Fire Commander: An Interactive, Probabilistic Multi-agent Environment for Heterogeneous Robot Teams-Tutorial Slides

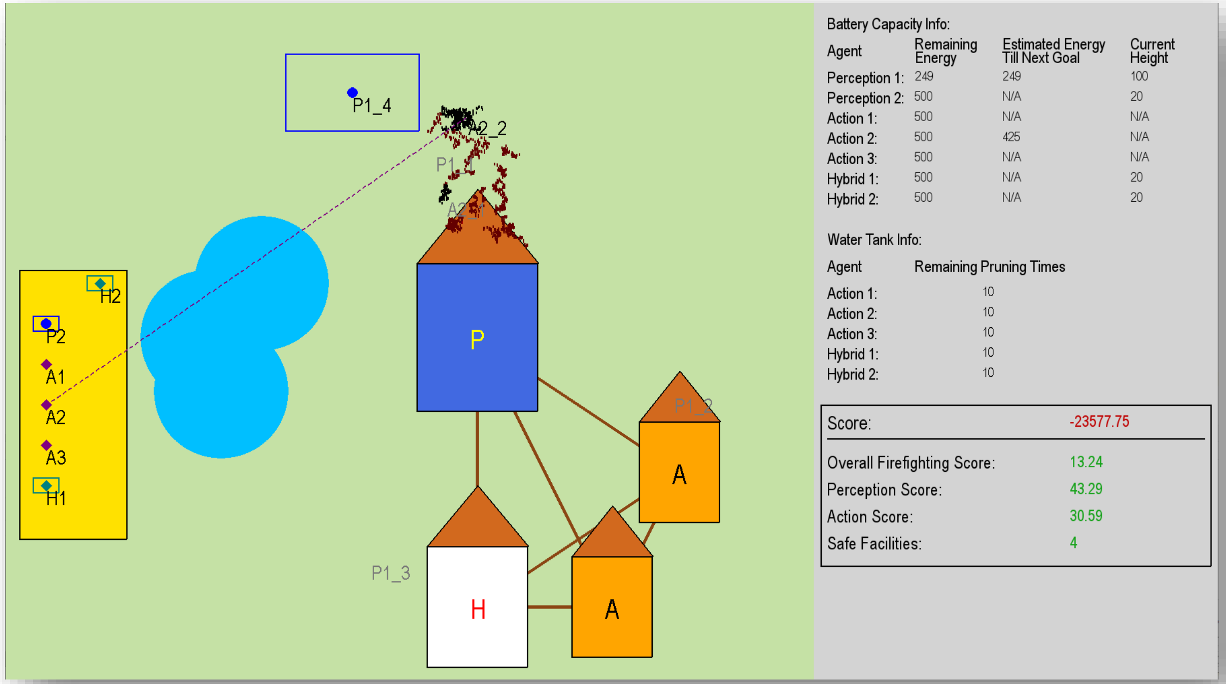
Esmaeil Seraj*, Xiyang Wu and Matthew C. Gombolay

Institute for Robotics & Intelligent Machines (IRIM)
Oct. 27th, 2021

(PPT Documentation – Version 2.1)

FireCommander

- **FireCommander:**
  - A multi-agent strategic game
  - An interactive probabilistic joint Perception-Action reconnaissance environment
- **Objective (our purpose):**
  - Heterogeneous Coordination
  - Learning Heterogeneous Communication
  - Multi-agent Learning from Heterogeneous Demonstrations
- **Other Applications:**
  - Multi-agent Planning, Scheduling, Task Assignment
  - Teaming
  - Wireless Sensors and Actor Networks
  - Psychology
  - HRI (single or multi-agent scale)
  - etc.

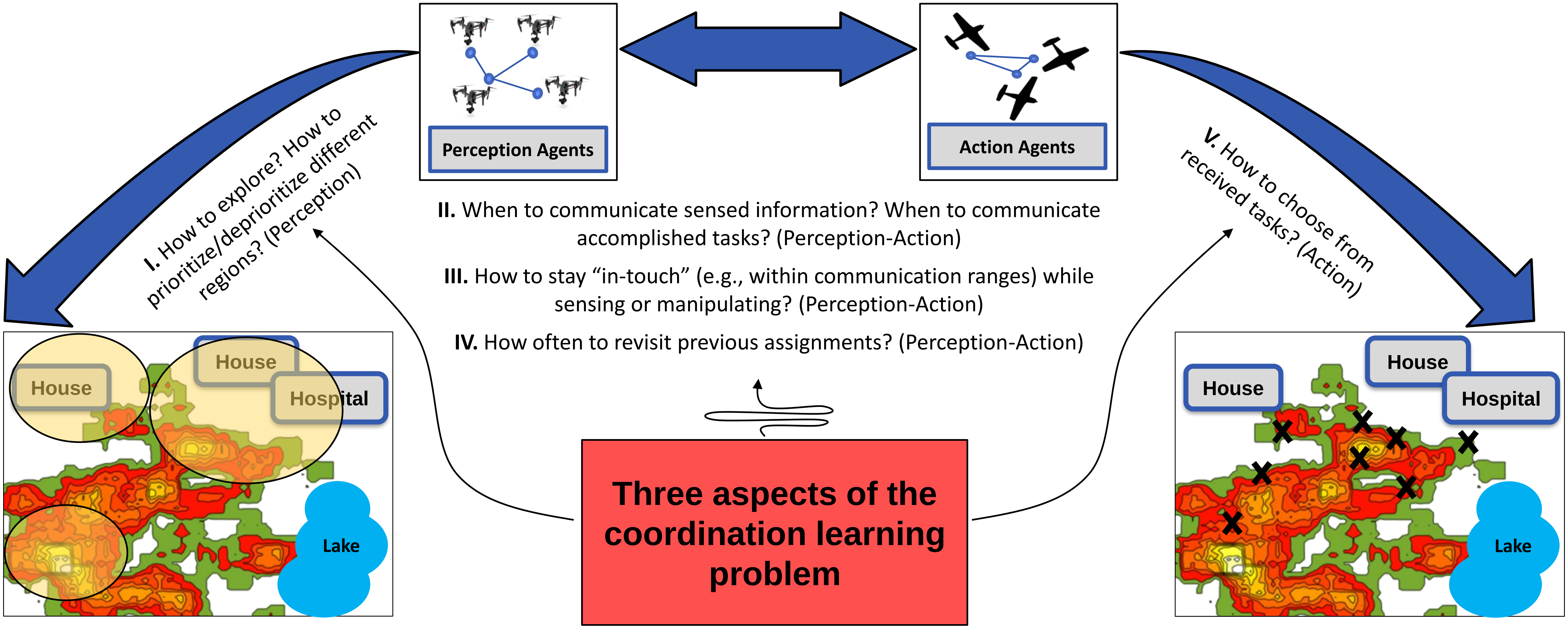


Esmaeil Seraj, Xiyang Wu, Matthew Gombolay

FireCommander

- **FireCommander: The Environment/Game Overview**
  - A *composite* robot team: Perception-only and Action-only
  - Fire propagates and there are some facilities/targets. Fire must be put out and targets must be kept safe
  - To put fire out:
    1) Firespots must be discovered by a Perception agent
    2) Perception agent must communicate the sensing information to an Action agent
    3) An Action agent with information about fire location puts fire out
  - Stochasticity in the game:
    - Stochasticity of Fire: Fire can appear at anytime during the game, anywhere on the map. Fire propagation model is stochastic.
    - Stochastic Sensing: Perception uncertainty varies with agent's altitude. Altitude has direct and reversed relation with observable area (FOV) and sensing quality (uncertainty), respectively.
    - Stochastic Action: There is a confidence coefficient associated with Action agents that determines the % of spots an Action agent can put out in each try.
  - Environment Challenges:
    - Constrained Communication: Step (2) requires agents to be within each other's communication range
    - Multi-objective game, Partially Observable, Uni-task Robots
    - Agents have limited velocities, battery limits, limited tanker capacity (Action Agents) and motion restrictions (Action agents cannot be left idle do not have UP/DOWN in their action space)

FireCommander



FireCommander

- **FireCommander: What do we have?**
  - Environments:
    - Env for Multi-agent Reinforcement Learning: *ready for a MARL algo to be applied on*
      1. *Simple_Env*: Only the Perception part of the game (Homogeneous version of the game)
      2. *Complex_Env#1*: Everything except for the targets/facilities (Cooperative Stochastic/Markov Games: Shared Reward and Objective)
      3. *Complex_Env#2*: Full version (Non-Cooperative Stochastic/Markov Games: different **&** Shared reward functions and objectives)
    - Env for LfD and HRI: *must be interactive (GUI) to record user data*
      1. *Predesigned_Scenarios*: Like missions in a strategic game
      2. *Open-World_Mode*: For instance to design environment with heavy/light workload and check expert's policy design efficiency
  - Documentations and Tutorials
    - Code and GitHub Blog: ………………………………………………………………………………….. https://github.com/EsiSeraj/FireCommander2020
    - A full documentation on arXiv: …………………………………………………………………………………………….. https://arxiv.org/abs/2011.00165
    - PPT Slides with visuals descriptions: ……………………………………………………………………….. https://hal.archives-ouvertes.fr/hal-02995093v1

Developing Package

- **Simulation Environment: PyGame**
  - Popular video game design framework, mature community
  - Simple but well-designed module, satisfy the demand for interactive visual environment
  - Portable and small amount code, fast response

- **User Interface: PyGt 5**
  - One of the most common GUI design tool
  - Well-designed functions and controllers, modular design
  - Easy to integrate with other packages, including PyGame

