Integrating Satellite Constellation and Mobile Operations for Non-Terrestrial Networks: Preliminary Results of Dynamic Scheduling

Arnau Singla
Space Communications Research Group
i2CAT Foundation
Barcelona, Spain
email: arnau.singla@i2cat.net

Franco Criscola
Aerospace Engineering
Embry-Riddle Aeronautical University
Daytona Beach, United States
email: CRISCOLF@my.erau.edu

David Canales
Aerospace Engineering
Embry-Riddle Aeronautical University
Daytona Beach, United States
email: CANALED4@erau.edu

Anna Calveras
Network Engineering
Universitat Politècnica de Catalunya
Barcelona, Spain
email: anna.calveras@upc.edu

Juan A. Fraire
Inria/INSA Lyon, CITI
CONICET-UNC
Villeurbanne, France
email: juan.fraire@inria.fr

Joan A. Ruiz-de-Azua
Space Communications Research Group
i2CAT Foundation
Barcelona, Spain
email: joan.ruizdeazua@i2cat.net

***Abstract*—The integration of terrestrial networks with Non-Terrestrial Networks requires a comprehensive management framework to address the mutual impact between satellite operations and network services. Past research by the authors introduced the Constellation Management System (CMS) as a solution, facilitating optimized plans for both satellite and network operators. This paper extends prior work by focusing on enhancing the adaptability of the CMS through dynamic execution methodologies, particularly by integrating telemetry feedback. The study presents a dynamic execution framework and a sequence diagram outlining the rescheduling process. Validation through telemetry emulation demonstrates the system's agility in responding to telemetry variations and the value of this new architecture using the CMS dynamically. Initial findings indicate a throughput enhancement of 10% with the implementation of a closed-loop approach compared to an open-loop approach. This research advances satellite constellation operations management towards automated communications, enhancing adaptability and robustness in unpredictable and constantly changing environments.**

***Keywords*—NTN; operations scheduling; satellite IoT; closed-loop optimization; 3GPP.**

I. INTRODUCTION

Over the last decade, the landscape of the space economy has undergone significant changes. While previously limited to governmental agencies mainly founded by public investment, technological advancements have increased private investment in space missions. This impact has been notable in Earth Observation (EO) and satellite telecommunications missions [1] [2]. One of the telecommunications domains poised to leverage the vast potential of satellite constellations is the realm of Internet of Things (IoT) [3]. Established terrestrial IoT networks stand to benefit significantly from augmented satellite coverage, particularly in remote and inaccessible regions, thereby extending ubiquitous global connectivity services [4]. This expansion proves especially valuable in areas where conventional IoT infrastructure faces technical and/or economic constraints, fostering a myriad of new application prospects [5].

The advent of 6th Generation (6G) technology heralds an era of expansive deployment of massive IoT networks, necessitating global coverage [6]. Consequently, there has been an effort to integrate satellite systems with ground-based telecommunications infrastructures, a trend underscored by initiatives, such as the Third Generation Partnership Project (3GPP) [7]. Notably, standardization efforts have led to the inclusion of spacecraft and aircraft within a 5th Generation (5G)-compliant architecture, thereby establishing them as Non-Terrestrial Networks (NTN) [8]. This represents an initial step toward integrating satellite systems with terrestrial infrastructure. Building upon this foundation, there is a growing emphasis on extending radio protocols, such as New Radio (NR) [9] and Narrow-Band Internet of Things (NB-IoT) [10], to accommodate satellite connectivity within the existing terrestrial framework, facilitating seamless integration of satellite systems into the broader telecommunications ecosystem [11].

The proposed solutions for satellite NB-IoT in the literature are mainly focused on Low Earth Orbit (LEO) constellations [12]. These constellations, often deployed for global coverage, may employ Store and Forward (S&F) mechanisms for data delivery in sparse constellations [13], adapting its protocols to 3GPP NB-IoT [14]. However, managing the operations of these LEO constellations, especially for telecom purposes, poses substantial challenges. The complexity arises from integrating satellite operations with mobile network operations. Examples are the diverse elements in a heterogeneous environment, the resource-constrained nature of satellite platforms, and the necessity to incorporate mobile network business criteria into satellite operations, among others.

