

Optimisation de plan de vol d'un drone faisant un cycle hamiltonien entre N points de mesures ; Problème du voyageur de commerce et durée de vie de batterie

Flight plan optimization of a UAV making a Hamiltonian cycle between N measurement points; Traveler problem and battery lifetime

Mohamed Abdessamia Chakchouk^{1,2}, Abdelkhalak El Hami³, Pierre Richard Dahoo⁴, Azzedine Lakhlifi⁵, Wajih Gafsi⁶, Mohamed Haddar⁷

¹ LMN, INSA de Rouen, Saint Etienne de Rouvray, 76800, France, mohamed.chakchouk@insa-rouen.fr

² Ecole Nationale des Ingénieurs de Sfax, Tunisie, mohamedabdelsami1992@gmail.com

³ LMN, INSA de Rouen, Saint Etienne de Rouvray, 76800, France, abdelkhalak.elhami@insa-rouen.fr

⁴ Université Paris-Saclay; UVSQ, CNRS, LATMOS, 78290 Guyancourt, France, pierre.dahoo@uvsq.fr

⁵ Université de Franche-Comté, UFC, CNRS, UTINAM, 25000 Besançon, France, azzedine.lakhlifi@obs-besancon.fr

⁶ LA2MP, Ecole Nationale des Ingénieurs de Sfax, Tunisie, wajih.gafsi@enis.tn

⁷ LA2MP, Ecole Nationale des Ingénieurs de Sfax, Tunisie, mohamed.haddar@enis.rnu.tn

* Corresponding authors: mohamed.chakchouk@insa-rouen.fr mohamedabdelsami1992@gmail.com

RÉSUMÉ. La planification de la trajectoire de couverture est un domaine d'étude fascinant pour les roboticiens, avec de nombreuses études disponibles dans la littérature de recherche. Pendant la phase de planification de la trajectoire. Comme la consommation d'énergie est fonction de la trajectoire qu'elle va prendre, le problème de consommation d'énergie sera partiellement converti en un problème d'optimisation de trajectoire. Pour la première phase, nous comprimerons le problème en un plan 2d, qui est connu sous le nom de "problème du voyageur de commerce".

Il n'existe pas de méthode connue de résolution du "problème du voyageur de commerce" qui permette d'obtenir des réponses exactes en un temps raisonnable pour les cas de grande taille (un grand nombre de villes). En raison de l'explosion combinatoire, nous devons souvent nous contenter de solutions approximatives pour ces énormes situations. Dans cet article on va montrer une heuristique et faire une simulation numérique du plan de vol puis on va caractériser l'effet de cette optimisation sur le temps de vol.

ABSTRACT. Cover path planning is a fascinating area of study for roboticists, with many studies available in the research literature. During the trajectory planning phase. Since energy consumption is a function of the trajectory it will take, the energy consumption problem will be partially converted to a trajectory optimization problem. For the first phase, we will compress the problem into a 2d design, which is known as the "traveling salesman problem".

There is no known method for solving the "traveling salesman problem" that provides accurate answers in a reasonable amount of time for large cases (a large number of cities). Due to the combinatorial explosion, we will often have to make do with approximate solutions for these huge situations. In this paper we will show a heuristic and make a numerical simulation of the flight plan and then characterize the effect of this optimization on the flight time.

MOTS-CLÉS. Problème du voyageur de commerce, monte Carlo, chaîne de Markov, batteries durées de vie, plan de vol, optimisation.

KEYWORDS. Commercial traveler problem, monte carlo, markov chain, batteries lifetime, flight plan, optimization.

Introduction

Le problème du voyageur de commerce est un problème de recherche opérationnelle qui consiste à déterminer l'ordre optimal des villes ou des machines dans lequel le total des kilomètres, des temps ou des coûts engagés est minimal.[1] Le problème tire son nom du problème de la détermination de l'itinéraire optimal pour un voyageur qui doit visiter n lieux et doit revenir à son point de départ à la

fin du voyage. [2]Un réseau peut être utilisé pour représenter la situation, les sommets représentant les villages, les arêtes les autoroutes et le poids sur les arêtes les distances.[3]