Targets/Facilities

Lake

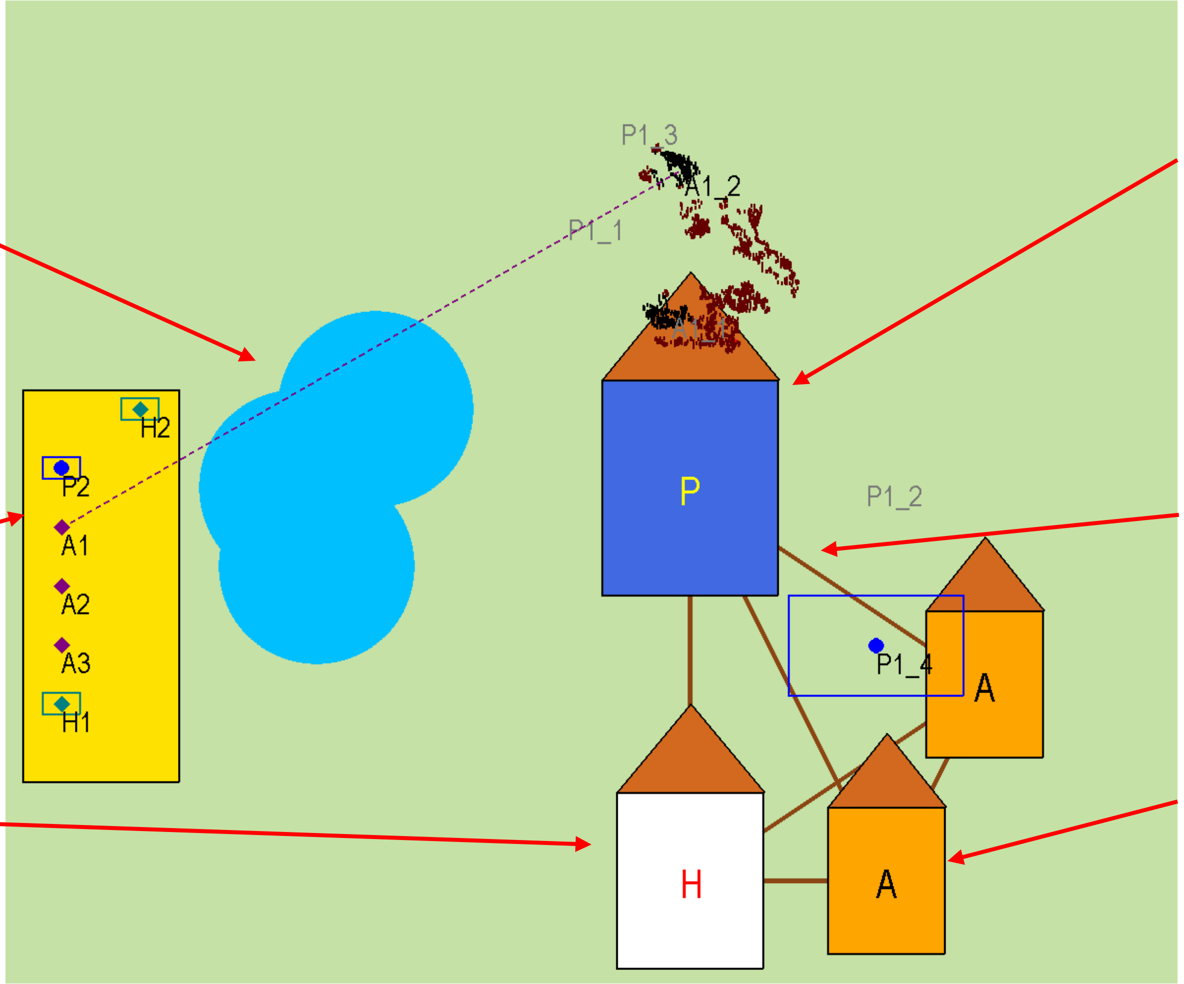
- At most 5 in Each Scenario
- Composed by 3 100-pixel Circles
- Relative Position for Each Circle Center: (0, 0), (80, -80), (20, 80)
- Importance: **Low, Resistive to fire**

Agent Base (Vertical)

- Unique in Each Scenario, Close to the Edge of the Scenario
- Geometry: 160 × 400
- Capacity: 9 Agents
- Importance: **Very High**

Hospital

- At Most 5 in Each Scenario
- Geometry: 150 × 180
- Importance: **High**



Power Station

- At Most 5 in Each Scenario
- Geometry: 180 × 220
- Importance: **Very High**

Road

- Connect the Center of Each Target
- Line Width: 5
- Importance: **Low, Resistive to fire**

House

- At Most 5 in Each Scenario
- Geometry: 120 × 150
- Importance: **Medium**

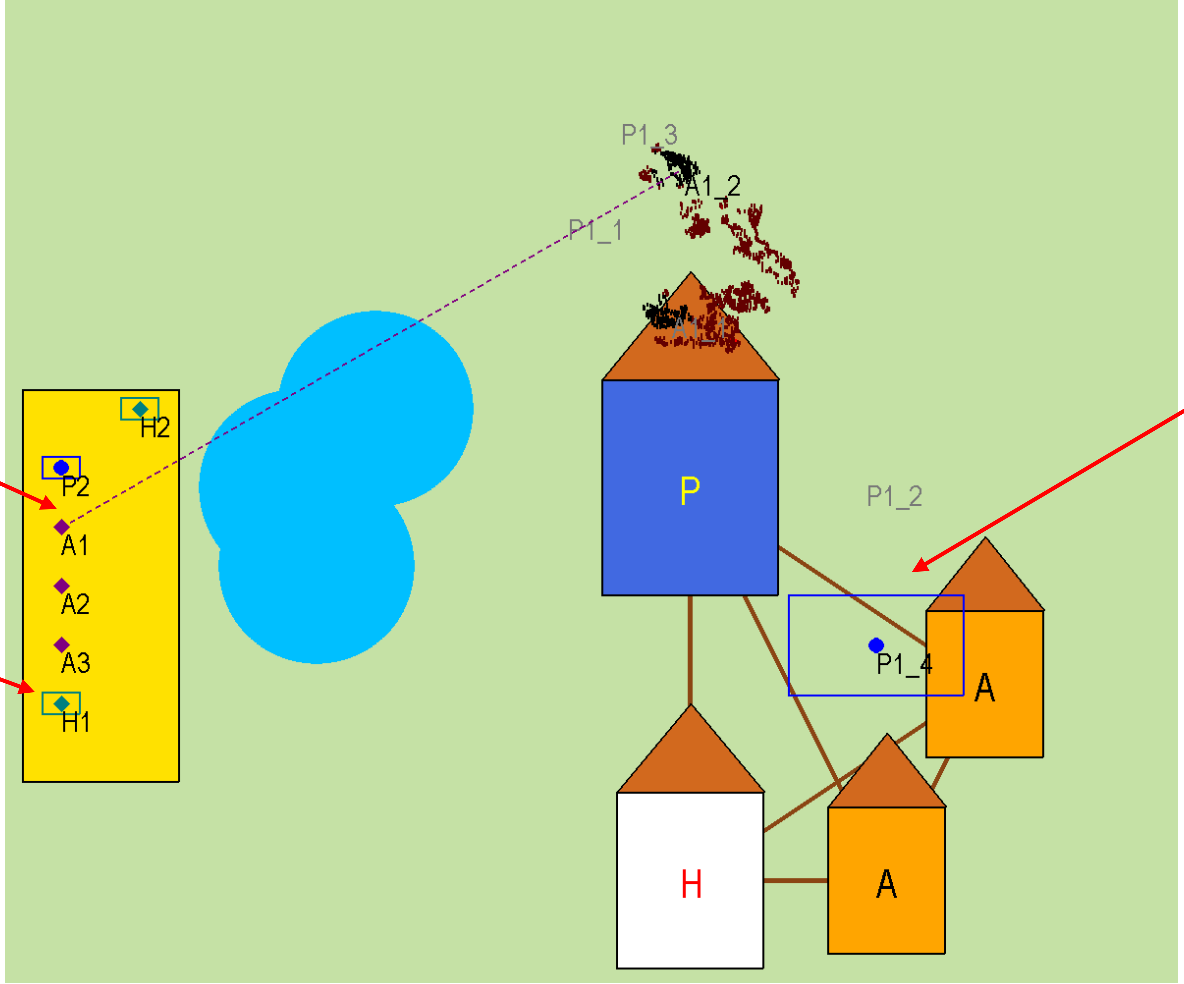
Agents

Action Agent (A)

- Purple diamond, No Visible Scope
- Default Pruning Height: 30 m (Unchangeable)
- Field of View: [π/6, π/4]

Hybrid Agent (H)

- Cyan diamond, Rectangle Scope
- Default Flight Height: [10, 100]
- Default Pruning Height: 20 m (Unchangeable)
- Field of View: [π/6, π/4]



Perception Agent (P)

- Blue diamond, Rectangle Scope
- Default Flight Height: [10, 100]
- Field of View: [π/6, π/4]

Switch and Control System

- **Switch**
  - Use digit key 1 – 9 to switch among each agent, in this order:
    - Perception 1, Perception 2, …, Action 1, Action 2, … Hybrid 1, Hybrid 2,…
  - To fit the keyboard layout, the maximum agent number is 9
- **Planar Motion**
  - Fly along the planar trajectory composed by several goals. Goals are set by mouse click.
  - Goals must be set when the given agent’s battery and water tank are all not empty
  - When the current agent is changed, the previous one will still move along the trajectory
- **Vertical Motion**
  - Use up and down arrow to adjust the flight height of the agent
  - Flight height is persevered after switching

