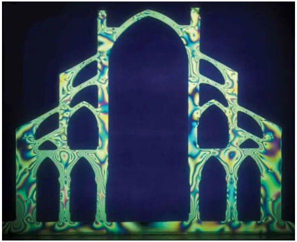
Analyse et estimations des contraintes mécaniques sur un mât de grue

Hadrien Mirade : 31892
Alexis Lacombe



Source : Robert Mark, Princeton University

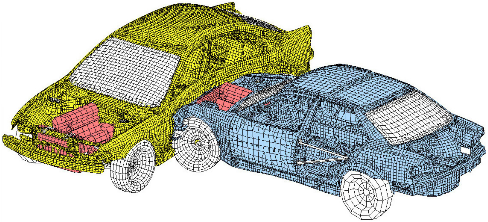
Problématique

Comment évaluer les contraintes résiduelles au sein d'un matériau ? Quelles sont les limites de l'analyse optique d'un modèle en plexiglass permettant de localiser les contraintes mécaniques afin de garantir la fiabilité d'une structure ?

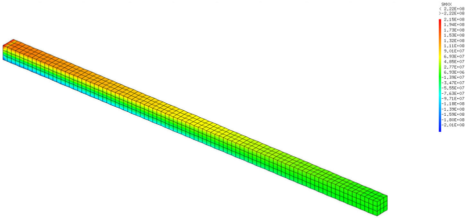
Source : depositphotos.com

Table des matières

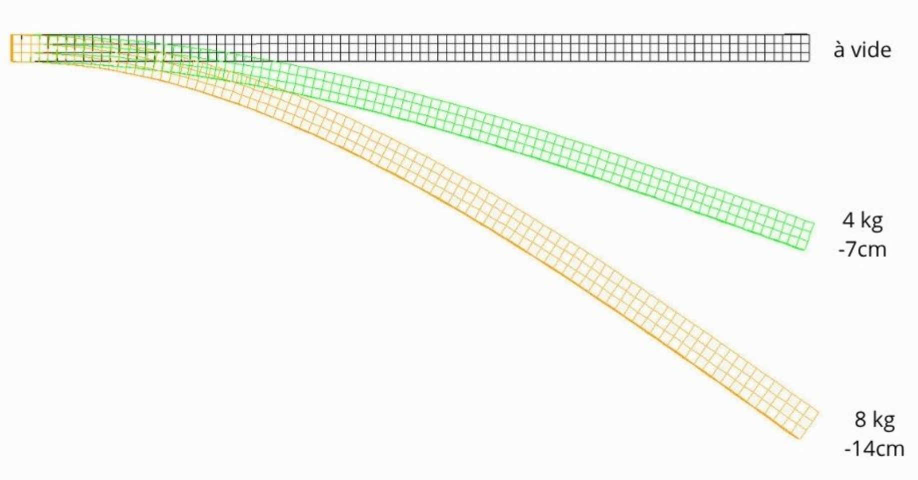
Méthodes des éléments finis



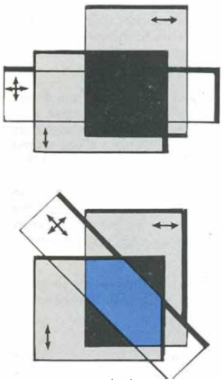
Source : news-cdn.softpedia.com



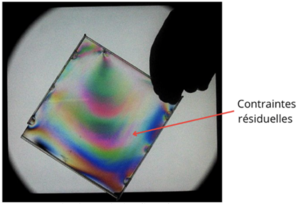
Simulation de la déformation



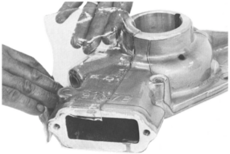
Photoélasticimétrie



cours.polymtl.ca

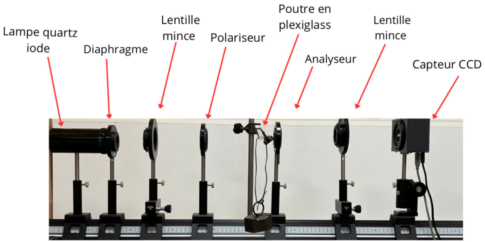


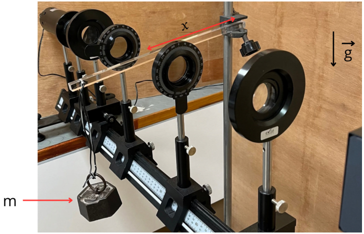
wikipedia.org/wiki/Photoélasticité

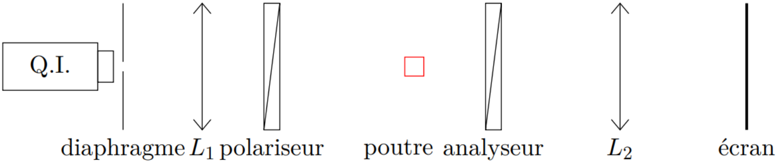


cours.polymtl.ca

Montage de photoélasticimétrie

