5 Circular Tunnel Problems

5.1 Problem Statement

This problem concerns stress analysis of a long circular opening in an infinite medium under various boundary conditions and material properties (see Figure 5.1).* Three variations to this problem will be considered:

1. Part A – tunnel in an elastic medium with a biaxial stress field;
2. Part B – tunnel in an elastic-plastic medium with a hydrostatic stress field; and

Upon excavation of a tunnel, the in-situ stresses within the rock or soil mass are redistributed from a uniform orthogonal stress field to a more complex stress distribution. Stress concentrations around a tunnel cause elastic deformations at the periphery and, if the yield strength of the material is exceeded, result in plastic deformations and redistribution of stresses due to yielding of the material. In the case of plastic yielding, a yield zone will develop around the tunnel, beyond which the stresses will be elastic. These processes are modeled by parts A and B of this problem.

Part C of this problem involves the interaction of a structural tunnel lining and an elastic medium. Although the actual design of a tunnel lining is more complex, this problem checks the basic interaction between the two types of material for non-axisymmetric loadings.

These problems have a closed-form analytical solution, and thus several aspects of the computer model can be tested:

1. the ability of the code to simulate an infinite medium by boundary elements;
2. the determination of displacements and stresses in a nonsymmetric problem in two dimensions;
3. the computation of plastic stresses and deformations; and
4. the interaction between structural lining and rock or soil mass.

---

* This section was prepared for the U.S. Nuclear Regulatory Commission under U.S. NRC Contract No. 02-85-002.
5.2 Analytical Solutions

5.2.1 Cylindrical Hole in an Infinite Elastic Medium

For a cylindrical hole in an infinite, isotropic, elastic medium under plane-strain conditions, the radial and tangential stress distributions are given by the classical Kirsch solution (e.g., see Jaeger and Cook 1976).

A point located at polar coordinate $(r, \theta)$ near an opening with radius $a$ (see Figure 5.1 (a)) has stresses $\sigma_r$, $\sigma_{\theta}$, $\tau_{r\theta}$, given by
The displacements can also be determined assuming conditions of plane strain:

\[
\frac{4G}{p_1 + p_2} \frac{u_r}{a} = U_r \left( \frac{r}{a}, \theta; \nu, \frac{p_1 - p_2}{p_1 + p_2} \right) = \frac{a}{r} + \frac{p_1 - p_2}{p_1 + p_2} \frac{a}{r} \left[ 4(1 - \nu) - \frac{a^2}{r^2} \right] \cos 2\theta
\]

\[
\frac{4G}{p_1 + p_2} \frac{u_\theta}{a} = U_\theta \left( \frac{r}{a}, \theta; \nu, \frac{p_1 - p_2}{p_1 + p_2} \right) = -\frac{p_1 - p_2}{p_1 + p_2} \frac{a}{r} \left[ 2(1 - 2\nu) + \frac{a^2}{r^2} \right] \sin 2\theta
\]

in which \( u_r \) is the radial outward displacement, and \( u_\theta \) is the tangential displacement. \( G \) is the shear modulus, and \( \nu \) is the Poisson’s ratio.

### 5.2.2 Cylindrical Hole in an Infinite Mohr-Coulomb Medium

The yield zone radius, \( R_o \) (see Figure 5.1 (b)), is given analytically by a theoretical model based on the solution of Salencon (1969):

\[
\frac{R_o}{a} = R_o^* \left( \frac{p_2}{q}, \frac{p_i}{q}, K_p \right) = \left( \frac{2}{K_p + 1} \frac{p_2}{q} + \frac{1}{K_p - 1} \frac{p_i}{q} + \frac{1}{K_p - 1} \right)^{1/(K_p - 1)}
\]

- \( K_p = (1 + \sin \phi)/(1 - \sin \phi) \);
- \( q = 2c \tan(45 + \phi/2) \); and
- \( p_i \) = internal pressure.
The radial stress at the elastic-plastic interface is
\[
\frac{\sigma_{re}}{q} = -\frac{1}{K_p + 1} \left( 2 \frac{p_2}{q} - 1 \right) \tag{5.4}
\]

The stresses in the plastic zone are
\[
\frac{\sigma_r}{q} = \sigma_r^* \left( \frac{p_i}{q}, K_p \right) = \frac{1}{K_p - 1} - \left( \frac{p_i}{q} + \frac{1}{K_p - 1} \right) \cdot \left( \frac{r}{a} \right)^{K_p - 1} \tag{5.5}
\]
\[
\frac{\sigma_\theta}{q} = \sigma_\theta^* \left( \frac{p_i}{q}, K_p \right) = \frac{1}{K_p - 1} - K_p \left( \frac{p_i}{q} + \frac{1}{K_p - 1} \right) \cdot \left( \frac{r}{a} \right)^{K_p - 1}
\]

The stresses in the elastic zone are
\[
\frac{\sigma_r}{q} = \sigma_r^* \left( \frac{p_2}{q}, \frac{p_i}{q}, K_p \right) = -\frac{p_2}{q} + \left( \frac{p_2}{q} + \frac{\sigma_{re}}{q} \right) \cdot \left( \frac{Ro}{r} \right)^2 \tag{5.6}
\]
\[
\frac{\sigma_\theta}{q} = \sigma_\theta^* \left( \frac{p_2}{q}, \frac{p_i}{q}, K_p \right) = -\frac{p_2}{q} - \left( \frac{p_2}{q} + \frac{\sigma_{re}}{q} \right) \cdot \left( \frac{Ro}{r} \right)^2
\]