Movement System

- **Planar Motion**
  – Follow the trajectory composed by the goal series
  – Move with the step size that equals to the agent's velocity during the middle of the trajectory
  – When the distance between the agent's current position and the goal is less than one step size, its next position will directly overlap with the goal
- **Vertical Motion**
  – When pressing up or down key, the flight height changes by 5 meters
  – When the adjusted flight height exceeds the upper or lower bound, the flight height will not change
- **Battery Constraint**
  – Agents could not move when their battery capacity is 0. If so, the agent will directly stop the current task and return the agent base
  – The battery consumption is 0.1 during planar flight, and 0.05 during waiting

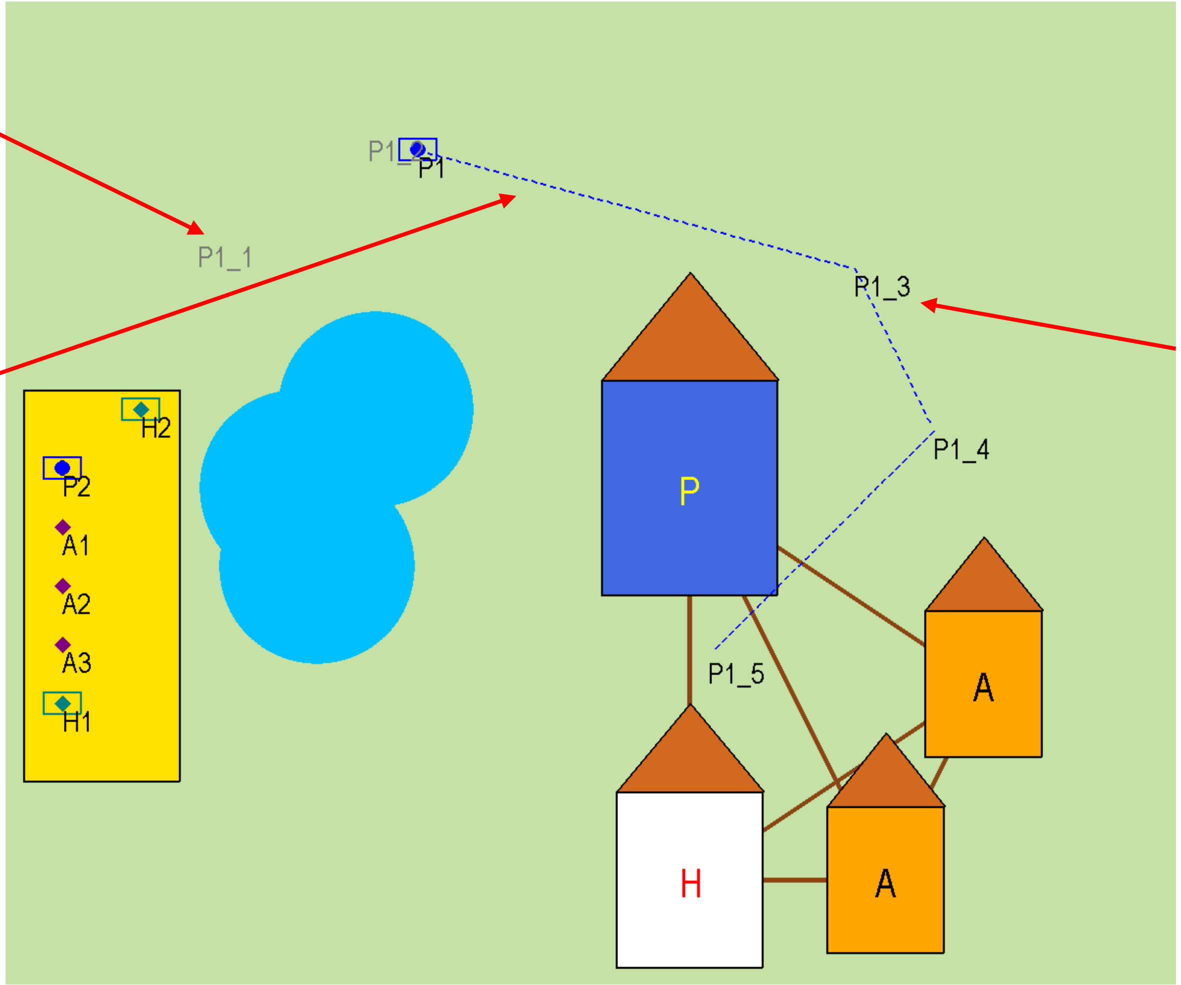
Goal and Trajectory

Passed Goal

- Marked with gray
- Only mark the goal position, ignore the passed trajectory

Trajectory

- Marked with dash line, using the same label color as the agent
- Present the trajectory that has not been passed yet



Pretending Goal

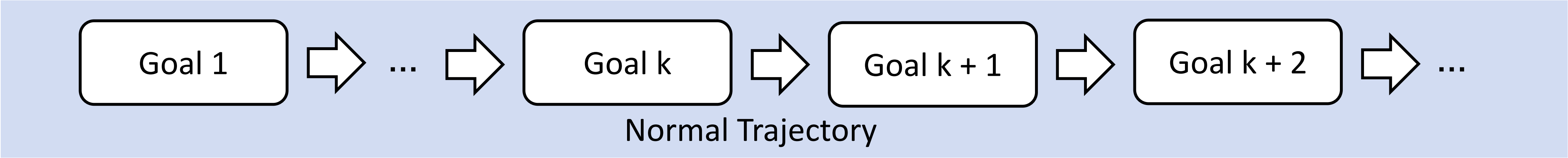
- Marked with black
- Positions are determined by mouse click
- If not being passed yet, connected with dash-line trajectory
- Goal Naming Policy:

Agent Name Goal Index

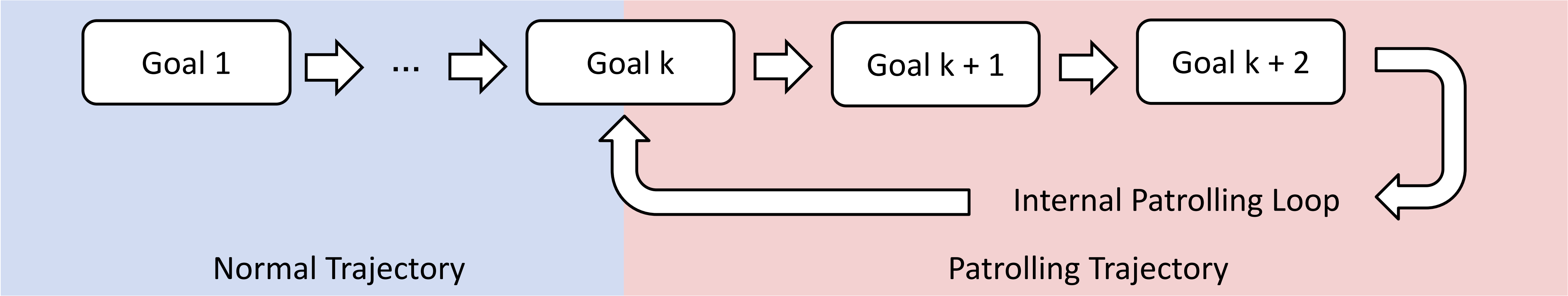
P1_6

Normal and Patrolling Trajectories

- **Normal Trajectory**
  - Agent's trajectory follows the sequential order in the goal list

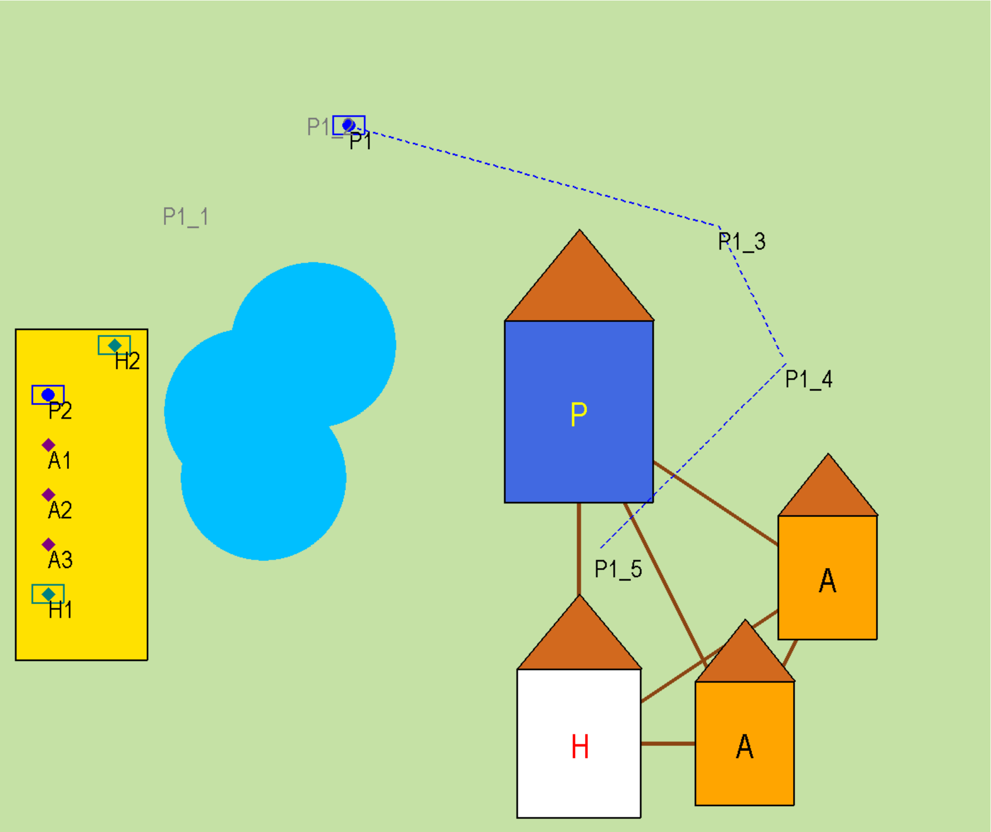


- **Patrolling Trajectory**
  - When the new goal locates close enough to one of existing goals in the goal list, goals in the list will form an internal loop trajectory called patrolling trajectory
  - When a new goal is added, the patrolling trajectory stops, and the previous patrolling goal list will be cleared. The agent follow the normal trajectory until a new patrolling loop is formed.

