JOURNAL OF GEOPHYSICAL RESEARCH, VOL. 112, C02S90, doi:10.1029/2005JC003329, 2007

Using the material-point method to model sea ice dynamics

Deborah Sulsky,[1] Howard Schreyer,[2] Kara Peterson,[1] Ron Kwok,[3] and Max Coon[4]

Received 25 September 2005; revised 17 August 2006; accepted 2 October 2006; published 22 February 2007.

[1] The material-point method (MPM) is a numerical method for continuum mechanics that combines the best aspects of Lagrangian and Eulerian discretizations. The material points provide a Lagrangian description of the ice that models convection naturally. Thus properties such as ice thickness and compactness are computed in a Lagrangian frame and do not suffer from errors associated with Eulerian advection schemes, such as artificial diffusion, dispersion, or oscillations near discontinuities. This desirable property is illustrated by solving transport of ice in uniform, rotational and convergent velocity fields. Moreover, the ice geometry is represented by unconnected material points rather than a grid. This representation facilitates modeling the large deformations observed in the Arctic, as well as localized deformation along leads, and admits a sharp representation of the ice edge. MPM also easily allows the use of any ice constitutive model. The versatility of MPM is demonstrated by using two constitutive models for simulations of wind-driven ice. The first model is a standard viscous-plastic model with two thickness categories. The MPM solution to the viscous-plastic model agrees with previously published results using finite elements. The second model is a new elastic-decohesive model that explicitly represents leads. The model includes a mechanism to initiate leads, and to predict their orientation and width. The elastic-decohesion model can provide similar overall deformation as the viscous-plastic model; however, explicit regions of opening and shear are predicted. Furthermore, the efficiency of MPM with the elastic-decohesive model is competitive with the current best methods for sea ice dynamics.

Citation: Sulsky, D., H. Schreyer, K. Peterson, R. Kwok, and M. Coon (2007), Using the material-point method to model sea ice dynamics, *J. Geophys. Res.*, *112*, C02S90, doi:10.1029/2005JC003329.

1. Introduction

[2] Ice dynamics models are used to generate short-term forecasts for waterways [*Sayed and Carrieres*, 1999], to predict the ice-edge position for offshore operations [*Flato*, 1993] and to understand the global climate in long-term studies [*Maslowski et al.*, 2001]. A continuum model of pack ice dynamics involves a mathematical description of the physical laws, such as momentum balance. The balance laws must be augmented by constitutive relations in order to distinguish a particular type of continuum, in this case, ice. The appropriate balance laws for our model are presented in section 2. Constitutive models for the pack ice are presented in section 4.

[3] Once the mathematical equations have been specified, one must decide on a numerical procedure to approximately solve these equations, since ordinarily analytical solutions are not available. Two mathematically equivalent forms of the balance laws are possible. One form is based on an Eulerian description, observing the solution at fixed points in space; or a Lagrangian description, following points in the flow or deformation of the material. These two descriptions suggest different numerical approaches. Solids are more typically modeled with Lagrangian finite elements [*Belytschko et al.*, 2000; *Hughes*, 2000] while fluids are more often modeled in an Eulerian frame [*LeVeque*, 1990; *Roache*, 1985]. Each approach has advantages and disadvantages. Eulerian methods are traditionally used to model ice because they can handle large deformations [*Hibler*, 1979; *Hunke and Dukowicz*, 1997]; however advection algorithms within the Eulerian framework have artificial numerical diffusion and tend to smear regions of thick and thin ice that should have sharp boundaries and to smear the ice edge [*Flato*, 1993]. Lagrangian methods do not have the advection errors but the mesh moves in the computed velocity field, so there is a limit to how much deformation can be accommodated without remeshing [*Morland and Staroszczyk*, 1998; *Wang and Ikeda*, 2004]. It is remeshing that causes diffusion in a Lagrangian scheme. However, Lagrangian methods are better suited to constitutive models that depend on the deformation history of a material point,

[1] Department of Mathematics and Statistics, University of New Mexico, Albuquerque, New Mexico, USA.

[2] Department of Mechanical Engineering, University of New Mexico, Albuquerque, New Mexico, USA.

[3] Jet Propulsion Laboratory, Pasadena, California, USA.

[4] Northwest Research Associates, Seattle, Washington, USA.

0148-0227/07/2005JC003329$09.00

such as elastic-plastic models, since the integration is done along material-point trajectories.

[4] The smoothed particle hydrodynamics (SPH) method has also been applied to sea ice. SPH is a fully Lagrangian method that has been used with viscous-plastic rheologies [*Gutfraind and Savage*, 1997; *Lindsay and Stern*, 2004]. *Lindsay and Stern* [2004] note that the Lagrangian description in SPH is also desirable in order to assimilate material-point trajectories from RADARSAT Geophysical Processor System (RGPS) data. Another option for examining sea-ice dynamics is discrete-particle methods [*Gutfraind and Savage*, 1997; *Hopkins et al.*, 1991; *Hopkins and Hibler*, 1991; *Hopkins*, 1998]. In these methods the ice is modeled as a collection of floes and equations are solved for the interactions between the floes. These methods are formulated directly as discrete models rather than by discretizing a continuum model.

[5] The material-point method (MPM) is a numerical technique for solving continuum problems in fluid and solid mechanics. Its origins are the particle-in-cell (PIC) method developed at Los Alamos in the 1950s [*Harlow*, 1957; *Evans and Harlow*, 1957] to model highly distorted fluid flow - such as the splash of a falling drop. The original PIC method was successful for its time, but was eventually replaced by Eulerian methods with sophisticated advection schemes that have less numerical dissipation. In the late 1980s, *Brackbill and Ruppel* [1986] and *Brackbill et al.* [1988] revived the PIC technology with simple modifications that reduced the numerical dissipation and made PIC competitive with current technologies for simulating hydrodynamics. MPM [*Sulsky et al.*, 1994, 1995; *Sulsky and Schreyer*, 1996] is an extension of this technology to solids with strength and stiffness. In some formulations, numerical dissipation is eliminated entirely [*Burgess et al.*, 1992; *Love and Sulsky*, 2006]. MPM has been used to model diverse applications such as impact, penetration, fracture, metal forming, granular media and membranes.

[6] In order to combine the advantages of Eulerian and Lagrangian methods, MPM uses two representations of the continuum. First, a set of material points (or particles) is identified in the body of fluid or solid that is tracked throughout the deformation process. Each material point has a mass, position, velocity and stress, as well as material parameters and internal variables as needed for constitutive models or thermodynamics. These material points provide a Lagrangian description of the material that is not subject to mesh tangling because no connectivity is assumed between the points. This Lagrangian frame models convection and transport in a natural manner since the trajectory and history of each material point is followed. Each point carries material properties without error and history variables can be integrated along the trajectory. However, computing gradients for solution of the momentum equation is complicated in this representation since the neighbors of a given point are not known a priori, and can change during a simulation. In order to keep the computational work linear in the number of material points, a second description is used for solving the momentum equation. This description is an often regular, background mesh that covers the computational domain. Information is transferred from the material points to the background mesh, the momentum equation is solved on the background mesh, and then information from the mesh solution is used to update the material points, at which time the background mesh can be modified if desired and then the cycle is begun again. The MPM algorithm is presented in section 3 with a particular emphasis on its application to simulating the dynamics of pack ice.

[7] Other forms of PIC methods [*Flato*, 1993; *Zhang and Savage*, 1998; *Sayed and Carrieres*, 1999] have been used previously to model sea ice. *Flato* [1993] makes the case that PIC methods are better suited to predicting the location of the ice edge because the artificial diffusion associated with pure Eulerian methods is removed. MPM shares this advantage, as does any Lagrangian formulation. The Lagrangian description in MPM may also aid in data assimilation [*Lindsay and Stern*, 2004]. MPM has another advantage in its method for evaluating constitutive equations that easily allows the use of any solid (or fluid) model for ice in addition to the traditional viscous-plastic models that are normally used in numerical simulations of ice dynamics. Even models with history dependence can be employed without incurring convection errors of an Eulerian method when convecting internal variables used in the model. Section 5 of this paper contrasts previous PIC implementations to MPM. Section 6 presents numerical simulations that demonstrate the effectiveness of the MPM advection algorithm and uses MPM to simulate wind-driven ice. Finally, section 7 contains concluding remarks.

2. Equations of Motion

[8] Since the pack ice is thin compared to its horizontal extent, it can be modeled in two spatial dimensions with a position vector indicated by $\mathbf{x}$, velocity by $\mathbf{v}$, mass density (per unit volume) by ρ, and thickness by h. The mass density per unit area is $m = \rho h$.

[9] In the plane of motion of the sea ice, the linear momentum balance is given by [*Coon*, 1980]

$$m\dot{\mathbf{v}} = \mathbf{F}^{\text{ext}} + \mathbf{F}^{\text{int}}. \tag{1}$$

The superposed dot denotes the material time derivative of the velocity (acceleration). The internal forces arise from the divergence of the depth-integrated, extra-stress, $\mathbf{N}$ [*Gray and Morland*, 1994]

$$\mathbf{F}^{\text{int}} = \text{div}(\mathbf{N}). \tag{2}$$

The external forces are due to air drag, water drag, Coriolis force, and sea surface tilt. Specific forms used in the simulations are given in section 6.

[10] In order to solve the preceding equations in a region Ω, we need to specify the initial ice thickness, $h(x, 0) = h_0(x)$, the initial displacement $u(x, 0) = u_0(x)$, and the initial velocity, $v(x, 0) = v_0(x)$ in this region. We also need conditions specified on the region boundary $\partial\Omega$. Typical boundary conditions might be prescribed displacement on part of the boundary $\partial\Omega_u$ and prescribed traction on the remaining part of the boundary $\partial\Omega_t$, with $\partial\Omega = \partial\Omega_u \cup \partial\Omega_t$ and $\partial\Omega_u \cap \partial\Omega_t = \emptyset$. The depth-integrated traction is given by $\mathbf{N} \cdot \mathbf{n}$ where $\mathbf{n}$ is a unit, outward normal to the boundary. Specifying the traction would imply specifying an applied

force per unit length on the boundary of the ice. Specified inflow or outflow are also possibilities. To complete the model, we must also specify a constitutive equation for the stress **N**. We postpone the detailed description of constitutive models until section 4.

3. Material-Point Method for Pack Ice

[11] MPM partitions the ice into material elements that are followed in a Lagrangian sense throughout the computation. Specifically, for the two-dimensional ice cover, assume the ice occupies a region $\Omega_0 = \Omega(0) \subset \mathbb{R}^2$ initially and $\Omega(t) \subset \mathbb{R}^2$ for $t > 0$. In a continuum, material points in the original configuration are labeled with coordinates $\mathbf{X}$. At time $t > 0$ the current position of the point that started at $\mathbf{X}$ is $\mathbf{x} = \boldsymbol{\varphi}(\mathbf{X}, t)$ with $\boldsymbol{\varphi}(\mathbf{X}, 0) = \mathbf{X}$. In the discretization, a finite set of material points is tracked. Divide the initial region Ω_0 into N_p disjoint subdomains, $\Omega_0 = \bigcup_{p=1}^{N_p} \Omega_p(0)$. Associate a position $\mathbf{X}_p$ with the centroid of $\Omega_p(0)$. The position of the material point initially located at the centroid of each subdomain is tracked in time. A mass, thickness and area associated with this point are also tracked. Let $h_p(0)$ be the initial average thickness of the ice in $\Omega_p(0)$. The initial mass of the material point is then

$$m_p(0) = \int_{\Omega_p(0)} \rho_0(\mathbf{X}) h_p(0) dA \sim \rho_0(\mathbf{X}_p) h_p(0) \Omega_p(0), \qquad (3)$$

where $\rho_0(\mathbf{X})$ is the original mass density and we use $\Omega_p(0)$ also to indicate the area of the region. Without melting or freezing, the mass m_p would be constant in time, and the total mass (sum of all material-point mass) would be trivially conserved. The area and thickness may change with time, even if the mass is fixed. Material points can be eliminated from the computation if their mass goes to zero due to melting and points can be added or the mass of existing points increased in regions where ice is forming. In this paper, the material-point mass is kept constant.

[12] The deformation gradient is a key quantity in continuum mechanics and is defined as the derivative of the current position with respect to the original position,

$$\mathbf{F}(\mathbf{X}, t) = \partial \boldsymbol{\varphi}(\mathbf{X}, t) / \partial \mathbf{X} = \text{Grad } \boldsymbol{\varphi}. \qquad (4)$$

The deformation gradient is a linear transformation with positive determinant and has a 2 × 2 matrix representation for the two-dimensional ice cover. The Jacobian of the transformation from original to current coordinates is $J(\mathbf{X}, t) = \det(\mathbf{F}(\mathbf{X}, t)) > 0$. The Jacobian transforms area elements in the original configuration, dA, to area elements in the current configuration, $da = JdA$. Thus, if the deformation gradient is known at a material-point location, the initial area associated with the material point can be transformed to the current area through the transformation

$$\Omega_p(t) = J(\mathbf{X}_\mathbf{p}, t) \Omega_p(0). \qquad (5)$$

[13] The average thickness associated with a material point might change due to mechanical forces, for example, through ridging, or due to melting or freezing of the ice. The thermodynamic effects are taken into account by a model of the ice growth rate. The growth rate can be parameterized by the ice thickness [*Hibler*, 1979; *Thorndike and Maykut*, 1973], or by freezing degree days [*Anderson*, 1961; *Lebedev*, 1940], for example. More sophisticated thermodynamic modeling is also possible [*Bitz and Lipscomb*, 1999]. In any case, an ordinary differential equation can be formulated for obtaining the current value of the ice thickness for a material point.

