Defence Research and
Development Canada

Recherche et développement
pour la défense Canada

DEFENCE R&D DÉFENSE

Routing Algorithm Analysis for a Nanosatellite Constellation in Low Earth Orbit

S. McKenzie-Picot and P. Gavigan

Defence R&D Canada – Ottawa

Technical Memorandum
DRDC Ottawa TM 2013-081
October 2013

Canada

Routing Algorithm Analysis for a Nanosatellite Constellation in Low Earth Orbit

S. McKenzie-Picot
Defence Research and Development Canada – Ottawa

P. Gavigan
Defence Research and Development Canada – Ottawa

Defence Research and Development Canada – Ottawa
Technical Memorandum
DRDC Ottawa TM 2013-081
October 2013

Abstract

This paper examines possible routing algorithms in a proposed network of nanosatellites with the goal of finding networking methods that could be effectively used in a dynamic environment with limited computing power. Three routing algorithms were created and their performance was compared to Dijkstra's shortest path algorithm. The chaos method transmitted to accessible satellites at random. The minimum distance method transmitted to the closest unused satellite. The grouping method grouped satellites by their physical position in the network and made transmission decisions based on these groups. Each algorithm was run for every possible combination of source and destination satellite in the network to find average transmission times, distances, and number of network nodes used. It was found that the grouping method had performance comparable to Dijkstra's algorithm while requiring less computing power, and was acceptable for high-speed network requirements, such as live voice. The chaos method was acceptable for lower speed applications and for satellites that had little computing power, and the minimum distance method improved on chaos method with little extra computing power required.

Résumé

Le présent document examine des algorithmes de routage qui pourraient être mis en œuvre dans un réseau proposé de nanosatellites afin de trouver des méthodes efficaces de réseautage à utiliser dans un environnement dynamique offrant une puissance de calcul limitée. Nous avons créé trois algorithmes de routage et comparé leur rendement à celui de l'algorithme du plus court chemin de Dijkstra. La méthode fondée sur le chaos transmet de manière aléatoire vers un satellite accessible. La méthode de la distance minimale transmet vers le satellite non utilisé le plus près. La méthode de regroupement crée des groupes de satellites selon leur position physique dans le réseau et prend des décisions dacheminement en conséquence. Chaque algorithme a été exécuté pour chaque combinaison possible de satellite source et de destination dans le réseau afin de déterminer les temps et les distances de transmission moyens ainsi que le nombre de nœuds de réseau utilisés. Nous concluons que la méthode de regroupement a un rendement comparable à celui de l'algorithme de Dijkstra tout en exigeant une moindre capacité de calcul, et qu'elle donne des résultats acceptables pour les besoins de réseaux à haute vitesse (p. ex., voix en direct). La méthode fondée sur le chaos est acceptable pour les applications à basse vitesse et pour les satellites disposant d'une faible capacité de calcul, alors que la méthode de la distance minimale donne de meilleurs résultats que la méthode du chaos sans exiger beaucoup de puissance de calcul supplémentaire.

This page intentionally left blank.

Executive summary

Routing Algorithm Analysis for a Nanosatellite Constellation in Low Earth Orbit

S. McKenzie-Picot, P. Gavigan; DRDC Ottawa TM 2013-081; Defence Research and Development Canada – Ottawa; October 2013.

Background: This document explores possible routing algorithms in a network of nanosatellites. A network of nanosatellites is a dynamic environment with potentially limited computing power, and this document seeks to provide networking methods that could be effectively used in these conditions. It compares the performance of three author-created routing algorithms with the performance of Dijkstra's algorithm, a commonly used routing algorithm that finds the shortest path between the source node and all other nodes in a network.

Principal results: Three routing algorithms were created by the author and their performance was compared to Dijkstra's shortest path algorithm. Dijkstra's shortest path algorithm finds all possible paths between the source satellite and all other satellites and selects the shortest available.

Satellites using the first algorithm, the "chaos method," selected a satellite at random from a list of accessible neighbours and transmitted to it, repeating until the destination had been reached.

Satellites using the second algorithm, the "minimum distance method", were given the path history of the transmission. The satellites transmitted to the closest unused accessible satellite. If the transmission entered a loop by revisiting two satellites in the same order, then the next satellite broke the loop by transmitting to an accessible satellite selected at random. Again, this continued until the destination had been reached.

Satellites using the third algorithm, the "grouping method", were separated into groups based on their physical position in the network. The source satellite first checked if it could transmit to the destination. If it could not, it checked if it was in, or could transmit to, the destination's group, in which case it sent the transmission to a satellite in the group. If all else failed, it sent the transmission to the closest available satellite. This repeated until the destination was reached.

The grouping method had performance comparable to Dijkstra's algorithm while requiring less computing power, and was acceptable for high-speed network requirements, such as live voice. The chaos method was acceptable for lower speed applications and for satellites that had little computing power, and the minimum distance method improved on chaos method with minimal increase in required computing power.

Future work: Potential work in this area could include handling multiple transmissions at the same time, or the response of algorithms to the changing network. Satellites are constantly moving, which causes the network to be constantly changing geometry. Thus, the fleeting nature of potential paths must be considered, particularly in cases where the path is planned out, as a transmission could get only part of the way to its destination before the planned path disappears.

Sommaire

Routing Algorithm Analysis for a Nanosatellite Constellation in Low Earth Orbit

S. McKenzie-Picot, P. Gavigan ; DRDC Ottawa TM 2013-081 ; Recherche et
développement pour la défense Canada – Ottawa ; octobre 2013.

Contexte : Le présent document examine des algorithmes de routage qui pourraient être mis en œuvre dans un réseau de nanosatellites. Un réseau de nanosatellites est un environnement dynamique qui peut disposer d'une puissance de calcul limitée. Nous présentons ici des méthodes de réseautage efficaces à utiliser dans ces conditions. Le document compare le rendement de trois algorithmes de routage créés par l'auteur à celui de l'algorithme de Dijkstra. Ce dernier algorithme est souvent utilisé pour trouver le plus court chemin entre le nœud source et les autres nœuds d'un réseau.

