2018

Energy-Aware Routing for CubeSat Swarms

Mohamed Atef Madni
University of Wollongong, maarm430@uowmail.edu.au

Raad Raad
University of Wollongong, raad@uow.edu.au

Mohamad Raad
Lebanese International University

Recommended Citation

Madni, Mohamed Atef; Raad, Raad; and Raad, Mohamad, "Energy-Aware Routing for CubeSat Swarms" (2018). *Faculty of Engineering and Information Sciences - Papers: Part B*. 4517.
https://ro.uow.edu.au/eispapers1/4517

Energy-Aware Routing for CubeSat Swarms

Abstract

CubeSats operating in a swarm are characterized by a mix of scheduled intermittent connectivity, high delays, and high failure rates. Each CubeSat is limited in size, usually has low data rates and has a low mass. Consequently, they have limited space for solar panels, and thus limit their available energy. Profitably, CubeSats can function in swarms using inter-CubeSat links as well as ground links. Accordingly, any routing protocols developed for CubeSats must be energy aware. We propose two novel Shortest and Energy Reliable Path (SERP) routing protocols; namely, SERP- Breadth-First Search (SERP-BFS) and SERP-Dijkstra. Both algorithms aim to minimize the overall energy cost and maintain connectivity over time. Both choose shortest paths that have CubeSats energy levels higher than or equal to an energy reliability threshold. We have compared our SERP algorithms with Epidemic algorithm. The results show the outperformance of our proposed algorithms in terms of saving the overall energy cost.

Keywords

energy-aware, routing, swarms, cubesat

Disciplines

Engineering | Science and Technology Studies

Publication Details

M. A. Madni, R. Raad & M. Raad, "Energy-Aware Routing for CubeSat Swarms," International Journal of Digital Information and Wireless Communications, vol. 8, (3) pp. 150-155, 2018.

International Journal of Digital Information and Wireless Communications (IJDIWC) 8(3): 150-155
The Society of Digital Information and Wireless Communications, 2018 ISSN: 2225-658X (Online); ISSN 2412-6551 (Print)

Energy-Aware Routing for CubeSat Swarms

Mohamed Atef Ali Madni[1], *Member, IEEE*, Raad Raad[1] , *Member, IEEE*, Mohamad Raad[2]

[1] School of Electrical, Computer and Telecommunications Engineering, University of Wollongong, NSW, Australia, 2522
[2] Computer and Communication Engineering, Lebanese International University, Beirut, Lebanon

***Abstract*—CubeSats operating in a swarm are characterized by a mix of scheduled intermittent connectivity, high delays, and high failure rates. Each CubeSat is limited in size, usually has low data rates and has a low mass. Consequently, they have limited space for solar panels, and thus limit their available energy. Profitably, CubeSats can function in swarms using inter-CubeSat links as well as ground links. Accordingly, any routing protocols developed for CubeSats must be energy aware. We propose two novel Shortest and Energy Reliable Path (SERP) routing protocols; namely, SERP- Breadth-First Search (SERP-BFS) and SERP-Dijkstra. Both algorithms aim to minimize the overall energy cost and maintain connectivity over time. Both choose shortest paths that have CubeSats energy levels higher than or equal to an energy reliability threshold. We have compared our SERP algorithms with Epidemic algorithm. The results show the outperformance of our proposed algorithms in terms of saving the overall energy cost.**

Keywords—energy reliability; delay tolerant network; space-time graph; picosatellites; cubesat swarms.

I. INTRODUCTION

Compared to conventional satellites, CubeSats are limited in size (1U=10x10x10cm), have low mass (1.3 kg for 1U), have low data rates (9.6 Kbit/s) and have limited power (2 W). The construction of a CubeSat costs \$20-\$200K in comparison to conventional satellites that cost \$0.1-\$2B. Another advantage is that CubeSats can be readily deployed on the Poly Pico-satellite Orbital Deployer (P-POD) [1]. Alternatively, they can be deployed from the International Space Station (ISS) using Japanese Experiment Module Small Satellite Orbital Deployer (J-SSOD) or NanoRacks CubeSat Deployer (NRCSD). Advantageously, CubeSats can operate in a swarm. A notable example is the QB50 project, which aims to realize a swarm of 50 CubeSats [2].

A CubeSat swarm facilitates global measurements and can potentially have higher data rates as compared to a single CubeSat. Thus, wider communication window with ground stations can be provided. Moreover, a CubeSat swarm allows data to be collected from different parts of space at the same time instant. Consequently, a swarm of CubeSats can help monitor the Earth; in particular, conduct atmospheric measurements and support missions related to space weather [3].

