

Premières simulations DUST

Alexandre Laleu

December 17, 2024

1 Présentation des données

Toutes les données utilisées ici s'appuient sur les résultats des codes d'optimisation de Boqiao du challenge ONERA 2024.

Ces données proviennent d'algorithmes cherchant à optimiser le bruit, la portance et la puissance consommée en ajustant deux valeurs : la corde (chord) et le vrillage (twist) en construisant un front de pareto.

Les résultats sont les suivants :

2 pales - 9 segments									
corde	2,5	3,34	3,41	3,41	3,35	2,77	2,66	2,15	0,92
twist	45	34	29	23	18	16	14	13	12

3 pales - 9 segments									
corde	1,78	2,37	2,32	1,99	1,52	1,29	1,03	0,96	0,96
twist	31,5	29,6	27,3	24,7	21,7	20,2	18,3	18,3	18

3 pales - 15 segments													
corde	1,3805	2,3844	2,3896	2,3901	2,3852	2,3268	2,0071	1,7388	1,5234	1,3421	1,1753	0,9858	0,9608
twist	27,1339	35,99	33,0447	30,2161	28,3164	26,5049	24,5488	22,6128	21,255	20,0277	19,0277	19,2054	18,2765
RPM = 4500													

4 pales - 15 segments													
corde	1,3786	2,3288	2,3567	2,3368	2,399	2,395	1,8532	1,5522	1,3427	1,1654	1,0077	0,9841	0,9636
twist	26,948	35,9371	35,5405	33,4427	30,5247	29,2961	25,7451	23,3183	22,1836	20,9668	19,7538	19,4007	19,8003
RPM = 4000													

Figure 1: Valeurs de corde et de vrillages optimisées

2 Objectif des simulations

Nous allons ici utiliser la simulation pour trouver les RPM optimum qui nous permettent d'atteindre la portance cible de 4,5N.

Pour ce faire, on lance les simulations pour trois valeurs de RPM distincts pour relever la portance produite à ces RPM et on calcule le polynôme de degré deux passant par ces points sur python.

Par manque de temps (certaines simulations pour les pales à 14 segments prenaient 45 min de calcul à mon ordinateur), je me suis limité aux traitements des tableaux 1 et 3.

3 Brève explication des codes

Les codes calculent la portance et le moment via la méthode des **sectional loads**. C'est à dire que le calcul des différentes grandeurs est effectué indépendamment sur chaque région de la pale définie dans le fichier **ll_wing.in**.

Chaque pale de l'hélice a une envergure de $\frac{23.9}{2} = 11.95\text{cm}$ auxquels il convient de retirer le rayon du moyeu de l'hélice, imposé à 4mm par l'ONERA (confirmez moi ceux du banc d'essai svp).

La simulation prend en compte le moyeu avec la définition de **starting_point = (/0.0, 0.004, 0.0/)** dans le fichier **ll_wing.in**.

L'envergure (**span**) des régions associées au tableau 1 (8 régions) est donc de **span = 1.44cm**.

De même, celle du tableau 3 (13 régions) est **span = 0.888cm**.

Pour chaque simulation, il faut aussi veiller à changer **u_ref = $r_{pale} \cdot \text{RPM}$** dans le fichier **dust.in**.

La vitesse de rotation est quant à elle définie en rad/s dans le fichier **References.in** avec la variable **rot rate**.

Les simulations tournent sur 6 révolutions de la pale pendant son régime stationnaire : **tstart = 0.0, tend = 0.072** dans **dust.in** pour définir les 6 révolutions et **start_res = 301, end_res = 361** dans **dust_post** pour la période en régime stationnaire (ces dernières données sont issus d'anciennes simulations sur lesquelles j'ai repéré l'ordre de grandeur des temps du régime stationnaire).

Casalino_plot_Fz/Mo.py du dossier **python**.

Le profil de chaque région est en **NACA4412**.

Tous les résultats sont traités par python qui va sommer les portances de toutes les régions avec les fichiers. On retiendra la moyenne temporelle de la portance dans l'interpolation avec le polynôme (cf dossier **optimisation_RPM**).

Les valeurs de *twist* et *chord* proviennent des tableaux de Boqiao présentés dans le premier paragraphe.