[14] The material points move according to the velocity computed as the solution to the momentum equation. In contrast to discrete particle methods [*Gutfraind and Savage*, 1997; *Hopkins et al.*, 1991; *Hopkins and Hibler*, 1991; *Hopkins*, 1998], or SPH [*Gutfraind and Savage*, 1997; *Lindsay and Stern*, 2004], the material points do not interact directly with one another. Instead information from the current state of the material points is used to initialize points on a background computational grid, and then the momentum equation is solved on that grid. The background grid points are allowed to move with the flow, and therefore the grid solution is Lagrangian over the time step. The use of a background grid makes the computational work scale linearly with the number of material points.

3.1. Spatial Discretization of the Momentum Equation

[15] One approach to obtaining discrete equations for the momentum balance on the background grid is the finite element method with an updated Lagrangian formulation. The background grid is subdivided into elements, Ω^e, $e = 1, 2, \ldots, N_e$. The nodes of this mesh are $\mathbf{x}_I$, $I = 1, \ldots, N_n$. For definiteness, consider a mesh made up of quadrilateral elements in two-dimensions, each with four nodes. In the finite element method, approximations to functions are constructed by interpolating from the nodal values using shape functions (see Appendix A). In terms of global shape functions, the motion of the ice, $\mathbf{x} = \boldsymbol{\varphi}(\mathbf{X}, t)$, can be approximated by

$$\mathbf{x}^h = \sum_{I=1}^{N_n} \mathbf{x}_I N_I(\mathbf{x}). \qquad (6)$$

The superscript h is used throughout to denote finite element approximations to the corresponding continuum quantity.

[16] The discrete displacement field is defined from the nodal positions

$$\begin{aligned} \mathbf{u}^h(\mathbf{x}, t) = \mathbf{x}^h - \mathbf{X} &= \sum_{I=1}^{N_n} (\mathbf{x}_I(t) - \mathbf{X}_I) N_I(\mathbf{x}) \\ &= \sum_{I=1}^{N_n} \mathbf{u}_I(t) N_I(\mathbf{x}). \end{aligned} \qquad (7)$$

The shape function is defined by a mapping from a master element so that its material time derivative is zero (Appendix A). Accordingly, the velocity can be approximated as

$$\begin{aligned} \mathbf{v}^h(\mathbf{x}, t) = \dot{\mathbf{u}}^h(\mathbf{x}, t) &= \sum_{I=1}^{N_n} \dot{\mathbf{u}}_I(t) N_I(\mathbf{x}) \\ &= \sum_{I=1}^{N_n} \mathbf{v}_I(t) N_I(\mathbf{x}). \end{aligned} \qquad (8)$$

Likewise, the acceleration is approximated by

$$\mathbf{a}^h(\mathbf{x},t) = \dot{\mathbf{v}}^h(\mathbf{x},t) = \sum_{I=1}^{N_n} \dot{\mathbf{v}}_I(t) N_I(\mathbf{x}) = \sum_{I=1}^{N_n} \mathbf{a}_I(t) N_I(\mathbf{x}). \tag{9}$$

[17] These approximations are used in the weak form of the momentum balance to obtain semidiscrete equations of motion. The weak form is equivalent to the principle of virtual work which states that the solution to the momentum balance equation must satisfy the following integral equation for any admissible, smooth, virtual displacement **w**

$$\int_{\Omega(t)} m\mathbf{w} \cdot \dot{\mathbf{v}} da = -\int_{\Omega(t)} \text{grad } \mathbf{w} : \mathbf{N} da + \int_{\partial\Omega_t(t)} \mathbf{w} \cdot \bar{\mathbf{t}} ds + \int_{\Omega(t)} \mathbf{w} \cdot \mathbf{F}^{\text{ext}} da, \tag{10}$$

where $\bar{\mathbf{t}} = \mathbf{N} \cdot \mathbf{n}$ is the prescribed traction on the part of the boundary $\partial\Omega_t(t)$. An advantage of the weak form is that less smoothness is required of solutions than with the strong form, equation (1). The virtual displacement is also called a test function, and it is admissible if it is zero on $\partial\Omega_u$ where displacement boundary conditions are prescribed. The test function has a representation similar to the other fields above

$$\mathbf{w}^h(\mathbf{x}) = \sum_{I=1}^{N_n} \mathbf{w}_I N_I(\mathbf{x}). \tag{11}$$

The test function is arbitrary except at points on the boundary $\partial\Omega_u$ where the displacement is prescribed, in which case the test function is zero.

[18] Substitute the representations (8) and (11) into (10) to obtain

$$\int_{\Omega(t)} m\mathbf{w}^h \cdot \dot{\mathbf{v}}^h da = \sum_{I=1}^{N_n} \mathbf{w}_I \cdot \sum_{J=1}^{N_n} \int_{\Omega(t)} m(\mathbf{x},t) N_I(\mathbf{x}) N_J(\mathbf{x}) \dot{\mathbf{v}}_J(t) da = \sum_{I=1}^{N_n} \mathbf{w}_I \cdot \sum_{J=1}^{N_n} M_{IJ}(t) \dot{\mathbf{v}}_J(t), \tag{12}$$

$$-\int_{\Omega(t)} \text{grad } \mathbf{w}^h : \mathbf{N} da = -\sum_{I=1}^{N_n} \mathbf{w}_I \cdot \int_{\Omega(t)} \text{grad} N_I(\mathbf{x}) \cdot \mathbf{N}(\mathbf{x},t) da \tag{13}$$

$$\int_{\partial\Omega_t(t)} \mathbf{w}^h \cdot \bar{\mathbf{t}} ds = \sum_{I=1}^{N_n} \mathbf{w}_I \cdot \int_{\partial\Omega_t(t)} \bar{\mathbf{t}}(\mathbf{x},t) N_I(\mathbf{x}) ds \tag{14}$$

$$\int_{\Omega(t)} \mathbf{w}^h \cdot \mathbf{F}^{\text{ext}} da = \sum_{I=1}^{N_n} \mathbf{w}_I \cdot \int_{\Omega(t)} \mathbf{F}^{\text{ext}}(\mathbf{x},t) N_I(\mathbf{x}) da. \tag{15}$$

In order to complete the spatial discretization a quadrature rule must be given to evaluate the integrals in equations (12)–(15). The material points are used as quadrature points and the integrals become sums over material points. For example, equation (12) discretizes the inertial term. The consistent mass matrix M_{IJ} has components

$$M_{IJ}(t) = \int_{\Omega(t)} m(\mathbf{x},t) N_I(\mathbf{x}) N_J(\mathbf{x}) da \sim \sum_{p=1}^{N_p} m(\mathbf{x}_p(t),t) N_I(\mathbf{x}_p(t)) N_J(\mathbf{x}_p(t)) \Omega_p(t) = \sum_{p=1}^{N_p} m_p(t) N_I(\mathbf{x}_p) N_J(\mathbf{x}_p), \tag{16}$$

where the material-point mass is $m_p(t) = m(\mathbf{x}_p(t), t)\Omega_p(t) = \rho(\mathbf{x}_p(t), t) h_p(t) \Omega_p(t)$. Note that conservation of mass can be expressed by the equation, $\rho(\mathbf{X}, t) J(\mathbf{X}, t) h(\mathbf{X}, t) = \rho_0(\mathbf{X}) h(\mathbf{X}, 0)$. If mass is conserved, then using equation (5) in the conservation equation, shows that the current material point mass is the same as its initial mass, $m_p(t) = m_p(0) \equiv m_p$, where the initial mass is given in equation (3).

[19] Equation (13) provides the nodal values of the internal forces

$$\mathbf{F}_I^{\text{int}}(t) = -\int_{\Omega(t)} \text{grad } N_I(\mathbf{x}) \cdot \mathbf{N}(\mathbf{x},t) da \sim -\sum_{p=1}^{N_p} \mathbf{G}_{Ip} \mathbf{N}_p(t) \Omega_p(t). \tag{17}$$

In the above equation, simpler notation has been introduced for the gradient of the shape function, $\mathbf{G}_{Ip} = \text{grad } N_I(\mathbf{x})|_{\mathbf{x}=\mathbf{x}_p}$, as well as for the stress at the material point position, $\mathbf{N}_p(t) = \mathbf{N}(\mathbf{x}_p(t), t)$.

[20] Finally, the nodal values of the external forces arise from the external body forces, equation (15), plus the applied traction, equation (14)

$$\mathbf{F}_I^{\text{ext}}(t) = \int_{\Omega(t)} \mathbf{F}^{\text{ext}}(\mathbf{x},t) N_I(\mathbf{x}) da + \int_{\partial\Omega_t(t)} N_I(\mathbf{x}) \bar{\mathbf{t}}(\mathbf{x},t) ds. \tag{18}$$

[21] The weak form of the momentum balance equates (12) to the sum of the forces (equations (13)–(15)). Since the weak form must hold for arbitrary $\mathbf{w}_I$, except at constrained nodes on the boundary where the displacement is prescribed, we obtain the semidiscrete equation for the nodal acceleration at unconstrained nodes

$$\sum_{J=1}^{N_n} M_{IJ}(t) \dot{\mathbf{v}}_J(t) = \mathbf{F}_I^{\text{int}}(t) + \mathbf{F}_I^{\text{ext}}(t). \tag{19}$$

The momentum equation is solved for the acceleration at unconstrained nodes, which is then integrated in time to obtain the corresponding velocity and displacement. Nodes constrained by the displacement boundary conditions move according to those prescribed constraints.

3.2. Time Discretization

[22] The semidiscrete equation (19) can be discretized in time using any scheme. Perhaps the simplest scheme is an explicit method with a lumped mass matrix. Lumping the

mass is standard practice for explicit finite element methods [*Belytschko et al.*, 2000; *Hughes*, 2000]. This matrix is diagonal with the diagonal entries obtained by summing over the corresponding row of the consistent mass matrix and using the property $\sum_{J=1}^{N_n} N_J(\mathbf{x}) = 1$,

$$M_I(t) = \int_{\Omega(t)} m(\mathbf{x}, t) N_I(\mathbf{x}) da \sim \sum_{p=1}^{N_p} m_p N_I(\mathbf{x}_p). \tag{20}$$

With the lumped mass matrix equation (19) becomes

$$M_I(t)\dot{\mathbf{v}}_I(t) = \mathbf{F}_I^{\text{int}}(t) + \mathbf{F}_I^{\text{ext}}(t). \tag{21}$$

The advantage of mass lumping is that a simple explicit time-integration method does not require a time consuming matrix inversion in each step.

[23] A superscript n is used to indicate the computed solution at time t^n. Then, for an explicit scheme, the discretized momentum equation is

$$M_I^n \mathbf{a}_I^n = \mathbf{F}_I^{n,\text{int}} + \mathbf{F}_I^{n,\text{ext}}. \tag{22}$$

This equation is solved for $\mathbf{a}_I^n$, the acceleration at time t^n. In order to solve this equation, the elements of the lumped mass matrix must be computed. Equation (20) defines the mass at a node at time t^n,

$$M_I^n = \sum_{p=1}^{N_p} m_p N_I\left(\mathbf{x}_p^n\right). \tag{23}$$

The internal forces at time t^n come from (17),

$$\mathbf{F}_I^{n,\text{int}} = -\sum_{p=1}^{N_p} \mathbf{G}_{Ip}^n \mathbf{N}_p^n \Omega_p^n. \tag{24}$$

The external forces are computed as in (18).

[24] A method that is formally second order in time is obtained by staggering the velocity and displacement in time. Let $t^{n+1/2} = \frac{1}{2}(t^{n+1} + t^n)$, $\Delta t^n = t^{n+1/2} - t^{n-1/2}$, and $\Delta t^{n+1/2} = t^{n+1} - t^n$. A centered difference formula for the acceleration is

$$\mathbf{a}_I^n = \frac{\mathbf{v}_I^{n+1/2} - \mathbf{v}_I^{n-1/2}}{t^{n+1/2} - t^{n-1/2}} = \frac{1}{\Delta t^n}\left(\mathbf{v}_I^{n+1/2} - \mathbf{v}_I^{n-1/2}\right). \tag{25}$$

This formula can be converted into an integration formula for the velocity

$$\mathbf{v}_I^{n+1/2} = \mathbf{v}_I^{n-1/2} + \Delta t^n \mathbf{a}_I^n. \tag{26}$$

Similarly, the velocity can be obtained by differencing the displacement

$$\mathbf{v}_I^{n+1/2} = \frac{\mathbf{u}_I^{n+1} - \mathbf{u}_I^n}{t^{n+1} - t^n} = \frac{1}{\Delta t^{n+1/2}}\left(\mathbf{u}_I^{n+1} - \mathbf{u}_I^n\right). \tag{27}$$

Likewise, this formula can be converted into an update for the displacement

$$\mathbf{u}_I^{n+1} = \mathbf{u}_I^n + \Delta t^{n+1/2} \mathbf{v}_I^{n+1/2}. \tag{28}$$

[25] In order to solve these equations, the velocity field $\mathbf{v}_I^{n-1/2}$ must be initialized from the material points. The nodal mass is defined in equation (23). In a similar manner, we can define the momentum at a node on the current background mesh from the material points

$$M_I^n \mathbf{v}_I^{n-1/2} = \sum_{p=1}^{N_p} m_p \mathbf{v}_p^{n-1/2} N_I\left(\mathbf{x}_p^n\right). \tag{29}$$

Divide the momentum (29) by the mass (23) to obtain the velocity $\mathbf{v}_I^{n-1/2}$. Now equation (26) can be used to find $\mathbf{v}_I^{n+1/2}$, given the acceleration from (22). It is actually not necessary to explicitly update the nodal displacement.

