Security-Aware Route Selection in DTNs: Towards Optimal Key Provisioning

Ramiro Detke*†, Pablo G. Madoery*†, Renato Cherini*, Juan A. Fraire*†‡, Jorge M. Finochietto*†
*Facultad de Ciencias Exactas, Físicas y Naturales, Universidad Nacional de Córdoba, Córdoba, Argentina
†Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET), Córdoba, Argentina
‡Inria, INSA Lyon, CITI, UR3720, 69621 Villeurbanne, France.
{ramirof.detke, pablo.madoery, renato.cherini, juanfraire, jorge.finochietto}@unc.edu.ar

***Abstract*—Bundle Protocol Security (BPSec) provides integrity and confidentiality for Delay-Tolerant Networks (DTNs) but deliberately leaves key management to the operator. In space networks, where intermittent contacts preclude interactive key exchange, the adopted security scheme—End-to-End, Hop-by-Hop, Edge-by-Edge, or Edge-to-Edge—determines which symmetric keys must be pre-provisioned for each route to remain usable. This paper introduces a security-aware route selection framework that couples routing feasibility with cryptographic provisioning: routes computed via Contact Graph Routing are translated into scheme-induced key scopes, and a mixed-integer linear program minimizes the active key scopes required to reach a target connectivity ratio. Results on a four-plane orbital scenario quantify the trade-off between key provisioning, cryptographic processing, and data-at-rest exposure: End-to-End minimizes exposure and processing at maximal key cost, Hop-by-Hop exhibits the opposite profile, while the edge-based schemes provide practical intermediate operating points, turning BPSec policy selection into a quantitative decision.**

***Index Terms*—Delay-Tolerant Networks, Bundle Protocol Security, key management, contact graph routing, optimization**

I. Introduction

Space communications are entering a new phase of expansion. NASA's Artemis campaign and its *Moon-to-Mars architecture* [1], together with interoperable lunar infrastructures such as NASA's LunaNet [2] and ESA's Moonlight initiative [3], are driving renewed interest in cislunar communication and navigation. In parallel, Mars relay networks [4] and the broader effort to internetwork space systems across the Solar System [5] extend this connectivity beyond Earth orbit. These architectures share two traits. First, their links are challenged by long propagation delays and intermittent, scheduled contacts that preclude continuous end-to-end connectivity. Second, they are operated by multiple independent space agencies, so the network has no single governing authority and is partitioned into distinct administrative and security domains that must interoperate. Together, these traits make secure connectivity difficult: protecting data across domain boundaries cannot rely on centralized key distribution, and must instead be treated as a distributed problem.

These conditions define the operating regime of Delay-Tolerant Networks (DTNs), the networking paradigm adopted for space information networks where conventional end-to-end routing protocols fail. To cope with the absence of continuous connectivity, the Bundle Protocol (BPv7) implements a store-carry-and-forward architecture where bundles are stored in a node's local buffer until the next routing link opens [6]. Since irregular contact windows often leave bundles stagnant for extended periods, DTN security frameworks must protect data in two distinct states: data-in-transit against wireless interception and data-at-rest in intermediate storage buffers against unauthorized access. Bundle Protocol Security (BPSec) natively provides this dual protection by embedding cryptographic metadata directly into the bundle structure [7].

The operational constraints of DTNs—massive round-trip times, link asymmetries, and the lack of real-time certificate revocation checks—render traditional public-key infrastructures unfeasible [8], so BPSec deployments must rely on symmetric cryptography [9] and shift the operational burden onto key management, a long-standing open problem in DTNs [10] that has motivated key distribution architectures ranging from the schedule-aware DTKA [11] to recent BPSec-compatible key management frameworks [12]. These efforts address the mechanics of establishing and revoking keys; *which* keys must be provisioned to sustain network connectivity remains an open, scheme-dependent question. The resulting design space is shaped by a fundamental tension: schemes that preserve end-to-end protection continuity eliminate data-at-rest exposure at intermediate nodes but inflate the symmetric key space, whereas segmented multi-domain architectures [13], [14] dampen key bloat through key reuse within and across administrative domains, at the cost of cryptographic translation—and thus transient cleartext exposure—at domain-edge gateways.

