Studying the Effect of Network Latency on an Adaptive Coordinated Path Planning Algorithm for UAV Platoons

Baisakhi Chatterjee
North Carolina State University
Raleigh, USA
bchatte@ncsu.edu

Rudra Dutta
North Carolina State University
Raleigh, USA
rdutta@ncsu.edu

ABSTRACT

UAVs can be directed to perform multiple tasks efficiently without human intervention. This is especially useful in tasks that require group coordination since UAVs can exchange information and alter their trajectories based on dynamic data. Thus, communication in multi-UAV platoons, with the goal of mission-oriented trajectory planning, is an area of considerable interest. In this paper, we focus on message dependence in a group of UAVs in the course of completing a task. To this end, we chose the task of Plume Wrapping. Plume Wrapping is the problem of finding the shape and extent of hazardous airborne material. This is a practical problem in real world scenario which requires a fully autonomous UAV group. Communication between UAVs is essential to coordinate decisions. In this work, we show that an algorithm can be designed which ensures correctness while being robust to variable message delay.

CCS CONCEPTS

• **Networks** → *Mobile networks*; • **Computing methodologies** → Simulation evaluation.

ACM Reference Format:

Baisakhi Chatterjee and Rudra Dutta. 2022. Studying the Effect of Network Latency on an Adaptive Coordinated Path Planning Algorithm for UAV Platoons. In *Eighth Workshop on Micro Aerial Vehicle Networks, Systems, and Applications (DroNet '22), July 1, 2022, Portland, OR, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3539493.3539581

1 INTRODUCTION

Autonomous UAVs are efficient at performing a wide variety of tasks without human intervention. A group of UAVs can be utilized to coordinate with each other and complete a diverse set of missions, like search and rescue [17], network coverage [13], environment monitoring [2] and much more. As is evident, the common thread in tasks like these, is that they require adaptive trajectory planning based on dynamically generated data. Thus, coordination and communication by a group of autonomous UAVs is critical.

In this work, we aim to study the communication in a multi-UAV group in the pursuit of completion of a task. To do this, we chose a problem which requires a fully autonomous group of UAVs to successfully solve it. The mission we selected for this was Plume Wrapping. Plume wrapping is the problem of finding the shape and extent of hazardous airborne material, such as gas or toxic leak. We developed an algorithm that can be utilized by a UAV platoon to solve this problem. We then evaluated the dependence on information exchange and showed that a task-oriented adaptive algorithm can be developed which is robust to message delay.

2 LITERATURE SUMMARY

The problem of analysing and tracking a plume has been studied considerably in literature and formulated using various ways. Some of the earliest works used statistical models [5, 18] to evaluate the distribution of a leak. Others tried to gather the data from the environment using static receptors [1, 20]. Modern approaches to this problem utilized unmanned vehicles. In some cases only a single vehicle was used to traverse the specified site [4, 19], while in others, groups of vehicles were required to gather data [6–9, 12, 15]. Unfortunately, in most cases these works were restricted to either 2D environments [6, 15], or only considered source tracking [4].

A completely distributed co-operative path planning algorithm was proposed by John and Dutta [3] for wrapping a group of UAVs around a plume. In this paper, UAVs tried to wrap a plume by continuously moving towards or away from each other based on *attraction* and *repulsion* forces, a commonly used technique as seen in [14]. This results in a jagged movement where each UAV is forced to move towards the plume and away from the nearest neighbouring UAV till a uniform shape emerges. In this paper, we plan to utilize the ability of UAVs to enter potentially dangerous environments by having the UAVs map the plume boundary from inside out.

3 METHODOLOGY

The goal of our research is to study the dependence of information exchange among a group of autonomous UAVs for completing a mission. Additionally we wish to show that an algorithm can be made robust to bounded message delays when solving such a problem. Unfortunately it is not feasible to develop a generic algorithm which can complete *any* task. As such, we limited our analysis to one particular mission, which was Plume Wrapping.

The problem of tracking and surrounding a plume is a practically relevant one and has generated considerable academic interest. While there have been notable statistical evaluations, more modern works focus on utilizing autonomous vehicles, specifically UAVs, to try and solve this problem. As such, inter-UAV communication is critical. This makes it an excellent model for our goal.

