

Proposal: Optimizing Mosquito Repellent Dispersion Utilizing Autonomous Drone Controls

Group 2: Joe Conroy, Noah Constable

Problem Statement

Our intent is to compare two EGI training methods for the purpose of seeking mosquito nests. The first method will utilize autonomous drones via edge protocols. The second will utilize near-edge protocols in the form of a truck that controls the drones autonomously.

Introduction

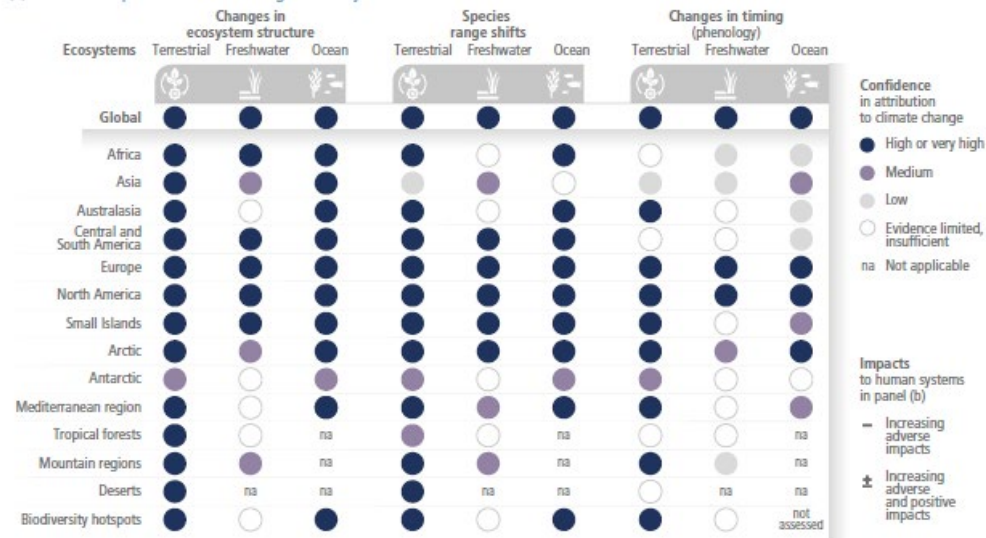
Automated analysis and prediction model generation has advanced quite considerably to present day. Notable examples range from Waymo's autonomous ridesharing vehicles [1], John Deere's Precision AG Technology [2], and the recent application of attack drones in Operation Spider's Web [3]. All three enterprises are also examples of spatial data being used to remove people from harm's way, reduce costs or improve production. Presuming that the hardware for data training remains readily accessible, applying spatial data will continue to provide economic and societal benefits.

One area where this technology is likely going to become critical is environmental preservation and adaptation. The most recent (2022) report of the IPCC indicates that various natural and developed ecosystems are all subject to adverse impacts from the changes in the climate caused by greenhouse gas emissions from human enterprises should the global average temperature reach 1.5°C above the global average in the pre-industrial era [6]. You can observe Figure SPM.2, which was taken directly from the IPCC's Sixth Assessment Report, on the next page to see how various systems are expected to be affected by climate change.

One significant expected trend to note of is that terrestrial ecosystems across the globe are highly likely to experience attribution from climate change. In addition, the North American, European, Small Island and Arctic regions are all highly likely to experience attributions from climate change. It is also clear that nearly all cities, infrastructure, health and (human) wellbeing are expected to experience adverse impacts without notable positives. Based on these trends, should they come to pass, we can expect that managing these risks will involve deciding whether to prioritize human systems or ecosystems.

Impacts of climate change are observed in many ecosystems and human systems worldwide