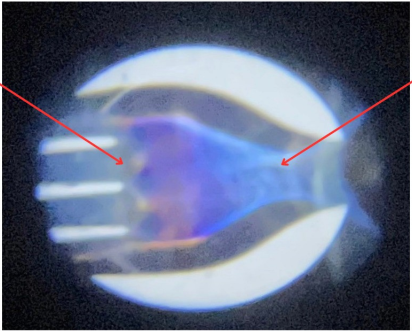






Observation d'une fourchette en plastique

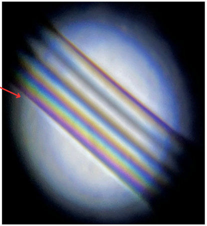
Contraintes résiduelles



Contraintes résiduelles

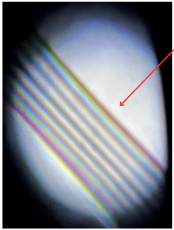
Contraintes avec sollicitations

Couleurs d'interférences en lumière blanche



Poutre en flexion 6kg

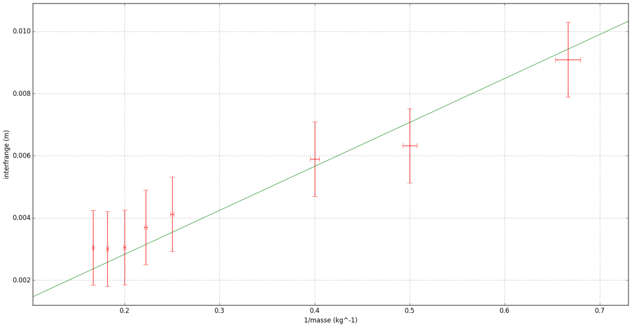
Couleurs d'interférences en lumière blanche



Poutre en flexion 8kg

Interfrange en fonction de l'inverse de la masse

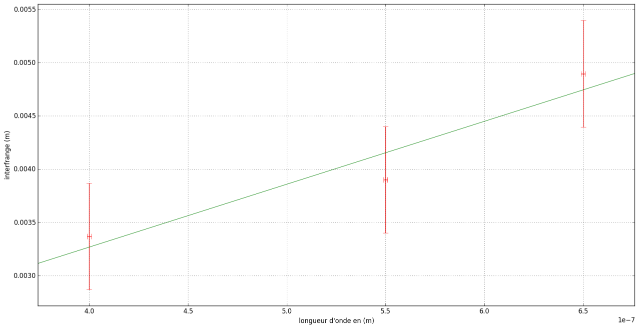
$$i(\frac{1}{m}) = \frac{\lambda.I}{g.\mathcal{C}.e.(L-x)} \cdot \frac{1}{m}$$



$$\text{pente} = 1,3 \pm 0,4 \times 10^{-2}\ m.kg \qquad \mathcal{C} = 2,4 \pm 0,3 \times 10^{-12}\ Pa^{-1}$$
$$\mathcal{C}_{tabulé} = 4,8 \times 10^{-12}\ Pa^{-1}$$

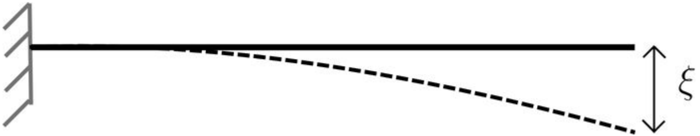
Interfrange en fonction de la longueur d'onde

$$i(\lambda) = \frac{I}{m.g.\mathcal{C}.e.(L-x)}.\lambda$$



$$\text{pente} = 5,9 \pm 0,3 \times 10^3 \qquad \mathcal{C} = 2,1 \pm 0,3 \times 10^{-12}\ Pa^{-1}$$
$$\mathcal{C}_{tabulé} = 4,8 \times 10^{-12}\ Pa^{-1}$$

