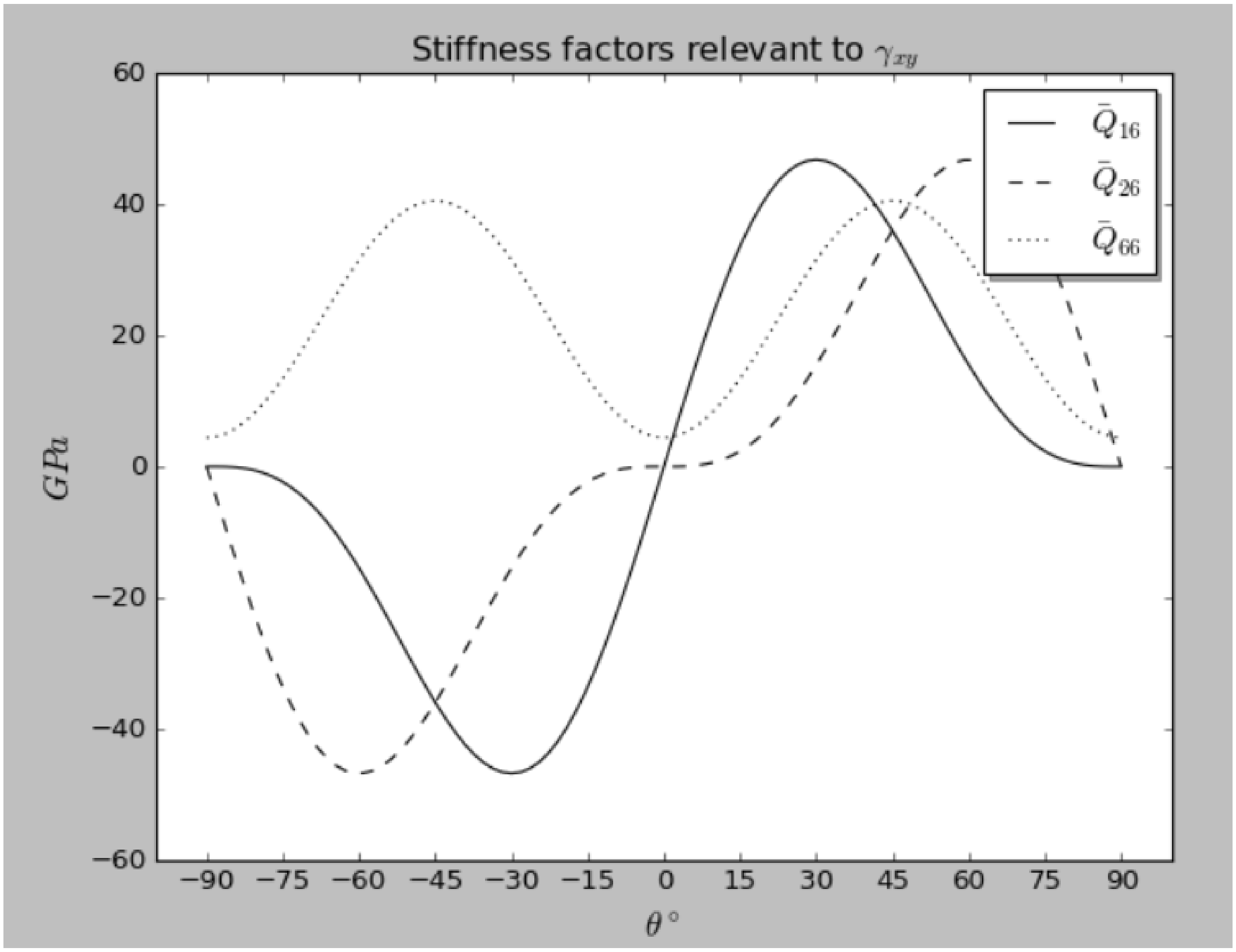
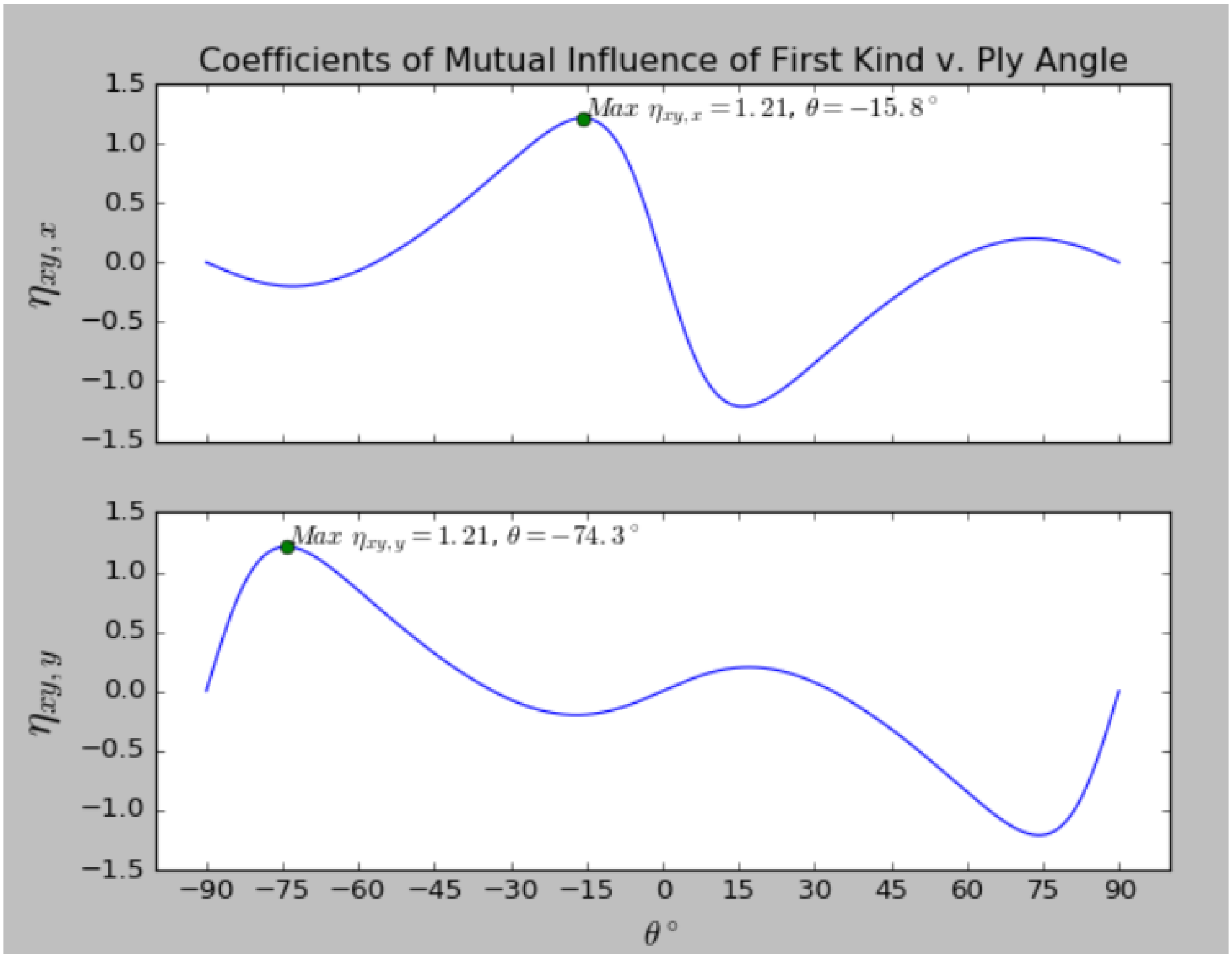
#1


Stiffness factors relevant to γ_{xy}
$\bar{Q}_{16}$
$\bar{Q}_{26}$
$\bar{Q}_{66}$
GPa
θ°
60
40
20
0
−20
−40
−60
−90
−75
−60
−45
−30
−15
0
15
30
45
60
75
90

#3



Both coefficients show symmetry about *Θ=0º*, in which the sign of the coefficient has flipped. Each coefficient also has a maximal value of 1.21, and knowing about the symmetry, each coefficient will have its maximum value 90º away from the minimum value of the other coefficient. $\eta_{xy,x}$ will have its minimum value of -1.21 at 15.8º, 90º away from $\eta_{xy,y}$ max at -74.3º.

