



5A ModIA

Design a liquid hydrogen powered aircraft

Auteurs :

Karima Ghamnia
Cassandra Mussard
Mickaël Song
Rémi Colin

Superviseur :

Matthias De Lozzo

26 juin 2024

1 Introduction

Dans ce projet, nous explorons la faisabilité d'un avion propulsé par du liquide hydrogène (LH2), inspiré de l'initiative ZEROe d'Airbus, pour promouvoir une aviation durable. Nous avons essayé de trouver les modifications nécessaires au stockage du carburant et à la gestion du poids de l'avion grâce à des modèles de simulation informatiques. Ainsi, notre objectif est de trouver un modèle physique cohérent qui minimise le poids maximal au décollage du LH2PAC tout en respectant les contraintes opérationnelles.

2 Optimisation avec et sans surrogate

Dans un premier temps, nous allons optimiser la conception d'un avion à hydrogène liquide, en se concentrant spécifiquement sur la minimisation du poids maximal au décollage de l'avion, tout en assurant la conformité avec les contraintes opérationnelles définies. Pour cela, nous allons utiliser le modèle complet mais très coûteux. Dans un second temps, nous utiliserons la modélisation de substitution afin d'avoir un modèle, moins précis mais, moins coûteux.

2.1 Définition de la discipline et de l'espace de Design

Le modèle `H2TurboFan` permet de simuler et analyser complètement les performances des avions. Il identifie des paramètres d'entrée et de sortie clés pour garantir une modélisation précise.

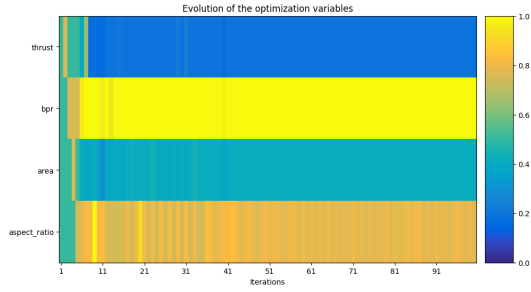
De plus, nous avons défini l'espace de conception `MyDesignSpace` afin d'encapsuler les variations possibles des paramètres de conception pour permettre l'exploration de designs potentiels dans des limites prédéfinies. Toutes les options réalisables sont donc garanties d'être prises en compte lors de l'optimisation.

2.2 Optimisation sur le modèle complet

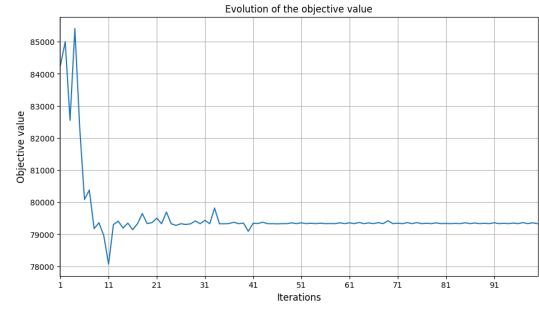
Dans cette section, nous utilisons le modèle `H2TurboFan` pour modéliser l'avion A320. Les simulations obtenues avec cette fonction sont très précises, mais elles impliquent des calculs très coûteux en raison de la haute fidélité du modèle complet. Par conséquent, nous nous limitons à 100 simulations avec ce modèle.

Nous cherchons ensuite à optimiser ce modèle complet en créant un scénario intégrant un plan d'expériences (DoE) qui inclut les contraintes appliquées au modèle. Puis, nous exécutons ce scénario à l'aide d'un algorithme d'optimisation comme suit :

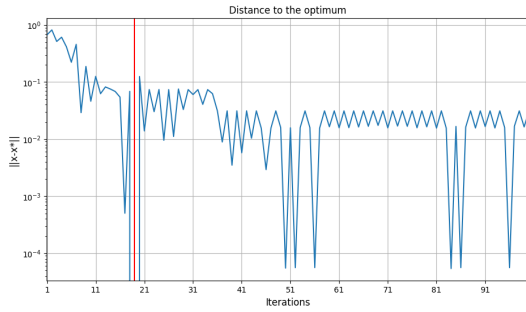
```
scenario.execute("algo": "NLOPT-COBYLA", "max-iter": 100)
```



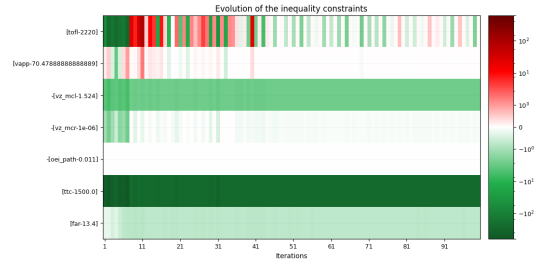
(a) Evolution de l'optimisation des variables



(b) Evolution de la valeur objectif



(c) Distance à l'optimum



(d) Evolution des contraintes d'inégalité

FIGURE 1 – Résultats de l'optimisation du modèle complet

Dans la figure 1, nous retrouvons les résultats de cette optimisation :