Principaux résultats : Nous avons créé trois algorithmes de routage et comparé leur rendement à celui de l'algorithme du plus court chemin de Dijkstra. L'algorithme de Dijkstra trouve tous les chemins possibles entre le satellite source et les autres satellites et choisit le chemin disponible le plus court.

Les satellites qui utilisent le premier algorithme, fondé sur le chaos, sélectionnent un satellite au hasard parmi les voisins accessibles et transmettent vers celui-ci. Le processus se répète jusqu'à larrivée à destination.

On donne aux satellites qui utilisent le deuxième algorithme, fondé sur la distance minimale, un historique du chemin de transmission. Les satellites transmettent vers le satellite accessible non utilisé le plus près. Si la transmission produit une boucle, en transmettant deux fois d'un satellite particulier vers un autre, le second satellite brise la boucle en transmettant vers un satellite accessible choisi au hasard. Encore une fois, le processus se répète jusqu'à l'arrivée à destination.

Les satellites qui utilisent la troisième méthode, fondée sur le regroupement, sont divisés en groupes selon leur position physique dans le réseau. Le satellite source vérifie premièrement s'il peut transmettre vers la destination. Sinon, il vérifie s'il se trouve dans le même groupe que le satellite de destination ou s'il peut transmettre vers ce groupe. Le cas échéant, il transmet vers un satellite de ce groupe. Si cela n'est pas possible, il transmet alors vers le satellite disponible le plus près. Le processus se répète jusqu'à l'arrivée à destination.

La méthode de regroupement donne un rendement comparable à celui de l'algorithme de Dijkstra tout en exigeant une moindre capacité de calcul, et elle donne des résultats acceptables pour les besoins de réseau à haute vitesse (p. ex., voix en direct). La méthode fondée sur le chaos est acceptable pour les applications à basse vitesse et pour les satellites disposant d'une faible capacité de calcul, alors que la méthode de la distance minimale donne de meilleurs résultats que la méthode du chaos sans exiger beaucoup de puissance de calcul supplémentaire.

Recherches futures : De futurs travaux dans ce domaine pourraient porter sur le traitement simultané de multiples transmissions ou sur la capacité d'adaptation des algorithmes aux changements du réseau. Les satellites bougent sans cesse, ce qui modifie constamment la géométrie du réseau. Par conséquent, il faut tenir compte du caractère transitoire des chemins possibles, surtout dans les cas où le chemin est planifié, puisque le chemin prévu pourrait disparaître avant que la transmission n'atteigne la destination.

This page intentionally left blank.

Table of contents

List of figures

List of tables

1 Introduction

This paper examines possible routing algorithms in a network of nanosatellites (satellites weighing less than 10 kg [1]) with the goal of finding networking methods that could be used effectively in a dynamic environment with limited computing power. The performance of the routing algorithms is characterised by the average time a transmission takes to reach its destination, the average path distance and the average number of satellites, or network nodes, the transmission passes through.

Three routing algorithms with a range of computing and network knowledge requirements were created and their performance is compared to Dijkstra's algorithm, an algorithm that finds the shortest path between network nodes. Routing algorithms are tested using a constellation of 48 nanosatellites, which is outlined in Section 3. The algorithms developed and results are discussed in Section 4. A summary and conclusion is shown in Section 5 and potential future work is discussed in Section 6.

2 Background

There is growing interest in satellite networks with intersatellite links for applications in defence [2] and for proposed internet-in-space endeavours [3]. Routing algorithms appropriate for dynamic networks are important for realizing the potential of these satellite networks. There has been research done on routing algorithms for satellite networks that have significant computational abilities and a small number of intersatellite links such as Iridium, a satellite phone company with a constellation of 66 satellites with four intersatellite links per satellite [4].

Previous research includes using the Iridium constellation to test possible routing techniques, including Dijkstra's algorithm and Bellman-Ford algorithm (both shortest-path algorithms) as methods of updating routing tables [4]. A distributed adaptive routing algorithm was proposed by [5] for a constellation of 36 low orbit and 12 high orbit satellites with 4 intersatellite links each, that produces new routing tables every 5 seconds and calculates all feasible paths to minimize delay and maximize throughput [4]. Dynamically routing messages by using distributed processing and calculating the cost of each route, considering congestion and intermodal distances, and selecting the least-cost route has also been proposed [6]. Finally, [2] assumes each satellite is given the connectivity of the network, forcing routing tables to be complex to account for the dynamic network changes [4]. It uses a version of the ALOHA protocol that adapts and compensates for collisions that occur, and uses the routing takes to transmit the message to the destination node [2].

These routing methods are for satellites with large amounts of computing power, and, in those with specified constellations, few intersatellite links. There is little research done for constellations with limited computing power and large numbers of links. The failed Teledesic satellite system planned to provide internet in space using a 924 satellite constellation with 8 intersatellite links per satellite [3]; however, Teledesic's planned routing methods are not publically available. Thus, this paper looks into possible solutions for satellite constellations with many intersatellite links and various levels of computing power.

3 Assumptions

The satellite constellation used in this paper was created in [1]. It includes three planes inclined at 51.6 degrees, each containing nine nanosatellites (shown in red, orange and yellow in Figure 1) and three sun-synchronous planes, each containing seven nanosatellites (shown in blue and green) for a total of 48 satellites, each of which are considered network nodes.

A constellation of nanosatellites (satellites weighing less than 10 kg) was chosen because of the recent interest in them, which includes discussion of using nanosatellites to create a network in space. Despite the interest in a nanosatellite network, little documentation is readily available to support such an idea [1]. Such small satellites have limited power for computing and limited space for powerful computers, potentially preventing them from using traditional routing methods.