Since our focus is on studying and analysing information exchange, it seemed prudent to simplify certain aspects of the problem.

DroNet '22, July 1, 2022, Portland, OR, USA

ACM ISBN 978-1-4503-9405-5/22/07…$15.00
https://doi.org/10.1145/3539493.3539581

We have, thus, considered the plume to be a sphere of arbitrary size with a monotonic concentration that decreases as we move closer to the boundary. We have also assumed that each UAV is fitted with a gaseous sensor that can accurately detect this concentration and communicate it to other members of the platoon. We say that a UAV has reached Plume boundary if the detected concentration lies between a pre-decided range. This area is said to be the boundary region. Each UAV is able to accurately detect its own position. A high level abstraction of communication medium has been also considered such that every UAV is able to send messages to every other group member. Based on this principle, messages may be delayed, but never lost.

3.1 Algorithm

With our assumption in mind, the problem of plume Wrapping can essentially be described as the task of placing *n* UAVs at or near the surface of the plume such that the UAV group completely surrounds the plume. There are many popular ways of doing this [16][10]. The Fibonacci spiral method [11] is one of the most well-known algorithms for this task. The idea behind this algorithm is to translate the 2D golden spiral into a 3D one. Points are placed along this spiral such that they cover the entire sphere, as is shown in Fig. 1. This method is fast, easy to implement and translates well to a distributed implementation due to use of unique identifiers for calculation of final position.

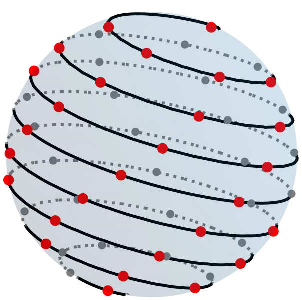


Figure 1: Points placed on a Fibonacci spiral

It is not enough, however, to simply place *n* UAVs, on the surface of a sphere. We need to know the center and radius of the plume as well. Only then can we map the Fibonacci spiral to the plume boundary. To do this we utilize two kinds of translations of the sphere – *Expansion* and *Rotation*. It is worth noting that, when we refer to translation of the Fibonacci sphere, we actually mean translation of the points representing each UAV, such that the overall position of the Fibonacci sphere is modified. Thus, *Expansion* refers to the movement of points such that the radius of the Fibonacci Sphere is increased. Similarly, *Rotation* refers to the movement of the UAVs such that the sphere pivots relative to a predefined anchor point in a particular direction.

The entire algorithm can be divided into four parts –

3.1.1 Initialization. Initially, the UAVs are placed at arbitrary points inside the plume boundary. Each UAV is assigned an ID number, which can uniquely identify the UAV. The UAVs are aware of the initial position of all agents in the group.

One UAV then begins moving in a predefined direction searching for the boundary of the plume. When the plume boundary, the UAV broadcasts its position to every member of the group.

3.1.2 Fibonacci Sphere Formation. Each UAV calculates its target position on an arbitrary sphere centered at the origin. Position is deterministically calculated based on the UAV ID. The sphere is then translated such that the anchor point can be placed on the surface of the sphere. This results in the UAV recomputing its target point for the modified sphere. Fig. 2 shows the expected movement of the imaginary Fibonacci sphere during translation. The last UAV to reach the target broadcasts a position update which directly leads to the next phase of the trajectory planning.

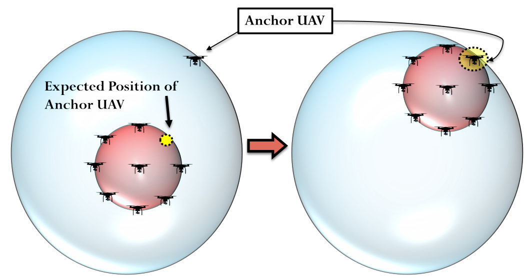


Figure 2: Translation of Initial Fibonacci Sphere

3.1.3 Stop-Click Stage. In this stage, each UAV evaluates the current position and density of every UAV in the group, including its own, and decides its future trajectory. This stage consists of two repeatedly alternating phases - Stop and Click. During the Stop Phase, each UAV broadcasts its current position, waits to receive a similar position update from every other UAV, and evaluates its next movement target. The Stop phase terminates when all UAVs are ready to move on to the Click phase. The UAVs start moving in Click phase. As shown in Fig. 3, there are two possible decisions each UAV can take here –

- **Expansion** - Expansion is selected when every UAV is inside the plume boundary. Each UAV increases the current sphere radius and calculates its target position using the Fibonacci Spiral algorithm.
- **Rotation** - Rotation occurs when at least one UAV is outside the plume. Direction is determined by the position of the UAV outside the plume boundary. The new center is calculated by calculating the point on the line joining the intermediary point and the anchor point such that the distance between anchor point and new center equals the desired radius. The UAV then computes its target position relative to the newly calculated center point.

