Experimental and Simulation of a Remote SDR Testbed with Distributed Cooperative DSS

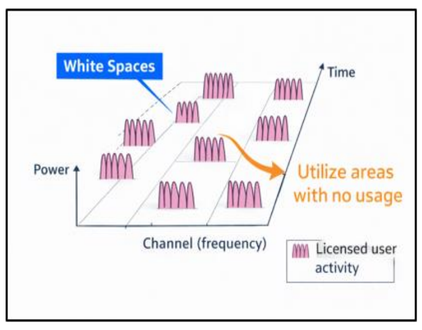
Austin Trinh, Simon Pham, Osiris Germain, Joshua Vo, Johnny Daou
Advisor: Dr. Michael Rahaim
Engineering Department - University Of Massachusetts Boston

Abstract

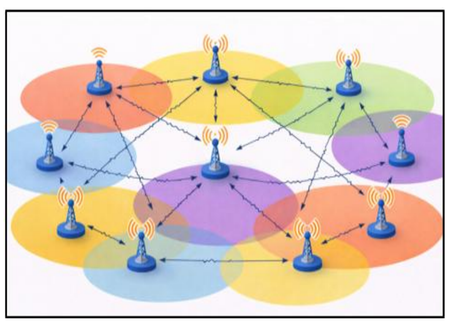
The radio spectrum is divided into licensed channels assigned to **Primary Users (PUs)**. Spectrum Congestion becomes a problem when many users are using a channel. To combat this, the **Dynamic Spectrum Access (DSA)** framework is introduced to help with efficient management of the frequency spectrum. Dynamic Spectrum Access (DSA) allows **Secondary Users (SUs)** to use idle channels when PUs are inactive. This research focuses on **Distributed Spectrum Sensing (DSS)**, where nodes make sensing decisions without relying on a central controller.

Background

Efficiency in Wireless

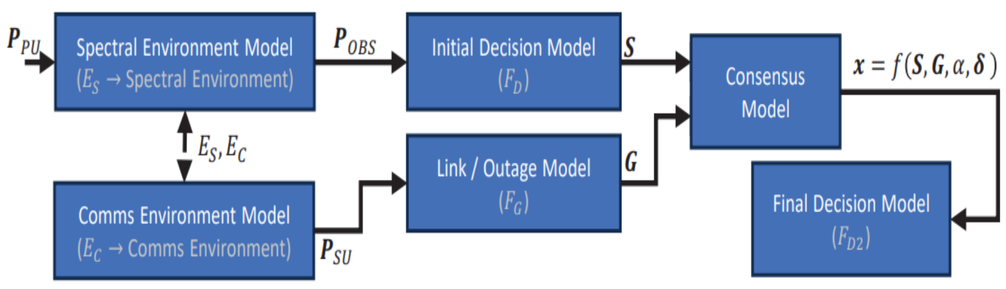


Time and Frequency



Spatial

Model Framework



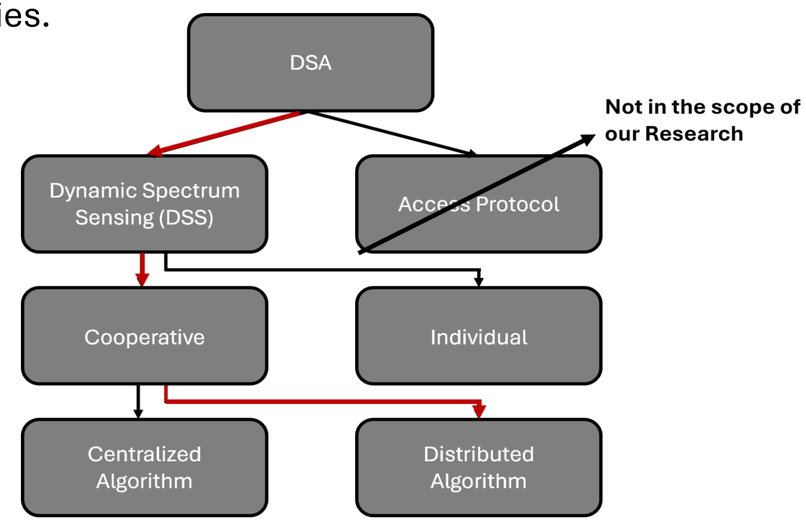
E_S	Spectral Environment Parameters
E_C	Comms Environment Parameters
F_D	Initial Decision Model
F_G	Link/Outage Model
F_{D2}	Final Decision Model

P_{PU}	Tx Power array from PUs
P_{OBS}	Observed PU Power at SUs
P_{SU}	Comms Environment Rx Power from SUs
S	Initial SU measurements or predictions
G	Time-Varying Graph
x	Time varying statistic for each node