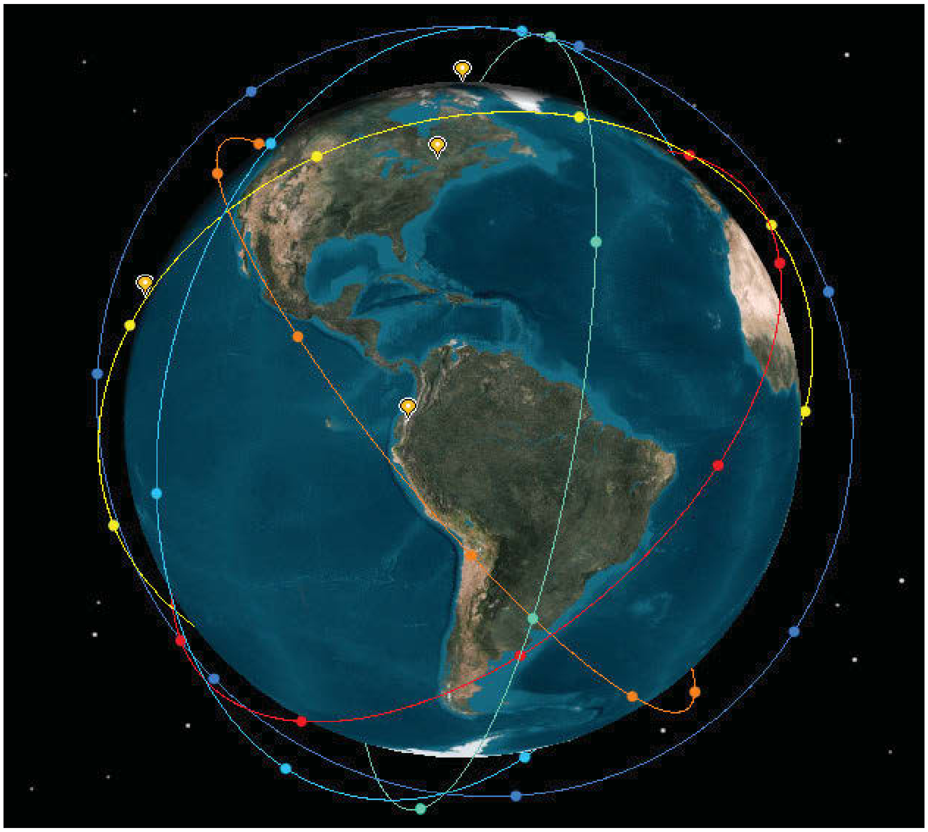

Figure 1: *Constellation Used for Intersatellite Networking.*

It is assumed that satellite movement is negligible during the time it takes for transmission of a message to occur. Thus, satellites are treated as fixed relative to each other for the duration of the transmission. Additionally, it is assumed that satellite processing time is instantaneous. Typical processing and queuing latencies are at present around 100 microseconds [7], which produce delays that are negligible compared to those caused by transmission distances.

4 Routing algorithm analysis

The following sections outline the routing algorithms used, discuss their strengths and weaknesses, and suggest appropriate uses. The performance of the algorithms created by the author (algorithms described in Sections 4.2 to 4.4) were compared to Dijkstra's algorithm, described in Section 4.1. The performance of each algorithm was characterised by the average path distance and time, and the network knowledge required by each satellite, and each algorithm was run for every possible combination of source and destination satellite in the network. For the purposes of this analysis, a loop is defined as two satellites revisited in the same order.

4.1 Dijkstra's algorithm

Dijkstra's shortest path algorithm is a commonly used routing algorithm that finds the shortest path between the source node and all other nodes in a network [4]. It searches for all possible paths and compares them all to find the shortest path (distance) between the source node and all other nodes.

4.1.1 Algorithm description

The algorithm used in this analysis was modified from [8], and is provided in Figure 2 (figure adapted from [9]). Satellites using Dijkstra's algorithm find every possible path between the source satellite and all other satellites in the network, and, from these potential paths, select the shortest paths between the source and every other satellite.

First, all available satellites in the network are identified. The source node is then assigned a cost of 0, or no distance, and is marked as "permanent". The source's neighbouring satellites are then found and costs, or distances between these satellites and source, are assigned. The satellite with the smallest cost is marked as permanent and all other neighbour satellites are checked if they can be reached by more than one route. If any can be reached by multiple routes, the shortest route is selected and that satellite is marked as permanent. If no other route is available, the current route is marked as tentative because there may be shorter routes found in the future. The last satellite marked as permanent becomes the current satellite and its neighbours are found. The process of finding neighbour satellites, assigning distances and comparing to find the shortest route continues until all satellites in the network have a permanent route.