(a) Observed impacts of climate change on ecosystems



(b) Observed impacts of climate change on human systems

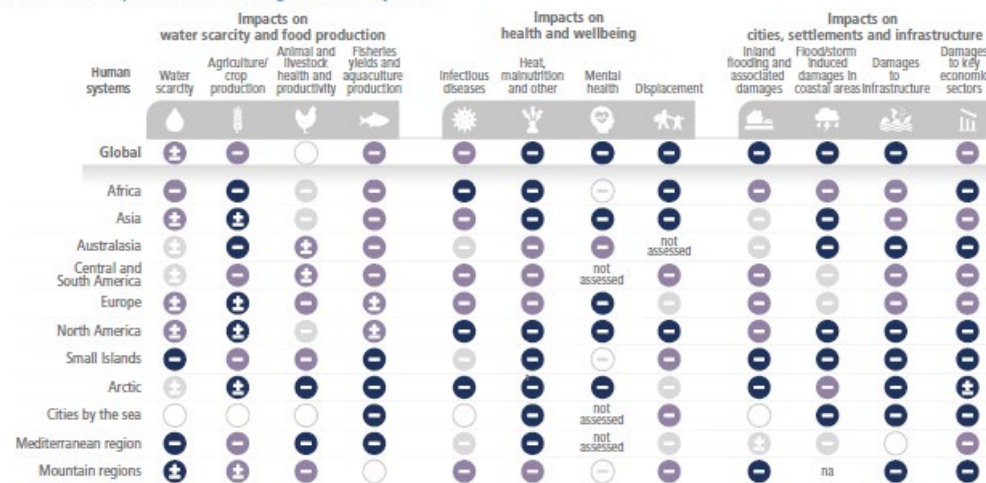


Figure SPM.2 | Observed global and regional impacts on ecosystems and human systems attributed to climate change. Confidence levels reflect uncertainty in attribution of the observed impact to climate change. Global assessments focus on large studies, multi-species, meta-analyses and large reviews. For that reason they can be assessed with higher confidence than regional studies, which may often rely on smaller studies that have more limited data. Regional assessments consider evidence on impacts across an entire region and do not focus on any country in particular.

Three years after the IPCC Sixth Report, studies had reported that we were likely to pass the 1.5°C threshold within a 20-year period [7] and that global temperature trends over a consecutive 12-month period are indicative of being on track to passing 1.5°C [8]. Assuming the assessments from these studies were correct, resource management is going to become critical in the coming decades. Human resources are particularly going to become increasingly strained for availability, as expertise and adaptability will be indispensable for handling multiple crises of different and overlapping varieties. Automation will be necessary to both free resources up and fill in gaps where people are not available to help.

A particular example of environmental control that interests us and will likely be affected by climate change is managing mosquito populations. Mosquitos can carry deadly diseases such as Dengue, Malaria, West Nile and Zika viruses [4]. Controlling mosquito populations is thus a critical line of defense for preventing deadly diseases from spreading to people. To control mosquito populations, the Minneapolis Mosquito Control District (MMDC) employs various larval controls such as *Bacillus thuringiensis israelensis* (Bti), methoprene and Spinosad [9]. Deploying these controls requires identifying mosquito nesting locations while the mosquitos are still in their larval state.

There are three main challenges to overcome when reducing mosquito populations with the MMCD's current method. Exposure to the mosquito population greatly increases the chances of mosquito bites and in turn the risk of transmitting diseases. At the same time, attempting to control mosquito populations can disturb the freshwater food webs they tend to be a part of if deployment isn't methodical. For example, Bti presents risks to chironomid larvae as well as to mosquito larvae [5]. Chironomids tend to be diverse, highly digestible, and protein-rich sources of food for various freshwater creatures. Deploying too much Bti risks significantly reducing the chironomid species and thus doing more harm to the environment. The third challenge is the required accuracy of deployment procedure itself, and that is where we believe our method will be able to fill a gap caused by resource constraints. Deployment can only be done on a seasonal basis to target the larvae before they develop into adults. A successful deployment framework for mosquito controls will need to deploy control methods over a familiar seasonal landscape, with accurate identification of the targeted areas and a measured volume of the control deployed.