Framework Architecture Variables

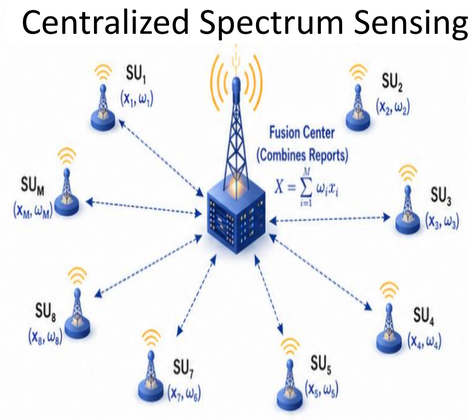
Dynamic Spectrum Access (DSA)

- Enables dynamic use of available spectrum.
- Allows user devices to automatically adjust their operating frequencies.



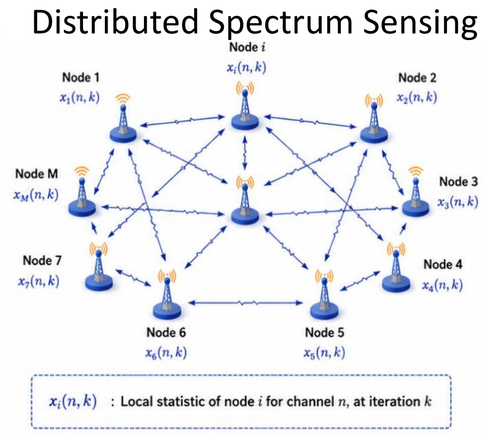
Spectrum Sensing (DSS vs CSS)

Centralized Spectrum Sensing



$$X = \sum_{i=1}^{M} \omega_i \, x_i$$

Distributed Spectrum Sensing

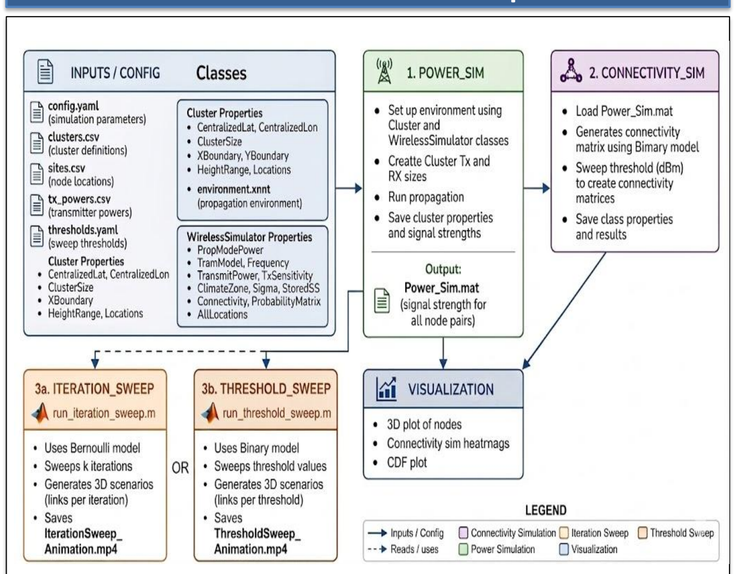


$$x_i(n, k+1) = x_i(n,k) + \frac{\alpha}{\delta_i} \sum_{j \in N_i(k)} (x_j(n,k) - x_i(n,k))$$

Pros/Cons
• Low complexity
• Faster Consensus Time → **Single Point of Failure**

Pros/Cons
• Network Security/Resilience → **Complex**
• Strong Infrastructure with scalability → **Slow Global Consensus time**