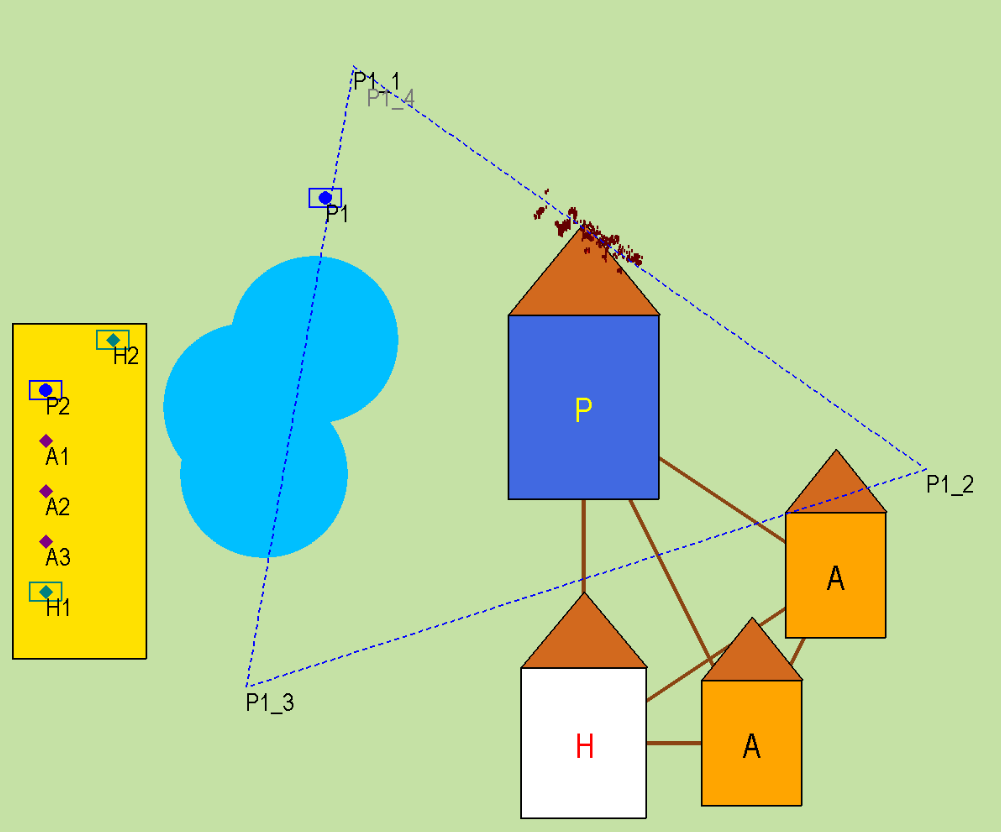


Normal and Patrolling Trajectory

- **Normal Trajectory**



- **Patrolling Trajectory**



FARSITE Wildfire Model

- **Wildfire Propagation Dynamics: FARSITE** [2]
  - Comprehensively consider the geographical, topographical and physical information about the environment, including the terrain, fuel and weather
  - Wildfire propagation dynamics using the FARSITE follows this form [3, 4, 6, 7]:

$$q_t^i = q_{t-1}^i + \dot{q}_{t-1}^i \delta t = q_{t-1}^i + \frac{\partial q_{t-1}^i}{\partial t} \delta t$$

    - q_t^i is the position for fire spot i at time t. In 2-D environment, $q_t^i = \left[x_t^i,\ y_t^i\right]$
    - $\dot{q}_{t-1}^i$ is the fire propagation velocity

- **Fire Propagation Velocity**
  - For the fire propagation velocity, its simplified representation form is [3, 4, 6, 7]:

$$\begin{cases} \dot{x}_t^i = C(R_t, U_t) sin\theta_t \\ \dot{y}_t^i = C(R_t, U_t) cos\theta_t \end{cases}$$

    - θ_t is the wind azimuth
    - $C(R_t, U_t)$ is the distance between initial fire position and the center of the ellipse. Its value is dependent on the fire growth coefficient (i.e., the fuel/vegetation coefficient R_t), wind speed (i.e., mid-flame wind velocity U_t)

FARSITE Wildfire Model

- **Fire Propagation Velocity**
  - The calculation process for C is [3, 4, 6, 7]

$$C(R_t, U_t) = \frac{R_t - \frac{R_t}{HB_t}}{2}$$

In which $HB_t = \frac{LB_t + (LB_t^2 - 1)^{0.5}}{LB_t - (LB_t^2 - 1)^{0.5}}$ and $LB_t = 0.936e^{0.2566U_t} + 0.461e^{-0.1548U_t} - 0.397$

- R_t is the fuel coefficient, which controls the fire spreading speed
- U_t is the amplitude of wind speed

- **Fire Intensity**
  - We leverage method proposed in [5, 6] to calculate the fire intensity.

$$I_t^q = 259.833\left(\frac{h_t^q}{\cos(\alpha_t^q)}\right)$$

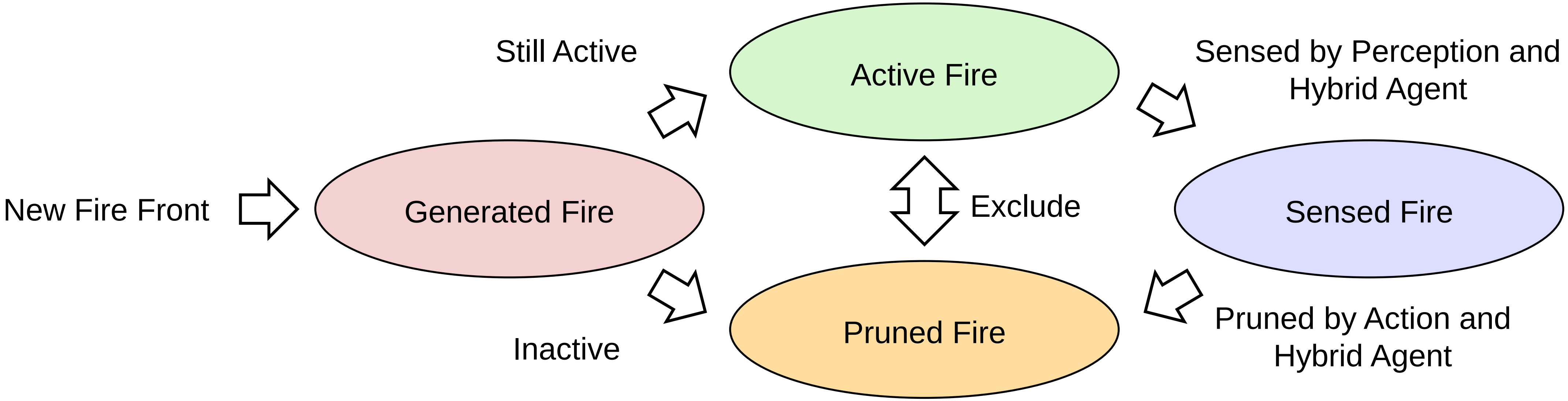
- Here, h_t^q and α_t^q are the flame height and flame tilt-angle for the fire in grid q at time t
  - We model the dynamic fire decay update over time as follows in which λ is the decay rate [6]:

$$I_{t+\delta t}^q = I_t^q . e^{-\lambda \frac{\delta t_q}{R_t}}$$

Reconnaissance Wildfire

- **Reconnaissance Wildfire**
  - In the reconnaissance wildfire mode, wildfire spots must be sensed before they could be pruned. In this case, wildfire spots that have been generated could be divided into the following genres:
    - **Sensed Fire:** Wildfire spots that have been sensed by the perception and hybrid agents
    - **Pruned Fire:** Wildfire spots that have been pruned by the action and hybrid agents

