# MPAS-Seaice: a new variable resolution sea-ice model

**Adrian K. Turner[1], William H. Lipscomb[1,2], Elizabeth C. Hunke[1], Douglas W. Jacobsen[1], Nicole Jeffery[3], Todd D. Ringler[1], and Jonathan D. Wolfe[1]**

[1]Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM, USA

[2]Climate and Global Dynamics Laboratory, National Center for Atmospheric Research, Boulder, CO, USA

[3]Computer and Computational Science Division, Los Alamos National Laboratory, Los Alamos, NM, USA

**Key Points:**

- We present a new sea-ice model capable of using variable resolution meshes
- MPAS-Seaice uses the Modeling for Prediction Across Scales framework
- MPAS-Seaice has been validated with idealized test cases and global simulations

Corresponding author: Adrian K. Turner, akt@lanl.gov

**Abstract**

The MPAS-Seaice sea-ice model is presented. MPAS-Seaice uses the Modeling for Prediction Across Scales (MPAS) framework and Spherical Centroidal Voronoi Tessellation (SCVT) unstructured meshes. The MPAS-Seaice velocity solver uses the variational discretization of the divergence of internal stress operator used by the CICE Sea-ice model, but adapted for the polygonal cells of MPAS meshes, or alternatively an integral ("weak") formulation of the divergence of internal stress operator, and the Elastic-Viscous-Plastic (EVP) rheology. An incremental remapping advection scheme is used for mass and tracer transport. These formulations are validated with idealized test cases. MPAS-Seaice shares the sophisticated column physics and biogeochemistry of CICE, and can be used with quadrilateral meshes to reproduce the results of CICE. Global simulations with realistic forcing have validated MPAS-Seaice against similar simulations with CICE and against observations. MPAS-Seaice is the current sea-ice component of the Energy Exascale Earth System Model (E3SM).

## 1 Introduction

Sea ice, the frozen surface of the sea at high latitudes, is an important component of the Earth climate system. Rejection of salt during sea-ice formation helps drive the thermohaline circulation [*Killworth*, 1983], and its high reflectivity has a significant impact on planetary albedo and can help drive the polar amplification of climate change through an albedo feedback mechanism [*Ingram et al.*, 1989]. Numerical modeling of sea-ice dynamics and thermodynamics is an important tool in understanding global climate. One of the most popular sea-ice models currently in use is CICE [*Hunke et al.*, 2015]. CICE approximates the sea-ice cover as a continuous fluid, and uses an Elastic-Viscous-Plastic (EVP) rheology to describe the relationship between stress and strain in that fluid [*Hunke and Dukowicz*, 1997]. This rheology adds a numerical elasticity to the Viscous-Plastic rheology of *Hibler III* [1979] to allow explicit time-stepping and parallelization of the algorithm. CICE uses a quadrilateral structured mesh, and has been used with both displaced pole [*Smith et al.*, 1995] and tripolar [*Murray*, 1996] grids. The strain rate and divergence of internal ice stress operators used by CICE are based on a variational principle [*Hunke and Dukowicz*, 1997] and account for metric effects in the curvilinear coordinates [*Hunke and Dukowicz*, 2002].

Recently, unstructured mesh climate models have been gaining popularity [e.g. *Ringler et al.*, 2013; *Wang et al.*, 2014]. Unstructured meshes allow the concentration of model degrees of freedom in regions of interest, improving computational efficiency, while avoiding the difficulties of the open boundary conditions present in limited-domain regional models. Several unstructured sea-ice models have been developed. *Hutchings et al.* [2004] proposed a finite-volume, cell-centered discretization. The Unstructured-Grid CICE [UG-CICE; *Gao et al.*, 2011] model, which is built on the FVCOM framework [*Chen et al.*, 2009], also uses a finite-volume formulation. In contrast, a finite-element discretization was proposed by *Lietaer et al.* [2008]. The Finite-Element Sea Ice Model [FESIM; *Danilov et al.*, 2015] also uses a finite-element discretization of its dynamics equations. Both UG-CICE and FESIM use triangular elements in their mesh.

Another unstructured modeling framework to be recently developed is the Modeling for Prediction Across Scales (MPAS) framework, which uses a Voronoi tessellation of the sphere to form a mesh [*Ringler et al.*, 2010]. Consequently, grid cells in MPAS meshes are polygons with four or more sides, rather than triangles. Several climate model components have been built with the MPAS modeling framework, including ocean [MPAS-O; *Ringler et al.*, 2013], atmosphere [MPAS-A; *Skamarock et al.*, 2012] and land ice [MPAS-LI; *Edwards et al.*, 2014] models. Here, we describe a new MPAS model for sea ice: MPAS-Seaice. This model uses a 'B' Arakawa type grid [*Arakawa and Lamb*, 1977] with sea-ice velocity on cell vertices and tracers at cell centers. Two discretizations of the momentum equation are implemented: one based on the variational scheme used by CICE [*Hunke and Dukowicz*, 2002] and another using the integral ("weak") form of the relevant operators. MPAS-Seaice uses the same EVP rheology and column physics as CICE. Sea-ice tracer transport uses an incremental remapping scheme [*Lipscomb and Hunke*, 2004; *Lipscomb and Ringler*, 2005] modified for the MPAS mesh. In section 2, we describe the modeling approach used in MPAS-Seaice, focusing on the solution of the sea-ice momentum equation and tracer transport. In section 3 we validate this new model, both with idealized test cases and with global simulations, and conclude in section 4. We focus in this paper on simulations with quasi-uniform global meshes: variable resolution meshes and computational performance will be considered in later publications.

## 2 Model description

### 2.1 The MPAS framework

The MPAS mesh consists of a Spherical Centroidal Voronoi Tessellation (SCVT), as described in *Ringler et al.* [2010], and consists of a primary mesh of the Voronoi cells tessellated on the sphere and a dual triangular Delaunay mesh, formed from joining the Voronoi cell centers into a triangulation. The primary mesh cells have their Voronoi generating points coincident with the centroid of the cell. The mesh consists of three types of points arranged on the sphere: the Voronoi cell center points, vertex points of the Voronoi cell and edge points at the midpoint of the Voronoi cell edges (see Figure 1). The mesh is constructed so that a line joining neighboring cell centers is perpendicular to the edge that line passes through. On a typical quasi-uniform MPAS mesh the majority of the cells are hexagons, but at least twelve pentagons are needed to complete the tessellation. In general, the cells are not regular polygons and may consist of polygons with edge numbers greater or equal to four. As well as quasi-uniform grids, meshes can be generated with regions of enhanced resolution, allowing computational effort to be focused in regions of interest. The MPAS mesh standard can also represent the quadrilateral meshes used by CICE, where four instead of three edges meet at each vertex. With these quadrilateral meshes the dual mesh is also quadrilateral.

### 2.2 Velocity solver

MPAS-Seaice uses a 'B' Arakawa type grid [*Arakawa and Lamb*, 1977] with both components of velocity defined at cell vertices and sea-ice concentration, volume and other tracers defined at cell centers (see Figure 1). When using CICE-like quadrilateral meshes, this allows the velocity solver algorithm of MPAS-Seaice to reduce to that of CICE, allowing CICE and MPAS-Seaice to use identical test cases and allow rapid testing and development.

In CICE the velocity components are aligned with the quadrilateral mesh. This is not possible, in general, with MPAS-Seaice since a SCVT MPAS mesh does not have edges with perpendicular directions as in a quadrilateral mesh. Instead, the velocity components at a given MPAS vertex are defined as eastwards ($u$) and northwards ($v$), irrespective of the orientation of edges joining that vertex. Such a definition, however, would result in a convergence of $v$ components at the geographic North Pole and strong metric

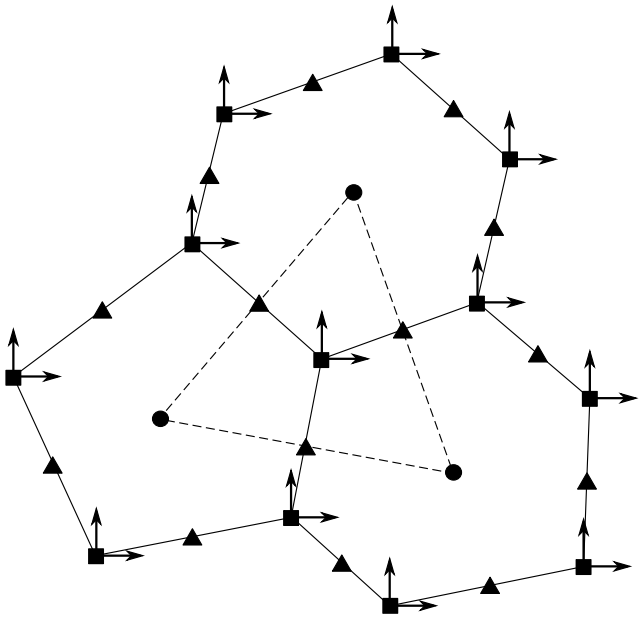

**Figure 1.** Schematic representation of three cells in an example MPAS mesh. The mesh is composed of cell centers (*circles*), cell edge points (*triangles*), and cell vertices (*squares*). The dual triangular Delaunay mesh is formed by joining cell centers (*dashed line*). MPAS-Seaice uses a 'B' type grid, with both velocity components defined at cell vertices. Unlike traditional quadrilateral meshes, the directions of the velocity components in MPAS-Seaice are not in general aligned with the cell edges.

terms in the velocity solution. Consequently, in addition, we rotate these definitions of $u$ and $v$ so that their pole lies on the geographical equator at 0° longitude.

To prognose sea-ice velocity we solve the same sea-ice momentum equation as CICE [*Hibler III*, 1979; *Hunke and Dukowicz*, 1997]:

$$m\frac{\partial \boldsymbol{u}}{\partial t} = \nabla \cdot \boldsymbol{\sigma} + \boldsymbol{\tau_a} + \boldsymbol{\tau_w} - \hat{\boldsymbol{k}} \times mf\boldsymbol{u} - mg\nabla H_o. \tag{1}$$

Here $m$ is the mass of snow and ice per unit area, $\boldsymbol{u}$ is the sea-ice velocity, $\boldsymbol{\sigma}$ is the ice internal stress tensor, $\boldsymbol{\tau_a}$ and $\boldsymbol{\tau_w}$ are the horizontal stresses due to atmospheric winds and ocean currents respectively, $\hat{\boldsymbol{k}}$ is the unit vector normal to the Earth surface, $f$ is the Coriolis parameter, $g$ is the acceleration due to gravity and $H_o$ is the ocean surface height. The second to last term represents the Coriolis force and the last term represents the force due to the ocean surface tilt. Only the divergence of internal stress and ocean surface tilt terms depend on horizontal differential operators. During coupled simulations the ocean model provides the ocean surface tilt term, whereas in non-coupled simulations we assume

that the ocean currents are in geostrophic balance so that

$$mg\boldsymbol{\nabla} H_o = mf\hat{\boldsymbol{k}} \times \boldsymbol{u_o} \tag{2}$$

where $\boldsymbol{u_o}$ is the ocean surface velocity. Consequently, only the divergence of internal stress depends on the properties of the horizontal grid employed, and only adaptations to this stress term are required to adapt the velocity solver of CICE to MPAS meshes. The other terms in the momentum equation are solved in an identical way to CICE.

Determination of the divergence of the internal stress can be broken down into three stages:

1. The strain rate tensor is determined from the velocity field.
2. The stress tensor at a point is determined, through a constitutive relation, from the strain rate tensor at that point.
3. The divergence of this stress tensor is calculated.

As in CICE we use an Elastic-Viscous-Plastic (EVP) rheology [*Hunke and Dukowicz*, 1997] for the constitutive relation. This step does not depend on the details of the horizontal mesh and we use the same formulation as CICE. We develop two schemes to calculate the strain rate tensor and the divergence of internal stress on MPAS meshes. A variational scheme is based on that used in CICE [*Hunke and Dukowicz*, 2002], whereas a weak scheme uses the line integral forms of the symmetric gradient and divergence operators. These schemes are described in the following sections.