5.2.3 Lined Tunnel in an Infinite Elastic Medium

The analytical solution for an elastic liner embedded in an elastic solid with non-slipping interface (see Figure 5.1 (c)) is given by Einstein and Schwartz (1979). The thrust or axial force in the liner, \(N\), and bending moment, \(M\), are given in Eqs. (5.7) and (5.8), respectively:

\[
\frac{N}{P_o a} = N^* \left( \theta; \frac{p_1}{p_2}, \nu, \nu_s, \frac{E_s}{E}, \frac{d}{a} \right) = \frac{1}{2} \left( 1 + \frac{p_1}{p_2} \right) (1 - a_0^*) + \frac{1}{2} \left( 1 - \frac{p_1}{p_2} \right) (1 + 2a_2^*) \cos 2\theta \tag{5.7}
\]
\[
\frac{M}{P_o a^2} = M^* \left( \theta; \frac{p_1}{p_2}, \nu, \nu_s, \frac{E_s}{E}, \frac{d}{a} \right) = \frac{1}{4} \left( 1 - \frac{p_1}{p_2} \right) (1 - 2a_2^* + 2b_2^*) \cos 2\theta \tag{5.8}
\]
\( E \) = Young’s modulus of the rock;
\( \nu \) = Poisson’s ratio of the rock;
\( E_s \) = Young’s modulus of the liner;
\( \nu_s \) = Poisson’s ratio of the liner;
\( d \) = thickness of the liner;
\( A = d \), cross-sectional area of the liner for a 1 m long section; and
\( I = d^3/12 \), liner moment of inertia;

\[ a_0^* = \frac{C^* F^* (1-\nu)}{C^* + F^* + C^* F^* (1-\nu)}; \]
\[ a_2^* = \beta \cdot b_2^*; \]
\[ \beta = \frac{(6+F^*) C^* (1-\nu) + 2F^* \nu}{3F^* + 3C^* + 2C^* F^* (1-\nu)}; \]
\[ b_2^* = \frac{2}{C^* (1-\nu) + 4\nu - 6\beta - 3\beta C^* (1-\nu)}; \]
\[ C^* = \frac{E a (1-\nu^2)}{E_s A (1-\nu^2)}; \]
\[ F^* = \frac{E a^3 (1-\nu^2)}{E_s I (1-\nu^2)}. \]

UDEC structural elements do not require a Poisson’s ratio to be specified. In order to comply with the analytical solution, a plane-strain correction of \((1 - \nu^2_s)\) is applied to the Young’s modulus of the liner.

### 5.3 UDEC Models

The following dimensionless parameters and values are used to describe the problems. Note that the density is not required by the analytical solution, but some value must be provided in UDEC. Since the solutions are independent of the choice of density, we used \( \rho = 1 \).

\[ \frac{p_2}{p_1} = 0.5 \]
\[ \frac{(p_1 - p_2)}{(p_1 + p_2)} = \frac{1 - p_2/p_1}{1 + p_2/p_1} = 1/3 \]
\[ p_2/q = 0.75 \]
\[ \nu = 0.20 \]
\[ E/q = 300 \]
\[ \phi = 20^\circ \]
\[ \psi = 20^\circ \]
\[
\nu = 0.20 \\
\frac{E_s}{E} = 3 \\
\frac{d}{a} = 0.1
\]

For each part, two different discretizations are used, such that characteristic lengths, \( \ell_z \), of zones at the tunnel contour are (1) \( \frac{a}{\ell_z} = 5 \) (see Figure 5.2), and (2) \( \frac{a}{\ell_z} = 10 \) (see Figure 5.3). In both geometries, the inner and outer radii were 5.0 m and 30.0 m, respectively. Also, boundary elements were coupled to gridpoints in the outer boundary in both cases. “Glued” joints were used to provide the needed discretization in each case.

In Part C of the problem, interaction of a structural lining with the surrounding material is modeled. For this part, the lining was divided into 48 (for \( \frac{a}{\ell_z} = 5 \)) and 96 (for \( \frac{a}{\ell_z} = 10 \)) linear segments. To satisfy the conditions of perfect bonding between the lining and surrounding material, high interface stiffness and strength parameters were specified.

![Figure 5.2 Coarse zoning used in circular tunnel problems](image-url)
5.4 Results

Part A – The results for Part A are compared graphically with the analytic solution in Figures 5.4 and 5.5. All results shown are for a line along the major principal stress direction. The finer zoning resulted in improved correspondence with the analytical solution.

Part B – The results of Part B are compared graphically with the analytic solution in Figure 5.6. The calculated radius of the elastic-plastic interface, $R_0^*$, based on the analytic solution is 1.164. For UDEC, the corresponding radius was determined following the procedure described in Section 3. The radius of the elastic-plastic interface was found to be 1.184 for both the coarse and fine zoning, or error of 1.72%.

