International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2020/195673 Volume 7, Issue 2, March – April (2020)

RESEARCH ARTICLE

Partial Topology-Aware Data Distribution within Large Unmanned Surface Vehicle Teams

Agyemang Isaac Osei
School of Information and Communication Engineering, University of Electronic Science and Technology of China, China.
agyemangisaac@gmail.com

Adjei-Mensah Isaac
School of Information and Software Engineering, University of Electronic Science and Technology of China, China.
iadjeimensah@gmail.com

Published online: 30 April 2020

Abstract – In a distributed team of Unmanned Surface Vehicles (USV), distributing data to attain full coverage but not to overwhelm the network is a peculiar problem as in most cases USV's within the overlay network formed have little knowledge about the network. This is associated to decision theoretical problems thus NEXPTIME. This paper presents a novel data distribution approach based on Distributed Hash Table (DHT). The proposed mechanism is partially proactive, USV's relay data to its neighbors by selecting a neighbor with the closest hash to the data hash. The data distribution is decomposed into three stages: initial data distribution, data request, and data forwarding. The decomposition leads to a very simple but effective and efficient data distribution mechanism. Presentation of the details of the proposed data distribution algorithm is presented in respective sections of this paper with further discussions and simulated results which depict that the proposed data distribution algorithm is effective and efficient in terms of scalability, adaptability, coverage, latency, and redundancy when compared with some existing data distribution algorithms.

Index Terms – Data Distribution, Unmanned Surface Vehicle (USV), Hashing, Request/Response.

1. INTRODUCTION

Data distribution among Unmanned Surface Vehicles (USV) herein DD-USV is a promising but difficult problem in large teams of USV's. In general, much attention has been channeled to data distribution within a networked team due to its widespread use in applications such as traffic flow control and monitoring, disaster response, search and rescue, security and surveillance, and activity recognition [1,2,3,4,5]. Data distribution among networked teams has remained an active research field, especially in unmanned systems due to both challenges and opportunities it presents. The data distribution problem can be categorized into receiving information, knowing or not knowing the content of the information and distributing the information within the network based on a policy. Some of the difficulties among the myriad data distribution problems are occlusion within the network, less in-depth knowledge of network, how to infer and sense the topology [1] and the dynamic state of the network. Data distribution problem has been studied in numerous research fields yielding many forms of application. In the field of robotics and networks, the goal of the majority of researchers is the understanding of data distribution among teammates as they cooperate to perform a variety of tasks in the real world [6]. The last few years went by has seen the importance of data distribution in unmanned surface vehicles and in general robotic systems as it's been used to solve many problems of humanity [5,7].

The underlying complexities attached to the network within which USV's find themselves has led to the keen interest in using diverse techniques to quench the data distribution problem. However, the current state of the art of USV's pose challenges to the existing data distribution approaches. Unmanned Surface Vehicles has onboard lightweight processing units, limited storage capacity and short endurance time which does not make the existing approaches effective and efficient with regards to data distribution due to the associated complexities, storage and bandwidth consumption and redundancy characteristics of these approaches [8]. In addition, the dynamic state of the overlay network formed by the mobility features of USV's also contributes to the challenges existing approaches encounters.

In this paper we put forward together a novel data distribution based on the combination of Distributed Hash Table and the degree attribute of complex networks. The proposed mechanism partially depends on querying as some of the USV's in the network proactively relay gained data to others by finding the closest hash a USV has to the data hash or making use of USV's with highest degrees based on a USV's local knowledge of the network. From this viewpoint, the distribution approach is decomposed into three phases: initial data distribution which is proactive, data request which partially depends on querying and data forwarding. Based on these three connected phases, a reliable, light-weight, and efficient data distribution is proposed where by each USV within the network makes a comparison between the hash values of the data and that of its neighbors or between the degree values associated to the USV's before transmitting data.

To manifest the feasibility of the proposed mechanism, the authors of this paper presents and discuss the experiment carried out, and the results depicts the proposed mechanism of data distribution scales well, it is adaptable, achieves full coverage of distribution, efficient with regards to latency and completely eliminates redundant transmission of data.

The rest of the paper is organized as follows. Section 2 entails relevant prior research conducted within the scope of data distribution among large teams followed with Section 3, which entails the problem definition. Section 4 present the proposed data distribution algorithm. Section 5 delineates the setup and the experiment conducted with concluding experimental results. Irrevocably, the conclusion drawn on the work as well as feasible future works is addressed in Section 6.

2. RELATED WORK

Over the past years, data distribution within large networks has experienced decentralized and centralized algorithm utilization. In dynamic environments, the importance of data distribution comes to play as it enables robots to have a comprehensive view of the environment [1,9,10,11].

