Computer Simulation of Stress Waves in Composite Rods

Li, Hongying ID:4995945

A Project

in

The Department

of

Mechanical and Industrial Engineering

Presented in Partial Fulfillment of the Requirements

for ENGR 6971

Concordia University

Instructor: Dr. Mamoun Medraj

August 2003

Table of Contents

Computer Simulation of Stress Waves in Composite Rods

Li, Hongying

Student No. 4995945

(Li_Hongying@Hotmail.com)

Abstract

The theory of stress wave is introduced, recapitulating and illustrating the most important formulas. The wave model for a uniform bar is derived on the basis of the former theory. An algorithm and a computer code were developed to simulate the propagation of an elastic compressive wave along a uniform bar. The stress history through the length of the bar is depicted with MATLAB. The results are explained and it is concluded that the program can be used to predict the fracture due to the transmission of the stress wave or due to its reflection in terms of compressive stress or tensile stress.

Keywords: Composite materials; Numerical analysis; Stress waves;

1. Introduction

There are two important theories which have something to do with the stress wave are reviewed here first. On the one hand, in rigid dynamics, when a force is applied to any one point on a body, it is assumed the resultant stresses set every other point in motion instantaneously, and by Newton's second law of motion, the force can be considered as creating a linear acceleration of the whole body. On the other hand, in the theory of elasticity, the body is considered as in equilibrium under the action of applied forces, and the stress-strain relation is linear, i.e. the elastic deformations are sufficiently accurate for problems in which the time is relatively long between the application of a force and the time in which the observations are made. However, when we are considering the forces which are applied for only very short periods of time, or are changing fast, its effects must be considered in terms of the propagation of stress waves[1].

There have been a lot of investigations on dynamical stress wave. The methods to study them are usually experimentally and theoretically. However, due to the limitation in computing, it was not possible to derive analytical solutions for many years. Fortunately, with the development of computer science, it is possible to perform the time-consuming numerical calculation so that more accurate results can be got easily now.

In this project, the theory of stress wave is reviewed first and then the wave equation for a uniform rod is derived. Finally, some numerical calculation examples are given.

2. The Theory of Stress Wave

2.1 Wave Propagation

2.1.1. Basic Components of Stress and Strain:

It is necessary to review some most important concepts and equations in wave propagations in order to derive the mathematical model for a bar.

A point P is assumed with its coordinates (x, y, z) and displacements (u, v, w) in the x, y, z directions, respectively (Fig. 1).

If the increments in coordinates are $(\delta x, \delta y, \delta z)$ are sufficiently small, and the increments in displacement $(\delta u, \delta v, \delta w)$ have the following relations with $(\delta x, \delta y, \delta z)$:

$$\delta u = \frac{\partial u}{\partial x}\delta x + \frac{\partial u}{\partial y}\delta y + \frac{\partial u}{\partial z}\delta z$$
$$\delta v = \frac{\partial v}{\partial x}\delta x + \frac{\partial v}{\partial y}\delta y + \frac{\partial v}{\partial z}\delta z$$

$$\delta w = \frac{\partial w}{\partial x}\delta x + \frac{\partial w}{\partial y}\delta y + \frac{\partial w}{\partial z}\delta z$$

Now we define the following concepts, set normal strains:

$$\varepsilon_{xx} = \frac{\partial u}{\partial x} \quad \varepsilon_{yy} = \frac{\partial v}{\partial y} \quad \varepsilon_{zz} = \frac{\partial w}{\partial z}$$

and shear strains:

$$\varepsilon_{zx} = \varepsilon_{xz} = \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}$$
$$\varepsilon_{xy} = \varepsilon_{yx} = \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}$$
$$\varepsilon_{yz} = \varepsilon_{zy} = \frac{\partial w}{\partial y} + \frac{\partial v}{\partial z}$$

and rotations:

$$2\varpi_x = \frac{\partial w}{\partial y} - \frac{\partial v}{\partial z}$$
$$2\varpi_y = \frac{\partial u}{\partial z} - \frac{\partial w}{\partial x}$$
$$2\varpi_z = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$$

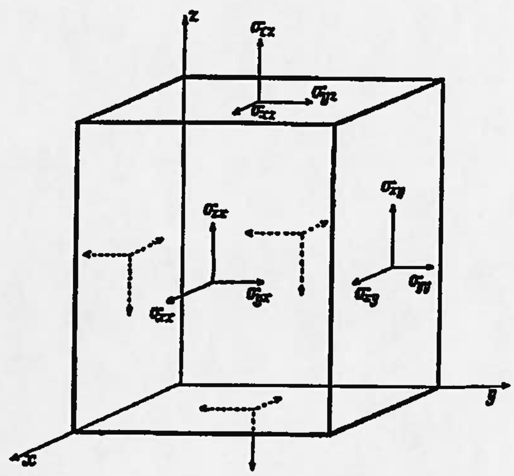

Fig.1 Illustration of Stress Components

2.1.2. Generalized Form of Hooker's Law

In an isotropic solid, the values of normal stresses are:

$$\sigma_{xx} = \lambda\Delta + 2\mu\varepsilon_{xx} \quad \sigma_{yy} = \lambda\Delta + 2\mu\varepsilon_{yy} \quad \sigma_{zz} = \lambda\Delta + 2\mu\varepsilon_{zz} \quad (1)$$

shear stresses are:

$$\sigma_{yz} = \mu\varepsilon_{yz} \qquad \sigma_{zx} = \mu\varepsilon_{zx} \qquad \sigma_{xy} = \mu\varepsilon_{xy} \qquad (2)$$

where $\Delta = \varepsilon_{xx} + \varepsilon_{yy} + \varepsilon_{zz}$;this represents the change in volume of a unit cube and is called the dilatation. μ is shear modulus or rigidity; μ and λ are known as Lame's constants.