To navigate this trade-off frontier, this paper introduces a security-aware route selection framework that couples routing feasibility with cryptographic provisioning. Rather than optimizing routing performance, the framework constructs a catalog of time-feasible candidate routes, identifies their domain transitions, and derives the cryptographic requirements induced by the selected security scheme. These requirements are fed into a Mixed-Integer Linear Program (MILP) that, given a target connectivity ratio γ, minimizes the active key scopes needed to sustain secure communication. Routing therefore defines the feasible communication alternatives, while the optimization focuses on security provisioning rather than delay, throughput, or other data-transmission metrics.

The remainder of the paper is organized as follows. Sec-

tion II summarizes the considered security schemes and the resulting trade-off. Section III derives the route-level key dependencies produced by the workflow. Section IV formulates the minimum-key connectivity MILP. Section V analyzes the results for a four-plane orbital scenario. Finally, Section VI concludes the paper.

II. Problem Statement

To formalize the tension between topological routing availability and cryptographic vulnerability, we must first analyze how cryptographic keys scale and how data is exposed across different architectural configurations. Rather than re-deriving the four security schemes—which are defined and evaluated extensively in [13], [14]—this section provides a consolidated overview of them: End-to-End (E2E), Hop-by-Hop (HbH), Edge-by-Edge (EbE), and Edge-to-Edge (EtE). Their core trade-offs regarding key management overhead and data vulnerability are summarized in Table I.

As detailed in Table I, a fundamental paradox shapes DTN security design: architectures that neutralize data-at-rest exposure (E2E, EtE) suffer from severe key distribution bottlenecks and the expansion of symmetric key spaces as discussed in [14]. Conversely, approaches that constrain key management overhead through localized associations or domain boundaries (HbH, EbE) introduce risks of cleartext data in storage buffers at intermediate relays or edge gateways. This interplay establishes the core trade-off frontier that motivates our optimization work.

A. Route-Level Security Realization

In practice, these security schemes are not applied over isolated node pairs or abstract domain relationships alone. They are realized over time-feasible end-to-end routes that traverse a sequence of contacts, intra-domain segments, and inter-domain gateway transitions. As a result, the cryptographic implications of a given protection scheme can only be understood at the route level.

For each candidate route, the selected security scheme induces a specific set of cryptographic requirements. In Edge-by-Edge protection, key activation is tied to the protected segments that terminate and restart at domain boundaries. In Edge-to-Edge protection, routes require long-span security associations that bypass intermediate domains, but demand broad cross-domain key availability. Therefore, different routes connecting the same source-destination pair may remain topologically feasible while differing significantly in their cryptographic cost in terms of the number of keys required.

This observation turns secure DTN design into a coupled routing-security problem. The relevant question is no longer only whether a scheme is conceptually stronger or cheaper, but which routes remain usable under that scheme and how many active keys must be provisioned to sustain them. Hence, the network's effective connectivity depends jointly on route structure, domain segmentation, and the cryptographic associations required by each candidate path.

B. The Optimization Frontier

These four paradigms, once instantiated over concrete multi-domain routes, demonstrate that secure routing in DTNs cannot be treated as a static scheme-selection problem alone. Instead, route usability depends jointly on temporal feasibility, domain crossings, and the key scopes required by the selected protection scheme.

This observation motivates the optimization problem studied in this paper: given a target connectivity ratio, determine the minimum number of active key scopes required to sustain secure communication across the network. Addressing this problem requires a formal mechanism to evaluate candidate routes along with their scheme-dependent cryptographic requirements, which directly motivates the workflow and MILP formulation developed in the following sections.

III. Security-Aware Route Selection Workflow

To address security-aware route selection, Fig. 1 summarizes a workflow that transforms the DTN scenario into an optimization-suitable representation. Starting from the contact plan, network topology, and selected security scheme, the workflow computes time-feasible end-to-end routes, annotates their inter-domain structure, and derives the key scopes required to realize each route. The result is an optimization-ready route-key dependency representation that serves as the direct input to the model presented next. The implementation of this workflow and the analysis presented on this paper is available on our public repository [15].