- La figure 2a représente l'évolution des quatre variables de conception en fonction des itérations. Nous pouvons observer que les trois premières variables (THRUST, BPR et AREA) se stabilisent très rapidement après quelques ajustements, soit environ après 10 itérations. En revanche, la variable ASPECT-RATIO nécessite davantage d'ajustements et se stabilise après environ 55 itérations.
- La figure 2b représente l'évolution de la valeur objective **MTOW** en fonction des itérations. Nous observons de nombreuses fluctuations au cours des premières itérations, car l'algorithme cherche encore la valeur optimale. La convergence est atteinte après 11 itérations, bien que quelques fluctuations persistent, avant de se stabiliser à partir de l'itération 42. Lorsque le modèle a convergé, MTOW atteint une valeur d'environ 79500 kg. Cette valeur est au dessus des 77000 kg (le minimum pour l'A320). Ceci peut être expliqué par le fait que nous n'avons pas pris en compte certaines contraintes comme par exemple la longueur de la piste de décollage, le facteur d'efficacité aérodynamique, etc.
- La figure 2c montre la distance entre la solution actuelle et la solution optimale. Nous observons que cette distance diminue à mesure que l'algorithme converge, car converger signifie qu'on se rapproche de la solution optimale, c'est donc équivalent à réduire cette distance. La distance atteint son minimum avec quelques fluctuations après environ 40 itérations, ce qui correspond à peu près au nombre

d'itérations nécessaires pour atteindre la convergence.

- La figure 2d illustre l'évolution des différentes contraintes d'inégalité au cours des itérations. Nous observons qu'au début, les contraintes TOFL, VAPP sont violées (car à certaines itérations les valeurs sont au dessus de 10). En revanche, les contraintes VZ MCL, FAR, OEI PATH, VZ MCR et TTC sont rapidement satisfaites dès les premières itérations (les valeurs prises sont inférieures à 10). Toutefois, toutes les contraintes sont respectées et satisfaites après environ 41 itérations, ce qui correspond à la convergence.

2.3 Construction du modèle surrogate

2.3.1 Explication de la création du surrogate

Dans la partie précédente, nous avons utilisé la fonction H2TurboFan qui permet de modéliser le décollage de notre A320. Comme expliqué précédemment, le problème de cette méthode est que les simulations sont très coûteuses.

Nous avons donc utilisé un modèle de substitution (moins coûteux) qui permet d'approximer la discipline H2TurboFan.

Pour approximer la discipline, il est d'abord nécessaire de créer un dataset. Celui-ci va comporter des valeurs pour chaque paramètre du modèle obtenu avec la discipline H2TurboFan.

Une fois ce dataset construit, nous pouvons initialiser un modèle surrogate à l'aide d'un RBFRegressor. C'est un modèle de régression qui va utiliser des fonctions de base radiale pour modéliser les relations non linéaires entre les variables d'entrées et les variables de sorties.

Il sera donc indispensable de regarder si cet approximateur est performant (proche du modèle complet) ou non.

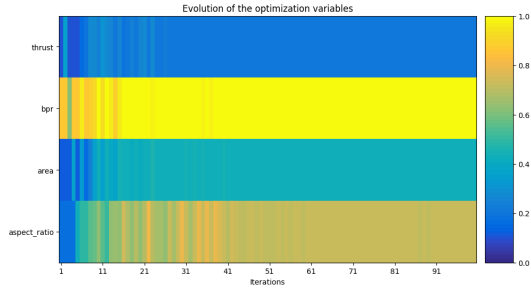
2.4 Résultats sur le sous-modèle

Pour optimiser notre surrogate, nous devons créer un scenario qui va prendre en entrée le design d'expériences (DoE), notre surrogate et les contraintes appliquées au modèle. Le problème s'écrit alors comme suit :

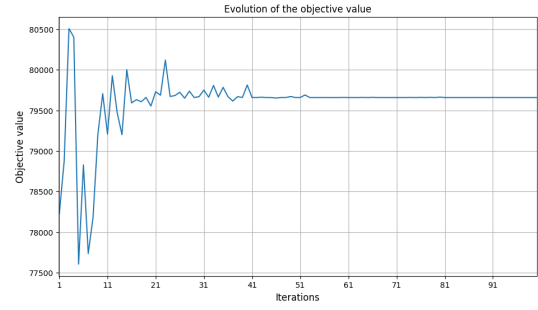
$$\min_{\text{MTOW}} \text{surrogate}(\text{area}, \text{aspect_ratio}, \text{bpr}, \text{drag}, \text{mass}, \text{sfc}, \text{tgi}, \text{thrust}, \text{tvi})$$

Sous les contraintes suivantes :

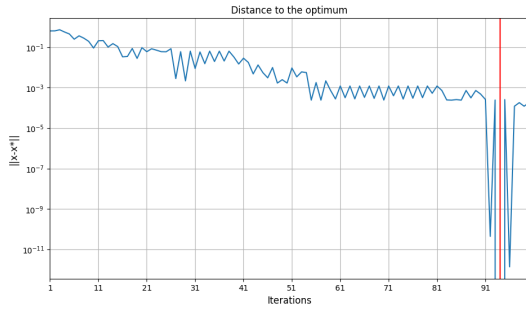
$$\begin{aligned} \text{TOFL} &\leq 2200 \text{ m}, \text{VAPP} \leq 137 \text{ kt}, 300 \text{ ft/min} \leq \text{VZ}_{\text{MCL}}, 0 \text{ ft/min} \leq \text{VZ}_{\text{MCR}}, \\ 1.1\% &\leq \text{OEI}_{\text{PATH}}, \text{TTC} \leq 25 \text{ min}, \text{FAR} \leq 13.4 \end{aligned}$$