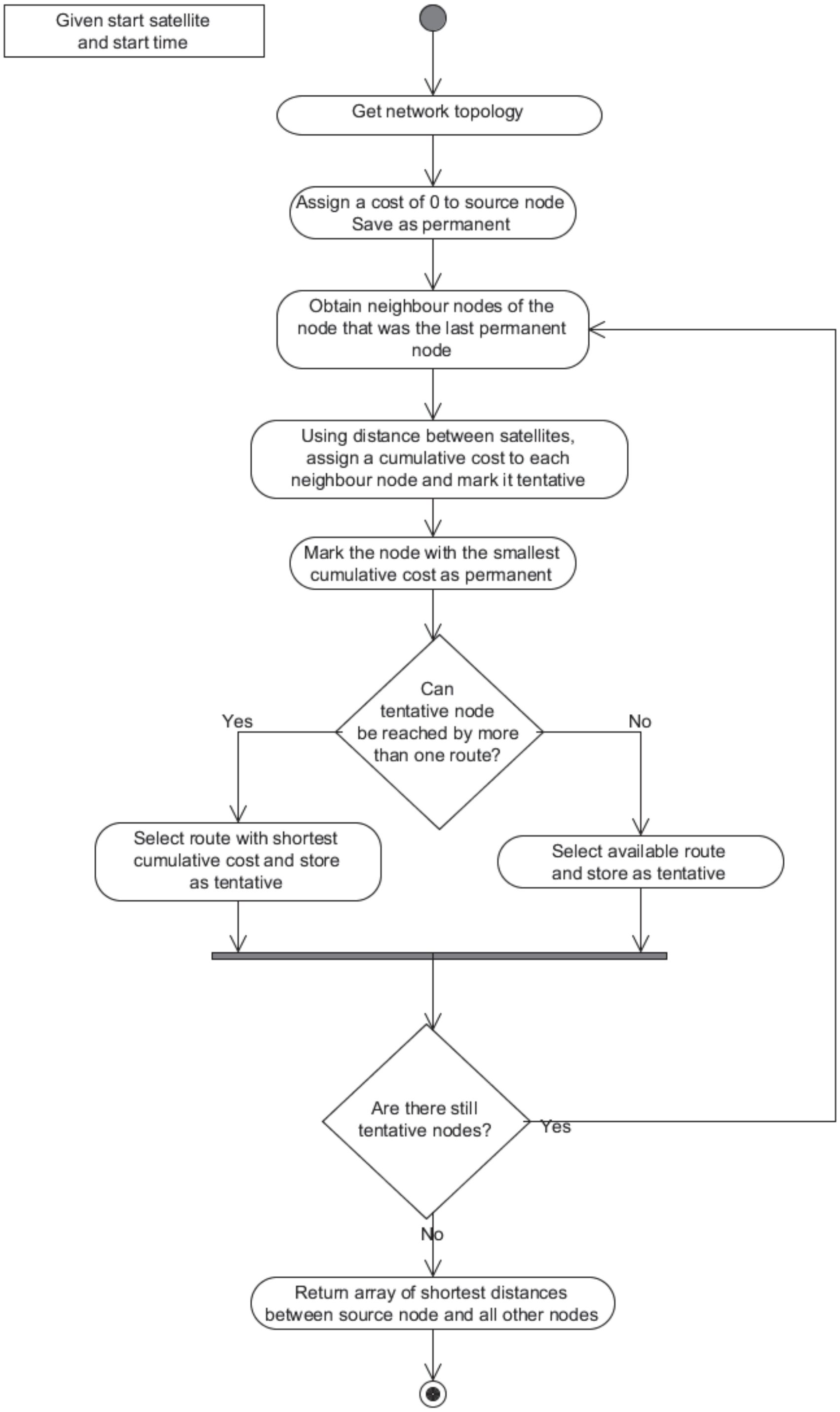

Figure 2: *Dijkstra's Algorithm.*

4.1.2 Results and discussion

Once all routes have been found, messages can be sent from the source satellite to any destination satellite. Because of this planning, there are no satellite revisits and message transmission takes little time and is reliable.

Because of the accuracy and shortness of the path produced by Dijkstra's algorithm, the results from this algorithm, provided in Table 1, are considered the best case. The results of the other algorithms used are compared to the results found with Dijkstra's. The results from all algorithms, including Dijkstra's, were created assuming the constellation described in Section 3.

Table 1: *Results of Transmissions Sent Using Dijkstra's Algorithm.*

Average Path Distance of Transmissions (km)	11507
Average Transmission Time (s)	0.038
Average Number of Nodes	3.5

Dijkstra's algorithm is only run once per path, with the path planning occurring before the transmission is sent. Thus, while Dijkstra's algorithm takes time to find the shortest path, this time cost only occurs once. However, Dijkstra's algorithm requires knowledge of the position of each network node. Thus, each satellite must be given the location of all other satellites in the constellation, which may not be feasible. Knowledge of other satellite positions could be attained through orbit propagation technology or intersatellite communications, both of which add cost to the satellite constellation. Alternatively, a ground station could be more powerful and be given the locations of all satellites and plan the path from start to finish, which would permit the satellites to be less complex, simply sending transmissions along the path as instructed. However, the computing power required may prevent the ground station from being easily portable.

Dijkstra's algorithm is less adaptable than other algorithms to changes in network topography. As the path is planned out in advance, a failure that leaves a future satellite inaccessible halts the transmission. If a failure occurs, the path plan must be recomputed, and if the satellites are simply following a path set by the ground station, they must receive a new path from that ground station.

4.2 Chaos method

Unlike Dijkstra's algorithm, the transmission path is not planned with the chaos method. The chaos method is based on the assumption that satellites and ground stations have no path planning ability and are not given overall network topology, but do know which satellites they themselves can contact.

4.2.1 Algorithm description

The chaos method algorithm, provided in Figure 3, selects a satellite at random from its available intersatellite contacts and transmits to it. This continues until the destination satellite has been reached.

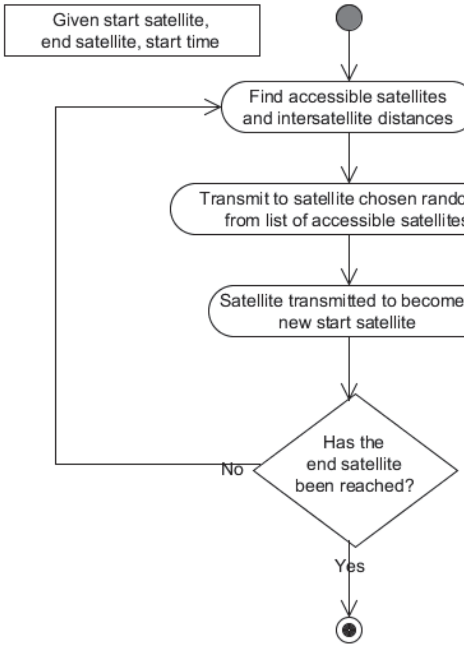

Figure 3: *Chaos Method.*

4.2.2 Results and discussion

The results for using the chaos method on the constellation described in Section 3 are provided in Table 2, which compares the minimum possible time, distance and number of nodes of constellations using the chaos method with the average time, distance and number of nodes. It also shows the typical time assuming only polar planes of the constellation described in Section 3 are used. The polar planes were chosen to show a constellation with far fewer nodes: when only polar planes are used, the constellation shrinks from 48 nodes to 21.

Table 2: *Results of Transmissions Sent Using Chaos Method.*

	48 Satellites		**21 Satellites**
	Minimum	*Average*	*Average*
Average Path Distance (km)	11507	284337	187158
Average Transmission Time (s)	0.038	0.95	0.62
Average Number of Nodes	3.5	62.9	35.7

The minimum possible time it would take for a satellite message to travel from source to destination is identical to the time it takes for a message to be sent using Dijkstra's algorithm. This assumes that the satellites picked, at random, the same path that was found using Dijkstra's algorithm. The probability of this occurring is dependent on the number of satellites in a network and the number of intersatellite contacts. The formula for that probability is shown below in Equation 1.

$$P = \frac{1}{M^n} \tag{1}$$

In Equation 1, M is number of intersatellite links, and n is the number of nodes the message passes through. Assuming each satellite has, on average, 8.5 intersatellite contacts at any given time (average for the full satellite constellation described in Section 3), and the average number of nodes the message passes through is 3.5 (see average number of nodes in Table 1), then the probability of the chaos method choosing Dijkstra's path is 0.056%.

