Disaster-Response Coverage and Relief with a Multi-Robot Team using Voronoi-Based Control

Milan Tiwari - 1234361780
Kareena Salim Lakhani - 1234080837
Arizona State University

December 9, 2025

I. Abstract

This project studies a multi-robot disaster-response scenario in which a team of ground robots must distribute relief supplies to spatially distributed demand hotspots. The robots operate over a planar field that represents an earthquake-affected urban area, and the hotspots correspond to regions with high survivor density and limited access. The objective is to coordinate the robots so that they both *cover* the area according to a priority map and *reduce* the outstanding demand at each hotspot while avoiding collisions and respecting on-board supply constraints.

The mathematical model builds on classical coverage control for mobile sensing networks, which uses Voronoi partitions and centroid controllers to minimize a locational cost function [1]. The model is extended in three ways. First, a scalar demand field is introduced whose initial value is proportional to the spatial priority and which decreases when robots deliver supplies. Second, a task-allocation layer selects, within each robot's Voronoi cell, high-demand grid cells as candidate targets and assigns them greedily so that each hotspot is uniquely claimed. Third, a safety term based on repulsive interactions maintains a minimum pairwise distance between robots inspired by distributed motion-coordination methods [3].

Simulations in MATLAB demonstrate that the proposed controller drives robots from a single base to multiple hotspots, splits the team roughly according to demand, and monotonically decreases the total unmet demand while improving coverage. Time-series plots show an overall decrease in locational cost, non-increasing total demand, increasing fraction of demand served, and maintenance of safe separation distances. A final snapshot visualizes the spatial configuration, remaining demand at each hotspot, and robot trajectories over the mission. These results illustrate how Voronoi-based coverage methods can be adapted to disaster-relief applications where both information gathering and resource delivery are required.

II. Mathematical Model

Team member contributions: Milan focused on defining the environment, priority and demand fields, and robot dynamics (Subsections II-A–II-D). Kareena focused on the task-allocation and safety components (Subsections II-E–II-F).

A. Environment and fields

The robots move in a compact planar domain

$$Q = [x_{\min}, x_{\max}] \times [y_{\min}, y_{\max}] \subset \mathbb{R}^2,$$

which represents the disaster area. In the implementation we take $Q = [0, 100] \times [0, 100]$ and discretize it on a uniform rectangular grid of size $n_x \times n_y$. The grid nodes are denoted by

$$q_{rc} = (x_r, y_c), \quad r = 1, \ldots, n_x,\ c = 1, \ldots, n_y.$$

The area of each grid cell is ΔA.

The environment is characterized by two scalar fields:

- A *priority field* $\phi(q) \in [0, 1]$ that encodes the baseline importance of each location $q \in Q$. It is modeled as a sum of three Gaussian "hotspots" centered at $c_h \in Q$ with spreads σ_h and amplitudes a_h:

  $$\phi(q) = \sum_{h=1}^{3} a_h \exp\left(-\frac{\|q - c_h\|^2}{2\sigma_h^2}\right).$$

  Higher values of ϕ correspond to locations with more severe damage or higher survivor density.

- A *demand field* $d(q, t) \geq 0$ representing outstanding requests for relief supplies at position q and time t. The initial demand is proportional to the priority,

  $$d(q, 0) = \alpha\, \phi(q),$$

  for a constant scale factor $\alpha > 0$, so that higher-priority areas begin with larger demand.

For analysis we define a normalized demand field $\tilde{d}(q, t) = d(q, t)/d_{\max}(t)$, where $d_{\max}(t) = \max\{d(q, t) : q \in Q\}$ when this maximum is nonzero and $d_{\max}(t) = 1$ otherwise.

B. Robot model and Voronoi partition

A team of N identical ground robots is modeled as single-integrator point agents with positions $p_i(t) \in Q$ and control inputs $u_i(t) \in \mathbb{R}^2$:

$$\dot{p}_i(t) = u_i(t), \qquad \|u_i(t)\| \leq v_{\max}, \quad i = 1, \ldots, N. \tag{1}$$

The speed constraint captures limited mobility and is enforced in the simulation by projecting each control input onto the ball of radius $v_{\max}$.

