  - Move everything on the table to the desk (should move the laptop, pencil, pen, plate, credit card, book, and newspaper)
  - Slice the lettuce, trash the mug and switch off the light
  - Put all electronics on the couch
  - Make a dish by microwaving eggs and tomato
  - Put all readable objects on the sofa
  - Wash all fruits
- **Implicit target and object types**: Here both the object type and the target are implicitly defined. E.g. `Clear the floor by placing the items at their appropriate positions.` Here the model is expected to keep items like pens, book, laptop on the study table, litter in the trash can, etc.
  - Clear the floor by placing items at their appropriate positions (depending on what's on the floor)
  - Clear the table by placing the items in their appropriate positions (depends on the floorplan, e.g. bread, apple, tomato, knife, bowl, book)
  - Clear the countertop by placing items in their appropriate positions (should move the lettuce, mug, and paper towel roll)
  - Clear the desk by placing the items in other appropriate positions (should move the statue, watch, and remote control)
  - Clear the table by placing the items in other appropriate positions (should move the book, credit card, laptop, plate, newspaper, pen, and pencil)
  - Clear the couch by placing the items in other appropriate positions (should move the pillow)
  - Make the living room dark
  - Make a mug of coffee and toast the bread
  - Trash all groceries
  - Slice all sliceable objects

D Search & Rescue Environment (SAR)

The Search & Rescue environment consists of multiple agents in an unknown environment that has multiple wildfires and missing personnel in the environment. The agents are tasked to extinguish all the fires before they spread and rescue the missing humans. Here, each fire is composed of a large flammable region with a fixed set of sources that spread through time. The higher the intensity, the faster the fire will spread. The fires can be of class A or B, extinguished through the use of water and sand respectively. Both these resources can be collected through resource reservoirs spread geographically. Each person is initially stranded in an unknown location. The goal is to rescue and transport each person to a drop-off location (known apriori). The person must have two agents simultaneously carrying them to be transported to the drop-off location.

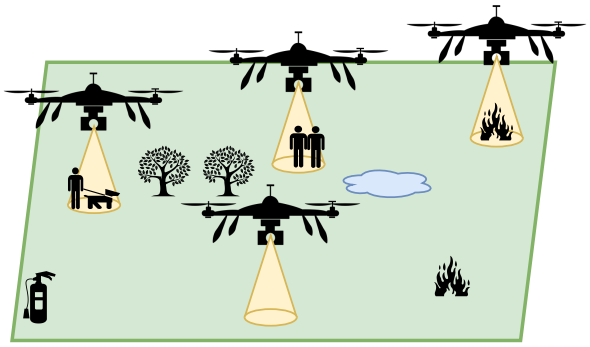

Figure 5: The search & rescue environment consists of multiple drones in an unknown environment with missing people, fires of different types, and water and sand reservoirs.

We provide a justification to why this task is relevant to long-term planning and multi-agent collaboration below.

D.1 Multi-agent Nature

- **Scalability** - Unlike the AI2THOR environment, the search & rescue environment is more spacious and hence prevents congestion amongst agents. With the addition of multiple parallel emergency disaster scenarios at once, the environment scales comfortably to more agents. Additionally, the complexity slowly increases as the model has to coordinate between more actors and competing emergencies.
- **Tasks requires explicit cooperation** -
  - To move a group of humans, at least 2 agents are required. Thus, explicit multi-agent collaboration is required.
  - Due to the time pressure, in order to successfully stop the fire, agents must all effectively collaborate in collecting and strategically using resources in high-intensity regions.
- **Exploration & Task Assignment** - There is an explicit tradeoff between allocating agents towards exploring to find lost people and fighting the current fires.

D.2 Long-term Planning

- **Dependency Chain**: To resolve a fire it requires that the source is identified, the type of fire is identified, and the appropriate resources to stop it are acquired and then used.
- **Uncertainty**: Inherent uncertainty as to where the lost people can be found, and at what point in the timeline they will.
- **Irreversibility and forced consequences** - With a fire that spreads, present actions have irreversible future consequences. A balance needs to be struck between fighting the fire's source (to stop it from continuing) versus a periphery (to prevent geographic spread).

D.3 Description of Scenes

We evaluate LLaMAR in five different scenarios in the search & rescue environment. Here each scene is evaluated on 5 different random seeds.