Typically, the chaos method takes 25 times as long as Dijkstra's algorithm to reach the destination satellite, and passes through 62.9 nodes. The time delay of 0.95 seconds makes it inappropriate for live voice transfers (i.e. satellite phone), which require delays of 0.4 seconds and below [7]. The delay decreases with smaller networks, such as the 21 satellite polar plane only network, because there are fewer available paths.

The chaos method is advantageous when satellites lack the computing power to make decisions. It is slower than other methods, but the message eventually reaches its destination. Additionally, it adapts well to failures in the network, because it transmits to whichever satellite it can reach, without assuming beforehand that these satellites are accessible.

However, the chaos method is not fast, and decreases in speed as the network increases in size, due to the larger number of options available. Thus, if high-speed networks are needed, or there are a large number of nodes in the network, another method would be more appropriate.

4.3 Minimum distance and known path history

Like the chaos method, satellites using minimum distance and known path history do not plan paths. However, satellites using minimum distance and known path history make note of previously used satellites and attempt to transmit to unused ones in hopes of improving on the performance of the chaos method.

4.3.1 Algorithm description

The minimum distance and known path history method assumes the current satellite is given the other satellites it can communicate with and the recent path history, P. The diagram of the algorithm is provided in Figure 4. The current satellite looks for the closest satellite that has not already been used. If the transmission enters a loop, the algorithm reverts to the chaos method and transmits to a satellite at random. After selecting a satellite at random, the next satellite returns to searching for an unused, accessible satellite.

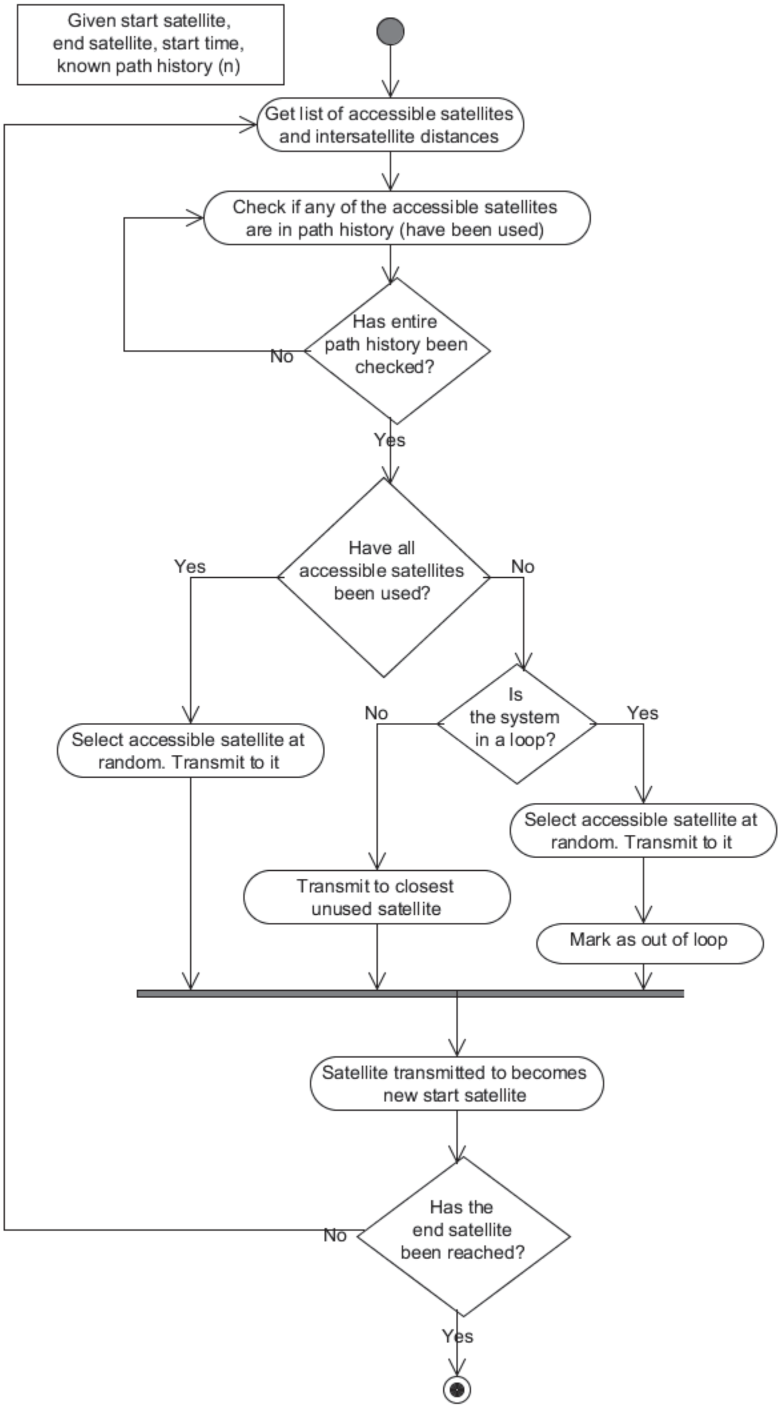

Figure 4: *Minimum Distance and Known Path History.*

4.3.2 Results and discussion

The algorithm was tested using a known path history, *P*, of two to eight nodes, and the results of those tests are provided in Table 3. Thus, each satellite in the path is given the previous *P* number of satellites that the transmission had passed through. When the current satellite is given the previous two satellites (*P* = 2), the percent of satellites that revert to chaos method is 88.9%. This is shown in Table 3. Despite the large percentage of paths that revert to the chaos method when *P* = 2, the path distance and transmission time drop by 30% when compared to the pure chaos method. The average number of nodes used is slightly larger (66 vs 63 nodes) because satellites transmit to their closest neighbour, but the additional nodes cause negligible changes in the overall time.