```
import matplotlib.pyplot as plt
import numpy as np
import sympy as sp

C = np.array([
    [ 158.   ,  5.64 ,  5.64 ,  0.   ,  0.   ,  0.  ],
    [   5.64 , 15.51 ,  7.21 ,  0.   ,  0.   ,  0.  ],
    [   5.64 ,  7.21 , 15.51 ,  0.   ,  0.   ,  0.  ],
    [   0.   ,  0.   ,  0.   ,  3.2  ,  0.   ,  0.  ],
    [   0.   ,  0.   ,  0.   ,  0.   ,  4.4  ,  0.  ],
    [   0.   ,  0.   ,  0.   ,  0.   ,  0.   ,  4.4 ]], np.float64) * 10**9

def Transform(theta):
    m = np.cos( np.deg2rad(theta) )
    n = np.sin( np.deg2rad(theta) )
    return np.array([
        [m**2, n**2, 2*m*n],
        [n**2, m**2, -2*m*n],
        [-m*n, m*n,  m**2 - n**2]], np.float64)

S = np.linalg.inv(C)

s = np.zeros((3,3), np.float64)
s[:2,:2] = S[:2, :2]
s[-1, -1] = S[-1, -1]

theta = np.linspace(-90, 90, 100)

T = Transform(theta)
T_ = np.rollaxis(T, 2)

S_bar = np.einsum('...jk,kl,...lm->...jm', T.T, s, T_) #[S_bar] = [T.T][S][T]

Q_bar = np.linalg.inv(S_bar)

eps_xyz = np.array([[0,0,1]]).T

sigma_xyz = np.dot(Q_bar, eps_xyz).reshape(100, -1)

Q_bar /= 10**9

fig, ax = plt.subplots()

plt.plot(theta, Q_bar[:,0,2], 'k-', label=r'$\bar Q_{16}$')
plt.plot(theta, Q_bar[:,1,2], 'k--', label=r'$\bar Q_{26}$')
plt.plot(theta, Q_bar[:,2,2], 'k:', label=r'$\bar Q_{66}$')

plt.title(r'Stiffness factors relevant to $\gamma_{xy}$')
```

```
plt.xlabel(r'$\theta^\circ$', fontsize=15)
plt.ylabel(r'$GPa$', fontsize=15)

legend = ax.legend(loc='upper right', shadow=True)
plt.xticks(np.linspace(-90, 90, 13))

plt.show()
```

```
import numpy as np
import matplotlib.pyplot as plt
import sympy as sp

np.set_printoptions(precision=3)

def Transform(theta):
    m = np.cos( np.deg2rad(theta) )
    n = np.sin( np.deg2rad(theta) )
    return np.array([
        [m**2, n**2, 2*m*n],
        [n**2, m**2, -2*m*n],
        [-m*n, m*n,  m**2 - n**2]], np.float64)

#solve problem assuming plane stress
E1 = 50 * 10**9
E2 = 15.2 * 10**9
v12 = 0.254
G12 = 4.70 *10**9

#using V12/E1 = V21/E2 symmetry
S = np.array([
    [1/E1       , -v12/E1   , 0],
    [-v12/E1    , 1/E2      , 0],
    [0          , 0         , 1/G12]], np.float64)

theta = np.linspace(-90, 90, 1000)

T = Transform(theta)
T_ = np.rollaxis(T, 2)

S_bar = np.einsum('...jk,kl,...lm->...jm', T.T, S, T_) #[S_bar] = [T.T][S][T]

S11_bar = S_bar[:,0,0]
S22_bar = S_bar[:,1,1]
S16_bar = S_bar[:,0,2]
S26_bar = S_bar[:,2,1]

nu_xy_x = S16_bar / S11_bar
nu_xy_y = S26_bar / S22_bar

#https://matplotlib.org/users/mathtext.html
fig, (ax1, ax2) = plt.subplots(2, 1, sharex=True)

ax1.plot(theta, nu_xy_x, label=r'$\eta_{xy,x}$')
ax1.set_title('Coefficients of Mutual Influence of First Kind v. Ply Angle')
ax1.set_ylabel(r'$\eta_{xy,x}$', fontsize=20)
```

```
i = np.where(nu_xy_x == nu_xy_x.max())
pair = (theta[i], nu_xy_x[i])
ax1.plot(*pair, 'o')
ax1.annotate(r'$Max$ $\eta_{xy,x}={%.2f}$, $\theta={%.1f}^\circ$' %pair[::-1],
xy=pair)

ax2.plot(theta, nu_xy_y, label=r'$\eta_{xy,y}$')
ax2.set_ylabel(r'$\eta_{xy,y}$', fontsize=20)
ax2.set_xlabel(r'$\theta^\circ$', fontsize=15)

i = np.where(nu_xy_y == nu_xy_y.max())
pair = (theta[i], nu_xy_y[i])
ax2.plot(*pair, 'o')
ax2.annotate(r'$Max$ $\eta_{xy,y}={%.2f}$, $\theta={%.1f}^\circ$' %pair[::-1],
xy=pair)

plt.xticks(np.linspace(-90, 90, 13))

plt.show()
```