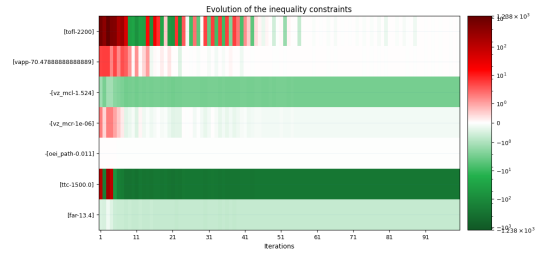
(a) Evolution de l'optimisation des variables



(b) Evolution de la valeur objectif



(c) Distance à l'optimum



(d) Evolution des contraintes d'inégalité

FIGURE 2 – Résultats de l'optimisation du modèle surrogate

La figure 2a représente l'évolution de l'optimisation des variables (THRUST, BPR, AREA, ASPECT_RATIO) au fil des itérations. Pour les variables THRUST, BPR et AREA nous voyons qu'il y a au début des ajustements puis une stabilisation après certaines itérations. Cependant, pour la variable ASPECT_RATIO, nous avons des ajustements continus au cours des itérations et une stabilisation vers la fin. Au final, toutes les variables se sont stabilisées montrant ainsi que l'algorithme a convergé.

La figure 2b représente l'évolution de la valeur objective (MTOW) au fil des itérations. Lorsque le nombre d'itérations est faible, on remarque que la valeur de l'objectif fluctue beaucoup, mais ceci paraît cohérent car l'algorithme explore l'espace des solutions. La valeur de l'objectif commence à se stabiliser autour de 79.500 kg, suggérant que l'algorithme se rapproche d'une solution optimale. Au final, l'algorithme converge en 41 itérations pour une valeur de l'objectif à 79 500 kg. Nous nous rapprochons de la valeur obtenue avec le modèle complet suggérant une bonne approximation de notre modèle surrogate.

La figure 2c représente la distance entre la solution actuelle MTOW et la solution optimale $MTOW^*$. Nous remarquons que cette distance à l'optimum diminue progressivement, ce qui indique que l'algorithme de convergence réduit l'écart avec la solution optimale. On observe des fluctuations autour de 10^{-3} , ce qui peut être dû à des oscil-

lations ou à une convergence non-monotone. Enfin, au bout d'environ 92 itérations, la distance diminue drastiquement, suggérant que l'algorithme a trouvé une solution très proche de l'optimum.

La figure 2d montre l'évolution des différentes contraintes d'inégalité au cours des itérations. Nous remarquons que, au cours des premières itérations, il y a une violation des contraintes TOFL, VAPP, et TTC. Les contraintes VZ_MCL, FAR, VZ_MCR et OEI_PATH sont satisfaites rapidement dès les premières itérations. Nous remarquons que, globalement, malgré la violation de certaines contraintes aux premières itérations, elles sont ensuite satisfaites au fil des itérations.

2.4.1 Validation du modèle surrogate

Afin de vérifier si notre surrogate a de bonnes performances nous calculons deux critères le R2 et la RMSE. Le critère R2 consiste à faire le calcul suivant :

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

Le R2 quantifie à quel point les prédictions d'un modèle de régression se rapprochent des valeurs observées.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

La RMSE permet d'évaluer la qualité des prédictions d'un modèle de régression en comparant la racine de la différence entre les prédictions du modèle et les vraies valeurs.

Ici y_i représente les données obtenues par la discipline H2TuroboFan et \hat{y}_i représente les données approximées de la discipline H2TurboFan par le modèle surrogate.

Nous obtenons les résultats suivants :

$$R2 = [0.92, 0.93, 0.95, 0.55, 0.77, 0.95, 0.95, 0.94]$$

Moyenne R2 : **0.87**.

$$RMSE = [8.58e-16, 8.81e-12, 5.77e-18, 1.64e-12, 7.15e-13, 0.00e+00, 1.43e-15, 1.43e-15]$$

Nous remarquons que le R2 est assez proche de 1 pour chaque paramètre montrant une bonne approximation du modèle surrogate. Nous remarquons cependant que le paramètre mtow n'est pas très bien approximé par le modèle (R2 de 0.54).

Ensuite, pour la RMSE nous voyons que les valeurs sont toutes très proche de 0 indiquant là aussi une bonne approximation du modèle surrogate.

3 Analyse de sensibilité des différents modèles

Après optimisation de notre sous-modèle, des analyses de sensibilité nous permettent de trouver les paramètres clés qui influencent le plus les résultats du modèle pour potentiellement simplifier de nouveau le modèle dans le futur.

3.1 Analyse de Sobol

Nous effectuons maintenant une analyse de Sobol sur les variables technologiques incertaines (correspondant à des imprévus par exemple la météo) afin de déterminer quelle(s) variable(s) aura une influence sur l'optimisation de mtow.

