

Buffalo Byte Swarm: Initial Design Memo

Group 6: Bob French, Dylan Pinos, Zhiyun Yu

Project Overview

Problem Definition

In disaster relief situations (earthquakes, tornadoes, etc.), rescue efforts must first search for people that require assistance before deploying a rescue team to that location. This searching stage consumes critical time in which lives could be saved, and spread resources thin as a result of having to split resources between searching and rescuing. Our project aims to eliminate this step, allowing rescue teams to begin their rescue much sooner, based on a list of suspected rescue locations.

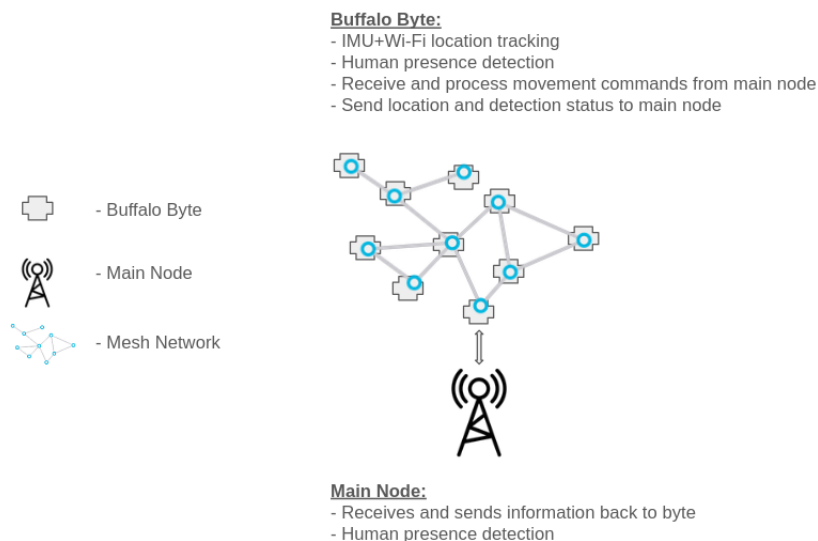
Alternative Solutions

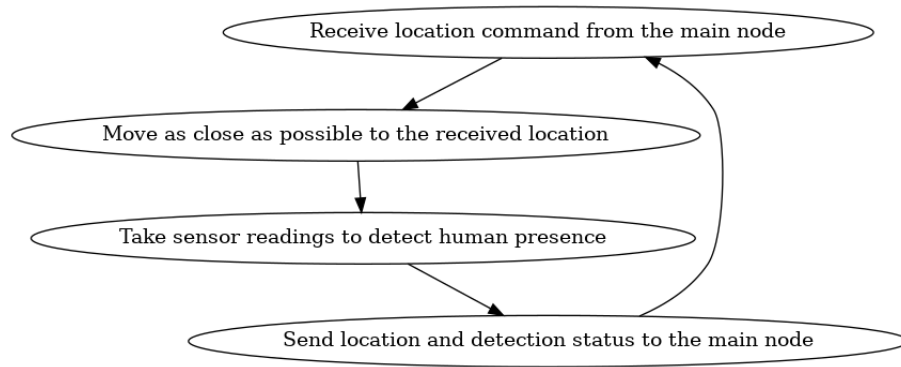
RecoNode [1] While our proposed Buffalo Byte mesh network offers a promising solution, alternative approaches including heterogeneous robot teams, tethered systems, WSAC networks, dual-baseband radios, vision-based AI, and real-time operating systems—can further enhance search and rescue efforts. Unlike RecoNode, which employs fully custom hardware and software, our design utilizes off-the-shelf components, making it significantly more cost-effective, scalable, and easier to deploy. Integrating insights from the RecoNode research will provide a strong technical foundation for designing a more robust disaster relief system.

Proposed Approach

The approach we plan to take is to create a mesh network using a large number of Buffalo Bytes, and a central node/server, hosted on a Raspberry Pi or similar. The Buffalo Bytes will travel across the affected area and take sensor readings at certain points specified by the main node. If human presence is detected, the Buffalo Byte will send its location through the mesh network back to the main node to signify that there may be someone in need of rescuing at that location.

Conceptual Model





Measure of Success

Technical Standards

ISO 13482:2014

This standard describes the safety requirements for non-medical and non-industrial human aid robots, and the factors that must be satisfied in order for the robot to be considered safe for human interaction.

IEEE 42010

This standards describes best practice for organizing software-heavy systems in a modular manner. This will be relevant to our project because many of our subsystems are heavily software reliant. Building them in a modular way will make testing each individual components much easier as it allows for each subsystem to be tested completely isolated from the others.

Measurement Criteria

Mesh Network: Measure the average latency and packet loss when sending data over the mesh network.

