



**HAL**  
open science

## On Selecting a Routing Protocol for Nanosatellite Swarm Networks

Louis Basset, Riadh Dhaou, Emmanuel Lochin, Pontet Bernard, David Pradas,  
Bastien Tauran

### ► To cite this version:

Louis Basset, Riadh Dhaou, Emmanuel Lochin, Pontet Bernard, David Pradas, et al.. On Selecting a Routing Protocol for Nanosatellite Swarm Networks. 101st IEEE Vehicular Technology Conference, IEEE, Jun 2025, Oslo, Norway. <hal-04999739>

**HAL Id: hal-04999739**

**<https://hal.science/hal-04999739v1>**

Submitted on 21 Mar 2025

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

# On Selecting a Routing Protocol for Nanosatellite Swarm Networks

Louis BASSET\*, Riadh DHAOU\*, Emmanuel LOCHIN†, Bernard PONTET‡, David PRADAS‡, Bastien TAURAN‡

\*IRIT, Université de Toulouse, CNRS, Toulouse INP, Toulouse, France. Email: {louis.basset, riadh.dhaou}@toulouse-inp.fr

†Fédération ENAC ISAE-SUPAERO ONERA, Université de Toulouse, Toulouse, France. Email: emmanuel.lochin@enac.fr

‡VIVERIS Technologies, Toulouse, France. Email: {bastien.tauran, david.pradas}@viveris.fr

§CNES, Toulouse, France. Email: bernard.pontet@cnes.fr

**Abstract**—Routing in nanosatellites swarms presents distinct challenges, including variable node availability, constrained bandwidth, and dynamic topology. Strategies like delay-tolerant networking (DTN) can be advantageous, as they adapt to intermittent connectivity by storing and forwarding data when connections are established. Moreover, geographic routing protocols that exploit satellite positions can improve efficiency, while machine learning approaches may optimize routing decisions based on changing network conditions. What about hybrid approaches that may combine some of these methods? Basically, the crucial question is where to begin.

The primary challenge for nanosatellites network designers is to determine which routing strategies to test prior to deployment. Given the vast number of existing routing protocols, testing all of them is not possible. This problem motivates the present study, which share the authors' experiences on selecting the most suitable routing algorithms for a given nanosatellites swarm. In particular, the study reports how the use of graph theory metrics helps in restricting the set of routing algorithms to be considered for network characterization and protocol selection.

**Index Terms**—Satellite networks, nanosatellites swarm, routing protocols.

## I. INTRODUCTION

The rapid advancement of satellite technology has opened new possibilities for satellite swarm networks, which offer cost-effective solutions for various space missions. While traditional satellite constellations operate in fixed orbits, nanosatellite swarms provide flexible configurations that are adaptable to mission-specific requirements. However, these swarms face significant challenges related to energy limitations, data rate constraints, and network connectivity.

The proposed method is built upon our previous research on network characterization and fair division of nanosatellite swarms, particularly in contexts requiring high resilience and adaptability. In this previous work [1], [2], we explored the unique challenges of communication and network management in satellite swarms, focusing on efficient bandwidth allocation and fair network resource division. These studies highlighted key issues such as congestion control and delay tolerance, which are crucial for satellites to maintain stable inter-satellite links (ISLs) despite frequent topology changes.

The authors contributed equally to the study presented in this article and thus are listed in alphabetical order.

In contrast to deep-space or lunar orbits, Low Earth Orbit (LEO) satellite swarms experience higher mobility and faster topological changes. Protocols such as Delay-Tolerant Networking (DTN) [3] have been adapted to maintain connectivity under such conditions. However, due to the need for low latency and high data throughput, hybrid approaches that combine fixed and dynamic routing mechanisms are increasingly being explored. Our previous findings on satellite swarm management inform the present study, specifically, by seeking protocols that combine traditional IP-based routing with adaptive routing mechanisms suitable for the LEO environment.

This paper specifically addresses the problem of routing within a swarm of nanosatellites. The goal is to identify routing protocols or at least, a class of routing protocols able to perform over such specific network. At last but not least, this study, drawing on our nanosatellite swarm management findings, contributes to a broader understanding of the adaptation of routing protocols for LEO nanosatellites.