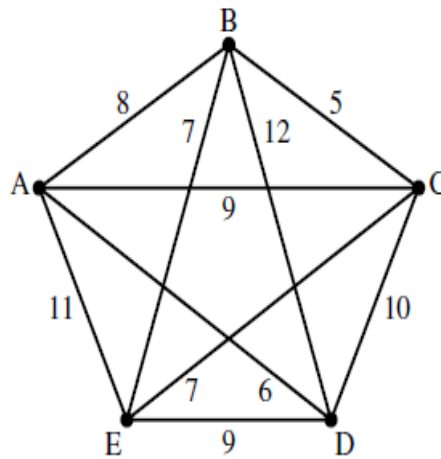


Figure 1.1. Cycle hamiltonien possible entre cinq point dans un plan 2D

Nous sommes dans le cas d'un TSP traditionnel si nous imposons la contrainte de ne visiter chaque sommet qu'une seule fois. Un cycle HAMILTONIEN est un cycle qui ne visite qu'une seule fois tous les sommets.[4]

Dans ce scénario, les villes ont été remplacées par des points de mesure.

Nous essayons de découvrir la plus courte distance entre 25 points de mesure.

Ce problème est plus difficile qu'il n'y paraît pour un ensemble de n points ; il y a un total de $N!$ Chemins possibles Comme le point de départ n'a aucun effet sur la longueur du chemin, nous pouvons le choisir au hasard, ce qui donne $(N-1)!$ trajets différents[5]. Enfin, comme chaque chemin peut être emprunté dans deux directions et que toutes les options ont la même longueur, nous pouvons diviser ce nombre par deux.

Pour identifier la meilleure approche dans notre application, par exemple, nous devons effectuer le calcul $[(25-1)! / 2]$, ce qui donne un total de $3.102242e+23$ façons alternatives.

Pour résoudre le problème du représentant de commerce, nous disposons de deux grandes catégories de méthodes [6]: les méthodes exactes, qui conduisent à un temps de résolution exponentiel qui explose avec de grands échantillons, et les méthodes approximatives, ou heuristiques, qui ne considèrent qu'une partie plus intéressante du problème pour approcher, et parfois trouver, la solution optimale en un temps de calcul raisonnable.[7]

1. Utilisation d'une heuristique pour résoudre le problème du voyageur de commerce

La résolution du TSP prend idéalement trop de temps ; on utilise plutôt des procédures d'approximation, ou heuristiques. La distinction est que les méthodes d'approximation garantissent à quel point les réponses peuvent être mauvaises. Typiquement spécifié comme c fois la meilleure valeur. Il existe plusieurs approches des méthodes heuristiques. [8]

On peut formuler le problème de voyageur de commerce de la façon suivante ;

$$x_i^j = \begin{cases} 1 & \text{de point } i \text{ vers } j \\ 0 & \text{dans tous autres cas} \end{cases}$$

Ce paramètre va garantir qu'on aboutit à un cycle unique qui passe par tous les points et non pas deux cycles détachés.

Pour $i = 1 \dots, n$ on a c_{ij} distance entre points donc on essaye de minimiser

$$\sum_{i=1}^N \sum_{j=1, j \neq i}^n c_{ij} x_{ij}^j \quad [1]$$

$$x_{ij} \in [0..1]$$

$$\sum_{i=1, j \neq i}^N x_{ij} = 1 \quad [2]$$

$$\sum_{j=1, j \neq i}^N x_{ij} = 1 \quad [3]$$

2. L'approche MCMC pour résoudre le problème du voyageur de commerce

La technique que nous allons employer est connue sous le nom d'approche de Monte Carlo par chaîne de Markov (MCMC). Dans ce cas, nous utiliserons la méthode de Metropolis-Hastings. L'algorithme de Metropolis-Hastings est une méthode de Monte Carlo par chaîne de Markov (MCMC) permettant d'extraire une séquence d'échantillons aléatoires d'une distribution de probabilité lorsque l'échantillonnage direct n'est pas pratique [8].

Pour commencer, il faut définir la fonction à minimiser. Il s'agit de la distance parcourue par le voyageur. L'ensemble des chemins que le voyageur peut emprunter est noté S_N . Un itinéraire $\sigma \in S_N$ est en fait un vecteur portant la liste triée des villes visitées par le touriste. Par exemple, dans l'itinéraire précédent, le voyageur visite le point 1, puis point 2, ..., et ainsi de suite jusqu'à revenir à la ville 1.

$$\sigma = (\sigma(1), \sigma(2), \sigma(3), \dots, \sigma(N))$$

Où i désigne la i -ième ville traversée par le voyageur.