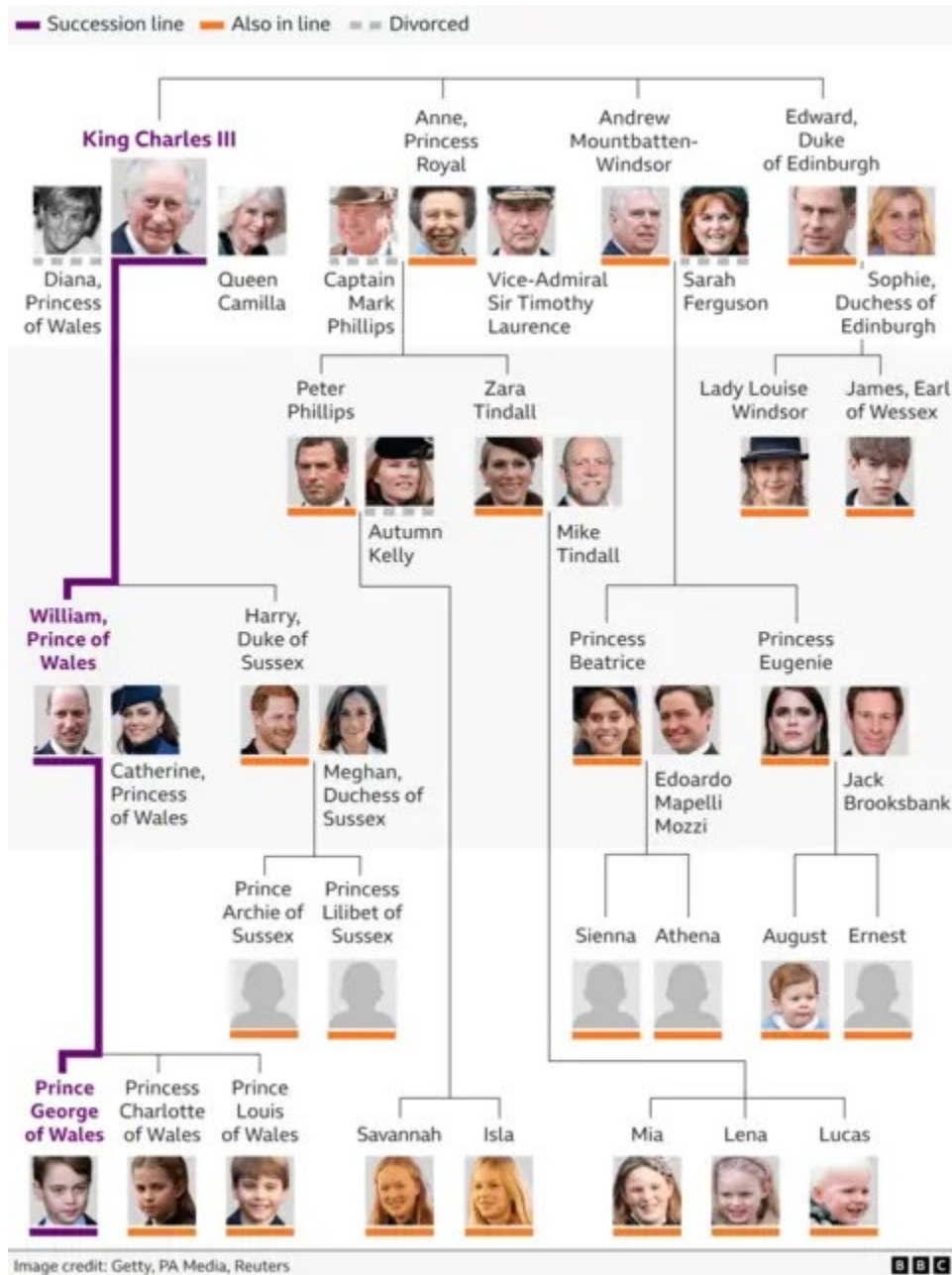
Literature Review

Graphs

Graphs are an organizational way to arrange data that is defined by nodes and vertices between nodes. Nodes represent data points and can contain multiple attributes that are specific to each data point. Vertices represent relationships between nodes and can contain attributes that are exclusively tied to the relationship between the connected nodes. Vertices can also be directional, meaning that one node is related to another but not the other way around.

Consider the following chart from a BBC article detailing the current British Royal Family [10]. Each family member has a name, a title of royalty, a picture of them when available, lines connecting them based on succession and direct descendance, and are positioned by their spouses. We can translate this to a graph by treating each person in the family as a node, with attributes tied to their name, title, if they are in the line of succession, and any details contained in the photos. Vertices would represent familial

relationships such as being a spouse, a sibling, who descends from them and who they are descendants from. Whether a family member is in the line of succession can be represented as either an attribute on the nodes or as an attribute of the vertices. Which representation we would prefer depends on the intended purpose of the graph we wish to construct.



Graph Intelligence (GI) Models

Graph intelligence (GI) models treat graphs as input and output either node predictions, vertex predictions or graph predictions. Node predictions are predictions

about attribute changes for each node. Vertex predictions are predictions on relationships between the existing nodes on the input graphs. Graph predictions are where an entire graph is provided as output.