Table 3: *Results of Transmissions Sent Using Minimum Distance Network.*

Known Path History	**2**	**4**	**6**	**8**
Percent of Transmissions Reverting to Chaos Method	88.9	45.9	28.8	31.4
Average Path Distance (km)	200240	119164	102019	95273
Average Transmission Time(s)	0.67	0.40	0.34	0.32
Average Number of Nodes	66.3	40.5	36.1	33.9

When a known path history of 6 nodes is used, the percent of paths that default to chaos method enters a minimum, and stabilizes at higher values of *P*. This is due to the number of intersatellite communications available. Each satellite could communicate with 8.5 satellites on average. When the known path history is greater than 8 nodes, and all satellites have been used, the algorithm again reverts to selecting a satellite at random.

Like the chaos method, satellites using minimum distance and known path history are not required to plan paths. The satellites do not need to be given the entire network topology, only the locations of the nodes they can communicate to. However, known path history improves on the transmission time seen using chaos method. Improvements are more evident in large networks, with known path history sending transmissions that take as little as 34% the time of chaos method in a network with 48 nodes. In a network of 21 nodes, known path history improves to a maximum of 70% of the transmission time of chaos method.

Thus, similar to the chaos method, this algorithm could be appropriate for a satellite constellation with minimal computing power and non-critical messages. This algorithm is potentially appropriate for voice transmissions, because with larger known path histories (*P*) the average delay in the full, 48 satellite constellation, drops below 0.4 seconds, the maximum delay for voice transmissions [7]. For this constellation, a value of *P* = 4 or larger is appropriate for voice transmissions.

Table 4: *Results of Transmissions Sent Using Minimum Distance Network, using only Polar Planes.*

Known Path History	**2**	**3**	**4**
Percent of Transmissions Reverting to Chaos Method	81.0	76.8	24.9
Average Path Distance (km)	179625	130421	168209
Average Transmission Time (s)	0.60	0.43	0.56
Average Number of Nodes	43.7	45.4	38.9

4.4 Grouping

Unlike both chaos and minimum distance methods, satellites using the grouping algorithm do a small amount of planning, making decisions that send the transmission closer to the destination satellite.

4.4.1 Algorithm description

The grouping algorithm considers the satellites to be in groups: satellites are identified as in-plane or out-of-plane in relation to the current satellite. Satellites using this algorithm do not plan the entire path of the transmission, but are given the destination satellite and make decisions to send the transmission as close to the destination satellite as they can. The diagram for the grouping algorithm is provided in Figure 5.

Starting with the source satellite, it searches its accessible neighbours for the destination satellite. If the destination satellite is accessible, it transmits to the destination satellite. If the destination satellite is not accessible, the source satellite checks if the destination satellite is in the source plane. If the destination satellite is in the source plane, the source satellite transmits to the next satellite in its plane, to send the transmission closer to the destination satellite. If the destination satellite is not in the source plane, the source satellite checks if it can access any satellites in the destination plane. If it can, it transmits to the closest satellite in the destination plane. If not, it transmits to the closest unvisited satellite in the current plane. The satellite that received the transmission becomes the source satellite and this process repeats until the destination satellite is reached.