2.1.3. Equations of Motion in an Elastic Medium

The resultant force acting in x-direction is:

$$(\sigma_{xx}+\frac{\partial\sigma_{xx}}{\partial x}\delta x)\delta y\delta z-\sigma_{xx}\delta y\delta z+(\sigma_{xy}+\frac{\partial\sigma_{xy}}{\partial y}\delta y)\delta x\delta z-\sigma_{xy}\delta x\delta z+$$
$$(\sigma_{xz}+\frac{\partial\sigma_{xz}}{\partial z}\delta z)\delta x\delta y-\sigma_{xz}\delta x\delta y$$

$$=\left(\frac{\partial\sigma_{xx}}{\partial x}+\frac{\partial\sigma_{xy}}{\partial y}+\frac{\partial\sigma_{xz}}{\partial z}\right)\delta x\delta y\delta z$$

If body forces are neglected and the Newton's second law of motion is applied, the above resultant force should be balanced, that is:

$$\left(\frac{\partial\sigma_{xx}}{\partial x}+\frac{\partial\sigma_{xy}}{\partial y}+\frac{\partial\sigma_{xz}}{\partial z}\right)\delta x\delta y\delta z=\rho\delta x\delta y\delta z\frac{\partial^2 u}{\partial^2 t}$$

In the right side of above equation, $(\rho\delta x\delta y\delta z)$ represents the mass and $\frac{\partial^2 u}{\partial^2 t}$ means accelerate, ρ is density, so that

$$\rho\frac{\partial^2 u}{\partial t^2}=\frac{\partial\sigma_{xx}}{\partial x}+\frac{\partial\sigma_{xy}}{\partial y}+\frac{\partial\sigma_{xz}}{\partial z}$$
$$\rho\frac{\partial^2 v}{\partial t^2}=\frac{\partial\sigma_{yx}}{\partial x}+\frac{\partial\sigma_{yy}}{\partial y}+\frac{\partial\sigma_{yz}}{\partial z} \quad (3)$$
$$\rho\frac{\partial^2 w}{\partial t^2}=\frac{\partial\sigma_{zx}}{\partial x}+\frac{\partial\sigma_{zy}}{\partial y}+\frac{\partial\sigma_{zz}}{\partial z}$$

Substitutes Equation (3) from Equation(1) and Equation(2), we have:

$$\rho\frac{\partial^2 u}{\partial t^2}=\frac{\partial}{\partial x}(\lambda\Delta+2\mu\varepsilon_{xx})+\frac{\partial}{\partial y}(\mu\varepsilon_{xy})+\frac{\partial}{\partial z}(\mu\varepsilon_{xz})$$

According to the above definitions, hence:

$$\rho\frac{\partial^2 u}{\partial t^2}=(\lambda+\mu)\frac{\partial\Delta}{\partial x}+\mu\nabla^2 u \quad (4)$$

where the operator ∇^2 is written for

$$\left(\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}+\frac{\partial^2}{\partial z^2}\right)$$

similarly,
$$\rho\frac{\partial^2 v}{\partial t^2}=(\lambda+\mu)\frac{\partial\Delta}{\partial y}+\mu\nabla^2 v \quad (5)$$

$$\rho\frac{\partial^2 w}{\partial t^2}=(\lambda+\mu)\frac{\partial\Delta}{\partial z}+\mu\nabla^2 w \quad (6)$$

Equations (4), (5), (6) are the motion equations of an isotropic elastic solid, in which body forces are absent, and they correspond to the propagation of two types of waves through the medium.

2.1.4. Wave Equations

If we differentiate both sides of equation (4) with respect to x, both sides of (5) w.r.t.y, and both sides of (6) w.r.t.z and add, we have:

$$\rho\frac{\partial^2 \Delta}{\partial t^2}=(\lambda+2\mu)\nabla^2\Delta \quad (7)$$

Equation (7) is *the wave equation* and shows that the dilatation Δ is propagated through the medium with velocity $\left[(\lambda+2\mu)/\rho\right]^{\frac{1}{2}}$

2.1.5. Rayligh Waves

When Rayligh waves are considered here, some equations can be drawn. Now take the boundary to be the xy plane with z positive towards the interior of the solid, and take the ***plane*** *wave* as traveling in the x-direction. Since the displacements will be independent of y we may define two potential functions ϕ and ψ to find a solution of the motion equations (4),(5),(6), such that:

$$u=\frac{\partial\phi}{\partial x}+\frac{\partial\psi}{\partial z} \qquad w=\frac{\partial\phi}{\partial z}-\frac{\partial\psi}{\partial x} \tag{8}$$

$$\therefore \Delta=\frac{\partial u}{\partial x}+\frac{\partial w}{\partial z}=\nabla^2\phi$$

Thus the general motion equations can be written as:

$$\rho\frac{\partial^2}{\partial t^2}(u,w)=(\lambda+\mu)\left(\frac{\partial\Delta}{\partial x},\frac{\partial\Delta}{\partial z}\right)+\mu\nabla^2(u,w) \tag{9}$$

If two potential functions are applied here and substituted Equation (8) into the equation of motion above Equation(9), two potential functions are solved as following:

$$\phi=A\exp[-qz+i(pt-fx)] \tag{10}$$

$$\psi=B\exp[-sz+i(pt-fx)]$$

where A and B are constants, f is frequency, p is angular frequency.

$$i=\sqrt{(-1)},\ q=\sqrt{f^2-h^2},\ s=\sqrt{f^2-k^2},\ h=p/c_1,\ k=p/c_2,\ c_1=\sqrt{\frac{\lambda+2\mu}{\rho}},\ c_2=\sqrt{\frac{\mu}{\rho}}$$

$\because \sigma_{zz}=\lambda\Delta+2\mu\frac{\partial w}{\partial z} \qquad \sigma_{zx}=\mu(\frac{\partial u}{\partial z}+\frac{\partial w}{\partial x})$ from equation (1) and equation (2), and then substitute equation (8), we have:

$$\sigma_{zz}=(\lambda+2\mu)\frac{\partial^2\phi}{\partial z^2}+\lambda\frac{\partial^2\phi}{\partial x^2}-2\mu\frac{\partial^2\psi}{\partial x\partial z} \tag{11}$$

$$\sigma_{zx}=\mu\left(2\frac{\partial^2\phi}{\partial x\partial z}-\frac{\partial^2\psi}{\partial x^2}+\frac{\partial^2\psi}{\partial z^2}\right) \tag{12}$$

Where ϕ,ψ are potential functions.