Part C – The UDEC results for Part C are presented in terms of lining thrust, $N^*$, and moment, $M^*$, in Figures 5.7 and 5.8. Results shown are for the first quadrant. Results for the other quadrants are similar.
Figure 5.4  Comparison of UDEC results of radial and tangential stresses versus radial distance along a line $\theta = 0^\circ$, with analytical solution for the case of a tunnel in an elastic medium with a biaxial stress field.
Figure 5.5  Comparison of UDEC results of radial and tangential displace-
ments versus radial distance along a line $\theta = 0^\circ$, with analytical
solution for the case of a tunnel in an elastic medium with a
biaxial stress field
Figure 5.6  Comparison of UDEC results of radial and tangential stresses versus radial distance along a line $\theta = 0^\circ$, with analytical solution for the case of a tunnel in Mohr-Coulomb medium with a biaxial stress field
Figure 5.7 Comparison of UDEC results for lining thrust with analytical solution for the case of a lined tunnel in an elastic medium with a biaxial stress field
Figure 5.8  Comparison of UDEC results for lining moment with analytical solution for the case of a lined tunnel in an elastic medium with a biaxial stress field

(a/ℓz = 5)

(a/ℓz = 10)
5.5 References


5.6 Listing of Data Files

Example 5.1 CYLBE_IN.DAT

;---------------------------------------------------------
; Verification test:
; Circular tunnel problems using boundary elements
;
; Input data
;---------------------------------------------------------
ca cont_cb.fis
;
; --- friction angle ---
set phi 20.
;
; --- dilation angle ---
set ksi 20.
;
; --- Poisson’s ratio ---
set poisson 0.2
;
; --- lateral stress coefficient ---
set k_min 0.5
;
; --- vertical stress ---
set sigma_q 0.75
;
; --- support - thickness normalized with radius, Young’s modulus
; --- normalized with Young’s modulus of medium and Poisson’s ratio
; set thick_a 0.10
set young_s 3.33333
set poisson_s 0.2
;
; --- zone size and number of support elements ---
set a_z 10.
set ndiv 96
;
; --- convergence criterion ---
;
UDEC Version 5.0
set conv_rat 2.e-6
;
ca cylbe.dat
new

cal cont_cb.fis
set phi 20.
set ksi 20.
set poisson 0.2
set k_min 0.5
set sigma_q 0.75
set thick_a 0.10
set young_s 3.33333
set poisson_s 0.2
set ndiv 48
set a_z 5.
set conv_rat 2.e-6
ca cylbe.dat
ret

Example 5.2  CONT_CB.FIS

;---------------------------------------------------------
; Verification test:
; Circular tunnel problems using boundary elements
;
; Model preparation, control and post-processing
;---------------------------------------------------------
;
def sconv
;
; --- converts real numbers into strings ---
; --- arguments: number - realn and number of decimal digits - ndig ---
;
    irealn = int(realn)
    srealn = string(irealn)
    if ndig = 0 then
        exit
    end_if
    srealn = string(irealn)+'.'
    realn = realn - irealn
    loop i (1,ndig)
        realn = 10.*realn
        irealn = int(realn)
        srealn = srealn+string(irealn)
realn = realn - irealn
end_loop
end

def setup
;
; --- titles of simulations ---
;
realn = a_z
ndig = 0
sconv
sa_z = srealn
;
realn = k_min
ndig = 1
sconv
sk_min = srealn
;
realn = poisson
ndig = 2
sconv
spoisson = srealn
;
realn = poisson_s
ndig = 2
sconv
spoisson_s = srealn
;
realn = thick_a
ndig = 2
sconv
sthick_a = srealn
;
realn = young_s
ndig = 0
sconv
syoung_s = srealn
;
realn = phi
ndig = 0
sconv
sphi = srealn
;
realn = ksi
ndig = 0
sconv

UDEC Version 5.0
sksi = srealn
;
realn = sigma_q
ndig = 2
sconv
ssigma_q = srealn
;
run_t1 = ‘Tunnel, elastic medium nu = ‘+spoisson
run_t1 = run_t1+’, s2/s1 = ‘+sk_min
run_t1 = run_t1+’, a/z = ‘+sa_z
;
run_t2 = ‘Supported tun.: Es/E=’+syoung_s
run_t2 = run_t2+’, nus=’+spoisson_s’, d/a=’+sthick_a
run_t2 = run_t2+’, el. rock nu=’+spoisson
run_t2 = run_t2+’, s2/s1=’+sk_min
run_t2 = run_t2+’, a/z=’+sa_z
;
run_t3 = ‘Tunnel, M.-C. medium nu=’+spoisson
run_t3 = run_t3+’, fric=’+sphi
run_t3 = run_t3+’, dil=’+sksi
run_t3 = run_t3+’, s1/q=’+ssigma_q
run_t3 = run_t3+’, a/z=’+sa_z
;
run_t4 = ‘Supported tun. el.-pl. rock nu=’+spoisson
run_t4 = run_t3+’, fric=’+sphi
run_t4 = run_t3+’, dil=’+sksi
run_t4 = run_t3+’, s1/q=’+ssigma_q
run_t4 = run_t3+’, a/z=’+sa_z
;

nam_t1 = ‘ctel’+sa_z’+’z’
nam_t1g = nam_t1+’g.sav’
nam_t1e = nam_t1+’e.sav’
nam_t1f = nam_t1+’f.sav’
nam_t2 = ‘ctsu’+sa_z’+’z’
nam_t2e = nam_t2+’e.sav’
nam_t3 = ‘ctep’+sa_z’+’z’
nam_t3e = nam_t3+’e.sav’
nam_t3f = nam_t3+’f.sav’
nam_t4 = ‘cteps’+sa_z’+’z’
nam_t4f = nam_t4+’f.sav’
;
; --- UDEC parameters ---
;
a = 5.
r_a = 6.
q = 20.