A. From Contact Opportunities to Annotated Routes

At this stage, the workflow remains independent of the selected security scheme and focuses exclusively on identifying candidate end-to-end routes that are feasible from a temporal and topological perspective. For illustration, we consider an orbital-network scenario composed of four polar planes, shown in Fig. 2, where each plane is treated as an independent administrative domain. Since intra-plane connectivity remains available while inter-plane communication is restricted to scheduled contacts, end-to-end feasibility depends jointly on topology and timing. The routing stage therefore computes

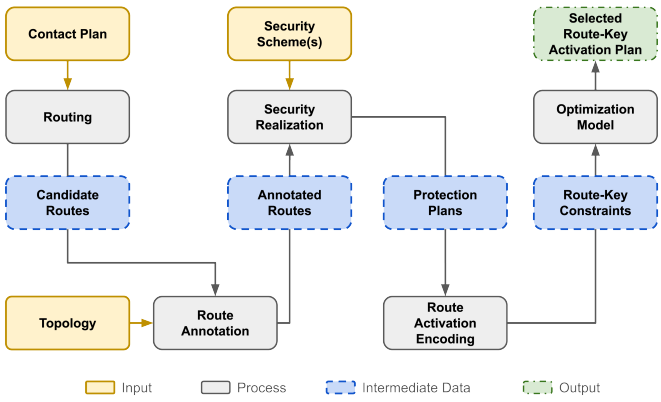

Fig. 1. Security-aware route selection workflow.

TABLE I
COMPARISON OF FOUNDATIONAL DTN SECURITY SCHEMES

Scheme	Protection Span	Data-at-Rest Exposure	Key Scaling
End-to-End (E2E)	From original source endpoint to final destination node.	**Zero**: Payloads remain protected at all intermediate transit relays.	Quadratic $\mathcal{O}(N^2M^2)$ for N nodes per M domains; requires unfeasible global key distribution.
Hop-by-Hop (HbH)	Individual, adjacent topological links.	**High**: Bundles decrypted into cleartext at every intermediate relay buffer.	Linear $\mathcal{O}(NM)$ with minimal topological links; minimizes individual node overhead.
Edge-by-Edge (EbE)	Internal administrative domains and domain boundaries.	**Medium**: Exposed only within edge gateway buffers during cryptographic translation.	Quadratic on nodes per domain $\mathcal{O}(N^2)$ but linear per domains $\mathcal{O}(M)$.
Edge-to-Edge (EtE)	Perimeter boundaries of different domains.	**Low**: Payloads remain protected across intermediate transit domains and gateways.	Quadratic on both nodes per domain $\mathcal{O}(N^2)$ and per domains $\mathcal{O}(M^2)$.

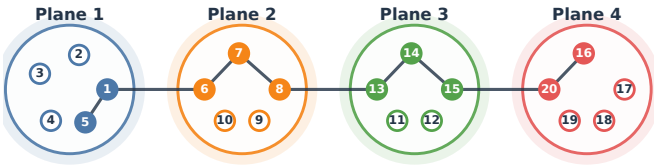

Fig. 2. Four-plane polar layout (route $(5, 16)$).

candidate routes for each source-destination pair under these constraints via Contact Graph Routing (CGR) [16], before any scheme-specific security realization is introduced.

These candidate routes are then annotated to make explicit the traversed domains, gateway roles, and inter-domain crossings associated with each route. For the pair $(5, 16)$, the route $5 \to 1 \to 6 \to 7 \to 8 \to 13 \to 14 \to 15 \to 20 \to 16$ is therefore interpreted not only as a feasible node path but also as an inter-domain progression defined by the gateway transitions $(1, 6)$, $(8, 13)$, and $(15, 20)$. This annotated view preserves route identity while exposing the structural information required for the subsequent cryptographic realization stage.