3.1.1 Analyse de Sobol du modèle complet

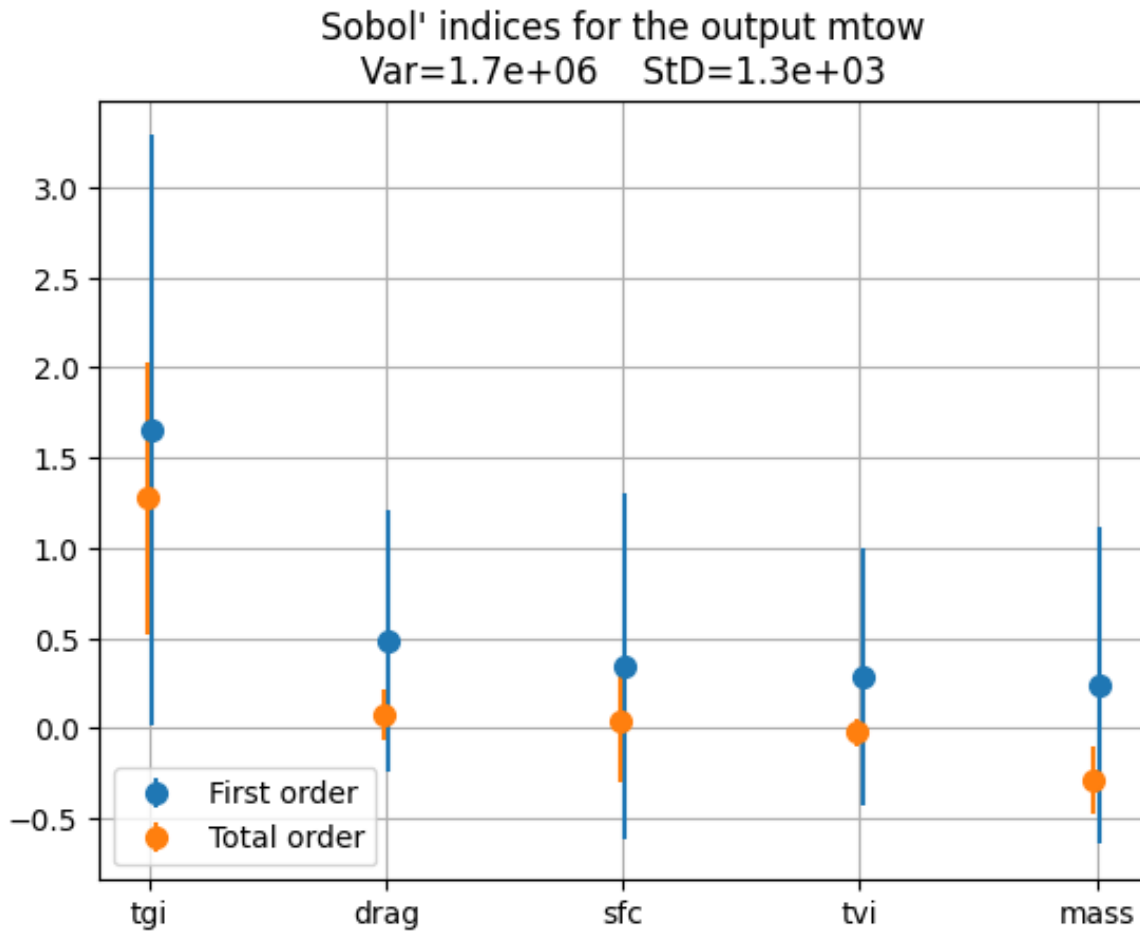


FIGURE 3 – Résultats de l'analyse de Sobol sur le modèle complet

Nous avons réalisé 100 évaluations car les évaluations pour le modèle complet sont trop coûteuses.

Nous remarquons, sur la figure 3, qu'il n'y a pas suffisamment de points pour estimer les variables. En effet, il y a 5 variables aléatoires pour 100 évaluations soit $\frac{100}{5+1}$ points. Avec ce faible nombre de points, les indices de Sobol sont mal estimés, les valeurs sont abérentes car elles ne sont pas comprises entre 0 et 1.