The Stop-Click phased approach imposes a forced synchronization since each UAV must wait for every other member before beginning the next phase. This is necessary to ensure consistent choices by all UAVs; we return to this important point in the next section.

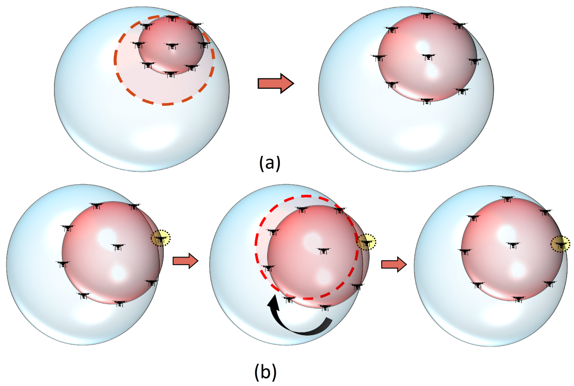


Figure 3: (a) Expansion (b) Rotation

3.1.4 Termination. Termination occurs when every UAV is at the boundary region. This is determined by an UAV staying at the same place for two successive Stop-Click rounds.

4 STUDY APPROACH AND RESULTS

Having defined the Plume Wrapping application and algorithm, we go on to study the dependence of the correctness and performance of such an application on UAV-to-UAV coordination messages. We developed a multi-agent simulation to study the interaction of a number of UAVs executing this algorithm, then varied some aspects of UAV-to-UAV communications in order to study the impact on Plume Wrapping performance criteria as described in the rest of this section. To this end, we designed a discrete event simulation and implemented it in Jave. In this implementation, an independent Engine is responsible for executing events from a list. Events are generated when any UAV performs an action. Events are of three types –

- **Broadcast Message Event –** A message is broadcast by a UAV detailing its current position and plume density.
- **Transmit Message Event –** The broadcast message is transmitted by the UAV.
- **Receive Message Event –** A message is received by an UAV from another member of the group. Once position updates from every UAV is received, the receiving UAV can plan its trajectory.

First, we note that we designed our algorithm to have minimal dependence on the variety of such coordination messages. Specifically, we only require that each UAV periodically broadcasts its location to every other UAV. (We make no specific restrictions as to whether this is accomplished by a single-medium broadcast, or any multi-hop mechanism such as flooding.)

Next, note that our algorithm design *already identifies and accommodates a crucial dependence on these position updates.* Specifically, the Stop-Click phased implementation is introduced to force coordinated state changes by each UAV. If all messages were delivered to all UAVs with zero delay (or the same constant delay), the Stop-Click phased approach would be unnecessary, and the algorithm could be simplified (and sped up) to allow each UAV to move to its next target point as soon as it received position updates from all other UAVs – the execution would be indistinguishable from a Stop-Click phased approach. However, we find that with variable delays, different UAVs can progress to different "rounds" of algorithm execution without the Stop-Click discipline, resulting in unrecoverable breakdown of the Fibonacci sphere formation in nearly every simulation run. Thus, the Stop-Click phased modification of the algorithm is forced by the possibility of variable message delay, because without it *the very correctness of the algorithm is compromised under variable (even though bounded) delay.*

As part of our baseline test scenario, we have considered zero message delay and constant message delay against a variable number of UAVs. We then tested the robustness of the algorithm under variable message delays with an upper bound on the delay. To remain agnostic to the nature of the network mechanisms, and hence the source of delays in message delivery, we use a generic model of non-deterministic delay. Message delay can occur due to a range of factors including, messages taking longer routes (for multi-hop case), requiring re-transmission (for either single or multi-hop), queuing delay etc. Our model does not focus on the underlying mechanism; instead, we choose to focus on a general high-level picture; accordingly, we use the memoryless exponential distribution that is typically used in networking studies to reflect the lack of knowledge of detailed underlying mechanisms. For practical purposes, in this study, we truncate the distribution to bound the delay to approximately 150 ms. Under this scenario, we studied cases with variable number of UAVs, plume radius, plume boundary and initial sphere formation time.