This article is structured as follows: in Section II, we review the existing literature on routing protocols for satellite networks, emphasizing their advantages and limitations for nanosatellite constellations. Section III details the methodology used to select and evaluate routing protocols suitable for nanosatellite swarms, including selection criteria and simulation setup. In Section IV, we analyze and compare the performance of the primary routing protocols, focusing on metrics such as latency, resilience, and throughput. Section V describes the practical implementation of these protocols in simulated scenarios, particularly in relation to update distribution and constant bit rate (CBR) traffic management. Finally, Section VI presents and discusses the simulation results, while Section VII concludes by highlighting the main contributions of our work and future research directions.

## II. RELATED WORK

Routing protocols for satellite networks have raised significant interest in networking research because of their unique challenges, including high latency, variable connectivity, and the need for efficient energy usage. Existing studies primarily classify routing protocols into three categories:

- **protocols from the wired world:** traditional protocols like OSPF [4] have been widely used in stable terrestrial networks. Although these protocols ensure robust path calculation, their static nature limits their efficiency in the dynamic topology of satellite networks. However, they still can be an efficient solution when there is no mobility within the swarms. In other words, when the mobility of the nodes are only controlled by the orbital laws. In particular, recent work by Zhang et al. [5] explores the adaptation of these protocols to satellite constellations by introducing adaptive path recalculation mechanisms;
- **Mobile Ad-hoc Network (MANET) protocols:** protocols such as Ad-Hoc On-Demand Distance Vector (AODV) [6] and Dynamic Source Routing (DSR) [7] are designed for dynamic, decentralized environments. These protocols have been studied for their potential use in low Earth orbit (LEO) satellite networks where node mobility is high. A recent study by Liu et al. [8] proposes a hybrid approach that combines AODV with proactive table management to improve connectivity during frequent topology changes;
- **Delay-Tolerant Network (DTN) protocols:** for satellite networks with intermittent connectivity, DTN protocols such as the Bundle Protocol are essential. Jenkins et al. [9] demonstrated DTN feasibility for space communication in environments with high latency. An extension of DTN, called Deep Space Network (DSN) routing, has also been applied to nanosatellites for deep space missions, as noted in recent works [10].

Recent studies have also explored hybrid protocols that combine elements of fixed and DTN routing to maintain connectivity and reduce packet loss. For example, Chen et al. [11] proposed a hybrid routing scheme that leverages fixed routing paths when possible and DTN-based store-and-forward in cases of temporary link failures. Furthermore, applications of machine learning to optimize routing in satellite networks are emerging, with Wu et al. [12] showing improved routing decisions through predictive modeling based on historical link states.

The field still evolves, with ongoing developments in multipath and cross-layer routing approaches that dynamically adjust to network conditions. These advances are crucial for supporting the high data rates and low-latency demands of modern satellite-based IoT applications [13].

### III. METHODOLOGY

This study employed a structured methodology to evaluate routing protocols suitable for the dynamic and resource-constrained environments of LEO nanosatellite swarms. The proposed approach includes characterization of the network environment, selection criteria for routing protocols, and a custom simulation setup to analyze key performance metrics. Each step is detailed in the following.

#### A. Simulation Setup

The simulations were conducted using a custom codebase, originally developed by Evelyne Akopyan, tailored specifically for nanosatellite swarm networks. The code, available on GitHub [14], provides a flexible simulation environment suited for LEO conditions, allowing precise adjustments to node positions, link capacities, and protocol behaviors. The key parameters set for the simulation environment included the:

- **node density:** a base configuration of 50 nodes was used to assess protocol performance in a moderately dense network, with additional tests scaling node density to observe the impact on connectivity and protocol overhead;
- **transmission range and power:** ISLs were restricted by a maximum communication range, simulating realistic power constraints in nanosatellite deployments. This parameter was critical in observing how different protocols adapt to connection losses and the formation of network partitions;
- **data rate variability:** Simulations were conducted at data rates of 100 *kbit/s*, 256 *kbit/s*, 512 *kbit/s*, 1024 *kbit/s*, and 2048 *kbit/s*, representing a range from low to high transmission loads. This variability allowed for an assessment of protocol efficiency under both low- and high-bandwidth scenarios, and provide various network configuration as the reachability of the swarm depends on the transmission bitrate as shown Fig. 1.