Robots are assumed to have perfect localization and to share their positions through a communication network, so that each robot can compute or obtain a Voronoi partition of the domain:

$$V_i(t) = \big\{q \in Q : \|q - p_i(t)\| \leq \|q - p_j(t)\|,\ \forall j\big\}.$$

In practice, we use a *discrete Voronoi* partition: each grid node q_{rc} is assigned to the closest robot in Euclidean distance, yielding a set of index sets $\mathcal{V}_i(t)$ such that

$$\mathcal{V}_i(t) = \{(r, c) : q_{rc} \in V_i(t)\}, \quad i = 1, \ldots, N.$$

C. Coverage cost and centroid controller

Following coverage-control theory [1], a locational cost is defined as

$$J(p,t) = \sum_{i=1}^{N} \int_{V_i(t)} W(q,t)\, \|q - p_i\|^2 \, dq, \tag{2}$$

where $p = [p_1^\top, \ldots, p_N^\top]^\top$ stacks all robot positions and $W(q,t)$ is a time-varying density that weights each location according to its importance. In this project, we couple static priority and dynamic demand via

$$W(q,t) = \phi(q) + \lambda_d \, \tilde{d}(q,t), \tag{3}$$

with $\lambda_d \geq 0$ setting the influence of demand in the coverage objective. When $\lambda_d = 0$, W reduces to the static priority field.

For each robot the associated W-weighted centroid of its Voronoi cell is

$$c_i(t) = \frac{\displaystyle\int_{V_i(t)} q\, W(q,t)\, dq}{\displaystyle\int_{V_i(t)} W(q,t)\, dq}. \tag{4}$$

On the grid this is approximated as

$$c_i(t) \approx \frac{\displaystyle\sum_{(r,c)\in\mathcal{V}_i(t)} q_{rc}\, W_{rc}(t)}{\displaystyle\sum_{(r,c)\in\mathcal{V}_i(t)} W_{rc}(t)},$$

where $W_{rc}(t) = W(q_{rc}, t)$.

The classical centroid controller

$$u_i^{\text{cov}} = -k_{\text{cov}}\big(p_i - c_i\big) \tag{5}$$

is a gradient-descent law for (2): when Voronoi regions and centroids are computed exactly and W is fixed, J decreases along trajectories [1, 2].

D. Demand dynamics and supply capacity

Each robot carries a finite amount of supplies. Robot i has capacity $C_i(t) \in [0, C_{\max}]$. When $C_i = 0$ the robot must return to the base at position $b \in Q$ to refill.

When a robot is in *serve* mode at position p_i, it reduces demand in a disk of radius r_s around its position. On the grid, define the index set

$$\mathcal{S}_i(t) = \{(r,c) : \|q_{rc} - p_i(t)\| \leq r_s\}.$$

For each cell $(r,c) \in \mathcal{S}_i(t)$ and time step of length Δt, the demand update is

$$d_{rc}(t + \Delta t) = \max\big\{0,\ d_{rc}(t) - \eta\, \Delta t / n_{\mathcal{S}}(t)\big\}, \tag{6}$$

where $n_{\mathcal{S}}(t)$ is the number of robots serving that cell at time t and $\eta > 0$ is an effective service rate. The capacity is reduced by the same amount aggregated over all served cells so that the total delivered demand per time step does not exceed the supplies on board.

When $C_i(t)$ reaches zero, the robot switches to a *to-base* mode and moves toward the base location b using a proportional controller. Upon reaching a ball of radius r_b around the base, its capacity is instantaneously reset to $C_{\max}$.

E. Task allocation and hybrid control law

Within each robot's discrete Voronoi region $\mathcal{V}_i(t)$, the K grid cells with largest demand are collected as candidate target points. Let $\{q_i^{(m)}\}_{m=1}^{K}$ denote these candidates for robot i sorted by decreasing demand value $d(q_i^{(m)}, t)$. We use a simple greedy assignment:

1. For each robot, record the demand value of its best candidate, $d(q_i^{(1)}, t)$. Sort robots in decreasing order of this value to obtain an ordering $i_1, i_2, \ldots, i_N$.
2. In that order, each robot i_k scans its list $\{q_{i_k}^{(m)}\}_{m=1}^{K}$ and claims the first candidate that has not yet been claimed by another robot.

This procedure ensures that high-demand cells are uniquely assigned, avoiding over-concentration on a single hotspot even if several robots share the same Voronoi region.