In classical Flooding, a node within a network saves received data and forward ones to all connected nodes, with the exception of the source the data came from [12]. Although classical flooding is able to attain the goal of distributing data to all parts of the network, it comes with a huge cost as the algorithm does not take into account any measures to control re-broadcasting of data within the network [13,14]. This drawback leads to congestion in the network [3]. In mitigating the drawbacks of the classical algorithm is the probability-based method. In probability-based methods, for a node to distribute data, the node calculates its contribution to the broadcasting process whether it affects the network in a positive or negative manner. If the nodes contribution is positive, that is if the contribution is greater than the given threshold, the data is retransmitted and on the other hand not retransmitted if the contribution false short of the threshold. Irrespective of the reduction in redundant transmission, a probability-based method for data distribution contributes to data relaying problems, it does not solve the problem completely [4]. The works of [15] present Delayed Flooding with Cumulative Neighborhood (DFCN) algorithm. DFCN has its emphasis based on four assumptions for data distribution which is: (a) knowledge of 1 hop neighbor which is obtained through a hello packet. (b) the message to be relayed has in its header the most recent sender ID which is a member of the 1-hop neighborhood. (c) all nodes maintain a piece of local information for each message received which comprises of the ID of the message and ID's of nodes that have received the message, plus an instruction to forward or not to forward the message. (d) a Random Assessment Delay (RAD) which prevents packet collisions. In DFCN, RAD's are set to message for re-broadcasting when received by nodes as a control measure to the redundant retransmission. Although the protocol in some way controls the emission of redundant data transmission, it does not fully attain the goal of proper data distribution and the complexity surrounding DFCN is high in the sense that, the parameters attached to the message contributes to higher consumption of bandwidth during data transmission. This gets worse when the network size increases and the data have been received by many nodes.

In Scalable Broadcast Algorithm (SBA) [16,17,18], each node keeps 2-hop neighbor information which is obtained through a "hello" packet. After subtracting the embedded ID's in the packet from known ID's of its neighbors, the data is dropped if the result is empty, else retransmitted. The basic principle in SBA is, nodes with higher degree broadcast ahead of nodes with a lesser degree as the nodes with high degree reach more nodes. The drawback associated with SBA is that it is not proactive but reactive. Another data distribution algorithm which is similar to SBA is that of Dominate Pruning (DP) [15]. In DP, unlike SBA, nodes assume to have knowledge of their 2-hop neighbors. When a node receives a packet, it proactively from its 1-hop neighbors chooses some or all to serve as receiving nodes for the rebroadcast. The criteria for choosing a receiving node is based on the inclusion or exclusion of nodes that are found in the header of the packet received. Despite the fact that DP is both reactive and proactive, the computation to select receiving nodes is huge and even gets worse if the network is dense. This slows down the time for data propagation within the network. In Multipoint Relaying, the selected nodes (MPR's) by the upstream sender serves as the forwarding nodes. From time to time, the nodes transmit a "hello" packet in which MPR's are embedded. Based on the arrival of a packet, if a node is an MPR, it retransmits the packet based on the instruction accompanied by the packet. Continues transmission of "hello" packets affect the network in terms of bandwidth consumption and latency. As time goes on, the metadata competes with the actual data for with regards to bandwidth [19,20]. In the works of [7], robots keep track of their local information for distribution purposes. The subject-based addressing approach breaks the mobile robot network into anonymous data producers and consumers. Robots which produces data appends some description to the content of the data and publishes it on the network for subscribers to consume. Robots that subscribes late after data has been published does not stand a chance of receiving the data. In Sensor Protocols

International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2020/195673 Volume 7, Issue 2, March – April (2020)

for Information via Negotiation (SPIN) [9], nodes advertise (ADV) to its neighbors upon arrival of new data. Neighbor nodes verify the ADV message through the Message Authentication Code (MAC). After successful verification, the receiver sends a request message to the source. The source verifies the request and transmits requested data to the destination node after successful verification. Despite the great contribution, SPIN has in the name of relaying information in energy-constrained networks as well as solving problems of overlapping and implosion other algorithms presents, SPIN has some drawbacks as well. SPIN does not guarantee arrival of data to the destination node and it is not suitable for high-density networks. Also, if nodes between data source nodes are not interested in data whiles nodes that are interested in data are situated far from the source node, the source node does not stand any chance of transmitting data to the node which is interested in the said data.

The aforementioned review of literature confirms the importance of data distribution within a large team distributed system, discusses specific solutions in such respective and further presents problems regarding the distribution of data. In contribution and as an improvement and an alternative to data distribution algorithms, hashing technique is adopted [21, 22,23,24]. The authors of this paper present a novel data distribution algorithm DD-DHT based on Distributed Hash Table and the degree attribute of sparse networks. DD-DHT ensures data delivery, low latency and also is characterized by low computation complexity hence it does not consume much energy.

