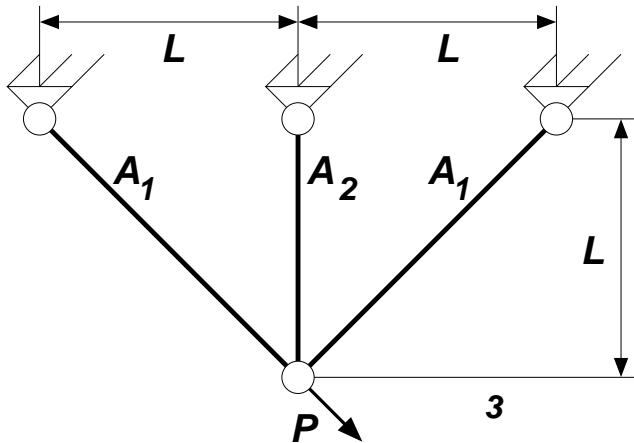


CHAPITRE 3 : APPLICATION AVEC ANSYS (OPTIMISATION PARAMÉTRIQUE)

Dans ce chapitre, on donne quelques éléments pour utiliser ANSYS afin de résoudre des problèmes d'optimisation paramétrique et de forme. L'optimisation topologique n'utilise pas la même approche.

Exemple d'application : premier sujet de TP.

On cherche les sections optimales de ce treillis afin d'avoir un poids minimum, sans dépasser des limites sur les contraintes dans les barres et un déplacement maximum.



Données :

$L = 10 \text{ mm}$ et $E = 10\,000 \text{ Mpa}$,

Sections initiales : $A_i = 2 \text{ mm}^2$ avec $i = 1, 2$.

Contraintes :

Sections minimales $A_i = 0,1 \text{ mm}^2$ avec $i = 1, 2$.

Les limitations sur les contraintes et sur les déplacements sont imposées :

$-15 < \sigma_i < 20$ avec $i = 1, 2, 3$ et $|u_{y_{max}}| \leq 0,01 \text{ mm}$.

On peut alors formuler ce problème sous forme mathématique (P), afin d'identifier ce qui est nécessaire pour mettre en œuvre la résolution avec ANSYS :

$$(P) \left\{ \begin{array}{l} \text{Minimiser : } F(A_1, A_2) = \text{volume du treillis} \\ A_1 \text{ et } A_2 \\ \text{sous les contraintes :} \\ g_1(X) : -15 < \sigma_{\text{barre}_1} < 20 \\ g_2(X) : -15 < \sigma_{\text{barre}_2} < 20 \\ g_3(X) : -15 < \sigma_{\text{barre}_3} < 20 \\ g_4(X) : |U_y(\text{max})| - 0,01 < 0 \\ \sigma_{\text{barre}_1}, \sigma_{\text{barre}_2}, \sigma_{\text{barre}_3} \text{ et } U_y(\text{max}) \\ A_i \in S \text{ défini par :} \\ 0,1 \leq A_i \end{array} \right. \begin{array}{l} \text{fonction objectif} \\ \text{variables de conception} \\ \text{contrainte d'inégalité} \\ \text{contrainte d'inégalité} \\ \text{contrainte d'inégalité} \\ \text{contrainte d'inégalité} \\ \text{variables d'état} \\ \text{domaine de recherche} \end{array}$$

Dans la suite, on détaille le fichier de commandes pour traiter ce problème avec ANSYS.

Le fichier comporte deux parties, **la première partie**, classique, correspond à la formulation par éléments finis du problème avec en plus l'extraction des **variables d'état** du problème : ici les contraintes dans les barres, le déplacement maximum. Sans oublier, la fonction objectif, ici le volume total de la structure (on suppose que le matériau est homogène alors le volume est directement proportionnel au poids du treillis).

La seconde partie, traite de l'optimisation du problème, il s'agit de déclarer les variables de conception (ou de design), d'état et d'indiquer les domaines admissibles (intervalle de variation) de chacune de ces variables. Il faut également définir la fonction objectif, en l'occurrence, on cherche à minimiser le volume du treillis. Enfin, il faut choisir l'algorithme de résolution et donner les paramètres de ces méthodes, essentiellement le nombre maximum des itérations autorisées. Il reste quelques commandes liées au post-traitement : visualisation des solutions et de la valeur des différentes variables, ...

Dans le fichier, toutes les variables définies par l'utilisateur sont soulignées : **a1**, **sig1**, ...

