**Practical for the Advanced Workshop on Earthquake Fault Mechanics: Theory, Simulation and Observation (2-14 September 2019, Trieste, Italy)**

**Lecture: Dynamic Rupture Simulation Methods**

By Dr. Luis A. Dalguer (luis.dalguer@alumni.ethz.ch), 3Q-Lab GmbH, Switzerland

## Objective:

The idea of this practical is to the student understand conceptually the implementation, in a numerical technique, the mixed boundary condition in an elastodynamic problem coupled to a friction law, that is used for earthquake rupture dynamic. You are not going to code. What you are going to do is to take a peace of paper and write and develop your own implementation. And then, during this workshop or after you can take your implementation to code and test it with the problem given at the end of this document.
Attached to this document is a matlab script file "Friction_1Dwave.m" (and also in appendix) and a movie that content the solution of the given (file: dx25.avi).

**Brief introduction on dynamic rupture model concept.**
First lets define kinematics and dynamics: ***Kinematics*** is the branch of mechanics that deals purely with motion, without analyzing the underlying forces that cause or participate in the motion. ***Dynamics*** is the branch of mechanic that deals directly with force systems, and with the energy balance that governs motion (Aki and Richard, 2002. Box. 5.3, page 129).

The main difference between these two models is the way in which the rupture discontinuity on the fault is modeled. The ***kinematic*** model associates the earthquake with prescribed fault slip (as a function of position and time) without taking into account the physics involved in the rupture. While the ***dynamic*** approach is investigates the physical processes involved in the fault rupture, incorporating conservation laws of continuum mechanics, constitutive behavior of rocks under interface sliding, and state of stress in the crust. The fault kinematics (slip) is determined dynamically as part of the solution itself, by solving, for example, the elastodynamic equation coupled to frictional siding.

***Dynamic*** models can be described as a two-step process: (1) formation of shear crack and (2) propagation or growth of the crack. The crack tip serves as a stress concentrator (Fig. 1) due to a driving force (for example, tectonic loading). If the stress at the crack tip exceeds some critical value then the crack grows unstably accompanied by a sudden slip and stress drops. Once a fault has been formed its further motion is controlled by friction sliding. Friction is the resistance to motion that occurs when a body slides tangentially to a surface on which it contacts another body.

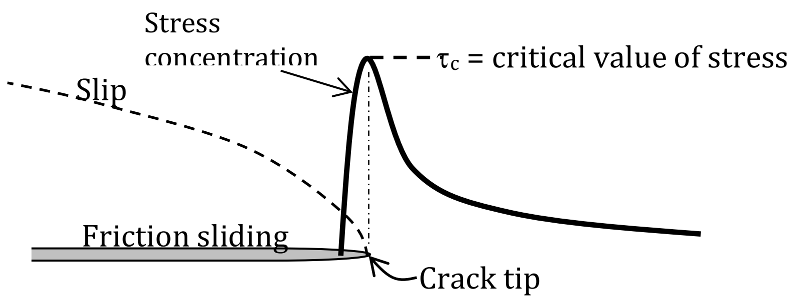


**Figure 1.** Idealized model of a crack rupture with stress concentration in the crack tip and slip in the friction sliding zone.

## Questions related to the physics of rupture

Dynamic models have greater potential for addressing science questions of earthquake rupture phenomena:

- *What are the physical bases for earthquake initiation and how rupture nucleates?*

- *How fast does the rupture propagate during earthquake?*

- *Why and how earthquake rupture stop?*

- *What are the physical basis for earthquake rupturing the free-surface?*

- *How rupture operates during an earthquake?*

Beside these questions, dynamic rupture models have also greater potential to study ground motion dominated by the source, for example:

-*The effects of surface and buried rupture on ground motion;*

-*Directivity pulses due to subshear and supershear rupture;*

-*The physical limits on extreme ground motion.*

-*The physical description of low and high frequency ground motion radiated from the fault.*

## Split node fault representation

### Mathematical Formulation of the problem in an idealized one point fault

Let assume the fault plane is perpendicular to the $z$ axis and located at $z$=0. To simplify the problem, we will implement the mixed boundary condition in a 1D wave equation, so all the fields depend only on $z$. This reduces to the condition that exactly the same thing is happening at every points along an infinitely fault plane.

