

Teleoperación asistida de cuadricópteros mediante aprendizaje por refuerzo profundo

Ana Laura Diedrichs*, Gerardo Puga†, Olmer Garcia-Bedoya† and Michel Hidalgo†

*Universidad Tecnológica Nacional Facultad Regional Mendoza,

Rodriguez 273, Ciudad de Mendoza, Argentina

Email: ana.diedrichs@docentes.frm.utn.edu.ar

†Ekumen, Honduras 5550, oficina 302

Ciudad Autónoma de Buenos Aires, Argentina

Email: glpuga, olmerg, michel@ekumenlabs.com

Resumen—Este trabajo detalla los avances alcanzados en la experimentación para desarrollar un prototipo de sistema de teleoperación asistida en tiempo real para cuadricópteros, utilizando técnicas de aprendizaje por refuerzo profundo. Estas técnicas posibilitan que el cuadricóptero aprenda de su experiencia previa y mejore progresivamente su rendimiento, lo que resulta en una reducción de la carga cognitiva para el operador humano en un modelo de teleoperación asistida. El sistema fue concebido para la empresa Ekumen, con el propósito de mejorar la seguridad y la eficiencia de las operaciones de cuadricópteros en entornos considerados peligrosos o inaccesibles para los seres humanos. Hasta la fecha, se ha establecido un entorno de simulación y experimentación reproducible para entrenar a los cuadricópteros utilizando diversas configuraciones de aprendizaje por refuerzo profundo.

I. INTRODUCCIÓN

En los últimos años, los vehículos aéreos no tripulados (VANT o UAV), en especial los cuadricópteros, han demostrado ser herramientas altamente efectivas en la realización de inspecciones para el mantenimiento de estructuras edilicias en diversas industrias. Sin embargo, cuando se trata de inspecciones en espacios interiores, encontramos un desafío singular: el espacio disponible para maniobrar a menudo es extremadamente reducido. Este tipo de entornos plantea un desafío significativo para el control de vuelo, ya que la proximidad a las paredes altera la dinámica del flujo de aire, lo que puede comprometer la maniobrabilidad del vehículo.

Otro reto para quienes operan un cuadricóptero es la presencia de obstáculos. Los objetos sólidos que sobresalen de la estructura edilicia aumentan el riesgo de colisiones, especialmente en situaciones de visibilidad limitada, como cuando se utiliza una única cámara monocular.

Para afrontar estos desafíos se propone una arquitectura de control compartido, basada en aprendizaje profundo por refuerzo, como muestra la figura 1. En esta representación, se observa cómo el modelo de aprendizaje por refuerzo utiliza como entradas las órdenes emitidas por el operador humano que controla el UAV (a_h) y el estado actual del UAV, proporcionado por sus sensores (s). La construcción del modelo no se limita únicamente a estas entradas, sino que también incorpora la retroalimentación proporcionada por el operador, lo que resulta en la formulación de una política (π). Esta política

actúa como la función de control encargada de determinar la próxima acción que el UAV ejecutará.

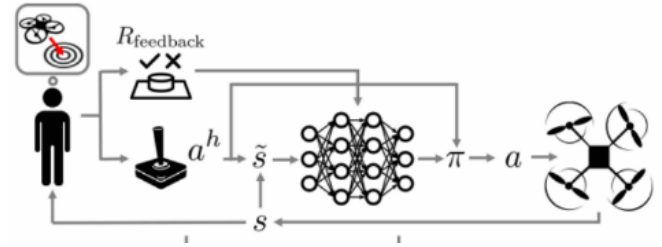


Figura 1. Diagrama en bloques de un sistema de autonomía compartida basado en aprendizaje por refuerzo profundo.²