Consider the graph we derived from the Royal Family chart above, which includes all relationships in the vertices. A node prediction model for this graph might attempt to predict changes in each member’s title, name or filling in a picture when none is present. Meanwhile, a vertex prediction model would try to predict changes in the lines of succession or whether a divorce was going to happen. Finally, a graph model would try to predict changes to the entire family tree structure, such as family members getting removed, new children being born, or whether the line of succession has effectively ended.

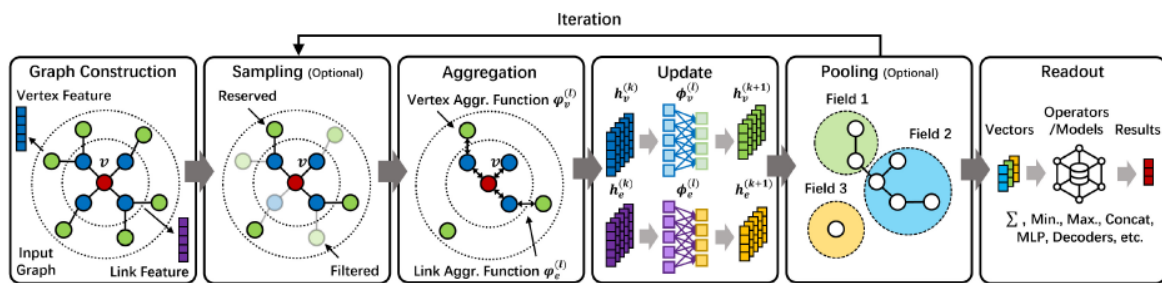


Fig. 3. General workflow of GI models. Given an input graph with feature vectors, a GI model iteratively performs sampling, aggregation, update, and pooling through consecutive model layers. The obtained embeddings will be finally converted to results in an expected form through a readout function.

GI models are trained through an iterative process, which is illustrated in Figure 3 shown above [11]. Each iteration starts with a subgraph being sampled from the main graph. The subgraph is then aggregated into a representation vector based on the neighbors of the sampled vertices. Neighbors for a vertex can be the nodes that form the vertex and the nodes connected to those initial nodes by other vertices. Which details are included in the representation vector is based on shared attributes between the initial nodes, and specifications from the user(s). Finally, the aggregation then gets passed through a neural network to update the representation to the desired output. If the output passes enough cutoff requirements for the neural network, then it is returned as the model. Otherwise, the iterative sampling starts over from the sampling step.

Edge Computing and Networks

Edge networks are networks of computers that are organized to manage limited or changing resources without a centralized resource. Computers in an edge network vary widely in computing or memory capacity and are dispersed geographically with multiple changing resource constraints that are a result of their environment. Edge networks are designed with the understanding that there is no guarantee of a consistent connection between any of its devices. To resolve this, devices on the network are categorized based

on their processing capacity. End devices, which are at the ends of the network, typically contain simple devices unable to handle high-cost processing such as sensors and mobile devices. Meanwhile edge devices can handle collecting data or simple processed information as the end devices connect and disconnect. Edge devices typically include micro data centers. Edge cloud devices are typically at the top of the network, and handle the heavy processing expected to be dispersed to the edge devices. Robots and vehicles can be categorized as either end or edge devices, depending on their function. Given the wide variety of computer types on an edge network, the software networking them is often required to be some combination of cross-platform capable, cross-protocol capable, language-agnostic and resource-efficient.

Edge Graph Intelligence (EGI)

Edge networks that utilize artificial intelligence (AI) are instances of edge intelligence, with EGI being the case where graphs are treated as the input and define the output. One key advantage of edge intelligence is that data centers are freed from storing large amounts of data and handling high amounts of traffic. The edge devices can handle the traffic and data for end devices connected to it, freeing specific tasks from the cloud servers. Meanwhile, the cloud devices provide more general services for the edge devices, preventing edge network devices from having to attempt to communicate with each other all at once.

In the *Edge IG: Reciprocally Empowering Edge Networks with Graph Intelligence* study from Zeng et al [11], EGI was classified into 6 different levels depending on where the AI was applied within the edge network and how much awareness of the graph was utilized in the training process. Applications of AI that are performed on the end devices alone are handled at levels 0 (not even aware of a graph structure) through 2 (GI). Meanwhile levels 3 (cloud and end devices) through 5 (the full network connection) involve applications that require interactions between edge and cloud devices to train models with full knowledge of the graph data itself.

