

Contribución a un sistema de simulación para satélites federados mediante Hardware in the Loop

Arnau Dolz⁽¹⁾⁽²⁾, Joan A. Ruiz-de-Azua⁽¹⁾, Adriano Camps⁽³⁾⁽⁴⁾⁽⁵⁾

arnau.dolz@estudiantat.upc.edu, joan.ruizdeazua@i2cat.net, adriano.jose.camps@upc.edu

⁽¹⁾Space Communications Research Group, i2CAT Foundation, C/Gran Capità 2-4 Edificio Nexus I 08034 Barcelona, España

⁽²⁾Universitat Politècnica de Catalunya, UPC Campus Nord, 08034 Barcelona, España

⁽³⁾CommsSens Lab UPC, Dpto de Teoría de la Señal y Comunicaciones, BarcelonaTech, 08034 Barcelona, España

⁽⁴⁾Institut d'Estudis Espacials de Catalunya (IEEC), Edifici RDIT, Campus PMT-UPC, 08860 Castelldefels (Barcelona), España

⁽⁵⁾ASPIRE Visiting International Professor, Universidad de los Emiratos Árabes Unidos, CoE, PO Box 15551, Al Ain, EAU

Resumen—Federated Satellite Systems arise as a solution to enhance the performance of Earth observation and telecommunications systems in response to the increasing demand and proliferation of small satellites in orbit. However, the simulation equipment for space communications is limited in terms of communication network simulation. With the growing demand for 5G services, there is a need for network simulation tools that consider both orbital mechanics and satellite characteristics. This contribution proposes the creation of a simulation system using the Distributed Satellite Systems Simulator developed by NanosatLab of the Polytechnic University of Catalonia, currently maintained and developed by i2Cat. This text demonstrates how the Hardware in the Loop philosophy and necessary protocols to simulate Federated Satellite Systems are integrated in the simulator.

I. INTRODUCCIÓN

Desde el lanzamiento de los primeros CubeSats en junio de 2003, el número de este tipo de satélites que se han puesto en órbita ha aumentado exponencialmente. El incremento de este tipo de misiones ha hecho cambiar el paradigma de las misiones espaciales, dando lugar al concepto conocido como *New Space*. Gracias a este cambio en la forma de entender las misiones espaciales, ahora mismo muchas misiones ya se plantean en constelaciones de satélites en lugar de grandes satélites dedicados a cumplir un único cometido. Sin embargo, aunque el espacio se haya democratizado y abierto a muchos más actores, no todos pueden permitirse el coste de implementar una constelación. A pesar de ello, esto no significa que aquellos actores que no dispongan de los mismos recursos que otros, no puedan mejorar las capacidades de sus misiones aprovechando los recursos sin utilizar de otros satélites. De esta premisa, para paliar las limitaciones de las plataformas CubeSat, A. Golkar presenta el concepto de Sistema de Satélites Federados (FSS, por sus siglas en inglés) en [1]. Este concepto promueve la colaboración entre distintos satélites que no tienen por qué formar parte de la misma misión para que compartan aquellos recursos que no están siendo usados. Los recursos a compartir por los diferentes satélites de una federación pueden ser desde capacidad computacional, a almacenamiento de datos, hasta el uso de canal de downlink. Esta cooperación entre satélites permite mejorar el rendimiento y las capacidades de las misiones. Por otro lado, se debe considerar que el concepto y su viabilidad ya han quedado demostrados, como se puede ver en [2] a

través del *FSS Experiment* y con la misión FSSCat [3] del NanosatLab de la Universidad Politécnica de Cataluña (UPC).

Por otro lado, hay que entender que una federación entre satélites no es posible sin los Enlaces Entre Satélites (ISL por sus siglas en inglés). Además, es importante entender que los enlaces que permiten realizar una federación se consideran comunicaciones *Peer to Peer* (P2P). Para poder evaluar las capacidades en una misión y las características que puede aportar federar un satélite, se debe disponer de un software de simulación de redes que contemple las dinámicas orbitales. A raíz de esta necesidad y como uno de los resultados de [4] se desarrolló el Simulador de Sistemas de Satélites Distribuidos (DSS-SIM, por sus siglas en inglés). Este simulador permite la simulación de contactos entre satélites, y al estar basado en el Simulador de Redes 3 (NS3, por sus siglas en inglés), permite la simulación de redes de comunicación. A pesar de ello, hay que considerar que los protocolos necesarios para poder simular un FSS no están aún implementados en el DSS-SIM. Aunque en [5] se presenta una implementación de los protocolos para poder generar una simulación, solo uno de los dos protocolos necesarios está implementado. Así pues, el protocolo presentado en [2] debe ser implementado e incorporado al simulador.