/TITLE, optimisation treillis	! titre du travail
/PREP7	! entrée dans la phase de pré-traitement
*set, a1, 2.	! assignation d'une valeur à un paramètre
*set, a2, 2.	! qui pourra être modifiée
	! en cours d'exécution par ANSYS
	! section initiale des barres = $2mm^2$
alpha1 = 45.	! angle pour la force
*afun, deg	! angles en degrés pour calcul des cosinus et sinus
et, 1, link1	! EF barre 2D : link1
r, 1, a1	! section des barres
r, 2, a2	
MP, ex, 1, 10000.	! matériau : module d'Young
n, 1	! définition des 4 nœuds
n, 2, -10.	
n, 3, 10.	
n, 4, 0, -10.	
real, 1	! section <u>a1</u> pour les barres 1 et 2
en, 1, 3, 4	! création de l'élément de barre 1
en, 2, 2, 4	
real, 2	! section <u>a2</u> pour la barre 3
en, 3, 1, 4	
/eshape, 1	! affiche la section des barres pour suivre
eplot	! l'évolution au cours des itérations d'optimisation
d, 1, all, , , 3, 1	! CL : noeuds 1 à 3 bloqués
f, 4, fx, 20*cos(alpha1)	! force extérieure
f, 4, fy, -20*sin(alpha1)	
/solu	! résolution
solve	
/post1	! post-traitement
etable, sig1, ls, 1	! création de la table des contraintes axiales de nom " <u>sig1</u> "
*get, s1, elem, 1, etable, sig1	! extrait la contrainte de la barre 1 = variable <u>s1</u>
*get, s2, elem, 2, etable, sig1	! idem barre 2
*get, s3, elem, 3, etable, sig1	! idem barre 3
nsort, u, y	! extraction des déplacements uniquement suivant y
*get, v1, sort, , min	! on extrait le déplacement max suivant $y = v1$
	! celui-ci étant négatif, on prend donc le min
	! qui provient de la dernière commande "nsort".
vabs1 = abs(v1)	! on prend enfin la valeur absolu de ce déplacement
etable, volume1, volu	! création de la table " <u>volume1</u> "
	! qui contient le volume de chaque élément
ssum	! somme du contenu de TOUTES les tables
*get, tvolume1, ssum, , item, volume1	! on extrait la somme pour la table " <u>volume1</u> "
	! qui est contenue dans la variable " <u>tvolume1</u> "
LGWRITE, os81tp1, txt	! partie à sauver dans un fichier
	! qui sera utilisé au cours des run d'optimisation
	! écriture du fichier log dans un fichier " <u>os81tp1</u> "
	! extension ".txt"
finish	fin de la formulation du problème par éléments finis

/opt	! début de l'optimisation
opanl, <u>os81tp1</u>, txt	! récupération du fichier descriptif du pb
opvar, <u>a1</u>, dv, 0.1, 10	! définition des variables de conception
opvar, <u>a2</u>, dv, 0.1, 10	! "DV" Design Variable, ici les sections <u>a1</u> , <u>a2</u>
	! et indication du domaine admissible [0,1 ; 10]
	!!! on peut ajouter une tolérance pour les différentes variables
	!!! sachant qu'il y en a une par défaut
opvar, <u>vabs1</u>, sv, 0, 0.01	! définition des variables d'état
opvar, <u>s1</u>, sv, -15, 20	! "SV" State Variable, ici le déplacement max suivant Y
opvar, <u>s2</u>, sv, -15, 20	! et les contraintes dans les barres
opvar, <u>s3</u>, sv, -15, 20	! avec borne de variation (min et max)
opvar, <u>tvolume1</u>, obj	! fonction objectif "OBJ", le volume total
! otype, first	! choix de la méthode de résolution "first" ou "subp"
otype, subp	
opsub, 30	! méthode "subp" avec le nb. d'itérations max (ici 30 par exemple)
opfrst, 30	! méthode "first" avec le nb. d'itérations max (ici 30 par exemple)
opexe	! lancement de l'optimisation
	! ATTENTION : on peut relancer plusieurs fois
	! des run d'optimisation, on poursuit les itérations vers des
	! solutions plus ou moins optimum taper simplement "opexe".
	!!! ATTENTION : en phase de mise au point du fichier de
	!!! commandes, il faut effacer la base de données
	!!! sinon ANSYS réutilise les infos à chaque nouveau
	!!! lancement si le "jobname" n'a pas changé : taper " opclr ".
oplist, all	! affiche la valeur des différentes variables
	! et fonction objectif pour TOUTES les itérations
	! (10 ou 15 itérations, dépend de la méthode)
	! affiche le résultat sous forme d'un tableau
	! le set étoilé est la solution optimale.
	! 2 valeurs : admissible ou non (in/feasible)
	! si infeasible indication de la contrainte violée
	!!! ATTENTION, des contraintes peuvent être violée de peu
	!!! il faut regarder les tolérances par défaut
	!!! ou relancer un "opexe" pour avoir une solution affinée
/axlab, x, nombre iterations	
/axlab, y, volume	
plvaropt, <u>tvolume1</u>	! affiche l'évolution du volume / itérations
/axlab, x, nombre iterations	
/axlab, y, section a1 et a2	
plvaropt, <u>a1</u>, <u>a2</u>	! affiche l'évolution des sections / itérations