3. PROBLEM DESCRIPTION

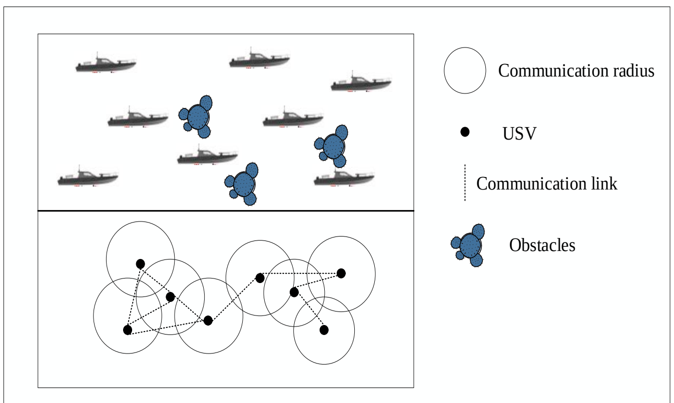

Figure 1 An Overlook of Deployed USV's and its Overlay Networked Formed based on Positions

Figure 1 depicts a typical scenario of Unmanned Surface Vehicles (USV) deployed on the sea to collect data and detect smuggling activities and Figure 2 shows at the actual simulation of the scenario. When a teammate obtains information, the distribution problem is among its neighbors, which neighbor it chooses and send the information to. Due to the dynamic feature of the USV's which affects the topology of the overlay network formed by the positions of the USV's, correct decisions have to be made to eliminate redundant data distribution in order not to overwhelm the network and ensure the information reaches each USV in the network.

To be precise, the key problem is summarized as follows:

A set of USV's are defined as $R = \{r_1, r_2, r_3 \ldots r_j\}$ and the network formed by the USVs as $N(t) = \bigcup_{r \in R} n(r,t)$ where each $n(r,t)$ defines a USV which can issue a request, receive a request, send data or receive data at a time t. A USV R_1 needs more environmental awareness information M^1, but it does not know which USV that has M^1 and it does not know where the information M^1 is. A USV R_2, obtains environmental cognition information M^2, but does not know which USV's need the information M^2, and does not know when and where these USV's need the information M^2. The information M^n is enclosed as a data package and its data structure is defined as $M^n = < msgHashID, flag, path, content, type >$. In the data structure, $msgHashID$ represents the information hash value, $flag$ denotes a true or false state regarding the information M^n,$path$ records the trail the information M^n has traveled, the $content$ holds the actual information and the $type$ represents the type of data package. The objective function is to distribute data to each and every USV within the network such that USV's can have full view of the environment and also eliminate redundant transmission completely. Through DHT, the authors of this paper propose a novel data distribution approach to solve the distribution problem.

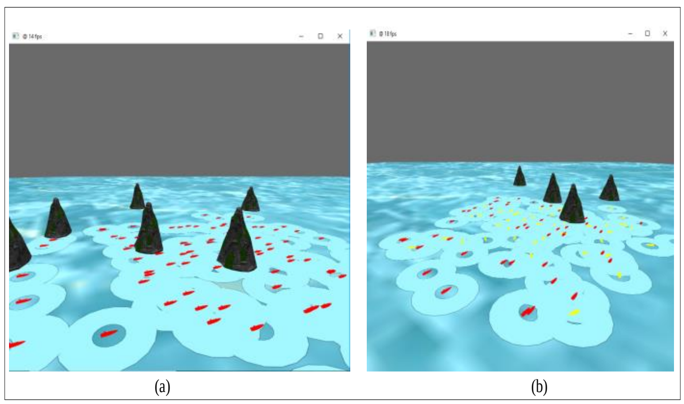

Figure 2 Schematic of Deployed USV's at Sea (a) Represents Simulation of USV's in a Connected Network in static mode (b) Represents Simulation of USV's in Dynamic Mode, in Here All Moving USV's are Indicated by Yellow Color. Sea Mountains Impeach Communication Between USV's.

4. DATA DISTRIBUTION ALGORITHM

DHT's relies on a decentralized distributed approach to provide faster lookup services to data for efficient and faster retrieval of associated data. Through DHT, hash values are assigned to both the USV's and generated data. Precondition: two kinds of data packages namely $requestPackage$ and $responsePackage$ are defined. DD-DHT algorithm is in three phases namely initial data distribution, data request, and data forwarding. Since the focus is not on cryptography, a hash function known as xxHash which evaluates collision, dispersion and randomness qualities of hash function is adopted. Data is relayed by means of comparison to find the closest hash of a neighbor USV to the hash value of the data or a neighbor with the highest degree in terms of connection. The proposed data distribution algorithm, DD-DHT, can run from the initial data distribution phase through to the data forwarding phase or from the data request phase to the data forwarding phase depending on the state of the USV which is depicted in Figure 3. In some instances, the data forwarding phase may run alone if the USV in context has the data at hand and has received request for the data.

