AN EFFICIENT HIGH ORDER METHOD FOR DISLOCATION CLIMB IN TWO DIMENSIONS

SHIDONG JIANG *, MANAS RACHH †, AND YANG XIANG ‡

Abstract. We present an efficient high order method for dislocation dynamics simulation of vacancy-assisted dislocation climb in two dimensions. The method is based on a second kind integral equation (SKIE) formulation that represents the vacancy concentration via the sum of double layer potentials and point sources located at each dislocation center, where the climb velocity of each dislocation (or the strength of each point source) is proportional to the integral of the unknown density on the boundary of each dislocation. The method discretizes the interfaces only. Furthermore, it avoids the need of introducing additional unknowns as compared with the formulation in [11]; and unlike the formulation in [31] in which the kernel has logarithmic singularity, the boundary integrals involved in the formulation are easily discretized via the trapezoidal rule with spectral accuracy. Thus, the number of unknowns in the linear system to achieve certain accuracy is optimal for typical settings in dislocation dynamics. We compare three different ways of solving the resulting linear system and demonstrate via numerical examples that the fast direct solver (FDS) in [20, 22] performs best for dislocation arrays, while the fast multipole method (FMM) accelerated iterative solver on the low accuracy FDS preconditioned system performs well for the general setting and exhibits roughly optimal complexity.

Key words. Dislocation climb, dislocation dynamics, second kind integral equation, fast multipole method, fast direct solver.

AMS subject classifications. 31A10, 35Q74, 45B05, 65N25, 65N99

1. Introduction. Dislocations are line defects in crystals and primary carriers of the plastic deformation [18]. Dislocation climb is a nonconservative motion in which dislocations move out of their slip planes with the assistance of diffusion and emission and/or absorption of vacancies. Dislocation climb plays important roles in the plastic deformation at high temperature. The vacancy assisted dislocation climb is a multiscale problem. The vacancies diffuse in the whole macroscopic domain, while the climb velocity of the dislocation lines with atomic-length size core is determined by the emission and/or absorption of vacancies near the dislocation core. It is challenging to incorporate dislocation climb accurately and efficiently in large scale dislocation dynamics simulations.

The model for the vacancy assisted dislocation climb, under the equilibrium assumption for vacancy diffusion, can be written as [15, 18]

\[
\begin{align*}
D_v \Delta c &= b v_{cl} \delta(\Gamma), \\
n_{\Gamma} &= c_0 \exp \left( \frac{f_{cl} \Omega}{k_B T} \right), \\
\end{align*}
\]

where \( c \) is the vacancy concentration, \( v_{cl} \) is the dislocation climb velocity, \( f_{cl} \) is the climb component of the Peach-Koehler force, \( \Gamma \) refers to the boundaries of all dislocations in the entire system, \( \delta(\Gamma) \) is the Dirac delta function of \( \Gamma \), and \( D_v \) is the vacancy diffusivity.
diffusion coefficient, \( r \) is the position of the dislocation, \( r_d \) is the dislocation core radius, \( c_0 \) is a constant reference vacancy concentration, \( c_\infty \) is the vacancy concentration at some outer boundary of radius \( r_\infty \), \( \Omega \) is the atomic volume, \( k_B \) is the Boltzmann constant, and \( T \) is the temperature. Here the dislocations are all edge which is the case considered in this paper.

Recently, Gu et al. [15] derived a dislocation climb formulation for the dynamics of multiple dislocations with general geometries in three dimensions, which accounts for the long-range interaction of dislocations through the vacancy diffusion that contribute to climb. The formulation is based on the Green’s function representation of the solution of the Poisson’ equation and gives accurate solutions for the classical vacancy assisted dislocation climb model. In this formulation, the dislocation climb velocity is determined by solving a boundary integral equation for the vacancy concentration given the climb component of the Peach-Koehler force. In earlier works in the literature, the climb velocity was determined from the local climb Peach-Koehler force by some mobility law similar to that for determining the glide velocity in the study of dislocation dynamics [1, 18, 23, 30, 34, 35]. However, it has been shown [15] such mobility laws for the climb velocity are not generally applicable except for a small set of special cases. A superposition method for boundary value problems in two dimensions was developed by Ayas et al. [2] in which the boundary value problem of vacancy diffusion was solved by a superposition of the fields associated with dislocation climb in an infinite medium and a complementary solution that enforces the boundary conditions. They proposed to solve the complementary vacancy diffusion problem by finite difference or finite element methods over the whole domain.

We consider the case when all the dislocations are straight and parallel, say in the \( z \) direction, and their motion is always uniform in the \( z \) direction. In this setup, the dislocation dynamics problem is reduced to a two dimensional problem in which dislocations are points in the \((x, y)\) plane. We consider edge dislocations with same Burgers vector \( \mathbf{b} = (b, 0, 0) \). The direction of each dislocation is either \( \mathbf{\tau} = (0, 0, 1) \) or \( \mathbf{\tau} = (0, 0, -1) \), and we call the former positive and the latter negative. Each dislocation has a core region which is a small disk centered at the dislocation with radius \( r_d \), and the on the boundary of the dislocation core the Dirichlet boundary condition of the vacancy concentration in (1.1) is satisfied.

In this two dimensional case, the three-dimensional Green’s function formulation for the dislocation climb velocity in [15] is reduced, after integrating along the \( z \) direction, into the following integral equations. Consider the two dimensional problem where the dislocations are located at \((x_i, y_i), i = 1, 2, \cdots, N\). For the \( i \)-th dislocation located at \((x_i, y_i)\), the following equation holds:

\[
\sum_{j=1}^{N} b G_2(x_d, y_d; x_j, y_j) v_{cl}^{(j)} + c_\infty = c_0 e^{-f_{cl}^{(i)} \Omega k_B T}, \tag{1.2}
\]

where \( f_{cl}^{(i)} \) denotes the climb force on the \( i \)-th dislocation located at \((x_i, y_i)\), \( v_{cl}^{(i)} \) denotes the corresponding unknown climb velocity, \((x_d, y_d)\) is a point located at a distance \( r_d \) from \((x_i, y_i)\), and \( G_2(x, y; \xi, \eta) \) is the Green’s function of the diffusion equilibrium equation in (1.1) in two dimensions modified by some constant involving \( r_\infty \):

\[
G_2(x, y; \xi, \eta) = \frac{1}{2\pi D_v} \log \frac{\sqrt{(x - \xi)^2 + (y - \eta)^2}}{r_\infty}. \tag{1.3}
\]
Note here $r_\infty$ has been used as the outer cutoff distance for the integrals over the whole $z$ axis in the derivation of (1.2) from the three dimensional formulation.

