

# Iterative 3D BEM solver on complex faults geometry using angular dislocation approach in heterogeneous, isotropic elastic whole or half-space

F. Maerten<sup>1,2</sup> & L. Maerten<sup>1</sup>

<sup>1</sup>*Igeoss, France*

<sup>2</sup>*University of Montpellier II, Dynamique de la Lithosphere, France*

## Abstract

Based on the analytical solution of the induced displacement caused by a 3D angular dislocation, it is possible to construct closed polygonal loops with constant Burgers's vector, from which the stress is derived using linear elasticity in homogeneous, isotropic whole- or half-space. In this BEM code, each fault is discretized as a triangulated mesh, where mixed boundary conditions are prescribed.

Incorporate material heterogeneity is done by using triangulated interfaces made of dual-elements with prescribed continuity and equilibrium conditions. Each interface and fault can therefore have a complex 3D geometry with no gaps or overlaps between elements.

We use an iterative solver where the system of equations is decomposed at the element level, allowing a simple formulation of the boundary conditions for elements making a fault, and continuity/equilibrium conditions at dual-elements making an interface. It is shown that strict diagonal dominance can be achieved only if continuity and equilibrium conditions, for a given dual-element, are solved simultaneously. Using a Gauss-Seidel-like method, we consequently reduce the complexity while automatically taking care of the sparsity of the system. Moreover, using a Jacobi-like solver, we show that the resolution of the system can simply be parallelized on multi-core processors. Some comparisons with a 2D analytical solution and a 2D BEM code are presented.

*Keywords: iterative solver, indirect method, heterogeneity, optimization, complex 3D geometry.*



## 1 Introduction

In structural geology and geophysics, fault slip distributions play an important role for the induced stress perturbation. This slip distribution depends mainly on the geometrical configuration and boundary conditions along the faults, the remote boundary conditions as well as the constitutive behavior of the host rock. In particular, it was shown that the change in Young's modulus of the rock can affect the slip distribution in a non negligible way [1]. Therefore, it appears that heterogeneity cannot be ignored.

In 2D, Crouch and Starfield [2] proposed a method to add material heterogeneity into the Displacement Discontinuity Method (DDM). In 3D, it is possible to apply this method using the Okada's code [3] where fault surfaces are discretized into planar rectangular elements. However, this formulation necessarily introduces non-physical gaps and overlaps between adjacent elements which can perturb the solution [4].

In this paper, we use the analytical solution of an angular-dislocation in elastic, homogeneous, isotropic whole- or half-space, where the stress is derived using linear elasticity in homogeneous, isotropic whole- or half-space [5]. A boundary element method is then formulated by discretization of all complex 3D faults into triangular elements. This BEM code (called Poly3D [6]) is very similar to DDM [2], in which triangular elements of constant displacement discontinuity are employed. The advantage compare to Okada's code is that three-dimensional fault surfaces more closely approximate curvilinear surfaces and curved tiplines without introducing overlaps or gaps. Such formulation is very well suited to study faults interaction in 3D, since only faults surfaces have to be discretized (see for example [7–9] among others).

The addition of heterogeneous and isotropic materials is presented in this paper, and particularly an iterative method for solving the system is investigated.

## 2 BEM formulation

The BEM formulation employed here is derived from the analytical solution of an angular dislocation in 3D elastic whole- or half-space [5]. A triangular dislocation (or more generally a polygonal dislocation) with constant displacement discontinuity, or Burgers's vector  $b$ , can be constructed [6, 10] simply by superposition of six angular-dislocations (see Fig. 1). Mixed boundary conditions (BC) are prescribed, and when Neumann BC are specified, we have to solve for the unknown Burgers's components. After the system is solved, it is possible to compute anywhere, within the whole- or half-space, displacement, strain or stress at observation points as a postprocess.