Simulation Setup

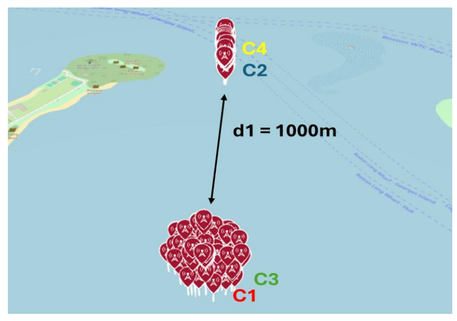


Technologies

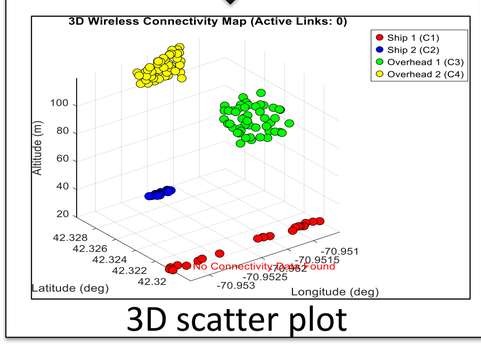
MATLAB®

OpenStreetMap

Simulation Configuration



Physical Location



3D scatter plot

Configuration

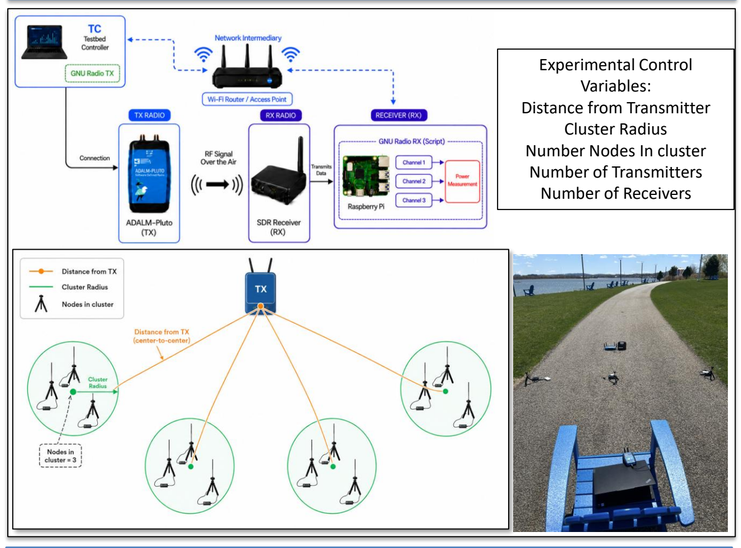
Parameters:

- Number of nodes per cluster:
  - C1, C2 = 20, 20
  - C3, C4 = 100, 100
- Height range per cluster:
  - Low_alt = [20, 20]
  - High_alt = [20, 120]
- Distance between ships:
  - d1 = 1000m

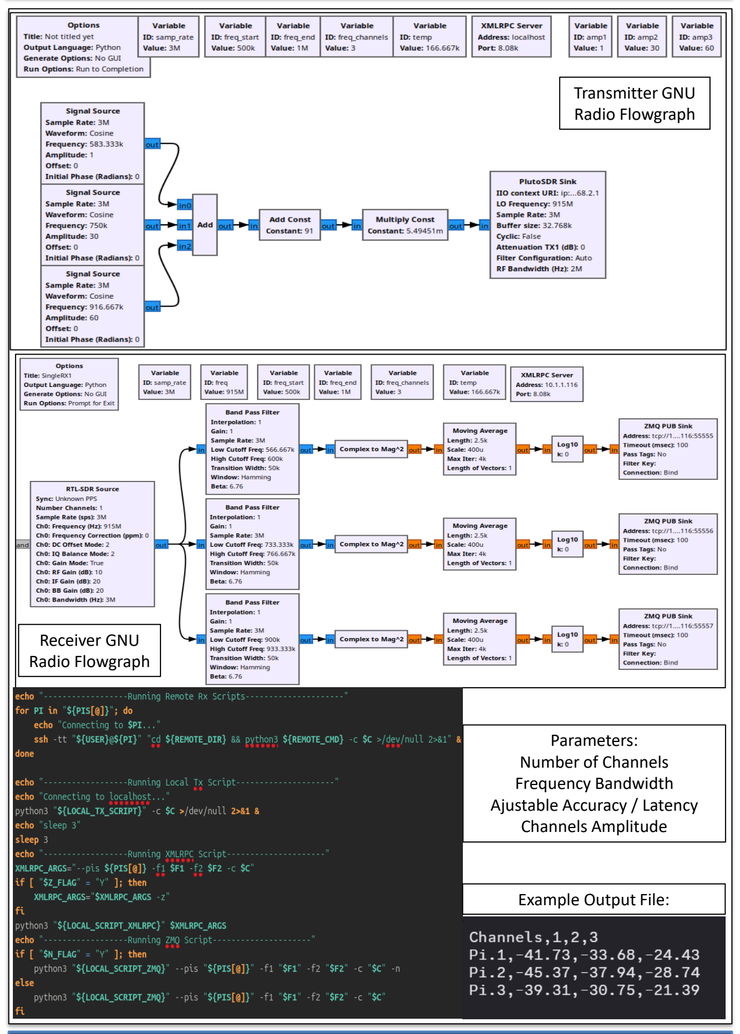
[OPTIONAL: boundary for rectangle/square clusters]
- C1 = 50m x 50m
- C2 = 50m x 50m

[OPTIONAL: boundary for cross-sectional/hemisphere clusters]
- Radius C3, C4 = High_alts(2) – High_alts(1) = 100

