A multigrid solver for modeling complex interseismic stress fields.

Seoleun Shin[a], Gert Zöller[a], Matthias Holschneider[a], Sebastian Reich[a]

[a] *Universität Potsdam, Institut für Mathematik,*
Am Neuen Palais 10, D-14469 Potsdam, Germany

Abstract

We develop a multigrid, multiple time stepping scheme to reduce computational efforts for calculating complex stress interactions in a strike-slip 2D planar fault for the simulation of seismicity. The key elements of the multilevel solver are separation of length scale, grid-coarsening, and hierarchy. In this study the complex stress interactions are split into two parts: the first with a small contribution is computed on a coarse level, and the rest for strong interactions is on a fine level. This partition leads to a significant reduction of the number of computations. The reduction of complexity is even enhanced by combining the multigrid with multiple time stepping. Computational efficiency is enhanced by a factor of 10 while retaining a reasonable accuracy, compared to the original full matrix-vortex multiplication. The accuracy of solution and computational efficiency depend on a given cut-off radius that splits multiplications into the two parts. The multigrid scheme is constructed in such way that it conserves stress in the entire half-space.

Keywords: Multigrid, Multiple time stepping, Strike-slip fault model

1. Introduction

Multiplications of a vector by a dense matrix demand high computational expense for half-space elastodynamic solutions in a fault model for the simulation of seismicity (Ben-Zion and Rice, 1993; Ben-Zion, 1996; Zöller et al., 2004, 2005). Such fault models calculate the evolution of slip, stress, and other related quantities as a response on long-term accumulation of stress in the Earth's crust resulting from the motion of the tectonic plates. The outcome of those models is an earthquake catalog including time, hypocenter and

magnitude of each event. These data are useful for purposes of earthquake statistics (e.g. frequency-size distributions, recurrence times) and seismic hazard studies. In this way, the poor statistics of observational earthquake data can be overcome to some extent. The interseismic build-up of stress period is related to plate motion with constant velocity. The release of stress comes from power-law creep (interseismic) accounting for aseismic processes and from earthquakes (coseismic). On average there is a balance between build-up and release of stress (backslip model). Recurrence times of large earthquakes are tens to hundreds of years, while the earthquake itself occurs on a time-scale of a few seconds. In simple fault models, different regimes of a fault are loaded independently during the periods between earthquakes. More realistic models include complex spatio-temporal interactions at each time. This leads to expensive multiplications as in many-body simulations in other physical problems. There have been a number of efforts to reduce the computational cost for many-body calculations such as the Barnes-Hut method, the parallel tree methods, and the fast multipole expansion method in tree algorithms for long-range potentials. Meanwhile mesh-based fast algorithms include the particle-particle particle-mesh method, the particle-mesh Ewald summation, and multigrid methods and adaptive refinement (Griebel et al. (2007) and references therein). These methods attempted to reduce the complexity of $N \times N$ to $\mathcal{O}(N)$ or $\mathcal{O}(N\log N)$ for interactions of a set of N particles or bodies. Moreover, these approaches can be combined with a multiple time stepping to further enhance the computational speed. To develop efficient solvers for elastodynamics some studies use, for example, fast multipole boundary element methods (e.g., Chaillat et al., 2008) or parallel computations on grids with different spacing (Aoi and Fujiwara, 1999).

Iterative multigrid methods are used as fast numerical methods for the solution of a linear equation arising for partial differential equations (Trottenberg et al., 2007). The idea of using multiple grids was adopted for an efficient multiplication by a dense matrix, in a non-iterative way (e.g., Brandt and Lubrecht, 1990). Amongst fast multiplication methods, multigrid methods have advantages in that they are relatively simple to implement and applicable to general potentials. In this work we test multigrid methods combined with a multiple time stepping, based on the multigrid approach in Skeel et al. (2002), hereafter STH02. Then we compare the results with those from the original full matrix-vector multiplications. In preliminary tests we found that the method of STH02 can lead to a lower error than using the multigrid method suggested by Brandt and Lubrecht (1990), hereafter BL90,

for the problem in the fault model. The difference between the algorithms in BL90 and STH02 will be discussed in the section of algorithm description.

In the following section, we describe the fault in earthquake modeling briefly and the kernel used in the multiplications. In section 3, the idea of the multigrid algorithm in STH02 is explained and the advantage of this approach over that in BL90. In section 4, we present the results from the multigrid multiplications in the fault model. Finally we summarize this study and deliver outlook for implementations in operational models.

2. Details of the fault model

The earthquake model under consideration includes two mechanisms: first the stress loading of a fault region resulting from plate motion (interseismic period), and second, the earthquake process which is initiated when the stress equals a material threshold and leads to a sequence of stress redistributions on the fault (coseismic period). While the interseismic period lasts for years to centuries, the coseismic period takes only seconds to minutes. The present investigation focuses on the interseisimic process, which dominate the computational effort due to complex stress interactions during each time step. The computational grid, where stress and slip are maintained, is a rectangular area which is segmented into cells (Ben-Zion, 1996; Zöller et al., 2005). The size of the grid is 70 km in length and 17.5 km in depth. This geometry corresponds approximately to the San Andreas fault near Parkfield in California (Ben-Zion and Rice, 1993; Ben-Zion, 1996). In fact, the entire fault is an infinite half-plane, but brittle processes are calculated only on the area depicted in Figure 1. The computational grid is discretized to 128×32 cells of uniform size where stress and slip are calculated through interaction between the cells. The material surrounding the fault is assumed to be a homogeneous elastic half-space, which is characterized by elastic parameters and a Green's function. For reasons of transparency, we start with a homogeneous half-space. Models for layered half-spaces can be derived by changing the Green's function (see e.g. Wang (1999)): A simple orthonormalization method for the stable and efficient simulations may be used, if corresponding data are given. The same holds for bimaterial interfaces and other model extensions. The static Green's function $G(\mathbf{x}_i; \mathbf{x}_j)$ in our study is based on Chinnery's solution for static dislocations of rectangular patches in a strike-slip fault embedded in an elastic Poisson solid with rigidity $\mu = 30\ GPa$ (Chinnery, 1963; Okada, 1992) and defines the interaction between two po-