2.2 Wave Propagation Model For an Isolate Element

If the incident pulse is applied in x direction, for an arbitrary z value, there is no force on external surface (at z =Z), so the incident pulse σ^I_{zz} would equal zero. Also, σ^I_{zx} is zero. That is:

$$\sigma_{zz}=0 \qquad \sigma_{zx}=0$$

$$\sigma_{zz}=0\Rightarrow A\{(\lambda+2\mu)q^2-\lambda f^2\}e^{-qz}-B(2\mu i f s)e^{-sz}=0 \tag{13}$$

$$\sigma_{zx}=0\Rightarrow 2A(ifq)e^{-qz}-B(f^2+s^2)e^{-sz}=0 \tag{14}$$

From Equation (14) we have,

$$B=-A\frac{2ifqe^{-qz}}{(f^2+s^2)e^{-sz}} \tag{15}$$

$\because u=\dfrac{\partial\phi}{\partial x}+\dfrac{\partial\psi}{\partial z}$ (Eq.8), substitute Equation (15) and potential functions equation (10) into equation (8), we have:

$$u=A\left[f-\frac{2fqs}{f^2+s^2}\right]e^{-qz}(-i)e^{i(pt-fx)} \tag{16}$$

The maximum amplitude occurs when | *cos(pt-fx)* | = 1, so the amplitude is:

$$u=A\left[f-\frac{2fqs}{f^2+s^2}\right]e^{-qz} \tag{17}$$

Similarly, $w=\dfrac{\partial\phi}{\partial z}-\dfrac{\partial\psi}{\partial x}$

Substitute equation (15) and potential functions equation (10) into equation (8), we have:

$$w=-A\left[q-\frac{2qf^2}{f^2+s^2}\right]e^{-qz}e^{i(pt-fx)}$$

The amplitude is:

$$w=-A\left[q-\frac{2qf^2}{f^2+s^2}\right]e^{-qz} \qquad (18)$$

From Equation(1), $\sigma_{xx}=\lambda\Delta+2\mu\varepsilon_{xx}=\lambda(\varepsilon_{xx}+\varepsilon_{yy}+\varepsilon_{zz})+2\mu\varepsilon_{xx}$, but ε_{yy} is not considered here, then

$$\sigma_{xx}=\lambda\left(\varepsilon_{xx}+\varepsilon_{zz}\right)+2\mu\varepsilon_{xx} \qquad (19)$$

If potential functions applied and Equation(19) is simplified, thus the amplitude is:

$$\sigma_{xx}=A\left[(\lambda+2\mu)f\left(f-\frac{2fqs}{f^2+s^2}\right)+\lambda q\left(q-\frac{2qf^2}{f^2+s^2}\right)\right]e^{-qz} \qquad (20)$$

where λ, μ, f, q, s are all constants for a given pulse. Therefore equation (20) can be written as:

$$\sigma_{xx}=Ae^{-qz}$$

Because z is arbitrary for a bar and not related to azimuth (i.e. independence of y direction), so z can be replaced by another character r; and r means distance from axis in radial direction. That is:

$$\sigma_{xx}=Ae^{-qr} \qquad (21)$$

However $\because q=\sqrt{f^2-h^2}$, and $f>>h$ (because $h=p/c1$ <<1), so Equation (21) can be simplified as following:

$$\sigma_{xx}=Ae^{-fr} \qquad (22)$$

There are two aspects should be noticed for equation (22). Firstly, it shows the stress amplitude decays with the increase of radius. If the radius are not change, the stress amplitudes doesn't decay in longitudinal wave propagation. Secondly, although it is derived from a point of element stress analysis, it is the general expression and can be suitable for a structure too.

3. Longitudinal Waves in Rods

3.1. Equation of Motion

For consider a small element PQ of length δx and let the cross sectional area of the rod be A (see Fig.2). If the stress on the face passing through P is σ_{xx} the stress on the other face will be given by $\sigma xx + \left(\partial\sigma_{xx}/\partial x\right)\delta x$, and if the displacement of the element is given by u, we have from Newton's second law of motion:

$$\rho A \delta x \frac{\partial^2 u}{\partial t^2} = A\frac{\partial \sigma_{xx}}{\partial x}\delta x$$ where ρ is the density of the rod.

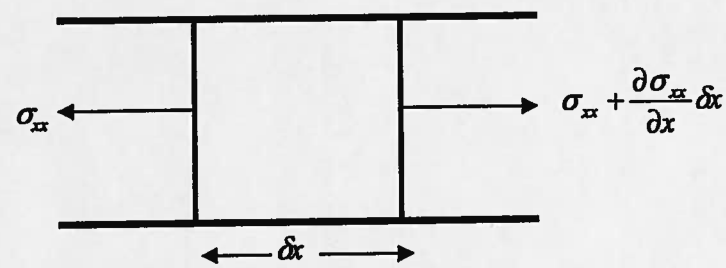

Fig.2 Element of rod with loads

The equation of motion is:

$$\rho\frac{\partial^2 u}{\partial t^2} = E\frac{\partial^2 u}{\partial x^2}$$

Using discrete Fourier series transform, the solution can be written as:

$$u = (x,t)\sum A e^{i\omega\eta} + \sum B e^{i\omega\zeta}$$

Consequently, the solution can be written directly as :

$u(x,t) = f(x - c_0 t) + F(x + c_0 t)$ where $c_0 = \sqrt{(ES)/(pS)}$,S is the area. This, of course ,is the D'Alembert solution and says that the pulse keeps the same shape as it propagates.

Using the subscripts i and r to represent the forward and backward moving waves, respectively, then the other mechanical quantities can be obtained as

Displacement: $u_i = Ae^{-i(kx-\omega t)}$ $\quad u_r = Be^{i(kx+\omega t)}$

strain : $e_i = -iku_i$ $\quad e_r = -iku_r$

stress: $\sigma_i = -ikEu_i$ $\quad \sigma_r = -ikEu_r$

Force: $F_i = -ikESu_i$ $\quad F_r = -ikESu_r$

3.2 Reflections and Transmissions at an Interface Between Two Media

Any change in cross section or material properties will cause the generation of new waves. While the actual situation is very complicated, in the present one-dimensional analysis only a longitudinal transmitted wave and a longitudinal reflection wave are generated.