B. Security Realization and Optimization-Ready Encoding

Once routes have been annotated, the workflow enters the stage that is central to this work: the security realization of each route under the selected security scheme. As anticipated in Section II, the same annotated route may induce substantially different key-scope requirements depending on the protection span and key-scope type associated with the scheme. The problem is therefore no longer only whether a route is temporally feasible, but how it must be realized cryptographically and what burden that realization imposes on the network.

Table II shows how the distinctions summarized in Table I materialize over a fixed annotated route. Following the key-scoping discussion in [14], we distinguish three scope types in this realization: node-to-node scopes, node-to-group scopes used for intra-network access, and group-to-group scopes used for inter-network transit. Although the route itself remains unchanged, its protection plan varies substantially across

TABLE II
SECURITY REALIZATION OF ROUTE $(5, 16)$.

Scheme	Scope Pattern	Required Scopes
HbH	9 successive node-to-node scopes, one per forwarding hop.	9
E2E	1 end-to-end node-to-node scope between source and destination.	1
EbE	3 node-to-group intra-network scopes, 3 group-to-group inter-network scopes, and 1 final node-to-node scope.	7
EtE	1 node-to-group source access scope, 1 long-span group-to-group transit scope, and 1 final node-to-node scope.	3

schemes. Hop-by-Hop binds protection to each forwarding step and therefore yields only node-to-node scopes, whereas End-to-End collapses the same route to a single node-to-node relationship. Edge-by-Edge and Edge-to-Edge occupy the intermediate region identified in Section II, combining mainly node-to-group and group-to-group scopes in different ways, and when the source and destination of a pair belong to the same administrative domain, the route is realized assuming node-to-node key scope, since no inter-domain protection transition is required; in this sense, security realization translates the trade-off among protection continuity, gateway exposure, and key scaling into route-specific protection plans.

Applying this process to the full route catalog yields a route-key dependency representation that records, for each route, the exact node-to-node, node-to-group, and group-to-group scopes required for that route to remain usable under the selected security scheme. For the four-plane scenario, this optimization-ready encoding already reflects the catalog-level trade-off introduced in Section II: Hop-by-Hop minimizes the global scope space ($|K| = 62$) but maximizes the scopes each route requires (6.05 on average); End-to-End exhibits the opposite behavior ($|K| = 380$ and 1, respectively); and the two edge-based schemes fall in between ($|K| = 106$ and 112, with 3.38 and 2.32 scopes per route) by combining intra-network access scopes with inter-network transit scopes. These scheme-induced route-key structures constitute the direct input to the model introduced in Section IV and motivate the more

integrated trade-off analysis developed in Section V.

IV. Minimum-Key Connectivity Model

A. Model Inputs

The optimization stage operates on the structured output produced by the secure path selection workflow described in Section III. At this point, the problem is no longer expressed directly over raw contacts or isolated topological links. Instead, the workflow has already transformed the underlying time-dependent DTN topology into a finite set of candidate end-to-end routes, each one annotated with the exact cryptographic requirements induced by the selected security scheme.

Let D denote the set of ordered source-destination pairs for which secure communication is evaluated. For each pair $(s, d) \in D$, the routing stage computes a finite subset of time-feasible candidate routes, denoted by P_{sd}. The union of all such route sets defines the global route catalog $P = \bigcup_{(s,d)\in D} P_{sd}$. Each route $p \in P$ represents a complete forwarding alternative across the contact graph and is characterized not only by its node sequence and temporal feasibility, but also by its domain traversal structure, including intra-domain segments and inter-domain gateway crossings.

Given a fixed security model, the security planning stage derives for every candidate route $p \in P$ the set of cryptographic key scopes required to protect that route according to the selected scheme (E2E, HbH, EbE or EtE). Let K denote the global set of distinct key scopes that may be activated in the network, and let $K(p) \subseteq K$ denote the subset of key scopes required by route p. This route-to-key dependency mapping constitutes the central input to the optimization problem, since it captures how cryptographic provisioning constraints restrict the simultaneous activation of secure routes.