Relations sur la déflexion

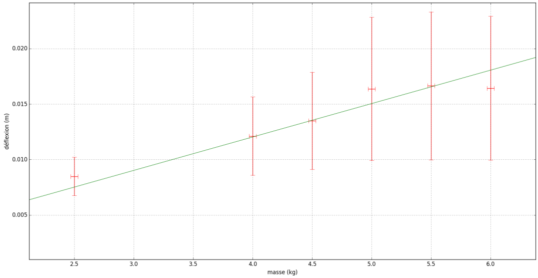


$$\xi(m) = \frac{g.x^2.(3L-x)}{6.E.I}.m$$

$$E = \frac{g.x^2.(3L-x).m}{6.I.\xi}$$

Déflexion en fonction de la masse

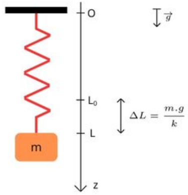
$$\xi(m) = \frac{g.x^2.(3L-x)}{6.E.I}.m$$

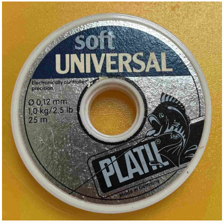


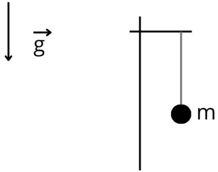
$$\text{pente} = 3,0 \pm 0,1 \times 10^{-3}\ m.kg^{-1} \qquad E = 5,5 \pm 0,2 \times 10^{9}\ Pa$$

$$E_{tabulé} = 2,5 \text{ à } 3,5\ GPa$$

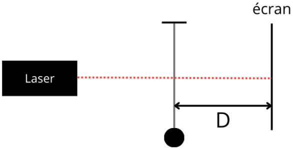
Schémas du montage







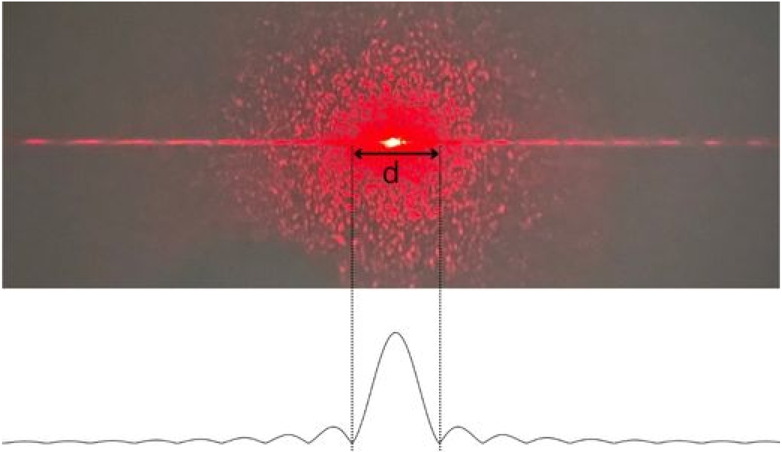
Vue de face



Vue de profil

$$\Delta L = L - L_0 = \frac{4.V_0}{\pi.a^2} - L_0$$

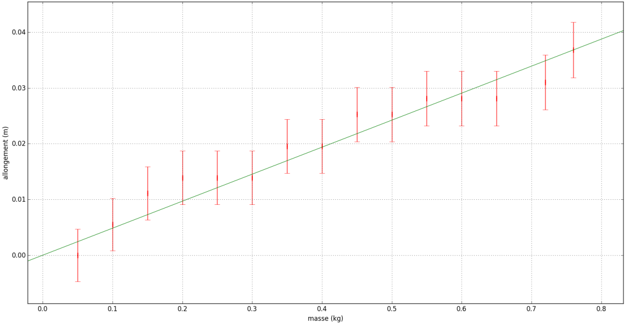
Figure de diffraction



$$a = \frac{2.\lambda.D}{d}$$

Écart de longueur en fonction de la masse

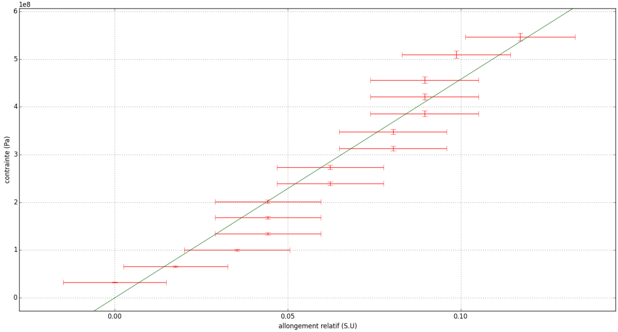
$$\Delta L(m) = \frac{g}{k}.m$$



$$\text{pente} = 4,9 \pm 0,1 \times 10^{-2}\ m.kg^{-1} \qquad k = 2,0 \pm 0,1 \times 10^{2}\ N.m^{-1}$$