[26] Once the solution on the background grid is obtained, the information must be used to update the material points. Over the time step, we imagine the grid nodes to move in the computed velocity field (i.e., the grid is Lagrangian). The material points move in this flow in a manner consistent with the grid solution and the interpolation to the interior of the elements via the shape functions. Thus the material points move according to

$$\begin{aligned} \mathbf{x}_p^{n+1} &= \mathbf{x}_p^n + \Delta t^{n+1/2} \sum_{I=1}^{N_n} \mathbf{v}_I^{n+1/2} N_I\left(\mathbf{x}_p^n\right) \\ \mathbf{v}_p^{n+1/2} &= \mathbf{v}_p^{n-1/2} + \Delta t^n \sum_{I=1}^{N_n} \mathbf{a}_I^n N_I\left(\mathbf{x}_p^n\right). \end{aligned} \tag{30}$$

Information from the grid is also used to update the material-point stress state. Some models are based on the rate of deformation (symmetric part of the velocity gradient), other models rely on the deformation gradient. The rate of deformation is computed as

$$\left(\text{grad}_{\mathbf{x}^n} \mathbf{v}|_{\mathbf{x}=\mathbf{x}_p}\right)_{\text{sym}} = \sum_{I=1}^{N_n} \left(\mathbf{G}_{Ip}^n \mathbf{v}_I^{n+1/2}\right)_{\text{sym}} \tag{31}$$

and the deformation gradient is updated by

$$\begin{aligned} \mathbf{F}_p^{n+1} &= \left(\text{grad}_{\mathbf{x}^n} \mathbf{x}^{n+1}\right) \mathbf{F}_p^n \\ &= \left(\mathbf{I} + \Delta t^{n+1/2} \sum_{I=1}^{N_n} \mathbf{G}_{Ip}^n \mathbf{v}_I^{n+1/2}\right) \mathbf{F}_p^n, \end{aligned} \tag{32}$$

where $\boldsymbol{I}$ is the second order identity tensor. This information is used along with any required internal variables to update the material-point stress. Details are given in section 4. We have already noted how the determinant of the deformation gradient is also needed to update the material-point area.

[27] Once the material points have been updated the possibly distorted Lagrangian grid is no longer needed. A new grid can be defined (possibly a regular square grid), and the solution procedure is repeated for the next compu-

tational time step. Having the mesh under user control avoids mesh distortion associated with solutions on Lagrangian meshes.

[28] In summary, the steps of the MPM computational cycle are as follows:

[29] 1. In the first step of the computational cycle, information carried by the material points is projected on to the background mesh. Material-point mass is interpolated to the nodes of a background mesh (23), as is the momentum (29). The ratio of mass to momentum gives the velocity on the background mesh. Internal forces at the nodes of the background mesh are determined directly from the material-point stress using a gradient weight and the material-point volume (24).

[30] 2. External forces on the nodes of the background mesh (18), for example due to wind or ocean drag, Coriolis or tilt forces, are added to the internal forces (the stress divergence), and the momentum equation is solved on the background mesh. During this step, the mesh is assumed to distort in the flow. The Lagrangian formulation means that the acceleration does not contain the convection term which can cause significant numerical error in purely Eulerian approaches. During this Lagrangian phase of the calculation, each element is assumed to deform in the flow of material so that points in the interior of the element move in proportion to the motion of the nodes. That is, given the velocity at the nodes, element shape functions are used to map the nodal velocity continuously to the interior of the element (8).

[31] 3. The mesh solution computed in the last step is used to update the solution for each material point. The positions of the material points are updated by moving them in the single-valued, continuous velocity field that arises from the mapping through element shape functions (30). Similarly, the velocity of a material point is updated by mapping the nodal accelerations to the material point position (30). Because the velocity field is single-valued, interpenetration of material is precluded, and also no-slip contact between impinging bodies is automatic. In its simplest form, strain increments are obtained from gradients of the nodal velocities on the background mesh, evaluated at the material point positions (31). Then, given a strain increment at a material point, along with current values of history variables and material parameters, continuum constitutive routines are used to update the stress and history variables. Constitutive routines are always evaluated separately for each material point, so there is no artificial numerical mixing of materials as in Eulerian schemes.

[32] 4. The material points now carry all information about the solution; therefore, one can choose whether to continue the calculation in the Lagrangian frame or map information from the material points to another grid. Most often MPM is used with a fixed, regular grid at the start of each time step, as is the case in this paper.

[33] With this method, the background mesh does not need to conform to the boundary of the ice. Instead, a grid is constructed to cover the potential domain for the boundary-value problem being solved. Then each body of ice is defined by a collection of material points. Complicated geometry is easily modeled by filling regions with material points. This process is much easier than standard meshing. Land mass boundaries can also be represented by material points and treated as rigid bodies in the calculation. An interface to atmosphere and ocean codes should also be relatively straightforward since the MPM data can be interpolated to any grid. The coupling to these codes can be done through fluxes or with the primitive quantities.

[34] The basic method described above will be augmented with a description of thermodynamics modeled by tracking the local ice thickness and compactness. These will be material-point variables and their evolution will follow similar equations to those in *Thorndike et al.* [1975] and *Coon et al.* [1998] for the convection and rate of growth (or melting). More sophisticated models and accounting of the heat budget are possible within our framework but are left for future studies.

4. Constitutive Models

[35] A variety of models have been proposed for the behavior of sea ice, including isotropic plastic [*Rothrock*, 1975], isotropic elastic-plastic [*Coon et al.*, 1974; *Pritchard*, 1975], isotropic viscous-plastic [*Hibler*, 1977b, 1979; *Flato and Hibler*, 1989; *Ip et al.*, 1991], isotropic elastic-viscous-plastic [*Hunke and Dukowicz*, 1997; *Hunke*, 2001], as well as some anisotropic models [*Hibler and Schulson*, 2000; *Wilchensky and Feltham*, 2004]. The continuum model most often applied to sea ice is an isotropic, viscous-plastic model [*Hibler*, 1979] or a modification based on an elastic-viscous-plastic model [*Hunke and Dukowicz*, 1997]. The latter model is introduced for numerical efficiency rather than to model specific elastic aspects of sea ice. Section 4.1 describes the implementation of the former model in MPM. Section 4.2 describes another constitutive model, a new elastic-decohesive model that allows ice to deform elastically to a point, but with sufficient loading the ice fractures and leads form. The result is that ice with leads behaves anisotropically. For example, an element of ice with a lead offers no resistance to opening motion perpendicular to the lead, but does offer resistance if motion is parallel to the lead. It is worth noting that any of the aforementioned constitutive models could be implemented in the MPM framework.

[36] As described previously, MPM evaluates the constitutive model independently for each material point. Thus, to simplify the notation, the subscript p indicating a material point quantity is omitted in this section but is assumed.

4.1. Viscous-Plastic Rheology

[37] The classical viscous-plastic model [*Hibler*, 1979] computes the depth-integrated, extra stress $\mathbf{N}$ from the strain rate $\dot{\varepsilon} = \frac{1}{2}(\nabla\mathbf{v} + (\nabla\mathbf{v})^T)$ according to the formula

$$\mathbf{N} = 2\eta\dot{\varepsilon} + [\zeta - \eta]\mathrm{tr}(\dot{\varepsilon})\mathbf{I} - \frac{1}{2}P\mathbf{I}. \qquad (33)$$

The notation tr(·) indicates the trace of a tensor. The strain rate at a material point is obtained from formula (31). The viscosity coefficients η and ζ in this formulation are nonlinear functions of the strain rate and the maximum ice strength $P_{\max}$. Specifically, $\zeta = P_{\max}/2\Delta$, and $\eta = \zeta/e^2$ where e is the ratio of the principal axes of the elliptical yield curve [*Hibler*, 1977a], and Δ depends on e and the strain rate

$$\Delta = \Big[\big(\dot{\varepsilon}_{11}^2 + \dot{\varepsilon}_{22}^2\big)\big(1 + e^{-2}\big). \\ + 4e^{-2}\dot{\varepsilon}_{12}^2 + 2\dot{\varepsilon}_{11}\dot{\varepsilon}_{22}\big(1 - e^{-2}\big)\Big]^{1/2}. \tag{34}$$

[38] As currently defined, the viscosity coefficients can become arbitrarily large for small strain rates. To avoid this difficulty, these coefficients are chosen to be the minimum of the values specified above and some large limiting values that depend on the ice strength. The limiting values are taken to be $\zeta_{\max}$ = (2.5×10^8 s) $P_{\max}$ and then $\eta_{\max} = \zeta_{\max}/e^2$. To insure that there is no stress at zero strain rates, a replacement pressure P is used in equation (33), where $P = 2\Delta\zeta_1$, $\zeta_1 = \min[P_{\max}/2\Delta, \zeta_{\max}]$.

[39] The maximum ice strength, $P_{\max}$, is taken to be a function of the average ice thickness, h, and its compactness, A, according to the formula $P_{\max} = P^* \ hA$ $\exp(-C(1 - A))$ which includes the fixed empirical constants P^* and C. The ice thickness and compactness evolve due to thermodynamics and ice dynamics. Thermodynamics causes changes due to melting and freezing of ice. Dynamics causes changes through the creation of leads during divergent flow and closing of open water or ridging of ice during convergent flow. Thus, in MPM, the thickness and compactness become material-point quantities and are naturally advected as the material points move. With these variables, it is possible to keep track of two ice categories, thin and thick ice.

[40] A simple model [*Hibler*, 1979] for the evolution of $\bar{h} = hA$ and A consists of

$$\dot{\bar{h}} = -(\nabla\cdot\mathbf{v})\bar{h} + S_h, \qquad \dot{A} = -(\nabla\cdot\mathbf{v})A + S_A. \tag{35}$$

These equations are simple continuity equations for $\bar{h}$ and A with thermodynamic source terms S_h and S_A. In this paper, we neglect thermodynamic changes and take S_h = S_A = 0.

4.2. Elastic-Decohesive Rheology

[41] We will also use a newly developed elastic-decohesive model for the numerical simulations presented in section 6. This model is described fully by *Schreyer et al.* [2006], so only a summary is presented. The most salient feature of this model is the explicit representation of lead formation. The basic idea is that the ice is modeled as an isotropic, elastic solid until failure begins. Failure begins when some measure of the traction on a failure surface reaches a critical value. During decohesive failure, the traction on the failure surface is reduced from non-zero to zero. At complete separation, there is no traction on the new free surface, however, other stress components can still be nonzero. There are three aspects of the model that need to be examined: (1) the criterion to initiate failure, (2) the determination of the failure direction (i.e., orientation of the crack or lead), and (3) the mechanism for traction reduction. These aspects are discussed briefly below.

[42] Given the current, extra Cauchy stress at a material point, $\boldsymbol{\sigma}$, the criterion to initiate failure is expressed by a failure function $F(\boldsymbol{\sigma})$. The elastic region in stress space is given by the condition $F < 0$. Failure occurs when $F = 0$ and $F > 0$ is not allowed. This failure function is similar to a yield function in plasticity theory. Figure 10 shows an example of the failure function for the decohesion model used in this work. Let $\mathbf{n}$ be a vector normal to the lead and $\mathbf{t}$ be a vector tangent to the lead, so that $\mathbf{n}$ and $\mathbf{t}$ form a right-handed coordinate system. The functional form of F is expressed in terms of the normal and tangential components of the traction on the failure surface, $\tau_n = \mathbf{n}\cdot\boldsymbol{\sigma}\cdot\mathbf{n}$ and $\tau_t = \mathbf{t}\cdot\boldsymbol{\sigma}\cdot\mathbf{n}$, and the tangential stress, $\sigma_{tt} = \mathbf{t}\cdot\boldsymbol{\sigma}\cdot\mathbf{t}$. The function F combines a brittle decohesion function with a ductile failure function to allow for multiple modes of failure including mixed modes [*Schreyer et al.*, 2006], as observed experimentally [*Schulson*, 2001].

[43] Specifically, the stress at which failure occurs is the envelope in stress space defined by $F(\boldsymbol{\sigma}) = \max_n F_n(\boldsymbol{\sigma}, \mathbf{n})$, where $F_n(\boldsymbol{\sigma}, \mathbf{n})$ is the failure function for a crack oriented with a normal to its surface given by $\mathbf{n}$

$$F_n = \left(\frac{\tau_t}{s_m\tau_{sf}}\right)^2 + e^{\kappa B_n} - 1 \\ B_n = \frac{\tau_n}{\tau_{nf}} - f_n\left(1 - \frac{\langle -\sigma_{tt}\rangle^2}{f_c'^2}\right). \tag{36}$$

In this expression, τ_{nf} is the tensile failure stress, f'_c is the failure stress in uniaxial compression, and τ_{sf} is the failure stress in shear. The McCauley bracket, $\langle\cdot\rangle$ indicates that the tangential stress term only appears when σ_{tt} is negative. The shear magnification factor, s_m, magnifies τ_{sf} when the ice is under compression. The softening parameter, f_n, is a function of the normal component of the jump in displacement, u_n and has a value of one at the initiation of decohesion and decreases to zero as the lead opens. Specifically, $f_n = \langle 1 - u_n/u_0\rangle$. The lead is fully opened when u_n, reaches a predetermined value, u_0. In pure shear, the only nonzero term in F_n is the tangential component of traction, τ_t. By the definition of τ_{sf} and f_n, failure initiates when $\tau_t = \tau_{sf}$ and $f_n = 1$, giving the relation $s_m^2(1 - e^{-\kappa}) = 1$ to define the parameter κ.