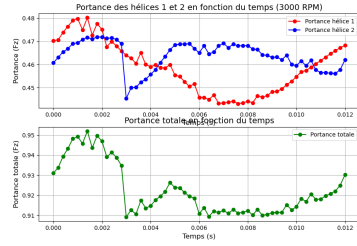
Vous trouverez en page suivante un exemple de **ll_wing.in** :

```

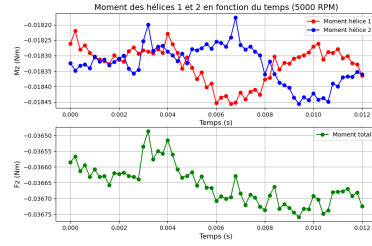
1      mesh_file_type = parametric
2      el_type = 1
3      starting_point = (/0.0, 0.004, 0.0/)
4      reference_chord_fraction = 0.0
5
6      !Section 1
7      chord = 0.025
8      twist = 45
9      airfoil_table = ./airfoil_table/naca4412.c81
10
11     !Region 1
12     span = 0.0144375
13     sweep = 0
14     dihed = 0
15     nelem_span = 7
16     type_span = uniform
17
18     !Section 2
19     chord = 0.0334
20     twist = 34
21     airfoil_table = ./airfoil_table/naca4412.c81
22
23     !Region 2
24     span = 0.0144375
25     sweep = 0
26     dihed = 0
27     nelem_span = 7
28     type_span = uniform
29
30     !Section 3
31     chord = 0.0341
32     twist = 29
33     airfoil_table = ./airfoil_table/naca4412.c81
34
35     !Region 3
36     span = 0.0144375
37     sweep = 0
38     dihed = 0
39     nelem_span = 7
40     type_span = uniform
41
42     !Section 4
43     chord = 0.0341
44     twist = 23
45     airfoil_table = ./airfoil_table/naca4412.c81
46
47     etc ...

```

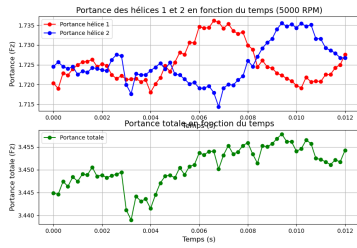
4 Résultats des simulations



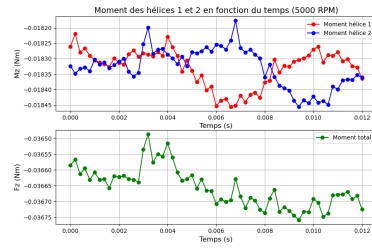
(a) Portance - 2 pales à 3000RPM



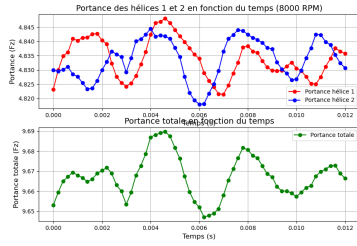
(b) Moment - 2 pales à 3000RPM



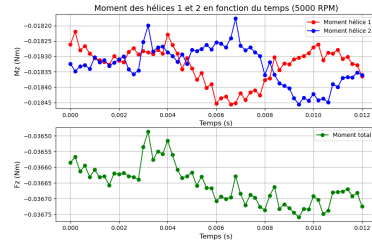
(c) Portance - 2 pales à 5000RPM



(d) Moment - 2 pales à 5000RPM

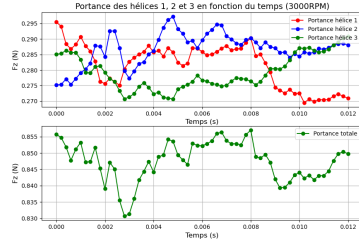


(e) Portance - 2 pales à 8000RPM

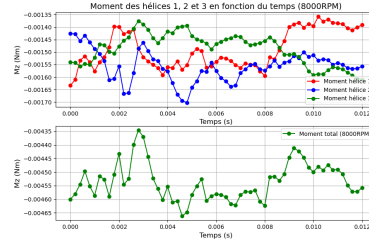


(f) Moment - 2 pales à 8000RPM

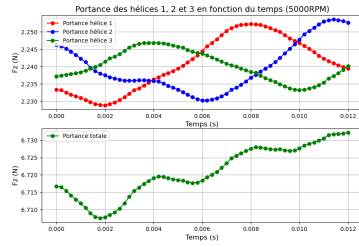
Figure 2: Résultats des simulations de portance et de moment pour 2 pales.