sitions $\mathbf{x}_i$ and $\mathbf{x}_j$ on the grid. Here $i = j = 1, 2, \cdots, N$, where N is the total number of cells, i.e., $N = 128 \times 32 = 4096$. More details about the fault model can be found in Ben-Zion and Rice (1993), Ben-Zion (1996), and Zöller et al. (2005).

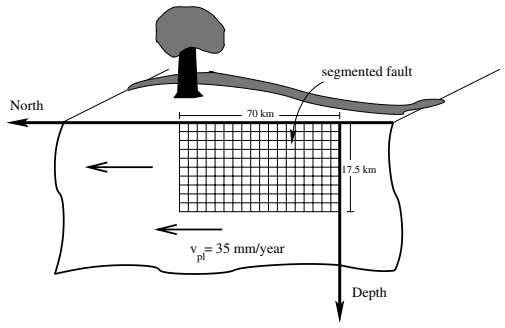

Figure 1: Schematic diagram of the fault model. Courtesy of Zöller et al. (2004).

We are concerned with interseismic processes in which stress in the fault zone builds up. The stress response (τ) at $\mathbf{x}_i$ to a static change of the displacement u at $\mathbf{x}_j$ is given by

$$\tau(\mathbf{x}_i, t) = \sum_{\mathbf{x}_j \in \text{grid}} G(\mathbf{x}_i; \mathbf{x}_j)[u(\mathbf{x}_j, t) - v_{\text{pl}} t], \tag{1}$$

at time t since the start of the simulation (Ben-Zion and Rice, 1993; Ben-Zion, 1996). The velocity of the tectonic plate, $v_{\text{pl}} = 35$ mm/yr based on empirical values for the San Andreas fault in California. We denote $\mathbf{x}_i$ as evaluation position and $\mathbf{x}_j$ as source point. We assume that the computational grid is embedded in a half-plane and undergoes constant creep. Then the build-up of stress can be reduced by the aseismic creep motion, whose time-rate is given by

$$\dot{u}(\mathbf{x}_j) = c(\mathbf{x}_j)\tau(\mathbf{x}_j, t)^3, \tag{2}$$

where $c(\mathbf{x}_j)$ is the rate of aseismic creep which is constant in time, but space-dependent.

The kernel G in the fault model (hereinafter called Chinnery kernel) has a finite element at $i = j$. Note that this is different to the electric potential ($\sim 1/r$) where the diagonal element is a singular point. The Chinnery kernel also

decreases more rapidly ($1/r^3$) with increasing distance from the source point. Consequently, the contribution of the long-range interaction to the total sum is much smaller than that between neighbouring cells. Nevertheless, the long-range contribution should be included in the integration to account for the conservation of stress in the entire half-space. Consequently, the number of computations for (1) is $N \times N$, which brings about computational complexity.

3. Idea of the multigrid algorithm in STH02

The $N \times N$ multiplication can be done, however, in multi levels within reasonable error ranges, which is proportional to $(h/a)^2$. Here h is a grid size and 0.5 km in this study, and a is a cut-off radius that will be explained more in the following description. The idea of using multiple grids for such problems was suggested in Hackbusch and Nowak (1989) and BL90. This has been relatively less popular than tree methods, for example, in Greengard and Rokhlin (1987), whose method has some similar features to that in Hackbusch and Nowak (1989). Recently STH02 compared the tree methods and multigrid methods, and found that the multigrid is twice as fast as simulations with the fast multipole method for the same accuracy in tests of 20,544-atom model of water. Moreover, the advantage of the multigrid method is simple implementation and continuously differentiable approximations to a Green's function.

The three essential elements of the multilevel solver is separation of length scale, coarsening, and hierarchy. The *separation* of length scale means the split between a rapidly changing part and a smooth part of the kernel, to distinguish short range and long-range interactions. The former is to be computed directly, while the latter is approximated using the results from coarsening. The long-range interaction is computed at levels with larger grid spacing, so called coarse levels. The grid spacing doubles with increasing levels so that it is $2h$ for the level 2 when the spacing is h at the level 1 where direct computation is done. For simplicity we use superscripts h, $2h$ to denote the quantities and grids at each level. For example, the grids of level 1 and level 2 are Ω^h and Ω^{2h} respectively. For the separation of length scale, we need to determine a cut-off distance a, inside which direct computation is done. It means that interactions between points with $r \leq a$ are classified as short-range interactions and otherwise long-range ones. The idea of STH02 is to split the kernel like

$$G = (G - \widetilde{G}) + \widetilde{G}, \tag{3}$$

where $G-\widetilde{G}$ vanishes for r beyond a so that this is calculated directly inside of a, while the smoothed function $\widetilde{G}$ is computed on coarse grids. As a increases $\widetilde{G}$ becomes more smooth. The scheme in BL90 is also based on the splitting of the kernel depending on the length scale of interactions. However, the difference is that it is *not* separated in such way that $G-\widetilde{G}$ vanishes beyond a. This difference will be detailed more in the following paragraphs.