Contrainte en fonction de l'allongement relatif

$$\sigma = \frac{F}{S} \qquad \epsilon = \frac{\Delta L}{L_0}$$
$$\sigma(\epsilon) = E.\epsilon$$



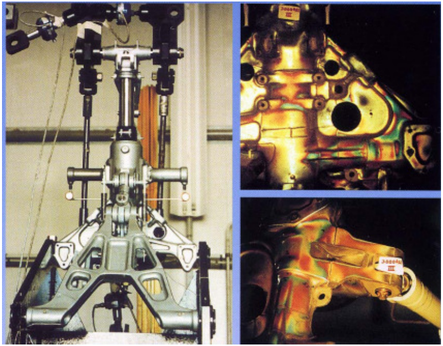
$$E = 4,5 \pm 0,2 \ GPa \qquad E_{tabulé} = 2 \text{ à } 5 \ GPa$$

Comparaison : Photoélasticimétrie vs Diffraction

Méthode	Avantages	Inconvénients
Photoélasticimétrie	• Visualisation qualitative des contraintes • Analyse quantitative des contraintes • Coût expérimental modéré	• Matériaux translucide requis • Limité à des modèles restreints et non adaptés à des structures imposantes
Diffraction	• Mesure précise • Montage basique et facile à mettre en place	• Limité à des pièces de très faible largeur

Conclusion

Source : www.silatecglass.com



Source : cours.polymtl.ca

Définition du solide

```
OPTI 'DIME' 3 'ELEM' 'CUB8';
*définition de la poutre
L1 = (0 0 0) DROI (0 0.01 0) 3;
S1 = L1 TRAN (0 0 0.01) 3;
V1 = S1 VOLU TRAN (0.3 0 0) 90;
*définition des propriétés du matériaux
YOUNGMAT=55.e8;
NUMAT=0.4;
MO1 = MODE V1 'MECANIQUE' 'ELASTIQUE' 'ISOTROPE';
MA1 = MATE MO1 'YOUN' YOUNGMAT 'NU' NUMAT;
*définition du blocage de la poutre
BLOCAGE = BLOQ DEPL S1;
*définition de la tranche recevant un effort
PTS = (V1 coor 1) point maxi;
```

Calculs et affichage

```
*définition de la matrice de rigidité
RIGID = RIGI MO1 MA1;
K4 = RIGID et BLOCAGE ;
K8 = RIGID et BLOCAGE ;
*définition des forces
F0 = FORC PTS (0 0 0);
F4 = FORC PTS (0 0 -39.24);
F8 = FORC PTS (0 0 -78.48);
*résolution de la matrice
U0 = RESO RIGID F0;
U4 = RESO K4 F4;
U8 = RESO K8 F8;
*tracé du champ des contraintes
SIGMA8 = SIGM MO1 MA1 U8;
TRAC SIGMA8 MO1;
*définition de la déformée initiale
DEF0 = DEFO U0 V1 1;
*définition des déformeés finale
DEF4 = DEFO U4 V1 1 'VERT';
DEF8 = DEFO U8 V1 1 'ORANGE';
*tracé des déformées de la poutre
TRAC (DEF0 et DEF4 et DEF8) cach;
*valeurs de déflexion
LIST ((exco U4 uz) MINI);
LIST ((exco U8 uz) MINI);
FIN;
```

Définition des fonctions

```
import numpy as np
import matplotlib.pyplot as plt
import statsmodels.formula.api as sm

def regression(form,var,dat):
    """fonction effectuant la régression linéaire à l'aide de statsmodels"""
    #modèle
    model = sm.ols(formula=form,data=dat)
    #regréssions
    reg = model.fit()
    return reg.params[var]


def trace_regression(X,Y,Xerr,Yerr,coeff,xlab,ylab):
    """fonction traçant les points expérimentaux et la courbe de régression
    à l'aide de matplotlib"""
    assert len(X)==len(Y)
    xmin,xmax=np.min(X),np.max(X)
    xlen=xmax-xmin
    ymin,ymax=np.min(Y),np.max(Y)
    ylen=ymax-ymin
    xmin-=0.1*xlen+max(Xerr)
    xmax+=0.1*xlen+max(Xerr)
    ymin-=0.1*ylen+max(Yerr)
    ymax+=0.1*ylen+max(Yerr)
    x=np.array([(xmin+j*((xmax-xmin)/len(X))) for j in range(len(X)+1)])
    plt.figure()
    plt.xlabel(xlab)
    plt.ylabel(ylab)
    plt.grid()
    plt.errorbar(X,Y, xerr = Xerr,yerr=Yerr,capsize=5,ecolor = 'red',linestyle = 'none')
    plt.plot(x,x*coeff,color='green',linewidth=0.75)
    plt.xlim(xmin,xmax)
    plt.ylim(ymin,ymax)
```