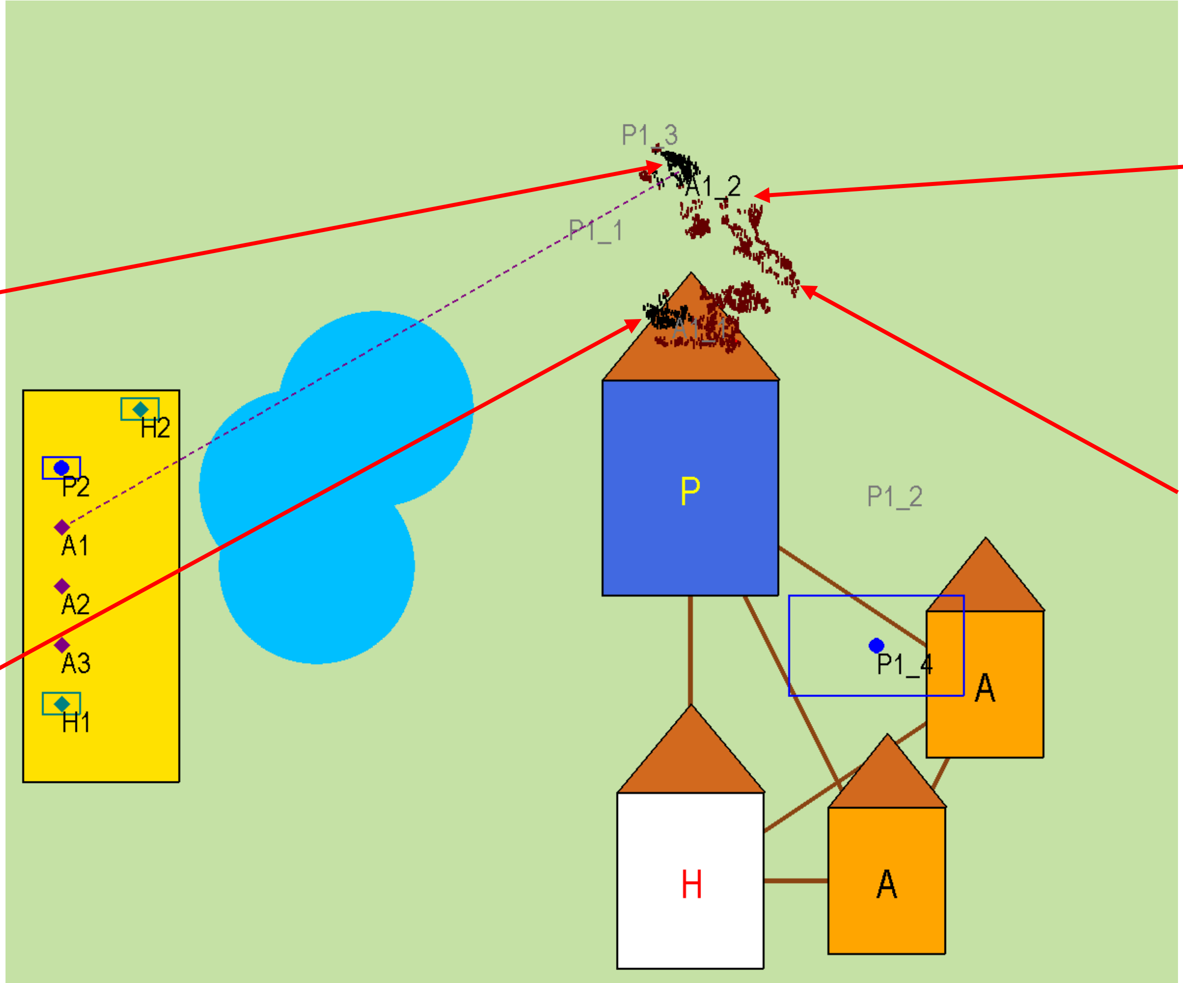
      Note: All the pruned fire spots are considered as sensed fire spots
    - **Active Fire:** Wildfire spots have not been pruned, including fire spots have or have not been sensed



Reconnaissance Wildfire Display

Pruned Fire
- 1-radius black points
- Composed by all fire spots that once locate with the sensing agent's scope
- New fire fronts could not be propagated into the pruned region
- **Note:** Fire spots must be sensed before they could be put out

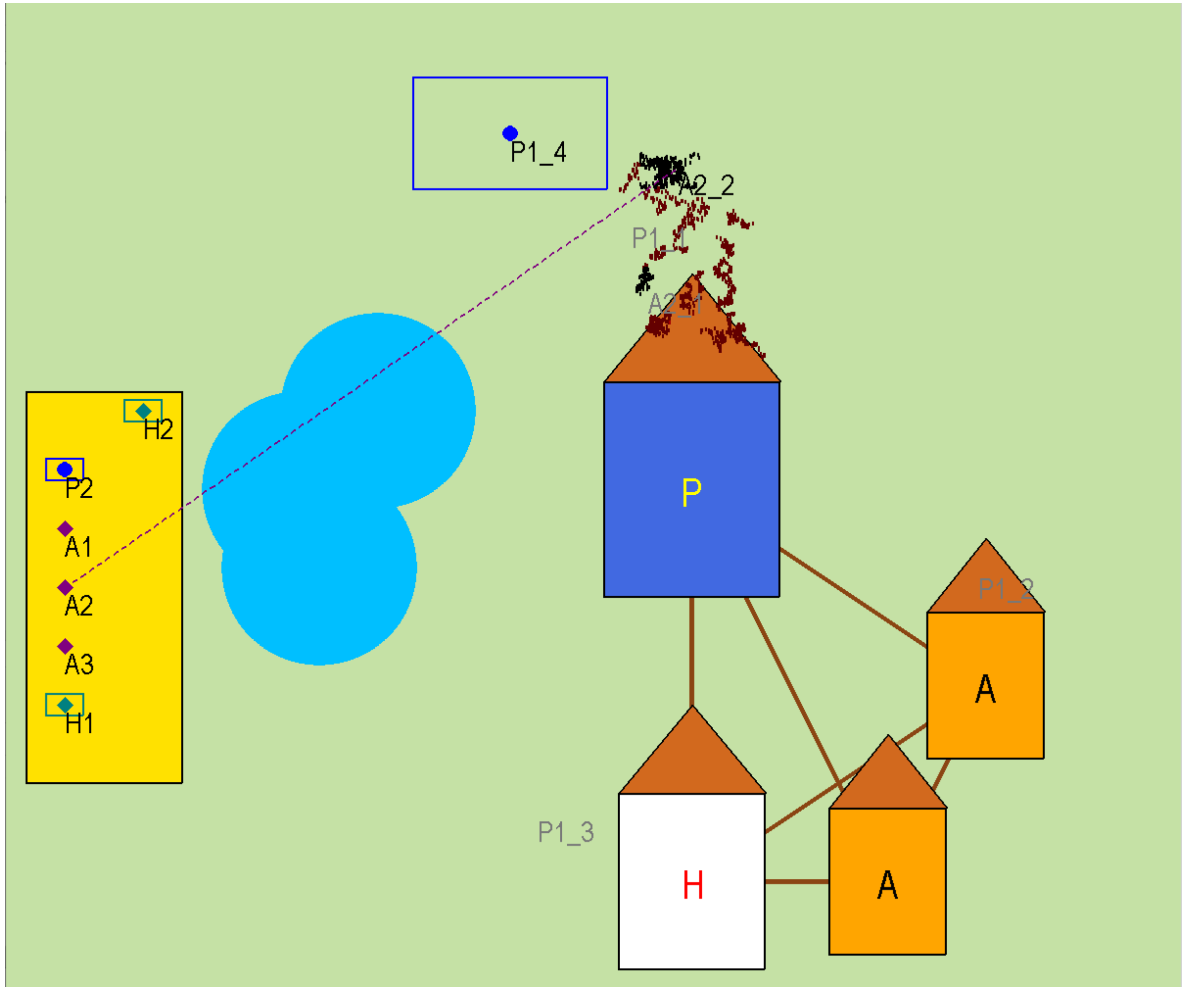
New Fire Fronts
- 3-radius black points
- New location determined by the FARSITE algorithm
- Updated in each iteration, could overlap with the previous fire spots



Active Fire Fronts (Not shown)
- 1-radius red points
- Composed by all new fire fronts generated before that have not been pruned

Sensed Fire
- 1-radius red or brown points
- RGB value is set based on the intensity, the range 100 - 255
- Composed by all the new fire spots that have been detected before

Information Display



Battery Capacity Info:

Agent	Remaining Energy	Estimated Energy Till Next Goal	Current Height
Perception 1:	249	249	100
Perception 2:	500	N/A	20
Action 1:	500	N/A	N/A
Action 2:	500	425	N/A
Action 3:	500	N/A	N/A
Hybrid 1:	500	N/A	20
Hybrid 2:	500	N/A	20

Water Tank Info:

Agent	Remaining Pruning Times
Action 1:	10
Action 2:	10
Action 3:	10
Hybrid 1:	10
Hybrid 2:	10

Score:	-23577.75
Overall Firefighting Score:	13.24
Perception Score:	43.29
Action Score:	30.59
Safe Facilities:	4

Battery Capacity Info

- Remaining Energy: Current remaining energy in the agent's battery
- Estimated Energy Till Next Goal: The estimated energy left when the agent arrives the next goal (Could be Negative)
- Current Flight Height (Sensing and Hybrid Agent)

Water Tank Info

- Computed by the remaining pruning times (Firefighter and Hybrid Agent)