[44] The orientation of the lead is determined as the $\mathbf{n}$ that maximizes F. The softening parameter forces the traction on the crack surface to zero as the crack opens (u_n increases). Thus when the crack is fully open, there is no traction on the surface. In this manner, the parameter f_n accounts for the reduction of traction as decohesion proceeds. At each loading step we find the critical direction $\mathbf{n}$ for which F is largest. As a lead with a particular orientation begins to open, the softening makes it likely that this orientation will remain the critical direction. However, it is possible that a changing stress state will make another direction critical, in which case a second lead can form intersecting the first. In this manner, the model accommodates multiple leads at a point, representing crack branching. If weak areas are known to exist in the ice, the softening parameter f_n can be initialized with a value less than one to account for this information. The numerical implementation of the elastic-decohesive constitutive model is similar to to that for an elastic-plastic constitutive model [*Schreyer et al.*, 2006].

Note, the depth-integrated, extra stress $\mathbf{N} = \boldsymbol{\sigma} h$, is used in the momentum equation.

5. Other Particle-in-Cell Methods for Pack Ice

[45] *Flato* [1993] applied PIC technology to sea ice for operational sea-ice forcasting using the traditional viscous-plastic rheology. His primary concern was obtaining a more accurate prediction of the ice edge. He used material points (particles) with fixed volume to advect the ice thickness and compactness. The grid was a square C-grid. The *x*-component of the velocity was located at the midpoint of vertical edges and the *y*-component of velocity was located at the midpoint of horizontal edges. The same interpolation scheme was used as in MPM, so the C-grid velocity components were averaged to obtain nodal values and then interpolated to the material-point positions. The material-point trajectories were computed using a mid-point rule in time. This rule required two interpolations of the grid velocity to the material points per time step, rather than just one in MPM. These details aside, Flato made minimal use of the material points. He basically used the *Zhang and Hibler* [1997] approach to solve the momentum equation on the grid and used the material points only to obtain values of thickness and compactness on the grid for use in that solution technique. Thus the principal difference between *Flato* [1993] and the current work is the evaluation of the constitutive equation on the grid rather than on the material points. This change makes it easier to implement constitutive models for solids with history-dependent, internal variables.

[46] *Zhang and Savage* [1998] use essentially the same method as *Flato* [1993], except their grid is a square, staggered, B-grid. Thus the grid velocity is located at the nodes as in MPM. Thickness and compactness are interpolated to cell centers and a pseudo-time stepping algorithm similar to *Zhang and Hibler* [1997] is used to solve the momentum equation on the grid. A similar approach is used by *Sayed and Carrieres* [1999], except the material points are assigned volume and area, rather than thickness and compactness. The particle area is used to obtain grid compactness, and then grid compactness and particle volume are used to determine grid ice thickness. A simple ridging model is included as follows. If the ice area (from the particles) divided by grid cell area is greater than one, the total ice area is reset to unity. The reduction in area is used to reduce the area of the particles in the cell by a constant factor. The thickness is then increased in order to maintain a constant volume. *Sayed et al.* [2002] modified this ridging model to reduce the particle area according to the particle thickness, essentially ridging the thinnest ice first. These authors have also begun to apply thermodynamics to the particles by interpolating thickness calculations on the grid to specific particle thicknesses.

[47] *Savage* [2002] and *Kubat et al.* [2005] have continued investigating thickness models for their PIC method. The approach has been to use a large number of particles in the simulation and to use the particle statistics to obtain a thickness distribution for a cell on the grid. We propose a different method where each material point has an associated thickness distribution. This approach should reduce the number of material points necessary in a simulation and cut the computational cost, although it is yet to be implemented and tested.

6. Examples

[48] The purpose of this section is to illustrate properties of MPM by considering some idealized problems. First, we examine the benefits of solving transport equations for sea ice models using the Lagrangian description provided by the material points. Next, we consider a wind-driven pack with low strength, using both the viscous-plastic and elastic-decohesive rheologies. This simulation shows the ability of MPM to handle large deformations and different rheologies.

6.1. Convection Tests

[49] Sea ice models contain transport equations for quantities such as the area and various ice-thickness categories. For example, in standard runs of the Los Alamos sea ice model, CICE, there are 46 such transported fields [*Lipscomb and Hunke*, 2004]. Thus there is a need for an accurate and efficient transport algorithm. Equations (35) with $S_A = S_h = 0$ are generic transport equations for the compactness A and volume $\bar{h} = hA$. These equations together imply an equation for transport of the thickness, $\dot{h} = 0$. Thus thickness is unchanged for a material point. Recall that the material time derivative is

$$\dot{h} = \frac{\partial h}{\partial t} + \mathbf{v} \cdot \mathrm{grad} h = 0. \qquad (37)$$

The required discretization of the nonlinear convective term $\mathbf{v} \cdot \mathrm{grad}\, h$ on an Eulerian grid leads to undesired results such as numerical diffusion, oscillations near discontinuities or dispersion errors. These errors are completely avoided in MPM since each material point is assigned a thickness and transport is accomplished by moving the material point.

[50] To illustrate transport in MPM, we consider three test problems also solved in *Lipscomb and Hunke* [2004]. The first is uniform advection of a square mesa. The second problem is a rigid-body rotation of a cylinder, and the final problem involves transport in a converging flow field. These convection problems are all solved in MPM using a uniform, square, background grid. Since the velocity field should not be advected for these problems, we skip the first step of MPM and do not map the material-point velocity to the grid. Instead, the velocity is a prescribed function of the current position and the grid values are set. The second step of MPM, solution of the momentum equation, results in no change to the velocity because there are no internal or external forces acting on the ice. The third step of MPM moves the material points in the given velocity field and updates their properties. In this step, ordinary differential equations in time are solved for A and $\bar{h}$, or for any other material-point quantity. The fixed background grid is retained for the entire calculation.

[51] For the first problem, the square grid is 32 × 32 with side length 4. Thus the entire computational domain lies between $0 \leq x \leq 128$ and $0 \leq y \leq 128$. A square region of ice is defined that has initial height, $h = 1$. The lower left corner of the ice is located at $(x, y) = (20, 20)$, with side length 20. The velocity field is directed northeastward at a 45° angle to the *x*-axis, $\mathbf{v} = (1, 1)$. The model is stepped

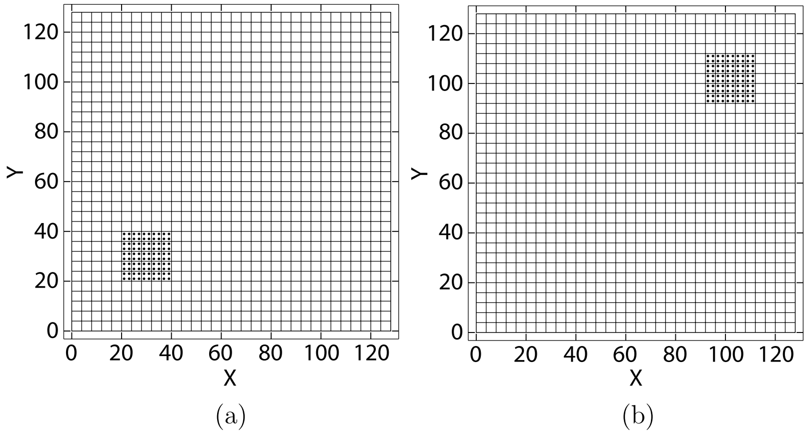


Figure 1. (a) Original and (b) final configuration of material points making up a square region of ice on a 32 × 32 background grid. The ice has been transported in a uniform velocity field, **v** = (1, 1), to the northeast.

forward 72 units in time. The exact solution is the same square of ice with height one, displaced 72 units in each of the *x* and *y* directions. Since this velocity field is divergence-free, the area, *A*, and the volume, $\bar{h}$, satisfy the same equation as *h*, and will have similar solutions. Figure 1 shows the MPM solution. The initial configuration, shown in Figure 1a, is discretized using four material points per computational element. Figure 1b shows the final configuration of the ice. Since bilinear shape functions are used to map quantities between the grid and material points, linear fields are mapped without error. Thus the velocity field in which the material points move is exact. The time integration scheme computes the position of the material points without error for this constant velocity field, and therefore the exact solution to this problem is obtained using MPM since the material points carry a thickness one. Figure 2 shows the solution carried by the material points projected onto the background mesh. The only error in the mesh representation comes at the edges of the ice where a transition from a value of $h = 1$ to $h = 0$ must occur over one mesh width. Figure 2 shows how this transition becomes sharper as the mesh is refined. Eulerian, grid-based solutions have several undesirable properties when applied to this problem. The simplest, first-order, upwind scheme is highly dissipative. The peak thickness decreases over time and the width of the mesa increases. More sophisticated schemes like MPDATA have less diffusion, and therefore maintain a shaper profile. However, MPDATA is not monotonicity preserving so the peak ice thickness increases with time beyond the value one [*Lipscomb and Hunke*, 2004]. The incremental remapping algorithm does not create spurious peaks and has reduced diffusion compared with the upwind scheme. However, even this method rounds the corners of the square profile [*Lipscomb and Hunke*, 2004]. The proficiency of the method also depends on the mesh resolution. MPM results are independent of mesh size, size of the ice, or its speed.

[52] The second test problem involves moving a cylinder of ice in a velocity field given by a rigid-body rotation. This velocity is spatially varying, but constant in time, and has the form $\mathbf{v}(\mathbf{x}, t) = \omega \mathbf{e}_3 \times (\mathbf{x} - \mathbf{x_c})$ where $\mathbf{x_c}$ is the center of rotation and ω is the angular speed. We use the same computational domain and grid as in the last problem, also with 4 material points per element. The center of rotation is taken to be the center of the domain, $\mathbf{x}_c = (64, 64)$, and $\omega = 1/64$ so that the speed at the midpoints of the edges of the domain is unity. A cylinder of ice with initial height $h = 1$ and radius 10 units, is placed in this domain with its center at (106,64). The time step is chosen so that the cylinder makes one revolution in 1000 steps. Again, the velocity field is divergence free so the exact solution consists of the cylinder transported without change in shape. As in the last example, as long as the material point trajectories are computed accurately, the thickness will be exactly correct.

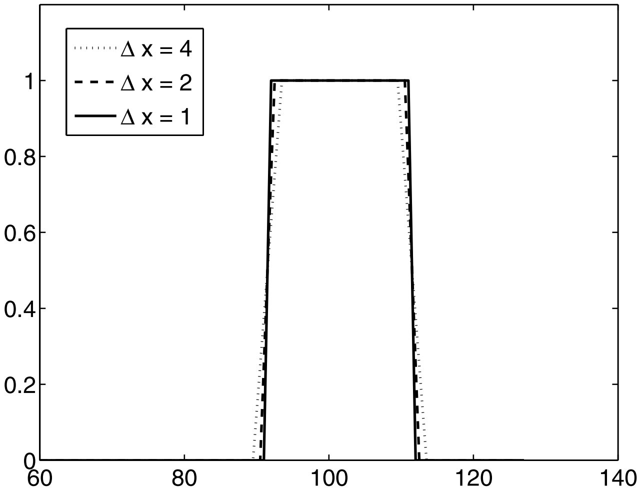
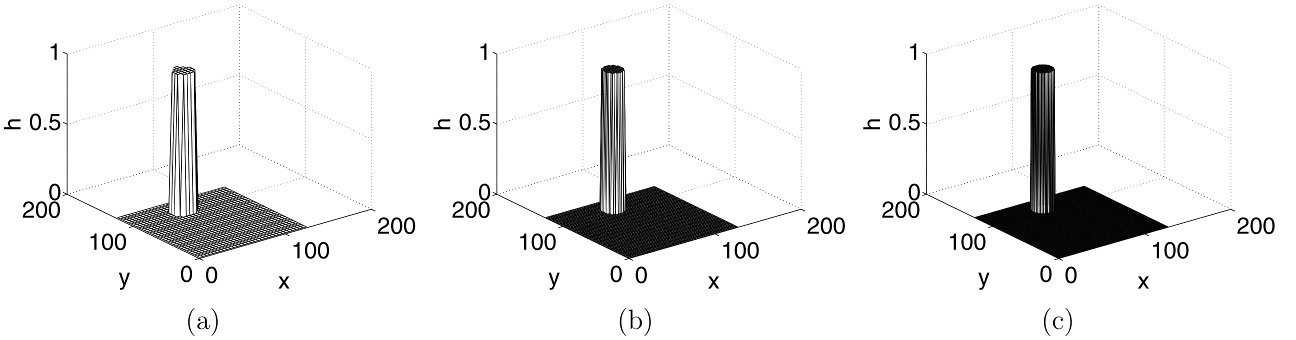


Figure 2. Cross-section through elements at $y = 100$ showing the final thickness profile of a square region of ice, as in Figure 1b. The thickness is interpolated onto the background grid. Profiles are compared for three simulations on mesh sizes with $\Delta x = \Delta y = 4$, $\Delta x = \Delta y = 2$ and $\Delta x = \Delta y = 1$.