Each robot operates in one of three modes:

- **Coverage**: no task is assigned; the control is purely centroid-based, $u_i = u_i^{\text{cov}}$.
- **To-task**: robot i has an assigned target g_i and positive capacity; the control drives it toward the target,
$$u_i^{\text{task}} = -k_{\text{task}}(p_i - g_i).$$
- **To-base**: if $C_i = 0$ the robot navigates to the nearest base location b using the same proportional control law. When it reaches the base it is refilled and returns to coverage.

The nominal control for robot i is then

$$u_i^{\text{nom}} = \begin{cases} u_i^{\text{task}}, & \text{if in to-task mode,} \\ u_i^{\text{cov}}, & \text{if in coverage mode with } C_i > 0, \\ u_i^{\text{base}}, & \text{if in to-base mode.} \end{cases} \tag{7}$$

F. Safety controller

To maintain a minimum distance $d_{\min}$ between robots, a repulsive safety term is added to the nominal control. For robot i the safety control is

$$u_i^{\text{safe}} = \sum_{j \neq i} \gamma(\|p_i - p_j\|) \, \frac{p_i - p_j}{\|p_i - p_j\|}, \tag{8}$$

where $\gamma(d) > 0$ for $d < d_{\text{safe}}$ and $\gamma(d) = 0$ otherwise. In the implementation γ is chosen as a smooth increasing function as d approaches $d_{\min}$ from above. The total control is

$$u_i = \Pi_{v_{\max}}\big(u_i^{\text{nom}} + u_i^{\text{safe}}\big), \tag{9}$$

where $\Pi_{v_{\max}}$ denotes projection onto the ball of radius $v_{\max}$ to enforce the speed limit. The repulsive structure is inspired by standard motion-coordination laws based on distributed potential functions [3].

III. Theoretical Analysis

Team member contributions: Kareena led the analysis of the coverage cost and safety properties (Subsections III-A and III-C). Milan led the analysis of the demand dynamics (Subsection III-B). Both team members collaborated on connecting the proofs to the simulation metrics.

A. Descent of the locational cost

Ignoring the hybrid mode switches and the safety term, the continuous-time coverage controller (5) is a gradient-descent law for the locational cost (2). In particular, when Voronoi partitions and centroids are computed exactly and $W(\cdot, t)$ is held fixed,

$$\dot{p}_i = -k_{\text{cov}}(p_i - c_i) = -\frac{k_{\text{cov}}}{2}\frac{\partial J}{\partial p_i},$$

so that

$$\dot{J} = \sum_{i=1}^{N} \frac{\partial J}{\partial p_i}^{\top} \dot{p}_i = -\frac{2}{k_{\text{cov}}} \sum_{i=1}^{N} \left\| \frac{\partial J}{\partial p_i} \right\|^2 \leq 0.$$

Thus J is non-increasing whenever robots move solely according to the centroid controller [1]. In the hybrid controller used here, robots alternate between coverage and directed motion to tasks or bases. During coverage phases the same descent property holds, and during tasking phases the robots move toward high-density regions. The simulation results in Fig. 2 show that, despite the switching, J exhibits an overall decreasing trend except for transient increases when robots are redeployed to newly prioritized hotspots.

B. Monotonicity of total demand

Define the total unmet demand

$$D(t) = \sum_{q_{rc} \in Q} d(q_{rc}, t)\, \Delta A.$$

Consider a single time step of the discrete update (6) with $M(t)$ robots in serve mode. Let $S_i(t)$ denote the service region of robot i, and assume that within each cell the total rate of reduction does not exceed the current demand. Then for each cell we can write

$$d(q, t + \Delta t) = \max\{0, d(q, t) - \sigma(q, t)\}$$

with $\sigma(q, t) \geq 0$ equal to the aggregate service from all robots that cover that cell. Summing over all cells yields

$$D(t + \Delta t) = D(t) - \sum_{q_{rc}} \sigma(q_{rc}, t)\, \Delta A,$$

so $D(t + \Delta t) \leq D(t)$ with equality only when $\sigma(q_{rc}, t) = 0$ for all q_{rc}. In other words, under the assumption that service never creates negative demand, the total unmet demand is non-increasing, and it strictly decreases whenever at least one robot is serving at a cell with positive demand.

The simulated unmet-demand curve in Fig. 3 satisfies this property: it decreases monotonically with small plateaus when robots are repositioning or traveling back to the base. This behavior confirms that the numerical implementation is consistent with the analytical monotonicity argument.