Finally, let $\gamma \in [0, 1]$ denote the target connectivity ratio imposed by the network designer. This parameter specifies the minimum fraction of ordered node pairs in D that must remain securely connected. Therefore, the optimization model does not seek to activate all feasible secure routes indiscriminately. Instead, it operates over the route catalog P and the associated key-scope set K to determine the smallest cryptographic footprint capable of sustaining the required connectivity objective. It is worth mentioning that complete connectivity ($\gamma = 1$) is not typically required in real scenarios as traffic can be well-known in advance, and hence a selection of only the required routes could be targeted.

B. Decision Variables

Based on the route catalog and the associated cryptographic requirements introduced above, the optimization model decides which secure routes are activated, which key scopes must be provisioned, and which source-destination pairs are effectively covered. To represent these choices, the formulation uses three families of binary decision variables. For each candidate route $p \in P$, we define a binary route activation variable

$$y_p \in \{0, 1\},$$

where $y_p = 1$ indicates that the route p is selected as part of the secure connectivity configuration, and $y_p = 0$ otherwise. For each key scope $k \in K$, we define a binary key activation variable

$$z_k \in \{0, 1\},$$

where $z_k = 1$ indicates that the corresponding cryptographic key scope is activated and made available to support secure forwarding operations, while $z_k = 0$ means that such scope is not provisioned. These variables capture the footprint of the selected configuration and constitute the quantity that the model minimizes. Finally, for each ordered pair $(s, d) \in D$, we define a binary connectivity variable

$$x_{sd} \in \{0, 1\},$$

where $x_{sd} = 1$ indicates that the pair is securely connected, meaning at least one candidate route from P_{sd} is activated and all required key scopes are available. Conversely, $x_{sd} = 0$ denotes that no secure route is sustained for that pair under the selected key allocation.

Together, these variables provide a compact representation of the coupled routing-security decision process. Route variables express the operational forwarding choices, key variables represent the cryptographic resources committed by the network, and pair variables translate those decisions into an aggregate connectivity outcome that can be constrained by the target ratio γ.

C. Objective Function

The model formalizes the central problem of this paper: given a target connectivity ratio, it minimizes the total number of active key scopes required to sustain it, expressed as

$$\min \sum_{k\in K} z_k.$$

D. Constraints

First, the formulation must enforce the target connectivity level. Given the target connectivity ratio γ introduced above, the total number of covered pairs must be at least

$$\sum_{(s,d)\in D} x_{sd} \geq \lceil \gamma |D| \rceil .$$

This constraint guarantees that the selected secure configuration satisfies the desired global connectivity objective.

Second, pair coverage must be consistent with route activation. A pair can only be considered securely connected if at least one of its candidate routes is activated, and any activated route must contribute only to the pair it belongs to. This relation is expressed through

$$y_p \leq x_{sd}, \quad \forall (s, d) \in D, \forall p \in P_{sd},$$

and

$$x_{sd} \leq \sum_{p\in P_{sd}} y_p, \qquad \forall (s, d) \in D.$$

Together, these constraints ensure that pairwise connectivity is induced by the activation of feasible candidate routes.

Finally, route activation must remain cryptographically feasible. Each candidate route p depends on a specific subset of key scopes $K(p)$ determined by the selected security scheme. Therefore, a route may only be activated if all its required key scopes are simultaneously provisioned:

$$\sum_{k \in K(p)} z_k \geq |K(p)|\, y_p, \qquad \forall p \in P.$$

This constraint is the core coupling mechanism between the routing and cryptographic layers, since it ensures that secure connectivity is counted only when the corresponding key is available.