Recognizing the need for efficient and autonomous management systems for telecom LEO constellations is crucial in this context [15]. Although available, traditional satellite-by-satellite operations and commercial solutions often fall short of meeting the intricate requirements of this evolving landscape. As current operations management research predominantly focuses on EO missions, there is a call for tailored solutions to address the unique challenges of the telecom use case effectively. Towards this goal, we have designed and developed an autonomous and reactive constellation management system, integrated with the Core Network (CN), to enhance the NTN architecture by extending IoT services with a LEO satellite constellation using S&F. The innovative system and its architecture are elaborated upon in this paper by the same individuals as the present study [16]. This study focuses on extending the previous capabilities of the Constellation Management System (CMS) and its dynamic capabilities, showcasing the autonomous integration of satellite telemetry feedback (e.g., power, memory) during continuous operation.

The rest of the paper is structured as follows. Section II explains the motivation behind a reactive scheduling system. Section III briefly presents the CMS. Then, Section IV describes the novel proposed dynamic execution of the CMS. After that, Section V depicts the scenario used for dynamic validation. The most relevant results to validate the execution are presented in Section VI. Finally, Section VII concludes the work and highlights possible future research topics.

II. Problem statement

Addressing the intricacies of a S&F LEO constellation as IoT-NTN, the CMS confronts numerous challenges from an operations perspective [17]. These scenarios often present highly heterogeneous environments where various entities with diverse priorities intersect. The optimization criteria of satellite and network/service operators differ, yet they are interlinked. Nevertheless, the CMS aims to devise a contact plan for satellites that optimally balances the requirements of both resource efficiency and the business criteria set forth by the service provider. It is essential to note that when extending existing terrestrial networks, operations must also be integrated into systems with specific standards (e.g., 3GPP).

Given the complexity of constellations characterized by discontinuous connectivity and limited onboard resources [2], centralized operations planning and optimization become imperative. However, the telecom scenario's unpredictability poses a challenge in planning, prompting the need for strategies to compensate for this evolving behavior. Viable solutions include utilizing a highly accurate traffic model [18] [19]. Nevertheless, a satellite constellation can serve different types of users or offer different services, making the traffic model not a suitable solution. So, as not to depend solely on the traffic model, another approach is enhancing the scheduling system's reactivity by dynamically incorporating feedback from the scenario and continually re-configuring the constellation contact plan, following a closed-loop optimization approach [20]. This work primarily focuses on addressing the latter challenge by assessing the impact of integrating telemetry feedback on the planned tasks' completion status and the constellation satellites' present resource levels. This evaluation is conducted using preliminary results obtained from a case study.

The telemetry feedback methodology has previously undergone testing in the GOMX-4 mission, aimed at refining battery prediction for the mission's dual satellites. Authors in [21] and [22] describe the GOMSPACE Hands-Off Operations Platform (HOOP), a commercial automatic satellite operations tool provided by the company. While sharing a fundamental concept with the CMS, HOOP is specifically optimized for EO missions, particularly in its scheduling engine. Notably, the referenced studies primarily focused on battery readings within their telemetry feedback, omitting considerations of scenario variability and disturbances. This current research contributes to the telemetry feedback approach, extending its application to telecommunications scenarios and incorporating additional layers to account for failed tasks and other resources, such as memory. This enhances the traffic model's precision and the management system's robustness to disturbances. The novelty of this work is optimizing satellite operations for telecom IoT services using a task scheduling approach and enhancing it with reactive planning to overcome the scenario unpredictability challenge.

III. Constellation Management System

The CMS is an independent satellite operations management system capable of integrating and orchestrating various agents by generating a resource- and business-aware schedule. This schedule, in turn, forms the foundation for generating: (1) a contact plan used by the satellite forwarding system, (2) an operations plan for the Mission Provider (MP), (3) a CN plan to coordinate traffic to and from terrestrial networks, and (4) a Ground Station Network (GSN) plan to synchronize the ground segment. The modular architecture of the CMS is represented in Figure 1.