Let use the velocity-stress form of the elastodynamic equations, in which the velocity $v(z,t)$ and shear stress $\tau(z,t)$ are the dependent variables:

$$\frac{\partial v}{\partial t} = \frac{1}{\rho}\frac{\partial \tau}{\partial z} \qquad [1]$$

$$\frac{\partial \tau}{\partial t} = \mu\frac{\partial v}{\partial z} \qquad [2]$$

Where $\mu$ is the shear module and $\rho$ the density.

Let use the standard staggered grid finite difference for the spatial discretization of the equation (Figure 2). The fault normal is in the z direction and located at z=0.

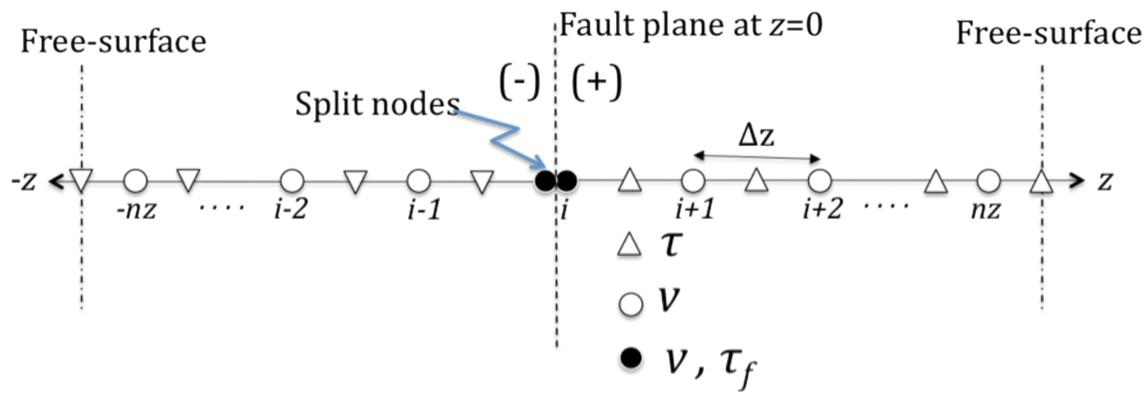


**Figure 2.** Staggered-grid discretization of the 1D elastodynamic equation with grid cells (split nodes) adjacent to the fault plane.

Rupture of the fault leads to a true discontinuity of velocity and displacement at z=0. Defining negative and positive sides of the fault (Figure 1). Slip velocity $\Delta v$, defined as the tangential velocity discontinuity, is the velocity of the positive side relative to negative side at $z$=0

$$\Delta v = v^{+} - v^{-} \qquad [3]$$

Which can be integrated to obtain slip.

This velocity discontinuity leads to a stress boundary value problem in which the shear traction $\tau_f$ that acts at the frictional interface (fault) during rupture is conditioned to follow a constitutive law. The shear traction $\tau_f$ on the fault, that is exerted by the positive side upon the negative side, is opposed by an antiparallel shear traction, i.e., the negative side exerts traction -$\tau_f$ on the positive side. The magnitude of $\tau_f$ is bounded by the frictional strength $\tau_c$

$$\tau_c - \left|\tau_f\right| \geq 0 \qquad [4]$$

The frictional strength $\tau_c$ is assumed to be proportional to normal stress $\sigma_n$

$$\tau_c = \mu_f \sigma_n \qquad [5]$$

Where $\mu_f$ is the coefficient of friction that can follow any friction law.

**What to do?**

The best practice to understand how this problem works in a numerical model, is developing by yourself the formulation and coding it. Then take a piece of paper, a pencil and first develop your own implementation:

*1) Approximate the spatial derivatives of equations 1 at the split nodes*

To do that, write separate equations for each side of the fault, taking into account the shear traction $\tau_f$ acting at the interface, and its initial static equilibrium value $\tau_0$. Introduce the following one-sided difference approximations for $\tau$, applicable to the plus and minus sides of the fault, respectively.

$$\left(\frac{\partial \tau}{\partial z}\right)_i^{\pm} = \pm \frac{\left[\tau_{i\pm 1/2} - (\tau_f - \tau_0)\right]}{0.5\Delta z} \qquad [6]$$

The time derivatives of equation 1 at time *t* approximates by second-order central differences:

$$\left(\frac{\partial v}{\partial t}\right)_t = \frac{v(t+\Delta t/2) - v(t-\Delta t/2)}{\Delta t} \qquad [7]$$

Where *Δt* is the time step.