Incorporate material heterogeneity (or general piecewise inhomogeneous bodies) requires discretizing an interface between two regions of different material properties into two triangulated meshes, one for each region. These two surfaces have the particularity to be perfectly coalescent, but with opposite normals (see Fig. 2). According to [2], continuity conditions on one side and equilibrium



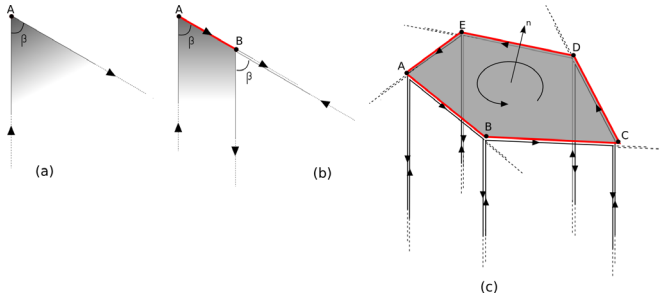


Figure 1: Construction of a polygonal element. (a) angular-dislocation, (b) a dislocation-segment made of two angular dislocations, and (c) a polygonal-element made of five dislocation segments of 10 angular-dislocations.

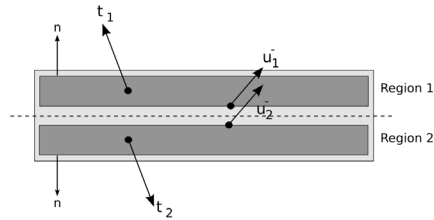


Figure 2: A dual-element part of an interface with equilibrium condition for tractions ( $t_1 = -t_2$ ) and continuity conditions for displacement ( $u_1^- = u_2^-$ ) on the negative side (in global coordinate system).

conditions on the other in element local coordinate system are applied [2] at interfaces (Fig. 2). Doing so requires particular attention for two main reasons. First, since we are in 3D, it is impossible to have two opposite local coordinate systems. Indeed, if  $x_i$  denotes the vectors of the first coordinate system ( $i \in [0..2]$ ) and  $X_i$  the second, we have  $X_0 = -x_0$ ,  $X_1 = -x_1$  and  $X_2 = X_0 \times X_1 = (-x_0) \times (-x_1) = x_0 \times x_1 = x_2 \neq -X_2$  (where  $\times$  denotes the vector product). Consequently, we have to express continuity conditions in global coordinate system. The second reason is that since the displacement is discontinuous when going from one side of an element to the other, we have to make sure that  $U_{ee}^-$ , the self displacement influence matrix at the element center on the negative side, is correctly computed. Due to the machine precision, it is not always guaranteed, and consequently, we force the element center to be on the negative side by applying an infinitesimal translation of the center along the reversed normal.

Therefore, equilibrium and continuity conditions can be written in global coordinate system as:

$$\begin{cases} \tau_e = -\tau_e \\ u_e^- = u_e^- \end{cases} \quad (1)$$

where polygonal elements  $e$  in region 1 and 2 are perfectly coincident (they are called dual elements in the remaining of the paper).

A global system of equations is then built, which incorporates both equilibrium, continuity and boundary conditions:

$$\begin{bmatrix} T_I^1 & T_F^1 & T_I^2 & T_F^2 \\ -D_I^1 & -D_F^1 & D_I^2 & D_F^2 \\ T_I^1 & T_F^1 & 0 & 0 \\ 0 & 0 & T_I^2 & T_F^2 \end{bmatrix} \begin{Bmatrix} b_I^1 \\ b_F^1 \\ b_I^2 \\ b_F^2 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ t_F^0 \\ t_F^0 \end{Bmatrix} \quad (2)$$

where  $I$  and  $F$  stands for elements at interfaces and faults respectively.  $D^-$  and  $T$  represent the displacement influence matrix on the negative side and the traction influence matrix respectively, both of them in global coordinate system. The two first rows define the equilibrium and continuity conditions at interfaces, while the last two represent the classical boundary conditions applied to fault surfaces, and  $t_F^0$  represents the initially prescribed traction boundary values for Neumann BC.

### 3 Iterative solver

In order to reduce the model complexity from  $O(n^3)$  to  $O(n^2)$  and to take advantage of the sparsity of the system (Eq. 2), an iterative solver is used.