- **Scene 1**: This consists of 1 Type A fire with 2 initial sources, 1 Type B fire with 1 initial source, and 1 lost person at a random location in the environment.
- **Scene 2**: This consists of 1 Type A fire with 1 initial source, 1 Type B fire with 2 initial sources, and 1 lost person at a random location in the environment.
- **Scene 3**: This consists of 2 Type A fires each with 1 initial source, 1 Type B fire with 1 initial source, and 1 lost person at a random location in the environment.
- **Scene 4**: This consists of 1 Type A fire with 3 initial sources, and 1 lost person at a random location in the environment.
- **Scene 5**: This consists of 1 Type A fire with 1 initial source, 1 Type B fire with 1 initial source, and 2 lost persons at random locations in the environment.

D.4 Observation Space

The agent's observation $\mathcal{O} = \mathcal{O}_G \cup \mathcal{O}_L \cup \mathcal{O}_L$ is a union of the following observations.

Global Observations $\mathcal{O}_G$ consists of all the objects that are globally visible by the agent:

- `Fires`: If visible, fire name, type, and average intensity.
- `Fire Regions`: If fire visible, fire region name, type, and average intensity.
- `Reservoir`: If visible, reservoir name, and reservoir resource type.
- `Deposit`: If visible, deposit name, inventory of all the resources (water, sand) and persons in deposit.
- `Person`: If visible, person name, carried or not status, dropped-off or not status.
- `Agent inventory`: List of all the resources (water, sand) and person (being carried) in the agent's inventory.

Local Observations $\mathcal{O}_L$ consists of all the objects that are visible in grid cells adjacent to the agent:

- For all the directions `<direction>` in `(Up, Down, Left, Right, and Center)`, output of the following.

- `<direction>`: At direction `<direction>`, either 'Empty' if there is no object, 'Flammable' along with corresponding intensity & fire name if object is a part of a fire, or 'Obstacle' for any other object.

Names $\mathcal{O}_N$ consists of a list of the name of all visible and interactable objects.

D.5 Action Space

The action space $\mathcal{A}$ consists of navigation actions $\mathcal{A}_{NAV}$, interaction actions $\mathcal{A}_{INT}$, exploration action $\mathcal{A}_{EXP}$.

Navigation actions $\mathcal{A}_{NAV}$ consists of the following actions:

- `Move(<direction>)`: Moves the agent to the neighboring grid cell in the specified direction where `<direction>` can be one of `(Up, Down, Left, Right, and Center)`
- `NavigateTo(<targetID>)`: Moves the agent next to the location of the object `<targetID>` if `<targetID>` is visible.

Interaction actions $\mathcal{A}_{INT}$ consists of the following actions:

- `Carry(<person>)`: Makes agent carry a person `<person>` if `<person>` is visible and interactable. The person is successfully 'group' carried, if at least the required number of agents successfully does the '`Carry(<person>)`' action. If carry action is successful, all other resources in agent's inventory are dropped.
- `DropOff(<person>, <deposit>)`: Drops off person `<person>` at location `<deposit>`. This action is only successful if the person has been 'group' carried, all the agents carrying the person have the deposit be visible and interactable, and after all the agents do this action.
- `StoreSupply(<deposit>)`: Stores all the resources from the agent's current inventory in the deposit `<deposit>`.
- `UseSupply(<fire>, <supply-type>)`: Uses all the supplies of type `<supply-type>` on the fire `<fire>` at the location the agent is at.
- `GetSupply(<deposit>, <supply-type>)`: Fills the remaining of the agent's inventory space with the available supply of type `<supply-type>` in `<deposit>`.
- `GetSupply(<reservoir>)`: Collects 1 unit of supply from reservoir `<reservoir>` and stores it in the agent's inventory.

Exploration action $\mathcal{A}_{EXP}$ consists of the following actions:

- `Explore()`: Takes the agent in a direction of exploration as described by the heuristic exploration function described in algorithm 2.

Algorithm 2 SAR Exploration Heuristic

Input: Agent A, Environment env, Previous Direction D, Current Position P, Max steps M
Initialize: New direction $ND \leftarrow \emptyset$

$ND \leftarrow$ randomly choose an angle from $[0, 2\pi)$ at $\pi/4$ intervals
while $ND = D$ **or** $ND \equiv D + \pi \mod 2\pi$
    or a barrier exists at most $\frac{3}{4}M$ steps in direction ND from P **do**
    $ND \leftarrow$ randomly choose an angle from $[0, 2\pi)$ at $\pi/4$ intervals
end while
Move agent A, M steps in direction ND in environment env