The climb velocity $v_{cl}$ and the climb force $f_{cl}$ are defined in the direction of $\tau \times b$ (i.e. the climb direction) [15]. That means the climb direction is along the $+y$ direction for a positive dislocation, and $-y$ direction for a negative dislocation. The Peach-Koehler force is given by [18]

$$f_{PK} = (\sigma \cdot b) \times \tau,$$

where $\sigma$ is the stress tensor. Since $b$ is along $x$-axis and $\tau$ is along $z$-axis, the climb force $f_{cl}$ (in the climb direction defined above) is then given by

$$f_{cl}^{(i)} = -b\sigma_{11}(x_i, y_i).$$

The stress component $\sigma_{11}$ at the location of dislocation $(x_i, y_i)$ is [18]

$$\sigma_{11}(x_i, y_i) = -\sum_{j \neq i} \text{sgn}(j) \cdot \frac{\mu b}{2\pi(1-\nu)} \frac{(y_i - y_j)[3(x_i - x_j)^2 + (y_i - y_j)^2]}{[(x_i - x_j)^2 + (y_i - y_j)^2]^2},$$

where $\mu$ is the shear modulus, $\nu$ is the Poisson ratio, and $\text{sgn}(j) = 1$ if the $j$-th dislocation is positive and $\text{sgn}(j) = -1$ if it is negative.

We observe that (1.2) represents the vacancy concentration via a sum of point sources centered at each dislocation whose strength is proportional to the climb velocity. The formulation works well for systems of small size in the low accuracy regime. However, it is known that (1.2) becomes increasingly ill-conditioned when there are a large number of dislocations in the system. The accuracy of the formulation is also quite low since it sets up the linear system by putting only one point on the boundary of each dislocation, which may be insufficient even when the dislocations are well separated from each other. Moreover, unlike the three-dimensional Green’s function formulation in [15], the solution of (1.2) only satisfies the condition at the outer boundary in (1.1) in an approximated way which holds when all the dislocations are away from the outer boundary. In the superposition method [2], the complementary boundary value problem of vacancy diffusion was solved over the whole domain by finite difference method. This limits the method only to domains of small size; and the authors of [2] tried to partially overcome this difficulty at the cost of accuracy by choosing the dislocation core size $5-10$ times larger than the actual size.

In this paper, we present a second kind integral equation formulation for the dislocation climb problem (1.1). The formulation represents the dislocation vacancy concentration via a sum of double layer potentials and point sources located at each dislocation center, where the strength of each point source which is also the climb velocity of that dislocation is proportional to the integral of the unknown density on the boundary of each dislocation. When the average spacing between dislocations is no less than ten times larger than the size of the dislocation cores, which is almost always the case in dislocation dynamics (otherwise special treatments will be invoked [1]), our formulation requires nearly optimal number of unknowns to achieve the desired accuracy due to two reasons: a) the kernel in our representation is smooth and trapezoidal rule is spectrally accurate for smooth functions, and b) no additional unknowns are required in our formulation. For typical geometries in the dislocation dynamics, 4 discretization points are sufficient for 4–5 digit accuracy in computing the climb velocity using our formulation. Our formulation applies directly to boundary value
problems of vacancy-assisted dislocation climb over any domain in two dimensions, not limited to the circular disk specified in (1.1). The formulation is similar to the unconstrained integral formulations for the Stokes flow in [13]. There have been several other SKIE formulations for this type of mathematical problems. More specifically, in [11], the solution to the Laplace equation is represented via the sum of double layer potentials and unknown point sources, which introduces an extra unknown for each dislocation. In [31], the solution is represented via a linear combination of single and double layer potentials, whose kernel has logarithmic singularity as in [15].

In the past, the resulting linear system is often solved with an iterative solver such as GMRES coupled with an FMM for accelerating the matrix-vector product. Recently, fast direct solvers (see, for example, [3, 5, 6, 8, 12, 20, 22, 24, 26, 27, 28, 33]) which construct an efficient factorization for applying the matrix inverse have evolved rapidly over the last decade to solve various ill-conditioned boundary value problems. Even though the number of iterations in GMRES is independent of the number of unknowns for an SKIE formulation for a fixed geometry (see, for example, [25]), a greater concern for large-scale dislocation dynamics simulation is how the number of iterations depends on the number of dislocations since the number of discretization points per dislocation is very small. Our numerical experiments indicate that the number of iterations grows like $O(N^\alpha)$ with $\alpha \in (0, 1/2]$ with $N$ the number of dislocations in the system, independent of the configuration of the dislocations in the plane. Thus, even though the FMM has $O(N)$ complexity, the iterative solver on the unpreconditioned system exhibits suboptimal $O(N^{1+\alpha})$ complexity. On the other hand, the performance of the fast direct solver depends very sensitively on the configuration of the dislocations. Indeed, when the dislocations are located along a 1D curve such as a dislocation array along $y$-axis, fast direct solvers scale linearly and are much faster than the iterative solvers. However, when the dislocations are located in the whole plane on a uniform lattice with small random perturbations, fast direct solvers slow down dramatically due to the increase in the interaction rank of off-diagonal blocks. We have implemented and studied three algorithms: (a) FMM+GMRES iterative solver on the original SKIE system; (b) FDS on the original SKIE system; (c) FMM+GMRES on the preconditioned system with low accuracy $A^{-1}$ obtained by FDS as the preconditioner. Our numerical experiments show that the last algorithm performs best in the general setting. It reduces the number of iterations to almost constant even when the number of dislocations increases, and exhibits roughly optimal complexity.

The rest of the paper is organized as follows. In Section 2, we collect some analytical preliminaries to be used subsequently. In Section 3, we present our SKIE formulation and show that the system does not have any nontrivial nullspace. Section 4 presents the numerical algorithm for solving the system of SKIEs. Section 5 discusses an FMM for computing the climb force. In Section 6, we show several numerical examples to demonstrate the performance (i.e., convergence rate, number of iterations, and timing results, etc.) of the overall scheme and comparisons with existing results. Finally, we conclude our paper with further discussions and extensions to three dimensional problems and time-dependent cases.

2. Preliminaries. The cross section in $\mathbb{R}^2$ for dislocation climb is illustrated in Figure 2.1. The outer boundary at $r_\infty$ is denoted by $\Gamma_0$. The boundary of the core region of the $i$-th dislocation is denoted by $\Gamma_i, i = 1, 2, \ldots, N$ with $N$ the total number of dislocations in the system. We will use boldface quantities such as $\mathbf{r}, \mathbf{s}$ to denote points in $\mathbb{R}^2$. The outward unit normal vector at a point $\mathbf{r}$ on the boundary
is denoted by $\nu_r$.

![Diagram showing the computational domain D, the outer boundary at $r_\infty$ denoted by $\Gamma_0$, and the boundaries of dislocations denoted by $\Gamma_1, \Gamma_2, \ldots, \Gamma_N$. The unit normal $\nu$ points out of $D$ on each component curve.]