The Burgers's vector solution at a regular boundary element  $e$  making a fault surface is given by:

$$b_e = T_{ee}^{-1} \left\{ t_e^0 - \sum_{f \neq e} T_{ef} b_f \right\} \quad (3)$$

where  $T_{ef}$  denotes the traction influence matrix at the centroid of element  $e$  due to element  $f$ .

Computing the solution for a dual-element part of an interface is tricky since we are dealing with an iterative solver for which the convergence is guaranteed if and only if the system is strict diagonal dominant [11], i.e.  $\forall i, \forall j \neq i, |a_{ii}| > |a_{ij}|$ . As an element  $e_1$  and its dual part  $e_2$  have the same geometry (only the orientation changes),  $T_{e_1 e_1}$  and  $T_{e_2 e_2}$  have the same diagonal values, and strict diagonal dominance is not honored (the same apply for  $D^-$  matrices). Therefore, equilibrium condition in region  $R_1$  and continuity condition in region  $R_2$  have to be solved simultaneously for a given dual-element. This leads to the following coupled elemental system:

$$\begin{Bmatrix} b_e^1 \\ b_e^2 \end{Bmatrix} = \begin{bmatrix} T_{ee}^1 & T_{ee}^2 \\ -D_{ee}^1 & D_{ee}^2 \end{bmatrix}^{-1} \begin{Bmatrix} -\sum_{f \neq e} T_{ef}^1 b_f - \sum_{g \neq e} T_{eg}^2 b_g \\ \sum_{f \neq e} D_{ef}^1 b_f - \sum_{g \neq e} D_{eg}^2 b_g \end{Bmatrix} \quad (4)$$

Eqs. (3) and (4) are then solved using a Gauss-Seidel procedure as follow:

```

while  $\epsilon >$  tolerance do
  for all element  $e$  in region  $R_1$ 
    if  $e$  is not part of an interface
      let  $t$  = initial traction vector of  $e$ 
      for all element  $f$  in region  $R_1$  except  $e$ 
         $t = t - T_{ef}b_f$ 
      end for
      Use Eq. (3)
      update  $\epsilon$  and set  $b$  to  $e$ 
    else
      let  $t = 0$  and  $u^- = 0$ 
      for all element  $f$  in region  $R_1$  except  $e$ 
         $t = t - T_{ef}b_f$  and  $u^- = u^- + D_{ef}^-b_f$ 
      end for
      for all element  $g$  in region  $\bar{R}_2$  except  $e$ 
         $t = t - T_{eg}b_g$  and  $u^- = u^- - D_{eg}^-b_g$ 
      end for
      Use Eq. (4)
      update  $\epsilon$  and set  $b^1$  to  $e$  and  $b^2$  to dual( $e$ )
    end if
  end for
end while

```

This algorithm automatically takes care of the sparsity of the system, while allowing large model computation since practically no memory allocation is required.

## 4 Results

In order to check the validity of the formulation, we first compare the 2D analytical solution [2] of an annulus (with  $\nu_1 = 0.25, G_1 = 1$ ) inside a circular hole in a large plate ( $\nu_2 = 0.25, G_2 = 0.5$ ) (Fig. 3(b)), with a corresponding 3D model (Fig. 3(a)). The hole is subjected to an internal pressure of 0.001 Pa, and the computed normalized  $\sigma_3$  and  $\sigma_1$  along the bold lines are compared. In order to avoid boundary effects, the tubes, defining the annulus and the hole in 2D, are extended far from the zone of interest, materialized by the squared observation plane (Fig. 3(a)). It can be seen that the computed values along the bold line in Fig. 3(a) match the analytical solution.