UDEC Version 5.0
young_q = 300.
young = young_q*q
shear_m = 0.5*young/(1.+poisson)
bulk_m = 0.333333*young/(1.-2.*poisson)
sinphi = sin(phi*degrad)
cosphi = cos(phi*degrad)
sinksi = sin(ksi*degrad)
cosksi = cos(ksi*degrad)
coh_v = 0.5*q*(1.-sinphi)/cosphi

; sigma_2 = -sigma_q*q
sigma_1 = sigma_2/k_min

; --- principal stresses are 30 deg clockwise from the horizontal ---

alpha = 30
cosal = cos(alpha*degrad)
sinal = sin(alpha*degrad)
sigma_xx = sigma_1*cosal*cosal+sigma_2*sinal*sinal
sigma_xy = (sigma_2-sigma_1)*sinal*cosal
sigma_yy = sigma_2*cosal*cosal+sigma_1*sinal*sinal
sigma_zz = 0.5*sigma_yy

; --- support properties ---

thick_s = thick_a*a
young_s = young_s*young
ndiv = ndiv
shear_s = 0.5*young_s/(1.+poisson_s)
bulk_s = 0.333333*young_s/(1.-2.*poisson_s)
young_sp = young_s/(1.-poisson_s*poisson_s)
poisson_sp = poisson_s/(1.-poisson_s)

; --- parameters used in generation of model geometry ---

conv_rat = conv_rat
a2 = 2.*a
a4 = 4.*a
large = bulk_m*1e10
if a_z < 5. then
  a_z = 5.
end_if
if r_a < 5. then
  r_a = 5.
end_if
zone = a/a_z
\[ j_{\text{stiff}} = 10.*\left(\text{bulk}_m + 4.*\text{shear}_m/3.\right)/\text{zone} \]
\[ c_{\text{stiff}} = j_{\text{stiff}}*2.*\pi*a/\text{ndiv} \]
\[ \text{mod}_\text{size} = a*r_a \]
\[ \text{ref}_\text{poin} = 3.*\text{mod}_\text{size} \]
\[ _\text{ref}_\text{poin} = -\text{ref}_\text{poin} \]
\[ a_{\text{half}} = 0.5*a \]
\[ \text{small}_\text{len} = 0.2*\text{zone} \]
\[ n_{\text{small}_\text{len}} = -\text{small}_\text{len} \]
\[ a_{\text{l}} = a - \text{small}_\text{len} \]
\[ a_{\text{u}} = a + \text{small}_\text{len} \]
\[ \text{mod}_\text{size}_u = 1.02* \text{mod}_\text{size} \]
\[ \text{mod}_\text{size}_l = 0.98* \text{mod}_\text{size} \]
\[ \text{rrat} = 1.3 \]
\[ \text{conv}_\text{rat}_{10} = 0.1*\text{conv}_\text{rat} \]
end
end_def

def rings

--- generation of circular rings ---

d1 = 0.5*a_z*\text{zone}
if \text{rrat} \# 1. then

nrings = ln(1.-mod_size-a)/(1.-rrat)/d1)/ln(rrat)
nrings = int(nrings)
coef = nrings*ln(rrat)
coef = exp(coef)
d1 = (mod_size-a)/(1.-rrat)/(1.-coef)
pow = exp(nrings*ln(rrat))
mod_size = a + d1*(1.-pow)/(1.-rrat)
else

nrings = int((mod_size-a)/d1)
d1 = (mod_size-a)/nrings
end_if

_mod_size = -\text{mod}_\text{size}

command
block circular 0 0 \text{mod}_\text{size} 48

command

_tunnel 0 0 \text{dr} 48
end_command

\[ \text{dr} = \text{dr} + \text{di} \]
Example 5.3  CYLBE.DAT

;---------------------------------------------------------
; Verification test:
; Circular tunnel problems using boundary elements
;---------------------------------------------------------
; --- problem setup ---
setup
title run_t1
round = 0.002
;
; --- generation of geometry
;
; rings
; gzones
; save nam_tlg
;
; --- set stresses - the stress field makes the major principal
; --- stress orientated 30 degrees clockwise from the horizontal axis
;
; bound stress sigma_xx sigma_xy sigma_yy
; insitu stress sigma_xx sigma_xy sigma_yy szz sigma_zz
;
; --- define material properties ---
;
; prop mat=1 den=1 k=bulk_m g=shear_m
;
; --- glue joints ---
;
; prop mat=1 jkn=j_stiff jks=j_stiff jcoh=large jten=large
;
; set dscan 10000
;
; --- auto damping ---
;
; damp auto
;
; --- histories ---
;
; hist vmax
;
; --- displacement and stress histories ---
;
; hist n=20 xdis a 0 xdis a2 0 xdis a4 0
; hist n=20 ydis a 0 ydis a2 0 ydis a4 0
; hist n=20 sxx a 0 sxx a2 0 sxx a4 0
; hist n=20 syy a 0 syy a2 0 syy a4 0
; hist n=20 sxy a 0 sxy a2 0 sxy a4 0
;
; --- cycle until new equilibrium ---
;
; solve rat conv_rat
;
; --- save the equilibrium state ---
;
; save nam_t1e
;
; --- excavate ---
;
delete ann 0 0 0 a
;
; --- set boundary elements ---
;
be gen _mod_size mod_size _mod_size mod_size
be mat=1
be fix 0 _ref_poin ref_poin 0
be stiff
;
; --- cycle until new equilibrium ---
;
solve rat conv_rat
;
; --- check equilibrium ---
;
;pl pe his 0
;pl pe his 1
;pl pe his 2 3 4 5 6 7
;
save nam_t1f
;
;----------------------------------------------------------
; Part C: restart from first equilibrium (before excavation)
; Select the appropriate filename.
;----------------------------------------------------------
;
restart nam_t1e
title run_t2
;
; --- excavate ---
;
delete ann 0 0 0 a
;
; --- set support and its properties ---
;
struct gen xc=0 yc=0 fang=0 npoin=ndiv mat=2 thick=thick_s
prop m=2 if_kn=c_stiff if_ks=c_stiff if_coh=large if_tens=large
;
; --- plane stress values ---
;
;prop m=2 st_den=1 st_ymod=young_s st_prat=poisson_s
;
; --- plane strain values ---
;
Circular Tunnel Problems