Otro aspecto a añadir al DSS-SIM es la interacción entre la simulación y los dispositivos físicos. Esta necesidad nace de la creciente demanda de los servicios 5G, así como el testeo de dispositivos de Internet de las cosas (IoT, por sus siglas en inglés), con las redes de satélites, como se presenta en [6]. Así pues, en el documento se propone una posible solución para incorporar una interfaz que permita al DSS-SIM conectarse a dispositivos reales a través de una conexión Ethernet. Con esta interfaz, la filosofía Hardware in the Loop HIL, pasaría a estar integrada dentro del DSS-SIM y se podría testear.

Este documento se estructura de la siguiente forma. En la Sección 2, se presenta una discusión detallada sobre el concepto de federación y los protocolos necesarios para su implementación. En la Sección 3, se introducen las herramientas de simulación, incluyendo el DSS-SIM y la filosofía HIL. Finalmente, en la Sección 4, se presenta el diseño para la integración del HIL y los protocolos necesarios para la simulación de los FSS.

II. SISTEMAS DE SATÉLITES FEDERADOS

El concepto de Sistemas de Satélites Federados se presenta en [1]. Dicha definición presenta la cooperación entre satélites diseñados con distintos fines, con tal de mejorar las posibilidades que ofrecen en términos de misión. Sin embargo, con este nuevo paradigma no solo se pretende incentivar la cooperación entre satélites para mejorar las capacidades de las misiones, sino que se puede generar un nuevo modelo de negocio basado en el consumo oportunista de recursos por parte de distintos satélites.

Llegado este punto, es importante entender los distintos pasos que se deben dar para la creación de una federación. Los tres principales procesos son: *Negociación*, *Consumo* y *Cierre*. Asimismo, en [1] se definen tres tipos de FSS según la arquitectura que de los satélites y las funciones que realizan dentro de los procesos mencionados: (1) *FSS Centralizada*: Consiste en satélites que actúan como nodos dedicados exclusivamente en los procesos de negociación y distribución de los recursos, así como del proceso para cerrar la federación. (2) *FSS Negociada*: Consiste en Satélites que actúan como nodos de unión entre los satélites que ofrecen los recursos y aquellos que quieren consumirlos. A diferencia de la anterior, estos nodos solo se encargan de manejar el proceso de negociación. (3) *FSS Distribuida*: Esta arquitectura no requiere de nodos dedicados a ningún proceso específico. Esto quiere decir que son los propios satélites que tienen la capacidad de federarse los que deben encargarse de gestionar los tres procesos.

De los procesos descritos anteriormente y de las distintas arquitecturas, surge la necesidad de definir los protocolos necesarios para que estos se desarrollen correctamente. Así pues, en [5] se presenta el Protocolo de Disponibilidad de Servicios Oportunistas (OSADP, de sus siglas en inglés). Este protocolo se encarga de manejar de forma eficiente la gestión y el manejo de las publicaciones de recursos disponibles. Paralelamente, en [2] se define el Protocolo de Control de Despliegue de la Federación (FeDeCoP, de sus siglas en inglés). Dicho protocolo se encarga del manejo de los tres procesos descritos anteriormente (Negociación, Consumo y Cierre). A continuación se encuentra una pequeña introducción a ambos protocolos.

OSADP

Este protocolo se encarga de gestionar la publicación de recursos disponibles, así como el manejo de las publicaciones recibidas por parte de otros satélites [5]. De forma resumida, este protocolo se encarga de generar la publicación de un recurso y transmitirla de forma periódica cuando dicho recurso se halla disponible. Este tipo de paquetes se transmiten usando el Protocolo de Datagramas de Usuario (UDP, de sus siglas en inglés). Se usa este protocolo, ya que la publicación se hace de forma periódica. Cuando esto sucede, el mismo protocolo propone los pasos a seguir por parte de los receptores de dicha publicación. La Figura 1 muestra los procesos a realizar por aquellos nodos que reciben una publicación.