The Green’s function for the Laplace equation $\Delta c = 0$ in two dimensions is given by the formula

$$G(r, r') = -\frac{1}{2\pi} \log |r - r'|.$$  \hspace{1cm} (2.1)

Suppose that $\gamma$ is a smooth closed oriented curve in $\mathbb{R}^2$. Given a function $\rho$ in $L^2(\gamma)$ (i.e., the space of square integrable functions on $\gamma$), we define the double layer potential by the formula

$$D_\gamma[\rho](r) = \int_\gamma \frac{\partial G(r, r')}{\partial \nu_{r'}} \rho(r') \, dS(r'),$$  \hspace{1cm} (2.2)

where the kernel $\frac{\partial G(r, r')}{\partial \nu_{r'}}$ has the following explicit formula

$$\frac{\partial G(r, r')}{\partial \nu_{r'}} = \nabla_{r'} G(r, r') \cdot \nu_{r'} = \frac{1}{2\pi} \frac{(r - r') \cdot \nu_{r'}}{|r - r'|^2}.$$  \hspace{1cm} (2.3)

It is clear that the double layer potential defined in (2.2) is harmonic (i.e., satisfies the Laplace equation) in $\mathbb{R}^2 \setminus \gamma$. Furthermore, let $r_0$ be a point on $\gamma$. Then the double layer potential satisfies the following jump relations (see, for example, [17, 25, 29]):

$$\lim_{r \to r_0^+} D_\gamma[\rho](r) = \pm \frac{1}{2} \rho(r_0) + \int_\gamma \frac{\partial G(r_0, r')}{\partial \nu_{r'}} \rho(r') \, dS(r')$$

$$= \pm \frac{1}{2} \rho(r_0) + D_{PV,\gamma}[\rho](r_0),$$  \hspace{1cm} (2.4)
where \( \oint \) indicates the principal value integral over the curve \( \gamma \), and \( \mathbf{r} \to \mathbf{r}_0 \) implies that \( \mathbf{r} \) approaches \( \mathbf{r}_0 \) nontangentially from the exterior (+) or the interior (−) side, respectively.

It is well known that the kernel of the double layer potential defined in (2.3) is actually smooth when \( \gamma \) is smooth and has the following limiting value as \( \mathbf{r} \to \mathbf{r}' \):

\[
\lim_{\mathbf{r} \to \mathbf{r}'} \frac{\partial G(\mathbf{r}, \mathbf{r}')}{\partial \nu_{\mathbf{r}'}} = -\frac{\kappa(\mathbf{r}')}{{4\pi}},
\]

(2.5)

where \( \kappa(\mathbf{r}') \) is the curvature of \( \gamma \) at \( \mathbf{r}' \). Thus, the principal value integral in (2.4) is just the usual Riemann integral when the density \( \rho \) is, say, continuous. Moreover, since the kernel decays like \( \frac{1}{|\mathbf{r}|} \) as \( |\mathbf{r}| \to \infty \), we have

\[
D_{\gamma}[\rho](\mathbf{r}) \to 0 \quad \text{as} \quad |\mathbf{r}| \to \infty.
\]

(2.6)

Finally, the following lemma will be used in the computation of the climb velocity and the proof of Theorem 3.1.

**Lemma 2.1.** Suppose that \( \tilde{\gamma} \) is a simple smooth closed curve that lies either completely in the interior or the exterior of \( \gamma \) or coincides with \( \gamma \). Then

\[
\int_{\tilde{\gamma}} \left( \frac{\partial}{\partial \nu_{\mathbf{r}}} D_{\gamma}[\rho](\mathbf{r}) \right) d\mathbf{S}(\mathbf{r}) = 0.
\]

(2.7)

**Proof.** We have

\[
\int_{\tilde{\gamma}} \left( \frac{\partial}{\partial \nu_{\mathbf{r}}} D_{\gamma}[\rho](\mathbf{r}) \right) d\mathbf{S}(\mathbf{r}) = \int_{\tilde{\gamma}} \left( \int_{\gamma} \frac{\partial^2 G(\mathbf{r}, \mathbf{r}')}{\partial \nu_{\mathbf{r}} \partial \nu_{\mathbf{r}'}} \rho(\mathbf{r}') dS(\mathbf{r}') \right) d\mathbf{S}(\mathbf{r})
\]

\[
= \int_{\gamma} \rho(\mathbf{r}') \frac{\partial}{\partial \nu_{\mathbf{r}'}} \left( \int_{\tilde{\gamma}} \frac{\partial G(\mathbf{r}, \mathbf{r}')}{\partial \nu_{\mathbf{r}}} d\mathbf{S}(\mathbf{r}) \right) d\mathbf{S}(\mathbf{r}')
\]

\[
= 0.
\]

(2.8)

Here the second equality follows from interchange of the order of integration and differentiation; the third equality follows from Gauss’ Lemma (see, for example, [25]).

---

**3. Second Kind Integral Equation Formulation for Dislocation Climb.**

We first note that (1.1) is equivalent to the following standard Dirichlet boundary value problem for the Laplace equation in two dimensions:

\[
\Delta c = 0, \quad \text{in} \ D,
\]

\[
c(\mathbf{r}) = g_i = c_0 e^{-\frac{\xi^2}{b^2}}, \quad \text{on} \ \Gamma_i, \quad i = 1, \ldots, N, \quad (3.2)
\]

\[
c(\mathbf{r}) = c_\infty, \quad \text{on} \ \Gamma_0. \quad (3.3)
\]

where \( D \) is the simulation domain, \( \Gamma_i \) is the boundary of the \( i \)th dislocation which is a circle of radius \( r_d \) centered at \( \mathbf{r}_i = (x_i, y_i) \). and \( \Gamma_0 \) is the outer boundary which is a circle of radius \( r_\infty \) centered at the origin.

After the boundary value problem is solved, the climb velocity of the \( i \)th dislocation is obtained via the formula [15, 18]

\[
v_{cl}^{(i)} = \frac{D_v}{b} \int_{\Gamma_i} \frac{\partial c(\mathbf{r})}{\partial \nu_{\mathbf{r}}} dS(\mathbf{r}), \quad i = 1, \ldots, N. \]

(3.4)
We propose the following representation for the solution to the boundary value problem (3.1)-(3.3):
\[ c(r) = \sum_{i=1}^{N} \left( D_{\Gamma_i} [\rho_i](r) + \frac{1}{2\pi |\Gamma_i|} \int_{\Gamma_i} \rho_i dS \log |r - r_i| \right) + D_{\Gamma_0} [\rho_0](r), \tag{3.5} \]

where \( \rho_i \) is an unknown density supported on \( \Gamma_i \) \((i = 0, 1, \ldots, N)\), and \(|\Gamma_i|\) denotes the length of the boundary curve \( \Gamma_i \) \((|\Gamma_i| = 2\pi r_d \text{ for } i = 1, 2, \ldots, N \text{ in our case})\). In other words, the solution is represented via the sum of double layer potentials on boundary curves and point sources located at the center of each dislocation, where the strength of the point source is equal to the average value of the unknown density function on the boundary of that dislocation.