Each scenario was executed 50 times in order to generate a consistent result. We then plotted the the number of messages exchanged and time taken by the algorithm to reach the goal. To judge the *"goodness"* of our distribution, we have mapped the final position of each UAV to a 3-dimensional voronoi cell, as shown in Fig. 4. The standard deviation of the solid angles is then used to evaluate the difference in coverage.

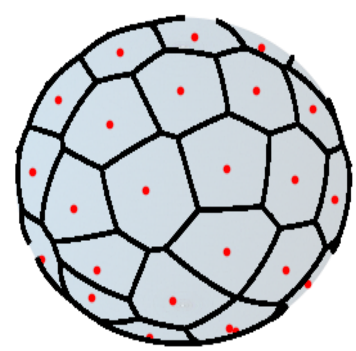
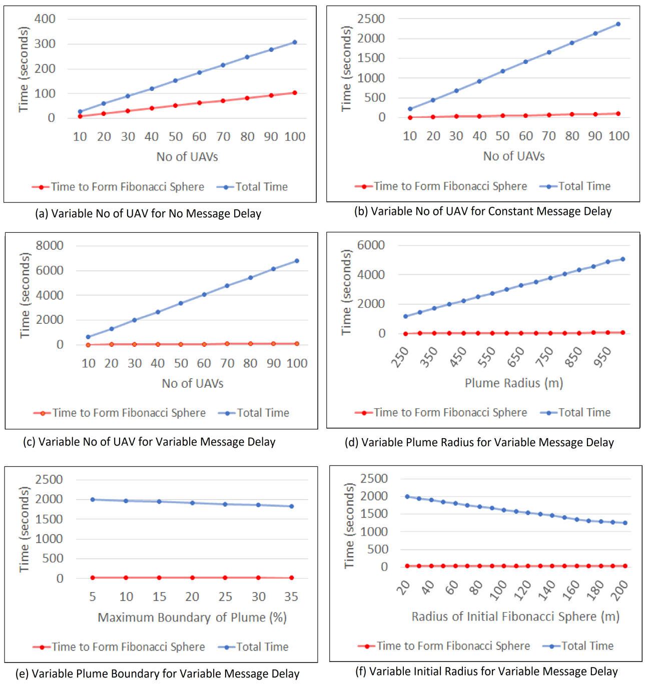


Figure 4: Voronoi cells mapped to each UAV

4.1 Time Taken

The first test criteria is that of time, shown in Fig. 5. We compared both total execution time and time taken to form the initial Fibonacci sphere. As expected execution time increased with increase in number of UAVs and plume radius, but decreases when the boundary region is extended or the initial sphere radius is increased. It is interesting to note the significant increase in total time as soon as message delay is introduced (Fig. 5(b) - 5(f)). Initial Fibonacci sphere formation time is nearly a straight line, leading us to believe that sequential movement of UAVs in the initial phase does not contribute to a considerable increase in time.



(a) Variable No of UAV for No Message Delay
(b) Variable No of UAV for Constant Message Delay
(c) Variable No of UAV for Variable Message Delay
(d) Variable Plume Radius for Variable Message Delay
(e) Variable Plume Boundary for Variable Message Delay
(f) Variable Initial Radius for Variable Message Delay