3.1.2 Analyse de Sobol du sous-modèle

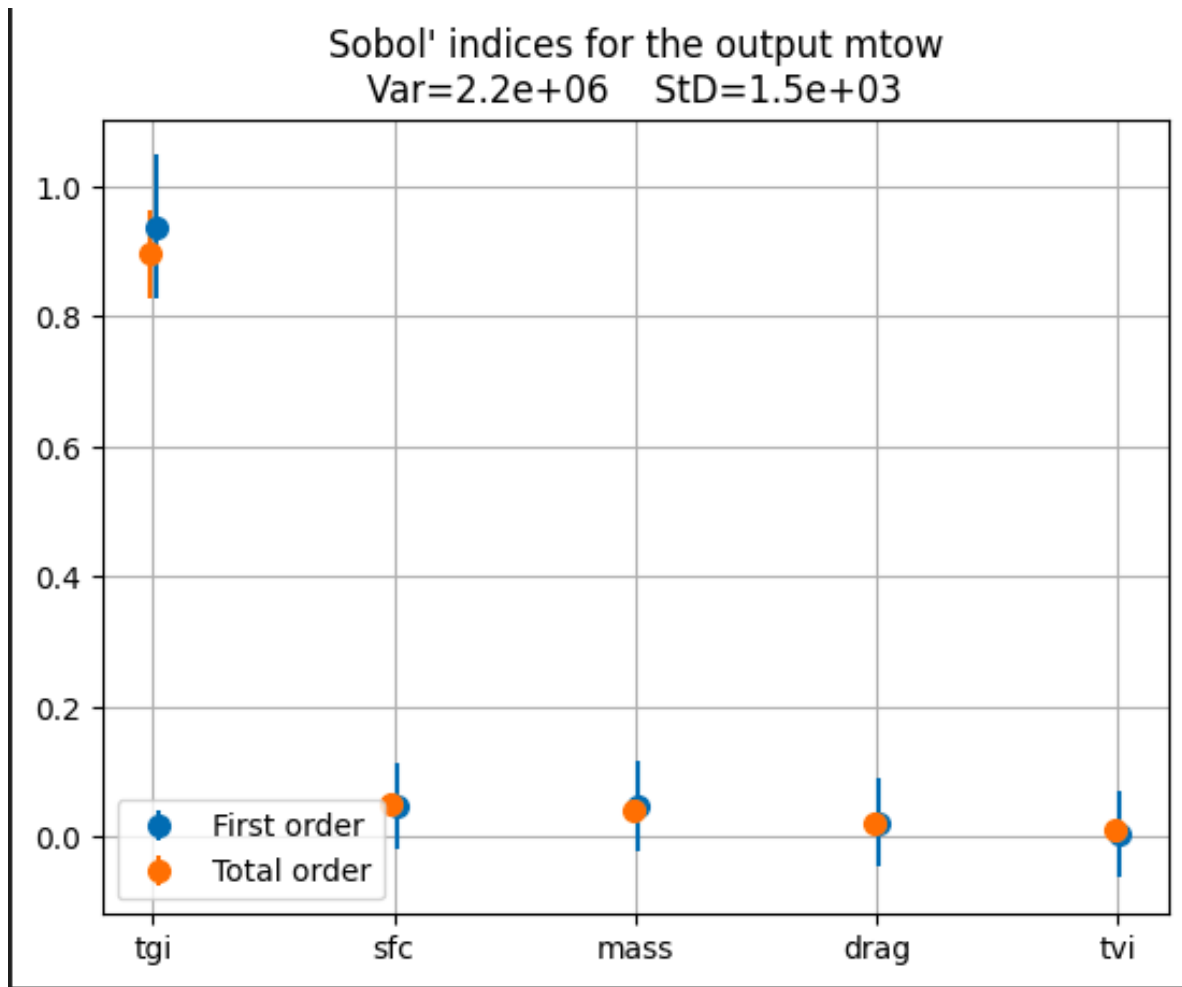


FIGURE 4 – Résultats de l'analyse de Sobol sur le modèle de substitut

Nous avons, cette fois-ci, réalisé 10 000 évaluations car le modèle de substitution est moins coûteux.

De plus, nous remarquons sur la figure 4 que la variable tgi a la plus grande influence

sur la sortie mtow, tant individuellement qu'en incluant les interactions (proches de 1 dans le graphe).

Les variables drag, sfc, tvi, et mass ont des influences moindres, avec des indices de Sobol proches de 0, pour les premiers ou les ordres totaux.

Nous pouvons donc conclure que la fluctuation de la variable technologique incertaine tgi aura un impact sur les variables de design ie la modélisation de l'avion.