V. Results and Discussion

The model of Section IV is evaluated over the four-plane polar scenario of Fig. 2: 20 nodes evenly distributed over four orbital planes, each treated as an independent administrative domain, with a contact plan of 64 contacts spanning a 2700 s horizon. The routing stage yields a catalog of $|P| = 1134$ candidate routes covering all $|D| = 380$ ordered pairs, obtained by computing up to three time-feasible candidate routes for each ordered pair, with scheme-specific key-scope spaces of $|K| = 62$ for Hop-by-Hop, 380 for End-to-End, 106 for Edge-by-Edge, and 112 for Edge-to-Edge. For each security scheme, the minimum key connectivity model is solved with Gurobi while sweeping the target connectivity from 0% to 100% in 5% increments. The resulting secure configurations are assessed using three complementary metrics: the required key share, defined as the fraction of the scheme-specific key-scope space that the solution activates; the data-at-rest exposure, computed by averaging, over the selected routes, the fraction of intermediate nodes that execute security operations with respect to all intermediate nodes in the full route, excluding source and destination in both cases; and the security operation density, defined as the total number of cryptographic operations induced by the selected routes, divided by their total number of hops. In segmented schemes, protection is terminated and re-instantiated precisely at the points that contribute to both exposure and processing, so the latter remains a useful operational indicator of the same underlying burden.

Figure 3 reports the fraction of available key scopes that must be activated to sustain the required connectivity, which directly reflects the objective minimized by the model. Figure 4 complements that view by exposing how each scheme translates the connectivity target into intermediate data-at-rest exposure. Figure 5 integrates both dimensions by placing the minimum active key scopes $K^*(\gamma)$ against the resulting data-at-rest exposure, with text annotations indicating the associated security operation density. Together, these views reveal how the four schemes distribute the security cost between key provisioning, intermediate exposure, and repeated cryptographic processing.

The two extreme schemes occupy opposite ends of this trade-off. End-to-End consistently yields the lowest data-at-rest exposure in Fig. 4, since protection remains largely

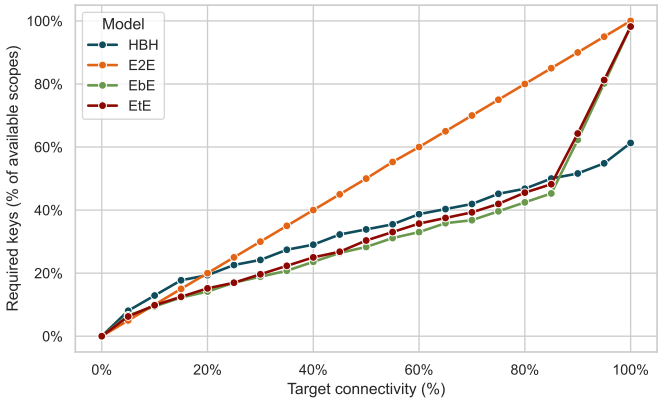

Fig. 3. Required key share versus target connectivity for the four-plane scenario.

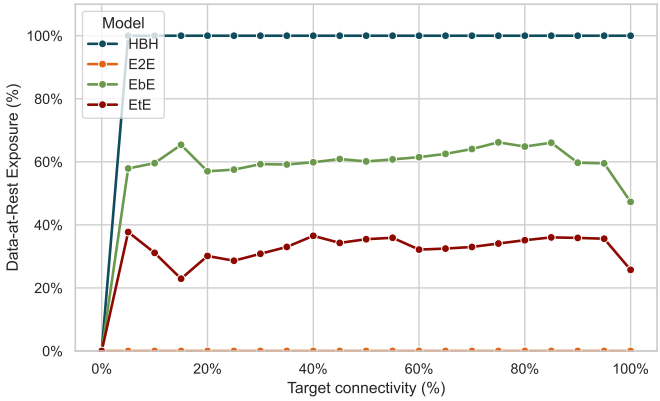

Fig. 4. Data-at-rest exposure versus target connectivity for the four-plane scenario.

continuous from source to destination and avoids repeated intermediate cryptographic processing. However, this benefit comes at the cost of the largest key-distribution burden, with the required key share in Fig. 3 increasing steadily and approaching full activation (380 scopes) as complete coverage is enforced. Hop-by-Hop shows the opposite behavior: its required key share remains comparatively low over much of the connectivity range, but its exposure rapidly reaches its maximum level because protection is terminated and re-instantiated at every forwarding step. In Fig. 5, this separation leaves the lower-left region of the plane empty: the safer configurations lie further to the right in terms of active key scopes, while the configurations with fewer active key scopes remain higher in exposure.