We use I, J, for grid indices at the level 2 and $I, J = 1, 2, ...N/4$ for a 2-D fault model. We consider 2-level algorithm only, for the relatively small domain in this study (see section 2). An extension to a higher level algorithm is easy to handle when it is necessary. The long-distance interactions can be approximated with fewer terms through a *Coarsening*. The coarsening means that $\widetilde{G}_h$, $\widetilde{G}$ on the grid Ω_h, is approximated by $\widetilde{G}_{2h}$ on the grid Ω_{2h} in such way that

$$\widetilde{G}_h \approx I_{2h}^h \widetilde{G}_{2h} I_h^{2h}, \tag{4}$$

where I_{2h}^h is an interpolation and I_h^{2h} a restriction operator. In our computations we use a linear interpolation using nearest points values and a restriction operator described in the Appendix. The coarse level calculation can be further split into a two-level computation, which leads to a 3-level scheme. This recursive application of the *Separation* and *Coarsening* is called *Hierarchy* (Skeel et al., 2002). Using the idea of (3) and (4), (1) can be approximated in a 2-level scheme by

$$\begin{aligned} \tau_i^h &\approx \sum_j (G_{i,j}^h - \widetilde{G}_{i,j}^h) u_j^h + I_{2h}^h \sum_J \widetilde{G}_{I,J}^{2h} u_J^{2h} \\ &\approx \sum_{||\mathbf{x_j}-\mathbf{x_i}||\le a} (G_{i,j}^h - \widetilde{G}_{i,j}^h) u_j^h + I_{2h}^h \sum_J \widetilde{G}_{I,J}^{2h} u_J^{2h}, \end{aligned} \tag{5}$$

along with the definition $\widetilde{G}_{i,j}^h \equiv G_{i,j}^h$ for $||\mathbf{x_j}-\mathbf{x_i}|| > a$ in the scheme. STH02 used a smoothed potential for $\widetilde{G}$, which is based on the Taylor expansion of the potential function $(1/r)$ to soften the original kernel function. In the same way, we also derive an approximation to the Chinnery kernel in the fault model. It is worth noting that the idea of using a *softened kernel* is also suggested in Brandt (1991) although the instruction for application is more straightforward in STH02. Multigrid computations with a softened kernel for a logarithmic kernel are tested also in Brandt and Venner (1998).

The advantage of using $\widetilde{G}$ is evident when it is compared with the scheme in BL90, where splitting is built in a different way. Here we use notation K

for G to avoid confusion. The approximated kernel $\widetilde{K}$ in BL90 is defined by

$$K_{i,j}^h - \widetilde{K}_{i,j}^h = \begin{cases} 0 & x_j = x_{2J} \\ K_{i,j}^h - I_{2h}^h K_{I,J}^{2h} I_h^{2h}, & \text{otherwise.} \end{cases} \tag{6}$$

$$\tau_i^h \approx \sum_{||\mathbf{x_j}-\mathbf{x_i}||\leq a} (K_{i,j}^h - \widetilde{K}_{i,j}^h) u_j^h + \sum_{||\mathbf{x_j}-\mathbf{x_i}||>a} (K_{i,j}^h - \widetilde{K}_{i,j}^h) u_j^h + I_{2h}^h \sum_J \widetilde{K}_{I,J}^{2h} u_J^{2h}. \tag{7}$$

Then, the cut-off radius a is decided beyond which the second term in (7) can be neglected so that finally

$$\tau_i^h \approx \sum_{||\mathbf{x_j}-\mathbf{x_i}||\leq a} (K_{i,j}^h - \widetilde{K}_{i,j}^h) u_j^h + I_{2h}^h \sum_J \widetilde{K}_{I,J}^{2h} u_J^{2h}, \tag{8}$$

to reduce the number of multiplications. In fact, the second term in (7) is zero by definition in the scheme of STH02, but it is not in BL90, which can lead to additional errors in some multilevel formulations. Eventually, the absence of the second multiplication term in STH02 can leads to an increased efficiency without any additional loss of accuracy in the multilevel approach. In preliminary 1D tests for the Chinnery kernel, we found that the resulting error is higher for the BL90 scheme. We additionally tested the BL90 scheme for the realistic simulation described in section 5.1 and the error was greater than that in the test with STH02 type scheme. Hence, the approach in STH02 is chosen for solving the multiplication problems in this work.

4. Details and results of application

Differently from Coulombic potential, the Chinnery kernel has a negative value at the source ($G < 0$ at $i = j$) and decreases as $1/r^3$ away from the source. Figure 2 shows the sharpness of the kernel G in 2D perspectives. The kernel decreases rapidly from a source so that the contribution from long-range interactions is relatively small. Figure 3 shows the center of cells on Ω^h and Ω^{2h} in our 2-level scheme. Computations on Ω^h are done for the interaction between the positions denoted by the black squares. To improve the efficiency, only interactions between the neighbouring cells are included. The neighbour size is determined by the cut-off radius a. We use a linear interpolation matrix I_h^{2h} to map the function on the fine grid onto the coarse grid. More details of the interpolation between the two levels are described

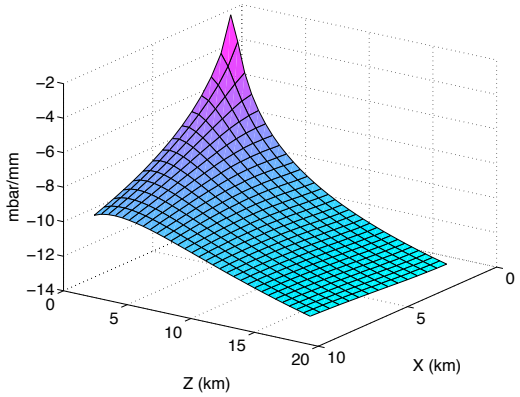

Figure 2: The Chinnery function in log-scale when the source is at $i = 1$. The values are shown here except at $i = j = 1$.