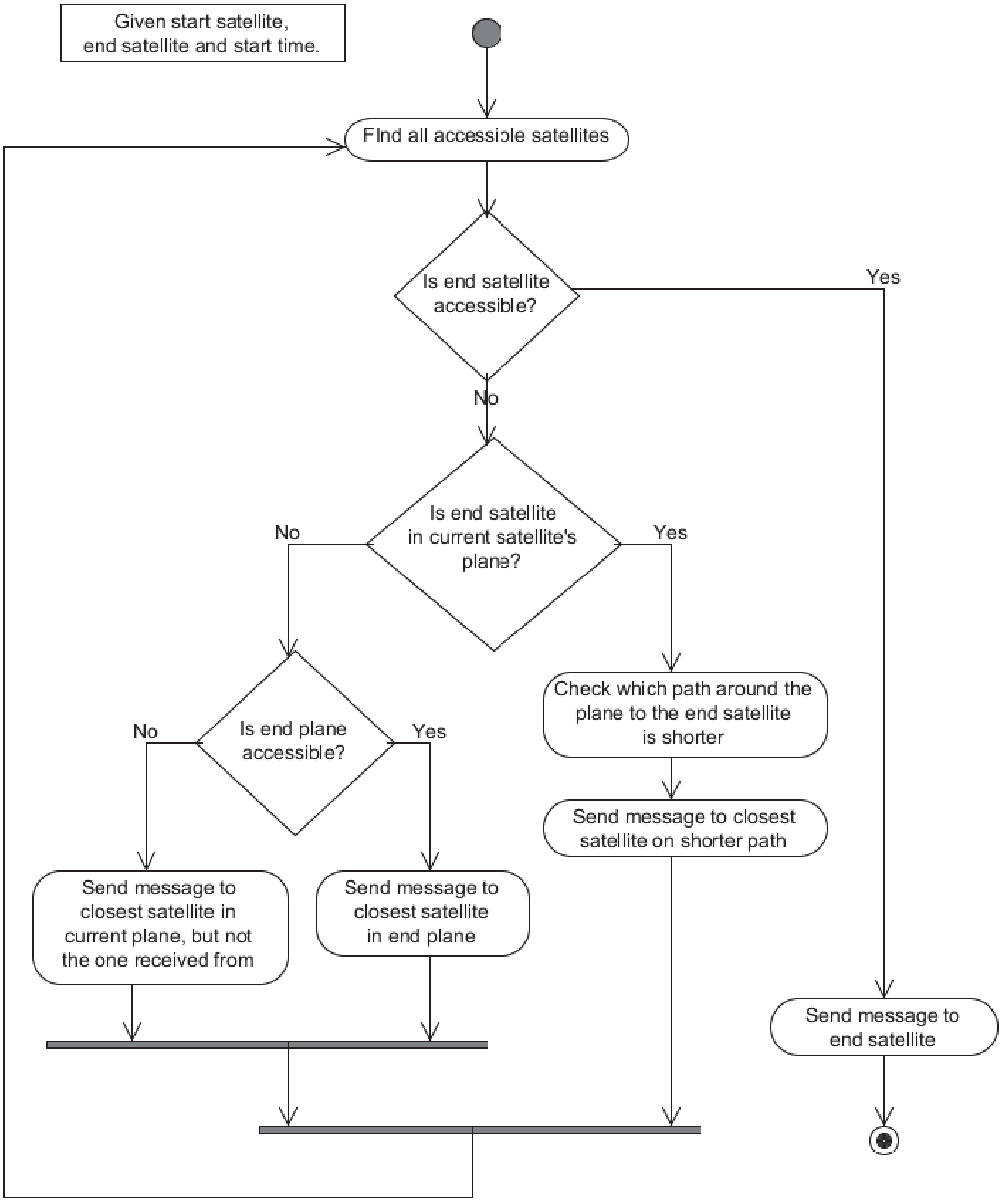

Figure 5: *Grouping Method.*

4.4.2 Results and discussion

Results of the grouping algorithm are provided in Table 5. The results are similar to Dijkstra's algorithm but, unlike using Dijkstra's algorithm, satellites using the grouping algorithm do not require knowledge of the entire network. Each satellite must be given the destination satellite, the satellites it can communicate with, if those satellites are in its group or not, and the distances to those satellites. Satellites could be given the destination satellite through a tag on the message. Each satellite must also be given the number of satellites per plane, or the distribution of satellites around the plane to make decisions about which direction around the plane is faster.

When compared to Dijkstra's algorithm, 40% of satellite paths found by the grouping method are identical to those found by Dijkstra. The average distance a transmission must travel using the grouping method is on average 136% that of Dijkstra paths, creating a time difference between the two methods of 0.014 seconds. The grouping method uses very few satellites when compared to chaos and known path history methods, similar to the numbers seen using Dijkstra, using on average 4.2 satellites, while Dijkstra's uses 3.5 satellites. It is more adaptable than Dijkstra's algorithm to network failures because it does not plan the entire path, and will adapt its path to whichever satellites it is able to communicate with.

Because satellites using the grouping algorithm do not plan the entire path, the delay from running the algorithm is smaller than the delay from Dijkstra's algorithm. However, unlike Dijkstra's algorithm, the grouping algorithm must be run once per node, as each satellite makes a new decision about where to send the transmission.

The grouping algorithm is appropriate for time-sensitive transmissions, and can be used for live voice transfers. It can be used on larger constellations that have the computing power and network knowledge to make decisions about where to send the message.

Table 5: *Results of Transmissions Sent Using Grouping Method.*

Average Path Distance of Transmissions (km)	15631
Average Transmission Time (s)	0.052
Average Number of Nodes	4.2

4.5 Comparison of transmission methods

A comparison of results is provided in Table 6. It shows the trade-off between the performance of the network in terms of transmission delay, path distance and number of network nodes that process the message for each algorithm.

Required network knowledge is calculated as the percent of network topography each satellite must be given. Percentages are valid for the constellation described in Section 3 (see Figure 1).

In the chaos method, each satellite must be given which satellites it can communicate with at any given time. In the constellation used in this paper, the number of intersatellite links per satellite is on average 8.5.

In minimum distance, each satellite must be given the satellites it can communicate with plus the satellites that have been used. Depending on the path history used, this varies between averages of 10.5 and 16.5 satellites.

In the grouping method, each satellite must be given which satellites it can communicate with, plus the satellites in its group, plus the destination satellite and the satellite it received the transmission from. Assuming an average of 8 satellites per group, this gives an average of 18.5 satellites.

Satellites using Dijkstra's method must be given the topography of the entire network. In the case of the constellation described in Section 3, this is 48 satellites.

It should be noted that, while minimum distance and grouping methods have similar percentages of required knowledge, grouping method requires additional routing planning, requiring more computing power. Additionally, the required network knowledge in Table 6 is the maximum that is seen with minimum distance. If a known path history of two nodes was used, the required network knowledge drops to 22%.

Dijkstra's algorithm requires the most extensive knowledge of the network, and also the most computing power, but message transmission takes little time. Constellations using the chaos method require minimal knowledge of the network, but lose the performance of Dijkstra's algorithm. Minimum distance and grouping methods fall between, with grouping approximating results found with Dijkstra, while requiring significantly less network knowledge.

Table 6: *Comparison of Transmission Methods.*

	Dijkstra	**Chaos**	**Minimum Distance Known Path History $P = 8$**	**Grouping**
Average Path Distance (km)	11507	284337	95273	15631
Average Transmission Time (s)	0.038	0.95	0.32	0.052
Average Number of Nodes	3.5	63	33.9	4.2
Required Network Knowledge	100%	18%	34%	39%

5 Conclusion

Three routing algorithms were created and adapted for use in intersatellite communications with the goal of finding networking methods that could be effectively used in a dynamic environment with limited computing power. The performance of these algorithms was compared to Dijkstra's shortest path algorithm, and transmission path distance, delay and number of nodes for each network algorithm were found.

It was found that the grouping algorithm, which grouped satellites by their physical position in the network, is nearly as effective as Dijkstra and requires less knowledge of the satellite network and less decision-making by the satellites. The chaos method, which involved sending transmissions to a satellite selected at random from the message source's accessible neighbours, is appropriate for smaller networks, low speed networks, or networks that have little computing power, but it can be improved on with little extra network knowledge or required computing by simply noting which satellites the message has passed through and sending to an unused satellite, as known path history method did.

This research will potentially assist in better intersatellite networking by offering solutions for message routing in various circumstances.

6 Future work

Future work in this area could include handling multiple transmissions at the same time, and the response of algorithms to the changing network due to satellite movement or failures in the network.

The algorithm should have some protocol for occasions when multiple different transmissions pass through the network at the same time. Collisions will occur when multiple users or multiple other nodes attempt to communicate with the same satellite at the same time. Thus, the algorithm should chose which, if any, user or node to communicate with, and have some alternate solution for the other transmissions.