Figure 5: Execution Time

4.2 Number of Messages

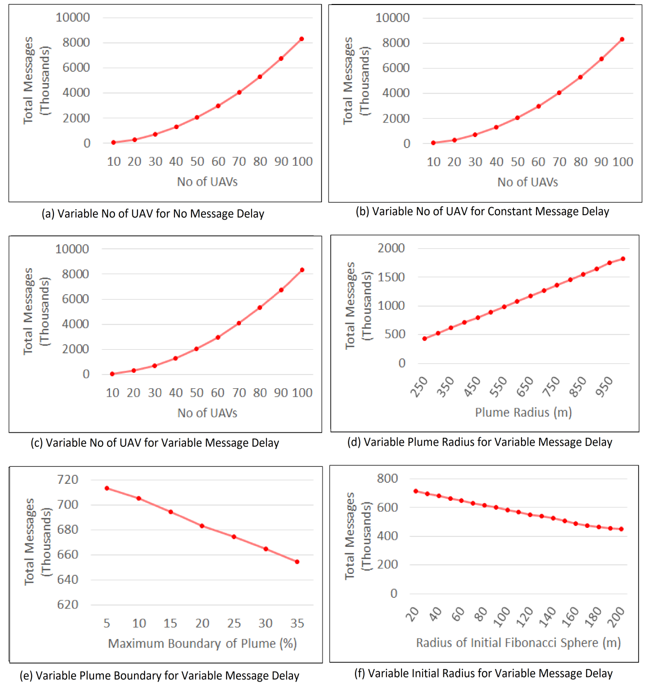
Number of messages exchanged followed similar trends to execution time. Execution time directly co-relates to information exchanged. The plots are shown in Fig. 6

4.3 Standard Deviation of Solid Angle

Finally, we plotted the solid angle standard deviation for the same scenarios, as shown in Fig. 7. It was easy to see that as the number of UAVs increased, the distribution became more uniform, while increasing the plume radius decreased the homogeneity of the coverage. Interestingly, changing the boundary region had a variable effect, with the distribution becoming better initially, before becoming worse. Changing initial sphere radius also produced notable results, since the coverage varied based on the radius. However, in both cases, the variation in coverage was not large.

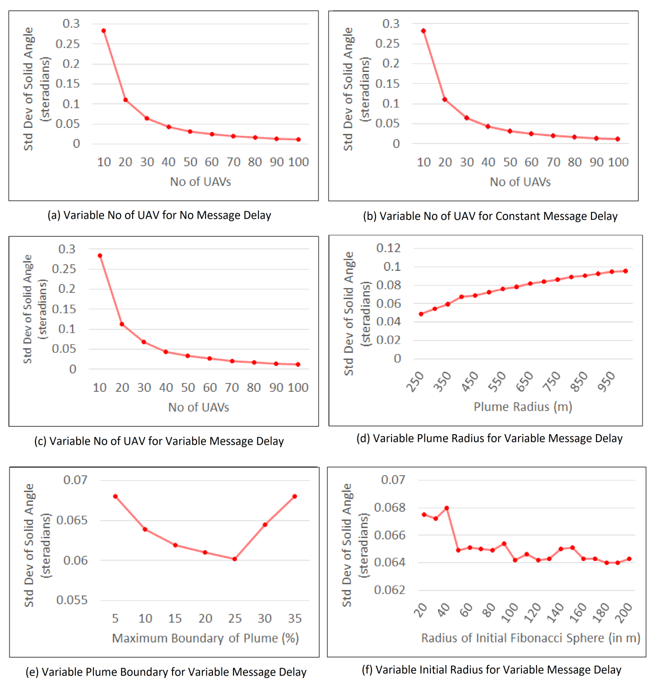
4.4 Confidence Interval

The first and second scenario give deterministic outputs since there is no randomness to the data. Every subsequent output considers messages to be delayed within a given bound. This produces a number of different results, out of which we have plotted the mean. We have also plotted the confidence interval of this data with a confidence of 95%, shown in Fig. 8.



(a) Variable No of UAV for No Message Delay
(b) Variable No of UAV for Constant Message Delay
(c) Variable No of UAV for Variable Message Delay
(d) Variable Plume Radius for Variable Message Delay
(e) Variable Plume Boundary for Variable Message Delay
(f) Variable Initial Radius for Variable Message Delay

Figure 6: Number of Messages



(a) Variable No of UAV for No Message Delay
(b) Variable No of UAV for Constant Message Delay
(c) Variable No of UAV for Variable Message Delay
(d) Variable Plume Radius for Variable Message Delay
(e) Variable Plume Boundary for Variable Message Delay
(f) Variable Initial Radius for Variable Message Delay

Figure 7: Standard Deviation of Solid Angle

Figure 8: Confidence Intervals

It initially surprised us to find that the confidence intervals produced a very narrow margin of error. Having introduced a variation of exponential distribution, we had been expecting to see some more randomness in the data. Due to the nature of our stop-click synchronization, some UAVs must wait for other members of the group to finish broadcasting their updated positions before they can begin moving. Thus, in every round of this stop-click phase, there is some "evening out" of variation.