prop m=2 st_den=1 st_ymod=young_sp st_prat=poisson_sp
;
; --- set boundary elements ---
;
be gen _mod_size mod_size _mod_size mod_size
be mat=1
be fix 0 _ref_poin ref_poin 0
be stiff
;
; --- cycle until new equilibrium ---
;
solve rat conv_rat
;
; --- check equilibrium ---
;
;pl pe his 0
;pl pe his 1
;pl pe his 2 3 4 5 6 7
;
; --- save the supported elastic solution state.
;
save nam_t2e
;---------------------------------------
; Start afresh for elastic-plastic model
;---------------------------------------
; Circular tunnel. (Part B and Part D)
;---------------------------------------
;
; --- use the saved geometry state ---
;
restart nam_t1g
title run_t3
;
; --- Mohr-Coulumb behavior ---
;
ch cons 3
damp auto
;
; --- give state of stresses (hydrostat.) ---
;
bound stress sigma_2 0 sigma_2
insitu stress sigma_2 0 sigma_2 szz sigma_zz
;
; --- set material properties ---
prop mat=1 den=1 k=bulk_m g=shear_m
prop mat=1 coh=coh_v ten=large fric=phi dil=ksi
;
--- glue joints ---
;
prop mat=1 jkn=j_stiff jks=j_stiff jcoh=large jten=large
;
--- histories ---
;
hist vmax
hist n=20 xdis a 0 xdis a2 0 xdis a4 0
hist n=20 ydis a 0 ydis a2 0 ydis a4 0
hist n=20 sxx a 0 sxx a2 0 sxx a4 0
hist n=20 syy a 0 syy a2 0 syy a4 0
hist n=20 sxy a 0 sxy a2 0 sxy a4 0
;
--- cycle until new equilibrium ---
;
solve rat conv_rat_10
;
--- check equilibrium ---
;
pl pe his 0
pl pe his 1
pl pe his 2 3 4 5 6 7
;
--- save the equilibrium state ---
;
save nam_t3e
;
--- excavate ---
;
delete ann 0 0 0 a
;
--- set boundary elements ---
;
be gen _mod_size mod_size _mod_size mod_size
be mat=1
be fix 0 _ref_poin ref_poin 0
be stiff
;
--- cycle until new equilibrium ---
;
solve rat conv_rat_10
;
save nam_t3f
;
; --- check equilibrium ---
;
;pl pe his 0
;pl pe his 1
;pl pe his 2 3 4 5 6 7
;----------------------------------------------------------
; Part D: restart from first equilibrium (before excavation)
; Select the appropriate filename.
;----------------------------------------------------------
restart nam_t3e
title run_t4
;
; --- excavate ---
;
delete ann 0 0 0 a
;
; --- set support and its properties ---
;
struct gen xc=0 yc=0 fang=0 npoin=ndiv mat=2 thick=thick_s
prop m=2 if_kn=c_stiff if_ks=c_stiff if_coh=large if_tens=large
;
; --- plane stress values ---
;
;prop m=2 st_den=1 st_ymod=young_s st_prat=poisson_s
;
; --- plane strain values ---
;
prop m=2 st_den=1 st_ymod=young_sp st_prat=poisson_sp
;
; --- set boundary elements ---
be gen _mod_size mod_size _mod_size mod_size
be mat=1
be fix 0 _ref_poin ref_poin 0
be stiff
;
; --- cycle until new equilibrium ---
;
solve rat conv_rat_10
;
; --- check equilibrium ---
;
;pl pe his 0
;pl pe his 1
;pl pe his 2 3 4 5 6 7
save nam_t4f
return

Example 5.4  COMP_A.FIS

; fish file to compare the radial, and hoop (tangential) stresses
; and displacements between UDEC and an analytic solution.
; Elastic solution.
rest cte15z.sav
;rest cte110z.sav
def compare
;
; --- find zones along the requested section line ---
;
  n_max = 50
array sec_zo(50)
diff_min = 1e30
iblock = block_head
loop while iblock # 0
  izone = b_zone(iblock)
  loop while izone # 0
    x = z_x(izone)
y = z_y(izone)
    if x > 0 then
      if y < 0 then
        theta = atan2(y,x)
diff = abs(theta*180./pi+alpha)
        if diff<diff_min then
          diff_min = diff
          theta_min = theta
        end_if
      end_if
      izone = z_next(izone)
    end_if
  end_loop
  iblock = b_next(iblock)
end_loop
sinth = sin(theta_min)
costh = cos(theta_min)
i = 0
iblock = block_head
loop while iblock # 0
  izone = b_zone(iblock)
  loop while izone # 0
    x = z_x(izone)