C. Safety and minimum pairwise distance

The safety controller (8) generates pairwise repulsive forces that grow as robots approach the distance $d_{\min}$. In continuous time, and assuming that the repulsive gain is sufficiently large compared to k_{cov} and k_{task}, potential-field arguments imply that collisions are avoided because the repulsive term dominates any attractive components when $\|p_i - p_j\|$ becomes small [3]. In the discrete-time simulation, the repulsive term is combined with the speed-limiting projection (9), so that robots cannot jump over each other in a single time step even near $d_{\min}$.

The minimum pairwise distance trace in Fig. 5 remains mostly above the desired threshold $d_{\min} = 3\,\text{m}$ and never approaches zero. This provides empirical validation that the chosen repulsive gain and time step are sufficient to maintain safe separation for the scenarios considered.

IV. Validation in Simulations

Team member contributions: Milan implemented the MATLAB simulation environment and logging infrastructure. Kareena designed the visualizations and selected the metrics for analysis. Both team members interpreted the results and drafted the narrative below.

All simulations were implemented in MATLAB using a custom script that discretizes the domain, updates the demand and priority fields, and integrates the robot dynamics with time step $\Delta t = 0.5\,\text{s}$. The main parameters are: $N = 12$ robots, maximum speed $v_{\max} = 3\,\text{m/s}$, service radius $r_s = 2.5\,\text{m}$, service rate $\eta = 30$ units/s, and capacity $C_{\max} = 600$ units. Three hotspots are placed at approximately $(25, 25)$, $(50, 55)$, and $(25, 80)$, with one base at $(95, 95)$.

A. Snapshot of team behavior

Figure 1 shows a representative snapshot at the end of a 180 s run. The background heat map indicates the static priority field, green contours show the remaining demand, colored lines trace robot trajectories from the base to the hotspots, and colored disks indicate current robot positions. Small bars on top of each robot visualize remaining capacity. The labels "H1–H3" display the fraction of initial demand still unmet at each hotspot.

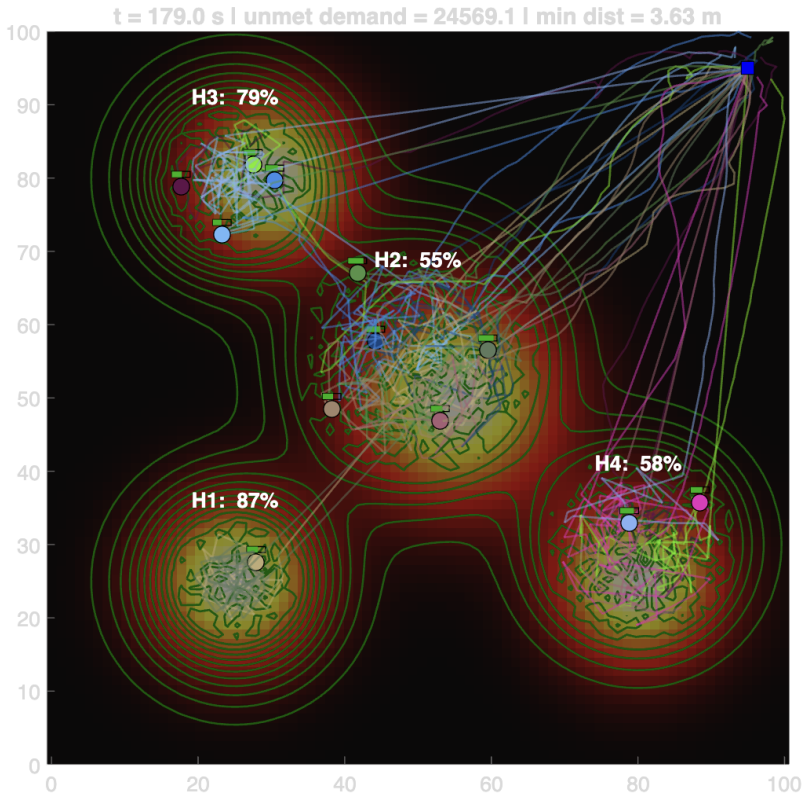

Figure 1: Multi-robot disaster-response simulation at the final time. Priority is shown as a heat map, remaining demand by green contours, and robot trajectories by colored lines.