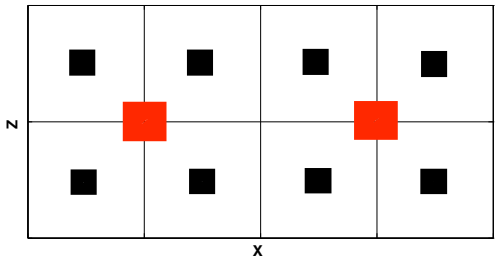

Figure 3: Schematic diagram of 2-level cell positions. Only a part of the fault segment is shown here. The black square represents center of cells on Ω^h and red square on Ω^{2h}.

in the Appendix. We obtain a *softened kernel* $\widetilde{G}$ on the basis of the second-order Taylor expansion of $1/r^3$, but modified coefficients to further smooth the function. There is also an option to use a higher order approximation. The approximation is, of course, not unique. We tested different formula for the second-order expansion and the one presented here is found to be optimal. The smoothed function is given by

$$\widetilde{G}(r) = \begin{cases} \frac{1}{(4a)^3}\left(-\frac{1}{16} - \left(\frac{r}{a}\right)^2 + 2\left(\frac{r}{a}\right)^4\right), & r \le a \\ G(\mathbf{x};\mathbf{x}'), & r > a \end{cases} \tag{9}$$

in this study. We need to implement this function in such way that $\sum_{i,j} G_{ij}$ is nearly conserved in the multilevel scheme. Figure 4 shows the Chinnery kernel (G) and the smoothed function ($\widetilde{G}$) when $a = L/64 = 1.0984$ km and the source and evaluation depth are at $z = 0.5492$ km. It is shown that the profile of the smoothed function survives nearly unchanged through interpolations. This is consistent with the well conserved sum of the kernel G

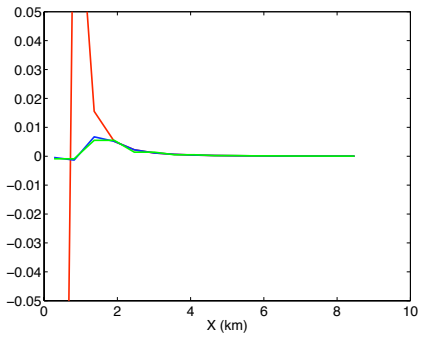

Figure 4: The distribution of the Chinnery function $G^h(r)$ (red), $\widetilde{G}^h(r)$ (blue), and $I^h_{2h}\widetilde{G}^{2h}I^{2h}_h(r)$ (green) when $a = L/64$ and $z = 0.5492$ km.

to a degree of machine accuracy for all cut-off radii tested here (see Table 1). This conservation property is important not to interfere with physical features of the fault model when we implement the multigrid method. In Table 1, comparisons about the sum are made between the 2-level multigrid scheme and the direct computation only within a cut-off radius. This direct computation differs from the original full direct multiplication in that it simply exclude the long-range interactions beyond the cut-off radius. It

Table 1: Total sum of the kernel (G) in the tests: $\sum_{l,m,n} G_{\text{test}}$, where l is an index of the depth of the evaluation, m of the source, and n is a horizontal index difference between them. It is -7.7423 for the original full direct multiplication ($\sum_{l,m,n} G$). For comparison, we show $|(\sum_{l,m,n} G_{\text{test}} - \sum_{l,m,n} G)/\sum_{l,m,n} G|$ in the 2-level multigrid scheme and the direct computations only within each cut-off radius.

a	Multigrid scheme	Direct computation within a
70 km ($a = L$)	1.15×10^{-16}	3.5×10^{-6}
35 km ($a = L/2$)	1.15×10^{-16}	1.6×10^{-3}
17.5 km ($a = L/4$)	6.88×10^{-16}	6.8×10^{-3}
8.75 km ($a = L/8$)	4.59×10^{-16}	2.28×10^{-2}
4.375 km ($a = L/16$)	4.59×10^{-16}	6.12×10^{-2}
2.1875 km ($a = L/32$)	4.59×10^{-16}	1.44×10^{-1}
1.0938 km ($a = L/64$)	9.18×10^{-16}	3.16×10^{-1}

is shown that physical features of the fault model could be interfered in the computations without the long-range interactions.

Since the kernel is not symmetric in the vertical direction we consider using a modified distance

$$r_m = ||\mathbf{x} - \mathbf{x}'||_m = \sqrt{(x - x')^2 + (z - z')^2/16}, \tag{10}$$

to include more vertical interaction into computations on the fine grids. We attempt to evaluate the performance of the multigrid scheme with the standard radius r and that with r_m. They are also compared with the direct computations within a. We use zero initial $\tau(\mathbf{x}_i)$, the slip $u(\mathbf{x}_i) = (\sin(x))^2$, which is dependent on horizontal direction only, and v_{pl}=0 for a computation of (1) (Test 1-1). We measure the performance by errors against the original full matrix-vector computation on the fine grids using

$$\text{ME} = \max_i(|\widetilde{\tau}_i - \tau_i|), \tag{11}$$

where the index $i = 1, 2, \cdots N$ to label cells in the domain, $\tau_i = \tau(\mathbf{x}_i)$ is from the full direct computation, and $\widetilde{\tau}_i = \widetilde{\tau}(\mathbf{x}_i)$ is from each test case. In table 2, we list ME in the three test cases. It is shown that errors are lower in the multigrid scheme than those in the partial direct computations. The use of r_m reduces ME only slightly. The results may suggest that the standard radius