### *2.2.1 Variational Scheme*

We develop a variational scheme for calculating the divergence of stress based on that of *Hunke and Dukowicz* [2002] but adapted for arbitrarily shaped and sided convex polygons. This scheme is based on the fact that over the entire domain, $\Omega$, and ignoring boundary effects, the total work done by the internal stress is equal to the dissipation of mechanical energy:

$$\int_\Omega \boldsymbol{u} \cdot (\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}) \mathrm{d}A = -\int_\Omega (\boldsymbol{\sigma_{11}\dot{\epsilon}_{11}} + 2\boldsymbol{\sigma_{12}\dot{\epsilon}_{12}} + \boldsymbol{\sigma_{22}\dot{\epsilon}_{22}}) \mathrm{d}A. \tag{3}$$

Here $\dot{\boldsymbol{\epsilon}}$ is the strain rate tensor and the integrals are area integrals over the whole model domain. The work done over the whole domain can be split into a sum over the contribution to the work done from each cell on the dual Delaunay mesh. Each dual cell on the

dual mesh consists of a triangle surrounding a single vertex point where the discretized velocity is defined. Equation 3 can then be written as

$$\sum_{i}^{n_d} \int_i \boldsymbol{u} \cdot (\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}) \mathrm{d}A = D(u_1, u_2, ..., u_n, v_1, v_2, ..., v_{n_d}) \tag{4}$$

where the left-side sum is over the $n_d$ cells of the dual mesh, the integral is an area integral over each dual cell, and the dissipation of mechanical energy has been written as a function of the discretized velocity components. Writing the two components of the divergence of stress as $F_u = (\nabla \cdot \sigma)_u$ and $F_v = (\nabla \cdot \sigma)_v$, then

$$\sum_{i}^{n_d} \int_i (uF_u + vF_v) \mathrm{d}A = D(u_1, u_2, ..., u_n, v_1, v_2, ..., v_{n_d}). \tag{5}$$

If we assume that within the dual cell the velocity is constant, it follows that

$$\sum_{i}^{n_d} (u_i F_{ui} + v_i F_{vi}) A_{ui} = D(u_1, u_2, ..., u_n, v_1, v_2, ..., v_{n_d}) \tag{6}$$

where $A_{ui}$ is the area of the dual mesh cell. The variation of these expressions with respect to the $u$ component of the discretized velocity at a particular vertex point $j$ is given by

$$\frac{\partial}{\partial u_j} \sum_{i}^{n_d} (u_i F_{ui} + v_i F_{vi}) A_{ui} = \frac{\partial}{\partial u_j} D(u_1, u_2, ..., u_n, v_1, v_2, ..., v_{n_d}) \tag{7}$$

Assuming $F_u$ and $F_v$ are not functions of velocity,

$$F_{uj} = \frac{1}{A_{uj}} \frac{\partial}{\partial u_j} D(u_1, u_2, ..., u_n, v_1, v_2, ..., v_{n_d}). \tag{8}$$

$F_v$ is obtained in a similar way by taking the variation of $D$ with respect to $v_j$. The dissipation of mechanical energy, $D$, can be split into three terms:

$$D = D_1 + D_2 + D_3 \tag{9}$$

with

$$D_1 = -\int \boldsymbol{\sigma_{11}} \boldsymbol{\dot{\epsilon}_{11}} \mathrm{d}A, \quad D_2 = -\int 2\boldsymbol{\sigma_{12}} \boldsymbol{\dot{\epsilon}_{12}} \mathrm{d}A, \quad D_3 = -\int \boldsymbol{\sigma_{22}} \boldsymbol{\dot{\epsilon}_{22}} \mathrm{d}A. \tag{10}$$

We will calculate the contribution to $F_u$ and $F_v$ from $D_1$. Similar contributions come from $D_2$ and $D_3$. Using the expression for $\dot{\epsilon}_{11}$ in terms of the velocity components and latitude $\phi$, $D_1$ becomes

$$D_1 = -\int \sigma_{11} \left[ \frac{\partial u}{\partial x} - \frac{v \tan \phi}{r} \right] \mathrm{d}A \tag{11}$$

where $x$ and $y$ are locally Cartesian coordinates, with $x$ in the rotated due eastwards direction and $y$ in the rotated due northwards direction, $\phi$ is the latitude, and $r$ is the radius of the Earth. The second term in $\dot{\epsilon}$ accounts for the metric effects of the curved domain

[*Batchelor*, 1967]. The integral can be broken up into a sum over the $n_p$ cells in the primary mesh:

$$D_1 = -\sum_k^{n_p} \int_k \sigma_{11} \left[ \frac{\partial u}{\partial x} - \frac{v \tan\phi}{r} \right] \mathrm{d}A \tag{12}$$

where the integral is over the interior area of the $k$th cell. To perform this integral we use a set of basis functions, $\mathcal{W}_l$, to represent functions within a cell of the primary mesh. If a function, $\psi$, has a value of $\psi_l$ at vertex $l$ of a cell, then the value of the function at a position $(x, y)$ within the cell can be approximated as

$$\psi(x, y) = \sum_l^{n_v} \psi_l \mathcal{W}_l(x, y) \tag{13}$$

where the sum is over the $n_v$ vertices of the cell in the primary mesh. Using those basis functions, equation 12 can be written as

$$D_1 = -\sum_k^{n_p} \int_k \left[ \sum_l^{n_v} \sigma_{11l} \mathcal{W}_l \cdot \sum_m^{n_v} \left( u_m \frac{\partial \mathcal{W}_m}{\partial x} - \frac{\tan\phi}{r} v_m \mathcal{W}_m \right) \right] \mathrm{d}A \tag{14}$$

where the derivative with respect to $x$ has been taken inside the summation. Rearranging

$$D_1 = -\sum_k^{n_p} \sum_l^{n_v} \sum_m^{n_v} \sigma_{11l} \left( u_m \int_k \mathcal{W}_l \frac{\partial \mathcal{W}_m}{\partial x} \mathrm{d}A - \frac{\tan\phi}{r} v_m \int_k \mathcal{W}_l \mathcal{W}_m \mathrm{d}A \right). \tag{15}$$

In moving the integral, we have assumed that $\phi$, the latitude, is constant in the cell. The terms involving integrals are now only a function of the geometry of the mesh and can be calculated once during the initialization phase of the model run. Defining

$$\mathcal{S}^x_{lm} = \int_k \mathcal{W}_l \frac{\partial \mathcal{W}_m}{\partial x} \mathrm{d}A \tag{16}$$

and

$$\mathcal{T}_{lm} = \int_k \mathcal{W}_l \mathcal{W}_m \mathrm{d}A. \tag{17}$$

we have

$$D_1 = -\sum_k^{n_p} \sum_l^{n_v} \sum_m^{n_v} \sigma_{11l} \left( u_m \mathcal{S}^x_{lm} - \frac{\tan\phi}{r} v_m \mathcal{T}_{lm} \right). \tag{18}$$

Taking the variation with respect to a discretized velocity component at a particular vertex point, $j$, as in equation 8, now gives us the contribution from $D_1$ to the components of the divergence of stress tensor at that velocity point:

$$(\nabla \cdot \sigma)^{D_1}_{u_j} = \frac{\delta D_1}{\delta u_j} = -\sum_k^{n_p} \sum_l^{n_v} \sigma_{11l} \mathcal{S}^x_{lj} \tag{19}$$

$$(\nabla \cdot \sigma)^{D_1}_{v_j} = \frac{\delta D_1}{\delta v_j} = \sum_k^{n_p} \sum_l^{n_v} \sigma_{11l} \frac{\tan\phi}{r} \mathcal{T}_{lj} \tag{20}$$

Only cells that border the vertex point $j$ contribute to the $k$ sum over cells. The total divergence of stress at the point $j$ is then the sum from the contributions from $D_1$, $D_2$, and $D_3$:

$$(\nabla \cdot \sigma)_{u_j} = (\nabla \cdot \sigma)_{u_j}^{D_1} + (\nabla \cdot \sigma)_{u_j}^{D_2} + (\nabla \cdot \sigma)_{u_j}^{D_3} \tag{21}$$

$$(\nabla \cdot \sigma)_{v_j} = (\nabla \cdot \sigma)_{v_j}^{D_1} + (\nabla \cdot \sigma)_{v_j}^{D_2} + (\nabla \cdot \sigma)_{v_j}^{D_3}. \tag{22}$$

All that remains now is to determine the stress for each cell at its vertices. As in the formulation in CICE, each cell has its own values of the stress at its vertices, so each vertex has several values of the stress, each corresponding to a different surrounding cell. The stresses are calculated from the strain rate tensor at each vertex using the constitutive relation. Including metric effects [*Batchelor*, 1967] the strain rate tensor is given by:

$$\dot{\epsilon}_{11} = \frac{\partial u}{\partial x} - \frac{v \tan \phi}{r} \tag{23}$$

$$\dot{\epsilon}_{22} = \frac{\partial v}{\partial y} \tag{24}$$

$$\dot{\epsilon}_{12} = \frac{1}{2}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) + \frac{u \tan \phi}{2r}. \tag{25}$$

The strain rate tensor at cell vertex $l$ is then given by

$$\dot{\epsilon}_{11l} = \sum_m^{n_v} u_m \left.\frac{\partial \mathcal{W}_m}{\partial x}\right|_l - \frac{v_l \tan \phi_l}{r} \tag{26}$$

$$\dot{\epsilon}_{22l} = \sum_m^{n_v} v_m \left.\frac{\partial \mathcal{W}_m}{\partial y}\right|_l \tag{27}$$

$$\dot{\epsilon}_{12l} = \frac{1}{2}\left(\sum_m^{n_v} u_m \left.\frac{\partial \mathcal{W}_m}{\partial y}\right|_l + \sum_m^{n_v} v_m \left.\frac{\partial \mathcal{W}_m}{\partial x}\right|_l\right) + \frac{u_l \tan \phi_l}{2r} \tag{28}$$

The derivatives of the basis functions are taken at cell vertex $l$.

In MPAS-Seaice we provide two options for the choice of basis functions, $\mathcal{W}_l$: Wachspress basis functions and Piece-Wise Linear (PWL) basis functions. Both basis functions have a value of one on vertex $l$ and zero on the other vertices of a cell, and are linear on the cell boundaries. The Wachspress basis functions are smooth rational polynomials in the cell interior [*Dasgupta*, 2003]. The integrals of the Wachspress basis function within a cell are performed using the eighth order quadrature rules of *Dunavant* [1985]. PWL basis functions divide the polygonal cell into sub-triangles and use a linear basis within each sub-triangle [*Bailey et al.*, 2008]. To divide the polygonal cell into sub-triangles, a point is chosen within the cell and sub-triangles formed using this point and two adjacent vertices. The central point in the cell, $\mathbf{x}_c$, is chosen as

$$\mathbf{x}_c = \sum_i^{n_v} \alpha_i \mathbf{x}_i \tag{29}$$

where the sum is over the $n_v$ vertices of the cell each with position $\mathbf{x}_i$. The simplest choice for the $\alpha_i$ is to set them all equal to the inverse of the number of cell vertices, $1/n_v$. For quadrilateral meshes the Wachspress basis functions reduce to the bilinear basis functions used in CICE.

### *2.2.2 Weak Scheme*

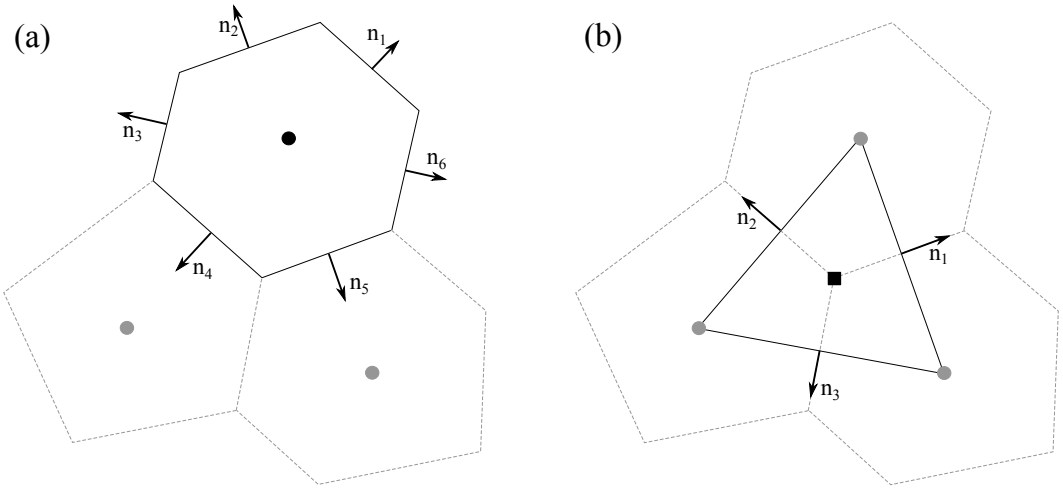


**Figure 2.** Contour integration lines used by the weak scheme. (a): Strain rate at cell centers (*circle*) are calculated from line integrals around primary mesh cells (*solid line*). (b): Divergence of stress at cell vertices (*square*) are calculated from line integrals around the dual mesh cells (*solid line*). Directions of normal vectors used in the integrals are shown for both figures.

For the weak scheme we use line integrals around cells in the primary and dual meshes to calculate the strain rate tensor and the divergence of stress, respectively. To determine the strain rate tensor we start from the following vector identity:

$$\boldsymbol{\nabla} \cdot (\boldsymbol{u} \otimes \boldsymbol{v}) = (\boldsymbol{u} \cdot \boldsymbol{\nabla})\boldsymbol{v} + (\boldsymbol{\nabla} \cdot \boldsymbol{u})\boldsymbol{v} \tag{30}$$

and from the divergence theorem:

$$\int_\Omega [\boldsymbol{\nabla} \cdot (\boldsymbol{u} \otimes \boldsymbol{v})]\, \partial\Omega = \oint_S [\boldsymbol{n} \cdot (\boldsymbol{u} \otimes \boldsymbol{v})]\, \partial S = \oint_S [(\boldsymbol{n} \cdot \boldsymbol{u})\boldsymbol{v}]\, \partial S \tag{31}$$

where $\boldsymbol{n}$ is a normal vector to the surface $S$ and $\otimes$ is the tensor product. If equation 30 is integrated over $\Omega$, using equation 31 we obtain

$$\int_\Omega [(\boldsymbol{u} \cdot \boldsymbol{\nabla})\boldsymbol{v} + (\boldsymbol{\nabla} \cdot \boldsymbol{u})\boldsymbol{v}]\, \partial\Omega = \oint_S [(\boldsymbol{n} \cdot \boldsymbol{u})\boldsymbol{v}]\, \partial S \tag{32}$$

If $\boldsymbol{u}$ is chosen as constant then $\boldsymbol{\nabla} \cdot \boldsymbol{u}$ vanishes as does the second term in equation 32. Taking, also, $\boldsymbol{u}$ sequentially as the cartesian unit vectors spanning $\Omega$ and summing the results we obtain

$$\int_{\Omega} [\boldsymbol{\nabla} \boldsymbol{v}] \, \partial\Omega = \oint_{S} [\boldsymbol{n} \otimes \boldsymbol{v}] \, \partial S \tag{33}$$

The symmetric version of this operator is then obtained as:

$$\int_{\Omega} [\boldsymbol{\nabla}_{\boldsymbol{S}} \boldsymbol{v}] \, \partial\Omega = \oint_{S} [\boldsymbol{n} \otimes \boldsymbol{v} + \boldsymbol{v} \otimes \boldsymbol{n}] \, \partial S \tag{34}$$