Location Tracking: Measure the average offset between the predicted and measured location.

Human Sensing: Measure success rates for human detection and percentage of false positives.

Pathfinding: Create a list of obstacles and measure how reliably a Byte can navigate around them.

Measurement Plan

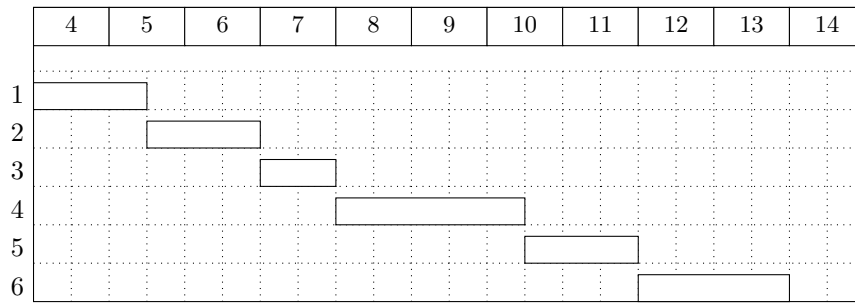
When comparing possible approaches to designing a subsystem (e.g. sensor type, pathfinding algorithm, etc.), we will use the above measurement criteria to evaluate the alternatives and choose the most performant. Upon the completion of a subsystem, it will be tested by the appropriate criteria to establish a baseline measurement. As these subsystems are tweaked, we will test again to ensure that the changes result in better performance with respect to these criteria. Before implementing a design, we will check with the relevant standards mentioned above to ensure that the design would adhere to them.

Timeline

Bob's Timeline

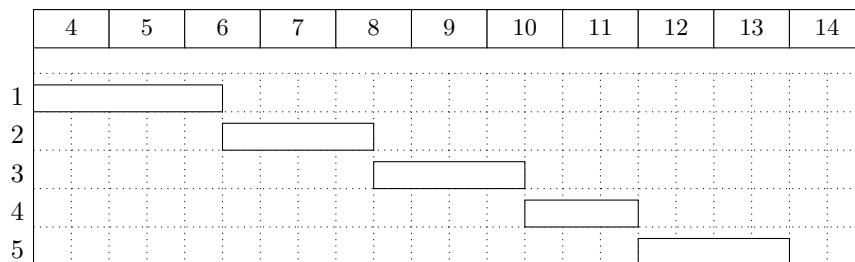
1. Create a software simulation environment for Buffalo Byte which simulates movement and IMU data
2. Set up the mesh network, and confirm communication is possible between Bytes and the main node
3. Create protocol for choosing a new location and sending the Byte to it
4. Create a pathfinding algorithm for the Buffalo Byte to find the locations given to it by the main node

5. Create the server/client applications for the Byte and the main node
6. Conduct testing on the Byte Swarm in a physical environment, and make adjustments accordingly



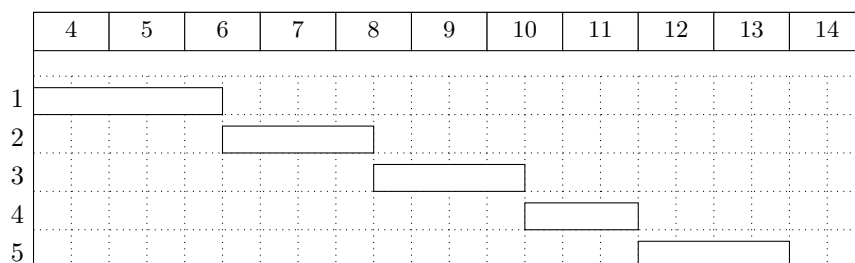
Dylan's Timeline

1. Write equations and create a program to track position data using the IMU and EKF
2. Create a program to enhance this movement tracking via Wi-Fi triangulation
3. Create a pathfinding algorithm for the Buffalo Byte to find the locations given to it by the main node
4. Finalize and implement protocols for server/Byte communication
5. Conduct testing on the Byte Swarm in a physical environment, and make adjustments accordingly



Zhiyun's Timeline

1. Explore sensor options and test alternatives to choose the best option
2. Interface the selected sensor with the ESP8266
3. Create a pathfinding algorithm for the Buffalo Byte to find the locations given to it by the main node
4. Finalize and implement protocols for server/Byte communication
5. Conduct testing on the Byte Swarm in a physical environment, and make adjustments accordingly



References

- [1] R. M. Voyles, S. Povilus, R. Mangharam, and K. Li, “Reconode: A reconfigurable node for heterogeneous multi-robot search and rescue,” in *2010 IEEE Safety Security and Rescue Robotics*, 2010, pp. 1–7.