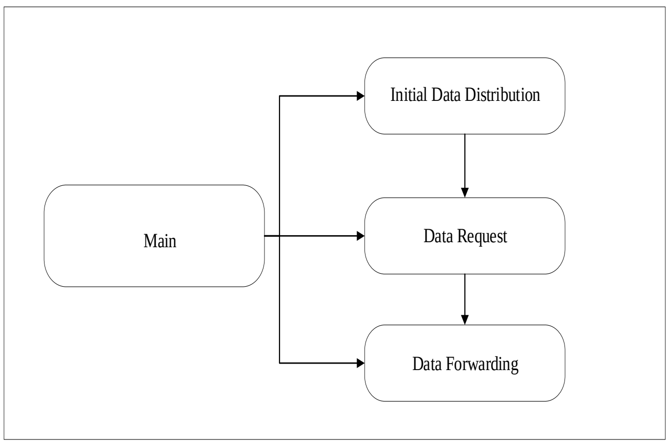

Figure 3 Data Distribution Algorithm Architecture

4.1. Initial Data Distribution

At the initial phase of Algorithm 1, each USV sends a hello message to its neighbors together with its degree to indicate its presence online. With this information, each USV gets a fair detail of its neighbor. At this point in time, the source USV compares its hash with the hash value of the data obtained, compares each of its neighbor's hash with the hash value of the data as well and forwards the data to one of the neighbors based on the following conditions:

- If the source USV hash has the least hash distance to the data hash, then it would choose and forward the data to the neighbor with the highest degree.
- If among the source USVs' neighbors, there is a neighbor whose hash is closer to the data in terms of hash, than any other USV with the source USV inclusive, then the source USV would send the data to the USV with the closest hash value regardless of the degree of the USV in context.

Begin:

1. Cache the packet
2. Compare the hash distance of the neighbor USV's, the current USV excluding the path-USVs and the current data packet hash. Get USV V_i with the smallest distance.
3. If (V_i is not the current USV && status of flag==False)
4. Set packet flag to true.
5. Add the current USV to the path tail.
6. Forward packet to V_i.
7. End if.
8. If (V_i is the current USV || status of flag == True)
9. If (V_i is the current USV with data && V_i has the least hash value to the message hash)
10. Get neighbor USV V_j with max degree excluding the path USV's
11. End if
12. End if
13. If (V_j != null)
14. Set packet flag to true
15. Add the current USV V_j to the path-USV tail
16. Forward packet to V_j.
17. End if

End

Algorithm 1 Initial Data Distribution

4.2. Data Request

The second phase of the data distribution mechanism, Algorithm 2 starts when neighbors of a USV with the data are all part of the USV-path tail. Here, USV's without the data issues a data request package to one of its neighbors. The choosing of a neighbor a USV sends a request to is done as it is done in the initial data distribution phase. If a USV receives a request and it does not have the data, it is triggered to requests data from one of its neighbors. This continues until the data request reaches a USV with the data.

Begin:

1. For each USV

2. Check for request.
3. If (request is available)
4. Check cache for the package.
5. If (package is available)
6. Generate a response package.
7. Get the last USV V_j from the request path.
8. Forward package to USV V_j.
9. Delete USV V_j from the request path.
10. else
11. Choose a neighbor USV V_j from neighbors as done in the initial data distribution phase.
12. Add current USV V_i to the tail of the request path package.
13. Forward request package to chosen USV, V_j.
14. End if
15. else
16. Check for the package.
17. End if
18. If (package is available)
19. Do nothing.
20. else
21. Choose a neighbor USV V_j from neighbors as done in the initial data distribution phase.
22. Add current USV V_i to the tail of the request path package.
23. Forward request package to chosen USV, V_j.
24. End if

End

Algorithm 2 Data Request

After the completion of data request phase, USV's without data would have issued a request to appropriate USVs and awaits the response. During the request phase, if a USV gets a request and it has the data, it responds back with the data and does not wait for the data forwarding phase before responding to all requests received.

4.3. Data Forwarding

In Algorithm 3, USVs that have received request but does not have data retransmits data to requested USV's ones the data becomes available.

Begin:

1. For each USV
2. Check if current USV V_i has received a request
3. If (request received== true)
4. Check cache for request response
5. If (response is available in cache)
6. Get USV V_j from USV V_i request info
7. Delete USV V_j from request path info
8. Forward package to USV V_j
9. else
10. Do nothing
11. End if
12. else
13. Do nothing.
14. End if

End

Algorithm 3 Data Forwarding

5. EXPERIMENT AND RESULTS

Experiment is conducted to evaluate and analyze the performance of the proposed algorithm, DD-DHT, alongside with some competing algorithms for data distribution in a large team of Unmanned Surface Vehicles (USV) deployed on the sea for data collection to search and detect smuggling activities. The experiment is conducted in two folds; static and dynamic topology.