3.2 Analyse de Morris

3.2.1 Analyse de Morris du modèle complet

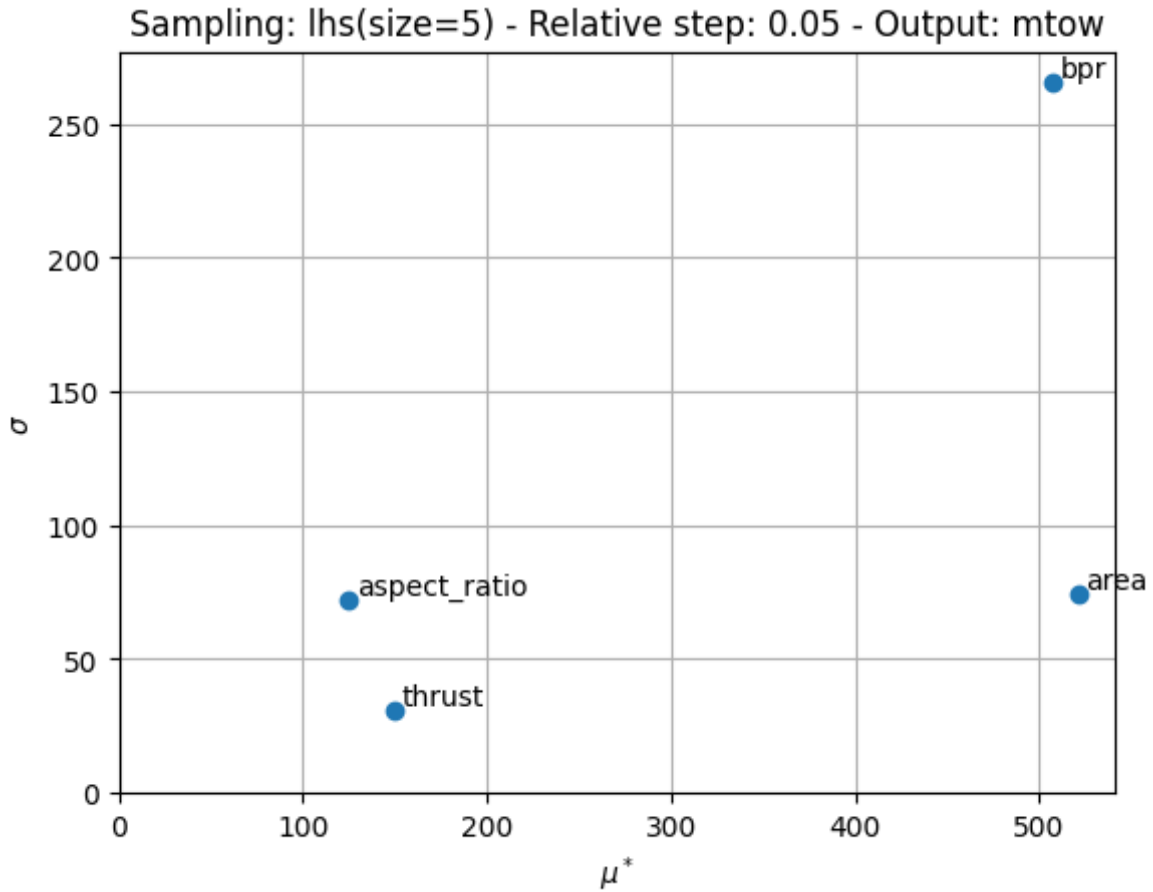


FIGURE 5 – Résultats de l'analyse de Morris sur le modèle complet

Afin d'évaluer la sensibilité des variables d'entrée du modèle complet et donc identifier les paramètres influents sur les sorties, nous avons effectué une analyse de Morris (sur le modèle complet). Cette méthode de criblage observe les variations des sorties

du modèle lorsque les entrées sont ajustées par petites étapes. Après 100 simulations (les simulations étaient très coûteuses), les résultats des indices de Sobol dans la figure 5 montrent des valeurs aberrantes, ce qui indique des défis liés à la haute fidélité des méthodes utilisées.

Il est donc important de choisir judicieusement le nombre de simulations pour assurer la précision des résultats.