The satellites are constantly moving which causes the network geometry to be continually changing. Thus, it is possible that the source satellite takes too long to make a decision and, through relative movement of the satellites, the potential path is lost. Additionally, if a large packet of data is sent, a path that lasts long enough for the duration of transmission will need to be found, or the data packet will have to be sent via multiple routes. Thus, optimizing paths for length of existence or modifying an algorithm to find multiple paths should be considered.

Failures in the satellite network should also be planned for. Algorithms should be able to adapt to network nodes failing, or not appearing where or when expected. An algorithm adaptable to failures will ensure a transmission reaches its destination regardless of unexpected events along the way.

References

[1] McKenzie-Picot, S. and Gavigan, P. (2013), Communication Link Analysis for a Nanosatellite Constellation in Low Earth Orbit, (Technical Report DRDC Ottawa TM 2013-068) DRDC.

[2] Clare, L., Wang, C. Y., and Atkinson, M. W. (1987), Multiple Satellite Networks: Performance Evaluation via Simulation, *MILCOM 87: IEEE Military Communications Conference*.

[3] Sturza, M. A., The Teledesic Satellite System (online), Teledesic Corporation, `http://www.iet.unipi.it/f.giannetti/documenti/60ghz/Fonts/THE%20TELEDESIC%20SATELLITE%20SYSTEM.htm` (Access Date: August 2013).

[4] Stenger, D. K. (1996), Survivability Analysis of The Iridium Low Earth Orbit Satellite Network, Master's thesis, Air Force Institute of Technology, Wright-Patterson Air Force Base, Ohio.

[5] Cain, J. B., Adams, S. L., Noakes, M. D., Kryst, T., and Althouse, E. L. (1987), A 'Near-Optimum' Multiple Path Routing Algorithm for Space-Based SDI Networks, *MILCOM 87: IEEE Military Communications Conference*.

[6] Chakraborty, D. (1989), Survivable Communication Concept Via Multiple Low Earth-Orbitng Satellites, *IEEE Transactions on Aerospace and Electronic Systems*, 25(6).

[7] Pratt, S. R., Raines, R. A., Fossa Jr, C. E., and Temple, M. A. (1999), An Operational and Performance Overview of the IRIDIUM Low Earth Orbit Satellite System, *IEEE Communications Surveys*, 2(2).

[8] Wang, X., Dijkstra Shortest Path Routing (online), `http://www.mathworks.com/matlabcentral/fileexchange/5550-dijkstra-shortest-path-routing` (Access Date: July 2013).

[9] Nagib, G. and Ali, W. G. (2010), Network Routing Protocol using Genetic Algorithms, *International Journal of Electrical and Computer Sciences*, 10(02).

This page intentionally left blank.

DOCUMENT CONTROL DATA

(Security markings for the title, abstract and indexing annotation must be entered when the document is Classified or Designated.)

1. ORIGINATOR (The name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Centre sponsoring a contractor's report, or tasking agency, are entered in section 8.)

Defence Research and Development Canada – Ottawa
3701 Carling Avenue, Ottawa ON K1A 0Z4, Canada

2a. SECURITY MARKING (Overall security marking of the document, including supplemental markings if applicable.)

UNCLASSIFIED

2b. CONTROLLED GOODS

(NON-CONTROLLED GOODS)
DMC A
REVIEW: GCEC APRIL 2011

3. TITLE (The complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S, C or U) in parentheses after the title.)

Routing Algorithm Analysis for a Nanosatellite Constellation in Low Earth Orbit

4. AUTHORS (Last name, followed by initials – ranks, titles, etc. not to be used.)

McKenzie-Picot, S.; Gavigan, P.

5. DATE OF PUBLICATION (Month and year of publication of document.)

October 2013

6a. NO. OF PAGES (Total containing information. Include Annexes, Appendices, etc.)

32

6b. NO. OF REFS (Total cited in document.)

9

7. DESCRIPTIVE NOTES (The category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

Technical Memorandum

8. SPONSORING ACTIVITY (The name of the department project office or laboratory sponsoring the research and development – include address.)

Defence Research and Development Canada – Ottawa
3701 Carling Avenue, Ottawa ON K1A 0Z4, Canada

9a. PROJECT OR GRANT NO. (If appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant.)

15ET03

9b. CONTRACT NO. (If appropriate, the applicable number under which the document was written.)

10a. ORIGINATOR'S DOCUMENT NUMBER (The official document number by which the document is identified by the originating activity. This number must be unique to this document.)

DRDC Ottawa TM 2013-081

10b. OTHER DOCUMENT NO(s). (Any other numbers which may be assigned this document either by the originator or by the sponsor.)

11. DOCUMENT AVAILABILITY (Any limitations on further dissemination of the document, other than those imposed by security classification.)

(X) Unlimited distribution
(   ) Defence departments and defence contractors; further distribution only as approved
(   ) Defence departments and Canadian defence contractors; further distribution only as approved
(   ) Government departments and agencies; further distribution only as approved
(   ) Defence departments; further distribution only as approved
(   ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (Any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (11). However, where further distribution (beyond the audience specified in (11)) is possible, a wider announcement audience may be selected.)

13. ABSTRACT (A brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual.)

This paper examines possible routing algorithms in a proposed network of nanosatellites with the goal of finding networking methods that could be effectively used in a dynamic environment with limited computing power. Three routing algorithms were created and their performance was compared to Dijkstra's shortest path algorithm. The chaos method transmitted to accessible satellites at random. The minimum distance method transmitted to the closest unused satellite. The grouping method grouped satellites by their physical position in the network and made transmission decisions based on these groups. Each algorithm was run for every possible combination of source and destination satellite in the network to find average transmission times, distances, and number of network nodes used. It was found that the grouping method had performance comparable to Dijkstra's algorithm while requiring less computing power, and was acceptable for high-speed network requirements, such as live voice. The chaos method was acceptable for lower speed applications and for satellites that had little computing power, and the minimum distance method improved on chaos method with little extra computing power required.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (Technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus. e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

Routing
Networking
Network Algorithms
Constellations
Nanosatellites
Small Satellites