(a) (b) (c)

Figure 3. Final configuration of a cylinder of ice after one full rotation through (a) 32 × 32, (b) 64 × 64 and (c) 128 × 128 meshes. The material-point thickness is projected onto the background grid to construct the surface.

Figure 3 shows the solution using MPM after one full rotation of the cylinder on three different mesh sizes. The finest mesh is the one used in *Lipscomb and Hunke* [2004] to solve this problem. Again, Eulerian schemes either spread the cylinder or produce unphysical peaks in the thickness. The MPM solution gives the exact ice thickness. As the mesh is refined, the cylinder geometry is better represented.

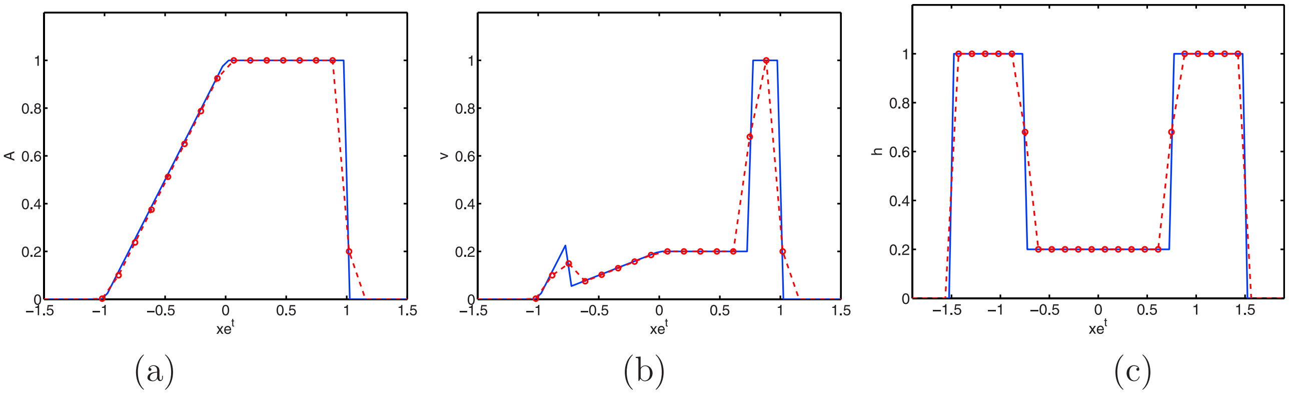
[53] The third test problem involves transport in a convergent flow field given by $\mathbf{v}(x, y) = (-x, 0)$. Because of the nonzero divergence, A and $\bar{h}$ evolve differently than h. This example illustrates compatibility of the fields; namely if A and $\bar{h}$ are obtained numerically and then $h = \bar{h}/A$ is computed by division when $A \neq 0$, unphysical peaks in h are not produced [*Lipscomb and Hunke*, 2004]. For arbitrary initial conditions, $A(x, y, 0) = A_0(x, y)$, $\bar{h}(x, y, 0) = \bar{h}_0(x, y)$, and $h(x, y, 0) = h_0(x, y)$, the solution at time t is

$$\begin{aligned} A(x,y,t) &= e^t A_0(xe^t, y) \\ \bar{h}(x,y,t) &= e^t \bar{h}_0(xe^t, y) \\ h(x,y,t) &= h(xe^t, y). \end{aligned} \tag{38}$$

We solve this problem as in *Lipscomb and Hunke* [2004] using a background mesh with size $\Delta x = \Delta y = 0.05$ and a time step $\Delta t = 0.025$. The model is integrated in time for 40 steps, giving a final time, $t = 1$. The initial conditions, and thus the solutions, are independent of y

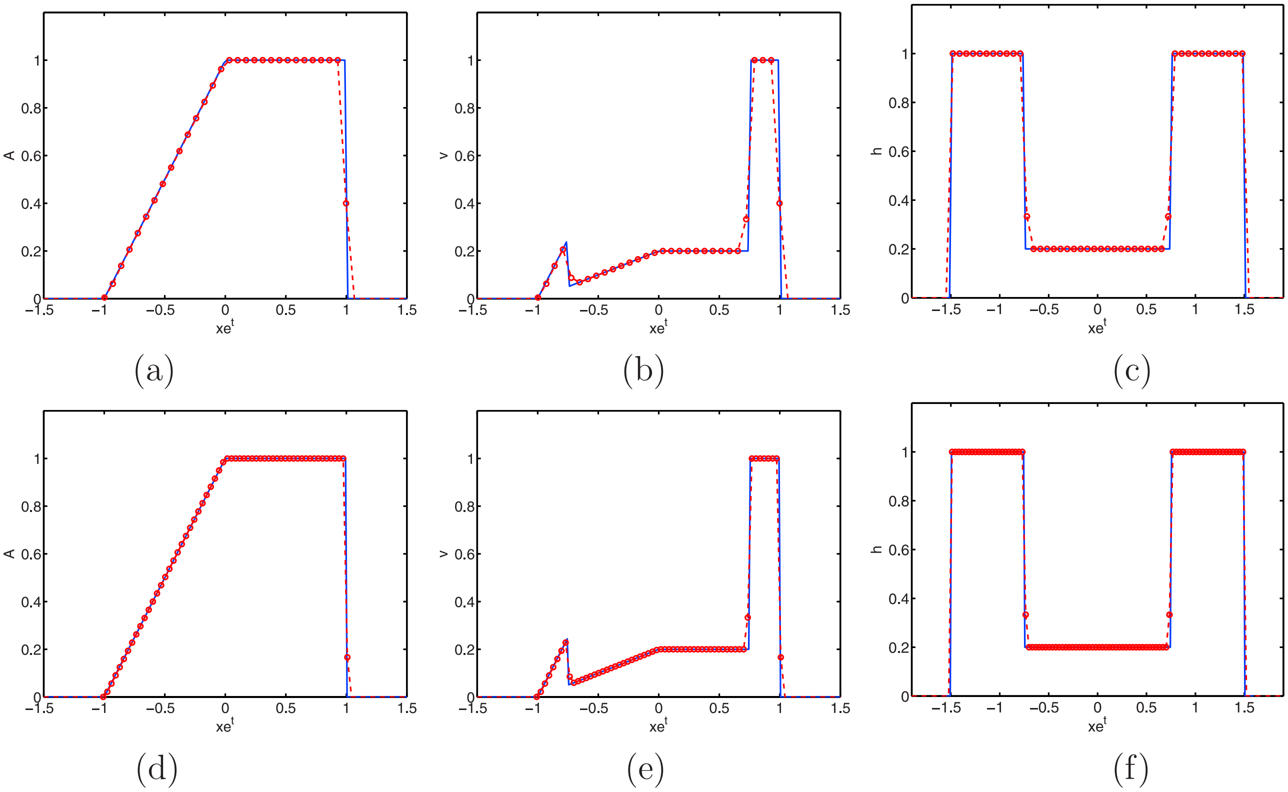
$$\begin{aligned} A_0(x) &= \begin{cases} 1 + x & \text{for } -1 \leq x \leq 0, \\ 1 & \text{for } 0 \leq x \leq 1, \\ 0 & \text{otherwise.} \end{cases} \\ h_0(x) &= \begin{cases} 1 & \text{for } |x| \geq 0.75, \\ 0.2 & \text{otherwise.} \end{cases} \end{aligned} \tag{39}$$

Since the solutions are independent of y, we use a computational domain wider in the x direction than in the y direction, given by $-1.6 \leq x \leq 1.6$ and $0 \leq y \leq 0.5$. The ice initially occupies $-1.5 \leq x \leq 1.5$, $0 \leq y \leq 0.5$ and contracts in the flow with time to about 1/3 of its initial length in the x direction. The vertical edges are stress free and the top and bottom boundaries are allowed to slip freely tangentially. The computed solution is also independent of y and we display the x-variation in the solution in Figure 4. If the x-coordinates are scaled by e^t and $A(x, t)$ and $\bar{h}(x, t)$ are multiplied by e^{-t}, the analytical solution, equation (38), shows that the solution at time t should be the same as the initial conditions. Figure 4 compares the scaled numerical



(a) (b) (c)

Figure 4. Exact (solid line) and computed (dashed line and circles) scaled solutions at $t = 1$ for (a) area, (b) volume and (c) thickness using a mesh size $\Delta x = \Delta y = 0.05$ and a time step $\Delta t = 0.025$.



(a) (b) (c)

(d) (e) (f)

Figure 5. Exact (solid line) and computed (dashed line and circles) scaled solutions at $t = 1$ for (a) area, (b) volume and (c) thickness using a mesh size $\Delta x = \Delta y = 0.025$ and a time step $\Delta t = 0.0125$; and (d) area, (e) volume and (f) thickness using a mesh size $\Delta x = \Delta y = 0.0125$ and a time step $\Delta t = 0.00625$.

and analytical solutions at time $t = 1$. Note that the linear velocity is interpolated to the material points exactly using the bilinear shape functions. Also, derivatives of a linear field are computed exactly in MPM. The time integration of the material-point positions and the material-point values of A and $\bar{h}$ should also be exact, since the equations are linear. The only errors are associated with projecting the the computed material-point profiles onto the background grid. The figure shows that these errors are small. Initially there are 60 grid points of the background mesh in the x-direction and 10 in the y-direction over the ice. At time $t = 1$, there are only about 16 grid points covering the ice in the x-direction. The low resolution of the final configuration is apparent, especially in the plot of $\bar{h}$. For comparison, the solution is shown computed with mesh sizes $\Delta x = \Delta y = 0.025$ and $\Delta x = \Delta y = 0.0125$ in Figure 5. These meshes are able to resolve the variations in $\bar{h}$ and produce high fidelity solutions.

6.2. Wind-Driven Ice

[54] The purpose of this section is to demonstrate the full MPM algorithm using the two constitutive models described in section 4. *Schulkes et al.* [1998] identify a simple test problem that seems to distinguish properties of various viscous and viscous-plastic ice rheologies. In that work, a simple rectangular region of ice is subjected to wind forcing and ocean drag. Two adjacent boundaries of the rectangle represent free surfaces and the other two boundaries represent shorelines. Along the shore, the ice has no normal component of velocity but is allowed to slip freely, relative to the shore, in the tangential direction. For the MPM

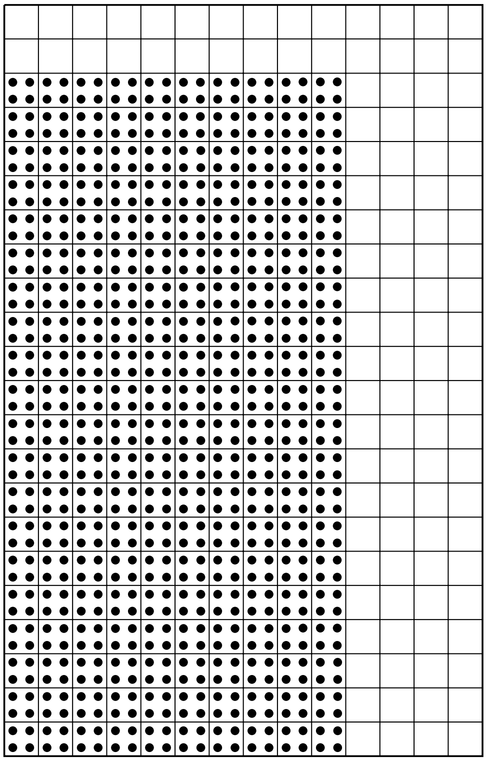


Figure 6. The ice is a 25 km by 50 km region discretized using 4 material points per element. The background grid consists of 2.5 km square elements.