UDEC Version 5.0
\[ y = z_y(izone) \]
if \( x > 0 \) then
  if \( y < 0 \) then
    \[ \theta = \text{atan2}(y, x) \]
    \[ \text{diff} = \text{abs}(\theta - \theta_{\text{min}}) \]
    if \( \text{diff} < 0.1/a_z \) then
      \( i = i+1 \)
    end_if
  end_if
end_if
izone = \( z_{\text{next}}(izone) \)
end_loop
iblock = \( b_{\text{next}}(iblock) \)
end_loop
n_sec = i
;
; --- sort zones with increasing radius ---
;
loop j (1, n_sec-1)
  loop i (1, n_sec-1)
    izone = \( \text{sec}_{\text{zo}}(i) \)
    x = \( z_x(izone) \)
    y = \( z_y(izone) \)
    rad = \( \sqrt{x^2+y^2} \)
    izone1 = \( \text{sec}_{\text{zo}}(i+1) \)
    x1 = \( z_x(izone1) \)
    y1 = \( z_y(izone1) \)
    rad1 = \( \sqrt{x1^2+y1^2} \)
    if \( \text{rad1} < \text{rad} \) then
      \( \text{sec}_{\text{zo}}(i) = \text{izonel} \)
      \( \text{sec}_{\text{zo}}(i+1) = \text{izone} \)
    end_if
  end_loop
end_loop
;
; --- calculate analytic solution and retrieve numerical solution ---
;
loop i (1, n_sec)
  izone = \( \text{sec}_{\text{zo}}(i) \)
  x = \( z_x(izone) \)
  y = \( z_y(izone) \)
  rad = \( \sqrt{x^2+y^2}/a \)
  a_rad = 1/rad
theta = atan2(y, x)
costh = cos(theta)
sinth = sin(theta)
theta = theta + alpha * degrad
cos_2th = cos(2*theta)
sin_2th = sin(2*theta)
part1 = (1. - a_rad*a_rad)
part2 = (1. - 4.*a_rad*a_rad+3.*a_rad*a_rad*a_rad*a_rad)*cos_2th
ytable(1,i) = part1 + part2*(1.-k_min)/(1.+k_min)
part1 = (1.+a_rad*a_rad)
part2 = (1.+3.*a_rad*a_rad*a_rad*a_rad)*cos_2th
ytable(2,i) = part1 - part2*(1.-k_min)/(1.+k_min)
part2 = (1.+2.*a_rad*a_rad-3.*a_rad*a_rad*a_rad*a_rad)*sin_2th
ytable(3,i) = -part2*(1.-k_min)/(1.+k_min)
part1 = a_rad
part2 = a_rad*(4.*(1.-poisson)-a_rad*a_rad)*cos_2th
dis_r = part1 + (1.-k_min)*part2/(1.+k_min)
ytable(4,i) = dis_r
part2 = a_rad*(2.*(1.-2.*poisson)+a_rad*a_rad)*sin_2th
dis_t = -(1.-k_min)*part2/(1.+k_min)
ytable(5,i) = dis_t
ytable(6,i) = sqrt(dis_r*dis_r+dis_t*dis_t)
xtable(1,i) = rad
xtable(2,i) = rad
xtable(3,i) = rad
xtable(4,i) = rad
xtable(5,i) = rad
xtable(6,i) = rad

str_sc = 0.5*(sigma_1+sigma_2)
dis_sc = 0.5*str_sc*a/shear_m

sig_x = z_sxx(izone)/str_sc
sig_y = z_syy(izone)/str_sc
sig_xy = z_sxy(izone)/str_sc

igp1 = z_gp(izone, 1)
igp2 = z_gp(izone, 2)
igp3 = z_gp(izone, 3)

u_x = 0.33333*(gp_xdis(igp1)+gp_xdis(igp2)+gp_xdis(igp3))/dis_sc
u_y = 0.33333*(gp_ydis(igp1)+gp_ydis(igp2)+gp_ydis(igp3))/dis_sc

sig_r = sig_x*costh*costh+2.*sig_xy*sinth*costh+sig_y*sinth*sinth
sig_t = sig_x*sinth*sinth+2.*sig_xy*sinth*costh+sig_y*costh*costh
sig_rt = (sig_y-sig_x)*sinth*costh+sig_xy*(cosh*cosh-sinth*sinth)
\begin{verbatim}
;
  \[ u_r = u_x \cosh + u_y \sinh \]
  \[ u_t = u_y \cosh - u_x \sinh \]
;
  xtable(11,i) = rad
  ytable(11,i) = sig_r
  xtable(12,i) = rad
  ytable(12,i) = sig_t
  xtable(13,i) = rad
  ytable(13,i) = sig_rt
  xtable(14,i) = rad
  ytable(14,i) = u_r
  xtable(15,i) = rad
  ytable(15,i) = u_t
  xtable(16,i) = rad
  ytable(16,i) = \sqrt{u_r^2 + u_t^2}
end_loop
end

compare
label table 1
Radial Stress - Analytic
label table 2
Hoop Stress - Analytic
label table 3
Shear Stress - Analytic
label table 4
Radial Displacement - Analytic
label table 5
Hoop Displacement - Analytic
label table 11
Radial Stress - UDEC
label table 12
Hoop Stress - UDEC
label table 13
Shear Stress - UDEC
label table 14
Radial Displacement - UDEC
label table 15
Hoop Displacement - UDEC
ret
\end{verbatim}
Example 5.5  COMP_B.FIS