En conséquence, la distance d'un chemin est donnée par la relation :

$$D_\sigma = \sum_{i=1}^N d((\sigma(i), \sigma(i+1))) + d(\sigma(N), \sigma(1)) \quad [4]$$

La fonction à minimiser est la distance parcourue :

$$F : \sigma \in S_N \longrightarrow D_\sigma \quad [5]$$

La technique de Metropolis-Hastings crée une séquence de valeurs d'échantillon de telle sorte que la distribution des valeurs se rapproche de la distribution cible P au fur et à mesure que des valeurs d'échantillon sont créées (x). Ces valeurs d'échantillon sont créées de manière répétée, la distribution de l'échantillon suivant ne dépendant que de la valeur d'échantillon actuelle (faisant de la séquence d'échantillons une chaîne de Markov). À chaque itération, l'algorithme choisit un candidat pour la valeur d'échantillon suivante en fonction de la valeur d'échantillon actuelle.

L'organigramme de l'algorithme utilisé est celui-ci ;

Étape 0 : une route aléatoire σ_0 (Initialisation), fixer une limite: Itermax.

Étape 1 : Faire passer σ_1 par une transition un point $Q(\sigma_0, \sigma_1)$.

Étape 2 : Déterminer le rapport de Metropolis-Hastings $r(\sigma_0, \sigma_1)$,

Étape 3 : le $r(\sigma_0, \sigma_1)$, est compris dans l'intervalle $[0, 1]$ on a $\sigma_0 = \sigma_1$, sinon on garde le chemin choisi . Itérer l'algorithme jusqu'à ce qu'il atteigne Itermax.

la solution est soit acceptée (auquel cas la valeur candidate est utilisée dans l'itération suivante) soit rejetée (auquel cas la valeur candidate est écartée et la valeur courante est réutilisée dans l'itération suivante) - la probabilité d'acceptation est déterminée en comparant les valeurs de la fonction $f(x)$ des valeurs d'échantillon courantes et candidates par rapport à la distribution souhaitée $P(x)$.