Table 1. Simulation Parameters

Name	Symbol	Value
	Physical parameters	
Ice density	ρ	918 kg m^{-3}
Air density	ρ_a	1.20 kg m^{-3}
Air drag coefficient	C_a	5×10^{-2} kg m^{-1} s^{-2}
Seawater density	ρ_w	1026 kg m^{-3}
Seawater drag coefficient	C_w	5×10^{-4} m s^{-1}
	Initial conditions	
Initial ice thickness	h_0	2 m
Initial ice compactness	A_0	0.9
	Viscous-plastic parameters	
Ice strength parameter	P^*	5×10^3 kg m^{-1} s^{-2}
Ice strength-compactness parameter	C	15
Eccentricity of ellipse	e	2
Viscosity cutoff	$\zeta_{\max}$	2.5×10^8 s
	Decohesion parameters (set 1)	
Young's modulus	E	1 MPa
Poisson's ratio	ν	0.36
Failure strength in tension	τ_{nf}	15.0 KPa
Failure strength in shear	τ_{sf}	9.0 KPa
Decohesion length scale	u_0	100 m
Shear magnification factor	s_m	4
Compressive strength	f'_c	75 KPa

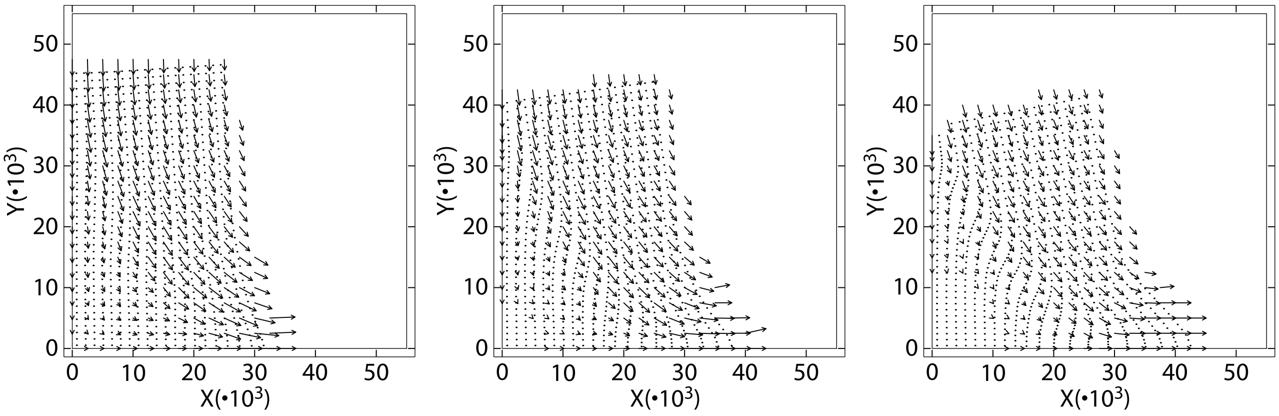
calculation, a 25 km by 50 km region of ice is placed so that the left and bottom ice boundaries coincide with the left and bottom background-grid boundaries. The left and bottom grid boundaries are identified with the straight shoreline. The ice is initially at rest and has a thickness of 2 m. A plot of the grid and ice region is shown in Figure 6. The background grid has 2.5 km square elements The ice is discretized using four material points per element. The simulation is run for three days.

[55] To begin, we compare a simulation using MPM and the viscous-plastic rheology to the same simulation in *Schulkes et al.* [1998] using a finite element method. A uniform surface wind stress is applied so that $\mathbf{t}_a = (0, -C_a)$. A linear drag relation is assumed between the base of the ice and underlying water, $\mathbf{t}_w = -\rho_w C_w \mathbf{v}$, with ρ_w representing the water density, and C_w the water drag coefficient. The velocity $\mathbf{v}$ is the ice velocity and the water velocity is assumed to be zero. The initial area fraction is a uniform constant, $A_0 = 0.9$ and the initial ice thickness is $h_0 = 2$ m. These quantities evolve according to equations (35) with zero thermodynamic source terms $S_A = S_h = 0$. Table 1 lists all of the parameters used in the simulation. The time step, Δt, is chosen for numerical stability of an explicit update so that, for all elements, $\Delta t < m\Omega^e/(2\eta)$, where Ω^e is the area of the computational element. The critical time step is about 0.01 s.

[56] Figure 7 shows the computed mesh velocity in one-day intervals. The dots in these plots represent the current material-point positions. Notice that the material points represent the geometry and a sharp edge can be maintained even when computations are performed on a square background mesh. As expected, the ice is driven towards the bottom solid boundary and flows out, away from the shoreline along this boundary. The ice deforms substantially over three days. The velocity and shape of the ice are quite similar to the velocity and deformed shapes shown in *Schulkes et al.* [1998], although it is impossible to make a quantitative comparison because coordinates are not marked, and the velocity scale is not given in the reference. The deformation on the third day differs slightly, but detectably, from the published results in *Schulkes et al.* [1998]. The MPM profile on day three is a bit less sloped at the top and right boundaries, and is thus a bit wider at the top. The maximum velocity is about 0.11 m and occurs at the lower right corner.

[57] Figure 8a shows the compactness of the ice after the first day. The ice is slightly more compact than the initial value of 0.9 along the bottom boundary. The largest changes in compactness occur along the upper left boundary, and the bottom right boundary. There is about a 20% reduction from 0.9 to 0.7 in these regions. The upper left boundary is also a region of divergence (Figure 8b). However, the predominant deformation pattern is a region of shear shown in Figure 8c that separates a low velocity region at the lower left corner from a high velocity region above and to the right. These fields also agree with data presented by *Schulkes et al.* [1998].

[58] The next simulation changes the rheology from viscous-plastic to elastic-decohesvie. In line with *Schreyer et al.* [2006], we use the first set of material parameters



(a)

(b)

(c)

Figure 7. The mesh velocity and material-point positions shown after (a) 1 day, (b) 2 days, and (c) 3 days using the viscous-plastic rheology.

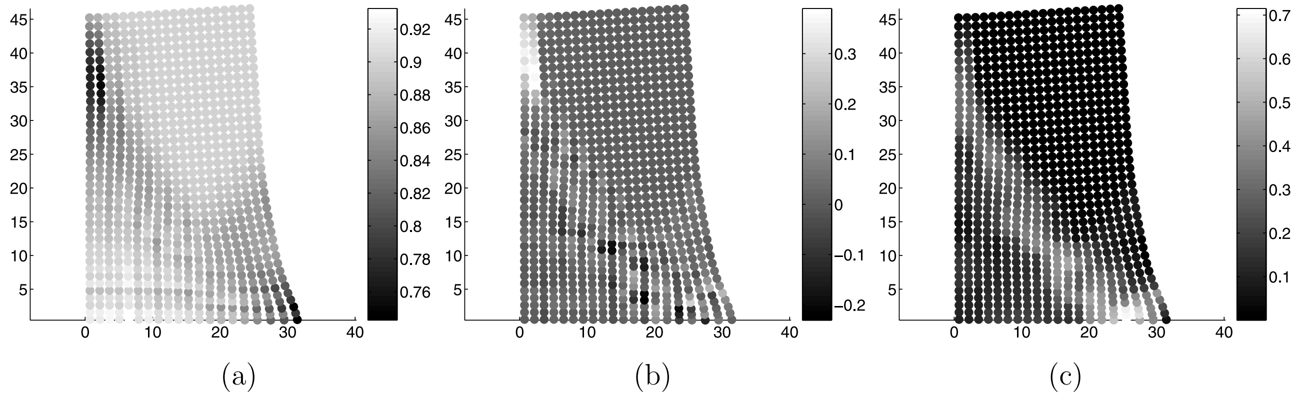


Figure 8. Material-point values of (a) compactness, A, (b) divergence, $\nabla \cdot \mathbf{v}$, and (c) shear strain rate invariant, γ, after one day, using the viscous-plastic rheology; $\gamma^2 = \frac{1}{2}\,\mathrm{tr}(\hat{\boldsymbol{d}} : \hat{\boldsymbol{d}})$, $\hat{\boldsymbol{d}} = \frac{1}{2}(\nabla\mathbf{v} + (\nabla\mathbf{v})^T) - \frac{1}{2}(\nabla \cdot \mathbf{v})\boldsymbol{I}$. The divergence and γ are measured in units of 10^{-5} s^{-1}.

given in Table 1 for intact ice. The isotropic elastic properties are characterized by the Young's modulus and Poisson's ratio, and the remaining parameters govern the decohesion. The time stepping is explicit, with the time step controlled by the CFL condition, $\Delta t < \sqrt{\Omega^e}/c$, based on the elastic wave speed, $c \sim \sqrt{E/\rho}$, and the background element area, Ω^e. For this problem, the time step is about $\Delta t = 30$ s. It is worth noting that this time step is about four orders of magnitude larger than the time step required for the stable, explicit viscous-plastic simulation with the same mesh size.

[59] In this calculation the ice did not deform significantly over the three days. The maximum stress in the ice is not sufficient to initiate decohesion and the ice response to forcing is purely elastic. After a brief transient period, the motion is quasistatic and the momentum balance is primarily a balance of the wind stress and the internal forces. An approximate solution is obtained by assuming the ice velocity is zero and the only nonzero Cauchy stress component is the yy-component, σ_{yy}. In order to balance the wind stress, σ_{yy} is linear in y, zero at the top of the ice and −1250 Pa at the bottom. The two-dimensional simulation with slip boundary conditions shows this general behavior with an approximately linear stress profile along the y-direction, as shown in Figure 9.

[60] The calculation with the viscous-plastic constitutive model shows more flow than the elastic model. The uniaxial

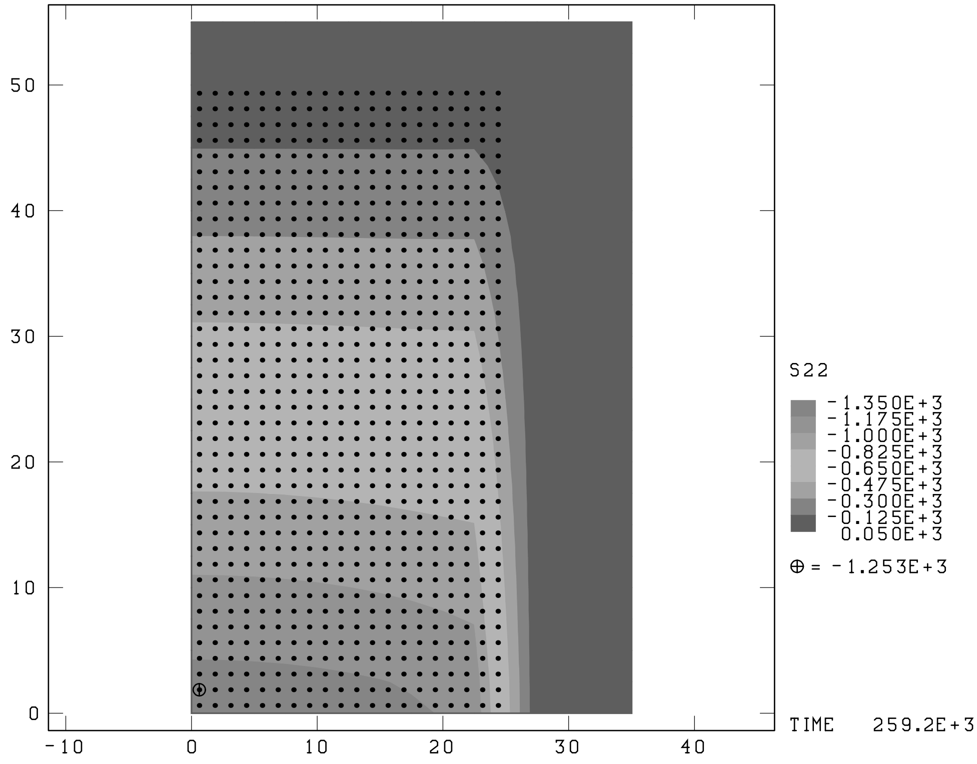


Figure 9. Element values of the yy-component of the extra Cauchy stress using parameter set 1 from Table 1 in the elastic-decohesive rheology.

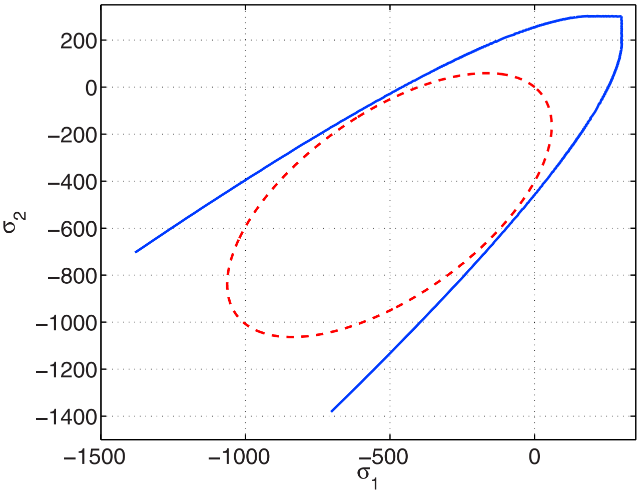


Figure 10. The dashed line is a viscous-plastic yield surface and the solid line is a decohesive failure surface. The axes show the values of principal stresses.

compressive strength, per unit thickness, based on this elliptical yield curve in the viscous-plastic model is $2P^* A \exp(-C(1-A))/(1+e^2)$. For the parameters in Table 1, the uniaxial compressive strength is about 400 Pa, which is much smaller than the 75 KPa used in the elastic-decohesive model. In the next MPM calculation, the parameters defining the failure function are reduced to more closely match the viscous-plastic yield surface to the decohesion surface. Specifically, τ_{nf} is reduced to 300 Pa, τ_{sf} = 180 Pa, and f'_c = 1500 Pa. The other parameters are unchanged. Figure 10 compares the decohesive failure function and the yield function associated with the viscous-plastic model plotted in principal stress space.

[61] To better resolve the large deformation, the background grid is reduced to a 1.25 km square mesh. The calculation is explicit in time, with the time step determined by the CFL condition, as before. Due to the finer mesh, the time step is about Δt = 15 s. Figures 11a–11c show the velocity vectors after one, two and three days, respectively, for this calculation. The maximum velocity is about 0.14 m and occurs at the lower right. At the bottom boundary there is considerable deformation of the ice as the free surface moves out to the right. The velocity is smaller in the lower left corner of the domain than elsewhere. The region of lower velocity is separated from the main flow by a zone of high shear. These observations are qualitatively similar to the results using the viscous-plastic rheology; however, details of the shape are different. The decohesive model shows more of a dip in the top surface and more of a bulge on the right, compared with the viscous-plastic model. Figures 12a and 12b show plots of the normal and tangential components of the displacement jump, respectively. These components are scaled by u_0 = 100 m. Thus the largest normal opening is about 600 m and occurs near the left side of the domain. There is also a similarly large opening along a crack roughly normal to the right side of the ice cover. The tangential component of the displacement jump has a larger maximum magnitude, almost 2 km. It is apparent that the shear zone seen in the velocity field corresponds to a lead where there is a large tangential displacement between the sides of the lead. Thus two main leads appear in the simulation. The first lead opens and is accompanied by relatively large shear, and the second lead mainly opens with relatively little shear. The opening of the second lead does not seem to correspond to any feature of the viscous-plastic simulation. The combination of fractures indicates that a block of ice at the lower right breaks away, and the region just above this block falls in a shear motion relative to the comparatively stationary ice in the lower left corner of the domain.

