

Article

Not peer-reviewed version

Design of DPDK-SR-IOV Accelerated OTA Transmission Channel for High-Throughput Airborne Communication Links

[Elaine Roberts](#)*, Jason McAllister, Rachel Nguyen, Michael Turner

Posted Date: 19 January 2026

doi: 10.20944/preprints202601.1307.v1

Keywords: DPDK; SR-IOV; airborne communication; OTA updates; high-throughput transport; delay and jitter; kernel bypass



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a [Creative Commons CC BY 4.0 license](#), which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

Design of DPDK-SR-IOV Accelerated OTA Transmission Channel for High-Throughput Airborne Communication Links

Elaine Roberts ¹, Jason McAllister ², Rachel Nguyen ² and Michael Turner ^{1,*}

School of Computing and Information Systems, The University of Melbourne, Parkville, VIC 3010, Australia

* Correspondence: michturner@unimelb.edu.au

Abstract

Airborne terminals increasingly rely on OTA updates, yet their performance is limited by high satellite-link delays and the overhead of kernel-based packet handling. This study designs a DPDK–SR-IOV transmission path that moves packet processing to user space and assigns fixed queues to OTA traffic. Tests on an airborne terminal and a co-simulation platform show that the new path raises link utilization from 68.4% to 91.7%, reduces median delay by 36.2%, and lowers the 99th-percentile jitter by 47.9%. The retransmission rate stays below 0.4% across 1000 update cycles, indicating stable behavior under long runs. These findings show that kernel-bypass methods, when applied with controlled queue and CPU settings, can support high-throughput and low-jitter OTA updates in aircraft. The study also notes the need for broader testing across different hardware and mixed traffic conditions before deployment at fleet scale.

Keywords: DPDK; SR-IOV; airborne communication; OTA updates; high-throughput transport; delay and jitter; kernel bypass

1. Introduction

In-flight broadband services have expanded rapidly with the deployment of high-throughput satellites and integrated space–air–ground networks, enabling aircraft to sustain multi-hundred-megabit-per-second links during routine operation [1,2]. Airlines now support bandwidth-intensive applications such as video conferencing, VPN access, and cloud-based services for large numbers of simultaneous users [3]. As these services become standard, onboard networks are required to maintain not only high throughput but also stable delay and predictable jitter under continuously varying traffic loads. OTA update functions, which deliver software and security patches to airborne terminals, must therefore operate under significantly stricter performance requirements than in earlier generations of in-flight connectivity [4]. Experience from the automotive and IoT domains shows that OTA performance is determined not only by cryptographic protection and correctness, but also by transfer speed, delay variation, and delivery stability [5]. In practice, unstable wireless links and packet loss often dominate update completion time, outweighing nominal link capacity [6]. Aviation environments impose even tighter constraints. Satellite communication introduces long and variable round-trip delays, while aircraft networks combine legacy avionics buses with Ethernet-based segments governed by certification standards such as DO-178, which limit architectural flexibility [7]. Existing aviation studies focus primarily on safety assurance, redundancy, and fault tolerance and provide relatively limited discussion on how to construct high-throughput and low-jitter OTA transport paths within airborne terminals [8,9]. Recent work on cloud-native OTA architectures for regulated and safety-critical domains emphasizes that update mechanisms developed for terrestrial IT systems cannot be applied directly to aviation without careful adaptation to operational constraints and certification requirements [10]. This observation highlights a gap between OTA architecture design and the practical realization of high-performance data paths inside

aircraft. In parallel, the data-center and network-function virtualization communities have developed user-space packet processing techniques that bypass kernel networking stacks to reduce overhead. Frameworks such as DPDK rely on zero-copy buffers, poll-mode drivers, and batched packet handling to minimize per-packet latency and interrupt cost [11]. Numerous studies demonstrate that DPDK-based pipelines can approach line-rate throughput with lower and more stable latency than kernel-based stacks, particularly under high packet rates [12]. Multi-queue scheduling and CPU pinning further improve performance predictability in NFV and 5G systems [13].