Construir y operar un cuadricóptero real puede ser costoso y potencialmente peligroso, especialmente durante la fase de aprendizaje y experimentación de modelos de aprendizaje por refuerzo. La simulación permite realizar pruebas en un entorno virtual sin riesgo de dañar equipos o causar lesiones. Además es posible probar múltiples enfoques y ajustar parámetros de manera eficiente en un entorno virtual simulado sin tener que esperar por el tiempo real de vuelo o las condiciones climáticas adecuadas. Mediante la simulación, es posible crear escenarios de vuelo controlados y reproducibles, lo que facilita la comparación entre diferentes enfoques de aprendizaje y la evaluación del rendimiento del modelo en una amplia gama de situaciones. El entorno simulado puede utilizarse para generar grandes cantidades de datos de entrenamiento etiquetados a un costo mínimo, que serán necesarios para entrenar modelos de aprendizaje por refuerzo profundo antes de ser transferidos al mundo real, lo que puede ayudar a reducir la cantidad de tiempo necesario para entrenar un modelo en un entorno físico.

Este trabajo es un proyecto interno de investigación y desarrollo para la empresa Ekumen Labs en el marco del programa de vinculación con empresas de la Carrera de Especialización en Inteligencia Artificial de la Facultad de Ingeniería de la Universidad de Buenos Aires. A continuación se presenta un resumen del estado del arte y desarrollo del entorno de

²Imagen tomada de [1]

simulación utilizado para el entrenamiento y experimentación sobre el modelo.

II. ESTADO DEL ARTE

Comandar un UAV en las condiciones mencionadas en la sección anterior demanda una toma de decisiones inteligente y ejecución precisa. Estos dos aspectos son áreas en las que los robots y los humanos pueden complementarse mutuamente, y por ende, colaborar eficazmente. Esta idea se inscribe en el marco más amplio de la autonomía compartida, un concepto que ya se ha explorado en estudios previos [2], [3]. Existen múltiples estrategias cooperativas de acuerdo a las estrategias de autoridad y los métodos de decisión sobre los diferentes modos de control [4]. Los avances en inteligencia artificial en este ámbito, especialmente en el desarrollo de algoritmos de aprendizaje por refuerzo que no requieren la especificación previa de un modelo [1], [5], [6], prometen allanar el camino para una adopción más amplia de estas tecnologías.

III. CONFIGURACIÓN EXPERIMENTAL

El modelo de cuadricóptero a utilizar es Crazyflie 2.1 [7] que se encuentra integrado al simulador Webots [8] que utilizaremos. Deepbots [9] nos ayuda a integrar el entorno OpenAIGym [10] al simulador Webots. La librería con los algoritmos de aprendizaje por refuerzo a usar es stable-baselines3 [11]. Todo el desarrollo será en lenguaje Python 3.x. En la figura 2 se detalla la integración de todas las librerías mencionadas.

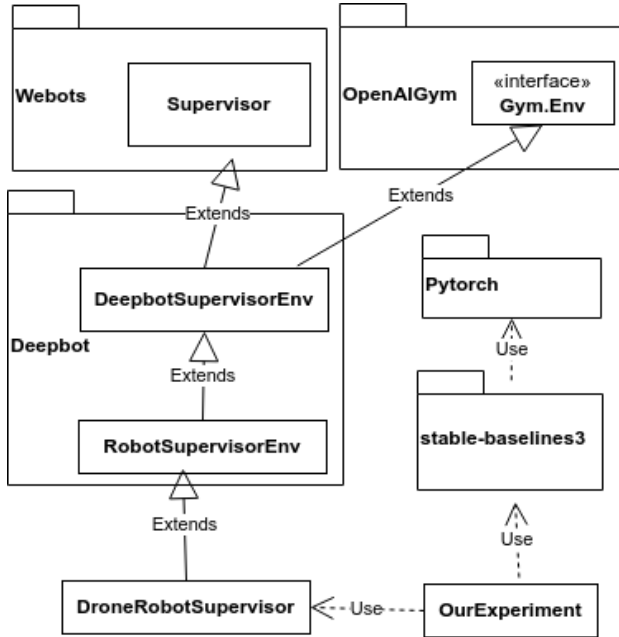


Figura 2. Diagrama de clases de la integración de las librerías para la configuración experimental. La clase *DroneRobotSupervisor* implementa el comportamiento del cuadricóptero en el entorno simulado. La clase *OurExperiment* utiliza la anterior para correr los experimentos de aprendizaje por refuerzo profundo.