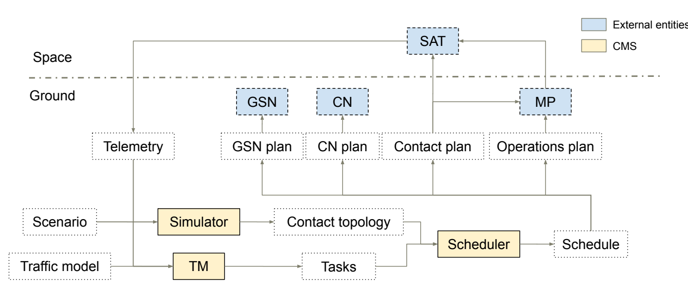

Figure 1. CMS global architecture.

The operator introduces scenario elements to the simulator module and the traffic model to the Task Manager (TM) module. The simulator, in turn, is entrusted with generating the contact topology of the problem, propagating contact windows between satellites and other elements (e.g., Ground Station (GS) or Service Area (SA)). Simultaneously, the TM produces input tasks for the scheduler module. The scheduler module employs a constraint satisfaction engine and artificial intelligence search algorithms to optimize task assignments.

The entire process and post-processing of the schedule into various plans and distributing them to their respective agents is overseen by the Plan Executor (PE) module. This module is responsible for dynamically controlling the execution of the CMS and managing the inputs and outputs of the system. Earlier work focused on the scheduler module, its model, and mathematical formulation [16]. This research focuses on the rest of the support modules surrounding the scheduler and their interactions to achieve reactive planning.

IV. Dynamic scheduling

This section describes the dynamic execution process of the CMS, along with the call flow process initiated upon triggering a reschedule. Figure 2 visually represents this dynamic execution through a sequence diagram.

Step 1 is a reschedule trigger to the plan executor. This trigger might come from different situations, such as (1) new inputs to the database, (2) a change in the scenario elements (e.g., a new SA), (3) a periodic time according to the scheduled time horizon, or (4) a manual trigger from the operator. The PE then starts the dynamic scheduling sequence by requesting the contact windows between the scenario satellites and the rest of the elements (GS, SA, and other satellites). The current scenario elements are stored in the simulator database. As such, if any of the scenario elements has changed, the PE will first store the new elements in the database (step 2) and then proceed to request a contact window search to the simulator module (step 3). Upon receiving a Windows request, the simulator shall fetch the scenario elements from its database, compute the contact windows, and store them in the database (steps 4-7). Finally, the simulator responds to the PE with the list of contact windows (step 8). If the scenario remains the same, then the PE directly requests the list of contact windows to the simulator database (steps 9-10). The database is periodically fed by the simulator with future contact windows. This way, the PE does not need to wait for the scenario propagation when requesting the contact windows, should the scenario remain the same.

Once the PE has the contact windows, the initial tasks are requested to the TM module (step 11). The TM requires the contact windows to generate the expected tasks according to the traffic model used. Before new tasks are computed, the TM keeps the assigned tasks of the previous schedule, which are yet to happen. The TM fetches the previous schedule and its timestamp from the schedule database (steps 12-13). Furthermore, the TM adds to the initial task list all previously assigned tasks reported to fail. The TM fetches the failed tasks from the telemetry database (steps 14-15). After computing the new tasks according to the traffic model and the scenario windows (step 16), the TM responds to the PE with the task list (step 17). In our task management system, each task is associated with an expiration date parameter, indicating when the data within the task becomes outdated. This parameter remains constant even if the task is rescheduled. As a result, our scheduler prioritizes tasks that are nearing their expiration date over newer tasks with later expiration dates. This ensures that we address impending data expiration and maintain data relevance within our system.

After that, the only thing required before sending the optimization request to the scheduler is to set the initial resources of each satellite. Towards that purpose, the PE fetches the predicted resources from the previous schedule stored in the schedule database (steps 18-19). The PE also fetches the latest telemetry updates regarding resources for each satellite from the telemetry database (steps 20-21). With all that information, the PE updates the predicted resources with the latest telemetry values and sets the initial values of the resources for that given point in time (step 22).

Finally, the PE module sends the scheduling request to the scheduler module and receives an optimized schedule (steps 23-25). To close the circle, the PE updates the schedule and telemetry databases with the resulting schedule and predicted resources (steps 26-27). The obtained optimized schedule is then prepared to be distributed to the corresponding agents like satellite operators and GSN operators (step 28).