Hay 3 casos posibles: (1) El nodo está interesado en formar una federación y consumir un recurso, (2) El nodo no está dispuesto a formar parte de una federación para consumir el recurso, pero sí puede formar parte de la ruta para que

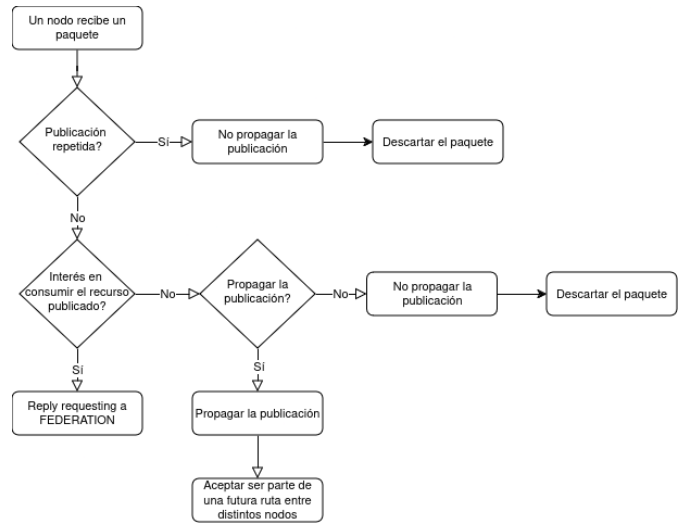


Fig. 1. Procesos a realizar tras recibir una publicación.

otros nodos lo usen como nodo de la red para establecer la federación, y (3) El nodo no está dispuesto a formar parte de la federación y descarta la publicación, puesto que tampoco está interesado en que otros nodos lo usen como nodo. Entonces, este protocolo gestiona la publicación y se puede considerar que acaba en cuanto el paquete se descarta, o bien cuando el nodo receptor informa al nodo que ha generado el paquete que está interesado en formar una federación. Entonces la negociación del consumo del recurso empieza. A partir de este punto, las comunicaciones pasan a realizarse mediante el Protocolo de Control de Transporte (TCP, de sus siglas en inglés), puesto que se considera que ya se ha empezado el proceso de negociación descrito por el protocolo FeDeCoP.

FeDeCoP

Como se presenta en [2] este protocolo se encarga de manejar los procesos necesarios para la creación, mantenimiento y desmantelamiento de una federación. La Figura 2 muestra los estados del protocolo de un FSS. Este protocolo es capaz de: (1) Gestionar la negociación de los recursos disponibles en el proceso de negociación entre los dispositivos que se van a federar. (2) Manejar el consumo de dichos recursos. Y finalmente (3) cerrar la conexión para finalizar la federación. Es importante entender que, dada la complejidad de las transacciones realizadas, un nivel de alto de fiabilidad es necesario. Por este motivo, este protocolo está basado en TCP.

Como se ha mencionado anteriormente, este protocolo puede considerar su punto de inicio tras el procesado de una publicación. Es entonces cuando se negocia el consumo del recurso, puesto que se puede dar el caso de que quien haya publicado el recurso ya no lo tenga disponible, o esté federado con otro dispositivo. Tras recibir la petición de federación, el nodo que haya publicado el recurso debe responder con el recurso y la cantidad disponible. Una vez se ha acordado qué cantidad de recursos se va a consumir, se acepta la petición y se da por establecida la federación. Una vez está la federación establecida y se ha pasado al estado de consumo, es cuando se explota el recurso. Hay que considerar que en un momento determinado se puede interrumpir la comunicación

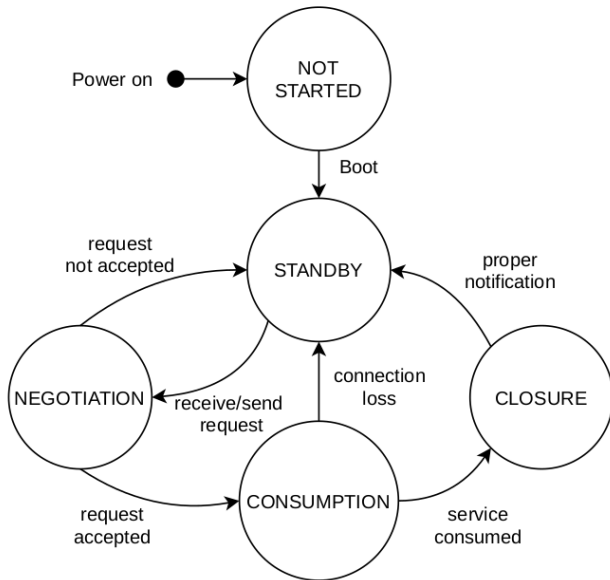


Fig. 2. Estados y transiciones del protocolo de FSS [2].