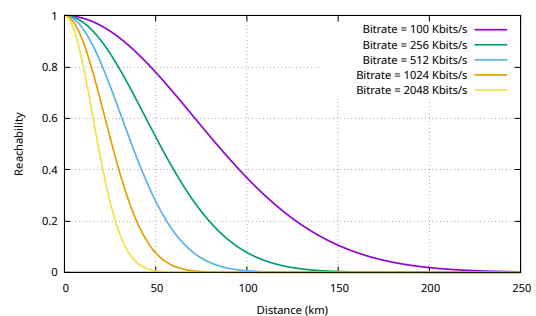


Fig. 1: Reachability of the node of the swarm as a function of their bitrate. Note these link-budget values have been computed by CNES.

#### B. Network Characterization

Then, to perform the characterization of the network as explained in Section III, we selected the following graph-theoretic metrics to select the most adequate routing protocol:

- **average node degree:** this metric represents the average number of direct connections per node and indicates overall network connectivity. Higher node degrees suggest more robust connectivity, which impacts protocol routing stability;
- **shortest path length:** the average shortest path length provides insights into latency and route efficiency, which are critical for real-time data transfer within the swarm;

- **connected components:** monitoring the number of connected components helps to determine network fragmentation and the protocol ability to maintain cohesion under dynamic conditions;
- **network diameter:** the diameter, or maximum path length between two nodes, was measured to gauge overall network resilience. A smaller diameter typically reflects better connectivity and robustness against partitioning.

The combination of these metrics provide a powerful tool allowing to weight up the potential efficiency of a given routing protocol considering the simulation environment chosen. Obviously, both the simulation parameters and the set of graph theory metrics can be extended to increase the compliance of the routing protocol under tests by considering, for instance, specific network services or mission that should handle the swarm.

Data was collected across multiple simulation runs for each protocol and parameter configuration. Each simulation scenario was run a minimum of ten times to ensure result consistency, with average values recorded for each metric. The results were then analyzed to determine how well each protocol met the requirements of low-latency, high-resilience communication within the swarm. Comparative analysis allowed for the identification of protocols most suited to the specific characteristics of LEO nanosatellite swarms. While the simulations provided valuable insights, certain limitations must be noted. The custom codebase, though highly adaptable, may not fully capture all real-world factors, such as radiation-induced link degradation or power constraints in long-duration missions. Future work will expand simulations to incorporate more detailed orbital dynamics and test the scalability of the identified protocols under higher node densities.

#### IV. MEASUREMENTS AND ANALYSIS

We analyse in this section the results obtained while explaining the interest of the metrics evaluated.

##### A. Average Node Degree

The average node degree reflects the overall connectivity of the swarm, representing the average number of direct neighbors each node has in the network. The achievable data rates typically decrease as a function of distance. This is because the signal strength decreases with distance due to path loss. As the signal travels farther, it becomes weaker, leading to a lower SNR at the receiver. A lower SNR means that the system may need to use more robust (but less efficient) modulation and coding schemes, which results in lower data rates. As shown in Figure 2, lower data rates result in higher average degrees (e.g., approximately 36 at 100 *kbit/s*), while higher data rates reduce connectivity, with an average degree dropping to around 5.7 at 2048 *kbit/s*. This decrease indicates that as data rates increase, maintaining robust connections between nodes becomes challenging.

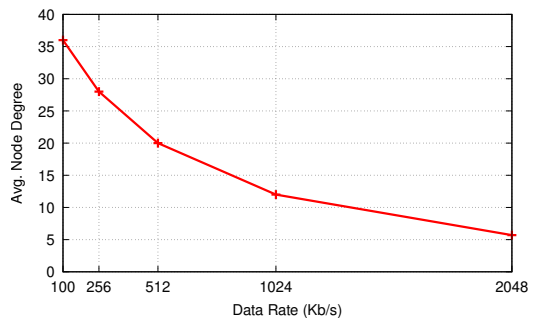


Fig. 2: Average Node Degree vs. Data Rate

##### B. Shortest Path Length

The shortest path metric complements the average node degree by assessing network latency and load. Lower average path lengths correlate with reduced latency, benefiting real-time applications as data traverses fewer hops. Figure 3 illustrates that path lengths range from 1 to 3 hops across various data rates, favoring protocols that minimize hops like link-state protocols. At this stage, these findings suggest that link-state protocol, leveraging shortest-path calculations, is well-suited to reduce latency and optimize the traffic load in satellite swarms, at least, for lower transmission rates.