6.3. Discussion

[62] The first calculation where the ice cover remains elastic provides a simple test of the numerical method since an approximate analytical solution can be used for comparison. The values of τ_{nf}, τ_{sf} and f'_c used in this first calculation did not result in significant deformation under the prescribed wind stress. Experiments and kinematic studies [*Schreyer et al.*, 2006] indicate that the parameters

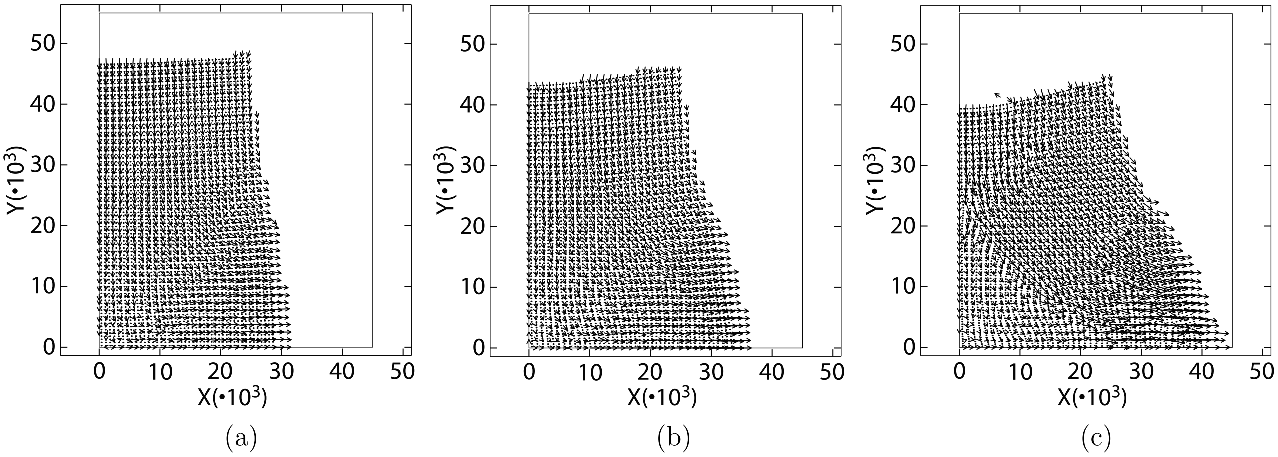


Figure 11. Ice velocity after (a) 1 day, (b) 2 days, and (c) 3 days plotted on the background grid, using the reduced strength parameters in the elastic-decohesive rheology.

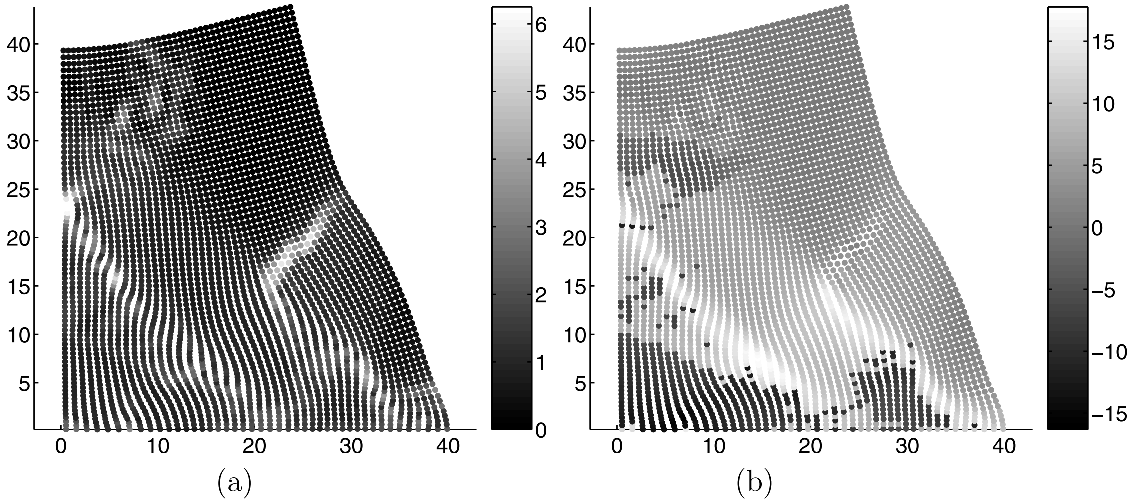


Figure 12. Material-point values of the (a) normal and (b) tangential components of the displacement jump after 3 days using the reduced strength parameters in the elastic-decohesive rheology.

used in this simulation are characteristic of an intact ice cover. In contrast, there is significantly more deformation seen in computations with a viscous-plastic model shown in *Schulkes et al.* [1998]. In those simulations, the nominal strength P^* is low, and the initial compactness of 0.9 implies that the ice cover is not intact, but contains 10% open water. The presence of water reduces the strength of the intact ice cover. Note that this reduction occurs uniformly and isotropically throughout the domain.

[63] The second calculation, where the decohesion parameters are reduced, is an isotropic approximation to the case where the ice has multiple leads throughout that reduce its strength. This is analogous to having a low strength and a compactness less than one for the viscous-plastic model. Under these conditions the extent of the deformation using the elastic-decohesive model is similar to that observed using a viscous-plastic model. Using the elastic-decohesive model, the deformation is seen to result from the formation of two principal leads. One lead opens and shears while the failure mode for the second lead is primarily an opening mode. This simulation provides an example of the information that is available through the decohesion model. It should be noted that once decohesion initiates the ice is no longer an isotropic material. The traction on the surface of a lead goes to zero as it opens, but components of stress orthogonal to the lead can be nonzero. Although not done in this paper, the area associated with the open lead can be calculated. Future studies can add thermodynamic effects that would allow the exposed water to freeze and grow new ice. Standard methods that track ice thickness distributions can also be added to give a more complete model of the ice cover.

[64] Instead of uniformly reducing the strength of the ice cover when water is present, it is possible, using the elastic-decohesive model, to initialize water concentrated in leads that would provide directional weaknesses at various locations. The ice would have material parameters associated with intact ice, as used in the first calculation. Using this approach, the weakness in the ice would no longer be isotropic and the response to wind loading would depend on the orientation of the preexisting leads. One can imagine two simple cases. In the first case, preexisting leads are oriented parallel to the wind velocity, and in the second case preexisting leads are oriented perpendicular to the wind velocity. In the first case, the parallel leads would not evolve under the loading and the ice would have a purely elastic response as in the first calculation. There would be rather little deformation. In the second case, the loading would initially result in the leads closing, but once the leads closed an elastic response similar to that shown in the first calculation would occur and any significant deformation of the ice would be due to the closing leads.

7. Conclusion

[65] The material-point method has been examined for modeling sea-ice dynamics. The tests in section 6 show that the transport of conserved quantities and material constants can be performed accurately and efficiently using the material points. The representation of ice by a collection of unconnected Lagrangian material points also can handle the large deformations observed in the Arctic ice pack and can predict the location of the ice edge. MPM easily allows the use of any solid (or fluid) model for ice in addition to the traditional viscous-plastic models that are normally used in numerical simulations of ice dynamics. Even models with history dependence, such as elastoplasticity, can be employed. As examples, a viscous-plastic model and an elastic-decohesion model were used in this paper.

[66] The newly developed elastic-decohesion constitutive equation is a natural approach for modeling material failure and lead formation. The implementation in MPM through the jump in displacement as an internal variable along with weak compatibility provides a simple, efficient algorithm that does not exhibit pathologies associated with distorted finite elements or artificial anisotropies due to orientation effects on the computational mesh. A simple mechanism for the initiation of a new lead has been designed and tested as part of this work. We also suggest an algorithm to track thickness distributions and ice compactness that conforms

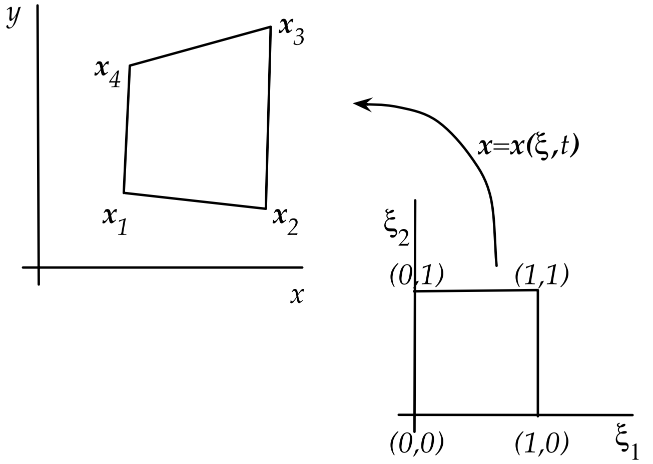


Figure A1. Quadrilateral element with mapping from the master element indicated.

with the information available through the decohesion constitutive model.

[67] As has been noted in the literature, an explicit solution of the viscous-plastic model is too costly for practical basin-scale simulations. In practice, either an implicit method, such as *Zhang and Hibler* [1997], or the elastic-visccous-plastic model [*Hunke*, 2001] is used. In the latter case, two-hour time steps are usual, with 120 subcycles on a 5 km square mesh. Thus the time interval between subcycles is about 1 min. The explicit time step for the elastic-decohesion model is roughly the same as the subcycle time used in the elastic-viscous-plastic model for the same mesh size. Thus the cost of the elastic-decohesion model is comparable to the elastic-viscous-plastic model. Overall, the elastic-decohesion model shows promise as a method for explicitly representing leads and their effect on ice motion. The solution algorithm in MPM is cost effective and is a plausible alternative to current basin-scale simulation techniques.

Appendix A: Shape Functions

[68] Shape functions are used in the MPM to interpolate between the background mesh and the material points, as well as to construct approximations for the finite element method on the background mesh. The background grid is subdivided into elements, Ω^e, $e = 1, 2, \ldots, N_e$. The nodes of this mesh are $\mathbf{x}_I(t)$, $I = 1, \ldots, N_n$. If each element Ω^e has m nodes then we can refer to the nodes belonging to an individual element with the notation $\mathbf{x}_I^e(t)$, $I = 1, \ldots, m$. For definiteness, consider a mesh made up of quadrilateral elements in two-dimensions with four nodes, $m = 4$.

[69] The shape functions can be constructed through a mapping from a master element. There are two domains under consideration, the master element $\square$ and the element in the current configuration $\Omega^e(t)$. There is a map connecting these two domains, the map from the master element to the current configuration, $\mathbf{x} = \mathbf{x}(\boldsymbol{\xi}, t)$. Figure A1 illustrates these domains and map.

[70] For a four-node, quadrilateral mesh, the master element is a square and the natural coordinates are denoted by $\boldsymbol{\xi} = (\xi_1, \xi_2)$, $0 \leq \xi_1 \leq 1$, $0 \leq \xi_2 \leq 1$. The map between the master element and a finite element is

$$\begin{aligned}\mathbf{x}^h(\boldsymbol{\xi}, t) &= [\mathbf{x}_1^e(1-\xi_1) + \mathbf{x}_2^e\xi_1](1-\xi_2) \\ &\quad + [\mathbf{x}_4^e(1-\xi_1) + \mathbf{x}_3^e\xi_1]\xi_2 \\ &= \mathbf{x}_1^e(1-\xi_1)(1-\xi_2) + \mathbf{x}_2^e\xi_1(1-\xi_2) \\ &\quad + \mathbf{x}_3\xi_1\xi_2 + \mathbf{x}_4^e(1-\xi_1)\xi_2 \\ &= \sum_{I=1}^{4} N_I^e(\boldsymbol{\xi})\mathbf{x}_I^e(t), \qquad \text{on } \Omega^e(t)\end{aligned} \tag{A1}$$

where

$$\begin{aligned} N_1^e(\boldsymbol{\xi}) &= (1-\xi_1)(1-\xi_2) \quad & N_2^e(\boldsymbol{\xi}) &= \xi_1(1-\xi_2) \\ N_3^e(\boldsymbol{\xi}) &= \xi_1\xi_2 & N_4^e(\boldsymbol{\xi}) &= (1-\xi_1)\xi_2 \end{aligned} \tag{A2}$$

These element shape functions have the property that nodes get mapped to nodes and also, edges in the master element get mapped to the corresponding edge in the finite element.

[71] The element shape functions N_I^e can be assembled into a global shape function. For node J the shape functions from the surrounding elements contribute to the global shape function N_J, as illustrated in Figure A2. Specifically, the formula is $N_J = \sum_{e=1}^{N_e} \sum_{I=1}^{4} N_I^e L_{IJ}^e$, where the $4 \times N_n$ matrix L^e is the connectivity matrix. The connectivity matrix has zeros and ones. The IJ entry is one if element node number $I = 1, 2, \ldots, 4$ corresponds to the global node, $J = 1, 2, \ldots, N_n$, and is zero otherwise. In the figure, global node J corresponds to node one of element one, node two of element two, etc. In terms of the global shape functions, the motion is approximated by

$$\mathbf{x}^h(\boldsymbol{\xi}, t) = \sum_{I=1}^{N_n} \mathbf{x}_I(t) N_I(\boldsymbol{\xi}). \tag{A3}$$

The map $\mathbf{x}^h(\boldsymbol{\xi}, t)$ is one-to-one and continuous in the spatial variable.