The strain rate at a point is then obtained from the limit

$$\dot{\boldsymbol{\epsilon}} = \boldsymbol{\nabla}_S \boldsymbol{v} = \lim_{A \to 0} \frac{1}{A} \oint \frac{1}{2} [\boldsymbol{n} \otimes \boldsymbol{v} + \boldsymbol{v} \otimes \boldsymbol{n}] \, \mathrm{d}l \tag{35}$$

where the integral is around a closed loop with area $A$ and normal vector $\boldsymbol{n}$, and $\boldsymbol{v}$ is the sea-ice velocity. To determine the strain rate tensor at the centers of the primary mesh, we take this integration around the edges of the cells in the primary mesh. First the cell is projected onto a flat tangent plane perpendicular to the vector joining the center of the sphere to the cell center. We take the sea ice velocity at a cell edge as the average of the values on the two vertices forming that edge projected onto the tangent plane:

$$\dot{\boldsymbol{\epsilon}}' = \frac{1}{A} \sum_{i}^{n_e} \frac{1}{2} [\boldsymbol{n}_i \otimes \boldsymbol{v}_i + \boldsymbol{v}_i \otimes \boldsymbol{n}_i] \, l_i \tag{36}$$

Here, $A$ is the area of the primary cell, the summation is over the $n_e$ edges of the primary cell, $\boldsymbol{n}_i$ is the normal vector to the edge $i$ that lies in the tangent plane, $\boldsymbol{v}_i$ is the edge velocity and $l_i$ is the length of edge $i$. We use the tangental projection of the velocity and account for metric terms separately. The full strain rate tensor including these metric terms is [*Batchelor*, 1967]:

$$\dot{\epsilon}_{11} = \dot{\epsilon}'_{11} - \frac{v \tan\phi}{r} \tag{37}$$

$$\dot{\epsilon}_{22} = \dot{\epsilon}'_{22} \tag{38}$$

$$\dot{\epsilon}_{12} = \dot{\epsilon}'_{12} + \frac{u \tan\phi}{2r} \tag{39}$$

where the prime symbol signifies a strain rate without metric terms. The stress, which is determined from the strain rate tensor using the constitutive relation, is now defined on cell centers. To find its divergence we use the divergence theorem:

$$\iint \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} \mathrm{d}A = \oint [\boldsymbol{\sigma} \cdot \boldsymbol{n}] \, \mathrm{d}l \tag{40}$$

or

$$\boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = \lim_{A \to 0} \frac{1}{A} \oint [\boldsymbol{\sigma} \cdot \boldsymbol{n}] \, \mathrm{d}l \tag{41}$$

for the divergence of stress at a point. The divergence of internal stress is determined at primary cell vertices (where the velocity is defined and momentum equation solved) by performing a sum around the edges of the dual mesh on a tangent projected plane, tangental to the primary cell vertex. The vertices of the dual mesh are the cell centers of the primary mesh where the strain rate has been determined. The divergence of stress at primary cell vertices is then given by

$$(\boldsymbol{\nabla} \cdot \boldsymbol{\sigma})' = \frac{1}{A_d} \sum_{i}^{n_c} [\boldsymbol{\sigma}_i \cdot \boldsymbol{n}_i] \, l_i \tag{42}$$

where $A_d$ is the area of the dual mesh cell, the sum is over the $n_c$ vertices of the dual mesh, $l_i$ is the length of the $i$ edge of the dual mesh, and $n_i$ is a normal vector to the $i$ edge on the projected plane. As before, this gives a result without taking into account metric effects of the mesh. With those effects the divergence of stress is:

$$(\nabla \cdot \sigma)_u = (\nabla \cdot \sigma)'_u - \frac{2\sigma_{12} \tan \phi}{r} \tag{43}$$

$$(\nabla \cdot \sigma)_v = (\nabla \cdot \sigma)'_v + \frac{(\sigma_{11} + \sigma_{22}) \tan \phi}{r} \tag{44}$$

where the components of $\boldsymbol{\sigma}$ are approximated as the average of the values on the dual mesh vertices.

### 2.3 Transport

To transport sea ice fractional area and various tracers, MPAS-Seaice uses an incremental remapping (IR) algorithm similar to that described by *Dukowicz and Baumgardner* [2000], *Lipscomb and Hunke* [2004] (henceforth LH04) and *Lipscomb and Ringler* [2005] (henceforth LR05). The LH04 scheme was designed for structured quadrilateral meshes and is implemented in CICE [*Hunke et al.*, 2015]. The LR05 scheme was implemented on a structured SCVT global mesh consisting of quasi-regular hexagons and 12 pentagons.

For MPAS-Seaice the IR scheme was generalized to work on either the standard MPAS mesh (hexagons and other n-gons of varying sizes, with a vertex degree of 3 as in LR05) or a quadrilateral mesh (with a vertex degree of 4 as in LH04 and CICE). Since MPAS meshes are unstructured, the IR scheme had to be rewritten from scratch. Most of the code is mesh-agnostic, but a small amount of code is specific to quad meshes as noted below.

Here we review the conceptual framework of incremental remapping as in [*Hunke et al.*, 2015] and describe features specific to the MPAS-Seaice implementation. IR is de-

signed to solve equations of the form

$$\frac{\partial m}{\partial t} = -\nabla \cdot (\mathbf{u} m) \tag{45}$$

$$\frac{\partial (m T_1)}{\partial t} = -\nabla \cdot (\mathbf{u} m T_1), \tag{46}$$

$$\frac{\partial (m T_1 T_2)}{\partial t} = -\nabla \cdot (\mathbf{u} m T_1 T_2), \tag{47}$$

$$\frac{\partial (m T_1 T_2 T_3)}{\partial t} = -\nabla \cdot (\mathbf{u} m T_1 T_2 T_3), \tag{48}$$

where $\mathbf{u} = (x, y)$ is the horizontal velocity, $m$ is mass or a mass-like field (such as density or fractional sea ice concentration), and $T_1$, $T_2$ and $T_3$ are tracers. These equations describe conservation of quantities such as mass and internal energy under horizontal transport. Sources and sinks of mass and tracers (e.g., ice growth and melting) are treated separately from transport.

In MPAS-Seaice, the fractional ice area in each thickness category is a mass-like field whose transport is described by (45). (Henceforth, "area" refers to fractional ice area unless stated otherwise.) Ice and snow thickness, among other fields, are type 1 tracers obeying equations of the form (46), and the ice and snow enthalpy in each vertical layer are type 2 tracers obeying equations like (47), with ice or snow thickness as their parent tracer. When run with advanced options (e.g., active melt ponds and biogeochemistry), MPAS-Seaice advects tracers up to type 3. Thus, the mass-like field is the "parent field" for type 1 tracers; type 1 tracers are parents of type 2; and type 2 tracers are parents of type 3.

Incremental remapping has several desirable properties for sea ice modeling:

- It is conservative to within machine roundoff.
- It preserves tracer monotonicity. That is, transport produces no new local extrema in fields like ice thickness or internal energy.
- The reconstructed mass and tracer fields vary linearly in x and y. This means that remapping is second-order accurate in space, except where gradients are limited locally to preserve monotonicity.
- There are economies of scale. Transporting a single field is fairly expensive, but additional tracers have a low marginal cost, especially when all tracers are transported with a single velocity field as in CICE and MPAS-Seaice.

The time step is limited by the requirement that trajectories projected backward from vertices are confined to the cells sharing the vertex (i.e., 3 cells for the standard MPAS mesh and 4 for the quad mesh). This is what is meant by incremental as opposed to general remapping. This requirement leads to a CFL-like condition,

$$\frac{\max(|\mathbf{u}|\Delta t)}{\Delta x} \leq 1, \tag{49}$$

where $\Delta x$ is the grid spacing and $\Delta t$ is the time step. For highly divergent velocity fields, the maximum time step may have to be reduced by a factor of 2 to ensure that trajectories do not cross.

The IR algorithm consists of the following steps:

- Given mean values of the ice area and tracer fields in each grid cell and thickness category, construct linear approximations of these fields. Limit the field gradients to preserve monotonicity.
- Given ice velocities at grid cell vertices, identify departure regions for the transport across each cell edge. Divide these departure regions into triangles and compute the coordinates of the triangle vertices.
- Integrate the area and tracer fields over the departure triangles to obtain the area, volume, and other conserved quantities transported across each cell edge.
- Given these transports, update the area and tracers.

Since all fields are transported by the same velocity field, the second step is done only once per time step. The other steps are repeated for each field.

With advanced physics and biogeochemistry (BGC) options, MPAS-Seaice can be configured to include up to ~40 tracer fields, each of which is advected in every thickness category, and many of which are defined in each vertical ice or snow layer. In order to accommodate different tracer combinations and make it easy to add new tracers, the tracer fields are organized in a linked list that depends on which physics and BGC packages are active. The list is arranged with fractional ice area first, followed by the type 1 tracers, type 2 tracers, and finally type 3 tracers. In this way, values computed for parent tracers are always available when needed for computations involving child tracers.

We next describe the IR algorithm in detail, pointing out features that are new in MPAS-Seaice.

### *2.3.1 Reconstructing area and tracer fields*

The fractional ice area and all tracers are reconstructed in each grid cell (quadilaterals, hexagons or other $n$-gons) as functions of $\mathbf{r} = (x, y)$ in a cell-based coordinate system. On a sphere, $\mathbf{r}$ lies in a local plane that is tangent to the sphere at the cell center. The state variable for ice area, denoted as $\bar{a}$, should be recovered as the mean value when integrated over the cell:

$$\int_A a(x, y) dA = \bar{a} A_C, \tag{50}$$

where $A_C$ is the grid cell area. Equation 50 is satisfied if $a(\mathbf{r})$ has the form

$$a(\mathbf{r}) = \bar{a} + \alpha_a \nabla a \cdot (\mathbf{r} - \bar{\mathbf{r}}), \tag{51}$$

where $\nabla a$ is a cell-centered gradient, $\alpha_a$ is a coefficient between 0 and 1 that enforces monotonicity, and $\mathbf{r}$ is the cell centroid:

$$\bar{\mathbf{r}} = \frac{1}{A_C} \int_A \mathbf{r} dA. \tag{52}$$

Similarly, tracer means should be recovered when integrated over a cell:

$$\begin{aligned} \int_A a(\mathbf{r}) T_1(\mathbf{r}) dA &= \bar{a} \bar{T}_1 A_C, \\ \int_A a(\mathbf{r}) T_1(\mathbf{r}) T_2(\mathbf{r}) dA &= \bar{a} \bar{T}_1 \bar{T}_2 A_C, \\ \int_A a(\mathbf{r}) T_1(\mathbf{r}) T_2(\mathbf{r}) T_3(\mathbf{r}) dA &= \bar{a} \bar{T}_1 \bar{T}_2 \bar{T}_3 A_C. \end{aligned} \tag{53}$$

These equations are satisfied when the tracers are reconstructed as

$$\begin{aligned} T_1(\mathbf{r}) &= \bar{T}_1 + \alpha_{T1} \nabla T_1 \cdot (\mathbf{r} - \tilde{\mathbf{r}}_1), \\ T_2(\mathbf{r}) &= \bar{T}_2 + \alpha_{T2} \nabla T_2 \cdot (\mathbf{r} - \tilde{\mathbf{r}}_2), \\ T_3(\mathbf{r}) &= \bar{T}_3 + \alpha_{T3} \nabla T_3 \cdot (\mathbf{r} - \tilde{\mathbf{r}}_3), \end{aligned} \tag{54}$$

where the tracer barycenter coordinates $\tilde{\mathbf{r}}_n$ are given by

$$\begin{aligned} \tilde{\mathbf{r}}_1 &= \frac{1}{\bar{a} A_C} \int_A \mathbf{r} a dA, \\ \tilde{\mathbf{r}}_2 &= \frac{1}{\bar{a} \bar{T}_1 A_C} \int_A \mathbf{r} a T_1 dA, \\ \tilde{\mathbf{r}}_3 &= \frac{1}{\bar{a} \bar{T}_1 \bar{T}_2 A_C} \int_A \mathbf{r} a T_1 T_2 dA. \end{aligned} \tag{55}$$

The integrals in (55) can be evaluated by applying quadrature rules for linear, quadratic and cubic polynomials as described in Section 2.3.3.

Monotonicity is enforced by van Leer limiting [*van Leer*, 1979]. The reconstructed area and tracers are evaluated at cell vertices, and the coefficients $\alpha$ are reduced as needed

so that the reconstructed values lie within the range of the mean values in the cell and its neighbors. When $\alpha = 1$, the reconstruction is second-order accurate in space. When $\alpha = 0$, the reconstruction reduces locally to first-order.

### *2.3.2 Locating departure triangles*

The next step is to identify the departure region associated with fluxes across each cell edge, and to divide the departure region into triangles. Figure 3a illustrates the geometry for the standard MPAS mesh. The edge has vertices $V1$ and $V2$. Each edge is oriented such that one adjacent cell ($C1$) is defined to lie in the left half-plane and the other ($C2$) in the right half-plane. The departure points $D1$ and $D2$ are found by projecting velocities backward from $V1$ and $V2$. The shaded departure region is a quadrilateral containing all the ice transported across the edge in one time step. In addition to $C1$ and $C2$, the departure region can include side cells $C3$ and $C4$. The side cells share edges $E1$ to $E4$ and vertices $V3$ to $V6$ with the central cells $C1$ and $C2$.