V. Validation scenario

This section proposed a scenario to validate the dynamic execution of the CMS. This constellation consists of a Walker star pattern with four polar planes and 16 satellites, 90º:16/4/1 in Walker notation. The scenario GSN, composed of AWS and KSAT, is shown in Table I. Regarding the traffic demand, twelve different SA have been defined. Six are in Europe: Spain, France, Germany, the Baltic, Italy, and the UK. The other six are in the USA: Florida, Washington, Dallas, Minnesota, Portland, and Las Vegas. These twelve regions are assumed to follow the same traffic model. The task types modeled for this simulation are Mobile Originated Upload (MOUL), Mobile Originated Download (MODL), Mobile Terminated Upload (MTUL), and Mobile Terminated Download (MTDL). Mobile Originated (MO) traffic is the one originated in the User Equipment (UE) and towards the IoT server, and Mobile Terminated (MT) traffic the other way around. The traffic model assumes that each service area generates 3.8 KB of data every three hours, which corresponds to the modeled memory cost of a single MOUL task. Likewise, the same amount of data generation is assumed from the IoT server to the service areas, corresponding to a single MTUL task.

TABLE I
SCENARIO GROUND SEGMENT

Name	Latitude (º)	Longitude (º)
AWS Punta Arenas	-52.93	-70.85
AWS Sydney	-33.74	151.18
AWS CapeTown	-33.95	18.43
KSAT Tromso	69.66	18.94
KSAT Inuvik	68.32	-133.61

For this validation, two different tests are conducted within the same time frame, one with open-loop optimization and another with a dynamic closed-loop optimization. The telemetry