CubeSat swarms can be in different formations according to the mission objectives. In [4], thirty-nine multiple CubeSat missions are reviewed. These missions are categorized according to the mission objective, satellites number and type of formation flying. Formation flying of small satellites can be mostly divided into three types; namely, leader-follower, cluster, and constellation. In leader-follower missions, all satellites are deployed on the same orbit, and they are separated by a specific distance and they all follow the leader. A cluster mission is when a group of satellites deployed close to each other on different orbits to cover an appointed area on the Earth. However, the constellation includes a group of satellites disseminated in different orbits to provide full coverage to the Earth [5] as shown in Fig. 1.

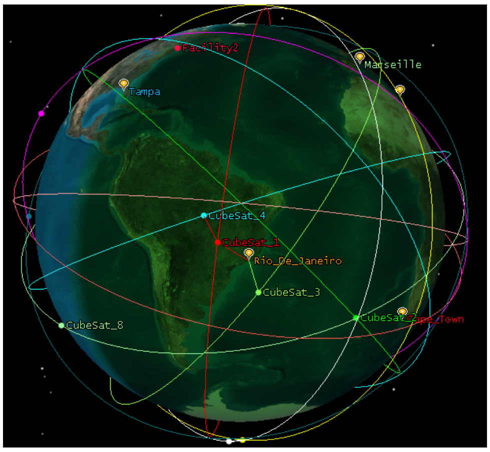

Fig. 1. CubeSat constellation

CubeSat communications are characterized by a mix of scheduled/predictable and intermittent connectivity, delay, limited life time [6] and high failure rates [7]. Consequently, these challenging conditions will cause the network topology to undergo continuous changes. This change leads to a variation in the encounter time between nodes. Encounter time is the time when two or more satellites come in communications range of one another. CubeSat communication, whether with the ground stations or inter-CubeSat communication, is considered as a significant source of power consumption [8]. Therefore, minimising the power budget of communication subsystem by considering routing protocols that include energy as a metric for data routing is very important.

Currently, routing protocols proposed for such environment are forcing on message delivery, throughput and/or delay minimization. Most of them are suggesting some improvements to the well-known flooding based Epidemic routing protocol. Epidemic routing is aiming to increase the possibility of message delivery by injecting many message replicas into the network. However, this poses a major problem of high energy harvesting, especially for networks that include nodes with restricted battery supply such CubeSat swarms. In [9], the authors proposed n-Epidemic routing protocol as an energy efficient solution for Delay Tolerant Networks (DTNs). They restricted the transmission of messages to the number of neighbours of the node. However, they do not consider the energy stored at each node.

This paper considers energy-aware routing problem for a CubeSat swarm with limited energy supply and charging ability. CubeSats only have a small surface area to install solar panels for energy collection. Thus, the communication capability of a CubeSat is bounded by the level of the available energy. The transmission range of each CubeSat is also limited by its inter-CubeSat antenna design [1] and communication power budget. A CubeSat can only forward messages to other CubeSats within its transmission range. Hence, forwarding decision based on energy level for each CubeSat along the path becomes critical in this case.

We introduce a DTN space-time graph as a networking paradigm for energy based routing and forwarding across swarms of low orbiting CubeSats or equivalent picosatellites. The space-time graph is used to identify all possible paths, which may connect any pair of nodes over time. We proposed two energy-aware routing protocols to operate on top of the space-time graph. The aim of our proposed SERP-BFS and SERP-Dijkstra protocols is to minimize the overall energy cost and maintain connectivity over time. Both choose paths that have nodes with energy levels higher than or equal to an energy reliability threshold. This means messages/bundles will only travel on the shortest and energy reliable path. An energy reliable path is the shortest path that only includes CubeSats with energy levels equal to or higher than the energy threshold. In this case, only paths that can satisfy these two requirements will be considered as potential paths. This minimizes the number of overall network potential links (edges) that connect any two CubeSats. Therefore, minimizing the number of overall links will considerably reduce communication costs including antenna transmutation and reception, processing and computation time.

Most of the previous works on inter-CubeSat communications are focused on developing physical layer links; i.e., the antenna design or study the suitability of current radios for use on CubeSats, see [1] and some other work is concentrating on the topology formation [2] [10]. However, as far we know the CubeSat swarms energy minimisation problem using DTN forwarding protocols has not been considered.

The rest of this paper is structured as follows. Section II is a summary of space-time graph and routing in DTN related work. Section III introduces the CubeSat routing problem including the network modelling and our proposed algorithms. Simulation results and discussion are presented in Section IV. Section V concludes the paper.