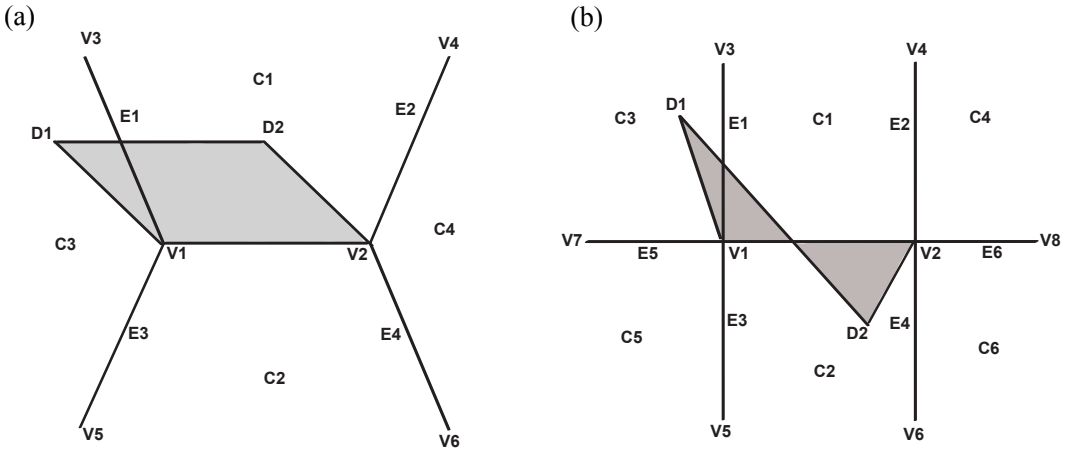


**Figure 3.** (a): Schematic showing transport across a cell edge on a standard MPAS mesh with 3 edges meeting at each vertex. The letters $C$, $E$ and $V$ denote cell centers, edges and vertices, respectively. Points $D1$ and $D2$ are backward trajectories, and the departure region is shaded. (b): Schematic showing transport across a cell edge on a quadrilateral MPAS mesh with 4 edges meeting at each vertex.

The edges and vertices in Figure 3a are defined in a coordinate system lying in the local tangent plane at the midpoint of the main edge, halfway between $V1$ and $V2$. These coordinates are pre-computed at initialization. During each time step, departure triangles are found by locating $D1$ and $D2$ in this coordinate system, and then looping through the

edges to identify any intersections of line segment $\overline{D12}$ (i.e., the segment joining $D1$ and $D2$) with the various edges. If $\overline{D12}$ intersects the main edge, then the departure region consists of two triangles (one each in $C1$ and $C2$) rather than a quadrilateral (as shown in the figure). If $\overline{D12}$ intersects any of edges $E1$ to $E4$, the departure region includes triangles in side cells.

Each departure triangle lies in a single grid cell, and there are at most four such triangles. There are two triangles in the central cells (either one each in $C1$ and $C2$, or a quadrilateral that can be split into two triangles), and up to two triangles in side cells. The triangle vertices are a combination of cell vertices ($V1$ and $V2$), departure points ($D1$ and $D2$), and intersection points (points where $\overline{D12}$ crosses an edge).

Figure 3b shows the geometry for a quadrilateral mesh. In this figure the departure region consists of two triangles, but it could also be a quadrilateral as in Figure 3a. For the quad mesh there are two additional side cells ($C5$ and $C6$), edges ($E5$ and $E6$) and vertices ($V7$ and $V8$). The search algorithm is designed such that the code used to find departure triangles for the standard mesh is also applied to the quad mesh. For quad meshes only, there is additional logic to find intersection points and triangles associated with the extra edges and cells. This is the only mesh-specific code in the run-time IR code. For the quad mesh there are at most six departure triangles: two in the central cells and one in each of the four side cells. If the edges meet at right angles as shown in the figure, the maximum is five triangles, but this is not a mesh requirement.

Once triangle vertices have been found in edge-based coordinates, they are transformed to cell-based coordinates, i.e., coordinates in the local tangent plane of the cell containing each triangle. (Coefficients for these transformations are computed at initialization.) Triangle areas are computed as

$$A_T = \frac{1}{2} \left| (x_2 - x_1)(y_3 - y_1) - (y_2 - y_1)(x_3 - x_1) \right| . \tag{56}$$

### *2.3.3 Integrating the transport*

Next, ice area and area-tracer products are integrated in each triangle. The integrals have the form (50) for area and (53) for tracers. Since each field is a linear function of $(x, y)$ as in (51) and (54), the area-tracer products are quadratic, cubic and quartic polynomials, respectively, for tracers of type 1, 2 and 3.

The integrals can be evaluated exactly by summing over values at quadrature points in each triangle. Polynomials of quadratic or lower order are integrated using the formula

$$I = \frac{A_T}{3} \sum_{i=1}^{3} f(\mathbf{x'}_i). \tag{57}$$

The quadrature points are located at $\mathbf{x'}_i = (\mathbf{x}_0 + \mathbf{x}_i)/2$, where $\mathbf{x_0}$ is the triangle midpoint and $\mathbf{x_i}$ are the three vertices. The products involving type-2 and type-3 tracers are cubic and quadratic polynomials, which can be evaluated using a similar formula with 6 quadrature points:

$$I = A_T \left[ w_1 \sum_{i=1}^{3} f(\mathbf{x}_{1i}) + w_2 \sum_{i=1}^{3} f(\mathbf{x}_{2i}) \right], \tag{58}$$

where $\mathbf{x_{1i}}$ and $\mathbf{x_{2i}}$ are two sets of three quadrature points, arranged symmetrically on trisecting medians of the triangle, and $w_1$ and $w_2$ are weighting factors. Coefficients and weighting factors for these and other symmetric quadrature rules for triangles were computed by *Dunavant* [1985]. These integrals are computed for each triangle and summed over edges to give fluxes of ice area and area-tracer products across each edge.

#### *2.3.4 Updating area and tracer fields*

The area transported across edge $k$ for a given cell can be denoted as $\Delta a_k$, and the area-tracer products as $\Delta(aT_1)_k$, $\Delta(aT_1T_2)_k$ and $\Delta(aT_1T_2T_3)_k$. The new ice area at time $n+1$ is given by

$$a^{n+1} = a^n + \frac{1}{A_C} \sum_k \pm \Delta a_k, \tag{59}$$

where the sum is taken over cell edges $k$, with a positive sign denoting transport into a cell and a negative sign denoting outward transport. The new tracers are given by

$$\begin{aligned}
T_1^{n+1} &= \frac{a^n T_1^n + \frac{1}{A_C} \sum\limits_k \pm \Delta(aT_1)_k}{a^{n+1}}, \\
T_2^{n+1} &= \frac{a^n T_1^n T_2^n + \frac{1}{A_C} \sum\limits_k \pm \Delta(aT_1T_2)_k}{a^{n+1} T_1^{n+1}}, \\
T_3^{n+1} &= \frac{a^n T_1^n T_2^n T_3^n + \frac{1}{A_C} \sum\limits_k \pm \Delta(aT_1T_2T_3)_k}{a^{n+1} T_1^{n+1} T_2^{n+1}}.
\end{aligned} \tag{60}$$

*Dukowicz and Baumgardner* [2000] showed that (60) satisfies tracer monotonicity, since the new-time tracer values are area-weighted averages of old-time values.

### 2.4 Column physics

CICE has sophisticated vertical physics and biogeochemical schemes, which include vertical thermodynamics schemes [*Bitz and Lipscomb*, 1999; *Turner et al.*, 2013; *Turner*

*and Hunke*, 2015], several melt-pond parameterizations [*Flocco et al.*, 2010; *Holland et al.*, 2012; *Hunke et al.*, 2013], a delta-Eddington radiation scheme [*Briegleb and Light*, 2007; *Holland et al.*, 2012], schemes for transport in thickness space [*Lipscomb*, 2001], representations of mechanical redistribution [*Lipscomb et al.*, 2007], and sea-ice biogeochemistry [*Jeffery and Hunke*, 2014; *Jeffery et al.*, 2016; *Duarte et al.*, 2017]. To include these developments in MPAS-Seaice, the column physics and biogeochemistry in CICE has been extracted into a separate library, which performs column calculations on individual grid cells with no knowledge of the details of the horizontal mesh used by the host model. This column package, now called IcePack, is used by both CICE and MPAS-Seaice.

## 3 Model validation

### 3.1 Velocity solver

To validate our implementation of the MPAS-Seaice velocity solver we investigate several idealized test cases. The first involves determining the divergence of internal stress on a unit sphere, from an analytical velocity field, assuming a simple linear constitutive relation of the form $\boldsymbol{\sigma} = \dot{\boldsymbol{\epsilon}}$. The expected analytical divergence of stress can also be calculated and compared to the results of MPAS-Seaice, thus validating the algorithms for calculating the strain rate and divergence of stress outlined in section 2.2. We use spherical harmonic functions $Y$ for the analytical velocity fields with $u(\theta, \phi) = Y_{l=5}^{m=3}(\theta, \phi)$ and $v(\theta, \phi) = Y_{l=4}^{m=2}(\theta, \phi)$ where $u$, $v$, the latitude, $\phi$, and the longitude, $\theta$, are on the rotated geographical mesh (see section 2.1). Figure 4 shows a histogram of errors between the $u$ component of the analytical divergence of stress calculated for the above spherical harmonic velocity fields and that determined for MPAS-Seaice. All three velocity solver options described in section 2.2 are shown. In general the errors are low, except higher-error tails in the distribution are present for both variational scheme methods. To determine the cause of these tails in the error distribution we examine spatial maps of the error for the three schemes. We find that errors are largest, as expected, around the pentagonal cells present in the mesh. Figure 5 shows the error around one of the pentagonal cells for the three methods. Both variational schemes have significantly larger errors here than the weak scheme resulting in the tails in the error distribution. Convergence of error with grid resolution is an important property of numerical discretizations. We calculated the $L_2$ error norm for regions of the grid with latitude, $|\phi| > 20^\circ$ (so that the poles of the rotated mesh were excluded) for four different grid resolutions. The $L_2$ error norm was calculated

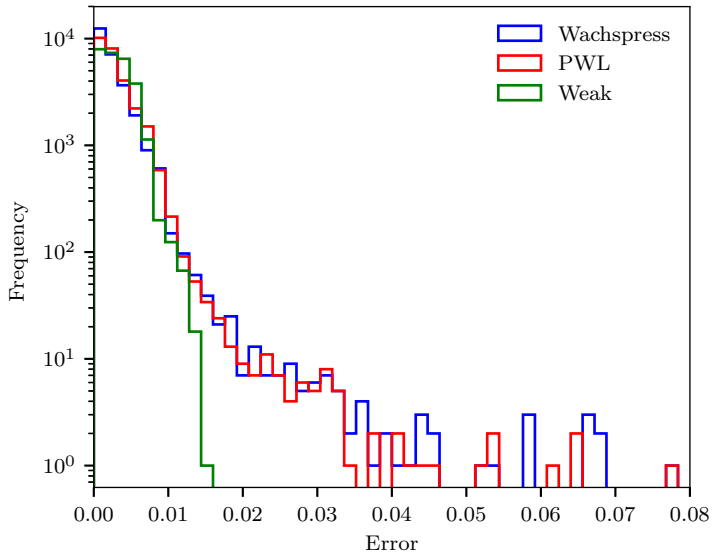


**Figure 4.** Error in the calculation of divergence of stress from an analytical velocity field for a spherical test case. The error is given as the absolute value of the difference between model and analytical stress divergence divided by the maximum analytical stress divergence on the sphere. The results are plotted as histograms for the two variational schemes (using Wachspress and PWL basis functions) and the weak scheme.

from

$$L_2 = \sqrt{\frac{\sum A_{di}(f_i - \hat{f}_i)^2}{\sum A_{di}\hat{f}^2}} \tag{61}$$

where the sum is over grid vertices in the region of interest, $A_{di}$ is the area of the dual cell surrounding vertex $i$, and $f_i$ and $\hat{f}_i$ are the $u$ components of the divergence of stress for the model results and analytical calculation respectively. Figure 6 shows that the $L_2$ error norm for the above test case scales linearly with grid resolution for all three velocity solver methods.

Since the MPAS framework and MPAS-Seaice support quadrilateral grids, direct comparisons can be made between MPAS-Seaice and CICE. For idealized planar test cases it is possible to set up MPAS-Seaice to be have a virtually identical velocity solver algorithm to CICE. This is achieved by using the Wachspress basis function with the variational scheme and defining the $u$ and $v$ velocity component directions in the same sense as CICE. To compare MPAS-Seaice to CICE, we use a simple test case, similar to that used in *Hunke* [2001]. This test case has a square planar domain of size 80km. Ice thickness is fixed at 2m, while ice concentration increases linearly in the eastwards direction from zero at the western boundary to one at the eastern boundary, and no snow is present.

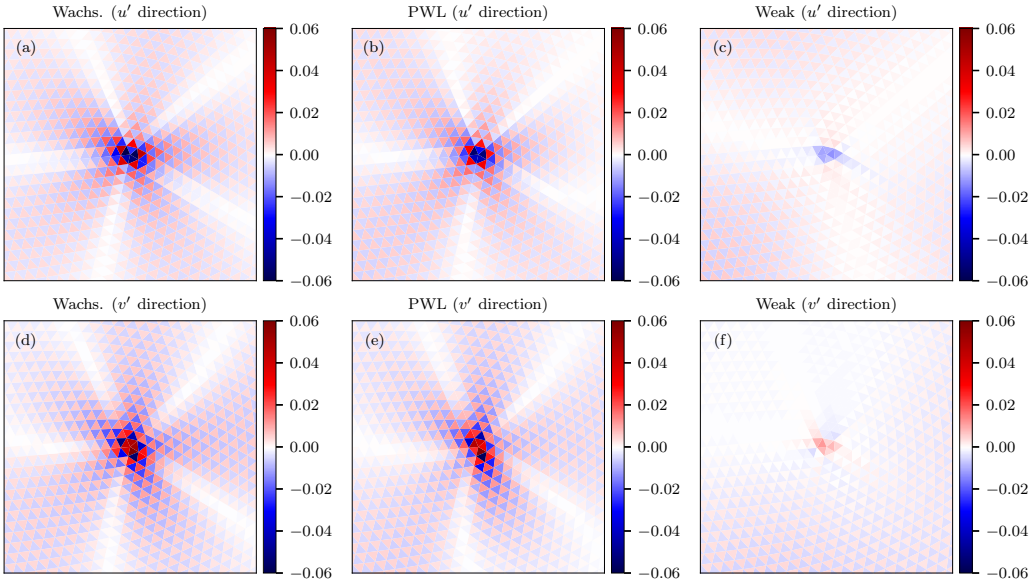


**Figure 5.** Detail of error in calculation of stress divergence from an analytical velocity field for a spherical test case. Detail is shown around one of the pentagonal mesh cells. The error is the difference between the model and analytical stress divergence divided by the maximum value of the analytical stress divergence on the sphere.