A maritime obstacle populated field measured in 2200 x 1100m2 with a team size comprising of 500 and 1000 USV's respectively is used. In each experiment, USV's are randomly distributed in a fully connected network. The obstacles within the environment impeach perfect communication among USV's, hence affects relaying of data which renders USV's to have a partial observation of the network. Each USV is equipped with a communication radar which can be varied based on preferences. In each run, a communication radius of 40meters is adopted, hence any USV within this radius becomes a neighbor of a USV in context.

In the second criteria of the experiment, dynamic mode, a ratio of 10% and 40% of the USV's is set to locomote. The criteria for a USV's to be part of this ratio is random. In this mode, any USV that is part of the ratio calculates a new position within a 40meter radius and moves to the new position. Moving to different locations continues until the simulation is over. In each run at timestep 30, the selected

International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2020/195673 Volume 7, Issue 2, March – April (2020)

RESEARCH ARTICLE

data distribution algorithm is triggered to commence data distribution with a random USV serving as the source with the data.

In here, the comparison lies with Scalable Broadcast Algorithm (SBA), Flooding and Sensor Protocol for Information via Negotiation (SPIN). In flooding, each USV rebroadcast data to all of its neighbors except the source the data came from without taking into consideration the lethal parameters of the USV's as well as network congestion [15]. For SBA, USV's acquire a 2-hop neighbor knowledge through a "hello" packet. After subtracting the embedded ID's in the packet from known ID's of its neighbors, the data is dropped if the result is empty, else retransmitted [5,6]. In that of SPIN, a 3-way handshake which is aimed at verification occurs after a request for data has been carried out before transmission of data [15]. In SPIN data distribution algorithm, the rate of interest in the data is set to 90% to mimic the characteristic of the algorithm.

In order to evaluate the performance of DD-DHT, the time taken to complete distribution is considered, the number of redundant receptions and the delivery ratio or coverage are also considered. In the first batch of the experiment, DD-DHT is evaluated in a static mode using the default parameters discussed earlier and compared with the other three algorithms. Afterward, the following parameters are adjusted: team size, number of obstacles and field size and perform the experiment again to evaluate the performance of DD-DHT and make a comparison to the other algorithms. In the second batch of the experiment, evaluation of the performance of the algorithms alongside DD-DHT in the dynamic network is conducted.

5.1. Benchmarking

Since SPIN, SBA and Flooding [10,17,18] can be classified under neighbor-based methods and area-based methods for data distribution and has been used for distribution in static networks, a familiar benchmark which runs through these algorithms is adopted. The same benchmarks are used in the dynamic network experiment as well. The evaluation criteria are as follows:

- Time taken: this is the total time taken for data to be distributed within the network. The smaller the time, the better the success the algorithm achieves. This is expressed as: $Time\_taken = Endtime - Start\_time$.
- Delivery ratio or Coverage: this measures the number of USV's with the transmitted data. A threshold of 90% of the team size is used. At the end of the distribution, if the percentage of USV's meets the threshold, then the algorithm passes this benchmark. The computation is expressed as: $$Coverage = \frac{USV\_with\_data * 100}{Team\_size}$$

Redundant receptions: at the end of each run, the excess data in the network denotes the redundant data. This is expressed computationally as: $\text{Redundant} = (\text{Total\_data}) - (\text{Team\_size})$.

5.2. Results in Static Network

In the first run, a team size of 500 USV's in a field of 2000 x 1000m^2 which is populated with 20 sea mountains as obstacles is used. All other parameters are set to default as described in the experiment setup.

In Figure 4a, DD-DHT takes a very short time to complete distribution as compared to the other three algorithms due to the faster lookup of data which DHT uses in the search of data. SBA has slightly better time to that of SPIN, as in SPIN data is transmitted after back and forth varication of requests which contributes to delay in data transmission. Flooding has the worst time to complete distribution as each and every USV transmits data upon arrival of data.

Figure 4b denotes the data coverage within the network. DD-DHT and Flooding achieve full distribution coverage but Flooding is accompanied by a huge cost. Although DD-DHT is based on request/response, data is able to reach all USV's because if a USV receives a request and it does not have the data and at the same time not interested in the data, it does not abandon the request as it is experienced in SPIN but rather retransmit a request to another USV. SBA achieves comparatively even data coverage as that of SPIN, as the only deficiency in SBA in terms of coverage is when data is received from USV's with the higher degree to USV's with a lesser degree, data is not retransmitted per the relaying mechanism of SBA.

