Blog Post 4 of: Divide and Conquer: Optimized Drone Coverage for Complex Disaster Zones

In our last post, we covered planning for energy efficiency. But what happens when the disaster area itself is extremely complex? An urban earthquake scenario isn't a simple open field; it's a complex polygon with "holes" (undamaged areas) and "no-fly zones" (e.g., restricted areas for evacuation helicopters).

To handle this, we developed a different approach: **Optimized Coverage Path Planning (CPP) Leveraging Drone Capabilities**.

This method uses a two-step process to efficiently cover non-convex regions.

Step 1: Decompose the Area Based on Drone Capabilities First, the algorithm takes the complex polygon of the interest area and a list of available drones. Each drone has a different capability, such as the total surface area it can cover on a single battery.

Our algorithm then uses a "bottom-up approach" to divide the complex area into partitions that are *sized* to match each drone's capability. The key objective is to maximize the "compactness" of each partition. A more compact (less "stringy") shape is much more efficient to fly.

As seen in Figure 8, we compared our bottom-up algorithm to two other methods (IHL and DARP). Our method consistently produced partitions with better "normalized compactness" and scaled more effectively.

Step 2: Plan the Trajectory for Each Partition Once each drone has its own "compact" partition, the planning algorithm gets to work.

- It transforms the partition into a **grid of points**. These points are spaced based on the drone's flight altitude and camera sensor (FoV) to ensure proper photo overlap.
- It then generates a "back-and-forth" trajectory to visit all "pass points" in the grid.
- Crucially, if a "no-pass point" is encountered (e.g., it's inside a "hole"), the algorithm makes a detour to the next valid point.
- It also intelligently avoids no-fly zones and prevents the drone from flying over a neighboring partition assigned to another drone.

The Result: A Faster, More Efficient Scan We benchmarked our complete method against other common path planning techniques. As shown in Table 1, our proposed CPP method resulted in a **shorter flight time** and **shorter flight distance** to cover the same complex area with a hole. This means we get a complete scan of the disaster zone, faster and more efficiently.

In our final post, we'll tie all these methods together and show how they are aligned with the PANTHEON project's real-world use cases.