Only the velocity solver is active, with no advection or column physics. The sea ice is forced by atmospheric winds and ocean currents. The atmospheric wind forcing has the form

$$u_a = 5 - 3 \sin \frac{2\pi x}{L_x} \sin \frac{\pi y}{L_y} \quad \text{ms}^{-1} \tag{62}$$

$$v_a = 5 - 3 \sin \frac{2\pi y}{L_y} \sin \frac{\pi x}{L_x} \quad \text{ms}^{-1} \tag{63}$$

while the ocean currents have the form

$$u_o = 0.1 \frac{2y - L_y}{L_y} \quad \text{ms}^{-1} \tag{64}$$

$$v_o = -0.1 \frac{2x - L_x}{L_x} \quad \text{ms}^{-1} \tag{65}$$

where $x$ and $y$ are the horizontal position and $L_x$ and $L_y$ are the domain size in the $u$ and $v$ directions respectively. These forcing velocity fields are shown in Figure 7. Sea-ice velocity was simulated for four time steps (each of length 1 hour), which was sufficient time for the ice state to relax to the elliptical yield curve. Figure 8 shows a comparison of the modeled eastwards velocity and divergence of stress component between CICE and MPAS-Seaice. In this comparison MPAS-Seaice uses an identical quadrilateral mesh to CICE. The eastwards component of wind stress pushes the sea ice against

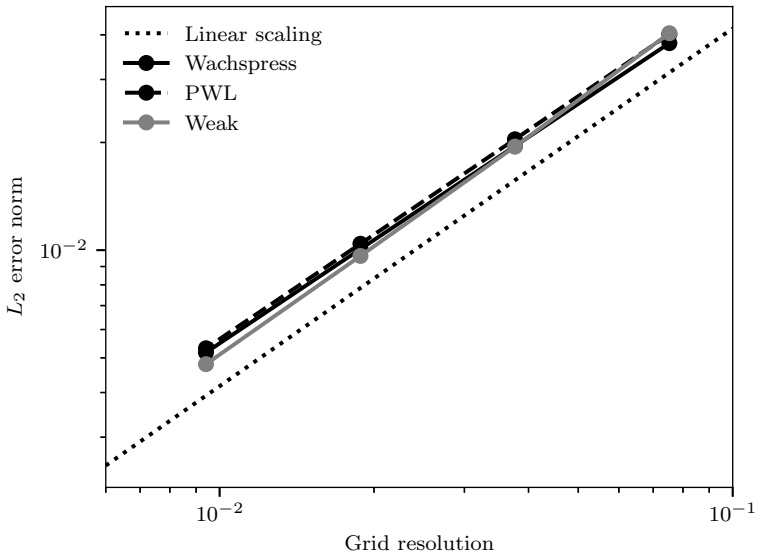


**Figure 6.** Scaling of $L_2$ error norm against grid resolution for calculation of the $u$ component of the divergence of stress on the unit sphere from an analytical velocity field for a spherical test case. Grid resolution is the mean distance between cell centers where $\phi > 20°$.

the east model boundary, and it is here that significant internal sea-ice stresses are present (see Figure 8e). The figure also shows that the three MPAS-Seaice schemes for calculating divergence of stress are capable of reproducing the results of CICE. As expected the variational scheme with Wachspress basis function best reproduces the results of CICE, since this algorithm is most similar to CICE. Differences with CICE appear as noise, a function of incompletely damped elastic waves from the EVP rheology [*Hunke*, 2001]. Figure 9 shows similar results for the same test case but with MPAS-Seaice using a regular hexagonal mesh. Here differences between the weak and variational scheme with PWL basis and the variational scheme with Wachspress basis are larger than for quadrilateral meshes, but still small. All the schemes have stress states that lie within or on the elliptical yield curve for both quadrilateral and hexagonal meshes (Figure 10).

### 3.2 Transport

To verify that the incremental remapping transport scheme works as expected, we ran two test cases on a global spherical grid, following LR05. In each case there is a steady eastward velocity field given by $\mathbf{u} = (u_0 \cos\theta, 0)$, where $u_0 = (2\pi R)/(12 \text{ days})$ and $R$ is the Earth's radius. We first advect a circular region of ice that has initial con-

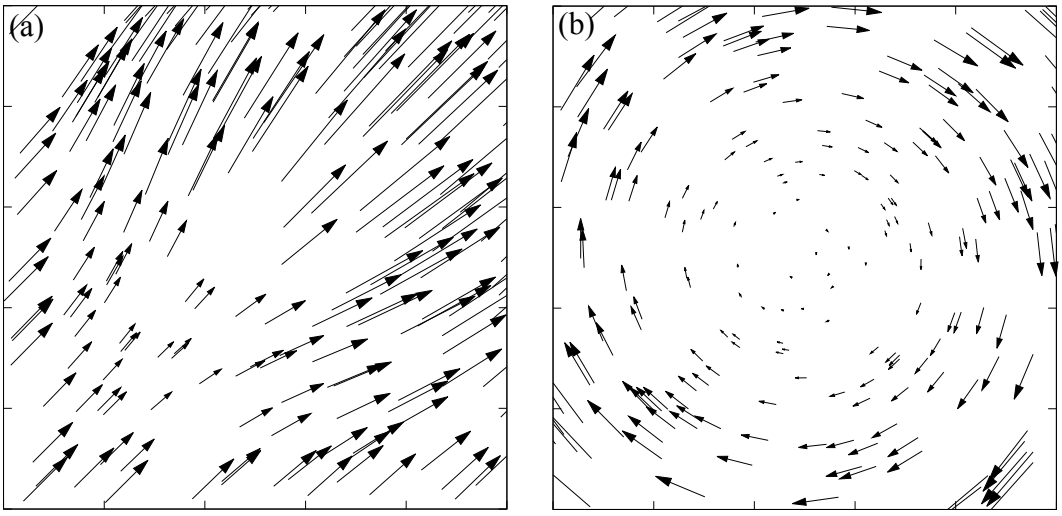


**Figure 7.** Forcing fields of (a) wind velocity and (b) ocean velocity used in a square domain test case.

centration given by a cosine-bell within a distance $R/3$ of a central point on the equator, and $a = 0$ elsewhere. The initial ice thickness is $h = 1$. For both the IR scheme and a first-order upwind scheme, the model was run at several grid resolutions for 12 days, at which time a perfect advection scheme would give a solution equal to the initial condition. For a grid resolution of 120 km, Figure 11a shows equatorial cross sections of $a$ for the initial condition, the upwind solution and the IR solution. Figure 12a-c shows the spatial distribution of ice concentration before and after the experiment for the same resolution. As expected, the upwind solution is very diffuse, while the IR scheme does a good job of preserving the initial shape.

Next, we advect a slotted cylinder with initial concentration $a = 1$, initial thickness $h = 1$ and radius $R/2$, also centered on the equator. We set $a = 0, h = 0$ for $r > R/2$ and also in a slot of width $R/6$ and length $5R/6$, with the long axis perpendicular to the flow. The model was run for 12 days at several resolutions. Figure 11b shows the initial condition and the upwind and IR solutions along the equator at resolution 120 km, while Figure 12d-f shows the spatial distribution of the ice concentration before and after the experiment for the same resolution. Again, the upwind scheme is very diffusive; all traces of the slot vanish. The IR scheme does well at maintaining the initial plateaus and a distinct slot, although diffusion into the slot raises the minimum concentration from 0 to ~0.2. Early in the IR simulation there is truncation at the leading and trailing edges of the cylinder, where the gradient is limited, but advection continues thereafter with lit-

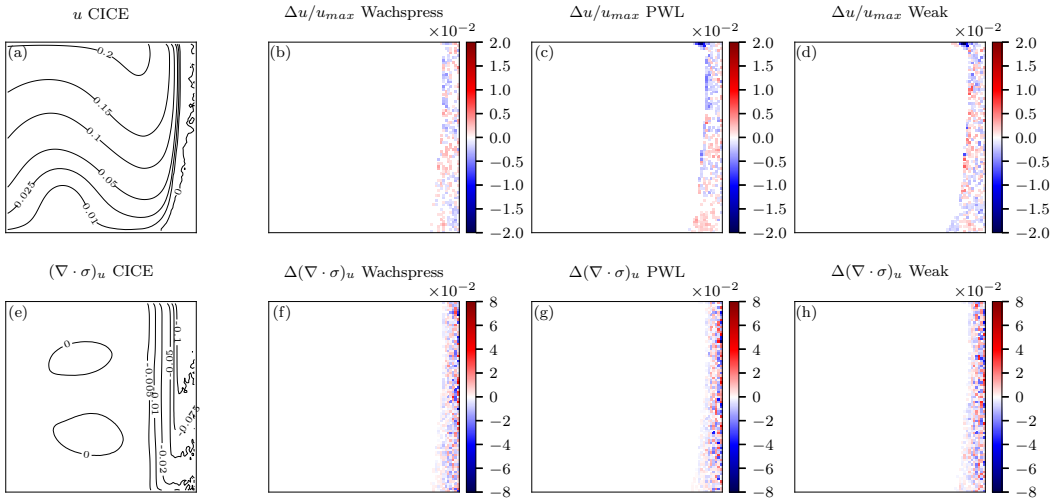


**Figure 8.** (a): $u$ velocity component of CICE for the square domain test case using a quadrilateral mesh. (b,c,d): Difference ($\Delta u/u_{max}$) between the $u$ component of MPAS-Seaice and CICE in the square domain test case for the MPAS-Seaice Wachspress variational, PWL variational and weak schemes respectively and using a quadrilateral mesh. (e-h): As (a-d) but for the $u$ component of the divergence of internal ice stress.

tle change in shape. The maximum concentration is just above 1 because the discretized velocity field is slightly convergent, and diffusion is small. On a plane (not shown) with steady $\mathbf{u} = (u_0, 0)$, the discretized velocity field is non-convergent, and $a$ remains bounded by [0,1]. Ice thickness, having been initialized to h = 1 everywhere, remains h = 1 everywhere (within roundoff), showing that both schemes preserve tracer monotonicity as expected.

Figure 13 shows the $L_2$ error norm of the 12-day solution for four grid resolutions ranging from 60 km to 480 km (where resolution is taken as the mean distance between neighboring cell centers). The $L_2$ norm is here given by

$$L_2 = \frac{\sqrt{\sum A_i (a_i - \hat{a}_i)^2}}{\sqrt{\sum A_i \hat{a}_i^2}}, \tag{66}$$

where $a_i$ is the 12-day solution for cell $i$, $\hat{a}_i$ is the initial ice area, and $A_i$ is the primary grid cell area. Each doubling of resolution should reduce the $L_2$ error norm by a factor of 2 for first-order-accurate schemes and a factor of 4 for second-order-accurate schemes. Figure 13 shows that the IR solution converges with close to second-order accuracy (indicated by the dotted diagonal line) for the cosine-bell and converges slightly below first-order accuracy for the slotted cylinder. This slow convergence for the slotted cylinder

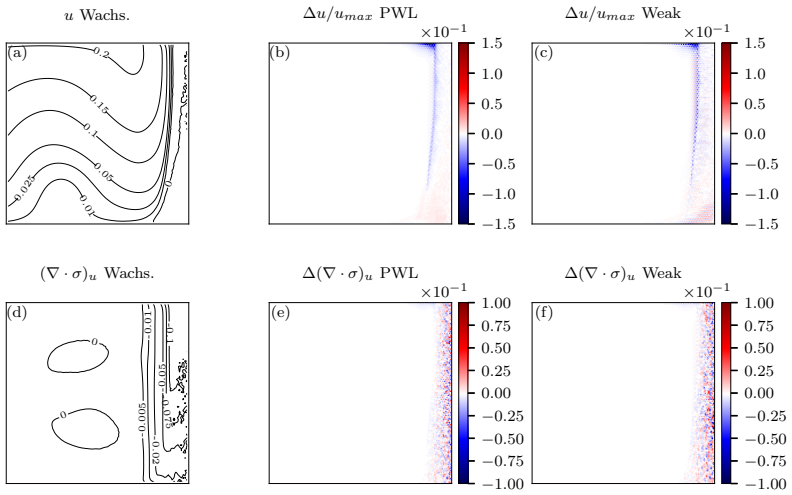


**Figure 9.** (a): $u$ velocity component of MPAS-Seaice with the Wachspress variational scheme the for the square domain test case using hexagonal elements. (b,c): Difference ($\Delta u/u_{max}$) between the $u$ component of MPAS-Seaice using the Wachspress variational scheme and MPAS-Seaice using the PWL variational and weak schemes respectively in the square domain test case using hexagonal cells. (d-f): As (a-c) but for the $u$ component of the divergence of internal ice stress.

is the result of the ice concentration discontinuity at the cylinder edge, which becomes sharper as the distance between neighboring grid cells decreases as resolution increases.The upwind scheme converges more slowly than the IR scheme, with larger errors at all resolutions.

### 3.3 Column physics

To validate the column physics in MPAS-Seaice we make use of the fact that CICE and MPAS-Seaice share identical code. CICE and MPAS-Seaice were run with identical forcing and with dynamics disabled. Results from the two models were bit-for-bit identical.