II. BACKGROUND

A. *Space-Time Graph Model*

Space-time graph model is used to represent network connectivity overtime for networks that are characterized by intermittent connectivity and tolerance for long delay. In [11], Liu et al. proposed an Expected Minimum Delay (EMD) metric and EMD-based routing protocol called Routing in Cyclic MobiSpace (RCM) for networks that are characterised by cyclic and intermittent connectivity. The network is modelled as a probabilistic space-time graph, where each encounter time of a node is anticipated from the historical encounter information or previous awareness about the network with an assumption that it will not change later. In addition to this, [12] proposes Mobility-Based Routing Protocol (MBRP) for routing data in DTNs, where mobility of nodes is characterized by semi-predictable patterns in a finite time period. MBRP creates its space-time graph based on historical mobility patterns information and node encounters. In [13], S. Merugu et al. considered the routing problem in wireless networks that are characterized by predictable mobility. Shortest Paths in Space and Time (SPST) routing algorithm is designed based on space-time graph model to specify the appropriate next hop and upcoming neighbours to minimize the end-to-end delay. On the other hand, in [14], the authors studied the problem of Topology Control (TC) for DTNs with predictable movements, where the network topology shape can be known prior with time evolution. They proposed three greedy-based algorithms that can maintain the connectivity over time, while considerably decreasing the total cost of network topology. The TC problem was defined for predictable DTNs with consideration of time evolution; however, there was an assumption about the reliability of future links to deliver data without any errors or distortion. In reality, such a strong assumption might be acceptable for particular forms of DTNs; however, it cannot be applied to CubeSat swarms, due to its limited resources. In [15] Li et al. also studied the reliable topology design problem in space DTN-based on a space-time graph approach. However, the aforementioned space-time graph based protocols do not put in the account the energy required for communication as most of them are proposed for conventional satellites.

B. *Routing in a DTN CubeSat Swarm*

We propose DTN Space Time Graph (DSTG) routing. DTN networking allows for nodes to communicate with each other with no contemporary paths present. This happens through store, carry and forward decisions. This occurs whenever data cannot be delivered by applying traditional Internet routing protocols [2]. In contrast to Internet protocols, which are characterized by bi-directional continuous end-to-end paths, high reliability, short Round Trip Time (RTT), a DTN can involve any mixture of the following features: frequent partitioning, intermittent connectivity, low reliability, sparse connectivity and predictable or semi-predictable mobility.

There are different types of routing/forwarding protocols for DTNs. These protocols can be classified into two main categories; namely, single copy/forwarding routing protocols and multiple copy/replication-based routing protocols. However, there are some other types of DTN routing such as history-based routing and space-time graph routing. A large number of current generic DTN routing protocols, i.e., Epidemic routing [16], PRoPHET [17], cyclic Mobispace [11], MaxProp [18] and spray and wait [19] have been studied under these different categories targeting different network environments, and cope with some of DTN optimization metrics, i.e., delay, message delivery ratio, buffer space and energy.

In this work, we consider DTN space time graph routing and focus on single copy/forwarding routing protocols. Most of current DTN routing protocols are flooding-based protocols or quota-based protocols. These protocols may exhaust the limited resources of the CubeSat and result in high overheads. However, in forwarding-based schemes the utilisation of the network resources is much less than other routing schemes, where there is only one copy of the message in the node buffer at any one time [20]. In addition, when the message is delivered to its destination no additional nodes can have a copy of it, thus there is no need for the destination node to give feedback to the other nodes to delete the message copies.

III. THE PROBLEM

Given a CubeSat swarm of a few hundred CubeSats, forming different network segments and any two CubeSats have the capability to communicate with each other when they become in the transmission range of one another. They can communicate through inter-CubeSat links as well as ground stations links; minimise the overall energy use of a CubeSat swarm for a given amount of routable data.