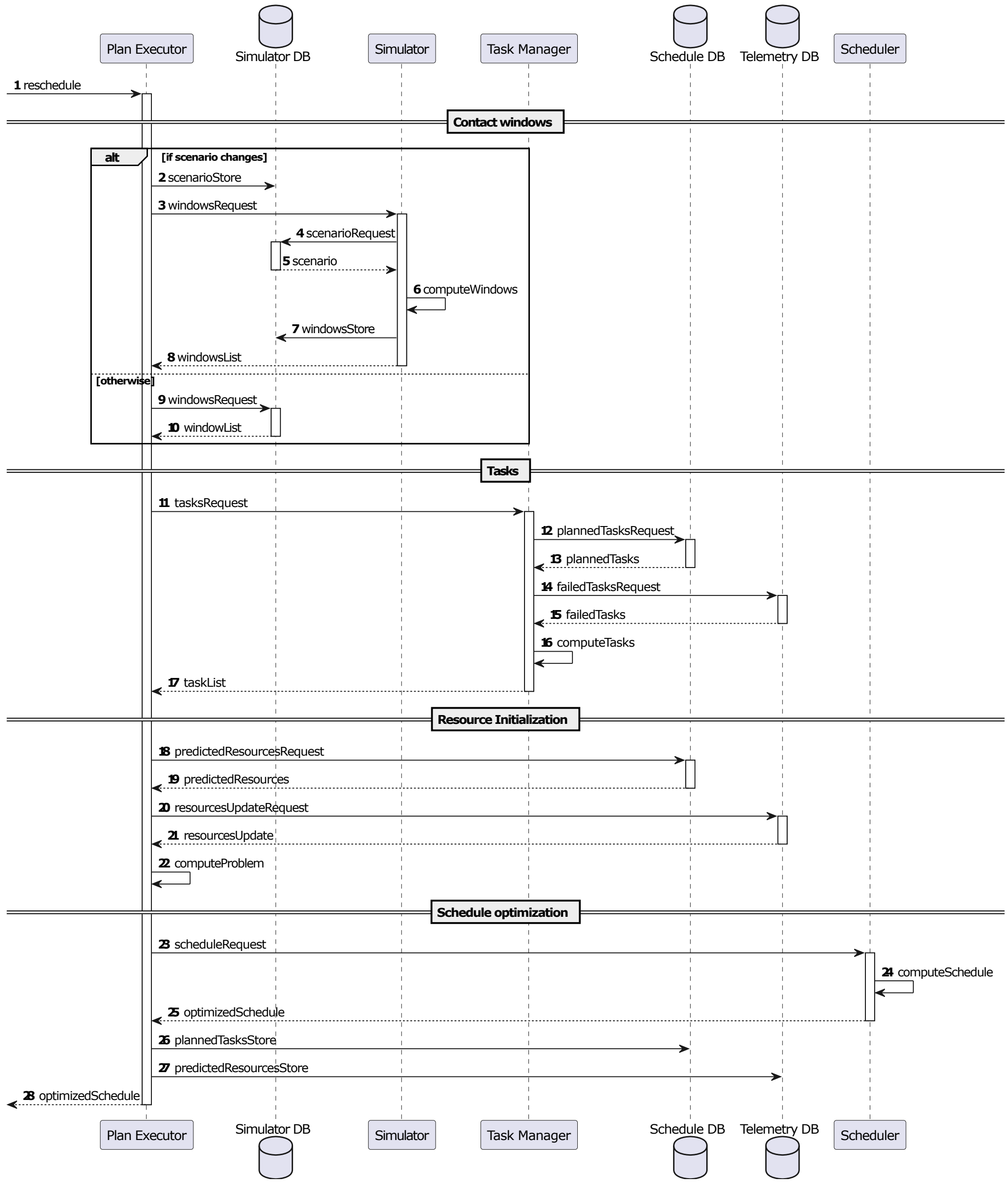

Figure 2. Sequence diagram of the CMS dynamic execution.

database does not contain any failed tasks at the beginning of the tests. Likewise, the schedule database is empty. After that, the PE is manually triggered to start, and approximately two hours later, new telemetry is added to the telemetry database. The telemetry will consist on new predefined resource levels for each satellite (within satellite resource boundaries), and a batch of failed tasks corresponding to the first 30 tasks of the original schedule, obtained two hours ago. In the closed-loop approach, the PE launches a reschedule every three hours with a time horizon of six hours. Therefore, an hour later to the telemetry input, a rescheduled request is automatically sent to the PE. For the open-loop test, the PE launches a reschedule at the end of the preceding schedule, in this case, every six hours. Figures 3 and 4 summarize the different test timelines using relative minutes. These tests aim to validate that the telemetry is correctly read and its changes are autonomously introduced in future schedules, both for the satellite resources and the failed tasks. Moreover, the open-loop test provides a fundamental reference point against which to assess the benefits of employing a closed-loop approach. Additional information on the software used in the simulations can be found in [16] and [23].

VI. Results

The results obtained from the test show that the telemetry feedback is correctly incorporated into the next autonomous operations schedule. Starting with the initial task list generated by the TM, Table II summarizes the number of tasks coming

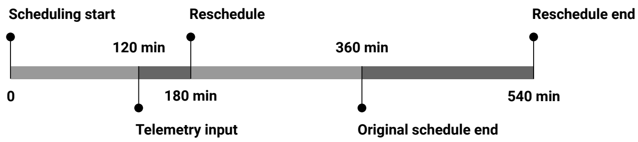

Figure 3. Closed-loop scheduling test timeline.

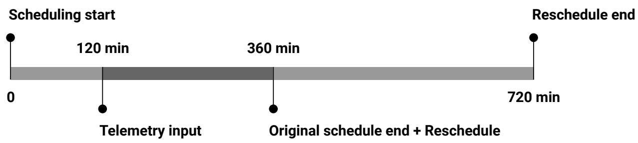

Figure 4. Open-loop scheduling test timeline.

from three different origins: (1) remaining tasks assigned in the last schedule, (2) failed tasks fetched from the telemetry database, and (3) the projected tasks until the end of the current schedule time horizon following the scenario traffic model. As can be seen, the starting schedule is not based on a past schedule. Therefore, no assigned tasks are still to be performed in the schedule database, nor any failed task in the telemetry database. Following the traffic model described in V and given that the total active SA is 12, the initial task list consists of only 96 tasks generated according to the traffic model. That is two MOUL and two MTUL tasks for each of the 12 SA and the same number of download tasks, a total of 8 tasks per SA. The automatic reschedule generated after three hours shows that it now consists of 129 tasks, divided into 51 remaining tasks from the original schedule, 30 failed tasks corresponding to the 30 first tasks of the original schedule, and 48 new tasks according to the traffic model. As can be seen, the TM has successfully fetched the original schedule and kept the assigned future tasks. It has also introduced the 30 tasks marked as failed in the telemetry database. Lastly, it has correctly computed the new projected tasks according to the traffic model, considering there are only three new hours to consider when generating new tasks. This yields one task of each type per SA, totaling 48.