### 3.4 Global simulations

To validate the full MPAS-Seaice model in a global setting we perform stand-alone global simulations. We use a quasi-uniform mesh with average cell center to neighbor cell center distances of 60 km. The unstructured MPAS mesh allows flexibility in defining

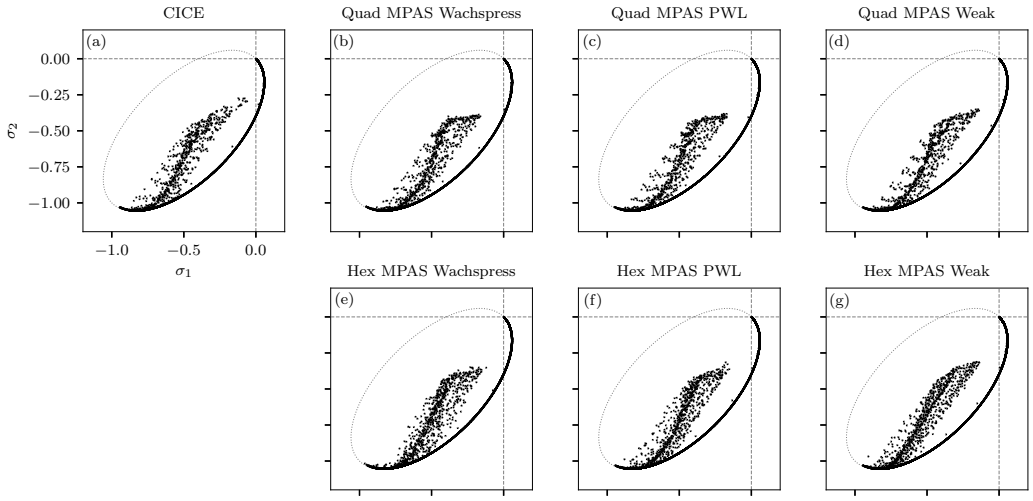


**Figure 10.** Principal stress components for the square domain test case and (dotted) the EVP yield curve. (a): CICE. (b-d): MPAS-Seaice on a quadrilateral mesh and using the Wachspress variational, PWL variational and weak schemes respectively. (e-g): MPAS-Seaice on a hexagonal mesh and using the Wachspress variational, PWL variational and weak schemes respectively.

model domains, so for computational efficiency we remove cells in the equatorial region away from high latitudes where sea ice is found. The mesh then consists of two separate polar caps. Culling equatorial cells in this way resulted in a halving of computation time with no loss of precision. We perform the simulation for 50 years from 1958 to 2007. Settings for the column physics are the standard ones for CICE [*Hunke et al.*, 2015].

For atmospheric and oceanic forcing we repeat the methods used by *Hunke et al.* [2013]. Air temperature, air specific humidity and air velocity at 10 m height and six-hourly frequency are taken from the Coordinated Ocean-ice Reference Experiments (CORE) Corrected Inter-Annual Forcing Version 2.0 [*Large and Yeager*, 2009; *Griffies et al.*, 2009]. Monthly climatologies of precipitation [*Griffies et al.*, 2009] and cloudiness [*Röske*, 2001] are also used. Downwelling shortwave radiation is calculated from the monthly climatology of cloudiness using the AOMIP shortwave forcing formula [*Hunke et al.*, 2015]. Downwelling longwave radiation is calculated according to *Rosati and Miyakoda* [1988]. Oceanic inputs, consisting of sea surface salinity, initial sea surface temperature, currents, sea-surface slope and deep ocean heat flux, come from monthly mean output of 20 years of a Community Climate System Model (CCSM) climate run (b30.009, Collins et al.,

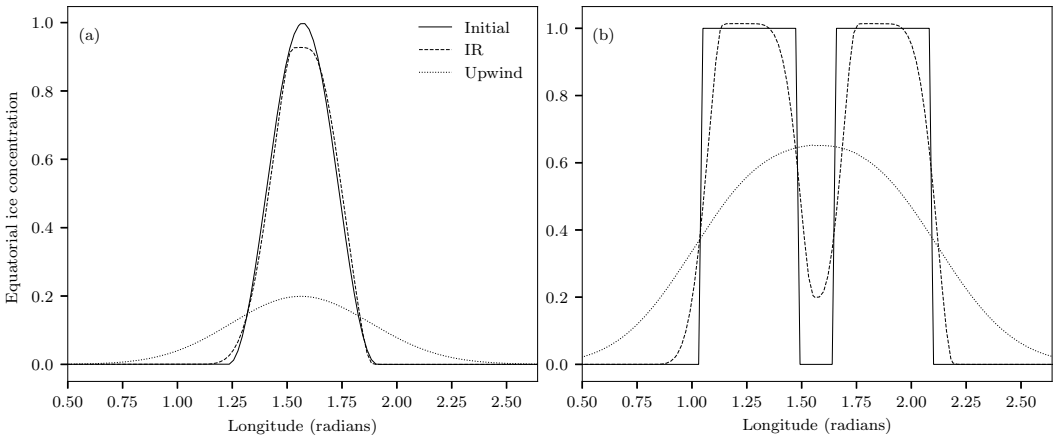


**Figure 11.** Comparison of incremental remapping to a first-order upwind scheme for advection around the sphere. (a) shows a circular region of sea ice whose center lies on the equator, with radius $R/3$ (where $R$ is the Earth's radius) and initial sea-ice concentration given by a cosine-bell. (b) shows a slotted cylinder whose center lies on the equator, with radius $R/2$. The grid resolution is 120 km. The exact solution (which corresponds to the initial condition) is shown by a solid line, IR by long dashes, and upwind by short dashes.

2006). All input forcing fields are interpolated linearly in time, although the MPAS forcing functionality can be easily extended to allow interpolation in time with arbitrary order.

We compare the MPAS-Seaice results to both observational datasets and to the results of simulations conducted with the CICE model [*Hunke et al.*, 2015, release version 5.1.2]. The CICE model uses a one-degree displaced-pole quadrilateral grid, and the default namelist options. The forcing used by CICE is the same as MPAS-Seaice but interpolated onto the one-degree mesh.

Total sea-ice extent is compared between the MPAS-Seaice and CICE models and observational values in Figure 14 for the years 1988 to 2007 inclusive. The observational sea-ice extent values for the Northern [*Cavalieri and Parkinson*, 2012; *Parkinson et al.*, 1999] and Southern Hemisphere [*Parkinson and Cavalieri*, 2012; *Zwally et al.*, 2002] show excellent agreement with both models, with the seasonal cycle of sea-ice extent well represented in both hemispheres. The largest discrepancy occurs in the Southern Hemisphere where Austral summertime sea ice extent is too low in both models.

A similar agreement between models and observations is found in Figure 15 where sea-ice concentration from SSMI observations using the NASATeam method [*Cavalieri et al.*, 1996, updated yearly] is compared spatially to model observations for summer and

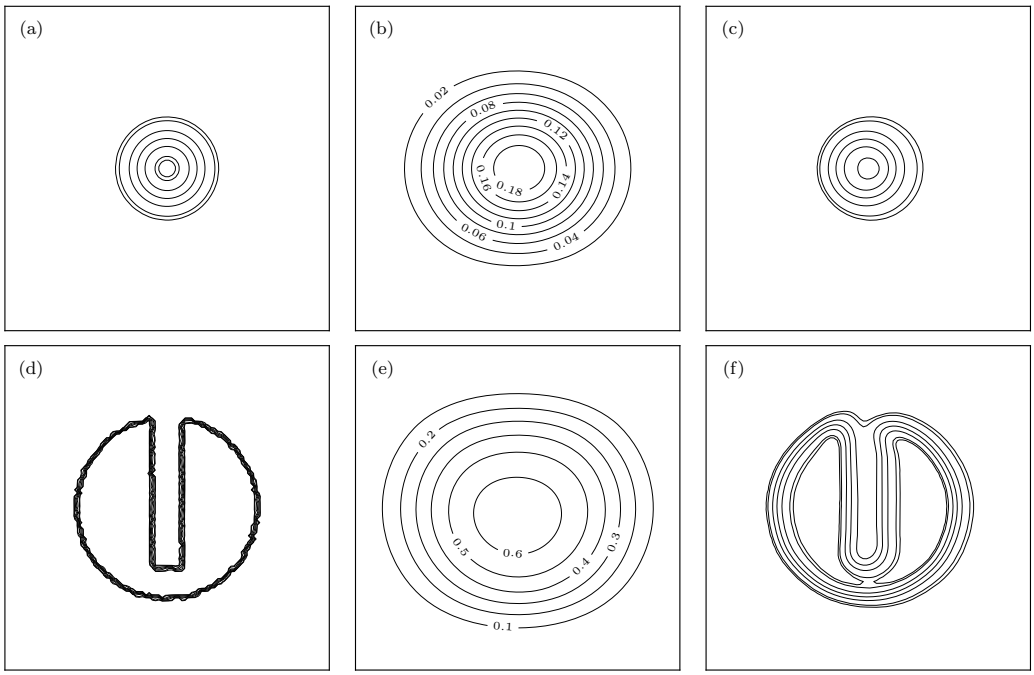


**Figure 12.** Initial (a,d) and final (b,c,e,f) ice concentration contours at120 km resolution for the cosine-bell (a-c) and slotted cylinder (d-f) advection test cases. Results for the upwind advection scheme are shown in (b) and (e), while results from the incremental remapping scheme are shown in (c) and (f). Contours are at levels 0.05, 0.1, 0.3, 0.5, 0.7, 0.9, and 0.95 unless otherwise shown.

winter periods in both hemispheres. Minor differences are present in both models at the Arctic ice edge during winter and in the pack interior in summer. In general the sea-ice extent is well reproduced. In the Southern Hemisphere the sea-ice extent is reasonably reproduced in summer by both models with more significant differences in the pack interior. As expected from Figure 14, more significant differences are found between the model results and observations in the Southern Hemisphere summer where ice concentration is particularly underrepresented in the models in the Weddell Sea. In general, agreement is much closer between the two models than between the models and observations. This is expected given the similarity of the models and model forcing.

Total sea-ice volume for the Northern Hemisphere is compared between the models and the Pan-Arctic Ice Ocean Modeling and Assimilation System (PIOMAS) assimilated data product [*Schweiger et al.*, 2011] in Figure 16. Both models and PIOMAS have the expected seasonal cycle with a similar variation between summer and winter and a decrease in total volume over time. Small differences exist between the two models and

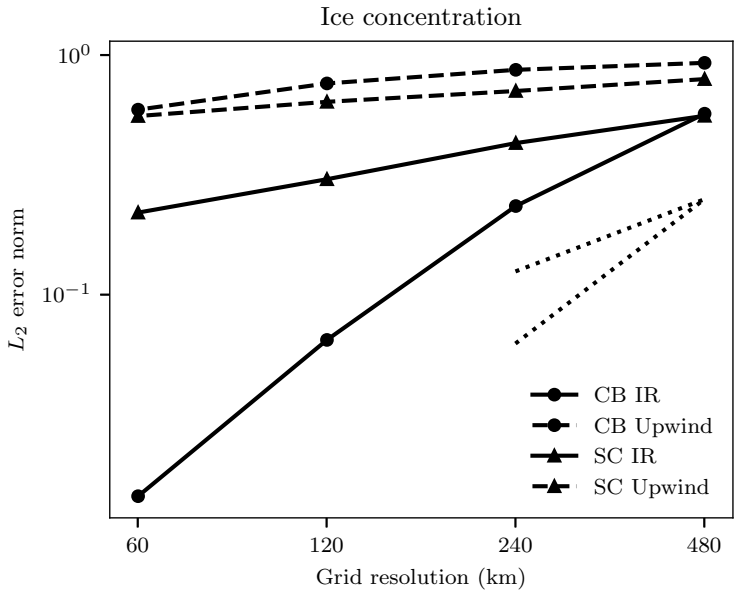


**Figure 13.** Scaling of the $L_2$ error norm with grid resolution for the cosine-bell (CB) and slotted cylinder (SC) advection tests shown in Figure 11. IR scaling is shown by the solid lines, upwind by long-dashed lines. Theoretical linear and quadratic scaling are shown by a short-dashed lines.

the PIOMAS product in terms of absolute ice volume level, with CICE experiencing the greatest sea-ice volume and PIOMAS the least. The cause of these differences is explored in Figure 17 where spatial patterns of sea-ice thickness are shown for the Northern Hemisphere in summer, autumn and winter. To better understand these differences in sea-ice volume we compare the models to observations of sea ice thickness from ICESat [see Figure 17, *Yi and Zwally*, 2009]. ICESat data is available from set periods from 2003 to 2008 during these seasons, and the model climatological maps are generated for the same periods. ICESat observations exclude sea ice with concentration less than 20%, so sea-ice thicknesses were excluded from the model results in the comparison where model ice concentration was less than 20%. Similar spatial patterns of sea ice are found in the results of both models and the ICESat observations, with thicker sea ice along the Canadian archipelago coast and thinner sea ice everywhere at the end of the summer melt season. Both MPAS-Seaice and CICE have excess sea ice thickness compared to ICESat observations in the Beaufort sea and western Arctic basin and a deficit of sea ice thickness in the Eurasian basin.