The incident wave generates a reflected wave in such a way that the two superpose at the boundary to satisfy the boundary conditions. The only waves that can be present are the two given by[2]:

$$u_1(x,t) = \sum A_1 e^{-i(k_1 x - wt)} + \sum B_1 e^{+i(k_1 x + wt)}$$
$$u_2(x,t) = \sum A_2 e^{-i(k_2 x - wt)}$$

Where A is associated with the known incident (transmitted) wave and B with the unknown reflected wave.

The case of transmission of waves from one medium to another is handled as following rules: balance of force at the interface and continuity of displacement. They give:

$$E_1 S\,(-A_1+B_1)(ik_1) = E_2\,S\,(-A_2)\,(ik_2)$$

$$A_1 + B_1 = A_2$$

Where S is area, solving for the reflected and the transmitted coefficients then gives

$$B_1 = \left(\frac{1-\sqrt{r_s r_D}}{1+\sqrt{r_s r_D}}\right)A_1 \qquad A_2 = \left(\frac{2}{1+\sqrt{r_s r_D}}\right)A_1$$

where the following notions are used

$$r_s = \frac{E_2}{E_1} \qquad r_D = \frac{\rho_2}{\rho_1}$$

Thus the response is independent of frequency. Because forces are in equilibrium, the expressions for the stresses are:

$$\sigma_r = \frac{\sqrt{r_s r_D} - 1}{\sqrt{r_s r_D} + 1}\sigma_i \qquad \sigma_t = 2\frac{\sqrt{r_s r_D}}{\sqrt{r_s r_D} + 1}\sigma_i$$

Because wavelength is greater than thickness of any media and $p/c_1 << f$, so for multi-layer media (along x direction), we have the following important equations:

(a). the equation of reflection pressure ratio:

$$r_p = \frac{\rho_2 c_2 - \rho_1 c_1}{\rho_2 c_2 + \rho_1 c_1} = \frac{\sigma_r}{\sigma_i} \Rightarrow \sigma_r = r_p \cdot \sigma_i \qquad (23)$$

(b). the equation of transmission pressure ration:

$$\tau_p = \frac{2\rho_2 c_2}{\rho_2 c_2 + \rho_1 c_1} = \frac{\sigma_t}{\sigma_i} \Rightarrow \sigma_t = \tau_p \cdot \sigma_i \qquad (24)$$

(c) the total incident pressure is:

$$\sigma_{i-total} = \sigma_i + \sigma_r = \sigma_t \qquad (25)$$

where σ_i is incident wave pressure, σ_r is reflection wave pressure, and σ_t is transmission wave pressure. $R_1 = \rho_1 c_1$ $R_2 = \rho_2 c_2$ ρ is medium density and c is propagation velocity. The schematic illustration is shown in Fig.3.

So far we have got Equations (22), (23), (24), (25) and they are called the mathematical model for a uniform bar.

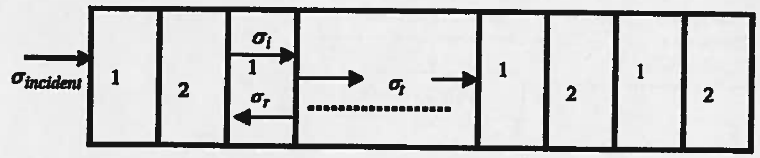

Fig.3 Schematic illustration of stress wave propagation at one interface

4. Examples of Calculation

The model of proposed above is based on the following assumptions:

(1) Internal friction is not considered.

(2) Wavelength is much greater than the thickness of any media

(3) The period of stress wave is considered as 10^{-2} to 10^{-3} second order.

Fig.4 presents all stress components act on an interface j .

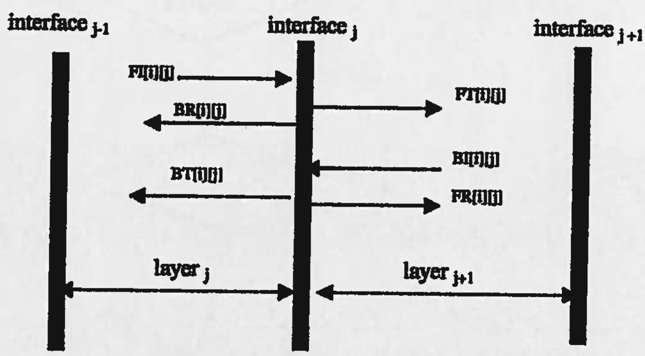

Fig.4 all stress components act on an interface j

Now we give the flow chart of computer numerical calculation of stresses in the interfaces propagation (see Fig.5).