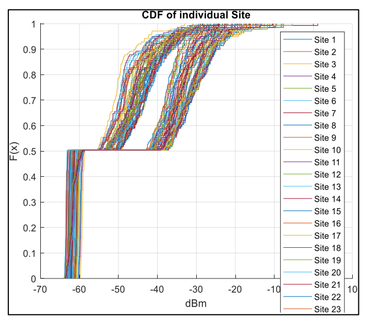
Experimental Setup



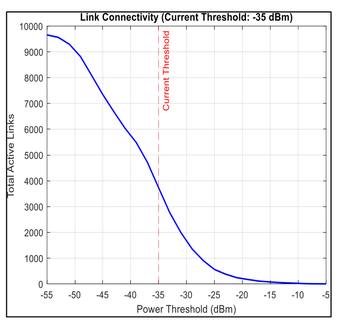
Experimental Automation



Simulation Results



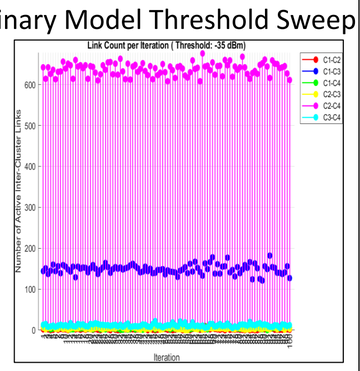
CDF plot



Binary Model Threshold Sweep

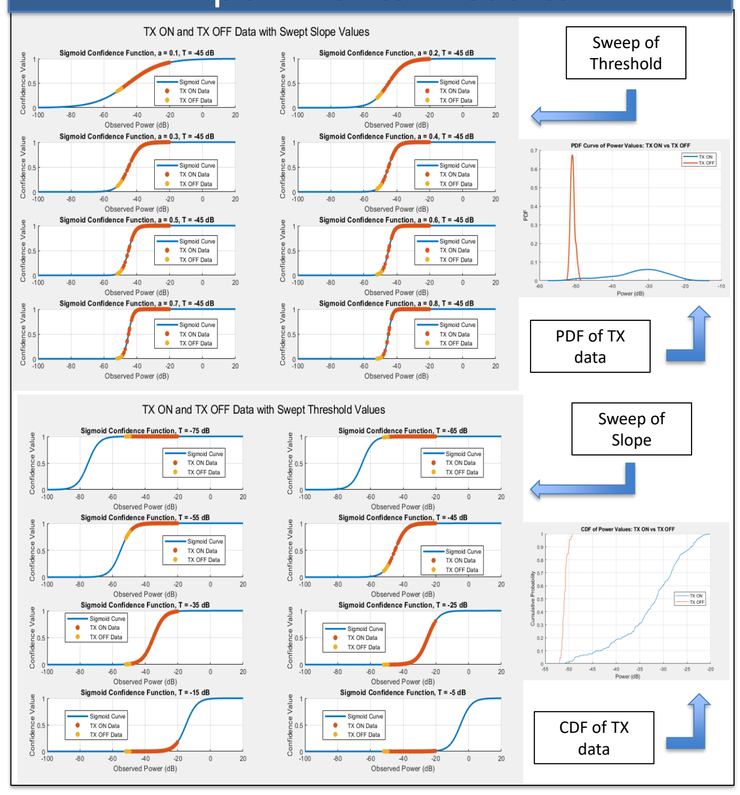
Signal Strength Probability Mapping (Bernoulli)

$$p(x) = \frac{1}{1 + e^{-(\frac{ss - threshold}{\sigma})}}$$



Bernoulli Model Iteration Sweep

Experimental Results

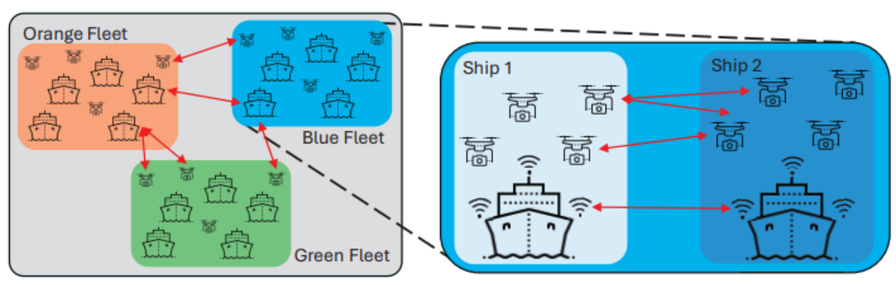


Future Works

This framework aims to be automated and repeatable, providing a baseline idea of what researchers could implement and collect in a more complex setup. The technologies can be varied to simulate a practical, dynamic environment, such as:

Drone

DJI RoboMaster S1



Improved various cluster configurations (i.e., Hemisphere for ship clusters), including the Pu to SU matrices for the spectral environment in maritime settings. Ensure scalability for different scenarios (e.g., urban and rural areas).

Acknowledgements

This independent study was inspired by Ubiquitous Communication and Networking (UCaN) Lab: