

JSBSim як інструмент моделювання динаміки польотів із використанням методів машинного навчання

Олексій Винокур¹ та Ірина Перова¹

¹ Харківський університет радіоелектроніки, Харків, Україна

Анотація

JSBSim є сучасною платформою з відкритим кодом для моделювання динаміки польотів літальних апаратів із шістьма ступенями свободи (6DOF). Робота присвячена оцінці можливостей JSBSim як симулятора для розробки та тестування систем автоматичного приземлення літальних апаратів із застосуванням методів машинного навчання. У дослідженні використано алгоритми підкріплювального навчання (Reinforcement Learning), зокрема DQN і PPO, а також рекурентні нейронні мережі LSTM для обробки часових рядів. Результати дослідження демонструють ефективність обраного підходу, підкреслюючи значний потенціал JSBSim у дослідженнях автономних систем керування польотами.

Ключові слова

машинне навчання, літальні апарати, симуляція польоту, динаміка польоту, нейронні мережі, LSTM, JSBSim, 6DOF

1. Вступ

Розвиток автономних літальних апаратів є одним із ключових напрямів сучасної авіації та космічної галузі. Особливе значення має автоматизація критичних етапів, таких як приземлення, що вимагає високої точності та стабільності. Моделювання цих процесів у реальних умовах часто є дорогим та небезпечним. Тому симуляційні середовища, такі як JSBSim, стають важливим інструментом для тестування нових алгоритмів та систем керування.

JSBSim — це платформа для моделювання динаміки польотів із відкритим кодом, що дозволяє відтворювати аеродинамічні характеристики літальних апаратів із шістьма ступенями свободи (6DOF) [1]. Її гнучкість та підтримка зовнішніх бібліотек дають змогу інтегрувати сучасні алгоритми машинного навчання, зокрема підкріплювального навчання та рекурентні нейронні мережі, для створення автономних систем керування польотами [3; 6]

Мета цієї роботи — дослідити можливості JSBSim у контексті розробки алгоритмів для автоматичного приземлення з використанням методів підкріплювального навчання.

2. Методологія дослідження

Для моделювання динаміки польоту літальних апаратів використовувався JSBSim. Симуляція враховувала вплив зовнішніх сил (гравітації, турбулентності) та роботу двигунів. Аеродинамічні параметри задавалися через конфігураційні файли JSBSim, що дозволяло налаштовувати симуляцію для різних моделей літальних апаратів [1].

Information Systems and Technologies (IST-2024), November 26-28, 2024, Kharkiv, Ukraine

✉ oleksii.vynokur@nure.ua (O. Vynokur); rikywenok@gmail.com (I. Perova);

🆔 0009-0001-4328-3886 (O. Vynokur); 0000-0003-2089-5609 (I. Perova);



© 2024 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

JSBSim інтегрувався з бібліотекою OpenAI Gym для створення тренувального середовища [4]. Використовувалися такі алгоритми:

- Deep Q-Network (DQN) для роботи з дискретними діями (зміна тяги або кута нахилу) [3].
- Proximal Policy Optimization (PPO) для моделювання безперервних дій, що дозволяє точніше керувати маневрами під час посадки [5].
- Для обробки часових рядів (вхідних даних, таких як висота, швидкість, кут нахилу, тяга двигунів та вплив вітру) застосовувалися рекурентні нейронні мережі (LSTM). Мережа моделює динамічні процеси з довготривалими залежностями, що є критичними для точного управління літальними апаратами.

Нейронна мережа складалася з двох LSTM-шарів з 128 нейронами кожен, за якими слідували повнозв'язні шари [7]. Мережа передбачала оптимальні параметри для приземлення літального апарата на основі поточних і попередніх станів системи.

3. Результати дослідження

Результати симуляції показали, що використання алгоритмів підкріплювального навчання дозволяє досягти стабільного і точного приземлення літального апарата в умовах симуляції.

- Алгоритм PPO показав ефективність у виконанні маневрів, таких як зміна тяги. Зокрема, модель досягла високих значень швидкості (близько 180 вузлів) при максимальній тязі, що дозволило стабільно підтримувати швидкість на рівні, необхідному для посадки [5].
- LSTM ефективно враховувала залежності між послідовними станами літального апарата, що допомогло підтримувати стабільний процес приземлення, незважаючи на коливання висоти та зміну зовнішніх параметрів, таких як турбулентність [7].

Layer (type)	Output Shape	Param #
lstm_46 (LSTM)	(None, 10, 128)	68,608
lstm_47 (LSTM)	(None, 128)	131,584
dense_23 (Dense)	(None, 5)	645

Рисунок: 1 Структура нейронної мережі LSTM

Низка тестів, у яких використовувалася максимальна тяга (throttle = 1.0), показала, що літальний апарат підтримував стабільний режим на висоті близько 1000 футів з помірним зниженням висоти. Швидкість літального апарата (vc-kts) постійно зростала від початкових значень до 180 вузлів, що є критично важливим для ефективного контролю.

Платформа JSBSim забезпечила реалістичне моделювання польоту з урахуванням складних фізичних процесів, таких як взаємодія з атмосферними умовами та робота системи керування. Тренування моделей вимагало значних обчислювальних ресурсів, особливо для LSTM-мереж, але результати тренування PPO та LSTM підтвердили доцільність цього підходу для автономних систем керування польотами [1; 6] [8].