Comparison with an existing 2D BEM code (Fig. 4) is described in [1]. The model is composed of a penetrating fault inside an inclusion, and various configurations are computed where the ratio of the Young's modulus in the inclusion to that of the host material, was set to 1, 0.1 and 10. It is subjected to a uniform remote unit shear stress  $\sigma_{yx}$  of 1 MPa. The corresponding 3D model is composed of two orthogonal vertical planar surfaces, one for the interface defining the inclusion and



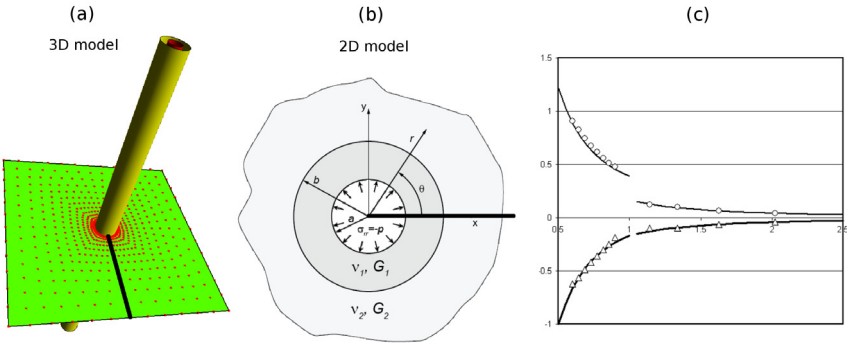


Figure 3: Comparison with a 2D analytical solution. (a) 3D model configuration, (b) 2D model, (c) comparison of normalized  $\sigma_3$  (triangles) and  $\sigma_1$  (circles), where solid lines are for the numerical solutions.

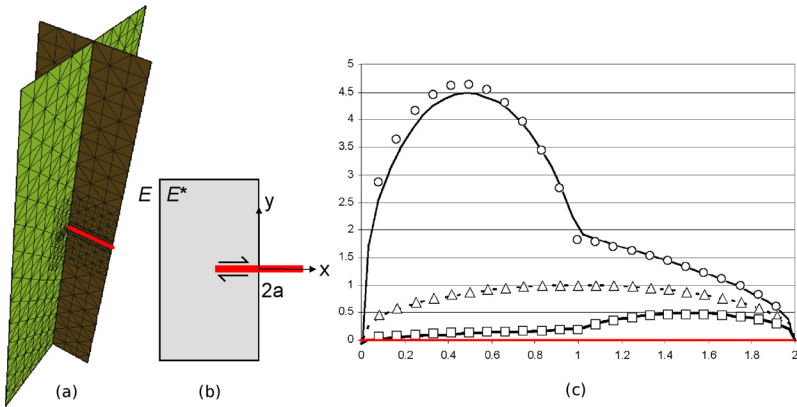


Figure 4: Comparison with a 2D BEM code. (a) 3D model configuration, (b) 2D model, (c) comparison of the computed dip-slip component onto the fault (along the bold line in (a) and (b)). Triangles is for  $E^*/E = 1$ , circle for  $E^*/E = 0.1$ , and squares for  $E^*/E = 10$ .

the other for the penetrating fault. The other five sides of the box inclusion are not taken into account. The bold lines in Fig. 4(a) and 4(b) represents the x-axis of Fig. 4(c). Computed dip-slip component are reported onto the Y-axis. Again, the results match the 2D BEM solutions.

## 5 Optimizations

Even if this iterative algorithm is in  $O(n^2)$ , it is slower than the direct solver, since at each iteration and for each element, the influence matrices (traction and displacement) due to all other elements have to be re-calculated.



Table 1: Speed comparison (in seconds) for different model sizes, performed on Intel Xeon 2Ghz with 8 cores running Linux Ubuntu. “LU” is the classical direct solver of the whole system, “Iter” is the iterative solver with no memory allocation, “Alloc” is the iterative solver with memory allocation, and “Thread” is the parallelized version on 2 or 8 threads. Models marked “x” where stopped before the end.

Model size	LU	Iter	Alloc	Thread 2	Thread 8
2200	49	380	9	8	6
4200	466	1018	46	26	8
9000	1145	5600	205	116	32
16000	x	x	1200	690	205

### 5.1 Bufferized elemental matrices

We can highly increase the speed if we store elemental matrices  $T_{ef}$ ,  $D_{ef}^-$ ,  $T_{eg}$  and  $D_{eg}^-$  from Eqs. (3) and (4) for each element  $e$ . Of course, if we reach the maximum available RAM (Random Access Memory), the remaining matrices have to be computed “on the fly” in order to avoid memory “swapping”.