Table 2: Test 1-1: Error comparisons between the multigrid scheme with the standard radius r, the scheme with the modified radius, r_m, and the direct computation (with r_m) only within a.

a_1	r	r_m	Computation within a
70 km ($a_1 = L$)	2.04×10^{-7}	1.62×10^{-7}	0
35 km ($a_1 = L/2$)	1.25×10^{-5}	8.96×10^{-6}	2.83×10^{-4}
17.5 km ($a_1 = L/4$)	1.36×10^{-4}	5.44×10^{-5}	1.5×10^{-3}
8.75 km ($a_1 = L/8$)	4.53×10^{-4}	3.53×10^{-4}	6.4×10^{-3}
4.375 km ($a_1 = L/16$)	1.3×10^{-3}	1.3×10^{-3}	2.17×10^{-2}
2.1875 km ($a_1 = L/32$)	3.3×10^{-3}	2.7×10^{-3}	5.32×10^{-2}
1.0938 km ($a_1 = L/64$)	1.38×10^{-2}	1.14×10^{-2}	1.25×10^{-1}

r could be a better choice for efficiency reason when the gain of accuracy does not increase significantly with the use of r_m.

Based on the error analysis in STH02 we expect the error in this test to be proportional to $(h/a)^p$, where p is the order of the approximation made on the smooth part of the kernel and here we use $p = 2$. Figure 5 shows the logarithm plot of ME in the multigrid test 1-1 as function of cut-off radius. This is compared to the theoretically expected error $C(h/a)^2$, where C is constant. By fitting this curve to the observed error in the test we found that convergence of the scheme behaves closely to the theoretical expectations when $C = 0.0598$. The error decreases slowly in the range of medium cut-off radius (around 10 km) and decreases similarly to the theoretical expectation for short cut-off radius. This error tendency implies that it might be optimal to choose a cut-off radius shorter than the medium cut-off radius for the problem in this study. We examine the spatial distribution of τ_i, $\widetilde{\tau}_i$, and $\widetilde{\tau}_i - \tau_i$ when $a = L/16 = 4.375$ km in the test 1-1 (Fig. 6). The error is relatively larger in the center of the domain, where $u(x)$ is largest. However, $\widetilde{\tau}_i$ is generally close to τ_i in most area in the plane. The results suggest that the multigrid computations with $a = L/16$ produce a solution with a reasonable accuracy. In the next section, we explain how a multiple time stepping is combined with the multigrid scheme and discuss the efficiency gain through the multigrid algorithm.

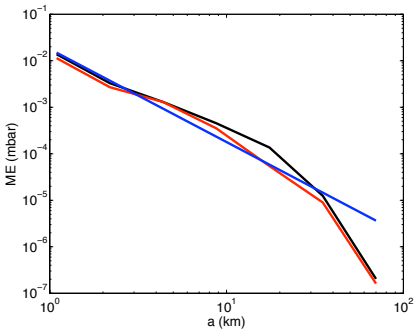

Figure 5: Test 1-1: Log-log plot for ME in the multigrid scheme as function of cut-off radius a: using standard r (black), modified radius r_m (red), and theoretical error (blue).

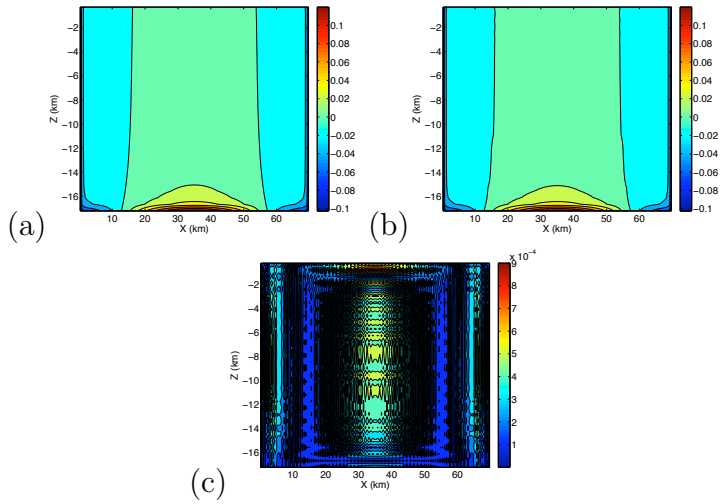

Figure 6: Test 1-1: (a) τ_i from the full direct computation, (b) $\widetilde{\tau}_i$ from the multigrid scheme with r, and (c) $\widetilde{\tau}_i - \tau_i$.

5. Multiple time stepping

Different from classical molecular dynamics, source and evaluation points are fixed in the earthquake modeling. This makes it easier to make the neighbour list. The list can be made once initially and used in further time stepping. We employ the approach suggested in Allen and Tildesley (1987), which is based on a Verlét algorithm. This algorithm enables each evaluation cell to have the list of neighbouring sources for the short-range interaction. Consequently, much less multiplication is needed for the short-range interaction on the fine grids. Also only this interaction is updated for each cell every time step, while the long-range interaction is updated less frequently. We choose to update short-range interactions twice as often as the long-range computations. In other words, if the full direct multiplication is originally updated every δt, then we update the short range interactions every δt, but the long-range interactions every $\Delta t = 2\delta t$.

We assume to have the slip u^n and the stress τ^n at a time step n. The long-range contribution to τ^n is approximated by $I_{2h}^h \sum_J \widetilde{G}_{I,J}^{2h} u_J^n$, which is updated only every Δt. Then the 2-level multiple time stepping Euler method can be expressed by

$$\begin{aligned}
u_h^{n+\frac{1}{2}} &= u_h^n + \frac{\Delta t}{2}(\tau^n)^3 \\
\tau^{n+\frac{1}{2}} &= \sum_{|\mathbf{x}-\mathbf{x}'|\le a} (G^h - \widetilde{G}^h) u_h^{n+\frac{1}{2}} + I_{2h}^h \sum_J \widetilde{G}_{I,J}^{2h} u_J^n \\
u_h^{n+1} &= u_h^{n+\frac{1}{2}} + \frac{\Delta t}{2}(\tau^{n+\frac{1}{2}})^3 \\
\tau^{n+1} &= \sum_{|\mathbf{x}-\mathbf{x}'|\le a} (G^h - \widetilde{G}^h) u_h^{n+1} + I_{2h}^h \sum_J \widetilde{G}_{I,J}^{2h} u_J^n \\
u_{2h}^{n+1} &= 4 I_h^{2h} u_h^{n+1}
\end{aligned} \tag{12}$$

Here the subscript or superscript, h and $2h$ denote the variables defined on the fine grids Ω_h and on the coarse grids Ω_{2h}, respectively. We use the Euler method in this study, but it is also possible to combine the multigrid scheme with a higher order method such as the 4th-order Runge-Kutta method.

We begin with testing the multigrid, multiple timestepping with τ perturbed randomly initially. We choose the cut-off radius $a = L/16 = 4.375$ km in all tests from now on. The efficiency is measured simply by the ratio of elapsed time of calculation (CPU time used by MATLAB) in the full direct

computation to that in multi-grid computations in MATLAB environment. However, we can also estimate the theoretical efficiency by

$$\text{Efficiency} \approx \frac{N^2 \times 2n}{N \times q \times 2n + (\frac{N}{4})^2 \times n} \tag{13}$$

where the number of neighbours q varies depending on the cut-off radius a. The number of cells on the grids Ω_{2h} is $N/4$, where N is the number of cells on Ω_h. The multiplication on the fine grids is updated $2n$ times, while on the coarse grids n times. This can be a theoretically expected maximum efficiency in the multigrid, multiple time stepping if we assume the interpolation and restriction processes add little computational cost. The theoretical efficiency is about 13.8 with the given parameters for the test 2-1 described in Table 3. Since errors are not significantly improved by using the modified radius r_m as discussed in section 4, we use the standard radius to achieve a higher efficiency in the test. The actual efficiency estimated by CPU time rate is 12.1 and this is not far from the theoretical efficiency 13.8. In table 3 we show that using multiple time stepping leads to a minor decrease in accuracy compared to the multigrid computation without multiple time stepping. We

Table 3: Description of test 2-1: given parameters and initial condition, c = $2.8451 \cdot 10^{-8}$, $u_0 = 0$, and τ_0 = [20 mbar, 90 mbar] with $\Delta t = 3/365$ yr and $n = 50$. Efficiency here is defined to be CPU time rate: [CPU time in direct computation]/[CPU time in multigrid scheme].

Test 2-1	Efficiency	ME
No multiple time stepping	8.8	3.18×10^{-2}
Multiple time stepping	12.1	4.13×10^{-2}

analyze the efficiency as well as the error tendency as function of cut-off radius. Figure 7 shows that the efficiency decreases relatively rapidly with the radius larger than $a = L/16 = 4.375$ km. Meanwhile the error decreases rather slowly with a at first, but rapidly beyond $a = L/4 = 17.5$ km. The error behaves similarly to that presented in Figure 5. These results suggest that the gain in the efficiency would diminish quickly beyond the radius a = 4.375 km, while the gain in the accuracy would increase with a rapidly if the cut-off radius becomes larger than $a = 17.5$ km. In consideration of

the trade-off issue between the efficiency and accuracy the choice of the cut-off radius $L/16$ might be appropriate in our computations with a tolerable relative error size of 0.001 in τ.

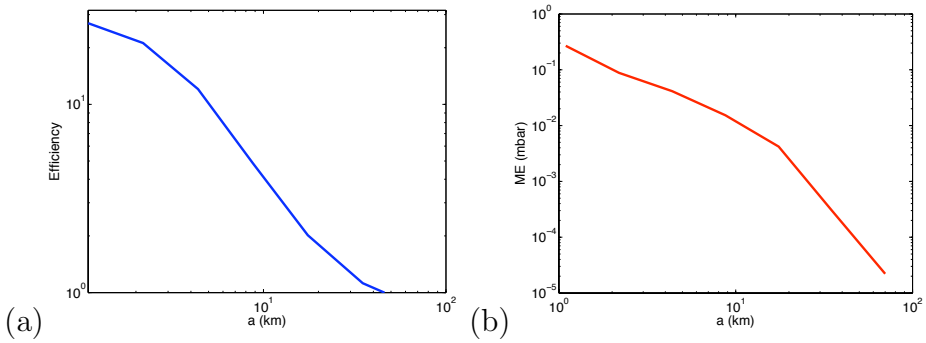

Figure 7: Test 2-1: Log-log plots for (a) Efficiency defined in Table 3 and (b) ME as function of a in multigrid, multiple time stepping simulations.

5.1. Test 2-2: realistic interseismic process

In this test we evaluate our multigrid, multiple time stepping algorithm by simulating a realistic interseismic process. We begin with the initial condition described in Table 4 and compare with the reference solution just before rupture occurs. Figure 8 shows that full direct and multigrid computations produce similar results to each other and to the reference. The error, the difference between the direct and multigrid computation, ranges between 0 and 0.06 for cut-off radius $L/16 = 4.375$ km (Fig. 8d). However, it decreases rapidly with increasing cut-off radius, and the multigrid solution converges to the solution from the full direct computation (not shown here). Table 4 shows the maximum error ME and efficiency when the multiple time stepping is used or not. The achieved efficiency in each case is similar to that shown in table 3 and the accuracy is not significantly affected by using multiple time stepping. The results from the tests show the potential of the multiple grid algorithm in an operational model for efficient simulations.

6. Conclusion

We develop a multigrid, multiple time stepping algorithm to efficiently simulate interseismic processes by reducing the complexity of direct computations in a simplified fault model. The reduction is achieved by computing

Table 4: Description of test 2-2: given parameters and initial condition, $c = c(x, z)$, $u_0 = 0$, and $\tau_0 \approx$ [50 mbar, 200 mbar] with $\Delta t = 0.01$ yr, $n = 78$. Cut-off radius $a = L/16 = 4.375$ km.

Test 2-2	Efficiency	ME
No multiple time stepping	8.8	5.39×10^{-2}
Multiple time stepping	12.5	6.55×10^{-2}

a multitude of long-range interactions on a coarse level and updating them less frequently. Computational speed-up for more realistic simulations may depend on specific implementation details of a model, but this study can provide a proof of concept that multigrid methods would be useful for efficient matrix-vector multiplications in earthquake modeling. In this work the gain in computing speed is about a factor of 10 with an accuracy to a reasonable degree. There are no clearly determined standards, but we consider a relative error of 0.001 to be reasonable; simulations including coseismic processes have shown that stress release is not disturbed by such an error. The conservation of the stress in the entire half-space is retained in the multigrid formulation, which is a clear advantage over direct computations within a cut-off radius. In our study we use the Euler method, but a higher order time stepping such as the 4th order Runge-Kutta method can be combined with the multigrid scheme. Our 2-level multigrid scheme can be also extended to more levels, which are not necessary here because the current work is applied to a relatively small domain. If the number of interacting cells would become larger, increasing the number of levels would be beneficial. The technique presented in this work is flexible and easy to implement. Therefore, it has a high potential to be also useful for the reduction of computational effort in other systems, in which spatial scales can be separated. Structural heterogeneities can be taken into account by changing the Green's function or by changing model parameters like the material strength as a function of space. The same holds for the implementation of additional faults leading to a growth of the interaction matrix. Such modifications will increase the computational effort, but the application of our method is nevertheless straightforward. Future work will examine the feasibility of our method with respect to those modifications and try a higher number of multi levels for the fault model with an increased complexity.

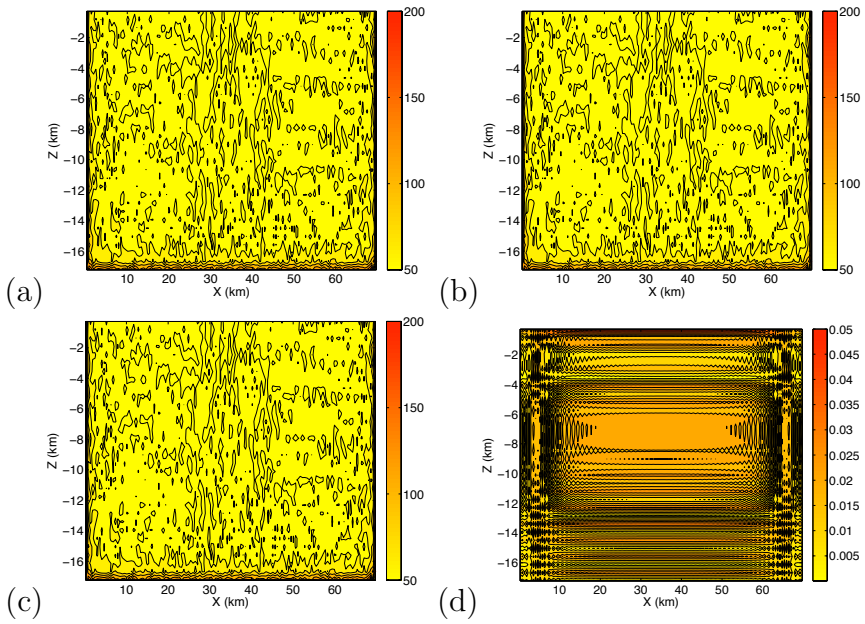

Figure 8: Test 2-2: (a) reference τ (mbar), (b) τ from the full direct computation, (c) τ from the multigrid, multiple time stepping, and (d) the difference between (b) and (c).

7. Acknowledgements

We thank the editor and two anonymous reviewers for helpful comments that improved the manuscript. This work was supported by the Collaborative research centre "Complex Nonlinear Processes" (SFB555) of the German Research Society (DFG) for financial support. Gert Zöller acknowledges partial support from DFG project HA4789/2-1.