The table also presents the mean throughput obtained in each schedule and for two different tests. The open-loop test serves as the baseline, and it involves initiating the rescheduling process after the preceding schedule time horizon, regardless of any telemetry alterations. As can be seen, the throughput obtained when using the close-loop approach explained in Section V increases by more than 10%. It is worth mentioning that in both schedules and tests, the scheduler assigned almost 100% of the initial tasks. This is because the scenario parameters allow an unconstrained scenario since this study focuses on dynamic scheduling rather than scheduler performance.

The other telemetry input that this dynamic scheduler integrates is the satellite resources. Figures 5 and 6 show a scenario of the satellite's memory and energy levels throughout the scheduling timespan. Three different plots are displayed:

TABLE II
TASK REASSIGNMENT RESULTS

	Original schedule	Reschedule
Remaining tasks	0	51
Failed tasks	0	30
Traffic model tasks	96	48
Total tasks	96	129
Schedule throughput (KB/h)	18.46	35.46
Open-loop test throughput (KB/h)	26.96	
Closed-loop test throughput (KB/h)	29.77	

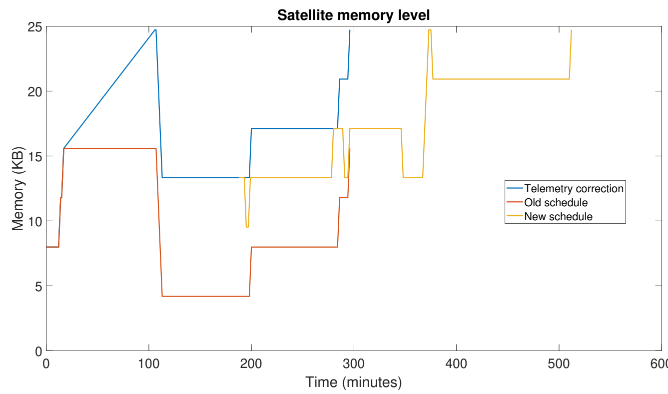

Figure 5. Satellite memory levels.

one for the propagated resources of the original schedule (old schedule in the figure legend), one for the telemetry correction on the propagated resources (telemetry correction in the figure legend), and another for the propagated resources of the reschedule (new schedule in the figure legend). As seen in both plots, the old schedule propagated resources and the ones corrected with the telemetry share the same pattern but diverge two hours into the propagation since this was when the telemetry was manually introduced to the database. It can also be observed that the new schedule starts from the telemetry-corrected levels of each resource. It is also worth mentioning that the new schedule is quite different from the old one since random telemetry values are added to each satellite, and 30 failed tasks have to be reassigned as well. Therefore, the scheduler has redistributed the tasks among all the 16 satellites of the constellation accordingly.

VII. CONCLUSION

This paper is a step forward in tackling the challenge of enhancing the robustness of the CMS, bringing it closer to the integration of network operations and satellite operations in 3GPP NTN. This work delves into the implementation of closed-loop operations management within the CMS framework, aiming to mitigate disruptions stemming from deviations in operational scheduling. This work presents an architecture enabling the autonomous integration of telemetry feedback and a sequence diagram outlining the rescheduling process. The outcomes of a case study underscore the necessity for dynamic scheduling and its potential advantages. The findings illustrate the system's ability to swiftly react, often within minutes, to