To get a sense of this disparity, we measured the time when each UAV starts moving and the time when the UAV stops and waits. The scenario we considered consisted of 30 UAVs. In Fig. 9 we see how the time for each round varies from execution to execution, but, the trend of the variations remains similar. This leads to a small variation in the overall execution time.

5 CONCLUSION

In this work, We presented a detailed study on communication and coordination in a group of autonomous UAVs. We have evaluated the dependence of the algorithm on information exchange and we have shown that a task can be successfully completed despite presence of variable (bounded) message delays. Our findings, however, show that there is a trade-off between ensuring correctness and overall mission delay.

There are several avenues we wish to explore in the future. For one, we have considered that in this system, there is no message loss. Going forward we wish to study the effect of message loss on mission parameters and design an algorithm which is resistant to the same up to some threshold. We also want to develop more generalized approaches for communication in Unmanned Aerial

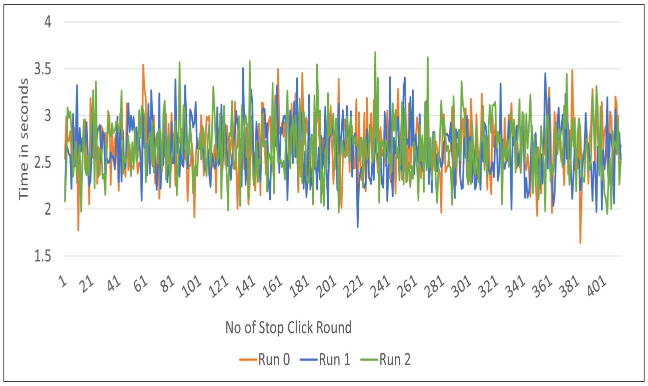


Figure 9: Variation in time for each stop-click round during different executions

Vehicles. Currently UAVs belong to the same platoon and can all send messages to each other. Our plan for the future is to focus on both limiting factors for communication range, as well as, consider the scenario of UAVs in different groups or working individually, who need to communicate for successful task completion.

The material presented in this paper is based upon work supported by the National Science Foundation PAWR Program under Grant No.: CNS-1939334, and its supplement for studying NRDZs.

REFERENCES

[1] Christopher T. Allen, George S. Young, and Sue Ellen Haupt. 2007. Improving pollutant source characterization by better estimating wind direction with a genetic algorithm. *Atmospheric Environment* 41, 11 (2007), 2283–2289. https://doi.org/10.1016/j.atmosenv.2006.11.007

[2] Thomas Arnold, Martin De Biasio, Andreas Fritz, and Raimund Leitner. 2010. UAV-based multispectral environmental monitoring. In *SENSORS, 2010 IEEE.* 995–998. https://doi.org/10.1109/ICSENS.2010.5690923

[3] Angelyn Arputha Babu John and Rudra Dutta. 2017. Cooperative trajectory planning in an intercommunicating group of UAVs for convex plume wrapping. In *2017 IEEE 38th Sarnoff Symposium.* 1–6. https://doi.org/10.1109/SARNOF.2017.8080382

[4] Mohamed Awadalla, Tien-Fu Lu, Zhao F. Tian, Bassam Dally, and Zhenzhang Liu. 2013. 3D framework combining CFD and MATLAB techniques for plume source localization research. *Building and Environment* 70 (2013), 10–19. https://doi.org/10.1016/j.buildenv.2013.07.021

[5] C. H. Bosanquet and J. L. Pearson. 1936. The spread of smoke and gases from chimneys. *Trans. Faraday Soc.* 32 (1936), 1249–1263. Issue 0. https://doi.org/10.1039/TF9363201249

[6] D.W. Casbeer, R.W. Beard, T.W. McLain, Sai-Ming Li, and R.K. Mehra. 2005. Forest fire monitoring with multiple small UAVs. In *Proceedings of the 2005, American Control Conference, 2005.* 3530–3535 vol. 5. https://doi.org/10.1109/ACC.2005.1470520