A. *Network Modelling*

A high density CubeSat swarm can be modelled as a directed space-time graph. Time is represented as discrete equally spaced intervals, i.e., $t = \{1 \dots, T\}$. Consider $V = \{v_1 \dots, v_n\}$ is the set of all nodes i.e., CubeSats and E is a set of directed edges. The position of each node is changing over time which in turn may result in a different topology at each time slot. The accumulation of these different topologies over period T represents the interaction between all nodes in such a DTN. Given a directed graph $G_t = (V_t, E_t)$ where V_t is a set of nodes n and E_t is a set of directed edges indicating a snapshot of the topology at time slot t and an edge $\xrightarrow[v_i^t v_j^t]{} \in E_t$ indicates that node v_i can transmit and receive from node v_j at time slot t, when the distance d_{i-j} is less than a certain threshold. We assume that there is at least one directed edge for any connected pair of nodes, so that the graph has in the extreme case $|E| = m <= 2n$ edges. Therefore, the dynamic network can be represented by the combination of all topology snapshots $\{G_t | t = 1 \dots \dots . T\}$. In graph G_t , which represents a network connectivity snapshot at time t, a fundamental path of length l from node v_i to node v_j is a sequence of l edges (hops) connecting nodes v_i and v_j without recursive nodes. However, analysing and designing DTN routing protocols using the sequence of static graphs is insufficient. For that reason, the static sequence of graphs can be converted to a space time graph $g = (\nu, \varepsilon)$, which is a directed graph with two types of links (spacial and temporal links) to be added among consecutive layers of the graph, allowing self-loop edges. Fig. 2 depicts a space time graph of 6 nodes and 6 time slots.

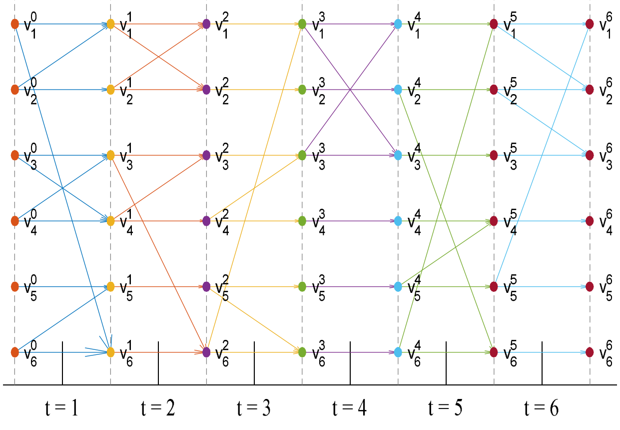

Fig.2. Space time graph g with 6 nodes and 6 time slots

In In Fig. 2, $T + 1$ layers of nodes are presented and there are 6 nodes in each layer. Therefore, $\nu = \{v_j^t | j = 1, \dots\dots n \; and \; t = 1 \dots \dots T \}$. A temporal link $\xrightarrow[v_j^{t-1} v_j^t]{}$ connects the node v_j with itself across successive $t - 1$ and t layers and this represents keeping a message on v_j, the message is not forwarded to another node during current time slot. A spacial link $\xrightarrow[v_j^{t-1} v_k^t]{}$ represents forwarding the message from node v_j to its neighbour v_k at time slot t, where temporal and spacial links $\xrightarrow[v_j^{t-1} v_j^t]{}$ and $\xrightarrow[v_j^{t-1} v_k^t]{} \in E$. Defining all temporal and spacial links on the graph g allows for data routing simulation and optimization. For example, there are multiple paths between v_4 and v_1, where v_4 can keep the message using the temporal link up to v_4^1 or v_4^2 and then forward it to $v3$ at time slot $t = 2$ or $t = 3$. Also, v_4 can directly forward the message to v_3 at time slot $t = 1$ using the spacial link and then v_3 can hold the message using the temporal link up to v_3^3. Then, v_3 can directly forward the bundle to v_1 at time slot $t = 4$. Therefore, the decision of which path should the message take to reach its destination is an optimisation problem. A decision whether a node may hold on the message or forward it to another node depends on many factors. One factor for example is the amount of energy which can be saved by holding onto the message. In addition, this reduces the overall traffic with less overhead for sending and receiving data.

B. *Shortest and Energy Reliable Path (SERP) Algorithm*

CubeSat swarm topology is changing over time. The encounter time and period between nodes is not fixed. Also, the unexpected failure of CubeSats leads to some nodes may disappear. Hence, an effective forwarding/routing protocol to optimize network performance is required. For instance, routing decisions can be taken based on the availability of energy at each node along a path. SERP aims to construct a space time graph R, as shown in Fig.3, which guarantees minimum connectivity between any two CubeSats over time, using only shortest paths with energy levels equal to or higher than a particular threshold which were 0.6 or 60% of charge in this case. This graph is extracted from the original space time graph (g) that includes a larger number of all possible paths between all pairs in the network as shown in Fig.4. SERP contains two main components: all pairs shortest path and most energy reliable path. In terms of location information, we assume that all nodes in the network have global knowledge. They are regularly updating their position information through ground stations. Also from the perspective of data link and physical layers, we assumed that all CubeSats will be capable to communicate through appropriate communication technology and antennas.

Definition-1: all pairs shortest path defines all minimum hop paths which connect any two nodes in the network over time T on the space time graph.