; fish file to compare the radial and hoop stresses between
; UDEC and an analytic solution.
; Plastic solution.
rest ctep5z.sav
;rest ctep10z.sav
def compare
;
; --- find zones along the requested section line ---
;
    n_max = 50
    array sec_zo(50)
    diff_min = 1e30
    iblock = block_head
    loop while iblock # 0
        izone = b_zone(iblock)
        loop while izone # 0
            x = z_x(izone)
            y = z_y(izone)
            if x > 0 then
                if y < 0 then
                    theta = atan2(y,x)
                    diff = abs(theta*180./pi+alpha)
                    if diff<diff_min then
                        diff_min = diff
                        theta_min = theta
                    end_if
                end_if
            end_if
            izone = z_next(izone)
        end_loop
    iblock = b_next(iblock)
end_loop

    sinth = sin(theta_min)
    costh = cos(theta_min)
    i = 0
    iblock = block_head
    loop while iblock # 0
        izone = b_zone(iblock)
        loop while izone # 0
            x = z_x(izone)
            y = z_y(izone)
            if x > 0 then
                if y < 0 then

UDEC Version 5.0
\[
\theta = \text{atan2}(y, x)
\]
\[
diff = \text{abs}(\theta - \theta_{\text{min}})
\]
\[
\text{if } \text{diff}<0.1/a_z \text{ then }
\quad i = i+1
\quad \text{if } i < n_{\text{max}} \text{ then }
\quad \text{sec}_z(i) = \text{izone}
\quad \text{end_if}
\text{end_if}
\text{end_if}
\text{end_if}
izone = z_{\text{next}}(izone)
\text{end_loop}
iblock = b_{\text{next}}(iblock)
\text{end_loop}
n_{\text{sec}} = i
;
; \quad \text{--- sort zones with increasing radius ---}
;
\text{loop } j (1, n_{\text{sec}}-1)
\quad \text{loop } i (1, n_{\text{sec}}-1)
\quad \text{izone} = \text{sec}_z(i)
\quad x = z_x(\text{izone})
\quad y = z_y(\text{izone})
\quad \text{rad} = \text{sqrt}(x^2+y^2)
;
\quad \text{izone1} = \text{sec}_z(i+1)
\quad x_1 = z_x(\text{izone1})
\quad y_1 = z_y(\text{izone1})
\quad \text{rad1} = \text{sqrt}(x_1^2+y_1^2)
\quad \text{if } \text{rad1}<\text{rad} \text{ then }
\quad \\text{sec}_z(i) = \text{izone1}
\quad \text{sec}_z(i+1) = \text{izone}
\quad \text{end_if}
\quad \text{end_loop}
\quad \text{end_loop}
;
; \quad \text{--- calculate analytic solution and retrieve numerical solution}
;
\quad \sin\phi = \sin(\phi \times \text{degrad})
\quad \cos\phi = \cos(\phi \times \text{degrad})
\quad K_p = (1.+\sin\phi)/(1.-\sin\phi)
\quad \text{plas}_a = 2.\times(\sigma_q\times(K_p-1.)+1.)/(K_p+1.)
\quad \text{plas}_a = \exp(ln(\text{plas}_a)/(K_p-1.))
\quad \text{sig}_o = -(2.\times\sigma_q-1.)/(K_p+1.)
;
\text{loop } i (1, n_{\text{sec}})
izone = sec_zo(i)
x = z_x(izone)
y = z_y(izone)
rad_a = sqrt(x*x+y*y)/a
xtable(1,i) = rad_a
xtable(2,i) = rad_a

; if rad_a < plas_a then
  pow_r = (Kp-1)*ln(rad_a)
pow_r = exp(pow_r)
ytable(1,i) = (1.-pow_r)/(Kp-1.)
ytable(2,i) = (1.-Kp*pow_r)/(Kp-1.)
else
  plas_r = plas_a/rad_a
  ytable(1,i) = -sigma_q+(sigma_q+sig_o)*plas_r*plas_r
  ytable(2,i) = -sigma_q-(sigma_q+sig_o)*plas_r*plas_r
end_if

; sig_x = z_sxx(izone)/q
sig_y = z_syy(izone)/q
sig_xy = z_sxy(izone)/q

; sig_r = sig_x*cosh*cosh+2.*sig_xy*sinh*cosh+sig_y*sinh*sinh
sig_t = sig_x*sinh*sinh-2.*sig_xy*sinh*cosh+sig_y*cosh*cosh
sig_rt = (sig_y-sig_x)*sinh*cosh+sig_xy*(cosh*cosh-sinh*sinh)

; xtable(11,i) = rad_a
ytable(11,i) = sig_r
xtable(12,i) = rad_a
ytable(12,i) = sig_t
end_loop

; plarea = 0.
toarea = 0.
iblock = block_head
loop while iblock # 0
  izone = b_zone(iblock)
  loop while izone # 0
    gp1 = z_gp(izone,1)
gp2 = z_gp(izone,2)
gp3 = z_gp(izone,3)
x1 = gp_x(gp1)
x2 = gp_x(gp2)
x3 = gp_x(gp3)
y1 = gp_y(gp1)
y2 = gp_y(gp2)
y3 = gp_y(gp3)
zoarea = abs(0.5*((x2-x1)*(y3-y1)-(x3-x1)*(y2-y1)))