*2) Find the slip and slip velocity considering the boundary conditions 4 and 5.*

To do that, evaluate $\tau_f$ to satisfy equation (4) and couple with the friction law given by equation (5). The most simple way to couple the wave equation with the friction models is satisfying the following two conditions:

for $\Delta v = 0$ then $|\tau_f| < \tau_c$ [8]

for $\Delta v \neq 0$ then $|\tau_f| = \tau_c$ [9]

When slip velocity $\Delta v=0$, there is continuity of tangential velocity, i.e,

$v^+ - v^- = 0$ [10]

Using this condition, find the shear traction (lets call trial traction $\tau^*$) required to satisfy eq. 10.

If $\tau^*$ satisfy eq. (8) then $\tau_f=\tau^*$, if not, then $\tau_f=\tau_c$.

*3) Approximate the spatial derivatives of equations 1 and 2 at interior grid points.*

To do that, approximate the derivatives with a second-order spatial difference

$$\left(\frac{\partial \phi}{\partial z}\right)_i = \frac{\phi_{i+1/2} - \phi_{i-1/2}}{\Delta z} \qquad [11]$$

where ϕ represents an arbitrary stress $\tau$ or velocity component $v$

4) Approximate the free-surface boundary condition positioning the free-surface at the stress node (see figure 2) and set stress at this node to zero

$$\tau_{nz+1/2} = 0$$
$$\tau_{-nz-1/2} = 0 \qquad [12]$$

5) If you finished it!!, then you can code it in any programming language you like (fortran, matlab, c, c++, etc…). If you do not finish now, you can do it during the conference or at home, and contact to me if you have problems… ;-)

For $\mu_f$ (the coefficient of friction in eq. 5) you can use the slip weakening model given as follow:

$$\mu_f = \begin{cases} \mu_s - (\mu_s - \mu_d)\dfrac{\Delta u}{d_0} & \Delta u < d_0 \\ \mu_d & \Delta u \geq d_0 \end{cases} \qquad [12]$$

Where $\Delta u$ is the slip, $\mu_s$ is the static friction coefficient, $\mu_d$ the dynamic friction coefficient, $d_0$ the critical slip distance.

If you succeed in 1D, you can also do in 2D and 3D!!!

## Test problem

Lets assume the fault is in the interface of two materials, plus side and minus side of the fault. We will let the fault break, i.e., one side of the fault will slide relative to the other side. Basically what we are going to solve is:

1) Evolution in time of slip velocity, slip and stress on the fault

2) wave radiated from the fault toward the free-surface. Due to different material properties, we will observe wave propagating at different speeds

3) The effect of the free-surface on the radiated wave.

***Model geometry:***
L = 5000m; domain size (m) on each side of the fault
$\Delta z$ = 25m; grid size

***Material properties:***
$c^+$ = 4000.0 m/s; wave speed plus side of the fault
$\rho^+$ = 2670.0kg/m^3; density plus side of the fault

$\mu^+ = \rho^+(c^+)^2$ shear module plus side of the fault (Pa)

$c^-$ = 2000.0 m/s; wave speed minus side of the fault
$\rho^-$ = 2670.0kg/m^3; density minus side of the fault

$\mu^- = \rho^-(c^-)^2$ shear module minus side of the fault (Pa)

***Friction and initial stress:***
$\sigma_n$= 120e6 Pa; initial normal stress on the fault
$\mu_s$ =0.677; static friction coefficient
$\mu_d$ =0.525; dynamic friction coefficient
$\tau_0$ =82.0e6 Pa; initial shear stress
d0=0.4 m; critical slip distance (m)

***Simulation time***
tmax = 1.5*L/$c^+$;

***Time discretization*** (CFL=0.5)
dt = 0.5*$\Delta z/c^+$; time step
nt = integer(tmax/dt)+1; Number of time steps

***Spatial discretization***
nz = ingeger(L/$\Delta z$)+1; Number of grid points

### Suggestions for other tests:

You can play with the grid size to evaluate convergence and numerical oscillations. Use the same data above, but for $\Delta z$=10m, 50m, 100m 200m 400m.