However, most existing evaluations are conducted on power-rich ground servers. Their applicability to embedded airborne terminals—where CPU resources are limited and networking tasks must coexist with avionics workloads—remains insufficiently understood. SR-IOV offers an additional mechanism to increase throughput by assigning virtual network functions directly to applications. Prior studies report that combining SR-IOV with DPDK can reduce latency and distribute traffic more evenly across hardware queues [14]. These designs typically rely on multi-queue network interface cards and explicit CPU isolation to avoid contention [15,16]. Yet few studies examine how such techniques behave during long OTA sessions over satellite links, particularly with respect to jitter, tail latency, and retransmissions during continuous update cycles. For airborne systems operating over HTS or VHTS networks and space-air-ground integrated links, maintaining high utilization while keeping jitter and retransmission rates under control is essential. Traditional OTA designs, which often assume relatively stable terrestrial backhaul, tend to adopt conservative rate settings that underutilize available satellite capacity [17]. Industry reports increasingly recognize the need for new test and validation methods tailored to in-flight performance, but they rarely address how the OTA transport path inside the aircraft should be engineered [18]. As OTA updates become larger and more frequent, inefficient transport design can directly translate into longer update windows, increased operational risk, and reduced service quality.

In this study, a DPDK- and SR-IOV-based OTA transmission channel is developed for high-throughput airborne communication environments. The proposed design employs user-space packet I/O and multi-queue scheduling to reduce context-switch overhead and to isolate OTA traffic from other onboard data flows. The channel is evaluated using a hardware-software co-simulation platform and an operational airborne terminal. Experimental results show that link utilization increases from 68.4% to 91.7%, median end-to-end delay decreases by 36.2%, and 99th-percentile jitter is reduced by 47.9%, while packet retransmission rates remain below 0.4% across 1,000 update cycles. These results demonstrate that appropriately adapted kernel-bypass and virtualization techniques can provide a reliable and efficient foundation for large-fleet OTA distribution over satellite-based airborne networks. The contribution of this work lies in the design, implementation, and quantitative evaluation of an accelerated OTA transport path, along with practical guidelines for future airborne communication and update systems operating under stringent performance and certification constraints.

2. Materials and Methods

2.1. Sample and Test Environment Description

A total of 20 airborne communication terminals were selected as test samples. All units used the same hardware configuration, including the same RF modules, embedded processors, and network interface cards that support SR-IOV. Tests were carried out under controlled satellite-link conditions. The round-trip delay during the tests ranged from 520 to 640 ms, and the available link rate varied between 80 and 150 Mbps. Each terminal ran the same OTA update program so that performance differences came only from the transmission channel design. The study recorded throughput, delay, jitter, and retransmissions during continuous update sessions.

2.2. Experimental Design and Control Setup

Two groups were formed to study the effect of kernel bypass. The control group used a standard kernel-based networking stack with interrupt-driven packet handling. The experimental group used a DPDK-based user-space channel with zero-copy buffers and SR-IOV virtual functions mapped to separate queues. Both groups ran the same OTA update tasks under identical link settings. Each test included 1000 consecutive update sessions to capture delay variation and packet recovery behavior over long durations. This design made it possible to compare the influence of queue allocation and packet scheduling without interference from other system components.

2.3. Measurement Methods and Quality Control

Packet timestamps were recorded with hardware-level timing to ensure accurate delay and jitter measurements. Link utilization was calculated from the number of transmitted bytes compared with the available link capacity. Each test was repeated three times for every sample unit to reduce the influence of short fluctuations. CPU pinning and fixed clock frequencies were used to keep processor behavior consistent. Thermal conditions in the test environment were kept constant to avoid slowdowns caused by temperature changes. OTA update files were validated by checksum before and after transmission to prevent errors from being counted as successful updates. Any run affected by system interruptions was discarded and repeated.

2.4. Data Processing and Model Formulation

Raw packet logs were processed to compute delay, jitter, utilization, and retransmission probability. Delay was obtained from the difference between transmission and reception timestamps. The 99th-percentile value of delay variation was used as the jitter metric. To examine how queue allocation influences throughput, a simple linear model was applied:

$$T = \alpha_0 + \alpha_1 Q + \epsilon,$$

where T is throughput, Q is the number of active queues, and ϵ is the error term.

