OptiDis: A parallel Fast Multipole Dislocation Dynamics code
Pierre Blanchard, Arnaud Etcheverry, Olivier Coulaud, Laurent Dupuy, Marc Bletry, Eric Darve

To cite this version:

HAL Id: hal-01095274
https://hal.archives-ouvertes.fr/hal-01095274
Submitted on 17 Dec 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
ABSTRACT

Among all the steps involved in DD simulations, the computation of the internal elastic forces and energy are by far the most resources consuming. However, since these are long-ranged and fast decreasing interactions, hierarchical algorithms like the Fast Multipole Method (FMM) are well suited for their fast evaluation.

The relatively low accuracy required for the interaction forces between dislocations brought us to develop a more efficient approximation method for the farfield. On the other hand, the nearfield interactions are still evaluated analytically, which required a rather performant implementation (AVX, GPU...).

Regarding parallelism, our code benefits from a hybrid OpenMP/MPI paradigm and a cache-conscious datastructure. Finally, an accurate handling of topological elements intersecting the structure of the octree was considered. The latter feature implied careful modifications of the P2M/L2P operators in order to deal with shared memory model of parallelism.

MODEL

The motion of dislocations is ruled by a prescribed mobility law (e.g. viscous glide) depending on the internal (i.e. created by dislocations themselves) and external forces (e.g. applied by user) contributions on the nodal forces.

The isotropic elastic stress field created by a dislocation loop (1/N) at point x in space is given by Mura’s formula [4]

\[ \sigma_{ij}(x) = \frac{1}{2\pi a} \int \left( (A_{ij}(y) + (A_{ij}(y)) + \frac{1}{r} - A_{ij}(y)) \right) \] 

where \((\mu, \nu)\) are the Lamé coefficients and

\[ (A_{ij}(y)) = \frac{1}{4\pi a} \int_\Gamma R_{ij}(x,y) y_i y_j d\Gamma \] 

where \(R_{ij}(x,y)\) are linear shape functions. While the isotropic elastic interaction energy is given by

\[ \frac{2\mu}{\rho} \int_\Omega \left( \nabla u(x) \cdot \nabla u(x) - \frac{1}{2} \nabla \cdot u(x) \right) \] 

The cost of updating the nodal forces and the energy at each time step is quadratic and involves the evaluation of 2 line integrals, therefore it is usually the bottleneck of DD simulations.

IMPLEMENTATION AND PERFORMANCES

The farfield approximation consists in 3 steps:

1. P2P Analytical evaluation of the nearfield.
2. P2M/L2P Integration of polynomial interpolators on all leaf cells return expansion:

\[ M_{ij} = \int \frac{1}{\rho} \left( \mathbf{S}_p \right)_{ij} d\mathbf{y} \] 

3. M2L/L2M Transfer expansions between levels.

The latter feature implied careful modifications of the P2M/L2P operators in order to deal with shared memory model of parallelism.

Both stress and energy expansions share the same M2L operator \(R_{ij}\), but the extra derivation involved in the stress field is applied to the interpolator. This results in a slightly lower accuracy but less M2L operations to be performed (see figure 2).

Acceleration by Fast Fourier Transform (FFT)

Even though the Chebyshev-based FMM [3] provides a very accurate algorithm it is very expensive in terms of memory and CPU time, mainly due to a large size of the vectors of the M2L step. Thus a new interpolation scheme based on an equispaced grid (Lagrange) was considered. In this approach \(R_{ij}\) is Tridiagonal since \(R_{ij} = R_{ij}(x-y)\). As illustrated below, \(R_{ij}\) can be seen as a convolution and applied in Fourier space with a linear cost.

\[ E(C', C) = \frac{1}{2\pi} \int_\Omega \int_\Omega \left( \frac{1}{\rho} \right)_{ij} dx dy \] 

The scheme remains stable and sufficiently accurate in the scope of DD simulations (min. accuracy on force computation \(\approx 10^{-3}\)), while dramatically decreasing the computational cost of the M2L, namely due to \(O(n^2)\) to \(O(n)\). Let \(n = 2^n\) then in double precision the memory required for the M2L equals:

\[ 316 \times (2^5 - 1)^2 \times 7 \times (2 \times 8) \approx 25 \text{ MBbytes} \]

REFERENCES

[1] Scaffm: software library to simulate large scale n-body interactions using the fast multipole method. Developed @ hepacs team, inria bordeaux.


FUNDINGS

This work was supported by the French ANR grants ANR-10-COSI-0011 and the associate team FastLA.

In collaboration with CEA Saclay.

More videos & infos →