**APPENDIX** (matlab script). You can also find in the attached file "Friction_1Dwave.m"

```
% Practical for the Advanced Workshop on Earthquake Fault Mechanics:
% Theory, Simulation and Observation (2-14 September 2019, Trieste, Italy)
%
% Numerical implementation of Elastodynamic equation coupled to friction
law for
% rupture dynamic problems%
% solved in a 1D wave equation with staggered grid scheme
% and slip weakening friction law boundary conditions
%
% (the description of the problem is in a pdf file
"Practical_Rupture_dnamic.pdf"
%
%%% By Luis A. Dalguer (luis.dalguer@alumni.ethz.ch)
%%% 3Q-Lab GmbH, Switzerland
%
%===============================================================
clear;
% movie? T or F
movie = 'F';
movfile = 'dx25.avi'; % file name for movie
%model geometry
L = 5000;% domain size (m) on each size of te fault
h =  25; % grid spacing (m)

% Material properties: wave speed, density, modulus
% plus side of the fault
c_plus = 4000.0; %wave speed (m/s)
rho_plus = 2670 ; % density (kg/m3)
mu_plus = rho_plus*c_plus^2; % shear modulus (Pa)

% minus side of the fault
c_minus = 2000.0; %wave speed (m/s)
rho_minus = 2670 ; % density (kg/m3)
mu_minus = rho_minus*c_minus^2; % shear modulus (Pa)


% friction parameterization
sigma= 120e6; %normal stress (Pa)
mus=0.677; %dynamic friction coefficient
mud=0.525; %static friction coefficient
T0=82e6; %initial stress (Pa)
d0=0.4; %critical slip distance (m)

Ts=sigma*mus; %static frictionnal strength (Pa)
Td=sigma*mud; %dynamic frictionnal strength (Pa)

% simulation time
tmax = 1.5*L/max(c_plus, c_minus);

% time discretization (CFL=0.5)
  dt = 0.5*h/max(c_plus, c_minus); %time step
  nt = round(tmax/dt)+1; % Number of time steps
  t = linspace(0,tmax,nt); % time vector

% spatial discretization
```

```
  nz = round(L/h)+1; % Number of grid points
  zmin = -L; zmax = L; z0=0;

  zv_plus = linspace(z0,zmax,nz); % velocity points plus side
of the fault
  zv_minus = linspace(z0,zmin,nz); % velocity points minus
side of the fault

  zs_plus = zv_plus+0.5*h; % stress points plus side of the
fault
  zs_minus = zv_minus-0.5*h; % stress points minus side of the
fault


  % set initial conditions to velocities and stresses

  v_plus = zeros(1,nz);
  v_minus = zeros(1,nz);

  s_plus = zeros(1,nz);
  s_minus = zeros(1,nz);

   D=zeros(1,nt); %slip
   V=zeros(1,nt); % slip rate
   Ds=0;
   Vs=0;
scrsz = get(0,'ScreenSize');
%figure('Position',[scrsz(4)/10 scrsz(4)*0.8 scrsz(3)*0.8
scrsz(4)*0.8]);
figure('Position',[1 1 scrsz(3)*0.6 scrsz(4)*1.3]);
%figure('Position',[1 scrsz(4) scrsz(3)*0.8 scrsz(4)*0.8]);

if movie =='T'
     mov = avifile(movfile);
  end



  % loop over time steps
  for it=1:nt

  %Trial traction. This equation was formulated solving Eq.
(1) for the two sides of the fault
  % assuming continuity of the tangential velocity with Eq.
(6) (7) and (10) ((i.e, for a locked fault)

  trial(it) = T0+ (s_plus(1)*rho_minus+s_minus(1)*rho_plus
+Vs*h*rho_plus*rho_minus/(2*dt))/(rho_plus+rho_minus);

  T (it)= trial(it); %Assume that traction T is equal to trial
(the fault is locked)