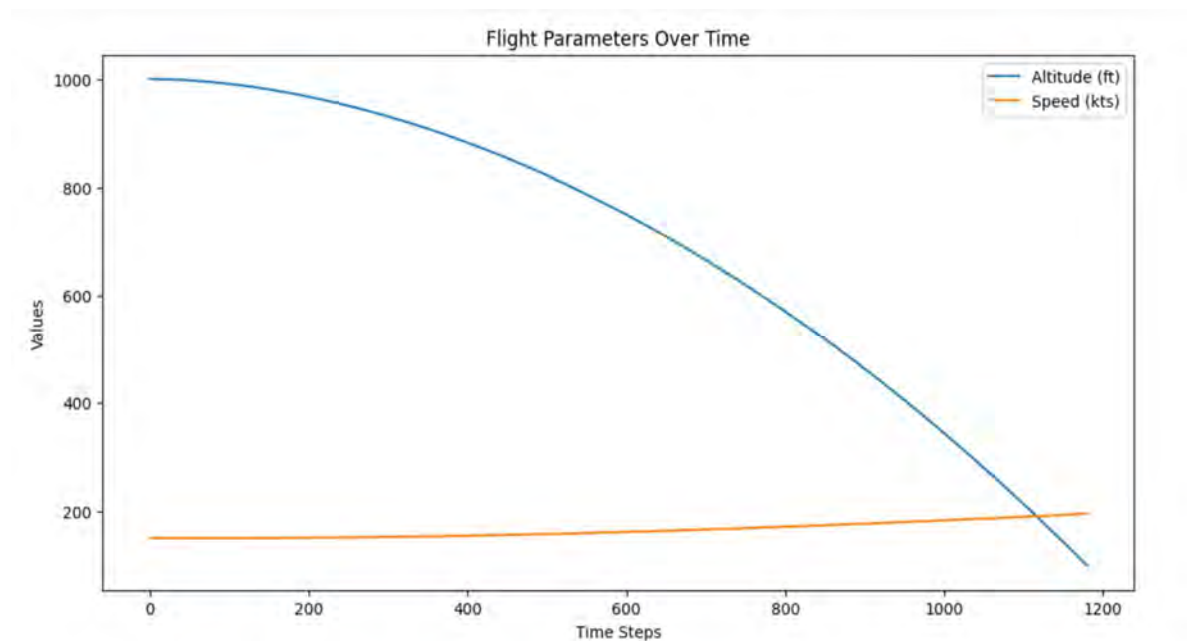


Рисунок: 2 Зміна параметрів висоти та каліброваної швидкості протягом приземлення

4. Обговорення

Результати дослідження підкреслили важливість інтеграції методів машинного навчання з точними симуляційними платформами, такими як JSBSim. Використання LSTM дозволило врахувати часові залежності в поведінці літального апарата, що є критичним для точного управління в реальних умовах [7].

Основні виклики включають:

- Потрібно більше даних для тренування нейронних мереж, що вимагає значних обчислювальних ресурсів.
- Складність налаштування параметрів симуляції для конкретних задач (наприклад, зміни умов вітру або турбулентності).

У подальших дослідженнях планується використання інших функцій активації в нейронних мережах, а також модифікація кількості шарів і кількості нейронів у мережі для підвищення ефективності моделювання [5; 9]. Це дозволить покращити точність та стабільність автономного управління в більш складних умовах.

5. Висновки

JSBSim є потужним інструментом для дослідження автономних систем польотів. Інтеграція з методами машинного навчання дозволяє створювати високоточні системи автоматизації, зокрема для приземлення літальних апаратів. Подальші дослідження мають бути спрямовані на розробку більш ефективних алгоритмів та покращення обчислювальної ефективності, що дозволить реалізувати ці підходи в реальних умовах.

Успішна інтеграція алгоритмів підкріплювального навчання (PPO) та рекурентних нейронних мереж (LSTM) в єдину програму дозволила створити автономну систему керування польотом, що забезпечує стабільне та точне приземлення, що є основним досягненням цього дослідження.

Список використаних джерел та посилань

- [1] JSBSim Documentation. JSBSim Flight Dynamics Model: The Open Source Flight Dynamics Model for Use in Aerospace Engineering, Research, and Education. [Електронний ресурс]. – Доступно за посиланням: <http://jsbsim.sourceforge.net>.
- [2] Sutton, R. S., & Barto, A. G. (2018). Reinforcement Learning: An Introduction. – 2nd Edition, MIT Press. – 552 с.
- [3] Lillicrap, T. P., Hunt, J. J., Pritzel, A., et al. (2016). Continuous control with deep reinforcement learning. – arXiv preprint, arXiv:1509.02971.
- [4] OpenAI. (2018). OpenAI Gym: Toolkit for Developing and Comparing Reinforcement Learning Algorithms. [Електронний ресурс]. – Доступно за посиланням: <https://gym.openai.com>.
- [5] Kochenderfer, M. J., & Wheeler, T. A. (2019). Algorithms for Optimization. – MIT Press.
- [6] FlightGear. (2024). Flight Simulation with JSBSim. [Електронний ресурс]. – Доступно за посиланням: <https://www.flightgear.org>.
- [7] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. – MIT Press.
- [8] Tieleman, T., & Hinton, G. (2012). Lecture 6.5—RMSProp: Divide the gradient by a running average of its recent magnitude. – COURSE: Neural Networks for Machine Learning.
- [9] Копитин, О. В., Кравченко, І. В. (2022). Розробка системи керування БПЛА на основі методів машинного навчання. – Наукові праці Національного авіаційного університету, 2(61), 123-129.
- [10] Nguyen, Q., Nguyen, T., & Hwang, D. (2020). Autonomous landing of UAVs using reinforcement learning. – IEEE Transactions on Aerospace and Electronic Systems, 56(4), 2902-2913.
- [11] Munos, R., et al. (2016). Safe and efficient off-policy reinforcement learning. – arXiv preprint, arXiv:1606.02647.