; --- calculating total area of the model and area of yielding zones ---
if z_state(izone) # 0 then
  plarea = plarea+zoarea
end_if
  toarea = toarea+zoarea
  izone = z_next(izone)
end_loop
iblock = b_next(iblock)
end_loop
aplas_a = sqrt(plarea*(mod_size*mod_size/(a*a)-1.)/toarea+1.)
end
compare
label table 1
Radial Stress - Analytic
label table 2
Hoop Stress - Analytic
label table 11
Radial Stress - UDEC
label table 12
Hoop Stress - UDEC
ret

Example 5.6  COMP.C.FIS

; fish file to compare the Axial force and bending moment between
; UDEC and an analytic solution.
res ctsu5ze.sav
;res ctsu10ze.sav
; --- Define the constants ---
def cons
  br = bulk_m
  sr = shear_m
  E = young
  nu = poisson
  nu_s = poisson_s
  Es = young_s
  Ar = thick_s
  II = Ar * Ar * Ar / 12.
  cs = E * a * (1. - nu_s*nu_s) / (Ar * Es * (1. - nu*nu))
  fs = E * a*a*a * (1. - nu_s*nu_s) / (Es * II * (1. - nu*nu))
  a0 = cs * fs * (1. - nu) / (cs + fs + cs * fs * (1.-nu))

UDEC Version 5.0
be = ((6.+fs)*cs*(1.-nu)+2.*fs*nu)/(3.*fs+3.*cs+2.*cs*fs*(1.-nu))
b2 = cs*(1.-nu)*0.5/(cs*(1.-nu)+4.*nu-6.*be-3.*be*cs*(1.-nu))
a2 = be*b2
c1 = 0.5*(1.+1./k_min)*(1.-a0)
c2 = 0.5*(1.-1./k_min)*(1.+2.*a2)
c3 = 0.25*(1.-1./k_min)*(1.-2.*a2+2.*b2)
n0 = sigma_2 * a
m0 = n0 * a
end
cons
ca str.fin
def compare

nin = str_elem_head
i = 0
ern = 0.0
erm = 0.0
loop while nin # 0
   p1 = imem(nin + $SELN1)
p2 = imem(nin + $SELN2)
tanteta = (fmem(p2+$SNDY)+fmem(p1+$SNDY))
tanteta = tanteta/(fmem(p2+$SNDX)+fmem(p1+$SNDX))
teta = atan(tanteta)+alpha*degrad
   if teta >= -3*pi/ndiv then
   if teta <=(ndiv/2+3)*pi/ndiv then
      i = i + 1
      tetad = teta / degrad
      temp1 = cl + c2 * cos(2.*teta)
temp2 = fmemb(nin+$SELFAX) / n0
      xtable(1,i) = tetad
      ytable(1,i) = temp1
      xtable(11,i) = tetad
      ytable(11,i) = temp2
      temp3 = 100. * abs(temp1 - temp2)
      ern = max(temp3,ern)
tanteta = fmemb(p1+$SNDY) / fmemb(p1+$SNDX)
teta = atan(tanteta)+alpha*degrad
      tetad = teta / degrad
      temp1 = c3 * cos(2.*teta)
temp2 = -fmemb(nin+$SELM1) / m0
      xtable(2,i) = tetad
      ytable(2,i) = temp1
      xtable(12,i) = tetad
      ytable(12,i) = temp2
      if abs(tetad - 45.) > 45.e-3 then
         temp3 = 100. * abs(temp1 - temp2)
      else

UDEC Version 5.0
temp3 = 0.0  
end_if  
erm = max(temp3,erm)  
end_if  
end_if  
nin = imem(nin+$SELNEXT)  
end_loop

;  
; --- sort tables ---  
;

loop j (1,i-1)
loop k (1,i-1)
    if xtable(1,k) > xtable(1,k+1) then
        x1 = xtable(1,k+1)
        x2 = xtable(2,k+1)
        x11 = xtable(11,k+1)
        x12 = xtable(12,k+1)
        y1 = ytable(1,k+1)
        y2 = ytable(2,k+1)
        y11 = ytable(11,k+1)
        y12 = ytable(12,k+1)
    
    xtable(1,k+1) = xtable(1,k)
    xtable(2,k+1) = xtable(2,k)
    xtable(11,k+1) = xtable(11,k)
    xtable(12,k+1) = xtable(12,k)
    ytable(1,k+1) = ytable(1,k)
    ytable(2,k+1) = ytable(2,k)
    ytable(11,k+1) = ytable(11,k)
    ytable(12,k+1) = ytable(12,k)
    
    xtable(1,k) = x1
    xtable(2,k) = x2
    xtable(11,k) = x11
    xtable(12,k) = x12
    ytable(1,k) = y1
    ytable(2,k) = y2
    ytable(11,k) = y11
    ytable(12,k) = y12
    end_if

end_loop
end_loop
ern = ern / ytable(1,1)
erm = erm / ytable(2,1)
end
compare
Verification Problems

1. Label table 1
2. Liner Thrust - Analytic
3. Label table 2
4. Liner Moment - Analytic
5. Label table 11
6. Liner Thrust - UDEC
7. Label table 12
8. Liner Moment - UDEC

ret