Obviously, the above representation is harmonic in \( D \) and thus satisfies the Laplace equation (3.1) in \( D \). Combining the boundary conditions (3.2)-(3.3) and the jump relations for the double layer potential (2.4), we obtain the following system of boundary integral equations for the unknown densities \( \rho_i \) \((i = 0, 1, \ldots, N)\)
\[
\begin{align*}
-\frac{1}{2} \rho_0(r) + & D_{\Gamma_0}^{PV} [\rho_0](r) \\
+ \sum_{i=1}^{N} & \left( D_{\Gamma_i} [\rho_i](r) + \frac{1}{2\pi |\Gamma_i|} \int_{\Gamma_i} \rho_i dS \log |r - r_i| \right) = c_{\infty}, \quad r \in \Gamma_0, \\
-\frac{1}{2} \rho_i(r) + & D_{\Gamma_i}^{PV} [\rho_i](r) + D_{\Gamma_i} [\rho_0](r) + \sum_{j=1, j \neq i}^{N} D_{\Gamma_j} [\rho_j](r) \\
+ \sum_{j=1}^{N} & \frac{1}{2\pi |\Gamma_j|} \int_{\Gamma_j} \rho_j dS \log |r - r_i| = g_i, \quad r \in \Gamma_i, \quad i = 1, \ldots, N.
\end{align*}
\](3.6)

The climb velocity can then be calculated from the obtained density of the double layer potential as
\[
v_{cl}^{(i)} = \frac{D_v}{2\pi r_d b} \int_{\Gamma_i} \rho_i dS, \quad i = 1, \ldots, N. \tag{3.7}
\]

The equations in (3.6) can be rewritten as \( A \rho = f \), where \( A = -\frac{1}{2} I + K \). Since \( K \) involves only the double layer potential operators and integration operators, \( K \) is compact and thus \( A \) is a second kind Fredholm operator. By the Fredholm alternative [25], if \( A \) is injective, then \( A^{-1} \) exists and is a bounded operator. In other words, in order to show that there exists a unique solution to the inhomogeneous system \( A \rho = f \), it suffices to show that the only solution to the homogeneous system \( A \rho = 0 \) is \( \rho = 0 \). That is, \( A \) does not have any nontrivial nullspace. The following theorem shows that \( A \) is injective.
Theorem 3.1. Let \( \rho_i (i = 0, 1, \ldots, N) \) solve the homogeneous system of equations
\[
- \frac{1}{2} \rho_0 (r) + D_{\Gamma_0}^{PV} [\rho_0] (r) + \sum_{i=1}^{N} \left( D_{\Gamma_i} [\rho_i] (r) + \frac{1}{2\pi |\Gamma_i|} \left( \int_{\Gamma_i} \rho_j dS \right) \log |r - r_j| \right) = 0, \quad r \in \Gamma_0,
\]
\[
- \frac{1}{2} \rho_i (r) + D_{\Gamma_i}^{PV} [\rho_i] (r) + D_{\Gamma_0} [\rho_0] (r) + \sum_{j=1, j \neq i}^{N} D_{\Gamma_j} [\rho_j] (r) + \sum_{j=1}^{N} \frac{1}{2\pi |\Gamma_j|} \left( \int_{\Gamma_j} \rho_j dS \right) \log |r - r_i| = 0, \quad r \in \Gamma_i, \quad i = 1, \ldots, N.
\]

Then \( \rho_i \equiv 0 \) for \( i = 0, 1, \ldots, N \).

Proof. Let \( \rho_i \) solve the system of equations (3.8) and let \( c(r) \) be as defined in equation (3.5). It then follows that \( c(r) \) satisfies the Laplace equation in \( D \) with \( c(r) \equiv 0 \) on the boundary \( \Gamma_i \) \( (i = 0, 1, \ldots, N) \). From the uniqueness of the solution to the interior Dirichlet problem for Laplace's equation, it follows that \( c(r) \equiv 0 \) in \( D \). Thus, \( \frac{\partial c}{\partial \nu} \equiv 0 \) on the boundary curves \( \Gamma_i \) \( (i = 0, 1, \ldots, N) \). A simple calculation shows that
\[
\frac{1}{2\pi} \int_{\gamma'} \left( \frac{\partial}{\partial \nu} \log |r - r'| \right) dS = \eta(r', \gamma')
\]
where \( \eta(r', \gamma') \) denotes the winding number of the curve \( \gamma' \) around the point \( r' \).

Combining equation (3.9) and property (2.7) of the double layer potential, we obtain
\[
\int_{\Gamma_i} \frac{\partial c}{\partial \nu} = - \frac{1}{|\Gamma_i|} \int_{\Gamma_i} \rho_i dS = 0.
\]

Thus,
\[
c(r) = \sum_{i=1}^{N} D_{\Gamma_i} [\rho_i] (r) + D_{\Gamma_0} [\rho_0] (r).
\]

From properties of the double layer potential, \( c(r) \) represents a harmonic function in each interior domain \( D_i \) enclosed by \( \Gamma_i \) as well as satisfies \( \frac{\partial c}{\partial \nu} \equiv 0 \) on \( \Gamma_i \). Thus from the uniqueness of the solution to the interior Neumann problem of the Laplace equation, we conclude that \( c(r) \equiv c_i \) in \( D_i \) where \( c_i \) is a constant. Using the jump relations for the double layer potential across \( \Gamma_i \),
\[
\rho_i = [c] = c_+ - c_-
\]
denotes the jump in \( c \) across the boundary \( \Gamma_i \). Thus in this case \( \rho_i \equiv c_i \). It then follows from equation (3.10) that \( c_i = 0 \), and hence \( \rho_i \equiv 0 \) on \( \Gamma_i \) for \( i = 1, 2, \ldots, N \).

Thus \( c(r) = D_{\Gamma_0} [\rho_0] (r) \), which satisfies \( c \equiv 0 \) in \( D_0 \) \( (D_0 \) is the whole domain enclosed by \( \Gamma_0 \)) and \( \frac{\partial c}{\partial \nu} \equiv 0 \) on \( \Gamma_0 \). From equation (2.6), we also have \( c(r) \to 0 \) as \( |r| \to \infty \). From the uniqueness of the solution to the exterior Neumann problem, we conclude that \( c \equiv 0 \) in \( \mathbb{R}^2 \setminus D_0 \). Finally, using the jump relations for the double layer potential across \( \Gamma_0 \), we conclude that \( \rho_0 \equiv 0 \). \( \square \)

Finally, combining (2.7), (3.5), (3.9), it is easy to get the climb velocity formula in (3.7).
4. Numerical Algorithms. We use the Nyström method [25] to discretize the integral equation system (3.6). Specifically, as pointed out in Section 2, the kernel of the double layer potential is smooth when the curve is smooth. The boundary data is constant on each boundary curve, so the unknown densities are also smooth. Therefore, the trapezoidal rule achieves spectral accuracy for the discretization of the system (3.6). Suppose that we use \( p \) equispaced points to discretize each dislocation core boundary \( \Gamma_i \) and \( p_0 \) equispaced points to discretize the outer boundary \( \Gamma_0 \). Then the size of the resulting discretized linear system is \((p_0 + pN) \times (p_0 + pN)\).