Online Score Display

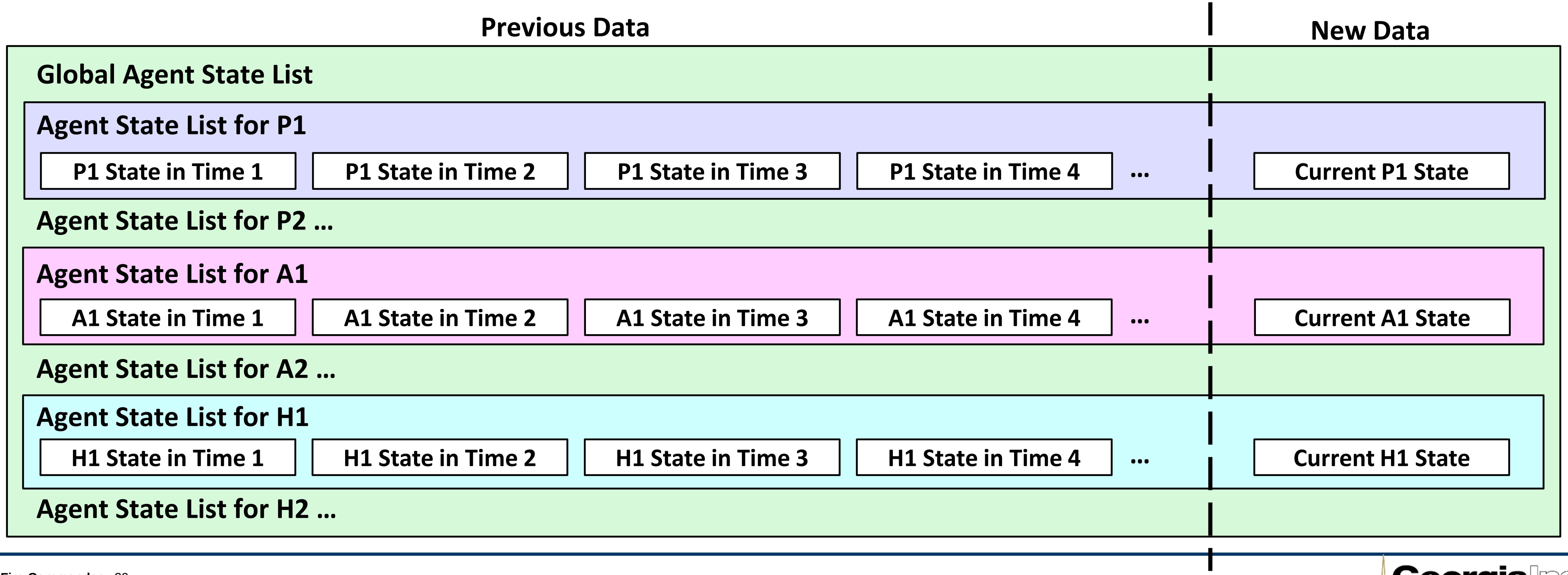
- **Negative:** Total Negative Score
- **Positive:** Overall Firefighting Score, Perception and Action Score, The Number of Safe Facilities

Data Storage

- **Motivation**
  - Transfer information between different part of the program
  - Track the scenario state at each moment
  - Recreate the scenario for animation reconstruction and LfD policy generation
- **Stored Data Type**
  - **Agent Info**
    - Current Agent State (Perception, Action and Hybrid Agent)
  - **User Data**
    - Keyboard Action and Goal Info
  - **Target Info**
    - Target Loci (House, Hospital, Power Station)
    - Agent Base Loci, Lake Loci
  - **Fire Info**
    - Fire Coordinates Info (Active, Sensed, Pruned)

Agent State List

- **Global Agent State List Structure**
  - Record each agent's state at each moment during the simulation
  - **Hierarchy Structure**: Global Agent State (A List for All Records) -> Agent State List for P1 (All Records for Agent P1) -> P1 State in Time 1 (Record for Agent P1 at Time 1)



Agent State List

- **Current Agent State List (Record for Agent X at Time t)**
  - 16-element list. Describe the state for agent X at the given moment t
- **Sample Structure (Record for P1 at Time 0)**

Current Position (X, Y, Z)

Current Time:
Synchronized in each iteration

Identity Recognition:
- Agent Type: 0: Perception; 1: Action; 2: Hybrid
- Agent Index in Each Genre: 1, 2, …

```
current_Agent_State_List = [120, 400, 30, 0, 0, 0, 0, 0, 1, 1, 0, 0, 500, 1, 0, 0]
```

Current Velocity (X, Y, Z):
Determine the agent position in the next step

Current Goal Index

Battery and Water Tank Info:
- Current Running Distance
- Current Waiting Time
- Water Tank Capacity

Control Flag
- Movement Flag
  Whether to finish trajectory or return to agent base
- Patrolling Flag
  Whether to follow the normal movement mode or patrolling mode
- Patrolling Goal Index

User Data List

- **User Data List**
  - Monitor and record the keyboard action
  - Store the goal list for trajectory generation, including the normal and patrolling trajectory

Previous Data | **New Data**

User Data List

User Data List for Agent 1

Time 2: New Goal	Time 5: New Goal	Time 7: New Goal	Time 12: New Goal	…	Current New Goal

User Data List for Agent 2…

- **New Goal Buffer (Time t: New Goal)**
  - New goal information for agent X acquired at time t

```
new_Goal_Buffer = [350, 400, 500, 0, 5]
```

Goal Index (Only for Planar Goal): → 5

Goal Position (X, Y): → 350, 400
- **Planar Goal:** Mouse Click Position
- **Lift / Down:** Current Agent Position

Current Time → 500

Keyboard Action Type: → 0
- **0:** Planar Goal
- **1:** Lift / Down

Target Loci List

- **Target Loci and Lake List (Using the Same Template)**
  - Record the status of all targets or lakes at each moment
- **Sample Target Loci List**
  - For lake list, substitute all target loci list units with lake loci list units)

Previous Data | **New Data**

Global Target Loci List

Target Loci List 1

Target Loci in Time 1 | **Target Loci in Time 2** | **Target Loci in Time 3** | … | **New Target Loci**

Target Loci List 2 …

- **Target Loci List Unit**
  - Loci information for target X at time t

Target Center Position (X, Y) **Edge Enable Flag** **Current Time**

```
Target_Loci = [700, 800, 150, 180, 1, 1, 0]
```

Target Size (X, Y)

Target Type:
- 0: House
- 1: Hospital
- 2: Power Station

- **Lake Loci List Unit**
  - Loci information for lake X at time t

Lake Center Position (X, Y)

```
Lake_Loci = [300, 500, 100, 0]
```

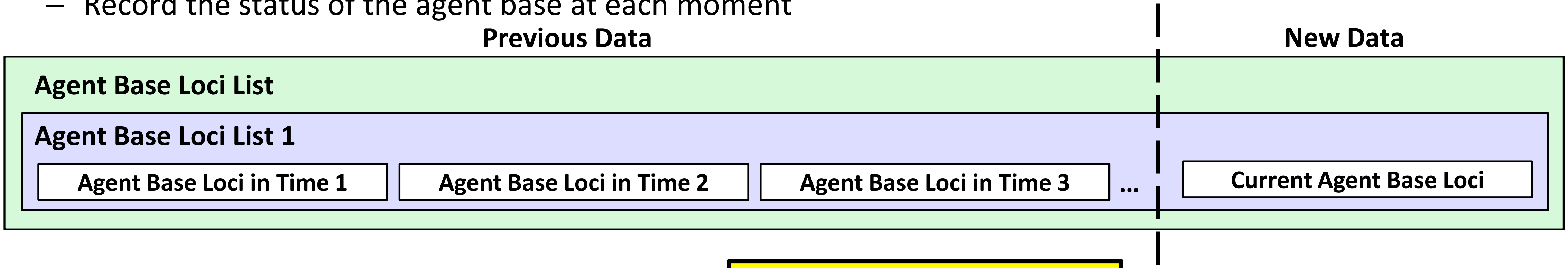
Current Time

Radius

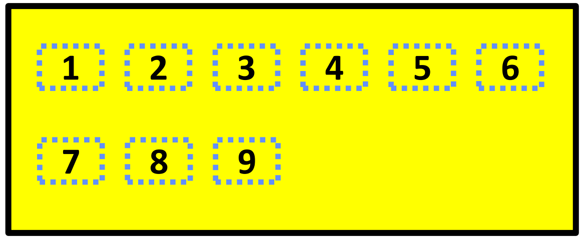
Esmaeil Seraj, Xiyang Wu, Matthew Gombolay
10/31/2020

Target Loci List

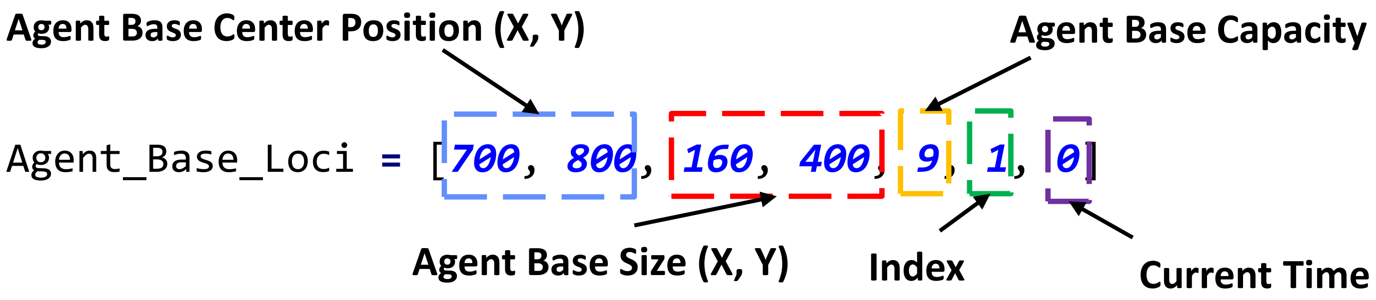
- **Agent Base Loci List**
  - Record the status of the agent base at each moment



- **Agent Base Arrangement**
  - Constant list, record the agent's relative position in the agent base



- **Agent Base Loci List Unit**
  - Record the loci and capacity information for the agent base at time t



Esmaeil Seraj, Xiyang Wu, Matthew Gombolay

Wildfire List

- **Fire State List**
  - Store all the fire fronts that have been generated

Previous Data | **New Data**

Global Fire State List

Fire State List for Fire Region 1

Time 1	Time 2	Time 3	Time 4	…	Current Time
New Fire Front 1	New Fire Front 1	New Fire Front 1	New Fire Front 1		New Fire Front 1
New Fire Front 2	New Fire Front 2	New Fire Front 2	New Fire Front 2		New Fire Front 2
…	…	…	…		…