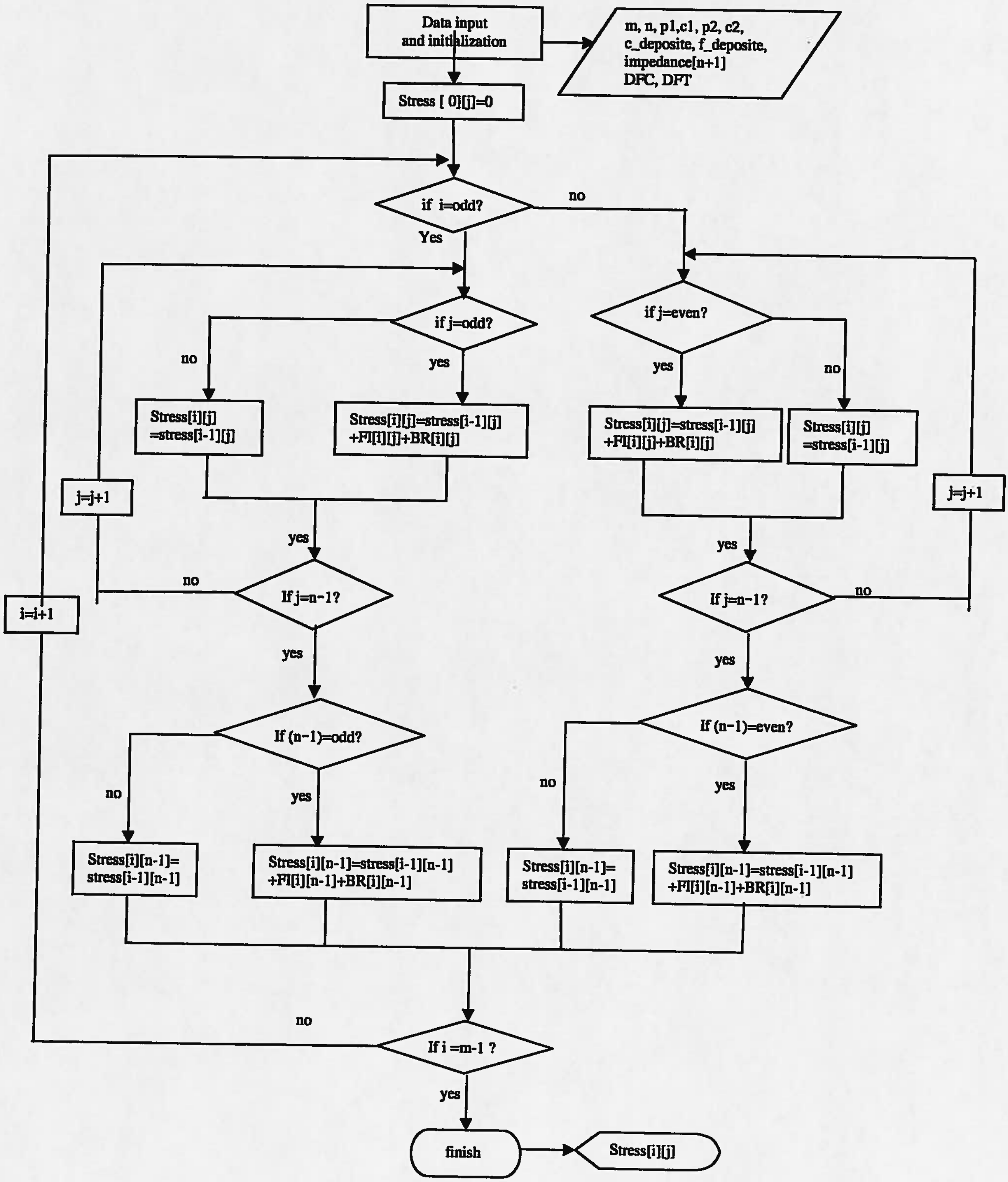

Fig.5 the flow chart of computer numerical calculation of stresses in the interfaces propagation

Case1: total time moments m = 20 unit, total interfaces n=21, incident pulse =100*10^6 pa, the tensile limit strength DFT= -100*10^6 pa, (minus means tensile), the compressive limit strength DFC=200*10^6 pa, p1=2000kg/m^3, c1=2738 m/s, p2=2200kg/m^3, c2=3371 m/s.

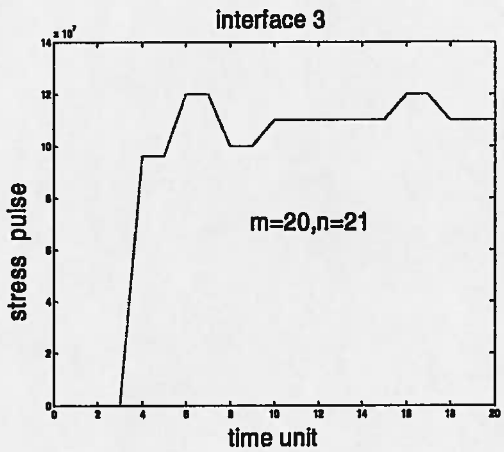

Fig. 6

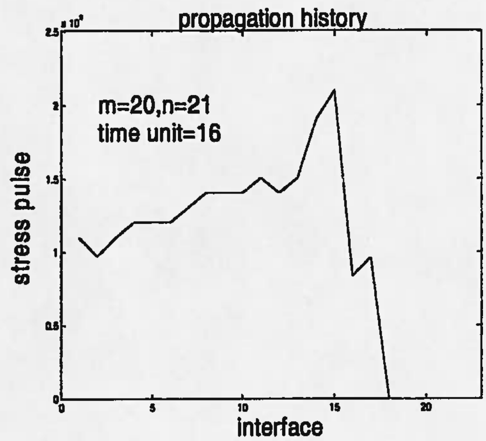

Fig.7

The result shows the first compressive fracture occurs at interface 14 and the moment 16 when stress pulse is 2.1e+008; the maximum stress is 2.2e+008,it occurs at interface 16 and the moment 18; the minimum stress is 0, it occurs at interface 1 and the moment 0.

Case 2: m=60, n=50,incident pulse =100*10^6 pa, the tensile limit strength DFT= -100*10^6 pa, the compressive limit strength DFC=200*10^6 pa, p1=2000kg/m^3, c1=2738m/s, p2=2200kg/m^3, c2=3371 m/s.