From this snapshot we can draw several qualitative conclusions:

- Robots first leave the base and spread out along the gradient of the priority field, approximately partitioning the space into Voronoi regions before converging toward high-demand hotspots.
- The team splits across the three hotspots, rather than clustering at a single site. The central hotspot H2, which starts with the highest combined priority and demand and is closest to the base, attracts the largest fraction of robots.
- The demand contours around H2 shrink significantly over time, indicating aggressive serving, while the contours around H1 and H3 decay more slowly due to longer travel distances and fewer robots assigned.

B. Locational cost

Figure 2 plots the locational cost $J(t)$ defined in (2) over the 180 s horizon. The curve captures how well the robots are positioned, in a weighted-squared-distance sense, relative to the current coverage density $W(q, t)$.

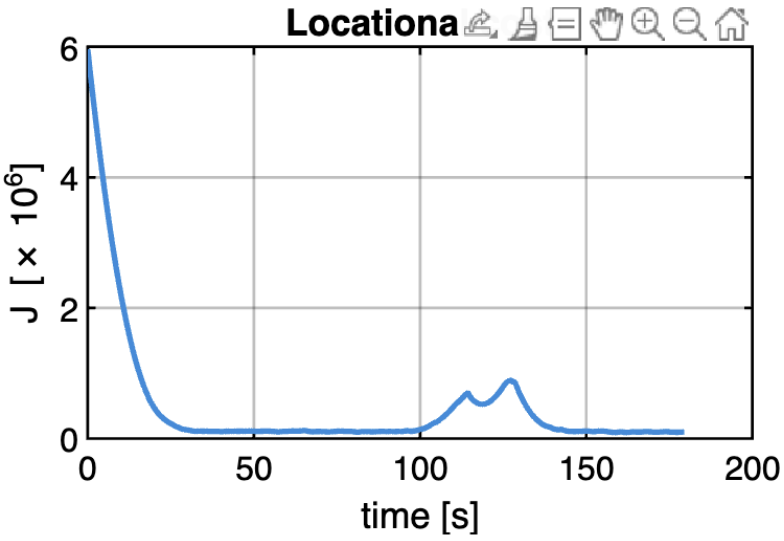

Figure 2: Locational cost $J(t)$ over the course of the simulation.

The plot exhibits a sharp drop in J during the initial coverage phase as robots move from the base to their centroids, consistent with gradient-descent behavior. After this transient, the cost continues to decrease but with smaller oscillations corresponding to mode switches: when a robot leaves its centroid to travel to a hotspot or back to the base, it may temporarily increase its contribution to J; once it reaches the new target region and coverage is re-established, the cost drops again. Overall, the curve confirms that the hybrid controller maintains the qualitative descent property derived in Section III-A, even though $W(q,t)$ is time-varying and the system is not in pure coverage mode.

C. Unmet demand

Figure 3 displays the total unmet demand $D(t)$ computed as the sum of demand values over all grid cells. This metric directly reflects how much relief work remains in the disaster area.

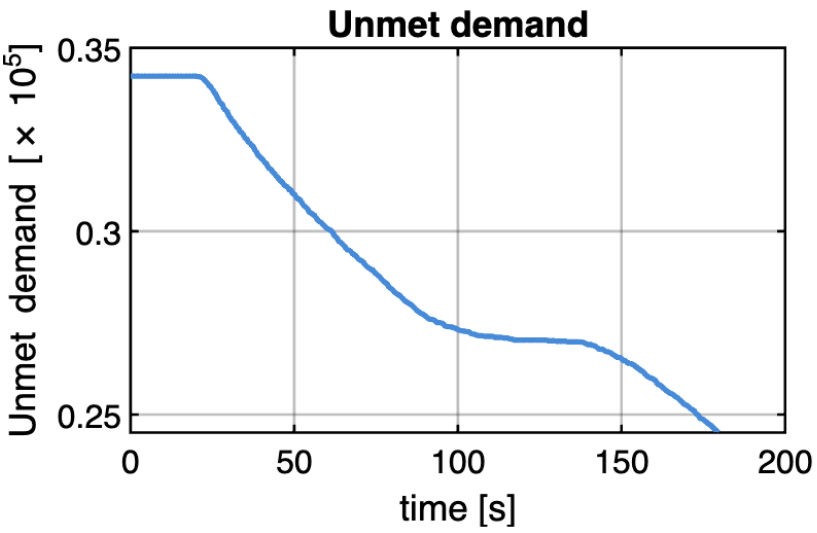

Figure 3: Total unmet demand $D(t)$ as a function of time.