Consider how a self-driving vehicle may utilize each level. Local activities tied to computer vision and movement through dynamic terrain would be classified as levels 0 through 2. Finding an optimal distance/minimally difficult path through a global environment would be level 3, where GI training is handled by a single cloud or edge device. Level 4 would involve training GI across multiple edge devices, such as a model to predict traffic changes across cities where each city monitors its own traffic. Finally, a dynamic weather forecast that requires all edge devices to train together using all their own data would be an example of a level 5 EGI framework.

Federated Learning

Federated learning is a framework in which edge devices communicate with a central server to implement iterative model aggregation. Each iteration involves the edge devices training their own models and sharing partial models with the central server. The central server then aggregates the partial models into its own generalized model, which is then divided into partial models and sent back to the edge devices to assist with their local training. In the paper *Peer-to-Peer Variational Federated Learning Over Arbitrary Graphs*, federated learning had the central server communication replaced with each edge devices' neighbors [12].

Gossip-Based Training

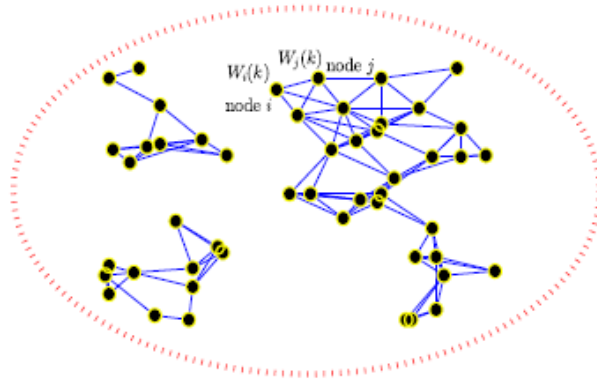


Fig. 1. A graphical illustration of the distributed learning problem over the time-varying undirected random network with 50 nodes in a time instant k , $k \in \mathbb{N}$, as an example. Each node i has its own raw dataset $\mathcal{S}_i = \{(x_i^l, y_i^l)\}_{l=1}^{N_i}$, whose size is N_i . The locally blocked datasets \mathcal{S}_i , $i = 1, 2, \dots, N$, compose the entire dataset \mathcal{S} . In the learning process at each time instant $k \in \mathbb{N}$, only two neighboring nodes (e.g., node i and node j) are randomly selected successively. Only the two selected nodes in the network have access to the output weight vectors (e.g., $W_i(k)$ and $W_j(k)$) from each other, and the other nodes stay silent and idle at the time instant. The goal of the problem is to obtain a global optimal output weight vector W^* only with the raw dataset that is distributed and blocked among the spatially distinct nodes in the network by using a gossip-based communication protocol.

Gossip-based communication protocols are designed to spread communication between edge devices. Figure 1 above highlights an example of a gossip-based network in action [13]. The protocol starts with each edge device randomly selecting one of its neighbor devices to pair with. Each pair of devices then shares information during a fixed period of either iterations of sharing, iterations of learning, or fixed time. At the end of each period, the devices select different neighbors to exchange information with. The intent of this process is to prevent a fixed line of communication and enable easier recovery when communication protocols are disrupted. One application of gossip-based training we studied for our project outlined an algorithm graph-based distributed cooperative learning (GBDCL) [13]. GBDCL measures weights of each paired edge device before returning each measured weight to be aggregated into a sum.

Limitations of Existing Work

The papers we reviewed for our project (11-13) all propose some form of device communication framework or analysis of graph data. Based on our research before carrying out our project, contrasting the utility of each of the frameworks and measuring their effective loss in performance in carrying out specific tasks was not previously done. The graph analysis papers focused on the performance of their systems with the assumption that communication was uninterrupted. The communication papers showed that their methods could provide further safeguards from network drops or lack of availability, but did not evaluate the loss in performance as a result of the communication losses.

Methodology

Evaluating each method's performance will be based on multiple factors. The number of false positives where a nest is incorrectly identified provides insight into how likely the method is to unnecessarily harm the environment. Failing to find a nest before the drones need to be recalled for battery consumption would constitute either a true negative, or accurately determining no nest is present. If there was a nest, then the robot presented a true negative that entails risk of letting populations grow. Measuring both false positives and true negatives will enable us to contrast the expected failure type.