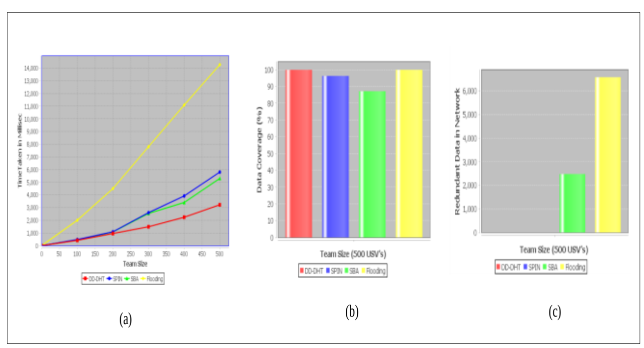

Figure 4. Experimental results in a static network with a team of 500 USV's in a field measured 2000 x 1000m^2 and 20 sea mountain obstacles. (a) Presents the time taken to complete data distribution; (b) presents the data coverage within the network; (c) presents the redundant data within the network after data distribution with the exclusion of meta-data.

RESEARCH ARTICLE

In Figure 4c, redundant data associated with the uncontrollable data coverage in flooding is manifested. Excluding meta-data transmission of data, DD-DHT and SPIN eliminate redundant data transmission completely as both are based on the principle of request/response. SBA, on the other hand, achieves a better elimination of redundant data than that of flooding.

To test the scalability and adaptability of DD-DHT algorithm and the other three algorithms, some parameters of the experiment are altered and simulated ones more. Here, the team size is increased to 1000, obstacles are increased from 20 to 30 sea mountains and slightly re-adjust the field size not in proportional to the increase in the team size. The field size is set to 2200 x 1100m^2. Any other parameter remains in its default state as described in the experiment setup section.

The change in parameters slightly, averagely and hugely affects the algorithms in the various benchmarks. In Figure 5a with regards to time, DD-DHT experiences a slight impact, whereas SBA and SPIN are impacted averagely. The worst case is experienced by the flooding due to the increase in team size. As the team size grows, flooding experiences a polynomial runtime whereas the other three algorithms including DD-DHT continues in a logarithmic form. For data coverage as per Figure 5b, DD-DHT and Flooding are not influenced by the change in the experiment parameters. Changes are noticed in SPIN and SBA. A similar conclusion is drawn for data coverage irrespective of the changes in the parameters of the experiment setup. On the part of redundant reception as presented in Figure 5c, there is a great influence on flooding, as the team size increases the rate of redundant transmission increases by a factor of 3. DD-DHT and SPIN do not experience any redundancy as data propagation is based on the principle of request /response. SBA experiences a slight increase in redundant propagation.

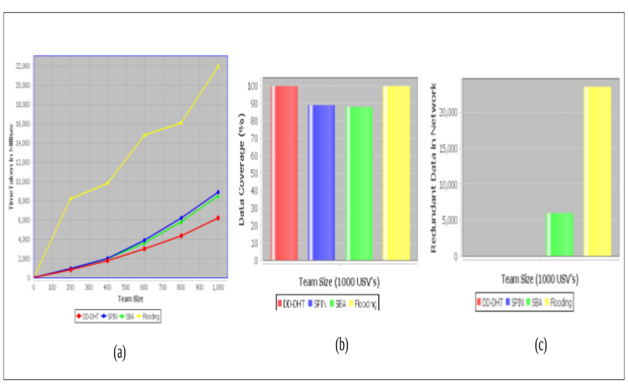

Figure 5. Experimental results in a static network with a team of 1000 USV's in a field measured 2200 x 1100m^2 and 30 obstacles. (a) Presents the time taken to complete data distribution; (b) Presents the data coverage within the network; (c) presents the redundant data within the network after data distribution.

From the results presented, it's obvious that DD-DHT and SPIN scales better but DD-DHT has a slight upper hand over SPIN. SBA and Flooding perform well in some jurisdiction but not suitable for large team size.

5.3. Results in Dynamic Network

In the second batch of the experiment, exploration of the influence of the dynamic topology on the algorithms are prioritized. The USV's are set into motion. In here, USV's that moves, have a constant speed of 3/system iterations. The ratio of USV's that is set to move is set at 10% and 40% for each run respectively. The movement is done as described in the experiment setup. In here, a team size of 500 USV's in a field of 2000 x 1000m^2 which is populated with 20 sea mountains as obstacles are the parameters. All other parameters are set to default as described in the experiment setup.

Figure 6a through to 6c and 6d through to 6f represents results for 10% and 40% of moving USV's respectively. Similar conclusions are drawn from the results in the static network for Figure 6a-c, although the USV's move, data travels faster than the moving USV, hence the impact of the 10% moving USV's affects the data distribution slightly from the results attained in the static mode.