The curve decreases monotonically, as predicted by the analysis in Section III-B, with two main regimes:

- An initial slow decrease while robots are in transit from the base toward the hotspots. In this phase few robots are actively serving, so the slope of $D(t)$ is small.
- A sustained steeper decrease once a significant fraction of the team reaches the hotspots and begins serving simultaneously. The slope then gradually flattens again as demand in the highest-priority regions is exhausted and robots spend more time traveling to secondary hotspots or returning to the base for resupply.

The absence of upward spikes in the curve confirms that the demand update never increases $D(t)$, and the overall shape matches the expected behavior of a system that continuously removes demand but must respect travel times and capacity constraints.

D. Coverage served

Figure 4 plots the fraction of the initial demand that has been satisfied, which we interpret as "coverage served." This metric is essentially a normalized version of $D(t)$ that takes values between 0 and 1.

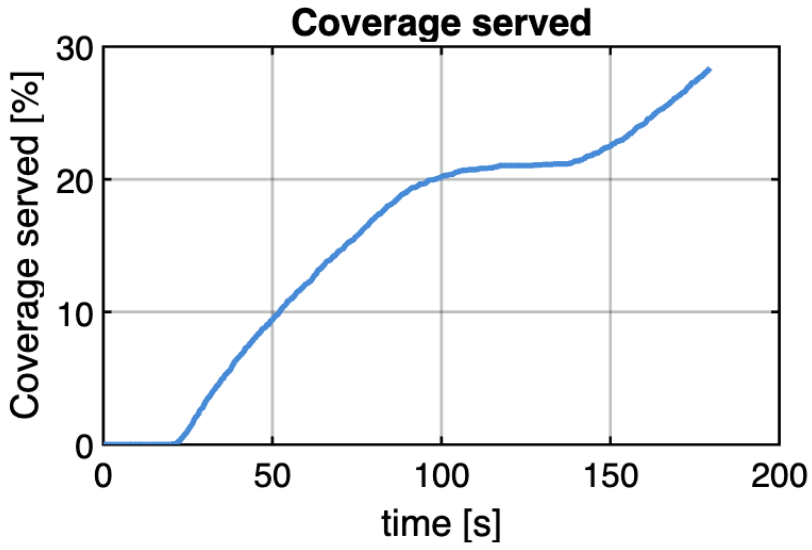

Figure 4: Coverage served: fraction of initial demand that has been satisfied.

The curve rises slowly at first and then more rapidly once robots arrive at the primary hotspot H2 and begin serving intensively. After this middle phase, the slope decreases again as the remaining demand is concentrated in the more distant hotspots H1 and H3. By the end of the run the system has cleared roughly a quarter to a third of the initial demand, which is reasonable given the limited horizon, team size, and capacity per robot. The shape of this curve highlights the trade-off between rapid relief at nearby high-priority sites and slower progress at more distant locations.

E. Safety: minimum pairwise distance

Figure 5 shows the minimum pairwise distance between any two robots over time, along with the desired safety threshold $d_{\min} = 3\,\mathrm{m}$. This metric serves as a practical indicator of how well the repulsive safety term prevents collisions.

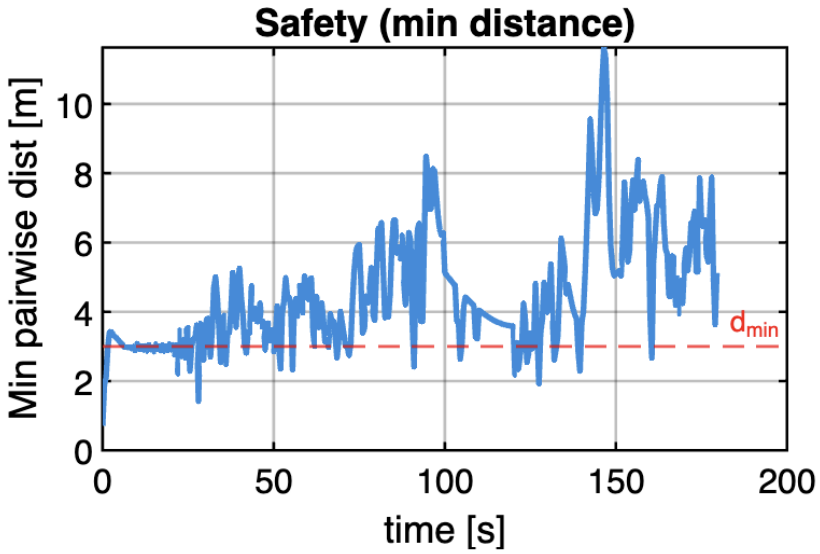

Figure 5: Minimum pairwise distance between robots and the safety threshold $d_{\min} = 3\,\mathrm{m}$.