Definition-2: most energy reliable path is the minimum path among all paths which use only nodes (CubeSats) that have energy levels equal to or higher than a certain energy threshold.

Definition-3: Energy reliability threshold (ERT) is the minimum energy percentage required for a node to be selected as a next hop on any path between source and destination.

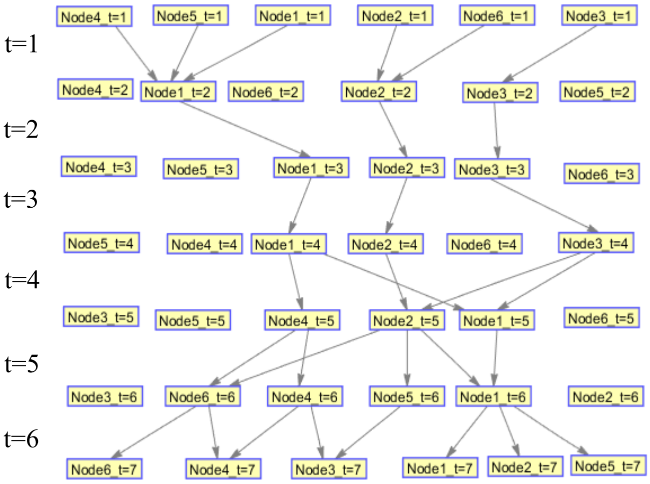

Fig.3. Space time graph R extracted from space time graph g at energy reliability threshold = 0.6

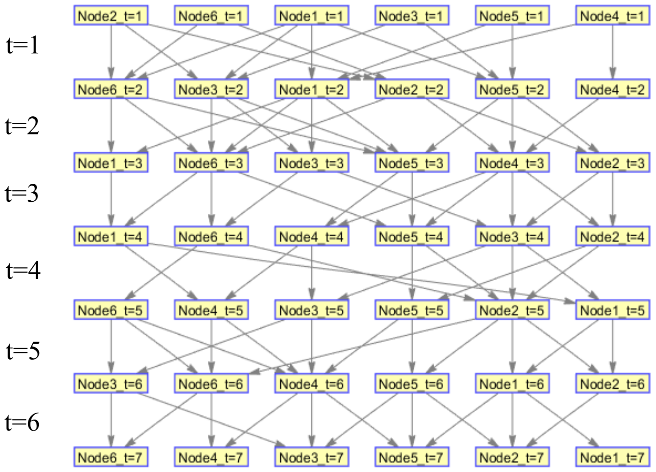

Fig.4. The space-time graph g represented as a biograph

Shortest and Energy Reliable Path (SERP)

```
Input: 3D adjacency matrix $Adj(:,:,t)$
Output: space-time graph $R$
Start,
1. Create space-time graph $g = n * (T + 1)$, where $t = 1 - T$ and $n$ is number of nodes.
2. graph $R = [\,]$
3. for all pairs in graph $(v_i^0, v_j^T) \in g$, where $1 \leq i,j \leq n$
     do
4.     find
         - all shortest using Dijkstra or BFS algorithms
         - among all shortest paths find $path_{SERP}$
           if $path_{SERP} \in$ graph g and $path_{SERP} \notin$ graph R
           then
             graph $R$ = graph $R \cup path_{SERP}$
5.         end if
6. end for
return graph $R$
```

IV. RESULTS AND DISCUSSION

The space-time graph g is generated from a series of static graphs. Each graph represents network topology connectivity at one time slot, as depicted in Fig.4. An adjacency matrix that reflects network topology at each time slot is generated using a random graph generator, so each node in the network has a random degree number and some nodes might be isolated. Then, a 3D adjacency matrix is created from all static graphs adjacency matrices. This 3D adjacency matrix represents the connectivity of all nodes over time T. We simulate a large set of networks with different number of nodes spreading on different time slots. SERP is then applied for the space-time graph g to extract another space-time graph R. Graph R only includes shortest paths that satisfy energy reliability requirements of SERP. SERP restricts the number of paths between any two nodes by routing bundles through shortest path that has nodes with minimum energy reliability requirement that is higher than or equal to a predefined energy threshold. That means routing decisions are based on two metrics the least cost path, which is the shortest path, and most energy reliable path.