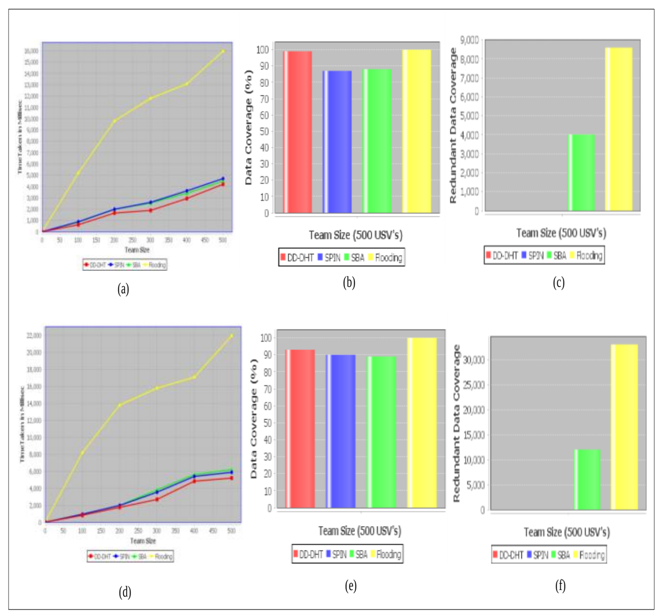

Figure 6. Experimental results in a dynamic network with a team of 500 USV's in a field measured 2000 x 1000m^2 and 20 obstacles. (a) and (d) presents the time taken to complete data distribution at a moving ratio of 10% and 40% respectively; (b) and (e) presents the data coverage within the network at a moving ratio of 10% and 40% respectively; (c) and (f) presents the redundant data within the network after data distribution at a moving ratio of 10% and 40% respectively.

RESEARCH ARTICLE

From Figure 6d through to 6f, when 40% moved, SBA takes slightly lesser time to complete distribution than the SPIN. The proposed algorithm DD-DHT still takes lesser time to complete distribution among the three algorithms. DD-DHT experiences a minor decrease in terms of data coverage from Figure 6e, this can be attributed to the moving USV's. A similar reduction is experienced in SBA and SPIN in Figure 6d and 6e respectively. Flooding does not experience any sort of reduction in terms of coverage irrespective of the movement of the USV's. For redundant reception of data, DD-DHT and SPIN eliminate redundancy based on request/response distribution mechanism. The redundancy in flooding grows exponentially due to the moving USV's. As USV's move, they transmit data to new neighbors. SBA experiences a fair share of the increase in redundant data, as the USV's move, they obtain different degrees which influence the distribution of data in SBA.

6. CONCLUSION

Data distribution is a critical aspect in the future of networks considering domains such as reconnaissance, target tracking, disaster response, and other relevant operations. In spite of this, some existing data distribution algorithms for USVs are no longer feasible due to the lightweight nature of the current USVs and the high computation complexities and scalability issues associated with these algorithms. The authors of this paper have presented a new data distribution algorithm, DD-DHT which utilizes DHT and the degree attribute of sparse networks. The break-up of the proposed algorithm into three phases aids USVs to reason and share data appropriately without keeping a complete knowledge of the network. The efficiency of the proposed algorithm DD-DHT has been demonstrated through an experiment and compared with well-known algorithms which include SPIN, SBA and Flooding for data distribution within a network. The resulting performance shows the proposed algorithm DD-DHT has an upper hand over SPIN, SBA and the Flooding algorithm and manifested that the proposed algorithm is scalable, lightweight and effectively eliminates redundant transmission of data, attains full coverage within the network and takes a short time for completion of data distribution. This is important to the network as the elimination of redundant transmission decongest the network. Again, due to the lightweight processor aboard the current USVs deployed in distributed systems, the minimum computation routes DD-DHT algorithm uses for distribution would not drain much energy of the USVs when deployed.