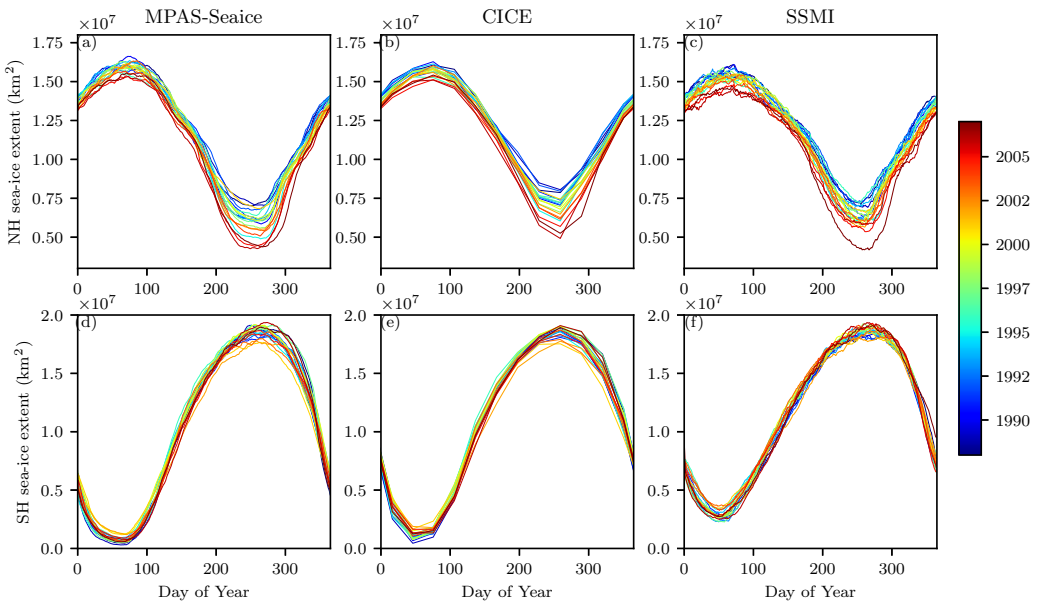


**Figure 14.** Sea ice extent (area with ice concentration greater than 15%) by year for the Northern (a-c) and Southern (d-f) Hemispheres, for MPAS-Seaice (a,d), CICE (b,e) and SSMI satellite observations (c,f).

## 4 Conclusions

The new sea-ice model, MPAS-Seaice, has been described here. Idealized test cases, both on planar and spherical meshes, have successfully validated the velocity solver and transport schemes. These schemes are heavily based on those implemented in the CICE sea-ice model, but adapted for the polygonal cells of MPAS meshes from the quadrilateral ones used in CICE. When using the variational scheme with Wachpress basis functions and a quadrilateral MPAS mesh, the velocity solver of MPAS-Seaice replicates the velocity solver algorithm of CICE, allowing rapid testing and validation. MPAS-Seaice and CICE share the sophisticated suite of column physics and biogeochemistry originally developed in CICE, again allowing its rapid development. Global simulations with realistic forcing have validated MPAS-Seaice against similar simulations with CICE and against observations for sea-ice concentration, extent and volume. MPAS-Seaice has been coupled into the Energy Exascale Earth System Model (E3SM), and the validation experiments described here give confidence in the sea-ice results from E3SM simulations. Future work will assess the fidelity and performance of MPAS-Seaice on variable resolution meshes.

**Acknowledgments**

This research was supported as part of the Energy Exascale Earth System Model (E3SM) project, funded by the U.S. Department of Energy, Office of Science, Office of Biological and Environmental Research. The authors would like to thank Michael Duda of the National Center for Atmospheric Research for his contributions to the MPAS software infrastructure.

## References

Arakawa, A., and V. R. Lamb (1977), Computational design of the basic dynamical processes of the UCLA General Circulation Model, in *General Circulation Models of the Atmosphere*, *Methods in Computational Physics: Advances in Research and Applications*, vol. 17, edited by J. Chang, pp. 173–265, Elsevier, doi:10.1016/B978-0-12-460817-7.50009-4.

Bailey, T. S., M. L. Adams, B. Yang, and M. R. Zika (2008), A piecewise linear finite element discretization of the diffusion equation for arbitrary polyhedral grids, *Journal of Computational Physics*, *227*(8), 3738–3757, doi: https://doi.org/10.1016/j.jcp.2007.11.026.

Batchelor, G. K. (1967), *An introduction to fluid dynamics*, Cambridge University Press.

Bitz, C. M., and W. H. Lipscomb (1999), An energy-conserving thermodynamic model of sea ice, *Journal of Geophysical Research: Oceans*, *104*(C7), 15,669–15,677, doi: 10.1029/1999JC900100.

Briegleb, B. P., and B. Light (2007), A Delta-Eddington multiple scattering parameterization for solar radiation in the sea ice component of the Community Climate System Model, *Tech. Rep. NCAR/TN-472+STR*, National Center for Atmospheric Research, Boulder, Colorado USA.

Cavalieri, D. J., and C. L. Parkinson (2012), Arctic sea ice variability and trends, 1979-2010, *The Cryosphere*, *6*(4), 881–889, doi:10.5194/tc-6-881-2012.

Cavalieri, D. J., C. L. Parkinson, P. Gloersen, and H. J. Zwally (1996, updated yearly), Sea ice concentrations from Nimbus-7 SMMR and DMSP SSM/I-SSMIS passive microwave data, version 1, *Tech. rep.*, NASA National Snow and Ice Data Center Distributed Active Archive Center, Boulder, Colorado USA, doi:10.5067/8GQ8LZQVL0VL.

Chen, C., G. Gao, J. Qi, A. Proshutinsky, R. C. Beardsley, Z. Kowalik, H. Lin, and G. Cowles (2009), A new high-resolution unstructured grid finite volume Arctic Ocean

model (AO-FVCOM): An application for tidal studies, *Journal of Geophysical Research: Oceans*, *114*(C8), doi:10.1029/2008JC004941, C08017.

Danilov, S., Q. Wang, R. Timmermann, N. Iakovlev, D. Sidorenko, M. Kimmritz, T. Jung, and J. Schröter (2015), Finite-Element Sea Ice Model (FESIM), version 2, *Geoscientific Model Development*, *8*(6), 1747–1761, doi:10.5194/gmd-8-1747-2015.

Dasgupta, G. (2003), Interpolants within convex polygons: Wachpress' shape functions, *J. Aerospac. Eng.*, *16*, 1–8, doi:10.1061/(ASCE)0893-1321(2003)16:1(1).

Duarte, P., A. Meyer, L. M. Olsen, H. M. Kauko, P. Assmy, A. Rösel, P. Itkin, S. R. Hudson, M. A. Granskog, S. Gerland, A. Sundfjord, H. Steen, H. Hop, L. Cohen, A. K. Peterson, N. Jeffery, S. M. Elliott, E. C. Hunke, and A. K. Turner (2017), Sea ice thermohaline dynamics and biogeochemistry in the Arctic Ocean: Empirical and model results, *Journal of Geophysical Research: Biogeosciences*, *122*(7), 1632–1654, doi:10.1002/2016JG003660.

Dukowicz, J. K., and J. R. Baumgardner (2000), Incremental remapping as a transport/advection algorithm, *Journal of Computational Physics*, *160*(1), 318–335, doi:10.1006/jcph.2000.6465.

Dunavant, D. A. (1985), High degree efficient symmetrical Gaussian quadrature rules for the triangle, *International Journal for Numerical Methods in Engineering*, *21*(6), 1129–1148, doi:10.1002/nme.1620210612.

Edwards, T. L., X. Fettweis, O. Gagliardini, F. Gillet-Chaulet, H. Goelzer, J. M. Gregory, M. Hoffman, P. Huybrechts, A. J. Payne, M. Perego, S. Price, A. Quiquet, and C. Ritz (2014), Effect of uncertainty in surface mass balance-elevation feedback on projections of the future sea level contribution of the Greenland ice sheet, *The Cryosphere*, *8*, 195–208, doi:10.5194/tc-8-195-2014.

Flocco, D., D. L. Feltham, and A. K. Turner (2010), Incorporation of a physically based melt pond scheme into the sea ice component of a climate model, *Journal of Geophysical Research: Oceans*, *115*(C8), doi:10.1029/2009JC005568, C08012.

Gao, G., C. Chen, J. Qi, and R. C. Beardsley (2011), An unstructured-grid, finite-volume sea ice model: Development, validation, and application, *Journal of Geophysical Research: Oceans*, *116*(C8), doi:10.1029/2010JC006688, C00D04.

Griffies, S. M., A. Biastoch, C. Böning, F. Bryan, G. Danabasoglu, E. P. Chassignet, M. H. England, R. Gerdes, H. Haak, R. W. Hallberg, W. Hazeleger, J. Jungclaus, W. G. Large, G. Madec, A. Pirani, B. L. Samuels, M. Scheinert, A. S. Gupta, C. A. Sever-

ijns, H. L. Simmons, A. M. Treguier, M. Winton, S. Yeager, and J. Yin (2009), Coordinated Ocean-ice Reference Experiments (COREs), *Ocean Modelling*, *26*(1), 1–46, doi:10.1016/j.ocemod.2008.08.007.

Hibler III, W. D. (1979), A dynamic thermodynamic sea ice model, *Journal of Physical Oceanography*, *9*(4), 815–846, doi:10.1175/1520-0485(1979)009<0815:ADTSIM>2.0.CO;2.

Holland, M. M., D. A. Bailey, B. P. Briegleb, B. Light, and E. Hunke (2012), Improved sea ice shortwave radiation physics in CCSM4: The impact of melt ponds and aerosols on arctic sea ice, *Journal of Climate*, *25*(5), 1413–1430, doi:10.1175/JCLI-D-11-00078.1.

Hunke, E. C. (2001), Viscous-plastic sea ice dynamics with the EVP model: Linearization issues, *Journal of Computational Physics*, *170*(1), 18–38, doi:10.1006/jcph.2001.6710.

Hunke, E. C., and J. K. Dukowicz (1997), An elastic-viscous-plastic model for sea ice dynamics, *Journal of Physical Oceanography*, *27*(9), 1849–1867, doi:10.1175/1520-0485(1997)027<1849:AEVPMF>2.0.CO;2.

Hunke, E. C., and J. K. Dukowicz (2002), 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, doi:10.1175/1520-0493(2002)130<1848:TEVPSI>2.0.CO;2.

Hunke, E. C., D. A. Hebert, and O. Lecomte (2013), Level-ice melt ponds in the Los Alamos sea ice model, CICE, *Ocean Modelling*, *71*, 26–42, doi:10.1016/j.ocemod.2012.11.008.

Hunke, E. C., W. H. Lipscomb, A. K. Turner, N. Jeffery, and S. Elliott (2015), CICE: the Los Alamos sea ice model documentation and software user's manual version 5.1, *Tech. rep.*, Los Alamos National Laboratory.

Hutchings, J. K., H. Jasak, and S. W. Laxon (2004), A strength implicit correction scheme for the viscous-plastic sea ice model, *Ocean Modelling*, *7*(1), 111–133, doi:10.1016/S1463-5003(03)00040-4.

Ingram, W. J., C. A. Wilson, and J. F. B. Mitchell (1989), Modeling climate change: An assessment of sea ice and surface albedo feedbacks, *Journal of Geophysical Research: Atmospheres*, *94*(D6), 8609–8622, doi:10.1029/JD094iD06p08609.

Jeffery, N., and E. C. Hunke (2014), Modeling the winter-spring transition of first-year ice in the western Weddell Sea, *Journal of Geophysical Research: Oceans*, *119*(9), 5891–

5920, doi:10.1002/2013JC009634.

Jeffery, N., S. Elliott, E. C. Hunke, W. H. Lipscomb, and A. Turner (2016), Biogeochemistry of CICE: The Los Alamos sea ice model documentation and software user's manual, zbgc_colpkg modifications to version 5.0., *Tech. Rep. LA-UR-16-27780*, Los Alamos National Laboratory, Los Alamos, NM, USA.

Killworth, P. D. (1983), Deep convection in the World Ocean, *Reviews of Geophysics*, *21*(1), 1–26, doi:10.1029/RG021i001p00001.

Large, W. G., and S. G. Yeager (2009), The global climatology of an interannually varying air–sea flux data set, *Climate Dynamics*, *33*(2), 341–364, doi:10.1007/s00382-008-0441-3.

Lietaer, O., T. Fichefet, and V. Legat (2008), The effects of resolving the Canadian Arctic Archipelago in a finite element sea ice model, *Ocean Modelling*, *24*(3), 140–152, doi:10.1016/j.ocemod.2008.06.002.

Lipscomb, W. H. (2001), Remapping the thickness distribution in sea ice models, *Journal of Geophysical Research: Oceans*, *106*(C7), 13,989–14,000, doi:10.1029/2000JC000518.

Lipscomb, W. H., and E. C. Hunke (2004), Modeling sea ice transport using incremental remapping, *Monthly Weather Review*, *132*(6), 1341–1354, doi:10.1175/1520-0493(2004)132<1341:MSITUI>2.0.CO;2.

Lipscomb, W. H., and T. D. Ringler (2005), An incremental remapping transport scheme on a spherical geodesic grid, *Monthly Weather Review*, *133*(8), 2335–2350, doi:10.1175/MWR2983.1.

Lipscomb, W. H., E. C. Hunke, W. Maslowski, and J. Jakacki (2007), Ridging, strength, and stability in high-resolution sea ice models, *Journal of Geophysical Research: Oceans*, *112*(C3), doi:10.1029/2005JC003355, C03S91.

Murray, R. J. (1996), Explicit generation of orthogonal grids for ocean models, *Journal of Computational Physics*, *126*(2), 251–273, doi:10.1006/jcph.1996.0136.

Parkinson, C. L., and D. J. Cavalieri (2012), Antarctic sea ice variability and trends, 1979-2010, *The Cryosphere*, *6*(4), 871–880, doi:10.5194/tc-6-871-2012.

Parkinson, C. L., D. J. Cavalieri, P. Gloersen, H. J. Zwally, and J. C. Comiso (1999), Arctic sea ice extents, areas, and trends, 1978–1996, *Journal of Geophysical Research: Oceans*, *104*(C9), 20,837–20,856, doi:10.1029/1999JC900082.