3.2.2 Analyse de Morris du sous-modèle

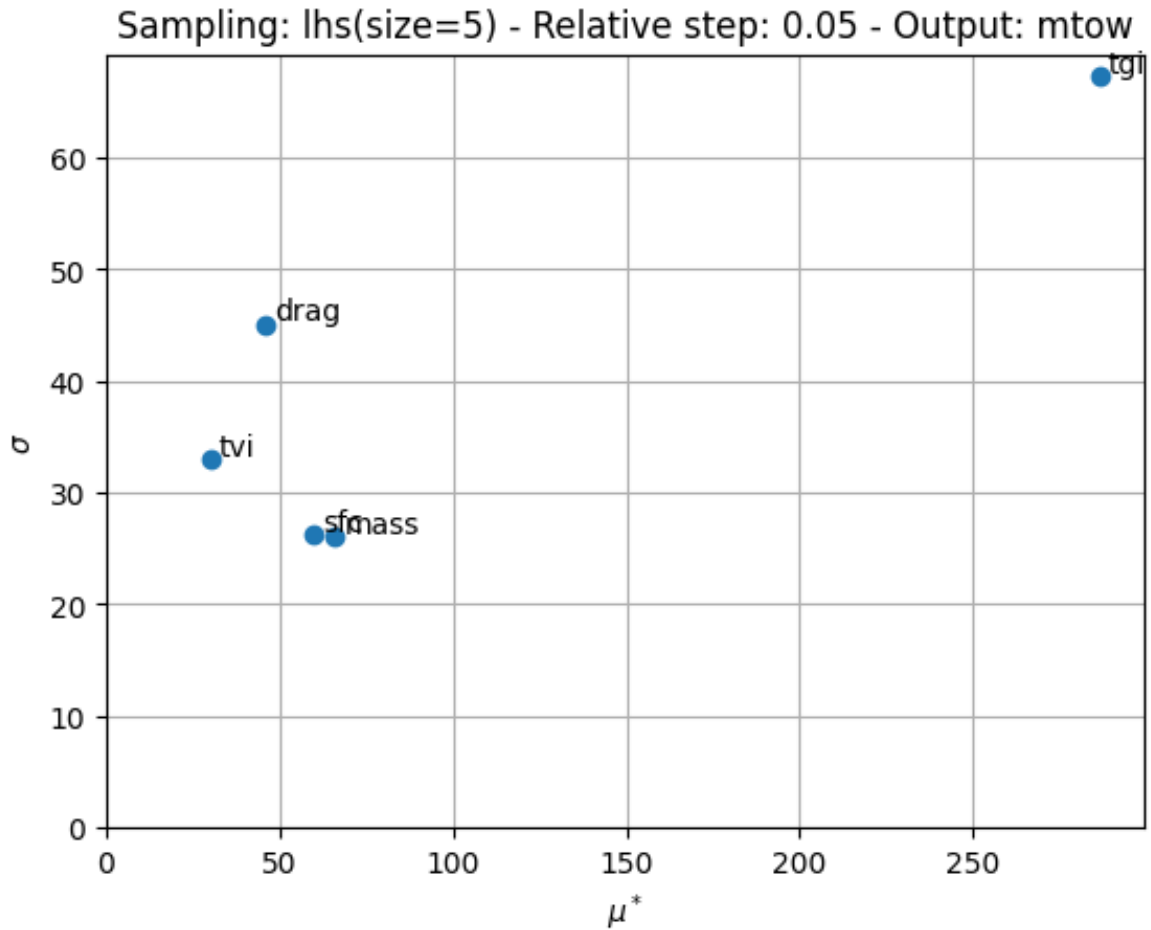


FIGURE 6 – Résultats de l’analyse de Morris sur le modèle de substitut

Encore une fois, pour identifier les facteurs les plus sensibles du modèle de substitution, nous avons fait une analyse de Morris sur ce dernier.

Les résultats de cette analyse, présentés dans la figure 6, mettent en évidence les variables cruciales pour la performance du modèle. En effet, ces informations sont essen-

tielles pour orienter les futurs ajustements du modèle de substitution afin d'améliorer sa précision en vue d'une conception plus globale de l'avion.

3.3 Conclusion sur l'analyse de Morris

Pour conclure, une analyse de Morris sur les modèles complet et de substitution nous a permis de montrer la sensibilité des différentes variables dans le développement de notre avion à hydrogène.

Pour le modèle complet, les simulations coûteuses ont limité la précision des résultats, ce qui souligne, encore une fois, l'importance d'optimiser le modèle afin d'équilibrer la précision des résultats et les coûts des calculs.

En revanche, l'analyse du sous-modèle a montré les variables clés, ce qui nous permet d'orienter les futurs ajustements.

Les résultats des différentes analyses de sensibilité nous permettent d'ajuster et optimiser notre sous-modèle.

4 Bonus

4.1 Interface Homme Machine

Dans cette section, nous avons développé une interface homme-machine en utilisant le package Python `streamlit`, qui permet de créer facilement des applications web interactives et de visualisation de données.

Cette interface permet à l'utilisateur de sélectionner les quatre paramètres de design suivants :

- **THRUST** : La poussée maximale du moteur, qui vérifie : $100, \text{kN} \leq \text{thrust} \leq 150, \text{kN}$ et a comme valeur par défaut : 125, kN.
- **BPR** : Le rapport de dilution du moteur, qui vérifie : $5 \leq \text{BPR} \leq 12$, et a comme valeur par défaut : 8.5.
- **AREA** : La surface alaire, qui vérifie : $120, \text{m}^2 \leq \text{AREA} \leq 200, \text{m}^2$, et a comme valeur par défaut : 160, m².
- **RATIO** : Le rapport d'aspect de l'aile, qui vérifie : $7 \leq \text{ar} \leq 12$, et a comme valeur par défaut : 9.5.

Ensuite, elle affiche un dessin de l'avion avec les paramètres sélectionnés par l'utilisateur, puis renvoie les résultats des autres paramètres de l'avion obtenus par : `aircraft-data = get-aircraft-data(discipline)`, pour que l'utilisateur puisse vérifier ses paramètres de design.

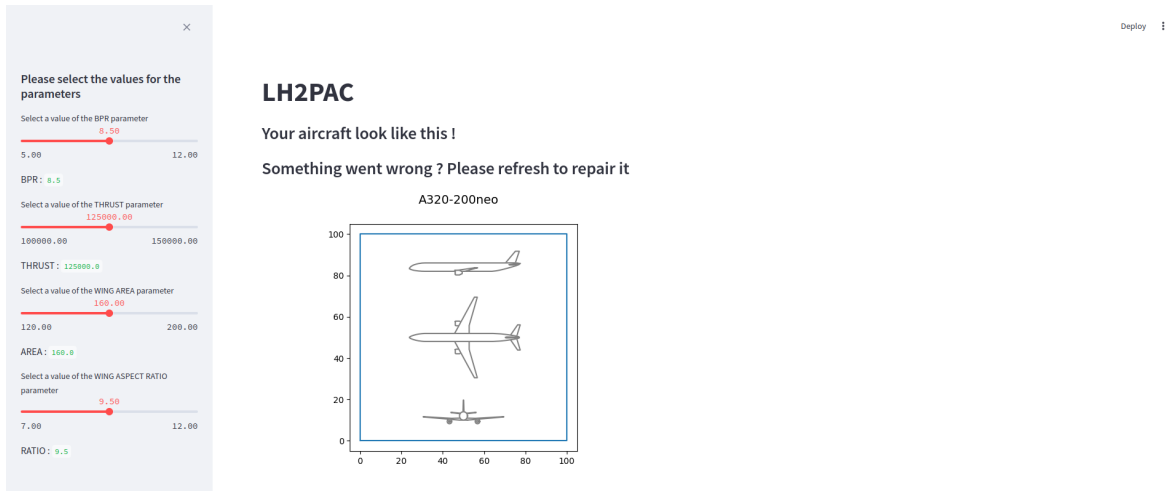


FIGURE 7 – Aperçu de l’interface homme-machine

La figure 7 montre un aperçu de ce qu’on obtient avec la commande :
`streamlit run streamlit.py`