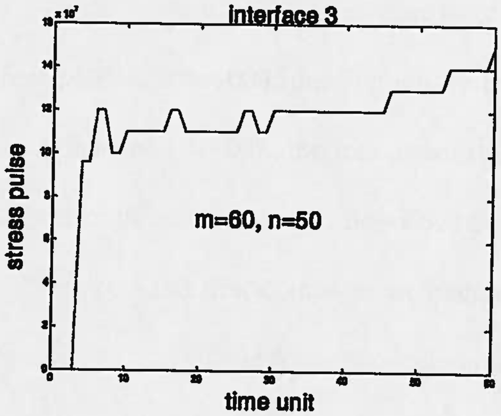

Fig.8

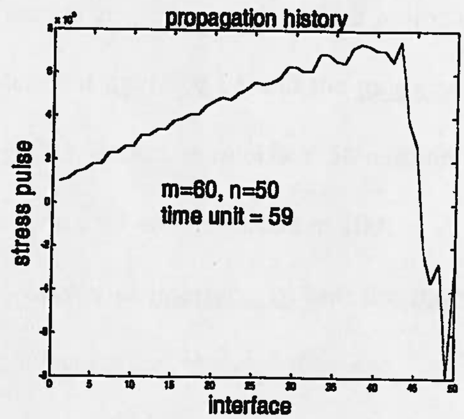

Fig.9

The result shows that the first compressive fracture occurs at <u>interface 14</u> and the <u>moment 16</u>, the stress pulse is 2.1e+008; the first tensile fracture occurs <u>at interface 48</u> and the <u>moment 56</u>,the stress pulse is -5.2e+008; the maximum stress is 7.5e+008,it occurs at interface 42 and the moment 58; the minimum stress is -7.9e+008,it occurs at interface 48 and the moment 58;

Case3: total time moments m = 110 unit, total interfaces n=100, incident pulse $=100*10^6$ pa, the tensile limit strength DFT= $-100*10^6$ pa, the compressive limit strength DFC=$200*10^6$ pa, p1=2000kg/m^3, c1=2738 m/s, p2=2200kg/m^3, c2=337 m/s.

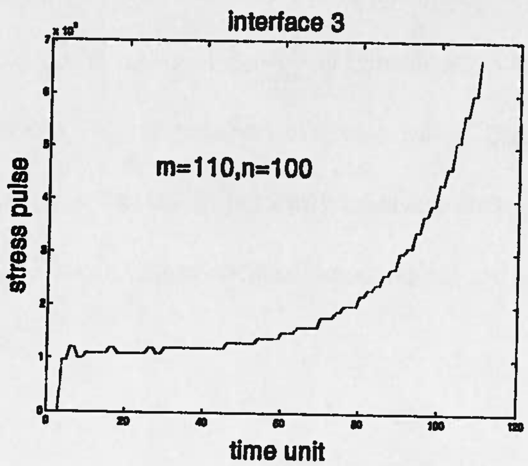

Fig.10

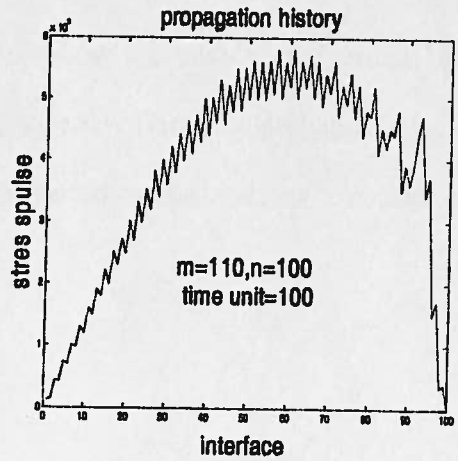

Fig.11

The result shows the first compressive fracture occurs at interface 14 and the moment 16, the stress pulse is 2.1e+008; the first tensile fracture occurs at interface 98 and the moment 108, the stress pulse is 2.7e+008; the maximum stress is 9e+009,it occurs at interface 58 and the moment 108; the minimum stress is -1.5e+009,it occurs at interface 99 and the moment 109;

Therefore, the first compressive fracture always occurs at interface 14 and the moment 16; whereas the first tensile fracture occurs at the end of interface and the time moment; the maximum stress increases with the number of layer and time but the minimum stress decreases with the number of layer and time because too much interactions are occurred.

5. Conclusion

From the above figures, it can be observed that stress discontinuities arise at the material boundaries, with constant levels in between if the radii are the same. When stress concentrations appear, they do so at the discontinuities, that demonstrates that these stress components cannot be neglected in a strength or fatigue analysis. So, the calculation model can be used to predict the damage induced by an impact pulse.

Real solids are neither perfectly elastic, that means when a pulse traveled through medium, some of mechanical energy is converted to heat energy whose mechanisms are termed internal frictions. The alternation of stress waves depend on frequency. Here, the propagation of stress waves are treated in perfectly elastic solids and one of period of stress waves are taken as 10^{-2}~10^{-3} so that alternation of stress waves are neglected.

References

1.Kolsky H., "stress waves in solids ", New York: Dover, 1963.

2. JamesF. Doyle, "wave propagation in structures: An FFT-Based spectral analysis methodology", Springer-Verlag , 1989.

3.B.Parga-Landa, S.Vlegels, F.Hernandez-Olivares, S.D.Clark, "Analytical simulation of stress wave propagation in composite materials", composite structures 45(1999), pp.125-129.

4.Ripperger, E.A., Abramson, H.N., 1957, "Reflection and transmission of elastic pulses in a bar at a discontinuity in cross section", the third Midwestern conf. On solid.

5. Richard J. Wasley , "Stress wave propagation in solids: an introduction", New York, M. Dekker, INC.1973.

6. Julius Miklowitz, Jan D.Achenbach, "Modern problems in elastic wave propagation: intentional union of theoretic and applied", A Wiley-interscience publication,1978.

7.Ivan Toistoy, " Wave propagation", McGraw-Hill, INC.,book company, 1973.