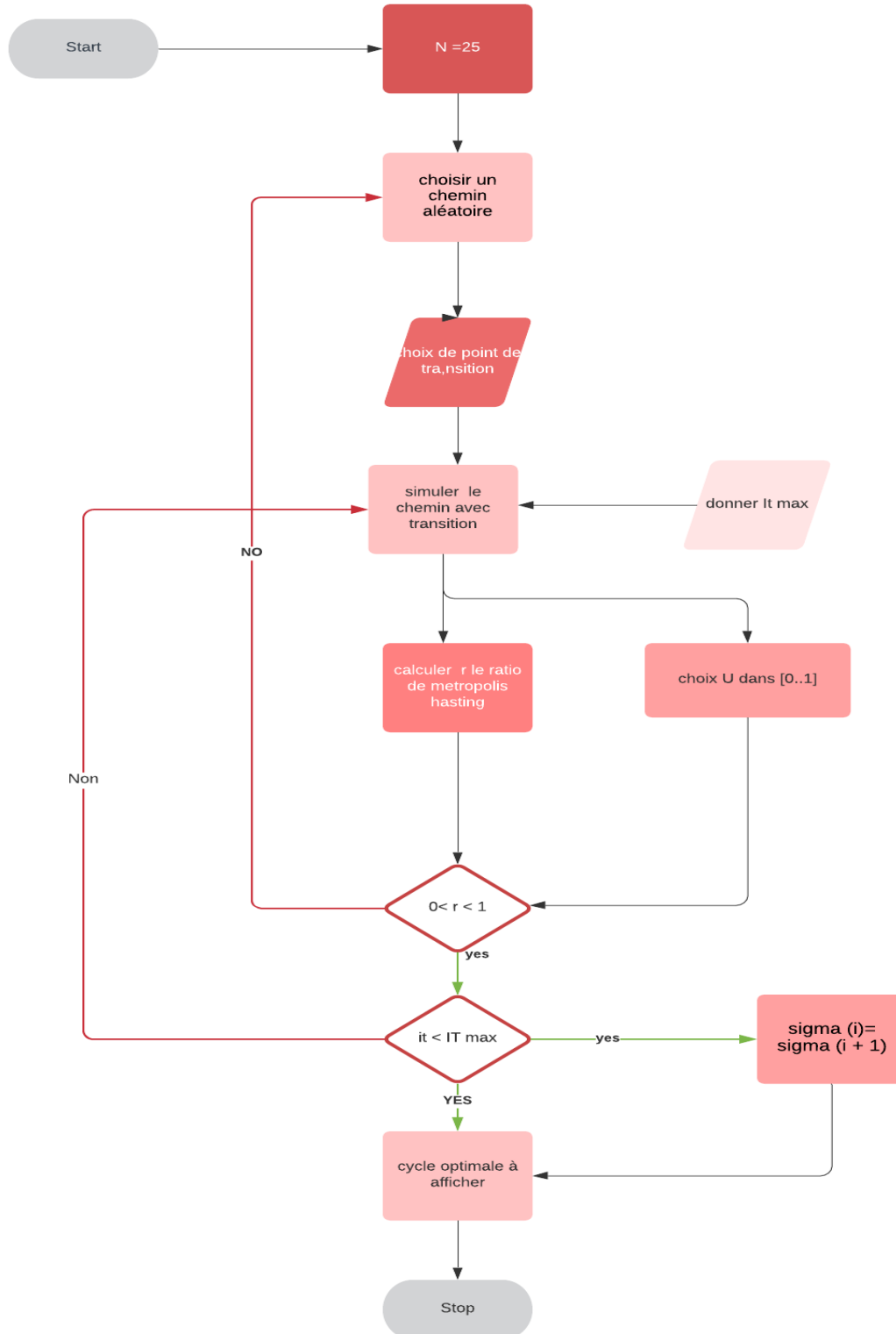


Figure 2.1. Organigramme de l'algorithme

3. Résultat et conséquence

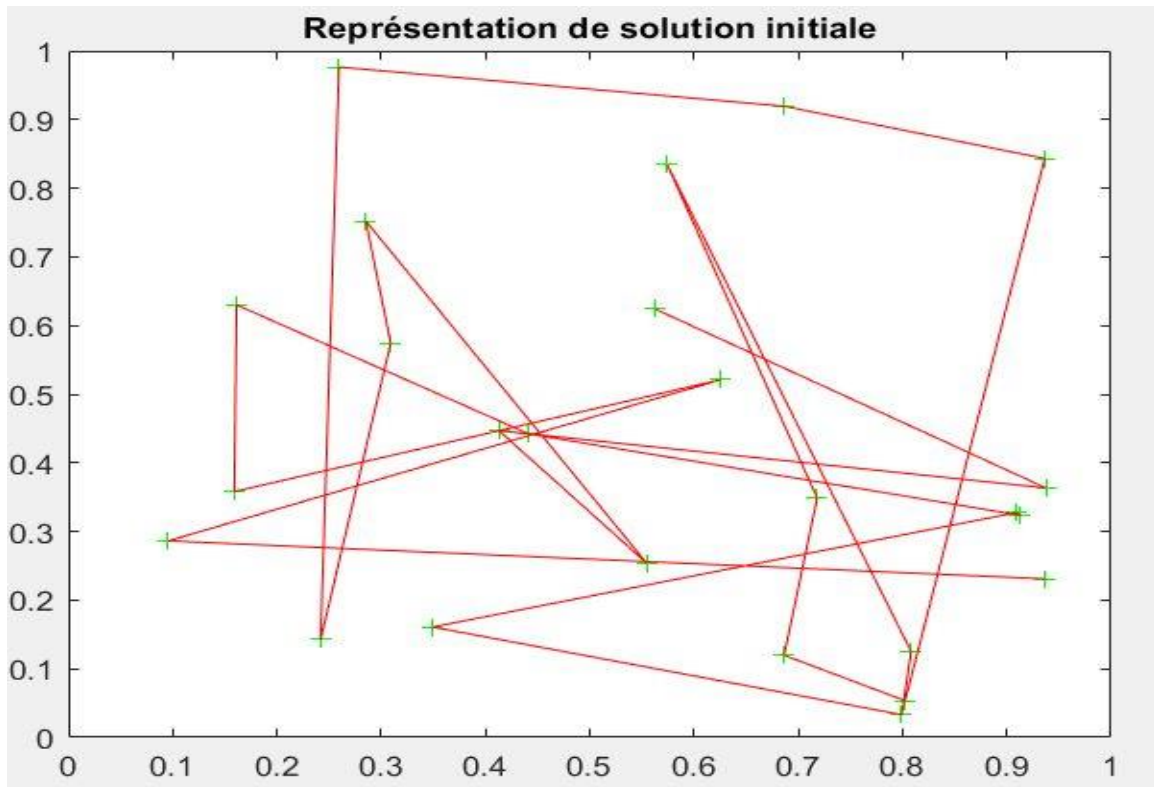


Figure 3.1. Distance initiale de 9.707

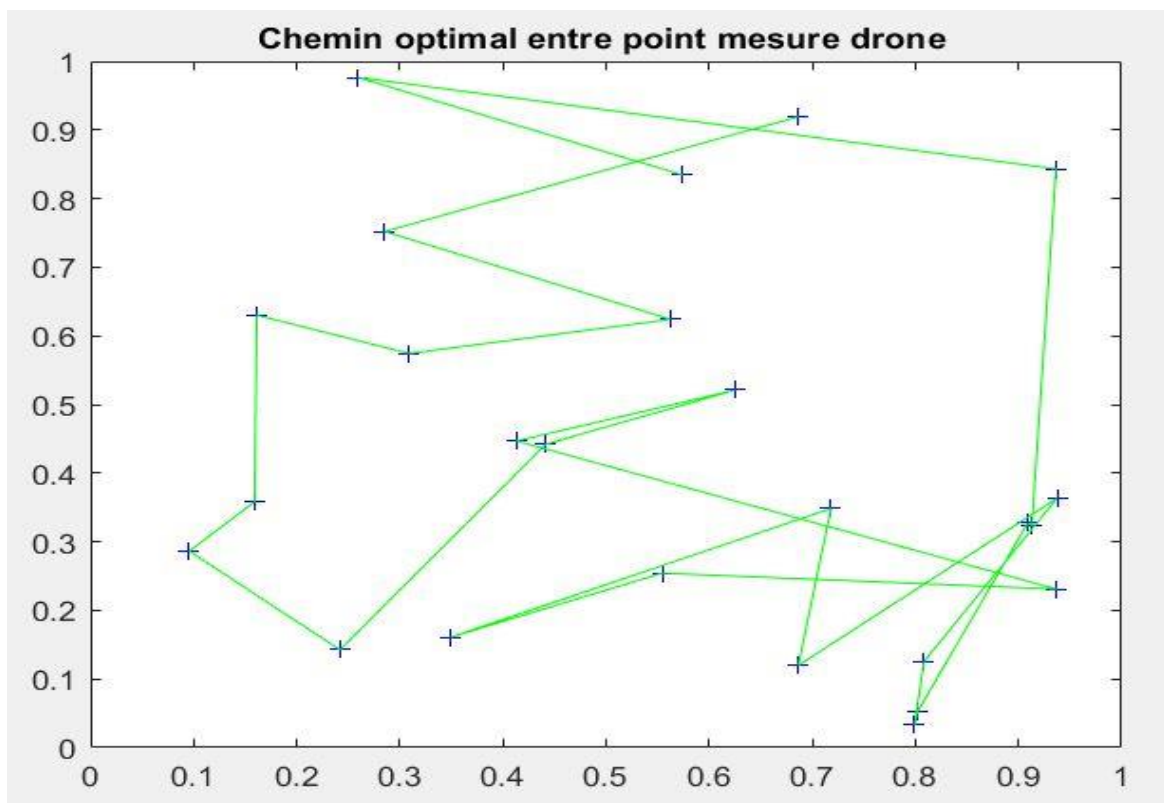


Figure 3.2. Distance pour un cycle optimisé de 7.064

4. La distance et la consommation énergétique ;

Les batteries lithium-polymère (LiPo) sont des batteries rechargeables qui reposent sur la technologie lithium-ion avec un électrolyte polymère semi-solide (gel) plutôt qu'un électrolyte

liquide elles sont de plus en plus utilisés pour les drones en fait l'estimation de nombre de batteries et le choix du meilleurs type à utiliser dépend du plan de vol, du drone et des conditions climatiques dans lesquelles le drone va opérer, allonger la durée de vie des batterie est le but ultime qui détermine le choix, ici on veut vérifier ce que la méthode d'optimisation du plan de vol a apporter pour les batteries et leurs autonomie .[9]

$$V_{\text{bat nominal}} = L_{\text{iPoCellVolt}} \cdot N_{\text{series}} \quad [6]$$

Donc on va considérer le nombre total de batteries en général on va travailler avec des quadcopter ou en met deux batteries en séries donc N sera égal à deux dans la majorité des cas.