To solve the resulting linear system \( Ax = b \), we have implemented the following three algorithms:

(a) Algorithm 1: FMM+GMRES, i.e, use GMRES to solve the linear system iteratively with the FMM [4, 14] to accelerate the computation of the matrix-vector product.

(b) Algorithm 2: FDS, i.e., use the fast direct solver [20, 22] to construct an efficient factorization for \( A^{-1} \) to high precision, then simply apply the compressed \( A^{-1} \) to \( b \) to obtain the solution vector.

(c) Algorithm 3: FDS+FMM+GMRES, i.e., use the fast direct solver to construct an efficient factorization for the matrix inverse with low accuracy, denoted by \( A^{-1} \), then apply FMM accelerated iterative solve to solve the preconditioned linear system \( A^{-1} A x = A^{-1} b \).

After the linear system is solved, the climb velocity of each dislocation can then be computed via (3.7) with the integral in (3.7) replaced by the discrete summation \( \sum_{j=1}^{p} \rho_i(j) \). The overall algorithm is spectrally accurate.

5. FMM for Computing the Climb Force. Before solving the linear system (3.6), we need to calculate the right-hand side coefficients \( g_i, i = 1, \cdots, N \), which depends on the climb force \( f^{(i)}_{cl}, i = 1, \cdots, N \) as given in (3.2). We can compute these climb forces in \( O(N) \) time using one FMM and one biharmonic FMM as follows.

We write \( f^{(i)}_{cl} \) given in (1.5) and (1.6) as

\[
f^{(i)}_{cl} = \text{Imag} \left\{ \frac{3}{2} P^{(i)} - \frac{1}{2} Q^{(i)} \right\},
\]

where \( \text{Imag}(z) \) is the imaginary part of the complex number \( z \),

\[
P^{(i)} = \sum_{j=1, j \neq i}^{N} \frac{q_j}{z_i - z_j},
\]

\[
Q^{(i)} = \sum_{j=1, j \neq i}^{N} \frac{q_j (z_i - z_j)}{(z_i - z_j)^2},
\]

\[
q_i = \frac{\mu b^2}{2\pi(1-\nu)} \text{sgn}(i),
\]

and the location of dislocation \( z_i = x_i + iy_i \), for \( i = 1, \ldots, N \).

Here \( P^{(i)} \) in (5.2) in its form is the same as the Coulombic dipole interaction in two dimensions with source locations \( z_j \) and charge strengths \( q_i \) for \( i = 1, \cdots, N \), and can be evaluated using the classical FMM [4, 14] with \( O(N) \) calculations. Similarly, the interaction \( Q^{(i)} \) in (5.3) can be evaluated by the biharmonic FMM [10] in \( O(N) \) computational time.
6. Numerical Results. The algorithms described above have been implemented in Fortran and MATLAB. For the fast direct solver, we have used the MATLAB code from [19], which implements the algorithms in [20, 21, 22]. Here we illustrate the performance of our scheme via several numerical examples. Since the number of grid points is not very large in examples 1-3, we directly use GMRES to solve the linear system (3.6) there. In examples 4 and 5, we use the three algorithms described in section 4. We set the number of numerical grid points \( p_0 = 128 \) in the outer boundary except in example 1. The timing results were obtained on a laptop with a 2.10GHz Intel(R) Core(TM) i7-4600U processor and 4GB of RAM.

In all the examples, we set \( \epsilon_d = b \) and \( \epsilon_\infty = 10^7 \epsilon_d \), unless specified otherwise. We consider dislocation climb in aluminum. The values of physical parameters are listed below, which can be found in [18, 30]. The Burgers vector has magnitude \( b = 2.86^\circ \text{A} \); the shear modulus is \( \mu = 26.5 \text{GPa} \); the Poisson ratio is \( \nu = 0.347 \); the Boltzmann constant is \( k_B = 8.62 \times 10^{-5} \text{eV} \cdot \text{K}^{-1} \); the temperature is \( T = 500 \text{K} \); the vacancy diffusion constant is \( D_v = D_v^0 e^{-\frac{E_m}{k_B T}} \) with \( D_v^0 = 1.51 \times 10^{-5} \text{m}^2 \cdot \text{sec}^{-1} \); the vacancy migration energy is \( E_m = 0.61 \text{eV} \); the atomic volume is \( \Omega = 16.3 \text{A}^3 \); the reference vacancy concentration is \( c_0 = e^{-\frac{E_f}{k_B T}} \); the vacancy formation energy is \( E_f = 0.67 \text{eV} \); finally, the vacancy concentration at \( \epsilon_\infty \) is set to \( c_\infty = c_0 \).

Example 1: A Single Dislocation. We first consider a single dislocation located at the center of the computation domain. For this simple case, the analytical solution to (3.1)–(3.3) is given by the formula

\[
c(r) = \frac{c_0 \log \left( \frac{r_\infty}{r} \right) + c_\infty \log \left( \frac{r}{r_d} \right)}{\log \left( \frac{r_\infty}{r_d} \right)},
\]

and the climb velocity for the dislocation is then given by

\[
v^{(1)}_{cl} = \frac{4\pi^2 r_d D_v (c_\infty - c_0)}{b \log \left( \frac{r_\infty}{r_d} \right)}.
\]

Our numerical experiments show that it only requires \( p_0 = 3 \), \( p = 3 \), and two GMRES iterations to achieve full double precision.

Example 2: Dynamics of a Dislocation Dipole. In this example, we study the dynamics of a dislocation dipole with a positive dislocation located at \( (0, r_0/2) \) and a negative dislocation at \( (0, -r_0/2) \). In this setup, it follows from symmetry that the positive and the negative dislocation travel towards each other with the same speed. Let \( r_i(t) \) denote the separation between the two dislocations. The dynamics of the dipole is simply described by

\[
\frac{dr_i(t)}{dt} = -2v_{cl}(r_i(t)).
\]

We calculate \( v_{cl}(r_i(t)) \) using our method and then evolve this equation. Each dislocation core boundary is discretized with \( p = 8 \) points in the simulation. The separation distance \( r_i(t) \) is updated using the forward Euler scheme with \( \Delta t = 10^{-4} \text{s} \). The initial distance between the two dislocations is \( r_0 = 40 \text{A} \).

The results of the numerical simulation are compared to an approximate solution for the setup given by

\[
\frac{dr_a(t)}{dt} = -2v_{cl,a}(r_a(t)),
\]
where
\[
v_{cl,a}(r_a(t)) = \frac{\pi D v}{b \ln(r_\infty/\sqrt{r_a(t)r_d})} \left( c_\infty - c_0 e^{\frac{f_{cl}(t)\Omega}{k_BT}} \right) .
\] (6.5)

This equation is also evolved using the forward Euler scheme from the same initial distance with \( \Delta t = 10^{-4}s \).