Furthermore, since the global system is never constructed nor inverted, there is no cumulative roundoff error when using elemental matrices with floating precision (4 bytes) instead double (8 bytes), allowing larger models to be computed.

Table 1 gives some examples of the gain of speed using a direct LU solver and the iterative solver with and without allocation.

### 5.2 Parallelization on multi-core processors

The iterative solver allows taking advantage of the new multi-core processors architecture, by parallelizing the computation onto different threads using the cross-platform package [12] (where the number of threads  $k$  is defined by number of processors on the mother board times the number of cores for each processor). At the beginning, the system is split into  $k$  sub-systems (the decomposition technique is irrelevant), one for each thread. Consequently,  $k$  iterative solvers are run in parallel. There is no need to update the communication between each sub-process at the end of each iteration since they share the same model memory. The only constraint is to update the new displacement onto the elements at the end of each iteration by waiting for each thread to finish its job in order to avoid read/write conflicts. This is simply achieved by using a Jacobi procedure instead of the Gauss-Seidel one. Table 1 gives an overview of the computation time for  $k = 2$  and  $k = 8$ .

## 6 Conclusions

Indirect BEM techniques appear to be an advantageous way of modeling stress perturbation around faulted area in whole- or half-space since only the fault



surfaces have to be discretized as boundary elements. Implementing material heterogeneity requires little effort and the use of an iterative solver have a great impact in terms of speed for the system resolution and memory consumption. The Jacobi version of the iterative solver also permits parallelization on multi-core processors in a simple and efficient way.

Since each dual element generates six unknowns into the system, it is necessary to reduce the complexity, which is part of our current research.

## References

- [1] Burgmann, R. & Pollard, D.D., Inverting for slip on three-dimensional fault surfaces using angular dislocations. *Journal of Structural Geology*, **16**(12), pp. 1675–1690, 1994.
- [2] Crouch, S.L. & Starfield, A.M., (eds.) *Boundary element methods in solid mechanics*. Unwin Hyman: London, 1983.
- [3] Okada, Y., Surface deformation due to shear and tensile faults in a half-space. *Bulletin of the Seismological Society of America*, **75**, pp. 1135–1154, 1985.
- [4] Maerten, F., Resor, P.G., Pollard, D.D. & Maerten, L., Inverting for slip on three-dimensional fault surfaces using angular dislocations. *Bulletin of the Seismological Society of America*, **95**, pp. 1654–1665, 2005.
- [5] Comninou, M. & J., D., The angular dislocation in a half space. *Journal of Elasticity*, **5**(3), pp. 203–216, 1975.
- [6] Thomas, A.L., *Poly3d: a three-dimensional, polygonal element, displacement discontinuity boundary element computer program with applications to fractures, faults, and cavities in the earth's crust*. Master's thesis, Stanford University, 1993.
- [7] Maerten, L., Willemse, E.J.M., Pollard, D.D. & Rawnsley, K., Slip distributions on intersecting normal faults. *Journal of Structural Geology*, **21**, pp. 259–271, 1999.
- [8] Maerten, L., Variation in slip on intersectiong normal faults: Implications for paleostress inversion. *Journal of Geophysical Research*, **105**(25), pp. 553–565, 2000.
- [9] Muller, J.R., Aydin, A. & Maerten, F., Investigating the transition between the 1967 mudurnu valley and 1999 izmit earthquakes along the north anatolian fault with static stress changes. *Geophysics Journal International*, **154**, pp. 471–482, 2003.
- [10] Jeyakamuran, M., Rudnicki, J.W. & Keer, L.M., Modeling slip zones with triangular dislocation elements. *Bulletin of the Seismological Society of America*, **82**, pp. 2153–2169, 1992.
- [11] Golub, G.H. & Loan, C.F.V., (eds.) *Matrix computation*. ohns Hopkins Studies in the Mathematical Sciences, 1996.
- [12] Trolltech, Qtconcurrent, <http://www.trolltech.com>.