Données de l’expérience de photoélasticimétrie

```python
"""Données de l'expérience de la poutre"""
#Expérience à masse variable
#masses suspendues
mkg=np.array([1.5,2,2.5,4,4.5,5,5.5,6])
umkg=np.array([3e-02 for j in range(len(mkg))])
#interfranges
im=np.array([9.09,6.32,5.89,4.12,3.7,3.05,3,3.04])*1e-3
uim=np.array([1.2e-03 for j in range(len(im))])
#inverse de la masse
Mkg=1/mkg
uMkg=Mkg*(umkg/mkg)
#déflexion
xi=(550e-09*8.40e-03)/(6*2.38e09*1e-02*4.8e-12*0.135*im)
uxi=xi*(uim/im)
#tableaux de données pour les régressions
Mkgim={'Mkg':Mkg,'im':im}
mkgxi={'mkg':mkg,'xi':xi}
#Expérience à longueur d'onde variable
#interfranges
im1=np.array([3.37,3.9,4.898])*1e-3
uim1=np.array([0.5 for j in range(len(im1))])*1e-3
#longueurs d'onde
lambdam=np.array([400,550,650])*1e-9
ulambdam=np.array([1 for i in range(len(lambdam))])*1e-9
#tableau de valeur pour la régression
lambdamim1={'lambdam':lambdam,'im1':im1}
```

Données de l'expérience de diffraction

```
"""Données de l'expérience d'extension du fil"""
#masses suspendues
mkgfil=np.array([0.05,0.1,0.15,0.2,0.25,0.3,0.35,0.4,0.45,0.5,0.55,0.6,0.65,0.72,0.76])
umkgfil=np.array([0.00057 for j in range(len(mkgfil))])
#longueur à vide
L0,uL0=0.3142,0.00017
#largeurs de la tâche centrale
imfil=np.array([2.28,2.3,2.32,2.33,2.33,2.33,2.35,2.35,2.37,2.37,2.38,2.38,2.38,2.39,2.41])*1e-2
uimfil=np.array([0.00017 for j in range(len(imfil))])
#diamètre du fil
a=(635e-09*5)/(imfil)
ua=a*(uimfil/imfil)
#longueur du fil
L=(4.785*10**-9)/(np.pi*((a/2)**2))
uL=L*2*(ua/a+(10**-5)/3.14159)
#allongement du fil
deltaL=L-L0
udeltaL=np.array([max(uL0,uL[j]) for j in range(len(L))])
#contrainte
sigma=(mkgfil*9.81)/(np.pi*(a/2)**2)
usigma=sigma*((10**-5)/3.14159+umkgfil/mkgfil+2*ua/a)
#déformation
epsilon=deltaL/L0
uepsilon=epsilon*(udeltaL/deltaL+uL0/L0)
#tableaux de données pour les régressions
mkgfildeltaL={'mkgfil':mkgfil,'deltaL':deltaL}
epsilonsigma={'epsilon':epsilon,'sigma':sigma}
```

Régressions et tracés

```
"""Régressions"""
C1=regression('xi ~ mkg','mkg',mkgxi)
C2=regression('im ~ Mkg','Mkg',Mkgim)
C3=regression('im1 ~ lambdam', 'lambdam',lambdamim1)
C4=regression('sigma ~ epsilon','epsilon',epsilonsigma)
C5=regression('deltaL ~ mkgfil','mkgfil',mkgfildeltaL)

"""Tracés des régressions"""
trace_regression(mkg,xi,umkg,uxi,C1,'masse (kg)','déflexion (m)')
trace_regression(Mkg,im,uMkg,uim,C2,'1/masse (kg^-1)','interfrange (m)')
trace_regression(lambdam,im1,ulambdam,uim1,C3,'longueur d\'onde en (m)','interfrange (m)')
trace_regression(epsilon,sigma,uepsilon,usigma,C4,'allongement relatif (S.U)','contrainte (Pa)')
trace_regression(mkgfil,deltaL,umkgfil,udeltaL,C5,'masse (kg)','allongement (m)')
plt.show()
```