  % Evaluate friction (Slip weakening friction model);
  fric=Td;
```

```
  if Ds < d0
  fric=(d0-Ds)*(Ts-Td)/d0 + Td; %( from eq. (5) and (12)
  end

  % update T if trial > fric to satisfay eq. (4)
  if abs(trial(it)) > fric
      T(it) = sign(trial(it))*fric;
  end

    % update velocities at t+dt/2 from stresses at t
    % at interior grid points, with a second-order spatial
difference
    % Solving eq(1), using Eq. (7) and (11)
    v_plus(2:nz) = v_plus(2:nz)+dt*(s_plus(2:nz)-s_plus(1:nz-
1))/(rho_plus*h);
    v_minus(2:nz) = v_minus(2:nz)+dt*(s_minus(1:nz-1)-
s_minus(2:nz))/(rho_minus*h);

    % at split nodes: %Solving eq(1), with Eq. (7) and (6)
    v_plus(1) = v_plus(1)+dt*(s_plus(1)-(T(it)-
T0))/(0.5*rho_plus*h);
    v_minus(1)= v_minus(1)-dt*(s_minus(1)-(T(it)-
T0))/(0.5*rho_minus*h);

    Vs = v_plus(1)-v_minus(1);% slip velocity (Eq.3)
    V(it) = Vs;% slip velocity
    Ds=Ds+V(it)*dt; % slip
    D(it)=Ds; % slip

    % update stresses at t+dt from velocities at t+dt/2
    % at interior grid points with a second-order spatial
difference
    % Solving eq(2), using Eq. (7) (but for stress) and
Eq.(11)
    s_plus(1:nz-1) = s_plus(1:nz-1)+dt*mu_plus*(v_plus(2:nz)-
v_plus(1:nz-1))/h;
    s_minus(1:nz-1) = s_minus(1:nz-
1)+dt*mu_minus*(v_minus(1:nz-1)-v_minus(2:nz))/h;

    % free-surface boundary condition (Eq. 12)
    s_plus(nz) = 0;
    s_minus(nz) = 0;

    % plot current solution

      subplot(4,1,1),
plot(zv_plus,v_plus,'r',zv_minus,v_minus,'b','LineWidth',2)
      set(gca,'fontsize',15)
      grid
      xlabel('z'), %ylabel('v(m/s)')
      xlim([min(zv_minus) max(zv_plus)])
      ylim([-4.0 4.0])
      %axis([min(zv_minus), max(zv_plus),min(min(vn)),
max(max(vn))])
```

```
      title(['Solution of velocity (m/s) at time t =
',num2str(t(it),'%6.4f')])

      subplot(4,1,2),
plot(zs_plus,s_plus/1e6,'r',zs_minus,s_minus/1e6,'b','LineWidt
h',2)
      set(gca,'fontsize',15)
      grid
      xlabel('z'), %ylabel('stress(MPa)') %, \sigma(x)')
      xlim([min(zs_minus) max(zs_plus)])
      ylim([-22 2])
      %axis([min(xv), max(xs),min(min(sn)), max(max(sn))])
      title(['Solution of stress change (MPa) at time t =
',num2str(t(it),'%6.4f')])

      subplot(2,3,4), plot(t(1:it),V(1:it),'LineWidth',2)
      set(gca,'fontsize',15)
      xlabel('Time (s)')
      ylim([0 6])
      xlim([0 tmax])
      title(['Slip velocity (m/s)'])
      axis square

      subplot(2,3,5), plot(D(1:it),T(1:it)/1e6,'LineWidth',2)
      set(gca,'fontsize',15)
      xlabel('Slip (m)')
      ylim([62 82])
      xlim([0 9.5])

      title(['Shear traction (MPa) vs Slip (m)'])
      axis square

%       subplot(2,3,6),
plot(t(1:it),T(1:it),'k',t(1:it),trial(1:it),'r')
      subplot(2,3,6), plot(t(1:it),T(1:it)/1e6,'LineWidth',2)
      set(gca,'fontsize',15)
       xlabel('Time (s)')

      ylim([62 82])
      xlim([0 tmax])
      title(['Shear Traction (Mpa)'])
      axis square

      pause(1e-6)

     if movie =='T'

      set(gcf,'Nextplot','replace')
      F=getframe(gcf);
      mov = addframe(mov,F);

     end

  end
```

```
if movie =='T'
   mov = close(mov);
end
```