[72] The integrals (12)–(15) are over the current configuration $\Omega(t)$ but the shape function is defined in (A1) as a function of the master element coordinates. We must

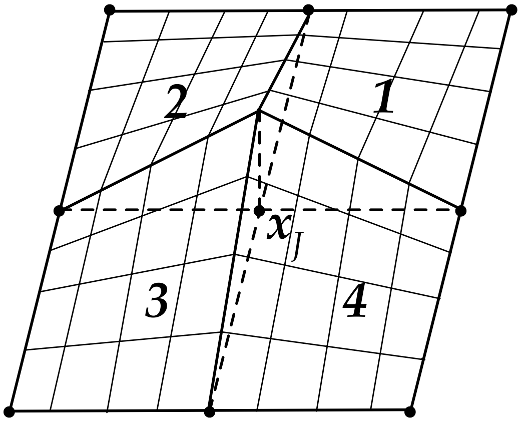


Figure A2. The global shape function associated with node J is assembled from shape functions defined on the surrounding elements.

view N_I as a function of the current configuration through the composition of maps, $N_I(\mathbf{x}, t) = N_I(\boldsymbol{\xi}(\mathbf{x}, t))$, where $\boldsymbol{\xi}(\mathbf{x}, t)$ is the inverse of (A1). Hence, the term grad $N_I(\mathbf{x})$ is computed using the chain rule, grad $N_I(\mathbf{x})$ = grad$_{\boldsymbol{\xi}}$ $N_I(\boldsymbol{\xi})\mathbf{F}_{\square}^{-1}(\boldsymbol{\xi}, t)$, where $\mathbf{F}_{\square}(\boldsymbol{\xi}, t)$ is the deformation gradient for the map between the master element and the current configuration, $\mathbf{F}_{\square}(\boldsymbol{\xi}, t) = \partial\mathbf{x}/\partial\boldsymbol{\xi}$. The components of this deformation gradient are easy to compute over an element from (A1), and then the 2 × 2 matrix is easily inverted to obtain $\mathbf{F}_{\square}^{-1}(\boldsymbol{\xi}, t)$.

[73] Over a time step on a Lagrangian grid, the shape function defined on the master element does not change with time. This fact is significant since then a time derivative of the shape function is not required in the formulas for velocity and acceleration (8)–(9). Also, the natural coordinates for a material point in an element remain constant over the Lagrangian time step. The natural coordinates of a material point, $\boldsymbol{\xi}_p = \boldsymbol{\xi}(\mathbf{x}_p, t)$, are determined by inverting (A1) at the beginning of a time step. These coordinates are then used throughout the Lagrangian step to evaluate the shape function $N_I(\mathbf{x}_p) = N_I(\xi(\mathbf{x}_p, t)) = N_I(\boldsymbol{\xi}_p)$.

[74] **Acknowledgments.** This material is based upon work partially supported by the National Science Foundation (under grant DMS-0222253), by the Minerals Management Service and the National Aeronautics and Space Administration (under contract NNH04 CC 45C) and the Office of Naval Research (under contract N000-14-04-M-0005).

References

Anderson, D. L. (1961), Growth rate of sea ice, *J. Glaciol.*, *3*, 1170–1172.

Belytschko, T., W. K. Liu, and B. Moran (2000), *Nonlinear Finite Elements for Continua and Structures*, John Wiley, Hoboken, N. J.

Bitz, C., and W. H. Lipscomb (1999), An energy-conserving thermodynamic model of sea ice, *J. Geophys. Res.*, *104*(C7), 15,669–15,677.

Brackbill, J. U., and H. M. Ruppel (1986), Flip: A method for adaptively zoned, particle-in-cell calculations in two dimensions, *J. Comput. Phys.*, *65*, 314.

Brackbill, J. U., D. B. Kothe, and H. M. Ruppel (1988), Flip: A low-dissipation particle-in-cell method for fluid flow, *Comput. Phys. Commun.*, *48*, 25–38.

Burgess, D., D. Sulsky, and J. U. Brackbill (1992), Mass matrix formulation of the FLIP particle-in-cell method, *J. Comput. Phys.*, *103*(1), 1–15.

Coon, M. (1980), A review of AIDJEX modeling, in *Sea Ice Processes and Models, Proceedings of the Arctic Ice Dynamics Joint Experiment*, edited by R. S. Pritchard, pp. 12–27, Univ. of Wash. Press, Seattle.

Coon, M., G. Maykut, R. Pritchard, D. Rothrock, and A. Thorndike (1974), Modeling the pack ice as an elastic-plastic material, in *AIDJEX Bulletin*, vol. 24, pp. 1–105, Univ. of Wash. Press, Seattle.

Coon, M. D., G. S. Knoke, D. C. Echert, and R. S. Pritchard (1998), An oriented thickness distribution for sea ice, paper presented at 17th Int. Symp. on Ice, Int. Assoc. of Hydraul. Eng., Saint Petersburg, Russia.

Evans, M., and F. Harlow (1957), The particle-in-cell method for hydrodynamic calculations, *Tech. Rep. LA-2139*, Los Alamos Sci. Lab., Los Alamos, N. M.

Flato, G. M. (1993), A particle-in-cell sea-ice model, *Atmos. Ocean*, *31*(3), 339–358.

Flato, G. M., and W. D. Hibler (1989), The effect of ice pressure on marginal ice-zone dynamics, *IEEE Trans. Geosci. Remote Sens.*, *27*(5), 514–521.

Gray, J., and L. Morland (1994), A two-dimensional model for the dynamics of sea ice, *Philos. Trans. R. Soc. London A*, *347*, 219–290.

Gutfraind, R., and S. B. Savage (1997), Smoothed particle hydrodynamics for the simulation of broken-ice fields: Mohr-Coulomb-type rheology and frictional boundary conditions, *J. Comput. Phys.*, *134*, 203–215.

Harlow, F. (1957), Hydrodyanamics problems involving large fluid distortions, *J. Assoc. Compos. Mach.*, *4*, 137.

Hibler, W. D. (1977a), Viscous sea ice law as a stochastic average of plasticity, *J. Res. Oceans Atmos.*, *82*(27), 3932–3938.

Hibler, W. D. (1977b), Viscous sea ice law as a stochastic average of plasticity, *Eos Trans. AGU*, *58*(8), 823.

Hibler, W. D. (1979), Dynamic thermodynamic sea ice model, *J. Phys. Oceanogr.*, *9*(4), 815–846.

Hibler, W. D., and E. M. Schulson (2000), On modeling the anisotropic failure and flow of flawed sea ice, *J. Geophys. Res.*, *105*(C7), 17,105–17,120.

Hopkins, M. A. (1998), Four stages of pressure ridging, *J. Geophys. Res.*, *103*(C10), 21,883–21,891.

Hopkins, M. A., and W. D. Hibler (1991), On the shear-strength of geophysical scale ice rubble, *Cold Reg. Sci. Technol.*, *19*(2), 201–212.

Hopkins, M. A., W. D. Hibler, and G. M. Flato (1991), On the numerical-simulation of the sea ice ridging process, *J. Geophys. Res.*, *96*(C3), 4809–4820.

Hughes, T. J. R. (2000), *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*, Dover, Mineola, N. Y.

Hunke, E. C. (2001), Viscous-plastic sea ice dynamics with the evp model: Linearization issues, *J. Comput. Phys.*, *170*(1), 18–38.

Hunke, E. C., and J. K. Dukowicz (1997), An elastic-viscous-plastic model for sea ice dynamics, *J. Phys. Oceanogr.*, *27*(9), 1849–1867.

Ip, C., W. Hibler, and G. Flato (1991), On the effect of rheology on seasonal sea ice simulations, *Ann. Glaciol.*, *15*, 17–25.

Kubat, I., M. Sayed, S. Savage, and T. Carrieres (2005), Implementation and testing of a thickness redistribution model for operational ice forcasting, paper presented at 18th Int. Conf. on Port and Ocean Eng. Under Arctic Conditions, Clarkson, Univ., Potsdam, N. Y.

Lebedev, V. V. (1940), New formulas on the growth of ice in Arctic rivers and seas, *Meteorol. Gidrolog.*, *6*(8), 40–51.

LeVeque, R. J. (1990), *Numerical Methods for Conservation Laws*, Springer, New York.

Lindsay, R. W., and H. L. Stern (2004), A new Lagrangian model of Arctic sea ice, *J. Phys. Oceanogr.*, *34*.

Lipscomb, W. H., and E. C. Hunke (2004), Modeling sea ice transport using incremental remapping, *Mon. Weather Rev.*, *132*(6), 1341–1354.

Love, E., and D. L. Sulsky (2006), An unconditionally stable, energy-momentum consistent implementation of the material-point method, *Comput. Methods Appl. Mech. Eng.*, *195*, 3903–3925.

Maslowski, W., D. Marble, W. Walczowski, and A. Semtner (2001), On large-scale shifts in the Arctic Ocean and sea-ice conditions during 1979–98, *Ann. Glaciol.*, *33*, 545–550.

Morland, L., and R. Staroszczyk (1998), A material coordinate treatment of sea-ice dynamics, *Proc. R. Soc. London*, *424*, 2819–2857.

Pritchard, R. (1975), An elastic-plastic constitutive law for sea ice, *J. Appl. Mech.*, 379–384.

Roache, P. J. (1985), *Computational Fluid Dynamics*, Hermosa, Albuquerque, N. M.

Rothrock, D. A. (1975), The energetics of the plastic deformation of pack ice by ridging, *J. Geophys. Res.*, *80*, 4514–4519.

Savage, S. (2002), Two category sea-ice thickness redistribution model, technical report, Can. Ice Serv., Ottawa, Ont., Canada.

Sayed, M., and T. Carrieres (1999), Overview of a new operational ice model, paper presented at Ninth Int. Offshore and Polar Eng. Conf. (ISOPE), Int. Soc. of Offshore and Polar Eng., Brest, France.

Sayed, M., T. Carrieres, H. Tran, and S. B. Savage (2002), Development of an operational ice dynamics model for the Canadian Ice Service, paper presented at Twelfth Int. Offshore and Polar Eng. Conf. (ISOPE), Int. Soc. of Offshore and Polar Eng., Kitakyushu, Japan.

Schreyer, H., L. Monday, D. Sulsky, M. Coon, and R. Kwok (2006), Elastic-decohesive constitutive model for sea ice, *J. Geophys. Res.*, *111*, C11S26, doi:10.1029/2005JC003334.

Schulkes, R., L. Morland, and R. Staroszczyk (1998), A finite-element treatment of sea ice dynamics for different ice rheologies, *Int. J. Numer. Methods Geomech.*, *22*, 153–174.

Schulson, E. M. (2001), Brittle failure of ice, *Eng. Fracture Mech.*, *68*, 1839–1887.

Sulsky, D., and H. L. Schreyer (1996), Axisymmetric form of the material point method with applications to upsetting and Taylor impact problems, *Comput. Methods Appl. Mech. Eng.*, *139*, 409–429.

Sulsky, D., Z. Chen, and H. L. Schreyer (1994), A particle method for history-dependent materials, *Comput. Methods Appl. Mech. Eng.*, *118*, 179–196.

Sulsky, D., S.-J. Zhou, and H. L. Schreyer (1995), Application of a particle-in-cell method to solid mechanics, *Comput. Phys. Commun.*, *87*, 236–252.

Thorndike, A. S., and G. A. Maykut (1973), On the thickness distribution of sea ice, *AIDJEX Bull.*, *21*, 31–48.

Thorndike, A., D. Rothrock, G. Maykut, and R. Colony (1975), The thickness distribution of sea ice, *J. Geophys. Res.*, *80*(33), 4501–4513.

Wang, L., and M. Ikeda (2004), A Lagrangian description of sea ice dynamics using the finite element method, *Ocean Modell.*, *7*, 21–38.

Wilchensky, A., and D. Feltham (2004), A continuum anisotropic model of sea-ice dynamics, *Proc. R. Soc. London A*, *460*, 2105–2140.

Zhang, J. L., and W. D. Hibler (1997), On an efficient numerical method for modeling sea ice dynamics, *J. Geophys. Res.*, *102*(C4), 8691–8702.

Zhang, Z. J., and S. B. Savage (1998), Particle-in-cell and finite difference approaches for the study of marginal ice zone problems, *Cold Reg. Sci. Technol.*, *28*, 1–28.

M. Coon, Northwest Research Associates, Seattle, WA 98009, USA. (max@nwra.com)

R. Kwok, Jet Propulsion Laboratory, Pasadena, CA 91109, USA. (ron.kwok@nasa.jpl.gov)

K. Peterson and D. Sulsky, Department of Mathematics and Statistics, University of New Mexico, Albuquerque, NM 87131, USA. (kpeterso@math.unm.edu; sulsky@math.unm.edu)

H. Schreyer, Department of Engineering, University of New Mexico, Albuquerque, NM 87131, USA. (schreyer@me.unm.edu)