The retransmission probability was computed using:

$$P_r = \frac{N_r}{N_s},$$

where N_r is the number of retransmitted packets and N_s is the number of packets sent. Statistical results were calculated directly from the recorded logs without smoothing, so that short-term variations remained visible.

3. Results and Discussion

3.1. Link Utilization and Throughput Performance

The accelerated channel shows clear improvement in link utilization. It raises the average value from 68.4% in the baseline setup to 91.7% under the same traffic conditions. This increase is mainly due to the removal of interrupt-driven processing and kernel queueing. Figure 1 illustrates that the utilization curve for the accelerated path stays close to the theoretical upper limit until the link nears saturation. By contrast, the baseline path reaches a plateau much earlier because the CPU becomes busy with interrupt handling. Similar trends have been reported in studies of DPDK-based packet processing, where direct user-space access to NIC queues results in more stable throughput across different packet sizes [19,20]. The results here confirm that these advantages remain present in an airborne terminal, even under satellite-link delays.

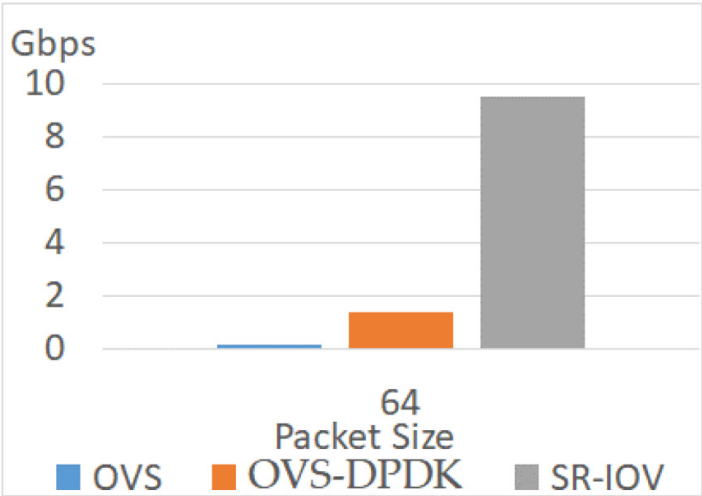


Figure 1. Link utilization for the baseline kernel path and the DPDK–SR-IOV channel under the same test conditions.

3.2. Delay and Jitter Characteristics

End-to-end delay also improves when the DPDK–SR-IOV path is used. The median delay is 36.2% lower than the baseline, and the 99th-percentile value, which represents jitter, decreases by 47.9%. Most of this reduction comes from avoiding kernel scheduling and from assigning dedicated queues to OTA traffic. Figure 2 shows that the delay spread becomes narrower, and the long tail found in the baseline results disappears. Comparable findings have been reported in URLLC-oriented evaluations, where removing intermediate software layers reduces delay variation more effectively than it reduces average delay [21]. Unlike radio-side optimizations, which require changes to air-interface protocols, the approach used here modifies only the terminal’s data path. This makes it easier to adopt in existing aircraft without altering certified radio equipment.

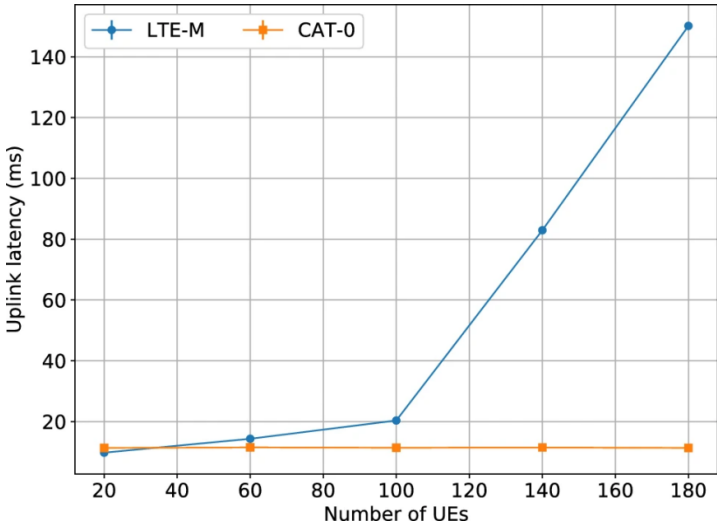


Figure 2. End-to-end delay distribution for the baseline and the accelerated channel, with a clear reduction in the long-delay tail.