A. Simulation

Breadth-first search (BFS) and Dijkstra graph search algorithms have been used with SERP to identify shortest paths between all pairs on the 3D adjacency matrix. Hence, SERP-BFS and SERP- Dijkstra are using different methods to search for shortest path, however both enforce same energy reliability requirement. Both SEPR algorithms are examined under different values of ERT. ERT can be any value between (0-1). The energy level at each node was uniformly distributed among all nodes in the network for all values of ERT. This means when ERT value increases the number of nodes that have energy levels higher than or equal to ERT will decrease. Also, any node with energy value less than ERT is not considered as a potential next hop in any path unless this node recharge its battery in the future to the ERT or higher. For example, a CubeSat can recharge its battery using solar cells. From the above, among all available shortest paths, SERP chooses the only shortest path that has nodes with energy levels higher than or equal to ERT. This process minimises the number of overall utilised edges in the network while maintaining connectivity between nodes over time. The amount of overall energy which can be saved in this case is equal to the amount of power per edge (α) multiplied by the number of non-used edges, as in (1). Therefore, saving more edges in the network reduces the overall energy consumption.

$$\mathrm{OSE} = (\mathrm{g_{Edges}} - \mathrm{R_{Edges}})\alpha \qquad (1)$$

Where OSE is the overall saved energy, while $\mathrm{R_{Edges}}$ and $\mathrm{g_{Edges}}$ are the number of edges in graph R and graph g respectively.

Graph R is generated by applying SERP algorithms on the graph g. Edge ratio is the ratio between number of edges of graph R to graph g and can be calculated from (2).

$$\text{Edge ratio} = \frac{\mathrm{R_{Edges}}}{\mathrm{g_{Edges}}} \qquad (2)$$

Both of SERP algorithms are compared with the Epidemic algorithm in terms of edges utilisation. Lower edge ratio means that an algorithm is capable of saving more edges and in turns saving more energy. All these three algorithms are examined under the same environment using same space-time graph as an input at each run. The simulation of each scenario is repeated 10 times in order to find the confidence intervals at each point.

B. Results

In Fig.5, the effect of a different number of time slots is studied. Edge ratio is examined at the different number of time slots 5-50 with ERT= 0.6. It can be seen that Epidemic algorithm has almost the same edge ratio at all time slots with no major impact. Similarly, SERP-Dijkstra is not significantly affected by the change of the number of time slots. However, edge ratio of SERP_BFS is decreasing when the number of time slots is increasing. This is because of the way how BFS and Dijkstra searching the graph is different. BSF is searching the graph layer by layer and this matches the concept of the space-time graph. In terms of edge ratio, the Epidemic algorithm has the highest edge ratio of about 60% compared with SERP algorithms. SERP-BFS achieved 40%-5% better performance than SERP-Dijkstra which has edge ratio fluctuating around 40%. The most obvious finding to emerge from this study is that SERP-BFS has very low edge ratio. This leads to some links will be saturated and create many bottleneck nodes. These nodes will lose their energy quickly and fall below ERT. In this case, smooth handover is required between these nodes and other neighbouring nodes with energy higher than ERT.

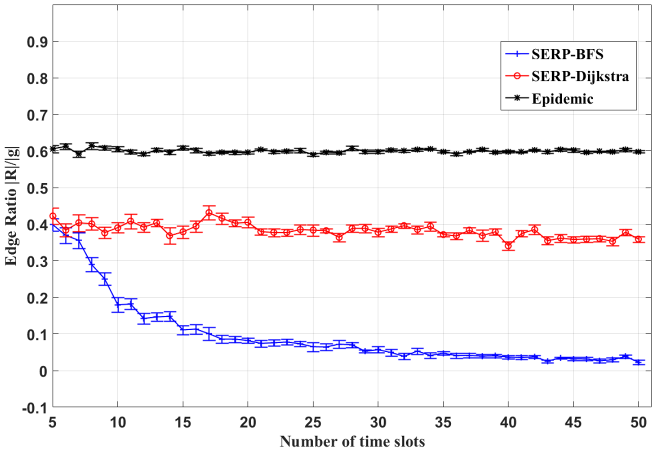

Fig.5 Edge ratio of Epidemic algorithm compared to SERP algorithms with 5-50 time slots and 50 nodes at ERT = 0.6

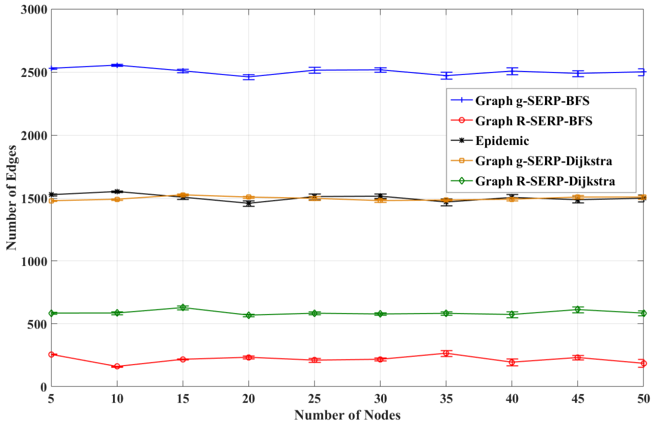

Fig. 6. Number of edges of graph g and graph R of both SERP algorithms vs number of edges of Epidemic algorithm.