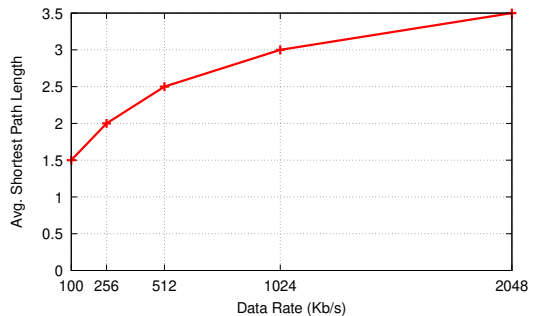


Fig. 3: Shortest Path Length vs. Data Rate

##### C. Number of Connected Components

Connected components provide insight on the network partitioning. As shown in Figure 4, connected components begin to appear at data rates around 512 *kbit/s*, and increase with higher data rates. This indicates temporary network partitioning, with nodes clustering into isolated subgroups. Although these partitions are not permanent at 2048 *kbit/s*, the persistence of fragmentation affects the choice of protocol. Link-state protocols should remain viable due to their capacity to update routing tables quickly, but they start to reach a limit depending on the inter-contact time of the number of connected components.

##### D. Network Resilience

Network resilience refers to the capability of a network to maintain a level of connectivity and performance despite failures or disruptions. The network diameter, defined as the

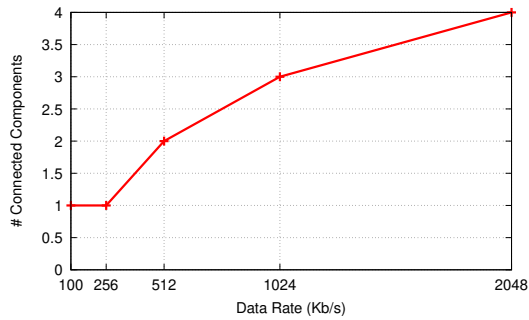


Fig. 4: Connected Components Over Time at Various Data Rates

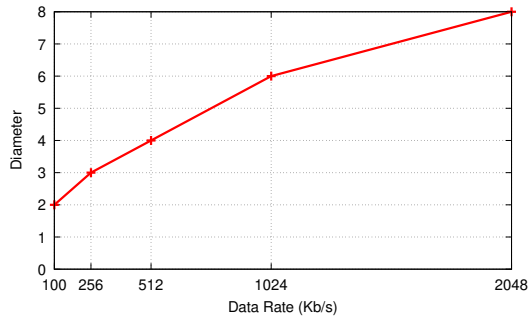


Fig. 5: Network Diameter vs. Data Rate.

longest shortest path between any two nodes, provides a key indicator of the resilience of the network. In resilient networks, the diameter remains relatively low, even under node or link failures, indicating that data can still be efficiently routed across the network.

A smaller diameter generally suggests that nodes are closely connected, allowing for multiple redundant paths. This is a case at lower rate in Figure 5. This redundancy means that if certain links or nodes fail, alternative paths are still available, preventing significant increases in latency or disconnection of segments. In other words, a network with a low diameter can re-route data efficiently and thus, remain robust to failure.

Moreover, as network diameter increases due to disruptions, the network may experience partitioning into isolated clusters (observed for higher rate in Figure 5). A resilient network limits this fragmentation by maintaining connections even under failure, keeping the diameter stable. When a network diameter rises sharply in response to minor failures, this results in a lower resilience, as communication becomes restricted to fewer, less efficient paths, or segments of the network become isolated entirely.

In summary, a low and stable network diameter under varying conditions reflects a network that is resilient, offering strong connectivity and adaptability in the face of disruptions. Such a network can handle failures gracefully, ensuring that data flow remains uninterrupted across the network despite adverse conditions.