Fig. 6.1: Dynamics of a dislocation dipole. (a) Evolution of the separation of the two dislocations in the dipole. The numerical solution \( r_i(t) \) is shown in blue dotted line and the approximate solution \( r_a(t) \) given by (6.4)-(6.5) is shown in red line. (b) The relative error \( |r_i(t) - r_a(t)|/|r_a(t)| \) versus time.

We plot out the numerical solution \( r_i(t) \) and the approximate analytical solution \( r_a(t) \) in Figure 6.1(a), and the relative error of the numerical solution against \( r_a(t) \) in Figure 6.1(b). Both figures show that the numerical solution obtained using our SKIE formulation agrees with the approximate solution very well.

**Example 3: Stability Analysis of Dislocation Arrays.** Consider an array of positive dislocations located along the \( y \)-axis centered at \((0, jD + \varepsilon \cos(2\pi j/N))\) for \( j = -N_0, \ldots, N_0 \). Here \( D \) is the average spacing between two adjacent dislocations and the dislocations are perturbed by a longitudinal wave with amplitude \( \varepsilon \) and wave length \( \lambda = ND \). When the climb force \( f_{cl}(j) \) is small, the boundary data can be approximated by its linear approximation as follows:
\[
c(r) \approx c_0 \left( 1 - \frac{f_{cl}(j)\Omega}{bk_BT} \right) , \quad \text{on } \Gamma_i .
\] (6.6)

In this case, it is shown in [16] that the climb velocity \( v_{cl}(j)(t) \) has the following expression
\[
v_{cl}(j)(t) = \omega \varepsilon e^{\omega t} \cos(2\pi j/N),
\] (6.7)
where under outer periodic boundary conditions the perturbation growth rate \( \omega \) is
given by the formula
\[
\omega = \frac{4\pi^2 c_0 \mu D_0 N}{k_B T(1 - \nu) D^2} \left( \frac{1}{N} \right) \times \sum_{p=1}^{N} \left[ \cos(2\pi p/N) \ln(1 - 2\cos(2\pi p/N)e^{-2\pi r_d/(ND)} + e^{-4\pi r_d/(ND)}) \right].
\] (6.8)

It is also shown in [16] that this perturbation growth rate \( \omega \) is negative for all \( N > 1 \), meaning that the dislocation array is always stable with respect to small perturbations in the dislocation climb direction.

---

**Fig. 6.2:** Stability of a dislocation array: Climb velocity of dislocations near the center of the perturbed array for various \( N \). The horizontal axis shows the unperturbed locations of dislocations with unit \( D \). The vertical axis shows the climb velocity in the unit \( \AA/s \). Blue lines are the theoretical results given by (6.7), while red \( \times \) markers are the numerical values.

We conducted several numerical experiments to study this stability using our SKIE formulation. We set \( D = 50 r_d \), \( N_0 = 500 \), and vary \( N \) from 10, 20, \ldots, 320. The perturbation amplitude \( \varepsilon = 0.0075 \lambda \). We discretize each dislocation core boundary with \( p = 8 \) points in the simulation. In order to reduce the effect of finite size and the difference between outer boundary conditions, we plot out the numerical climb velocity near the center of the array and compare the climb velocity at the center of the dislocation array with the formula given by (6.8). The results are shown in
An Efficient High Order Method for Dislocation Climb in 2D

$N = \lambda / D$

Fig. 6.3: Stability of a dislocation array: Perturbation decay rate $|\omega|$ (defined in (6.7)) evaluated at the dislocation at the center for $N = \lambda / D = 10, 20, \ldots, 320$. The unit of $\omega$ is $s^{-1}$.

Figures 6.2 and 6.3, respectively. We observe that the agreement is very good. In particular, the accuracy of the perturbation growth rate is about 3 digits.

**Example 4: Convergence and Timing Results of Dislocation Arrays.** In this example, we present a detailed numerical study for dislocation arrays, using the three algorithms described in section 4. We place $N$ dislocations along the $y$-axis with average spacing $D = 100r_d$ and a uniform random perturbation of magnitude $45r_d$. We set $r_{\infty} = 10^8r_d$. GMRES is set to terminate when the relative residual falls below $10^{-12}$ in both Algorithms 1 and 3. And the numerical rank tolerance for the fast direct solver is also set to $10^{-12}$ in Algorithm 2 and $10^{-6}$ in Algorithm 3.

Table 6.1 lists relative $L^2$ error of the climb velocity for various $p$ (number of discretization points per dislocation) and $N$ (total number of dislocations) using Algorithm 3, which shows that our scheme is spectrally accurate. The relative errors using Algorithms 1 and 2 are similar, except that we only calculate the results for $N$ up to $128^2$ using Algorithm 1 due to the memory and timing constraints of the computer we used.

Table 6.1: The relative $L^2$ error of the solution versus the number of discretization points $p$ per dislocation for various $N$ using Algorithm 3 for dislocation arrays. The reference solution is obtained with $p = 16$.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$8^2$</th>
<th>$16^2$</th>
<th>$32^2$</th>
<th>$64^2$</th>
<th>$128^2$</th>
<th>$256^2$</th>
<th>$512^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p = 2$</td>
<td>1.2e-03</td>
<td>9.3e-04</td>
<td>2.0e-03</td>
<td>2.1e-03</td>
<td>2.9e-03</td>
<td>2.5e-03</td>
<td>2.8e-03</td>
</tr>
<tr>
<td>$p = 4$</td>
<td>2.9e-06</td>
<td>2.4e-06</td>
<td>1.2e-05</td>
<td>1.3e-05</td>
<td>2.4e-05</td>
<td>1.9e-05</td>
<td>2.2e-05</td>
</tr>
<tr>
<td>$p = 6$</td>
<td>6.7e-09</td>
<td>6.0e-09</td>
<td>7.0e-08</td>
<td>8.3e-08</td>
<td>2.2e-07</td>
<td>1.5e-07</td>
<td>1.8e-07</td>
</tr>
<tr>
<td>$p = 8$</td>
<td>4.8e-11</td>
<td>3.8e-11</td>
<td>4.1e-10</td>
<td>5.3e-10</td>
<td>2.1e-09</td>
<td>1.3e-09</td>
<td>1.5e-09</td>
</tr>
</tbody>
</table>
Tables 6.2 and 6.3 list the number of iterations for Algorithms 1 and 3, respectively. Here we observe that the number of iterations is more or less independent of $p$ - the number of discretization per dislocation, which is characteristic for second kind integral equations. When the number of dislocations $N$ increases, the number of iterations in Algorithm 1 also increases gradually, indicating a geometric ill-conditioning of the problem. However, the preconditioner by the low accuracy fast direct solver is very effective, reducing the number of iterations in Algorithm 3 to almost a constant for all $N$ and $p$.