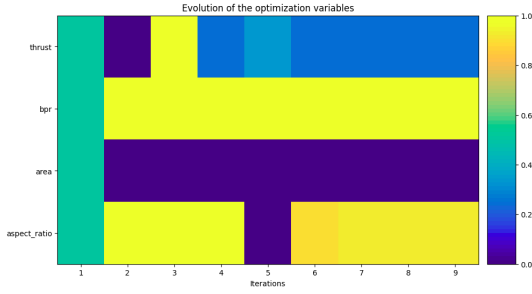
4.2 Robustesse du sous-modèle avec paramètres incertains

Nous avons remarqué dans les parties précédentes que notre sous modèle a de bons résultats mais est sensible aux variations des variables technologiques. Ainsi, si les scientifiques se sont trompés un tant soit peu sur ces dernières, que nous avons fixé aux estimations de 2030, notre modèle ne sera plus du tout précis.

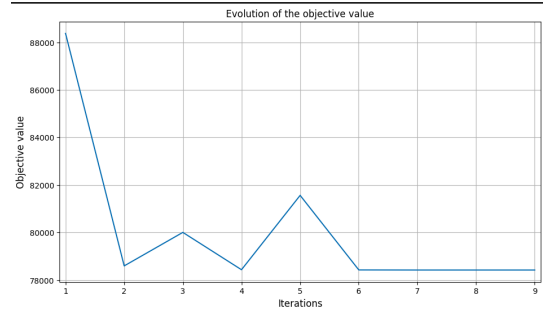
Dans le but de rendre notre modèle plus robuste, nous avons ajouté une variabilité dans les variables technologiques dans notre sous-modèle avant d’en optimiser les variables. Les variables de design ainsi optimisées prennent alors en compte la variabilité des variables technologiques et notre modèle est ainsi plus robuste au cas où les scientifiques auraient commis une erreur dans leurs prédictions.

Le problème revient alors à résoudre :

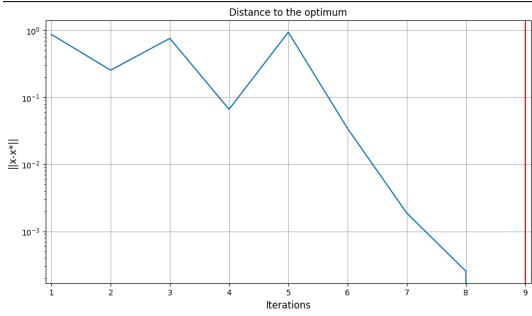
$$\begin{aligned} & \underset{x \in \mathcal{X}}{\text{minimun}} && \mathbb{E}[f(x, U)] + \kappa \times \mathbb{S}[f(x, U)] \\ & \text{sous contraintes} && \mathbb{P}[g(x, U) \geq 0] \leq \varepsilon \end{aligned}$$



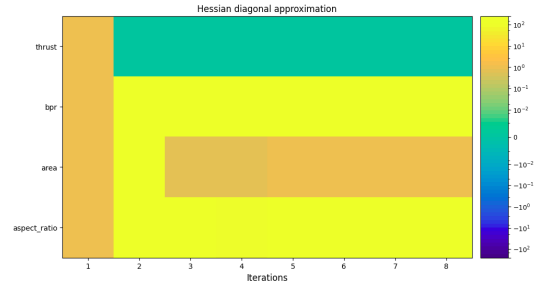
(a) Evolution de l'optimisation des variables



(b) Evolution de la valeur objectif



(c) Distance à l'optimum



(d) Approximation de la Hessienne des variables de design

FIGURE 8 – Résultats de l'optimisation sur le sous-modèle avec paramètres incertains

Le nombre d'échantillons joue un rôle important quant à la convergence de l'estimateur Monte-Carlo (convergeant en $\frac{1}{\sqrt{n_{sample}}}$).

Nous avons testé avec $n_{sample} = 10, 100$ et 1000 . Pour $n_{sample} = 10$, l'algorithme ne converge pas. Pour $n_{sample} = 100$, on converge à 12 itérations mais les résultats sont plutôt mauvais. On prend donc $n_{sample} = 1000$. Avec cette valeur, l'algorithme converge en 10 itérations comme le montre la figure 8.

En conclusion, cette optimisation nous a donc permis d'avoir les valeurs des variables de design x optimales même après fluctuations sur les variables technologiques. De plus, nous observons que, même si les valeurs des variables technologiques différentes, notre modèle continue de converger et nous atteignons à peu près la même valeur de MTOW que ce que nous avons précédemment. Notre modèle est donc robuste aux perturbations.

5 Conclusion

Ce projet explore la faisabilité d'un avion à hydrogène, inspiré par l'initiative ZEROe d'Airbus pour une aviation écologique et durable. Nous avons trouvé un modèle

physique cohérent qui minimise le poids maximal au décollage du LH2PAC tout en respectant les contraintes opérationnelles. En effet, nos résultats montrent une réduction significative du poids, essentielle pour améliorer l'efficacité énergétique et réduire les émissions de CO₂. De plus, notre modèle est robuste aux perturbations. Néanmoins, le paramètre MTOW pourrait encore être optimisé et, bien que les autres résultats semblent assez satisfaisants, il reste encore beaucoup de travail afin de rendre ce projet une réalité.