REFERENCES

[1] Y. Xu, X. Hu, Y. Li, D. Li, M. Yang. Using complex network effects for communication decisions in large multi-robot teams. "AAMAS'14", 685-692, 2014.
[2] O. Amir, B.J. Grosz, and K.Z. Gajos. MIP-Nets: Enabling Information Sharing in Loosely-Coupled Teamwork. In AAAI, 2016.
[3] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu. The broadcast storm problem in a mobile ad hoc network. Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking, pages 151–162, 1999.
[4] R. Zhi, L. Pengxiang, J. Fang, L. Hongbin, Q. Chen. SBA: An Efficient Algorithm for Address Assignment in ZigBee Networks. Wireless Pers Commun. 71: 719-734, 2013.
[5] V.V. Unhelkar, and J.A. Shah. "Enabling Effective Information Sharing in Human-Robot Teams", Robotics: Science and Systems (RSS), Workshop on RSS Pioneers, 2018.
[6] L. E. Parker, "ALLIANCE: An architecture for fault-tolerant multi-robot cooperation," IEEE Trans. Robot. Automat., vol. 14, pp. 220–240, Apr.1998.
[7] L. Zhu, Y. Xu, P. Scerri and H. Liang. An Information Sharing Algorithm for Large Dynamic.
[8] B. Vergouw, H. Nagel, G. Bondt, and B. Custers. "Drone Technology: Types, Payloads, Applications, Frequency Spectrum Issues, and Future Developments" in The Future of Drone Use. Springer, pp.21–45, 2016.
[9] S. Barrett, N. Agmon, N. Hazon, S. Kraus, and P. Stone. Communicating with Unknown Teammates. In Proc. ECAI, 2014.
[10] B. Gerkey, and M.J. Mataric. Principled communication for dynamic multi-robot task allocation. In Experimental Robotics, Vol. VII of LNCIS 271, D. Rus and S. Singh (Eds.), Springer-Verlag: Berlin, Heidelberg, pp. 353-362, 2001.
[11] Mobile Multi-agent Teams. In International Conference on Autonomous Agents and Multi-Agent Systems, 2012.
[12] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu. The broadcast storm problem in a mobile ad hoc network. Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking, pages 151–162, 1999.
[13] J. Kulik, H. Balakrishnan, and W. R. Heinzelman. "Adaptive Protocols for Information Dissemination in Wireless Sensor Networks", Proceedings on the 5th annual ACM/IEEE International Conference on Mobile Computing and Networking, pp. 174-185, 1999.
[14] J. Wu, and F. Dai. Broadcasting in ad hoc networks based on self-pruning. IEEE INFOCOM, San Francisco, 2003.
[15] L. Hogie, P. Bouvry, M. Seredynski, and F. Guinand. "A Bandwidth-Efficient Broadcasting Protocol for Mobile Multi-hop Ad hoc Networks," International Conference on Systems, Mobile Communications, and Learning Technologies, pp.71–71, 2006.
[16] W. Peng and W. Lu. "On the reduction of broadcast redundancy in mobile ad hoc networks", Mobile and Ad Hoc Networking and Computing, 2002.
[17] R. Zhi, L. Pengxiang, J. Fang, L. Hongbin, Q. Chen. SBA: An Efficient Algorithm for Address Assignment in ZigBee Networks. Wireless Pers Commun. 71:719-734, 2013.
[18] B. Williams and T. Camp. Comparison of broadcasting techniques for mobile ad hoc networks. In Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC), pages 194–205, 2002.
[19] O. Liang, A. Sekercioglu, and N. Mani. "Gateway Multipoint Relays-anMPR-based Broadcast Algorithm for ad hoc Networks. IEEE, 1-4244-0411-8/06, 2006.
[20] Y. Ma, Y. Zhao, X. Qi, Y. Zheng and R. Gan. Cooperative communication framework design for the unmanned aerial vehicles-unmanned surface vehicles formation. In Advances in Mechanical Engineering, 2018.
[21] V. E. Liong, J. Lu, G. Wang, P. Moulin, and J. Zhou. Deep hashing for compact binary codes learning. In CVPR. 2475–2483, 2015.
[22] D. Zhang, F. Wang, and S. Luo. Composite hashing with multiple information sources. In SIGIR. 225–234, 2011.
[23] L. Zhu, Z. Huang, X. Liu, H. Xiangnan, J. Sun, and Z. Xiaofang. Discrete multimodal hashing with canonical views for robust mobile landmark search. In TMM, 2017.

DOI: 10.22247/ijcna/2020/195673 Volume 7, Issue 2, March – April (2020)

RESEARCH ARTICLE

[24] Z. Peng-Fei, L. Chuan-Xiang, L. Meng-Yuan, N. Liqiang and X. Xin-Shun. Semi-relaxation supervised hashing for cross-modal retrieval. In MM.1762–1770, 2017.

Authors

Agyemang Isaac Osei received his B.Sc. from Christian Service University College, Ghana in 2016 and MEng in Computer Science Technology and Engineering from the University of Electronic Science and Technology of China (UESTC) in 2019. He worked as a research assistant at the multi-agent and robotics lab (MAR's Lab, UESTC) during his masters and severed as an intern at Intel Corporations, Chengdu-China and a back-end engineer with We-work organization, Chengdu-China. He is currently a Ph.D. student and a research assistant at the Xie Lab's (UESTC) at the School of Information and Communication Engineering, University of Electronic Science and Technology of China. His research areas include deep learning, unmanned systems, IoT, SAR and blockchain.

Adjei-Mensah Isaac received his B.Sc. from Valley View University, Ghana in 2017 and pursued his MEng in Software Engineering from 2018 at the University of Electronic Science and Technology of China. He is currently being considered as a Ph.D. candidate at the School of Information and Communication Engineering, University of Electronic Science and Technology of China. He is also a Network Engineer having pursued his CCNA in 2011 at the Kofi Annan Centre of Excellence, Ghana. His research areas include deep learning, IoT, Computer Networks, SAR and Blockchain Technology.