Nuestros experimentos iniciales consideran el bucle cerrado indicado en la figura 1 pero sin intervención humana por el

momento. El enfoque utilizado mantiene un controlador de vuelo PID (controlador proporcional, integral y derivativo) de bajo nivel. El algoritmo utilizado para entrenar el agente, el comportamiento del cuadricóptero en el entorno, es *Proximal Policy Optimization* (PPO)[12]. El entrenamiento inicia luego del despegue del UAV. El estado u observaciones brindadas por el cuadricóptero durante la simulación son referidas a su posición (x,y,z) , velocidad en x y en y , ángulos de orientación $(roll,pitch,yaw)$, y distancia a obstáculos en cuatro direcciones (adelante, atrás, izquierda y derecha). Como salida del modelo indicamos cuánto se desea avanzar hacia adelante o ir hacia atrás, a los costados o cuánto se desea girar, valores que luego son recibidos por el PID para el control del vuelo. La función de recompensa del modelo penaliza ante la cercanía de obstáculos.

IV. CONCLUSIONES

Como parte del trabajo por realizar probaremos diferentes esquemas de configuración y algoritmos de aprendizaje por refuerzo, además de realizar pruebas con y sin piloto asistido. Finalmente este modelo entrenado de aprendizaje por refuerzo puede ser integrado al software del cuadricóptero Crazyflie. Sería un módulo que recibe información de las órdenes del copiloto y los sensores, brindando la salida para manipular el controlador de vuelo.

REFERENCIAS

- [1] S. Reddy, A. D. Dragan, and S. Levine, "Shared autonomy via deep reinforcement learning," *arXiv preprint arXiv:1802.01744*, 2018.
- [2] D. A. Abbink, T. Carlson, M. Mulder, J. C. De Winter, F. Aminravan, T. L. Gibo, and E. R. Boer, "A topology of shared control systems—finding common ground in diversity," *IEEE Transactions on Human-Machine Systems*, vol. 48, no. 5, pp. 509–525, 2018.
- [3] K. Backman, D. Kulić, and H. Chung, "Reinforcement learning for shared autonomy drone landings," *arXiv preprint arXiv:2202.02927*, 2022.
- [4] C. Yang, Y. Zhu, and Y. Chen, "A review of human-machine cooperation in the robotics domain," *IEEE Transactions on Human-Machine Systems*, vol. 52, no. 1, pp. 12–25, 2021.
- [5] C. Schaff and M. R. Walter, "Residual policy learning for shared autonomy," *arXiv preprint arXiv:2004.05097*, 2020.
- [6] T. Yoneda, L. Sun, B. Stadie, G. Yang, and M. Walter, "To the noise and back: Diffusion for shared autonomy," *arXiv preprint arXiv:2302.12244*, 2023.
- [7] Bitcraze, "Crazyflie 2.1 datasheet," open source flying development platform. [Online]. Available: https://www.bitcraze.io/documentation/hardware/crazyflie_2_1/crazyflie_2_1-datasheet.pdf
- [8] Webots, "http://www.cyberbotics.com," open-source Mobile Robot Simulation Software. [Online]. Available: <http://www.cyberbotics.com>
- [9] M. Kirtas, K. Tsampazis, N. Passalis, and A. Tefas, "Deepbots: A webots-based deep reinforcement learning framework for robotics," in *Artificial Intelligence Applications and Innovations: 16th IFIP WG 12.5 International Conference, AIAI 2020, Neos Marmaras, Greece, June 5–7, 2020, Proceedings, Part II 16*. Springer, 2020, pp. 64–75.
- [10] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "Openai gym," *arXiv preprint arXiv:1606.01540*, 2016.
- [11] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, "Stable-baselines3: Reliable reinforcement learning implementations," *Journal of Machine Learning Research*, vol. 22, no. 268, pp. 1–8, 2021. [Online]. Available: <http://jmlr.org/papers/v22/20-1364.html>
- [12] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.