Edge-based schemes define the most relevant intermediate region. In Fig. 3, both Edge-by-Edge and Edge-to-Edge require a smaller key share than End-to-End over most of the connectivity range; however, as full coverage is approached, their curves become nearly linear, since once the reusable inter-domain scopes have been activated, further coverage increasingly depends on the specific node-to-node scopes, which are not reused across different source-destination pairs. In

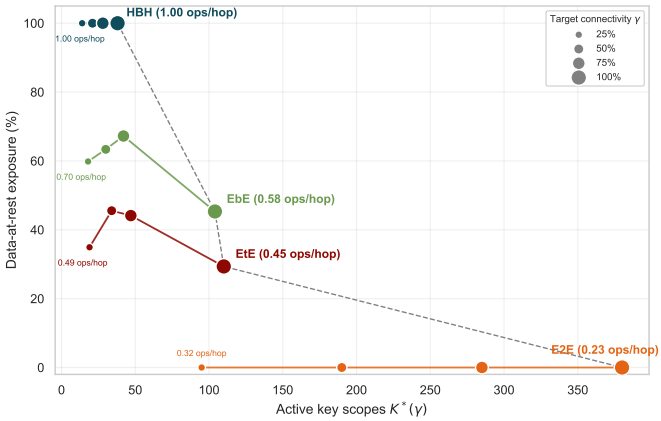

Fig. 5. Integrated trade-off plane for the four-plane scenario: minimum active key scopes $K^*(\gamma)$ versus data-at-rest exposure under each security scheme. Marker size denotes the target connectivity γ, and text annotations indicate the associated security operation density.

Fig. 4, they also remain well below Hop-by-Hop in exposure. Between them, Edge-to-Edge consistently reduces exposure relative to Edge-by-Edge, which is consistent with its longer protection span across inter-domain portions of the route and its lower need for cryptographic re-instantiation at successive boundaries. However, this operational advantage comes with a slightly larger cost of key-distribution. At full coverage, for example, Edge-to-Edge requires only a marginally larger active key set than Edge-by-Edge (110 versus 104 scopes), while remaining at a substantially lower exposure level.

These results show that the main design trade-off is not only how many key scopes must be activated, but also how much data-at-rest exposure the resulting secure configuration imposes on the network. The security operation density, reported in the annotations of Fig. 5, follows the same qualitative order, since every additional protection termination increases computation and creates another intermediate exposure point. End-to-End concentrates the cost on key distribution while preserving protection continuity; Hop-by-Hop concentrates it on repeated encryption and decryption with the highest exposure level; and the edge-based schemes provide intermediate compromises between the two burdens. In this sense, the value of the optimization model lies not only in minimizing active scopes for a given connectivity target but also in making explicit which security scheme best matches the network's dominant operational constraint.

VI. Conclusions

This paper cast secure DTN connectivity as a coupled routing–security problem: since BPSec leaves key management to the operator, the adopted security scheme determines the key scopes each route requires, and thus how many keys must be pre-provisioned over intermittent contacts. A workflow deriving route-level key dependencies from CGR route catalogs, combined with a MILP minimizing the active key scopes for a target connectivity, exposed the resulting trade-off frontier on a four-plane orbital scenario: End-to-End reaches full coverage only by activating its entire key-scope space, while minimizing cryptographic processing and data-at-rest exposure. Hop-by-Hop reaches the opposite operating point, requiring the smallest active key set at the cost of the highest intermediate exposure and the greatest security operation burden. The edge-based schemes occupy the practical middle ground—with Edge-to-Edge trading a slightly larger key share for lower exposure and fewer security operations than Edge-by-Edge. The framework quantifies BPSec policy selection in the evaluated four-plane scenario. Its static contact plan and offline formulation, however, do not establish scalability beyond this setting. Future work includes hybrid per-route policies, scheduling keys over contact plans, demand-specific connectivity models, larger multi-domain scenarios such as lunar architectures, and lighter optimization models.