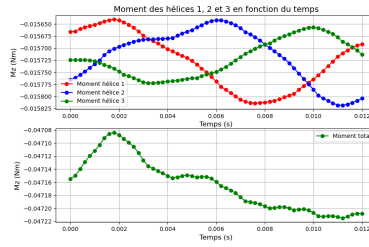
(a) Portance - 3 pales à 3000RPM



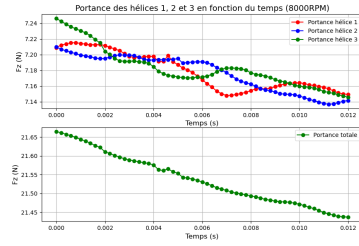
(b) Moment - 3 pales à 3000RPM



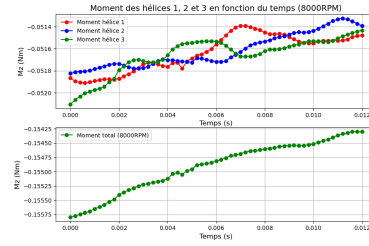
(c) Portance - 3 pales à 5000RPM



(d) Moment - 3 pales à 5000RPM



(e) Portance - 3 pales à 8000RPM



(f) Moment - 3 pales à 8000RPM

Figure 3: Résultats des simulations de portance et de moment pour 3 pales.

On a certaines simulations plus lisses que d'autres mais, en règle générale, les oscillations restent légères. On a juste une décroissance qui ne semble pas s'arrêter pour la portance des 3 pâles à 8000RPM. Uniquement pour cette figure, on retiendra la valeur la plus basse plutôt que la valeur moyenne pour ne pas surestimer la portance à ces RPM.

Il faudrait peut-être refaire la simulation sur plus de révolutions pour ces RPM, ce que je n'ai pas eu le temps de faire jusque ici.

J'ai noté qu'à certaines itération j'ai le warning suivant :

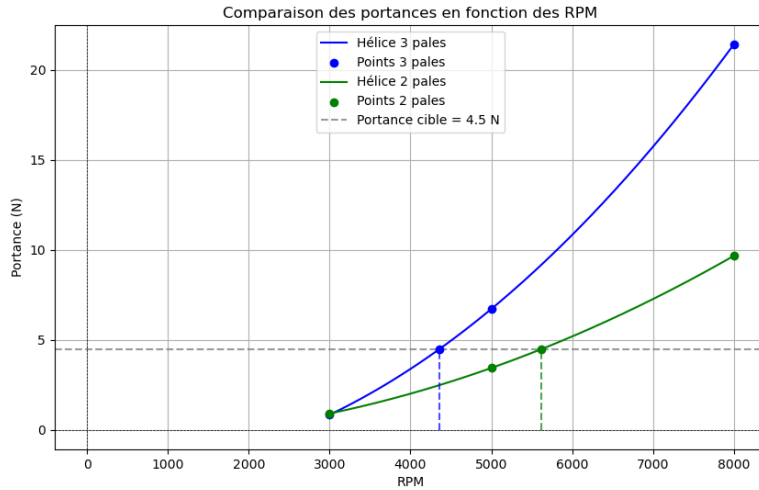
WARNING in "solve_liftlin", in module "mod_liftlin" Lifting lines iterative solution NOT CON

Ce qui m'a un peu inquiété mais, nous le verrons ensuite, les résultats finaux restent cohérents.

5 Interpolation

Avec les résultats précédents, on trace le polynôme de degré deux modélisant la portance fonction des RPM au voisinage de 5000RPM :

On obtient RPM pour l'hélice 3 pales de 4359 et de 5623 RPM pour l'hélice 2



pales. La valeur obtenue est cohérente avec celle obtenue par Boqiao dans son code qui était de 4500RPM pour la 3 pales. Ce qui est encourageant pour les prochaines simulations.

Ces simulations montrent aussi qu'il serait bien judicieux de faire une trois pales qui a une bien meilleure portance.