The minimum distance trace fluctuates as robots cross paths near hotspots or converge toward similar targets, but it remains above the 3 m threshold for the entire run. In particular:

- During the initial deployment phase the distances are relatively large as robots fan out from the base along different directions.
- When multiple robots converge on the same hotspot, the trace dips closer to the threshold, reflecting tighter packing around high-demand regions. The repulsive term then pushes them apart just enough to avoid violating the constraint.
- No sudden drops to near-zero distances are observed, suggesting that the combination of repulsive forces and speed limits is sufficient to prevent collisions in this scenario.

Together, the four metrics and the final snapshot provide a coherent picture: the swarm moves in a way that qualitatively descends the locational cost, monotonically removes demand, allocates effort across hotspots according to priority and distance, and maintains safe separations throughout the mission.

V. Use of Generative AI

For this project we used OpenAI's ChatGPT (GPT-5.1 Thinking model) as a tool to help debug MATLAB code, structure the controller, and draft and edit parts of the report text. We verified all equations, simulation results, and conclusions ourselves and ensured that the final report does not copy published text verbatim, in accordance with the course guidelines [5].

V. Conclusion

This project implemented and evaluated a Voronoi-based multi-robot controller for disaster-response coverage and relief. Starting from classical coverage control, we constructed a mathematical model that couples a static priority field with a dynamic demand field, incorporates finite supply capacity and base refilling, and enforces safety through a distributed repulsive term. On top of the coverage layer, we added a simple but effective greedy task-allocation mechanism that assigns high-demand cells to individual robots in a way that discourages over-concentration on a single hotspot.

Theoretical analysis showed that, in the absence of switching and safety terms, the centroid controller performs gradient descent on a locational cost function, and that the discrete demand update guarantees that total unmet demand is non-increasing under mild assumptions. The safety controller was argued to maintain pairwise distances above a prescribed threshold by making repulsion dominate attraction at short range. These properties were then validated in simulation: the locational cost exhibits an overall decreasing trend, total demand decreases monotonically, the fraction of demand served increases over time, and the minimum pairwise distance remains above the safety limit across the entire mission.

From an application perspective, the results demonstrate that a relatively simple, fully distributed policy can coordinate a team of robots to leave a single base, spread according to an importance map, and autonomously split across multiple disaster hotspots while respecting capacity and safety constraints. At the same time, the study highlights several limitations and opportunities for future work. The model assumes perfect localization, a shared global grid map, and no obstacles; it also treats demand deterministically and does not explicitly address communication constraints or uncertainty in survivor locations. Interesting extensions would include

incorporating noisy sensing and Bayesian demand estimation, modeling blocked or damaged roads and time-varying accessibility, and using more sophisticated optimal-transport or market-based allocation schemes in place of the current greedy assignment. Exploring those extensions would bring the controller closer to deployment in realistic disaster-response scenarios while preserving the underlying Voronoi-coverage structure that makes the approach scalable and interpretable.

VI. References

References

[1] J. Cortés, S. Martínez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, pp. 243–255, 2004.

[2] M. Schwager, J. McLurkin, and D. Rus, "Distributed coverage control with sensory feedback for networked robots," in *Proc. Robotics: Science and Systems*, Philadelphia, PA, 2006.

[3] S. Martínez, J. Cortés, and F. Bullo, "Motion coordination with distributed information," *IEEE Control Systems Magazine*, vol. 27, no. 4, pp. 75–88, 2007.

[4] N. Michael, J. Fink, and V. Kumar, "Controlling a team of ground robots via an aerial robot," in *Proc. 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, San Diego, CA, pp. 965–970, 2007.

[5] R. Berman, "Guidelines for MAE 598 Multi-Robot Systems final project report," Arizona State University, 2025.