Une formule simple permet de déterminer le temps de vol prévu :

$$T_{\text{flight}} = [Q/I] * 0.06 \quad [7]$$

T_{flight} Représente le temps de vol de votre drone en minutes ;

Q représente la capacité de la batterie LiPo en mAh ;[10]

Si on veut trouver le quotient $\frac{T'_{\text{flight}}}{T^0_{\text{flight}}}$ ceci revient à calculer $\frac{L^0_{\text{flying}}}{L'_{\text{flying}}}$

$$T'_{\text{flying}} = \frac{L^0_{\text{flying}}}{L'_{\text{flying}}} * T^0_{\text{flying}} = \frac{9.707}{7.064} * T^0_{\text{flying}} = 1.37 T^0_{\text{flying}} \quad [8]$$

On peut vérifier ce que ce gain apporte pour notre exemple de drone, en fait pour chaque vingt heures, le temps nominal pour charge et décharge de batterie de drone, on peut gagner environ sept heures, cela veut dire que sur quatre cycle de charge et décharge on a pu économiser un, donc pour la durée de vie d'une batterie Lippo, qui est relativement entre 150 et 250 cycle[11], peut être étendu d'une façon significative.

Conclusion

La détermination d'un cycle optimale en utilisant une heuristique rapide a permis un gain au niveau de temps de vol, ce qui était attendu, en fait la consommation majeur de l'énergie d'une batterie provient de vol verticale ou déplacement entre deux point, le problème traité dans cet article était pour robot dans un plan 2D, les travaux future vont prendre en considération le temps comme étant une contrainte et traité donc le TSPTW (travel sales man problem with time windows).

Références

- [1]G. Wang, J. Wang, M. Li, H. Li, and Y. Yuan, "Robot path planning based on the travelling salesman problem," *Chemical Engineering Transactions*, vol. 46, pp. 307–312, Dec. 2015, doi: 10.3303/CET1546052.
- [2]A. Chassein and M. Goerigk, "On the recoverable robust traveling salesman problem," *Optimization Letters*, vol. 10, no. 7, pp. 1479–1492, Oct. 2016, doi: 10.1007/s11590-015-0949-5.
- [3]M. Zeriab Es-Sadek, "Contribution à l'optimisation globale : approche déterministe et stochastique et application." [Online]. Available: <https://tel.archives-ouvertes.fr/tel-00560887>
- [4]F. Greco, "Travelling Salesman Problem."
- [5]H. Kona and A. Burde, "A Review of Traveling Salesman Problem with Time Window Constraint," *IJIRST-International Journal for Innovative Research in Science & Technology*, vol. 2, 2015, [Online]. Available: www.ijirst.org
- [6]el Hami A and Radi B, *Incertitude et optimisation et fiabilité des structures*. Paris: Hermes-Lavoisier, 2013.

- [7] C. Nilsson, “Heuristics for the Traveling Salesman Problem.”
- [8] “Travelling salesman problem: Mathematical Heuristic and problem solving.”
- [9] G. Avanzini, E. L. de Angelis, and F. Giulietti, “Optimal performance and sizing of a battery-powered aircraft,” *Aerospace Science and Technology*, vol. 59, pp. 132–144, Dec. 2016, doi: 10.1016/j.ast.2016.10.015.
- [10] Y. Chen, D. Baek, A. Bocca, A. Macii, E. Macii, and M. Poncino, “A Case for a Battery-Aware Model of Drone Energy Consumption.”
- [11] S. Lee and J. Kim, “Power capability analysis of lithium battery and supercapacitor by pulse duration,” *Electronics (Switzerland)*, vol. 8, no. 12, Dec. 2019, doi: 10.3390/electronics8121395.