Table 6.2: The number of iterations of GMRES in Algorithm 1 versus the number of discretization points $p$ per dislocation and the total number of dislocations $N$ in the system. The dislocations are located along $y$-axis with average spacing $D = 100r_d$.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$8^2$</th>
<th>$16^2$</th>
<th>$32^2$</th>
<th>$64^2$</th>
<th>$128^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p = 2$</td>
<td>31</td>
<td>50</td>
<td>90</td>
<td>140</td>
<td>200</td>
</tr>
<tr>
<td>$p = 4$</td>
<td>35</td>
<td>56</td>
<td>110</td>
<td>160</td>
<td>200</td>
</tr>
<tr>
<td>$p = 6$</td>
<td>35</td>
<td>56</td>
<td>110</td>
<td>160</td>
<td>200</td>
</tr>
<tr>
<td>$p = 8$</td>
<td>35</td>
<td>56</td>
<td>110</td>
<td>170</td>
<td>200</td>
</tr>
</tbody>
</table>

Table 6.3: The number of iterations of GMRES in Algorithm 3 versus $p$ and $N$.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$8^2$</th>
<th>$16^2$</th>
<th>$32^2$</th>
<th>$64^2$</th>
<th>$128^2$</th>
<th>$256^2$</th>
<th>$512^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p = 2$</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>$p = 4$</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>$p = 6$</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>$p = 8$</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>

Table 6.4: Timing results of Algorithm 1 for dislocation arrays.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$8^2$</th>
<th>$16^2$</th>
<th>$32^2$</th>
<th>$64^2$</th>
<th>$128^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p = 2$</td>
<td>0.232</td>
<td>0.512</td>
<td>2.06</td>
<td>10.3</td>
<td>56.3</td>
</tr>
<tr>
<td>$p = 4$</td>
<td>0.292</td>
<td>0.760</td>
<td>3.74</td>
<td>19.1</td>
<td>93.1</td>
</tr>
<tr>
<td>$p = 6$</td>
<td>0.332</td>
<td>0.932</td>
<td>5.13</td>
<td>27.3</td>
<td>136</td>
</tr>
<tr>
<td>$p = 8$</td>
<td>0.372</td>
<td>1.08</td>
<td>6.36</td>
<td>36.6</td>
<td>173</td>
</tr>
</tbody>
</table>

Table 6.5: Timing results of Algorithm 2 for dislocation arrays.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$8^2$</th>
<th>$16^2$</th>
<th>$32^2$</th>
<th>$64^2$</th>
<th>$128^2$</th>
<th>$256^2$</th>
<th>$512^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p = 2$</td>
<td>0.081</td>
<td>0.15</td>
<td>0.39</td>
<td>1.23</td>
<td>4.42</td>
<td>18.4</td>
<td>69.1</td>
</tr>
<tr>
<td>$p = 4$</td>
<td>0.142</td>
<td>0.26</td>
<td>0.75</td>
<td>2.58</td>
<td>9.88</td>
<td>39.7</td>
<td>153</td>
</tr>
<tr>
<td>$p = 6$</td>
<td>0.179</td>
<td>0.362</td>
<td>1.25</td>
<td>4.48</td>
<td>17.4</td>
<td>72.0</td>
<td>280</td>
</tr>
<tr>
<td>$p = 8$</td>
<td>0.171</td>
<td>0.402</td>
<td>1.36</td>
<td>4.95</td>
<td>19.4</td>
<td>78.8</td>
<td>313</td>
</tr>
</tbody>
</table>
Table 6.6: Timing results of Algorithm 3 for dislocation arrays.

<table>
<thead>
<tr>
<th>N</th>
<th>8²</th>
<th>16²</th>
<th>32²</th>
<th>64²</th>
<th>128²</th>
<th>256²</th>
<th>512²</th>
</tr>
</thead>
<tbody>
<tr>
<td>p = 2</td>
<td>0.111</td>
<td>0.196</td>
<td>0.444</td>
<td>1.37</td>
<td>5.06</td>
<td>21.1</td>
<td>96.7</td>
</tr>
<tr>
<td>p = 4</td>
<td>0.172</td>
<td>0.261</td>
<td>0.706</td>
<td>2.26</td>
<td>9.20</td>
<td>38.2</td>
<td>161</td>
</tr>
<tr>
<td>p = 6</td>
<td>0.211</td>
<td>0.348</td>
<td>1.05</td>
<td>3.70</td>
<td>15.4</td>
<td>60.8</td>
<td>252</td>
</tr>
<tr>
<td>p = 8</td>
<td>0.207</td>
<td>0.420</td>
<td>1.18</td>
<td>4.47</td>
<td>17.5</td>
<td>76.0</td>
<td>297</td>
</tr>
</tbody>
</table>

Fig. 6.4: Timing results for dislocation arrays. The dashed line shows $O(N)$ complexity. $N$ is the number of dislocations and $p$ is set to 8. Both axes use logarithmic scale.

Tables 6.4, 6.5, and 6.6 list the timing results in seconds for three algorithms. Algorithm 1 exhibits suboptimal complexity due to the increase in the number of iterations when $N$ grows. But both Algorithm 2 and Algorithm 3 are at least one order of magnitude faster than Algorithm 1 and have optimal complexity. For Algorithm 2, this is in agreement with known theoretical results that the computational complexity of fast direct solvers for intrinsically 1D problems with nonoscillatory kernels is $O(N)$. Algorithm 3 has very close performance as Algorithm 2 due to the stabilization of number of iterations. The timing results comparing three algorithms for $p = 8$ are also plotted in Figure 6.4.

Example 5: Convergence and Timing Results of 2D Dislocation Distributions. We now consider distributions of dislocations in 2D. The distributions are generated by placing $N_x \times N_y$ dislocations in a lattice of spacing $D = 100r_d$ with random perturbations of magnitude $45r_d$. See Figure 6.5 for an example of the distributions.

Tables 6.7 and 6.8 list the number of iterations of GMRES for Algorithms 1 and 3 for random distributions of dislocations. We observe similar behavior as in Tables 6.2 and 6.3 (we have lowered the GMRES stopping tolerance to $10^{-10}$ due to
Fig. 6.5: Random distribution of dislocations in Example 5.

Table 6.7: The number of iterations of GMRES for Algorithm 1 versus $p$ and $N$ for random distributions of dislocations.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$8^2$</th>
<th>$16^2$</th>
<th>$32^2$</th>
<th>$64^2$</th>
<th>$128^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p = 2$</td>
<td>33</td>
<td>47</td>
<td>77</td>
<td>120</td>
<td>200</td>
</tr>
<tr>
<td>$p = 4$</td>
<td>33</td>
<td>47</td>
<td>74</td>
<td>130</td>
<td>200</td>
</tr>
<tr>
<td>$p = 6$</td>
<td>33</td>
<td>47</td>
<td>77</td>
<td>120</td>
<td>190</td>
</tr>
<tr>
<td>$p = 8$</td>
<td>33</td>
<td>47</td>
<td>90</td>
<td>140</td>
<td>180</td>
</tr>
</tbody>
</table>

Table 6.8: The number of iterations of GMRES for Algorithm 3 versus $p$ and $N$ for random distributions of dislocations.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$8^2$</th>
<th>$16^2$</th>
<th>$32^2$</th>
<th>$64^2$</th>
<th>$128^2$</th>
<th>$256^2$</th>
<th>$512^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p = 2$</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>$p = 4$</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>$p = 6$</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>$p = 8$</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>

the stagnation of the residual error because of the ill-conditioning of the problem for Table 6.3). That is, the number of iterations is roughly independent of $p$ but gradually increases as $N$ increases for Algorithm 1. While the number of iterations roughly remains a very low constant for Algorithm 3 due to the excellent preconditioned by the low accuracy FDS factorization. A comparison of Tables 6.2 and 6.7 also shows that the number of iterations is somewhat insensitive to the configuration of dislocations for the unpreconditioned linear system.