 ISBN: 978-1-68558-179-4

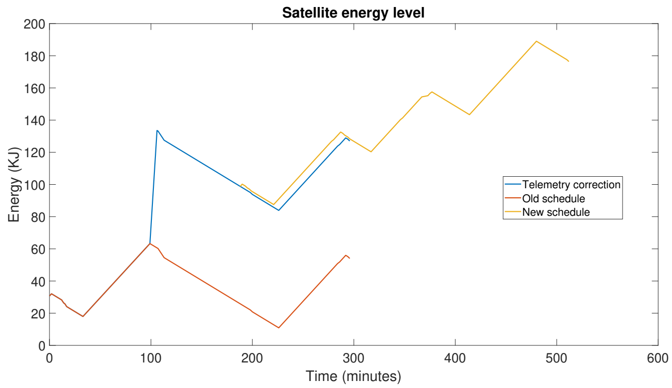

Figure 6. Satellite energy levels.

significant fluctuations in telemetry data, effectively managing variations in satellite resource levels and addressing task failures. Preliminary results have shown to improve throughput by 10% when using a closed-loop approach over an open-loop approach. Future research aims to optimize the timing of rescheduling, considering factors, such as scenario reactivity, telemetry fluctuations, and scheduling efficiency, among others. This research marks a significant step in enhancing the robustness and adaptability of centralized satellite constellation operations management in the face of dynamic and changing scenarios.

Acknowledgment

This work has been funded by the Government of Catalonia in the scope of the NewSpace Strategy for Catalonia. Additionally, this work was funded by the Spanish Ministry of Economic Affairs and Digital Transformation and the European Union – NextGeneration EU, in the framework of the Recovery Plan, Transformation and Resilience (PRTR) (Call UNICO I+D 5G 2021, ref. number TSI-063000-2021-1-6GSatNet-GS). This research was funded in part by the Spanish MCIN/AEI/ 10.13039/501100011033 through project PID2019-106808RA-I00, and by Secretaria d'Universitats i Recerca del departament d'Empresa i Coneixement de la Generalitat de Catalunya with the grant number 2021 SGR 00330. This paper is also part of the Grant 6G-DIFERENTE (CER-20231018) funded by CDTI and by the European Union NextGenerationEU/PRTR within the call "Ayudas Cervera para Centros Tecnológicos 2023. The research was partially supported by the Embry-Riddle Aeronautical University Faculty Research Development Program and GAANN fellowship.

References

[1] O. Kodheli, E. Lagunas, N. Maturo, S. K. Sharma, B. Shankar, J. F. M. Montoya, J. C. M. Duncan, D. Spano, S. Chatzinotas, S. Kisseleff, J. Querol, L. Lei, T. X. Vu, and G. Goussetis, "Satellite Communications in the New Space Era: A Survey and Future Challenges," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 1, pp. 70–109, 2021.

[2] J. A. R. de Azua, A. Calveras, and A. Camps, "Internet of Satellites (IoSat): Analysis of Network Models and Routing Protocol Requirements," *IEEE Access*, vol. 6, pp. 20 390–20 411, 2018.

[3] M. M. Azari, S. Solanki, S. Chatzinotas, O. Kodheli, H. Sallouha, A. Colpaert, J. F. M. Montoya, S. Pollin, A. Haqiqatnejad, A. Mostaani, E. Lagunas, and B. Ottersten, "Evolution of Non-Terrestrial Networks From 5G to 6G: A Survey," *arXiv preprint arXiv:2107.06881*, 2021.

[4] S. Wang and Q. Li, "Satellite Computing: Vision and Challenges," *IEEE Internet of Things Journal*, vol. 10, no. 24, pp. 22 514–22 529, Dec. 2023.

[5] J. A. Fraire, O. Iova, and F. Valois, "Space-Terrestrial Integrated Internet of Things: Challenges and Opportunities," *IEEE Communications Magazine*, vol. 60, no. 12, pp. 64–70, dec 2022.

[6] M. Harounabadi and T. Heyn, "Toward Integration of 6G-NTN to Terrestrial Mobile Networks: Research and Standardization Aspects," *IEEE Wireless Communications*, vol. 30, no. 6, pp. 20–26, Dec. 2023.

[7] M. Marchese, F. Patrone, and A. Guidotti, *The Role of Satellite in 5G and Beyond*. Cham: Springer International Publishing, 2023, pp. 41–66. [Online]. Available: https://doi.org/10.1007/978-3-031-30762-1_2

[8] 3rd Generation Partnership Project, "Study on architecture aspects for using satellite access in 5G (TR 23.737 Release 17)," Technical Specification Group, Tech. Rep., 2021.