8. T.L. Anderson, "Fracture mechanics: fundamentals and applications", CRC press, INC., 2000.

Appendix C++ Program Codes

```
//*****************************************************************
// The following program is used to calculate the stress wave values
//  at any moment in any one of the interfaces for a  uniform bar.
//*****************************************************************
//
//    FI[m][n] presents the forward incident on the nth interface at m
//    moment  (left side of the interface and the wave direction is to the right)
//
//   BR[m][n] presents the backward reflection on the nth interface
//   at m  moment produced by FI[m][n]
//
//    FT[m][n] presents the forward transmission on the nth interface
//    at m moment produced by FI[m][n];
//
//    BI[m][n] presents the backward incident on the nth interface
//    at m  moment (right side of the interface and the wave direction is to  the left)
//
//    BT[m][n] presents the backward transmission on the nth
//    interface at m moment produced by BI[m][n]
//
//    FR[m][n] presents the forward reflection on the nth
//    interface at m moment produced by BI[m][n]
//
//    stress[m][n] presents the history stress on the interfaces
//*****************************************************************


#include <iostream.h>
#include <math.h>
#include <iomanip.h>
#include <fstream.h>
#include <time.h>
#include <stdio.h>

void main ()
{
        //ofstream fcout ("c:\\mathlab6p1\\work\\hong.m")
        //ifstream fcout("c:\\matlab6p5\\work\\hong.m");
 int const m=20;    // m presents the total number of time traveled
 int const n=21;    // n presents the total number of interfaces


 double FI[m][n];
 double BR[m][n];
 double FT[m][n];
 double BI[m][n];
 double BT[m][n];
 double FR[m][n];
 double stress[m][n];


 double p1,c1;
 double p2,c2;
```

```
    cout<<"Please specify the density of medium_1 p1=";
            cin>>p1;
            cout<<endl;
    cout<<"Please specify the velocity of medium_1 c1=";
            cin>>c1;
            cout<<endl;
          cout<<"Please specify the density of medium_2 p2=";
            cin>>p2;
            cout<<endl;
          cout<<"Please specify the velocity of medium_2 c2=";
            cin>>c2;
     cout<<endl;

   double c_deposit;
   double f_deposit=0;

    cout<<"Please specify the incident pulse = ";
          cin>>c_deposit;
    cout<<endl;

          double DFC;    //compressive strength
          double DFT;    //tensile strength;

          cout<<"Please specify the compressive strength DFC=";
          cin>>DFC;
    cout<<endl;

    cout<<"Please specify the tensile strength DFT=";
          cin>>DFT;
    cout<<endl;

double impedance[n+1];// input impedance of each layer
 for(int k=0;k<n+1;k++)
 {
          if (k%2==0)
     impedance[k]=p1*c1;  //3*3000
          else
            impedance[k]=p2*c2;  //8.5*3500
 }


int i,j;
for (i=0;i<m;i++)
          for (j=0;j<n;j++)
          {
           FI[i][j]=0;
           BR[i][j]=0;
                  FT[i][j]=0;
                  BI[i][j]=0;
                  BT[i][j]=0;
                  FR[i][j]=0;
                  stress[i][j]=0;
          }

//j=0;
```

```
for(i=0;i<m;i++)
        {

    if(i%2==0) // when time moment is odd
            {
                        // time divided by even or odd and calculation of the 0th layer
        FI[i][0]=c_deposit;
                        BI[i][0]=BR[i-1][1];
                        BR[i][0]=FI[i][0]*(impedance[1]-impedance[0])/
                                        (impedance[1]+impedance[0]);
        BT[i][0]=BI[i][0]*(2*impedance[0])/
                    (impedance[1]+impedance[0]);
                        FR[i][0]=BI[i][0]*(impedance[0]-impedance[1])/
                    (impedance[1]+impedance[0]);
                        FT[i][0]=FI[i][0]*(2*impedance[1])/
                                        (impedance[1]+impedance[0]);

        BR[i][0]=BR[i][0]+BT[i][0];
                        FT[i][0]=FT[i][0]+FR[i][0];
        c_deposit=-BR[i][0];

                        if (i==0)
                          stress[0][0]=stress[0][0]+FI[0][0]+BR[0][0];
                        else
                        {
                                stress[i][0]=stress[i-1][0]+FI[i][0]+BR[i][0];
                        }
                        for(j=1;j<n-1;j++) //note : j take n-1
                        {
                                // the number of interfaces divided by even or odd
                                if(j%2==0) //calculation of other odd layers and the moment is odd too
                                {
                                        FI[i][j]=FT[i-1][j-1];
                                        BI[i][j]=BR[i-1][j+1];
                                        // note: j take n-1 layers
                    BR[i][j]=FI[i][j]*(impedance[j+1]-impedance[j])/
                                (impedance[j]+impedance[j+1]); //the relection ratio of the incident

                                        BT[i][j]=BI[i][j]*(2*impedance[j])/
                (impedance[j]+impedance[j+1]); //the refraction raio of the incident

                                        FR[i][j]=BI[i][j]*(impedance[j]-impedance[j+1])/
                (impedance[j]+impedance[j+1]);

                                        FT[i][j]=FI[i][j]*(2*impedance[j+1])/
                                            (impedance[j]+impedance[j+1]);

                                BR[i][j]=BR[i][j]+BT[i][j];

                                        FT[i][j]=FR[i][j]+FT[i][j];

                                        if (i==0)
                                                stress[0][j]=0;
                                        else
                                        {
                                                stress[i][j]=stress[i-1][j]+FI[i][j]+BR[i][j];
```

```
                        }
                }
        else  //if other layers are even but moment is odd
                {
                        if (i==0)
                                stress[0][j]=0;
                        else
                                stress[i][j]=stress[i-1][j];
                }
        }// end of for
            if((n-1)%2==0)
                {
                        //assume : the times are odd
                        //  the last layer is odd
                FI[i][n-1]=FT[i-1][n-2];

                BI[i][n-1]=f_deposit;

                BR[i][n-1]=FI[i][n-1]*(impedance[n]-impedance[n-1])/
                                (impedance[n]+impedance[n-1]);

                BT[i][n-1]=BI[i][n-1]*(2*impedance[n-1])/
                                (impedance[n]+impedance[n-1]);