The results show that SERP-BFS can save up to 90% of edges, while Epidemic can save about 40% of edges. Hence, SERP-BFS can save more than double of energy compared with the Epidemic. In meantime, SERP-Dijkstra can reduce the number of utilised edges to 60% of overall edges in graph g, as shown in Fig.6. Fig.7 depicts the relation between the number of nodes and the edge ratio at ERT=0.6. One of the more significant findings to emerge from this study is that there is no direct effect on the number of edges when the number of nodes is increased. However, it confirms the efficiency of SERP algorithms.

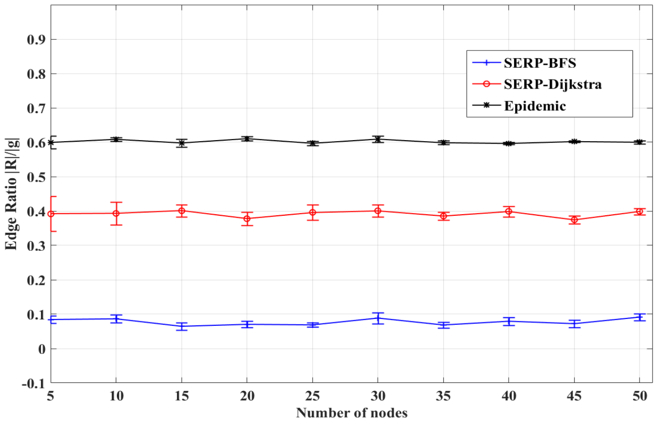

Fig.7. Edge ratio of Epidemic algorithm compared to SERP algorithms with 5-50 nodes and 20 time slots at ERT = 0.6

The relation between the edge ratio and ERT is also investigated. Fig.8 shows the edge ratio of the Epidemic algorithm against our SERP algorithms at different values of ERT. It can be seen that as the value of ERT rises the edge ratio gradually decays. The reason behind that is higher ERT values will restrict the number of edges to the edges belong to CubeSats with energy levels higher than or equal to ERT. Thus, only fewer CubeSats are available to form SERP paths. In contrast, minimizing the number of overall edges reduces the number of potential paths, which in turn reduces the message delivery ratio. Based on that, choosing the right value of ERT depends on various factors and it is an optimisation problem.

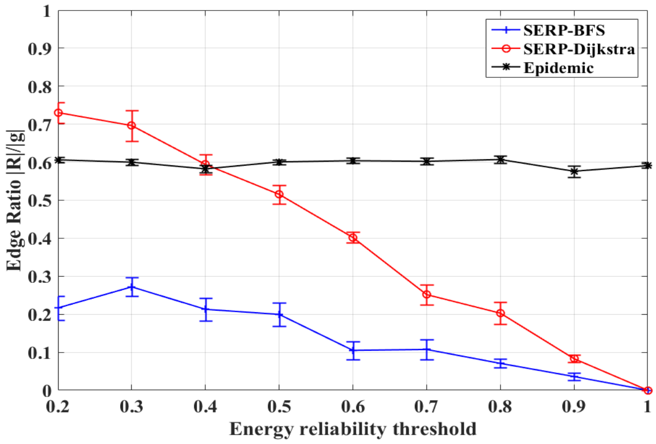

Fig. 8. Edge ratio of Epidemic algorithm compared to SERP algorithms with 50 nodes and 50 time slots at different ERT

Therefore, SERP_BFS and SERP-Dijkstra can significantly minimize the overall energy cost, while connectivity between CubeSats is maintained over time. Also, minimizing the number of edges will considerably reduce computation time and power required for updating routing tables. Moreover, all CubeSats with power levels less than ERT can go in battery save mode until they are able to recharge. This can also decrease the overall radio transmission and reception energy.

V. CONCLUSION

The problem of overall energy minimization for CubeSat swarm is studied. We proposed two novel SERP algorithms; namely, SERP-BFS and SERP-Dijkstra. These two algorithms are compared with Epidemic algorithm. The results show that both SERP algorithms have better performance than the Epidemic in terms of reducing the overall energy consumption, while connectivity between CubeSats is still maintained. However, SERP-BFS excels the SERP-Dijkstra with the ability to save much more edges. By reducing the overall number of edges, the overall energy cost can be minimized considering required power per edge.