To conclude and as shown in Figure 5, a link-state protocol

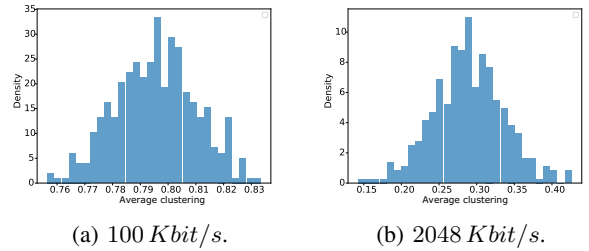


Fig. 6: Average clustering distributions for 100 Kbit/s and 2048 Kbit/s. We observe a higher clustering value for 100 Kbit/s compared to 2048 Kbit/s.

should be able to perform under lower mobility scenarios but may struggle to maintain a single connected component as node mobility increases, particularly at high data rates.

### E. Network clustering

These tests reveal a particular property of the network at low data rates. Actually, we can observe the emergence of small-world graphs at low data rates. This is a notable result, as small-world graphs have specific properties that can significantly influence routing. To be considered as small-world, a graph must satisfy certain specific properties:

- the first was verified in Section IV-B. First of all, we need to check that the average length of the shortest path is logarithmic as a function of the number of nodes. This means that, on average, the number of steps required to get from one node to another increases logarithmically as the number of nodes in the graph increases. Following IV-B, this is indeed the case for data rates lower or equal than 256 kbit/s (we recall that  $\log(50) \approx 1.69$ );
- in addition to this average length, the graph must have a high clustering coefficient. This means that nodes tend to form groups or clusters, where neighboring nodes are strongly interconnected.

Considering both the results of section IV-B and those of Figure 6, we are clearly in the presence of a small-world network for lower data rates. Link-state routing algorithms such as OSPF are well suited to small-world graphs. OSPF uses the Dijkstra algorithm to compute shortest paths. In a small-world graph, these paths are naturally short, which reduces the complexity and computation time required to find the optimal paths. Furthermore, the strong local connectivity provided by the clustering metric improves network resilience. If a link goes down, there are many alternative paths, where a link-state routing protocol should handle well thanks to its ability to quickly recompute paths in the event of a topology change.

## V. DISCUSSION AND CONCLUSION

Following this characterization, the selection of the most adequate routing protocol must be based on its suitability to meet the network Quality of Service (QoS) requirements. The three primary classes of protocols considered are those

previously listed in Section II. If we stick to the definitions provided in this Section II and considering the simulation hypothesis : at a first sight, the metrics obtained suggest that a link-state protocol such as OSPF, should be a good compromise due to its ability to handle changes in topology and maintain a comprehensive network map. The capability of link-state protocols, particularly in managing connected components within a certain threshold, and optimizing shortest-path routing, suggest that they should be suitable for such swarm of nanosatellite networks with intermittent connectivity. Furthermore, the protocol efficiency in handling dynamic topologies makes it well-suited for applications requiring low-latency communications.

However, it should be noted that this routing solution might be only valid for relatively low data rates, which do not seem to be suitable for bulk data transfer services. As data rates increase, enabling the use of more capacity-hungry applications, we approach the DTN routing case, which becomes *de facto* unsuitable for services requiring high data rates. Actually, the metrics also exhibit specific limitations under certain conditions that prevent the use of such class of routing protocol. If we consider the use of OSPF such as a link-state protocol candidate, at high data rates ( $>1024 \text{ kbit/s}$ ), OSPF will have to manage a network splitted into multiple connected components. This behavior arises from the increased overhead of path recalculations in highly dynamic topologies. Latency also increases, exceeding acceptable thresholds for real-time critical applications. On the other hand, we could consider the use of a DTN approach. Although DTN performs well under intermittent connectivity, its higher latency becomes a drawback in relatively stable networks. This limits its applicability in dense configurations of swarms.

While the computed metrics suggest that testing a link-state routing protocol is justified, definitive confirmation requires performance measurements of the routing itself. The primary goal of the method presented is **to prevent arbitrary testing of multiple routing schemes** and to facilitate the selection of an appropriate subset. A persistent challenge is to determine where to start, so we argue that such characterisation would help **to more effectively select the right subsets of routing protocols**. Subsequently, evaluating the performance of the routing protocols themselves is a logical next step.