Appendix. Algorithmic details

We build a two-level domain described in Figure 3. The level-1 grid points (black) corresponds to those in the original fault segmentation (Fig. 1). Then we calculate $G^h_{i,j}$ based on a routine for the Chinnery function. Along with this we also produce $\widetilde{G}^h_{i,j}$ using (9). Then we initialize u^h or τ^h from previous integration or using existing data. The displacement (u) on grids Ω^{2h} are defined using those on Ω^h in such way that:

$$u^{2h} \equiv 4 I_h^{2h}\, u^h. \tag{.1}$$

where we use the restriction operator I_h^{2h} for cell-centered discretization in such way that

$$\begin{aligned} I_h^{2h} A^h(x,y) = \frac{1}{4}[&A^h(x-\frac{h}{2}, y-\frac{h}{2}) + A^h(x-\frac{h}{2}, y+\frac{h}{2}) + \\ &A^h(x+\frac{h}{2}, y-\frac{h}{2}) + A^h(x+\frac{h}{2}, y+\frac{h}{2})] \end{aligned} \tag{.2}$$

This restriction operator transfers the data at the fine level (Ω^h) to the coarse level (Ω^{2h}).

Next we make a list of neighbourhood for every point $\mathbf{x_i}$ amongst $\mathbf{x_j}$. The size of a neighbour is given by

$$S = (a/h \cdot 2 + 1)^2 \tag{.3}$$

for each $\mathbf{x} = (x, y)$ so that the total length of the neighbours become $S \times$ length($\mathbf{x}$). This reduces the computation of $N \times N$ to $N \times S$, when N is the length of $\mathbf{x}$. We name a new kernel function $g(\mathbf{x_i}; \mathbf{x_j})$ for short-range interactions with neighbours. Components of $g(\mathbf{x_i}; \mathbf{x_j})$ are retrieved from $G(\mathbf{x_i}; \mathbf{x_j})$-$\widetilde{G}(\mathbf{x_i}; \mathbf{x_j})$ under the condition $|\mathbf{x_j} - \mathbf{x_i}| \leq a$. Likewise we define a

new displacement $v(\mathbf{x}_j, t)$ by collecting only the element in the neighbour list from the original vector $u(\mathbf{x}_j, t)$. Eventually, (5) is replaced by

$$\tau^h(\mathbf{x_i}) \approx \sum_{\mathbf{x_j} \in \text{ neighbour list of } \mathbf{x_i}} g(\mathbf{x_i}; \mathbf{x_j}) v(\mathbf{x_j}, t) + I_{2h}^h \sum_J \widetilde{G}_{I,J}^{2h} u_J^{2h}, \qquad (.4)$$

where I_{2h}^h is a *prolongation operator* (interpolation operator) that transfers data on the coarse grids to the fine grids and the method chosen here is taking the values on the nearest grid points in Ω_{2h} (see Fig. 3). Alternatively we can use a linear interpolation method for cell-centered discretization described in Trottenberg et al. (2007), which might lead to a slightly lower errors. However, we chose the nearest-point interpolation for reasons of efficiency since the gain in accuracy could be minimal for the problem under consideration.

Bibliography

Allen, M., Tildesley, D., 1987. Computer Simulation of Liquids. Clarendon Press, Oxford.

Aoi, S., Fujiwara, H., 1999. 3-d finite-difference method using discontinuous grids. Bull. Seismol. Soc. Am 89, 918–930.

Ben-Zion, Y., 1996. Stress, slip, and earthquakes in models of complex single-fault systems incorporating brittle and creep deformations. J. Geophys. Res. 101, 5677–5706.

Ben-Zion, Y., Rice, J. R., 1993. Earthquake failure sequences along a cellular fault zone in a three-dimensional elastic solid containing asperity and nonasperity regions. J. Geophys. Res. 98, 14109–14131.

Brandt, A., 1991. Multilevel computations of integral transforms and particle interactions with oscillatory kernel. Computer Physics Communcation 65, 24–38.

Brandt, A., Lubrecht, A. A., 1990. Multilevel matrix multiplication and fast solution of integral equations. Journal of Computational Physics 90, 348–370.

Brandt, A., Venner, C. H., 1998. Multilevel evaluation of integral transforms with asymptotically smooth kernels. SIAM Journal on Scientific Computing 19, 468–492.

Chaillat, S., Bonnet, M., Semblat, J.-F., 2008. A multi-level fast multipole bem for 3-d elastodynamics in the frequency domain. Computer Methods in Applied Mechanics and Enginnering 197, 4233–4249.

Chinnery, M., 1963. The stress changes that accompany strike-slip faulting. Bull. Seismol. Soc. Am 53, 921–932.

Greengard, L., Rokhlin, V., 1987. A fast algorithm for particle simulations. J. Comput. Phys. 73, 325–348.

Griebel, M., Knapek, S., Zumbusch, G., 2007. Numerical Simulation in Molecular Dynamics. Springer-Verlag, Berlin Heidelberg.

Hackbusch, W., Nowak, Z., 1989. On the fast matrix multiplication in the boundary element method by panel clustering. Numerische Mathematik 54, 463–491.

Okada, Y., 1992. Internal deformation due to shear and tensile faults in a half-space. Bull. Seismol. Soc. Am 82, 1018–1040.

Skeel, R., Tezcan, I., Hardy, D. J., 2002. Multiple grid methods for classical molecular dynamics. J. Comput. Chem. 23, 673–684.

Trottenberg, U., Oosterlee, C., Schüler, A., 2007. Multigrid. Elsevier Academic Press, London.

Wang, R., 1999. A simple orthonormalization method for the stable and efficient computation of green's functions. Bull. Seismol. Soc. Am 89, 733–741.

Zöller, G., Holschneider, M., Ben-Zion, Y., 2004. Quasi-static and quasidynamic modeling of earthquake failure at intermediate scales. Pure. Appl. Geophys 161, 2103–2118.

Zöller, G., Holschneider, M., Ben-Zion, Y., 2005. The role of heterogeneities as a tuning parameter of earthquake dynamics. Pure Appl Geophys 162, 1027–1049.