entre los miembros de la federación por cualquier motivo. Si esto sucede, se considera la conexión como perdida y se vuelve al estado inicial. Si la conexión no se interrumpe, y se puede consumir el recurso acordado en la fase previa, cuando este se acaba de consumir, se pasa a la fase de cierre. En esta fase, los nodos de la federación acceden a cerrarla de forma ordenada con los mensajes de cierre correspondientes especificados en [2]. Así queda explicado cómo el protocolo puede gestionar la creación y destrucción de federaciones de forma eficiente.

III. SISTEMAS DE SATÉLITES DISTRIBUIDOS

En [4], y [7], se hace un análisis detallado de los distintos sistemas de simulación de sistemas satélites distribuidos. En la discusión se acaba concluyendo que en general hay distintos sistemas con sus puntos a favor y sus puntos en contra. Sin embargo, hay dos factores clave para poder distinguirlos. El primero se trata del precio. Hay algunos sistemas de simulación propietarios que limitan el acceso a dichas herramientas, especialmente en el caso de la simulación de parámetros orbitales. Por otro lado, y más enfocado en el análisis de redes, los simuladores son poco modulares en cuanto a su utilización y expansión. A raíz de ello nace el DSS-SIM [4].

Por otro lado, los sistemas que se basan en combinar los simuladores de comunicaciones entre satélites y la emulación de hardware no solo están al alcance de pocas organizaciones debido a su elevado coste económico, sino que hay poca oferta en el mercado. En [7] se analizan también las ofertas disponibles en el mercado.

Sin embargo, en cuanto se intentan combinar la simulación de redes de comunicación con dispositivos hardware, las ofertas en el mercado son aún más reducidas que en los casos anteriores. Es por este motivo que este artículo pretende presentar una solución que combine la simulación generada por el DSS-SIM con las características que puede aportar un dispositivo real.

Simulador de Sistemas de Satélites Distribuidos

El DSS-SIM, es una herramienta basada en NS3. Se desarrolló como una solución para combinar la simulación de las dinámicas orbitales y las comunicaciones [7]. Dicho simulador permite al usuario implementar subsistemas de satélites y modelos de recursos personalizables por los usuarios. También proporciona una herramienta versátil y altamente extensible donde los usuarios pueden adaptar modelos físicos discretos que representan componentes e instrumentos de satélites.

Hay que mencionar que NS3 es un simulador por eventos, lo que significa que las simulaciones realizadas no son continuas en tiempo, sino discretas, lo que ahorra mucho tiempo de computación. Por otro lado, a diferencia de NS3, el DSS-SIM implementa su propio *NetDevice* llamado *SpaceNetDevice*. Este objeto es el que permite que se establezca la comunicación entre distintos dispositivos. Esta pieza será crucial para luego explicar la integración de los protocolos y los dispositivos hardware en el simulador.

Por otro lado, a pesar de sus ventajas, a continuación se presentará la integración de una interfaz para poder interconectar el simulador con dispositivos reales y poder ampliar así las posibilidades de esta herramienta.

Hardware in the Loop

Antes de presentar el diseño en sí, es importante definir que es el HIL. Para ello usamos la definición de National Instruments [8]: "Es una técnica en la que las señales reales de un controlador son conectadas a un sistema de pruebas que simula la realidad, engañando al controlador para que piense que está en el producto ensamblado". En este caso, la intención es conseguir conectar dispositivos con una interfaz Ethernet para que puedan representar sensores o satélites en órbita que incorporen los protocolos descritos anteriormente para así poder probar como reaccionarían sobre un dispositivo real y no uno simulado. En [7] se menciona el interés por parte distintos actores del sector espacial en desarrollar sistemas de simulación y emulación que cumplan con el diseño que se presenta en este artículo. Dada la definición de esta filosofía de testeo, y el motivo e integrarla en el DSS-SIM a continuación se presenta el diseño que pretende dar una solución a dicha problemática.