Tables 6.9, 6.10, and 6.11 list the timing results of the three algorithms for random distributions of dislocations. We observe that Algorithm 1 behaves quite similarly as
Table 6.9: Timing results of Algorithm 1 for random distributions of dislocations.

<table>
<thead>
<tr>
<th>N</th>
<th>8²</th>
<th>16²</th>
<th>32²</th>
<th>64²</th>
<th>128²</th>
</tr>
</thead>
<tbody>
<tr>
<td>p = 2</td>
<td>0.308</td>
<td>0.768</td>
<td>4.09</td>
<td>24.0</td>
<td>177</td>
</tr>
<tr>
<td>p = 4</td>
<td>0.380</td>
<td>1.46</td>
<td>8.58</td>
<td>61.7</td>
<td>393</td>
</tr>
<tr>
<td>p = 6</td>
<td>0.440</td>
<td>1.69</td>
<td>10.1</td>
<td>67.2</td>
<td>436</td>
</tr>
<tr>
<td>p = 8</td>
<td>0.472</td>
<td>1.95</td>
<td>14.1</td>
<td>91.8</td>
<td>485</td>
</tr>
</tbody>
</table>

Table 6.10: Timing results of Algorithm 2 for random distributions of dislocations.

<table>
<thead>
<tr>
<th>N</th>
<th>8²</th>
<th>16²</th>
<th>32²</th>
<th>64²</th>
<th>128²</th>
</tr>
</thead>
<tbody>
<tr>
<td>p = 2</td>
<td>0.056</td>
<td>0.461</td>
<td>1.72</td>
<td>10.5</td>
<td>82.3</td>
</tr>
<tr>
<td>p = 4</td>
<td>0.242</td>
<td>0.822</td>
<td>4.12</td>
<td>27.7</td>
<td>217</td>
</tr>
<tr>
<td>p = 6</td>
<td>0.349</td>
<td>1.17</td>
<td>6.18</td>
<td>43.7</td>
<td>339</td>
</tr>
<tr>
<td>p = 8</td>
<td>0.443</td>
<td>1.58</td>
<td>7.82</td>
<td>51.9</td>
<td>386</td>
</tr>
</tbody>
</table>

Table 6.11: Timing results of Algorithm 3 for random distributions of dislocations.

<table>
<thead>
<tr>
<th>N</th>
<th>8²</th>
<th>16²</th>
<th>32²</th>
<th>64²</th>
<th>128²</th>
<th>256²</th>
<th>512²</th>
</tr>
</thead>
<tbody>
<tr>
<td>p = 2</td>
<td>0.104</td>
<td>0.395</td>
<td>0.96</td>
<td>3.93</td>
<td>17.0</td>
<td>83.1</td>
<td>401</td>
</tr>
<tr>
<td>p = 4</td>
<td>0.171</td>
<td>0.501</td>
<td>1.56</td>
<td>6.16</td>
<td>27.0</td>
<td>126</td>
<td>582</td>
</tr>
<tr>
<td>p = 6</td>
<td>0.240</td>
<td>0.546</td>
<td>1.84</td>
<td>7.49</td>
<td>32.7</td>
<td>150</td>
<td>677</td>
</tr>
<tr>
<td>p = 8</td>
<td>0.310</td>
<td>0.727</td>
<td>2.53</td>
<td>9.97</td>
<td>43.2</td>
<td>196</td>
<td>870</td>
</tr>
</tbody>
</table>

in the case of dislocation arrays. Algorithm 2 becomes much slower due to the increase of the numerical rank in off-diagonal blocks. We would like to remark here that we use \texttt{rskelf} instead of \texttt{hifie2} or \texttt{hifie2x} from [19] as the fast direct solver simply due to the fact that \texttt{rskelf} is faster than \texttt{hifie2} by about a factor of 2 for our problem for problem sizes which can be studied on a laptop due to memory and time constraints. One should switch to \texttt{hifie2} or \texttt{hifie2x} for better scaling results for larger scale problems. Nevertheless, Algorithm 3 is still much faster than Algorithms 1 and 2 and exhibits roughly optimal complexity, albeit slower than the array case due to the increase of the numerical rank in off-diagonal blocks (it is clear that the exact asymptotic complexity of Algorithm 3 is determined by that of the FDS in [22]). The timing results comparing the three algorithms with \( p = 8 \) are also plotted in Figure 6.4.

7. Conclusions and Discussions. We have constructed a SKIE formulation and developed numerical algorithms based on it for dislocation climb in two dimensions that enable large scale dislocation dynamics simulations. The numerical algorithms are spectrally accurate and require nearly optimal number of discretization points for a given accuracy. We have compared three different algorithms for solving the resulting linear system. Among these three algorithms, we recommend Algorithm 3 which solves the preconditioned linear system via FMM+GMRES iterative solver where the preconditioner is obtained by the low accuracy FDS factorization of \( A^{-1} \) in [19, 22]. This algorithm reduces the number of iterations to almost a very small constant regardless of number of dislocations and their geometries in the system and
achieves roughly optimal complexity.

The SKIE formulation can be generalized to solve three dimensional dislocation problems in a straightforward manner. A high-order efficient numerical scheme can be developed based on that SKIE formulation using the FMM-accelerated QBX (“Quadrature By Expansion”) scheme (see, e.g. [31]), combined with FMM based algorithms for evaluating the Peach-Koehler force of dislocations in three dimensions (e.g. [1, 32, 36, 37]). For three dimensional problems, the conditioning seems to be better since the leading term in a multipole expansion is a constant instead of a logarithmic function (see, for example, [7, 9] for similar problems). Finally, when the diffusion of vacancies is not that fast, one needs to solve the heat equation instead of the Laplace equation. There are standard potential theory for the heat equation which leads to well-conditioned SKIE formulation. Efficient algorithms can be developed for evaluating the heat layer potentials and solving the associated time-dependent integral equations. These issues are currently under investigation and the results will be reported on later dates.

Acknowledgments. The authors would like to thank Prof. Leslie Greengard at Courant Institute, Prof. Vladimir Rokhlin at Yale University, and Dr. Kenneth L. Ho for helpful discussions.

REFERENCES

An Efficient High Order Method for Dislocation Climb in 2D