Ringler, T., J. Thuburn, J. Klemp, and W. Skamarock (2010), A unified approach to energy conservation and potential vorticity dynamics for arbitrarily-structured C-grids,

*Journal of Computational Physics*, *229*(9), 3065–3090, doi:10.1016/j.jcp.2009.12.007.

Ringler, T., M. Petersen, R. L. Higdon, D. Jacobsen, P. W. Jones, and M. Maltrud (2013), A multi-resolution approach to global ocean modeling, *Ocean Modelling*, *69*, 211–232, doi:10.1016/j.ocemod.2013.04.010.

Rosati, A., and K. Miyakoda (1988), A general circulation model for upper ocean simulation, *Journal of Physical Oceanography*, *18*(11), 1601–1626, doi:10.1175/1520-0485(1988)018<1601:AGCMFU>2.0.CO;2.

Röske, F. (2001), An atlas of surface fluxes based on the ECMWF reanalysis. A climatological data set to force global ocean general circulation models, *Tech. Rep. 323*, Max-Planck-Inst. Für Meteorol., Hamburg, Germany.

Schweiger, A., R. Lindsay, J. Zhang, M. Steele, H. Stern, and R. Kwok (2011), Uncertainty in modeled Arctic sea ice volume, *Journal of Geophysical Research: Oceans*, *116*(C8), doi:10.1029/2011JC007084, C00D06.

Skamarock, W. C., J. B. Klemp, M. G. Duda, L. D. Fowler, S.-H. Park, and T. D. Ringler (2012), A multiscale nonhydrostatic atmospheric model using centroidal Voronoi tesselations and C-grid staggering, *Monthly Weather Review*, *140*(9), 3090–3105, doi:10.1175/MWR-D-11-00215.1.

Smith, R., S. Kortas, and B. Meltz (1995), Curvilinear coordinates for global ocean models, *Tech. Rep. LA-UR-95-1146*, Los Alamos National Laboratory.

Turner, A. K., and E. C. Hunke (2015), Impacts of a mushy-layer thermodynamic approach in global sea-ice simulations using the CICE sea-ice model, *Journal of Geophysical Research: Oceans*, *120*(2), 1253–1275, doi:10.1002/2014JC010358.

Turner, A. K., E. C. Hunke, and C. M. Bitz (2013), Two modes of sea-ice gravity drainage: A parameterization for large-scale modeling, *Journal of Geophysical Research: Oceans*, *118*(5), 2279–2294, doi:10.1002/jgrc.20171.

van Leer, B. (1979), Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov's method, *Journal of Computational Physics*, *32*(1), 101–136, doi:10.1016/0021-9991(79)90145-1.

Wang, Q., S. Danilov, D. Sidorenko, R. Timmermann, C. Wekerle, X. Wang, T. Jung, and J. Schröter (2014), 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, doi:10.5194/gmd-7-663-2014.

Yi, D., and H. J. Zwally (2009), Arctic sea ice freeboard and thickness, version 1., *Tech. rep.*, NSIDC: National Snow and Ice Data Center, Boulder, Colorado USA., doi: 10.5067/SXJVJ3A2XIZT, updated 2014-04-15.

Zwally, H. J., J. C. Comiso, C. L. Parkinson, D. J. Cavalieri, and P. Gloersen (2002), Variability of Antarctic sea ice 1979–1998, *Journal of Geophysical Research: Oceans*, *107*(C5), 9–1–9–19, doi:10.1029/2000JC000733.

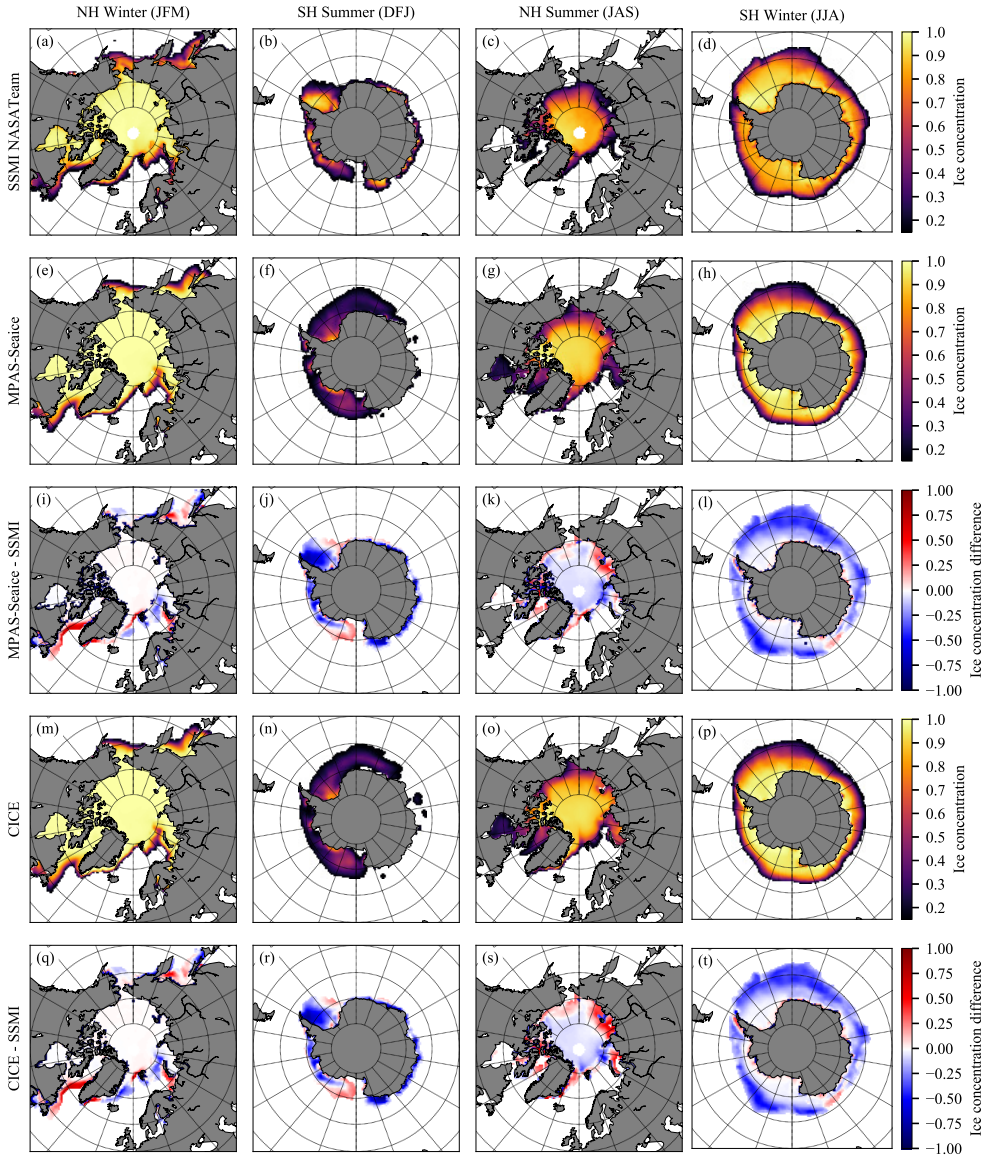


**Figure 15.** Spatial climatological maps for 1988 to 2007 of sea ice concentration from (a-d) SSMI satellite observations processed with the NASATeam algorithm, (e-h): MPAS-seaice, and (m-p): CICE. (i-l): Differences between MPAS-Seaice ice concentration and SSMI observations. (q-t): Differences between CICE ice concentration and SSMI observations. (a,e,i,m,q): Northern Hemisphere Winter: January, February, and March. (b,f,j,n,r): Southern Hemisphere summer: December, January, and February. (c,g,k,o,s): Northern Hemisphere summer: July, August, and September. (d,h,l,p,t): Southern Hemisphere winter: June, July, and August.

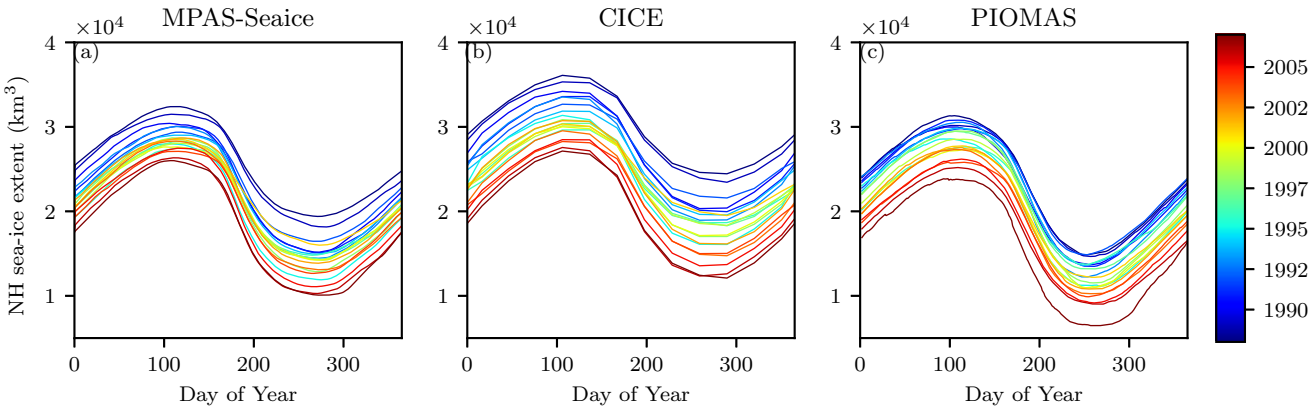


**Figure 16.** Northern Hemisphere total sea-ice volume by year. (a) MPAS-Seaice. (b) CICE. (c) PIOMAS.

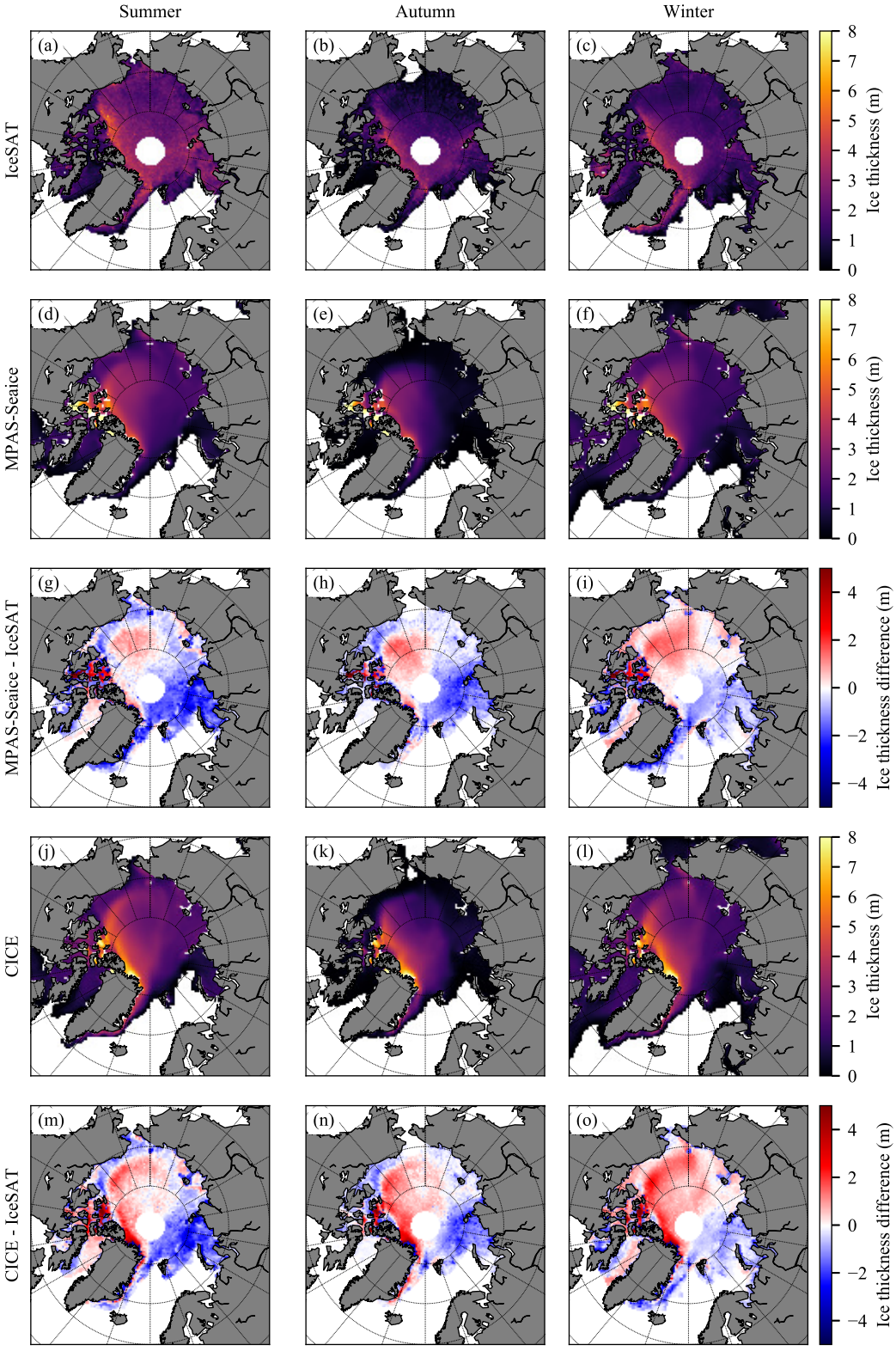


**Figure 17.** Spatial climatological maps of sea-ice thickness from ICESat (a-c), MPAS-Seaice (d-f), and CICE (j-l). (g-i): Differences between MPAS-Seaice ice thickness and ICESat observations. (m-o): Differences between CICE ice thickness and ICESat observations. (a,d,g,j,m): NH summer. (b,e,h,k,n): NH Autumn. (c,f,i,l,o): NH Winter.