Continuous Medium*. Number Monograph. 1969.

[19] Carolin Mehlmann, Sergey Danilov, Martin Losch, JF Lemieux, Nils Hutter, T Richter, P Blain, EC Hunke, and Peter Korn. Simulating linear kinematic features in viscous-plastic sea ice models on quadrilateral and triangular grids with different variable staggering. *Journal of Advances in Modeling Earth Systems*, 13(11):e2021MS002523, 2021.

[20] Carolin Mehlmann and Peter Korn. Sea-ice dynamics on triangular grids. *Journal of Computational Physics*, 428:110086, 2021.

[21] Mark R Petersen, Xylar S Asay-Davis, Anne S Berres, Qingshan Chen, Nils Feige, Matthew J Hoffman, Douglas W Jacobsen, Philip W Jones, Mathew E Maltrud, Stephen F Price, et al. An evaluation of the ocean and sea ice climate of E3SM using MPAS and interannual CORE-II forcing. *Journal of Advances in Modeling Earth Systems*, 11(5):1438–1458, 2019.

[22] Todd Ringler, Lili Ju, and Max Gunzburger. A multiresolution method for climate system modeling: Application of spherical centroidal Voronoi tessellations. *Ocean Dynamics*, 58(5-6):475–498, 2008.

[23] Todd Ringler, Mark Petersen, Robert L Higdon, Doug Jacobsen, Philip W Jones, and Mathew Maltrud. A multi-resolution approach to global ocean modeling. *Ocean Modelling*, 69:211–232, 2013.

[24] Todd D Ringler, John Thuburn, Joseph B Klemp, and William C Skamarock. A unified approach to energy conservation and potential vorticity dynamics for arbitrarily-structured C-grids. *Journal of Computational Physics*, 229(9):3065–3090, 2010.

[25] Ralph Timmermann, Sergey Danilov, Jens Schröter, Carmen Böning, Dmitry Sidorenko, and Katja Röllenhagen. Ocean circulation and sea ice distribution in a finite element global sea ice–ocean model. *Ocean Modelling*, 27(3-4):114–129, 2009.
[26] Ralph Timmermann, Hugues Goosse, Gurvan Madec, Thierry Fichefet, Christian Etche, and Valérie Dulière. On the representation of high latitude processes in the ORCA-LIM global coupled sea ice–ocean model. Ocean Modelling, 8(1-2):175–201, 2005.

[27] A. K. Turner, W. H. Lipscomb, E. C. Hunke, D. W. Jacobsen, N. Jeffery, D. Engwirda, T. D. Ringler, and J. D. Wolfe. MPAS-seaice (v1.0.0): Sea-ice dynamics on unstructured voronoi meshes. Geoscientific Model Development Discussions, 2021:1–46, 2021.

[28] Qiang Wang, Sergey Danilov, Dmitry Sidorenko, Ralph Timmermann, Claudia Wekerle, Xuezhu Wang, Thomas Jung, and Jens Schrötter. The Finite Element Sea Ice-Ocean Model (FESOM) v. 1.4: formulation of an ocean general circulation model. Geoscientific Model Development, 7(2):663–693, 2014.