Fire State List for Fire Region 2…

- **New Fire Front**
  - Record the position, intensity and generated time for new fire fronts generated at time t

Fire Front Position (X, Y)

```
New_Fire_Front = [700, 800, 10, 5]
```

Fire Intensity

Current Time

CORE ROBOTICS LAB

Wildfire List

- **Sensed and Pruned Fire List (Using the Same Template)**
  - Store all the fire fronts that have been sensed or pruned in the given interval. If no fire fronts are sensed or pruned, return null list

Previous Data | **New Data**

Global Sensed Fire List

Sensed Fire List for Agent 1

(Null)	**(Null)**	**Time 3** **Sensed Fire Info 1** **Sensed Fire Info 2** **…**	**(Null)**	**…**	**Current Time** **Sensed Fire Info 1** **Sensed Fire Info 2** **…**

Sensed Fire List for Agent 2…

- **Sensed Fire Info**
  - Record the position, intensity and velocity for sensed fire spots

```
Sensed_Fire_Front = [700, 800, 10, 5]
```

Fire Front Position (X, Y)

Fire Intensity

Fire Velocity, To compute the CoM of the sensed region

- **Pruned Fire Info**
  - Record the position, intensity for pruned fire spots

Fire Front Position (X, Y)

Fire Intensity

```
Pruned_Fire_Front = [700, 800, 10]
```


Georgia Institute of Technology

Target On Fire List

- **Target On Fire List**
  - Store the number of fire fronts that locate inside each target at each moment

Target On Fire List

Propagation			Pruning		New Data
The Number of Fire Fronts in House 1					
Time 1: 0	Time 2: 2	Time 3: 4	Time 4: 0	…	Current: 5
The Number of Fire Fronts in House 2 …					
The Number of Fire Fronts in Hospital 1					
Time 1: 0	Time 2: 3	Time 3: 5	Time 4: 5	…	Current: 8
The Number of Fire Fronts in Hospital 2 …					
The Number of Fire Fronts in Power Station 1					
Time 1: 0	Time 2: 0	Time 3: 2	Time 4: 5	…	Current: 0
The Number of Fire Fronts in Power Station 2…					

Score Policy

- **Negative Reward**
  - **Total Negative Reward**

$$0.1 \times \text{Number of active firespots} + \text{Penalty Coef} \times \text{Firespots Number in Targets}$$

    - **Penalty Coefficients: 0.1** Per Fire Spot, **1** Per House, **2** Per Hospital, **5** Per Power Station, **5** Per Agent Base
    - **Active Fire Spots:** All fire spots in each single coordinates that have not been pruned (Not Sensed + Sensed, Excludes the pruned fire spots)
    - **Fire Spots in Targets:** The active fire spot number in each targets in the given interval
  - **Expected Negative Reward**

$$0.1 \times (\text{Number of active firespots} + \text{Number of pruned firespots}) + \text{Penalty Coef} \times \text{Firespots Number in Targets}$$

    - **Penalty Coefficients:** The same
    - **Fire Spots:** All fire spots in each single coordinates that have ever been generated
  - **Negative Reward Ratio**

$$\frac{\text{Total Negative Reward}}{\text{Expected Negative Reward}} \times 100\%$$

Score Policy

- **Positive Reward**
  - **Perception Score**
    - The ratio of sensed fire spots in all the fire spots generated

$$\frac{\text{Number of discoverd firespots}}{\text{Total number of firespots}} \times 100\%$$

  - **Action Score**
    - The ratio of pruned fire spots in all the sensed fire spots

$$\frac{\text{Firespots that have been put out}}{\text{Number of discovered firespots}} \times 100\%$$

  - **Safe Facility Score:**
    - The ratio of safe facilities (Not on fire during the whole simulation) with the number all the facilities

$$\frac{\text{Number of facilities that have been saved}}{\text{Total number of facilities}} \times 100\%$$

Score Policy

- **Final Score**

$$\text{Perception Score} + \text{Action Score} + \text{Safe Facility Score} - 3\times\ \text{Negative Reward Ratio}$$

- **General Evaluation**

Grade	Final Score
Failed	< 50
Fair	50 - 60
Almost There!	60 - 80
Well Done	80 - 90
Excellent	>90

Animation Reconstruction

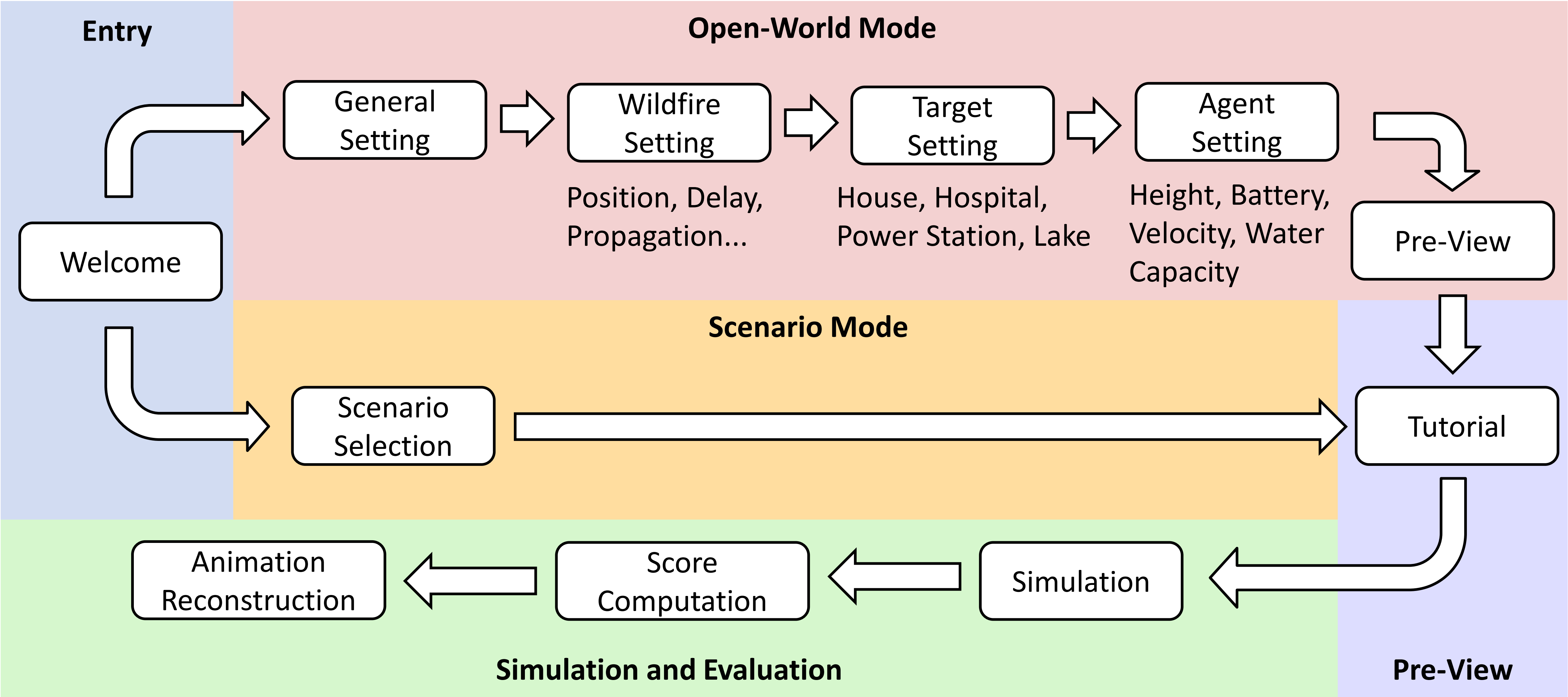
- **Motivation**
  - LfD uses the screenshot the generate the control policy, but online image I/O in Python is time-consuming.
    - **Maximum Simulation Refreshing Frequency:** Iteration with Online Image I/O: **10 Hz**; Pure Iteration: **100 Hz**
- **Solution**
  - Use the .pkl file saved to re-construct the scene during the simulation
- **Procedure**

Load Data	Reconstruct Scenario	Take Screenshots	Create Animation
Load the data from the .pkl file, which is the status of each object at each moment	Reconstruct simulation with for loop: • **Static:** Houses, Hospitals, Power Stations, Lakes… • **Dynamic:** Agents, Wildfire Propagation • **Ignoring:** Agent Trajectory and Goal Marks	Use the screenshot function to save the status of the simulation at each moment	Create the animation with the images saved when the simulation finished

Graphical User Interface

- **Motivation**
  - Complicated and confusing parameters in the environment setting
    - **Concise:** Help the user to design the environment step by step
    - **Interactive:** Enable the modification, Visualize the completed environment
  - Requirement for the various scenario with different settings
    - **Comprehensive**: Cover all the significant parameters in the environment
    - **Convenient:** Offer several pre-determined scenarios

GUI Frame


Entry
Open-World Mode
General Setting
Wildfire Setting
Target Setting
Agent Setting
Position, Delay, Propagation...
House, Hospital, Power Station, Lake
Height, Battery, Velocity, Water Capacity
Pre-View
Welcome
Scenario Mode
Scenario Selection
Tutorial
Animation Reconstruction
Score Computation
Simulation
Simulation and Evaluation
Pre-View

Welcome Page

Open-World Mode
Design the environment step by step

Tutorial
Learn about the regulation of the environment
Note: Could not enter the simulation from the tutorial page that enters from the welcome page