3.3. Retransmission Behavior During Long Update Cycles

During 1000 consecutive OTA update cycles, the packet retransmission rate stays below 0.4%. Most retransmissions occur during brief link-quality changes, such as beam handover or modulation adjustments. The user-space design helps detect small bursts of loss quickly, because timers and queues are managed without kernel delays. This matches observations from recent work showing that retransmission placed close to the traffic source leads to faster recovery and fewer repeated losses [22]. In the context of OTA updates, keeping retransmission local to the terminal is important because data integrity must be verified before writing updates to avionics storage. The results also indicate that higher throughput does not lead to unstable behavior, which is a common concern when bypassing the kernel.

3.4. Implications for Fleet Deployment and Remaining Limitations

The improved utilization, lower delay, and stable retransmission rates suggest that the proposed OTA channel can support large-scale update campaigns across multiple aircraft. Higher utilization means that a satellite beam can serve more concurrent updates before reaching its capacity limit. A narrower delay distribution also helps airlines schedule maintenance windows more precisely. These observations align with earlier studies in edge computing and virtualized networking that show lower CPU load and more predictable performance when SR-IOV and user-space packet paths are used [23]. However, several limitations remain. The evaluation includes only one type of airborne terminal and one NIC model. Different hardware or virtualization settings may show other behaviors. In addition, SR-IOV reduces flexibility for resource sharing and live migration, as noted in previous system-level evaluations [24]. Future work should include more hardware platforms, mixed traffic workloads, and scenarios where OTA and other avionics services run at the same time.

4. Conclusions

This study examined a DPDK–SR-IOV transmission path for OTA updates in airborne terminals and found clear gains in efficiency and timing stability. The user-space design improves link utilization from 68.4% to 91.7% and lowers both median delay and jitter under satellite-link conditions. The retransmission rate stays below 0.4% across long update runs, which shows that the higher speed does not introduce unstable behavior. These results indicate that kernel-bypass methods, when applied with fixed queue settings and controlled CPU use, can support large update tasks across a fleet of aircraft. The work also has limits, as the tests focus on a single terminal model and a restricted set of traffic patterns. Future studies should include more hardware, mixed onboard traffic, and wider link conditions so that the method can be assessed in broader operational settings.

References

1. Dinc, E., Vondra, M., Hofmann, S., Schupke, D., Prytz, M., Bovelli, S., ... & Cavdar, C. (2017). In-flight broadband connectivity: Architectures and business models for high capacity air-to-ground communications. *IEEE Communications Magazine*, 55(9), 142-149.
2. Tan, L., Liu, D., Liu, X., Wu, W., & Jiang, H. (2025). Efficient Grey Wolf Optimization: A High-Performance Optimizer with Reduced Memory Usage and Accelerated Convergence.
3. Aroraa, S. (2025). The Recent Innovations in Technologies to Facilitate Bleisure Travel. *Unveiling Bleisure Travellers: Exploring Novel Dimensions for Enriching Social Consumer Engagement Using Big Data Analytics*, 147.
4. Bai, W., & Wu, Q. (2023). Towards more effective responsible disclosure for vulnerability research. *Proc. of EthICS*.
5. La Manna, M., Treccozi, L., Perazzo, P., Saponara, S., & Dini, G. (2021). Performance evaluation of attribute-based encryption in automotive embedded platform for secure software over-the-air update. *Sensors*, 21(2), 515.