Another example of the benefit of applying graph theory to study nanosatellite swarms is the insight that no single routing solution fits all scenarios. A promising approach could be to design a hybrid protocol that ensures robust connectivity while minimizing latency during temporary link disruptions, as shown in Table I. However, this hybrid approach necessitates more advanced synchronization mechanisms, which remain an open area for further research.

#### A. Future work and critical review

The simulations driven in this study rely on a simplified, albeit relevant, model that does not fully account for some critical real-world factors. In particular, adding orbital perturbations and energy consumption models to the simulation

TABLE I: Resilience Matrix of Routing Protocols in Satellite Swarms

Protocol	Latency	Connectivity	Energy Efficiency
OSPF	High	Medium	Low
DTN	Medium	High	Medium
Hybrid (OSPF+DTN)	Medium	High	High

would greatly increase the consistency of these results. The simulation does not capture real satellite dynamics, such as atmospheric drag or gravitational variations. A critical factor in nanosatellite networks deals with energy consumption. Although the cost associated by the use of the protocols on such system is negligible compared to the operational cost, the routing traffic might have an impact on the energy consumption that should be investigated. We reserve these points to a future work.

Nevertheless, we believe the findings of this study provide a comprehensive method to evaluate the possible deployment of routing protocols within LEO nanosatellite swarms, specifically in terms of their adaptability, efficiency, and ability to maintain network cohesion under dynamic conditions.

#### ACKNOWLEDGEMENTS

The authors would like to thank Cecile Dechoz, Julien Galizzi and Felipe Alves for numerous discussions on this subject, and CNES for funding support.

#### REFERENCES

- [1] E. Akopyan, R. Dhaou, E. Lochin, B. Pontet, and J. Sombrin, "On the network characterization of nano-satellite swarms," in *IEEE Symposium on Computers and Communications (ISCC)*, pp. 944–949, 2023.
- [2] E. Akopyan, R. Dhaou, E. Lochin, B. Pontet, and J. Sombrin, "Fair network division of nano-satellite swarms," in *IEEE Vehicular Technology Conference (VTC2023-Spring)*, pp. 1–5, 2023.
- [3] K. Fall, "A delay-tolerant network architecture for challenged internets," in *ACM SIGCOMM*, 2003.
- [4] J. Moy, "Ospf version 2." RFC 2328, IETF, April 1998.
- [5] Y. Zhang, H. Li, and S. Wang, "New adaptive ospf mechanism for satellite networks," *Journal of Satellite Communications and Networking*, vol. 35, pp. 50–60, 2021.
- [6] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc on-demand distance vector (aodv) routing." RFC 3561, IETF, July 2003.
- [7] D. Johnson, Y.-C. Hu, and D. Maltz, "The dynamic source routing protocol (dsr) for mobile ad hoc networks for ipv4." RFC 4728, IETF, February 2007.
- [8] M. Liu and J. Xu, "Manet protocol adaptation for low earth orbit satellites," in *IEEE International Conference on Communications (ICC)*, pp. 1200–1207, 2022.
- [9] A. Jenkins, S. Kuzminsky, K. K. Gifford, R. L. Pitts, and K. Nichols, "Delay/disruption-tolerant networking: flight test results from the international space station," in *IEEE Aerospace Conference*, pp. 1–8, 2010.
- [10] K. Gifford and K. Nichols, "Applications of delay/disruption tolerant networking for deep space satellite missions," *Space Communications*, vol. 42, pp. 88–96, 2020.
- [11] J. Chen and L. Guo, "A hybrid routing protocol for leo satellite networks with intermittent connectivity," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 5, pp. 3005–3015, 2021.
- [12] Q. Wu, X. Zhao, and T. Wang, "Machine learning-based routing optimization in satellite networks," *IEEE Transactions on Network and Service Management*, vol. 19, pp. 45–57, 2022.
- [13] S. Lee and D. Park, "Cross-layer routing for iot services in leo satellite constellations," *IEEE Internet of Things Journal*, vol. 10, pp. 1153–1164, 2023.
- [14] E. Akopyan, "Swarmy simulator." [https://github.com/eakopyan/SwarmPy\\_Simulator](https://github.com/eakopyan/SwarmPy_Simulator), 2023. Accessed: 2023-10-27.