[7] J. Clark and R. Fierro. 2005. Cooperative hybrid control of robotic sensors for perimeter detection and tracking. In *Proceedings of the 2005, American Control Conference, 2005.* 3500–3505 vol. 5. https://doi.org/10.1109/ACC.2005.1470515

[8] Abhijeet Joshi, Trevor Ashley, Yuan R. Huang, and Andrea L. Bertozzi. 2009. Experimental validation of cooperative environmental boundary tracking with on-board sensors. In *2009 American Control Conference.* 2630–2635. https://doi.org/10.1109/ACC.2009.5159837

[9] Athanasios Ch. Kapoutsis, Iakovos T. Michailidis, Yiannis Boutalis, and Elias B. Kosmatopoulos. 2021. Building synergetic consensus for dynamic gas-plume tracking applications using UAV platforms. *Computers & Electrical Engineering* 91 (2021), 107029. https://doi.org/10.1016/j.compeleceng.2021.107029

[10] Ali Katanforoush and Mehrdad Shahshahani. 2003. Distributing Points on the Sphere, I. *Experimental Mathematics* 12, 2 (2003), 199–209. https://doi.org/10.1080/10586458.2003.10504492 arXiv:https://doi.org/10.1080/10586458.2003.10504492

[11] Benjamin Keinert, Matthias Innmann, Michael Sänger, and Marc Stamminger. 2015. Spherical Fibonacci Mapping. *ACM Trans. Graph.* 34, 6, Article 193 (oct 2015), 7 pages. https://doi.org/10.1145/2816795.2818131

[12] M. Kemp, A.L. Bertozzi, and D. Marthaler. 2004. Multi-UUV perimeter surveillance. In *2004 IEEE/OES Autonomous Underwater Vehicles (IEEE Cat. No.04CH37578).* 102–107. https://doi.org/10.1109/AUV.2004.1431200

[13] Tatsuaki Kimura and Masaki Ogura. 2020. Distributed Collaborative 3D-Deployment of UAV Base Stations for On-Demand Coverage. In *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications* (Toronto, ON, Canada). IEEE Press, 1748–1757. https://doi.org/10.1109/INFOCOM41043.2020.9155283

[14] Ali Marjovi and Lino Marques. 2013. Swarm robotic plume tracking for intermittent and time-variant odor dispersion. In *2013 European Conference on Mobile Robots.* 379–384. https://doi.org/10.1109/ECMR.2013.6698871

[15] Prathyush P. Menon and Debasish Ghose. 2012. Simultaneous source localization and boundary mapping for contaminants. In *2012 American Control Conference (ACC).* 4174–4179. https://doi.org/10.1109/ACC.2012.6315528

[16] E. B. Saff and A. B. J. Kuijlaars. 1997. Distributing many points on a sphere. , 5–11 pages. https://doi.org/10.1007/bf03024331

[17] Jürgen Scherer, Saeed Yahyanejad, Samira Hayat, Evsen Yanmaz, Torsten Andre, Asif Khan, Vladimir Vukadinovic, Christian Bettstetter, Hermann Hellwagner, and Bernhard Rinner. 2015. An Autonomous Multi-UAV System for Search and Rescue. In *Proceedings of the First Workshop on Micro Aerial Vehicle Networks, Systems, and Applications for Civilian Use* (Florence, Italy) *(DroNet '15).* Association for Computing Machinery, New York, NY, USA, 33–38. https://doi.org/10.1145/2750675.2750683

[18] Oliver Graham Sutton. 1947. The theoretical distribution of airborne pollution from factory chimneys. *Quarterly Journal of the Royal Meteorological Society* 73 (1947), 426–436.

[19] Jorge Edwin Sánchez-Sosa, Juan Castillo-Mixcóatl, Georgina Beltrán-Pérez, and Severino Muñoz-Aguirre. 2018. An Application of the Gaussian Plume Model to Localization of an Indoor Gas Source with a Mobile Robot. *Sensors* 18, 12 (2018). https://doi.org/10.3390/s18124375

[20] Qiaoyi Xu, Wenli Du, Jinjin Xu, and Jikai Dong. 2021. Neural network-based source tracking of chemical leaks with obstacles. *Chinese Journal of Chemical Engineering* 33 (2021), 211–220. https://doi.org/10.1016/j.cjche.2020.12.022