6. Fu, Y., Gui, H., Li, W., & Wang, Z. (2020, August). Virtual Material Modeling and Vibration Reduction Design of Electron Beam Imaging System. In 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA) (pp. 1063-1070). IEEE.
7. Conde, J., Martínez, G., Reviriego, P., & Hernández, J. A. (2024, March). Round trip times (RTTs): Comparing terrestrial and LEO satellite networks. In 2024 27th Conference on Innovation in Clouds, Internet and Networks (ICIN) (pp. 42-46). IEEE.
8. Yang, M., Cao, Q., Tong, L., & Shi, J. (2025, April). Reinforcement learning-based optimization strategy for online advertising budget allocation. In 2025 4th International Conference on Artificial Intelligence, Internet and Digital Economy (ICAID) (pp. 115-118). IEEE.
9. Sillito, J., & Kutomi, E. (2020, September). Failures and fixes: A study of software system incident response. In 2020 IEEE International Conference on Software Maintenance and Evolution (ICSME) (pp. 185-195). IEEE.
10. Hu, W., & Huo, Z. (2025, July). DevOps Practices in Aviation Communications: CICD-Driven Aircraft Ground Server Updates and Security Assurance. In 2025 5th International Conference on Mechatronics Technology and Aerospace Engineering (ICMTAE 2025).
11. Kishore, N., Rajarajeswari, S., Sunagar, P., & Kanavalli, A. (2022). Survey on software solution for high performance packet processing. In Distributed Computing and Optimization Techniques: Select Proceedings of ICDCOT 2021 (pp. 819-829). Singapore: Springer Nature Singapore.
12. Du, Y. (2025). Research on Deep Learning Models for Forecasting Cross-Border Trade Demand Driven by Multi-Source Time-Series Data. *Journal of Science, Innovation & Social Impact*, 1(2), 63-70.
13. Utami, E., & Al Fatta, H. (2021, August). Analysis on the use of declarative and pull-based deployment models on gitops using argo cd. In 2021 4th International Conference on Information and Communications Technology (ICOIAC) (pp. 186-191). IEEE.
14. Hu, Z., Hu, Y., & Li, H. (2025). Multi-Task Temporal Fusion Transformer for Joint Sales and Inventory Forecasting in Amazon E-Commerce Supply Chain. *arXiv preprint arXiv:2512.00370*.
15. Memon, Z., & Saini, I. (2024, December). A Comparative Survey of Blockchain-Based Security Mechanisms for OTA updates in CAVs. In 2024 IEEE/ACM 17th International Conference on Utility and Cloud Computing (UCC) (pp. 423-428). IEEE.
16. Gui, H., Wang, B., Lu, Y., & Fu, Y. (2025). Computational Modeling-Based Estimation of Residual Stress and Fatigue Life of Medical Welded Structures.
17. Levée, M. (2023). Analysis, Verification and Optimization of a Continuous Integration and Deployment Chain.
18. Ribeiro, J. E. F., Silva, J. G., & Aguiar, A. (2025). Scrum4DO178C: an Agile Process to Enhance Aerospace Software Development for DO-178C Compliance—a Case Study at Criticality Level A. *IEEE Access*.
19. Gu, J., Narayanan, V., Wang, G., Luo, D., Jain, H., Lu, K., ... & Yao, L. (2020, November). Inverse design tool for asymmetrical self-rising surfaces with color texture. In Proceedings of the 5th Annual ACM Symposium on Computational Fabrication (pp. 1-12).
20. Zeller, M. (2021, August). Towards continuous safety assessment in context of devops. In International Conference on Computer Safety, Reliability, and Security (pp. 145-157). Cham: Springer International Publishing.
21. Liu, S., Feng, H., & Liu, X. (2025). A Study on the Mechanism of Generative Design Tools' Impact on Visual Language Reconstruction: An Interactive Analysis of Semantic Mapping and User Cognition. *Authorea Preprints*.
22. Warraich, E., Imran, A., Zulfiqar, A., Vargaftik, S., Fahmy, S., & Shahbaz, M. (2025). Reimagining RDMA Through the Lens of ML. *IEEE Computer Architecture Letters*.
23. Mohanty, A. K. (2025). Resource Management and Performance Optimization in Constraint Network Systems. *Sustainable Resource Management in Next-Generation Computational Constrained Networks*, 269-314.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.