References

[1] NASA, "Moon to Mars Architecture Definition Document," NASA Exploration Systems Development Mission Directorate, Washington, DC, USA, Rep. ESDMD-001, Rev. C, Dec. 2025. [Online]. Available: https://ntrs.nasa.gov/citations/20250010956

[2] D. J. Israel *et al.*, "LunaNet: a flexible and extensible lunar exploration communications and navigation infrastructure," in *Proc. IEEE Aerosp. Conf.*, Big Sky, MT, USA, Mar. 2020, pp. 1–14.

[3] European Space Agency, "Moonlight: connecting Earth with the Moon," ESA. Accessed: Jun. 2026. [Online]. Available: https://connectivity.esa.int/moonlight

[4] P. Betriu, M. Soria, J. L. Gutiérrez, M. Llopis, and A. Barlabé, "An assessment of different relay network topologies to improve Earth–Mars communications," *Acta Astronaut.*, vol. 206, pp. 72–88, 2023.

[5] J. L. Torgerson, V. Cerf, S. U. DeBaun, and L. C. Suzuki, "Space system internetworking: The foundational role of delay and disruption-tolerant networking," *IEEE J. Sel. Areas Commun.*, vol. 42, no. 5, pp. 1359–1370, May 2024.

[6] S. Burleigh, K. Fall, and E. Birrane, "Bundle Protocol Version 7," IETF, RFC 9171, Jan. 2022.

[7] E. Birrane and K. McKeever, "Bundle Protocol Security (BPSec)," IETF, RFC 9172, Jan. 2022.

[8] W. D. Ivancic, "Security analysis of DTN architecture and Bundle Protocol Specification for space-based networks," in *Proc. 2010 IEEE Aerosp. Conf.*, Big Sky, MT, USA, 2010, pp. 1–12.

[9] E. Birrane, A. White, and S. Heiner, "Default Security Contexts for Bundle Protocol Security (BPSec)," IETF, RFC 9173, Jan. 2022.

[10] S. A. Menesidou, V. Katos, and G. Kambourakis, "Cryptographic key management in delay tolerant networks: A survey," *Future Internet*, vol. 9, no. 3, art. no. 26, 2017.

[11] S. C. Burleigh, D. Horres, K. Viswanathan, M. W. Benson, and F. Templin, "Architecture for delay-tolerant key administration," IETF, Internet-Draft draft-burleigh-dtnwg-dtka-02, Aug. 2018.

[12] F. Fuchs, F. Walter, and F. Tschorsch, "BERMUDA: A BPSec-compatible key management scheme for DTNs," in *Proc. IFIP Int. Conf. ICT Syst. Secur. Privacy Protect. (SEC)*, IFIP AICT, vol. 746. Cham, Switzerland: Springer, 2025.

[13] P. G. Madoery, R. Cherini, A. Cammarano, J. J. Grosso, and J. M. Finochietto, "Analysis of Bundle Protocol security policies for safeguarding space missions against threats," in *Proc. 2023 IEEE Int. Conf. Wireless Space Extreme Environ. (WiSEE)*, Aveiro, Portugal, 2023, pp. 65–70.

[14] P. G. Madoery, R. Cherini, A. Cammarano, J. Grosso, and J. M. Finochietto, "Bundle Protocol security models and policies for safeguarding space information networks," *IEEE J. Radio Freq. Identif.*, vol. 8, pp. 547–558, 2024.

[15] R. Detke, P. G. Madoery, R. Cherini, J. A. Fraire, and J. M. Finochietto, "DTN BPSec Pipeline," GitHub repository, branch `wisee2026`. Accessed: Jun. 2026. [Online]. Available: https://github.com/ramirofd/dtn-bpsec-pipeline/tree/wisee2026

[16] J. A. Fraire, O. De Jonckère, and S. C. Burleigh, "Routing in the space internet: A contact graph routing tutorial," *J. Netw. Comput. Appl.*, vol. 174, art. no. 102884, 2021.