Resource management can also be evaluated based on what was required to train a model to achieve acceptable accuracy and efficient use of time, as well as how many resources were required of the agents making the decisions. This measurement will help clarify if the preferred method involves treating drones as remotely piloted devices (our baseline), as edge devices (drones make decisions on their own), or as an extension of a platform hosting a trained model. Furthermore, we can determine if a portable computer, a computer-embedded vehicle, or a remote data center is a preferred host for the trained model.

Schedule

Our project milestones are listed below:

Date	Milestone
3/17 – 3/23	Finalize Task Formulation, Build Spatial Grid Simulator, and Implement Centralized GNN Baseline
3/24 – 3/30	Implement Partitioning Schemes and Federated Averaging, and Verify Correctness
3/31 – 4/6	Implement Decentralized Gossip Training; Simulate Communication Constraints
4/7 – 4/13	Run Experiment Matrix (Connectivity x Partition x Compute Constraints) and Collect Metrics

4/14 – 4/20	Analyze Results, Generate Plots, Evaluate Performance, Prepare Presentation and Report
4/21-4/30	Project Presentation
5/8	Final Paper Due

Resources

1. WAYMO; *WAYMO Home Page*; 2019-2026 Waymo LLC; <https://waymo.com/>
2. John Deere; *Prevision AG Technology Page*; 2026 Deere & Company; <https://www.deere.com/en/technology-products/precision-ag-technology/>
3. CSIS; *How Ukraine’s Operation “Spider’s Web” Redefines Asymmetric Warfare*; 2025 Center for Strategic & International Studies; <https://www.csis.org/analysis/how-ukraines-spider-web-operation-redefines-asymmetric-warfare>
4. Texas Health and Human Services, 2026 Texas Department of State Health Services, *List of Mosquito-Borne Diseases*, <https://www.dshs.texas.gov/mosquito-borne-diseases/list-mosquito-borne-diseases>
5. Stefanie Allgeier, Anna Friedrich, Carsten A. Brühl, *Mosquito control based on *Bacillus thuringiensis israelensis* (Bti) interrupts artificial wetland food chains*, *Science of The Total Environment*, Volume 686, 2019, Pages 1173-1184, ISSN 0048-9697, <https://doi.org/10.1016/j.scitotenv.2019.05.358>.
6. IPCC; 2022 Intergovernmental Panel on Climate Change; *Climate Change 2022: Impacts, Adaptation and Vulnerability; Working Group II Contribution to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change*; <https://www.ipcc.ch/report/ar6/wg2/>
7. Bevacqua, E., Schleussner, CF. & Zscheischler, J. A year above 1.5 °C signals that Earth is most probably within the 20-year period that will reach the Paris Agreement limit. *Nat. Clim. Chang.* **15**, 262–265 (2025). <https://doi.org/10.1038/s41558-025-02246-9>
8. Cannon, A.J. Twelve months at 1.5 °C signals earlier than expected breach of Paris Agreement threshold. *Nat. Clim. Chang.* **15**, 266–269 (2025). <https://doi.org/10.1038/s41558-025-02247-8>
9. MMCD, 2026, *Mosquito Control*, <https://mmcd.org/mosquito-control/>
10. BBC, 20 February 2026 BBC, *Royal Family tree: King Charles III’s closest family and line of succession*, <https://www.bbc.com/news/articles/c867plj4vvgqo>
11. L. Zeng et al., *Edge Graph Intelligence: Reciprocally Empowering Edge Networks With Graph Intelligence*, in *IEEE Communications Surveys & Tutorials*, vol. 27, no. 6, pp. 3417-3454, Dec. 2025, doi: 10.1109/COMST.2025.3527561.

12. X. Wang, A. Lalitha, T. Javidi and F. Koushanfar, "Peer-to-Peer Variational Federated Learning Over Arbitrary Graphs," in IEEE Journal on Selected Areas in Information Theory, vol. 3, no. 2, pp. 172-182, June 2022, doi: 10.1109/JSAIT.2022.3189051.
13. Pengfei Ren, Hao Dai, Weisheng Chen, *Distributed cooperative learning over time-varying random networks using a gossip-based communication protocol*, Fuzzy Sets and Systems, Volume 394, 2020, Pages 124-145, ISSN 0165-0114, <https://doi.org/10.1016/j.fss.2019.05.009>