REFERENCES

[1] F. E. Tubbal, R. Raad, and K. W. Chin, "A survey and study of planar antennas for pico-satellites," *IEEE Access,* vol. 3, pp. 2590-2612, 2015.

[2] E. Gill, P. Sundaramoorthy, J. Bouwmeester, B. Zandbergen, and R. Reinhard, "Formation flying within a constellation of nano-satellites: The QB50 mission," *Elsevier-Acta Astronautica,* vol. 82, pp. 110-117, January 2013.

[3] D. Selva and D. Krejci, "A survey and assessment of the capabilities of Cubesats for Earth observation," *Acta Astronautica,* vol. 74, pp. 50-68, 2012.

[4] B. Saptarshi, P. S. Giri, F. Rebecca, M. Daniel, C. Soon-Jo, and H. Fred, "A review of impending small satellite formation flying missions," in *53rd AIAA Aerospace Sciences Meeting*, ed: American Institute of Aeronautics and Astronautics, 2015.

[5] R. Radhakrishnan, W. W. Edmonson, F. Afghah, R. M. Rodriguez-Osorio, F. Pinto, and S. C. Burleigh, "Survey of Inter-satellite Communication for Small Satellite Systems: Physical Layer to Network Layer View," *IEEE Communications Surveys & Tutorials,* vol. 18, pp. 2442-2473, 2016.

[6] L. Qiao, C. Rizos, and A. G. Dempster, "Analysis and comparison of CubeSat lifetime," in *the 12th Australian Space Conference, Sydney, Australia*, 2013, pp. 249-260.

[7] B. Klofas, J. Anderson, and K. Leveque, "A survey of cubesat communication systems," in *5th Annual CubeSat Developers' Workshop*, 2008.

[8] S. S. Arnold, R. Nuzzaci, and A. Gordon-Ross, "Energy budgeting for CubeSats with an integrated FPGA," in *Aerospace Conference, IEEE*, 2012, pp. 1-14.

[9] L. Xiaofeng and H. Pan, "An Energy-Efficient n-Epidemic Routing Protocol for Delay Tolerant Networks," in *Networking, Architecture and Storage (NAS), 2010 IEEE Fifth International Conference on*, 2010, pp. 341-347.

[10] R. Radhakrishnan, W. Edmonson, F. Afghah, R. Rodriguez-Osorio, F. Pinto, and S. Burleigh, "Survey of inter-satellite communication for small satellite systems: an OSI framework approach," *IEEE Communications Surveys & Tutorials,* pp. 1-1, 2016.

[11] C. Liu and J. Wu, "Routing in a cyclic mobispace," in *the 9th ACM international symposium on Mobile ad hoc networking and computing*, 2008, pp. 351-360.

[12] S. Iranmanesh and K.-W. Chin, "A novel mobility-based routing protocol for semi-predictable disruption tolerant networks," *International Journal of Wireless Information Networks,* vol. 22, pp. 138-146, 2015.

[13] S. Merugu, M. H. Ammar, and E. W. Zegura, "Routing in space and time in networks with predictable mobility," Georgia Institute of Technology, GIT, Tech. Rep., CC-04-07, 2004.

[14] M. Huang, S. Chen, Y. Zhu, and Y. Wang, "Topology control for time-evolving and predictable delay-tolerant networks," *IEEE Transactions on Computers,* vol. 62, pp. 2308-2321, 2013.

[15] F. Li, S. Chen, M. Huang, Z. Yin, C. Zhang, and Y. Wang, "Reliable topology design in time-evolving delay-tolerant networks with unreliable links," *IEEE Transactions on Mobile Computing,* vol. 14, pp. 1301-1314, 2015.

[16] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," Technical Report CS-200006, Duke University, 2000.

[17] A. Lindgren, A. Doria, and O. Schelén, "Probabilistic routing in intermittently connected networks," *ACM SIGMOBILE mobile computing and communications review,* vol. 7, pp. 19-20, 2003.

[18] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, "MaxProp: routing for vehicle-based disruption-tolerant networks," in *INFOCOM*, 2006, pp. 1-11.

[19] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and wait: an efficient routing scheme for intermittently connected mobile networks," in *ACM SIGCOMM workshop on Delay-tolerant networking*, 2005, pp. 252-259.

[20] D. Henriksson, T. F. Abdelzaher, and R. K. Ganti, "A caching-based approach to routing in delay-tolerant networks," in *the 16th International Conference on*, 2007, pp. 69-74.