[9] ——, "Solutions for NR to support non-terrestrial networks (NTN) (TR 23.737 Release 17)," Technical Specification Group, Tech. Rep., 2023.

[10] ——, "RadioAccess Network; Study on Narrow-Band Internet of Things (NB-IoT)/ Enhanced Machine Type Communication (eMTC) Support for NonTerrestrial Networks (NTN) (Release 17)," Technical Specification Group, Tech. Rep., 2022.

[11] M. Luglio, M. Quadrini, C. Roseti, and F. Zampognaro, "Scenarios and implementation use-cases for satellite-based NB-IoT," in *2021 International Symposium on Networks, Computers and Communications (ISNCC)*. IEEE, Oct. 2021.

[12] R. B. Sørensen, H. K. Møller, and P. Koch, "5G NB-IoT via low density LEO Constellations," in *Small Satellite Conference*. arXiv, 2021.

[13] E. Birrane and J. Soloff, *Designing Delay-Tolerant Applications for Store-and-Forward Networks*. Artech House, 2020.

[14] T. Kellermann, R. P. Centelles, D. Camps-Mur, R. Ferrus, M. Guadalupi, and A. C. Auge, "Novel Architecture for Cellular IoT in Future Non-Terrestrial Networks: Store and Forward Adaptations for Enabling Discontinuous Feeder Link Operation," *IEEE Access*, vol. 10, pp. 68 922–68 936, 2022.

[15] W. Yang, L. He, X. Liu, and Y. Chen, "Onboard coordination and scheduling of multiple autonomous satellites in an uncertain environment," *Advances in Space Research*, vol. 68, no. 11, pp. 4505–4524, Dec. 2021.

[16] A. Singla, F. Criscola, E. Ponce, D. Canales, A. Calveras, and J. A. Ruiz-De-Azua, "Towards 6G Non-Terrestrial Networks - An Autonomous Constellation Management Engine," 2022, 33rd AAS/AIAA conference. Pending proceedings. [Online]. Available: https://www.researchgate.net/publication/369541726_Towards_6G_Non-Terrestrial_Networks_-An_Autonomous_Constellation_Management_Engine

[17] A. Iqbal, M.-L. Tham, Y. J. Wong, A. Al-Habashna, G. Wainer, Y. X. Zhu, and T. Dagiuklas, "Empowering Non-Terrestrial Networks With Artificial Intelligence: A Survey," *IEEE Access*, vol. 11, pp. 100 986–101 006, 2023.

[18] Q. Chen, Y. Zhang, J. Guo, L. Yang, C. Fan, Y. Zhao, and X. Chen, "Traffic Prediction Based on Surrogate Mdel in Satellite Constellation Networks," in *2019 12th IFIP Wireless and Mobile Networking Conference (WMNC)*. IEEE, Sep. 2019.

[19] M. Lopez-Benitez, C. Majumdar, and S. N. Merchant, "Aggregated Traffic Models for Real-World Data in the Internet of Things," *IEEE Wireless Communications Letters*, pp. 1–1, 2020.

[20] R. D. McAllister, J. B. Rawlings, and C. T. Maravelias, "The inherent robustness of closed-loop scheduling," *Computers & Chemical Engineering*, vol. 159, p. 107678, Mar. 2022.

[21] G. Stock, J. A. Fraire, T. Mömke, H. Hermanns, F. Babayev, and E. Cruz, "Managing fleets of LEO satellites: Nonlinear, optimal, efficient, scalable, usable, and robust," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 39, no. 11, pp. 3762–3773, 2020.

[22] G. Stock, J. A. Fraire, H. Hermanns, E. Cruz, A. Isaacs, and Z. Imbrosh, "On the automation, optimization, and in-orbit validation of intelligent satellite constellation operations," 2022.

[23] A. Singla, A. Calveras, F. Betorz, and J. A. Ruiz-De-Azua, "Enhancing Satellite Non-Terrestrial Networks Through Advanced Constellation Management: Optimizing In-Orbit Resources for NB-IoT," *IEEE Open Journal of the Communications Society*, vol. 5, pp. 2113–2131, 2024.