IV. INTEGRACIÓN DE PROTOCOLOS Y HIL

Ahora que se ha dado el contexto necesario, se puede explicar cómo se puede crear una interfaz para interconectar dispositivos reales con las simulaciones del DSS-SIM. Sin embargo, empezaremos explicando cómo integrar los protocolos de comunicación dentro del simulador. Como se presenta en [5] es posible implementar protocolos personalizados de forma satisfactoria. Sin embargo, en el caso del protocolo presentado en [2], la implementación debe ser revisada. Esto se debe al hecho de que el DSS-SIM sólo cuenta con una versión muy limitada de un protocolo de federaciones llamado *FSSProtocol* [5]. De este modo se seguirá el enfoque presentado en [5] para la implementación de ambos FeDeCoP y OSADP sobre el DSS-SIM. Los pasos a seguir consistirán en: (1) Aplicar una revisión sobre implementación del protocolo OSADP para que cumpla con los requisitos establecidos por los desarrolladores del DSS-SIM, y (2) Implementar el protocolo presentado en [2] para que los nodos del simulador puedan disponer de

FeDeCoP, valorando la implementación inicial presentada en [5].

Por otro lado, para poder integrar los dispositivos Hardware se usará el enfoque presentado en la Figura 3. En esta figura podemos ver como los dispositivos reales y los dispositivos simulados como *SpaceNetDevice* contendrán ambos protocolos (OSADP y FeDeCoP) necesarios para la gestión del FSS.

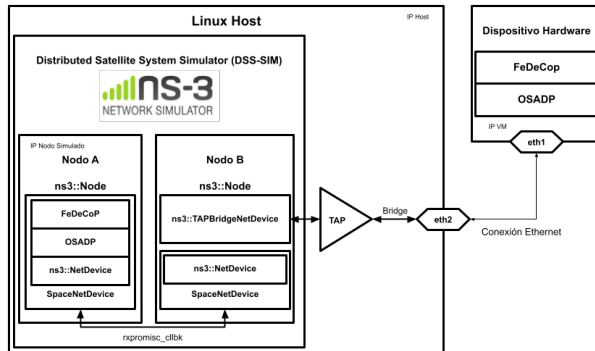


Fig. 3. Diseño de la integración y la interfaz en el DSS-SIM.

Asimismo, la Figura 3 también muestra la interfaz basada en NS3. La clase *TapBridgeNetDevice* de NS3 permite conectar el tráfico de red real al de la simulación. Esto sucede gracias a la interfaz Ethernet disponible en los ordenadores y los dispositivos que se van a conectar al simulador. Dicha de clase, dispone de un helper que se va a usar para instalar el `ns3::TapBridgeNetDevice` en el nodo de NS3 donde también está el *SpaceNetDevice*. Permitiendo que el TAP Bridge conecte la simulación a la red real. En resumen, el diseño presentado en la Figura 3 permite que el *SpaceNetDevice* se conecte a través del `ns3::TapBridgeNetDevice` al dispositivo TAP disponible en el sistema Linux. Accediendo finalmente a la red real, donde se podrá conectar con dispositivos físicos.

Con esta descripción a alto nivel del diseño se pretende dar solución a interconexión entre el simulador y los dispositivos reales. Esto hace posible la simulación de escenarios en que dispositivos hardware puedan disponer de conexión al DSS-SIM. Viendo cómo se comportaría un dispositivo real en el ambiente de las simulaciones generadas por el DSS-SIM. Además, tanto los nodos simulados, como los nodos reales, podrán incorporar los protocolos FeDeCoP y OSADP. Así mismo, esta contribución al DSS-SIM permite expandir las capacidades de éste simulador cubriendo las necesidades de mercado presentadas previamente. Esto permitirá la simulación de sistemas de satélites federados. Contribuyendo así a expandir las capacidades del DSS-SIM, y cubriendo una necesidad actual del mercado. Permitiendo ver el comportamiento de los dispositivos reales que incorporen los protocolos para crear FSS durante una simulación.

V. CONCLUSIONES Y LÍNEAS FUTURAS

Se ha introducido la necesidad de disponer de herramientas de simulación capaces de simular aquellos protocolos que se están desarrollando con tal de mejorar las capacidades de las redes de satélites. No obstante, se ha visto que no solo basta con disponer de un simulador de dinámicas orbitales, sino que debe ser capaz de simular protocolos de comunicación.