                FR[i][n-1]=BI[i][n-1]*(impedance[n-1]-impedance[n])/
                                (impedance[n-1]+impedance[n]);

                FT[i][n-1]=FI[i][j]*(2*impedance[n])/
                                (impedance[n]+impedance[n-1]);

                BR[i][n-1]=BR[i][n-1]+BT[i][n-1];

                FT[i][n-1]=FT[i][n-1]+FR[i][n-1];

                f_deposit=-FT[i][n-1];

                if(i==0)
                        stress[0][n-1]=0;
                else
                        stress[i][n-1]=stress[i-1][n-1]+FI[i][n-1]+BR[i][n-1];
                }
                        else
                        {
                                if(i==0)
                                        stress[0][n-1]=0;
                                else
                                        stress[i][n-1]=stress[i-1][n-1];
                        }//the last layer -- evev
    }

else //if moment is even
    {
    stress[i][0]=stress[i-1][0];
        for(j=1;j<n-1;j++)
        {
                if(j%2!=0)  //if layer is even
```

```
{
        FI[i][j]=FT[i-1][j-1];

        BI[i][j]=BR[i-1][j+1];
        // ----note : j take n-1 layers-----

        BR[i][j]=FI[i][j]*(impedance[j+1]-impedance[j])/
               (impedance[j]+impedance[j+1]);

    FT[i][j]=FI[i][j]*(2*impedance[j+1])/
               (impedance[j]+impedance[j+1]);

    BT[i][j]=BI[i][j]*(2*impedance[j])/
               (impedance[j]+impedance[j+1]);

    FR[i][j]=BI[i][j]*(2*impedance[j])/
               (impedance[j]+impedance[j+1]);

    BR[i][j]=BR[i][j]+BT[i][j];

        FT[i][j]=FT[i][j]+FR[i][j];

        stress[i][j]=stress[i-1][j]+FI[i][j]+BR[i][j];
}
else
{
 stress[i][j]=stress[i-1][j];
}
}
if((n-1)%2!=0) // the last layer is even
        {
   //assume : the number of the layers are even
   //and :  the times are also enen

    BI[i][n-1]=f_deposit;

     FI[i][n-1]=FT[i-1][n-2];

     BR[i][n-1]=FI[i][n-1]*(impedance[n]-impedance[n-1])/
                (impedance[n]+impedance[n-1]);

     FT[i][n-1]=FI[i][n-1]*(2*impedance[n])/
                (impedance[n]+impedance[n-1]);

     BT[i][n-1]=BI[i][n-1]*(2*impedance[n-1])/
               (impedance[n]+impedance[n-1]);

     FR[i][n-1]=BI[i][n-1]*(impedance[n-1]-impedance[n])/
              (impedance[n-1]+impedance[n]);

     BR[i][n-1]=BR[i][n-1]+BT[i][n-1];

     FT[i][n-1]=FT[i][n-1]+FR[i][n-1];

     f_deposit=-FT[i][n-1];
```

```
                stress[i][n-1]=stress[i-1][n-1]+FI[i][n-1]+BR[i][n-1];
                    }
                else
                {
                    stress[i][n-1]=stress[i-1][n-1];
                    //the last layer --- odd
                }
        }
    }

int h;
h=0;
for(i=0;i<m;i++)
    for(j=0;j<n;j++)
    {
        cout.width(10);
    cout.precision(2);
        cout<<stress[i][j]<<"    ";
        h=h++;
        if(h%5==0)
            cout<<" ..."<<endl;
    }


    int layer_1,layer_2,moment_1,moment_2;
    for(i=0;i<m;i++)
    {
        for(j=0;j<n;j++)
        if (stress[i][j]>DFC)
        {
          layer_1=j;
          moment_1=i;
      cout<<"The first compressive fracture occurs at interface "<<layer_1;
          cout<<" and the moment "<<moment_1<<endl;
          cout<<endl;
          cout<<"The stress pulse is "<<stress[i][j]<<endl;
      cout<<endl;
      break;
        }
                    if (stress[i][j]>DFC)
                        break;
    }

    for(i=0;i<m;i++)
    {
        for(j=0;j<n;j++)
         if(stress[i][j]<DFT)
        {
            layer_2=j;
            moment_2=i;
            cout<<"The first tensile fracture occurs at interface "<<layer_2;
            cout<<" and the moment "<<moment_2<<endl;
            cout<<endl;
            cout<<"The stress pulse is "<<stress[i][j]<<endl;
            cout<<endl;
            break;
```

```
                        }
                        if(stress[i][j]<DFT)
                                break;
            }

//************** Calculation of the fracture position *****************

    int layer_11,layer_12,moment_11,moment_12;
            int const g=m*n;
            double max,min;

            max=min=stress[0][0];
            for(i=0;i<m;i++)
              for(j=0;j<n;j++)
                if(max<stress[i][j])
                        {
                          max=stress[i][j];
                           layer_11=j;
                           moment_11=i;
                        }
                        else if (min>stress[i][j])
                        {
                           min=stress[i][j];
                          layer_12=j;
                           moment_12=i;
                        }
      cout<<"the maximumn stress is "<<max<<endl;
              cout<<endl;
              cout<<"It occurs at interface "<<layer_11;
              cout<<" and the moment "<<moment_11<<endl;
              cout<<endl;
              cout<<"the minimum stress is "<<min<<endl;
              cout<<endl;
              cout<<"It occurs at interface "<<layer_12;
              cout<<" and the moment "<<moment_12<<endl;
              cout<<endl;


 //************** end of above calculation ******************


   int p;
     cout<<"Which interface do you want to know ? ";
              cin>>p;
              cout<<endl;
              while (p>=n)
              {
                          cout<<"this interface is beyond the maximum interface, choose again!"<<endl;
                          cout<<"Which interface do you want to know ? ";
                          cin>>p;
                          cout<<endl;
              }

      h=0;
              cout<<endl;
              for(i=0;i<m;i++)
```

```
            {
              cout.width(10);
              cout.precision(2);
              cout<<stress[i][p]<<"    ";
              h=h++;
               if(h%5==0)
                        cout<<" ..."<<endl;
      }

          cout<<endl;
          int t;
          cout<<"What time do you want to calculate ?";
          cin>>t;
          cout<<endl;
          while (t>=m)
          {
                    cout<<"this moment is beyond the maximum moment, choose again!"<<endl;
                    cout<<"Which moment do you want to know ? ";
                    cin>>t;
                    cout<<endl;
          }
          for(j=0;j<n;j++)
                    cout<<" stress["<<t<<"]["<<j<<"] = "<<stress[t][j]<<endl;

          h=0;
          cout<<endl;
          for(j=0;j<n;j++)
          {
            cout.width(10);
             cout.precision(2);
             cout<<stress[t][j]<<"    ";
                    h=h++;
                    if(h%5==0)
                              cout<<" ..."<<endl;

      }
         cout<<endl;
}
```