En este contexto, se ha presentado el DSS-SIM como una buena solución para cubrir esta necesidad. También, se han presentado los conceptos de FSS y HIL. Además, se han visto los protocolos necesarios para que se puedan crear los FSS (OSADP y FeDeCoP). Al tiempo que se ha introducido el HIL también se ha demostrado la necesidad de disponer de una interfaz para poder conectar el DSS-SIM con dispositivos reales. Por todo esto, en la sección anterior se ha presentado cómo se pueden implementar protocolos personalizados en el DSS-SIM. Esto permitirá incluir los protocolos necesarios para la creación y el manejo de FSS en el simulador. Por otro lado, se ha presentado cómo crear una interfaz que permita la interacción entre dispositivos reales y las simulaciones basadas sobre NS3 del DSS-SIM.

Como trabajo futuro, debe acabarse la implementación del diseño presentado. Además, se plantea la siguiente simulación: un dispositivo real que actúe como un nodo de una red que contenga los protocolos OSADP y FeDeCoP, esté conectado a un ordenador donde se ejecute una simulación del DSS-SIM. Por otro lado, se usará la simulación de los protocolos de FSS con dispositivos físicos para verificar su comportamiento con dispositivos que emularan el ordenador de a bordo de los PocketQubes *PoCat-2* i *PoCat-3* [9] que han sido seleccionados como parte de la cuarta edición del programa Fly Your Satellite (FYS) de la Agencia Espacial Europea. Los resultados de las simulaciones se presentarán en la conferencia.

REFERENCIAS

- [1] Alessandro Golkar and Ignasi Lluch i Cruz. The federated satellite systems paradigm: Concept and business case evaluation. *Acta Astronautica*, 111:230–248, 2015, DOI: <https://doi.org/10.1016/j.actaastro.2015.02.009>.
- [2] Joan A. Ruiz-de Azua, Nicola Garzaniti, Alessandro Golkar, Anna Calveras, and Adriano Camps. Towards federated satellite systems and internet of satellites: The federation deployment control protocol. *Remote Sensing*, 13(5), 2021, DOI: 10.3390/rs13050982.
- [3] Adriano Camps and et al. Fsscatt: The federated satellite systems 3cat mission: Demonstrating the capabilities of cubesats to monitor essential climate variables of the water cycle [instruments and missions]. *IEEE Geoscience and Remote Sensing Magazine*, 10(4):260–269, Dec 2022, DOI: 10.1109/MGRS.2022.3219778.
- [4] Juan Adrián Ruiz De Azúa Ortega. *Contribution to the development of autonomous satellite communications networks : the internet of satellites*. PhD thesis, UPC, Departament d'Enginyeria Telemàtica, Nov 2020.
- [5] Joan A. Ruiz-De-Azua, Anna Calveras, and Adriano Camps. A novel dissemination protocol to deploy opportunistic services in federated satellite systems. *IEEE Access*, 8:142348–142365, 2020, DOI: 10.1109/ACCESS.2020.3013655.
- [6] Simone Scrocciolani, Vincenzo Messina, Ramón María García Alarcia, Jaspas Sindermann, and Alessandro Golkar. Advancing satellite network performance: Network analysis for federated satellite systems. *IEEE Access*, 12:45616–45630, DOI: 10.1109/ACCESS.2024.3378743, 2024.
- [7] Arnau Dolz Puig. Contributions to simulation and emulation of inter-satellite links, Bachelor's Thesis, UPC, Escola Tècnica Superior d'Enginyeria de Telecomunicació de Barcelona, Departament de Teoria del Senyal i Comunicacions, Jul 2022, Disponible: <http://hdl.handle.net/2117/379917>.
- [8] National Instruments. ¿qué es el hardware en el bucle (hil)?, Acceso: 5-04-2024, Disponible: <https://www.ni.com/es/solutions/transportation/hardware-in-the-loop/what-is-hardware-in-the-loop-.html>.
- [9] Stefan Podaru, Guillem Gracia-Sola, and Adriano Camps. The ieee geoscience and remote sensing society “open pocketcube kit”: An affordable open source approach to earth observation missions [education in remote sensing]. *IEEE Geoscience and Remote Sensing Magazine*, 11(4):163–170, 2023.