Scenario Mode
Use the pre-determined scenarios for simulation

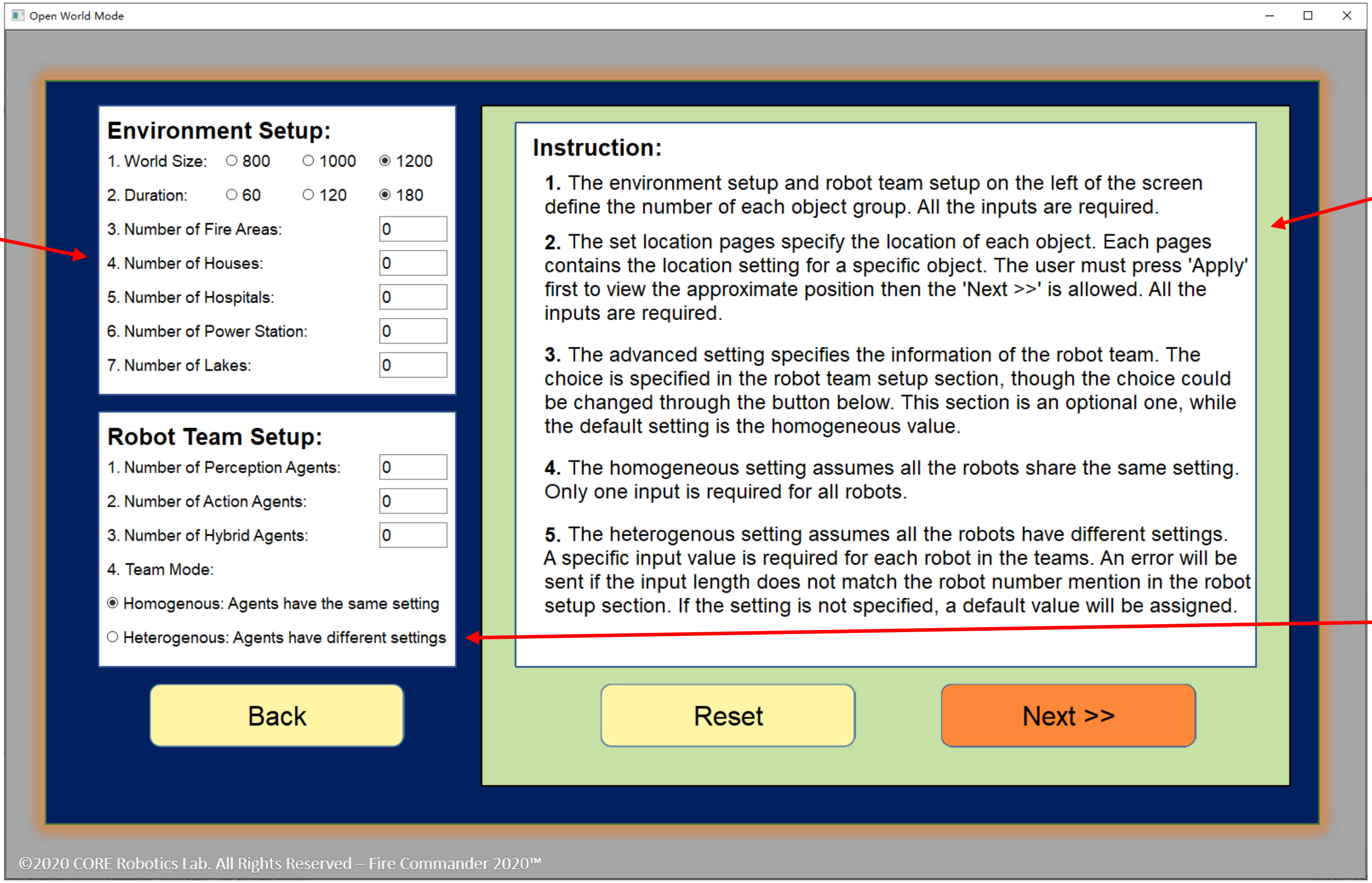
Open-World Mode

- **Motivation**
  - **Standardization**
    - **Data input and storage:** Standardized data input and storage method, help users to recreate the scenarios with these parameters, check and modify the finished scenarios
    - **Grid Map:** Divide the simulation world into grid map (E.g. A 1200 × 1200 World to 12 × 12 Grid), greatly decrease the number of potential states for the elements in the given environment
  - **User-friendly**
    - **Interface**: Separate the whole complicated design procedure into several simple stages:
      - **Environment Setup:** Number of targets
      - **Robot Team Setup:** Number and control mode of each kind of agents
      - **Target Setting:** Details of the wildfire and each kind of targets in the scenario
      - **Agent Setting:** Details of each kind of agents in the scenario
    - **Input Mode:** Incorporate coordinate input and multiple choices only
    - **Visualization:** Visualize the simultaneous scenario during (Grid map) and after the design (Environment pre-view page)

General Setting Page

Environment Setting

- **World Size**
  800 / 1000 / 1200
- **Duration**
  60 / 120 / 180
  The user can either exit the program by pressing the exit button or wait until the due time
- **Target number**
  Categorical, Maximum target number is 5



Brief Instruction on the GUI

Robot Setting

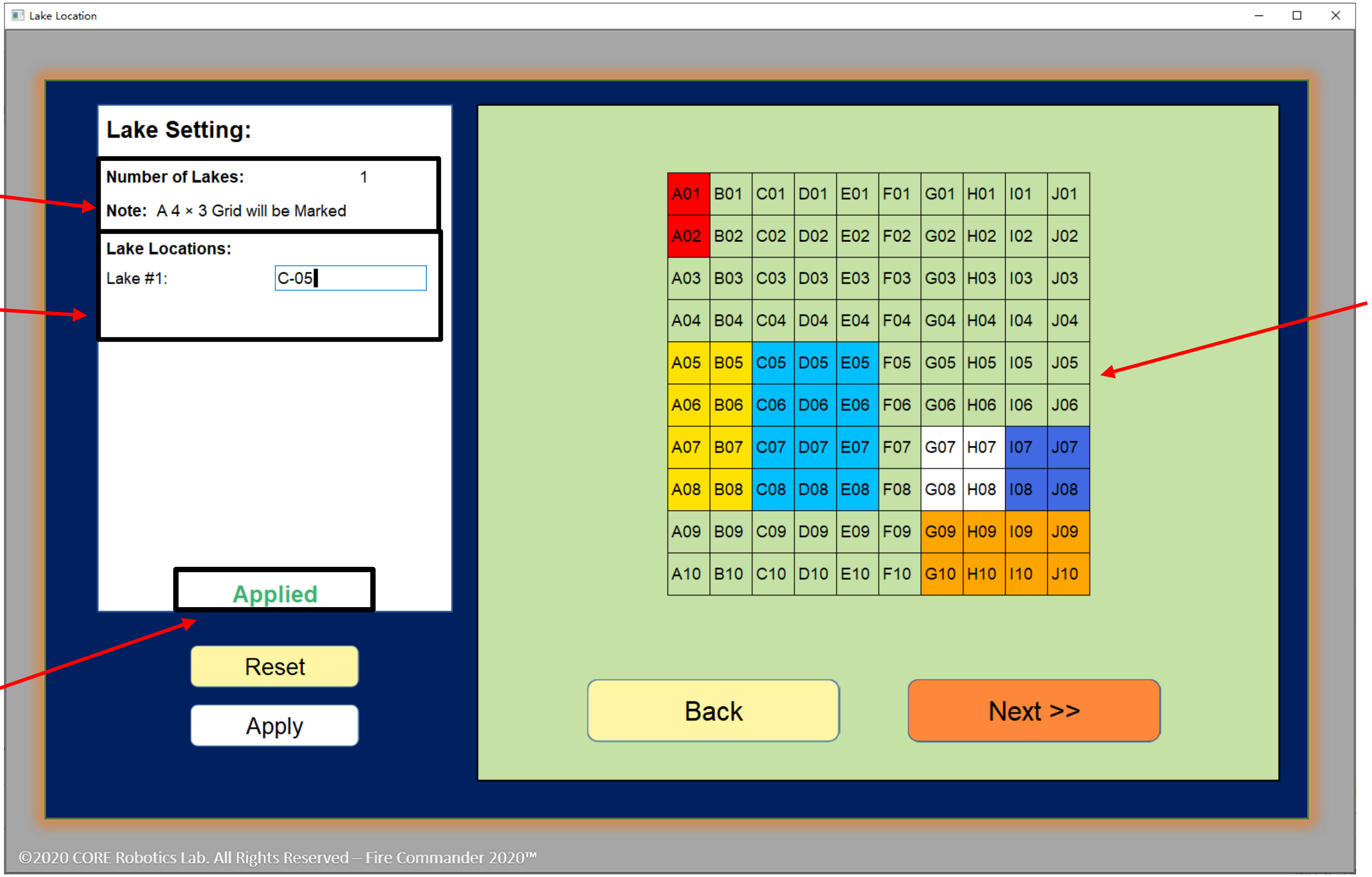
- **Agent number**
  Categorical, Maximum agent number is 9 in total
- **Team mode**
  Whether the agents have the same setting (Flight Height, Battery, Velocity, Water Tank Capacity)

Target Setting Page

Info Display
- Target Number
- Symbol Size on the Grid Map

Target Location
- In coordinates, Set One-by-one

Applied Flag
- Initial Status or After Any Target Location Changed:
  Not Applied
- Press Apply Button and All the Inputs are Valid:
  Applied



Grid Map
Use the grid and following symbols to represent the scenario generated

- Grassland
- Fire Region 1 × 1 Grid
- Agent Base 2 × 4 (Horizontal) / 4 × 2 (Vertical) Grid
- House 2 × 2 Grid
- Hospital 2 × 2 Grid
- Power Station 2 × 2 Grid
- Lake 4 × 3 Grid

Wildfire Setting Page

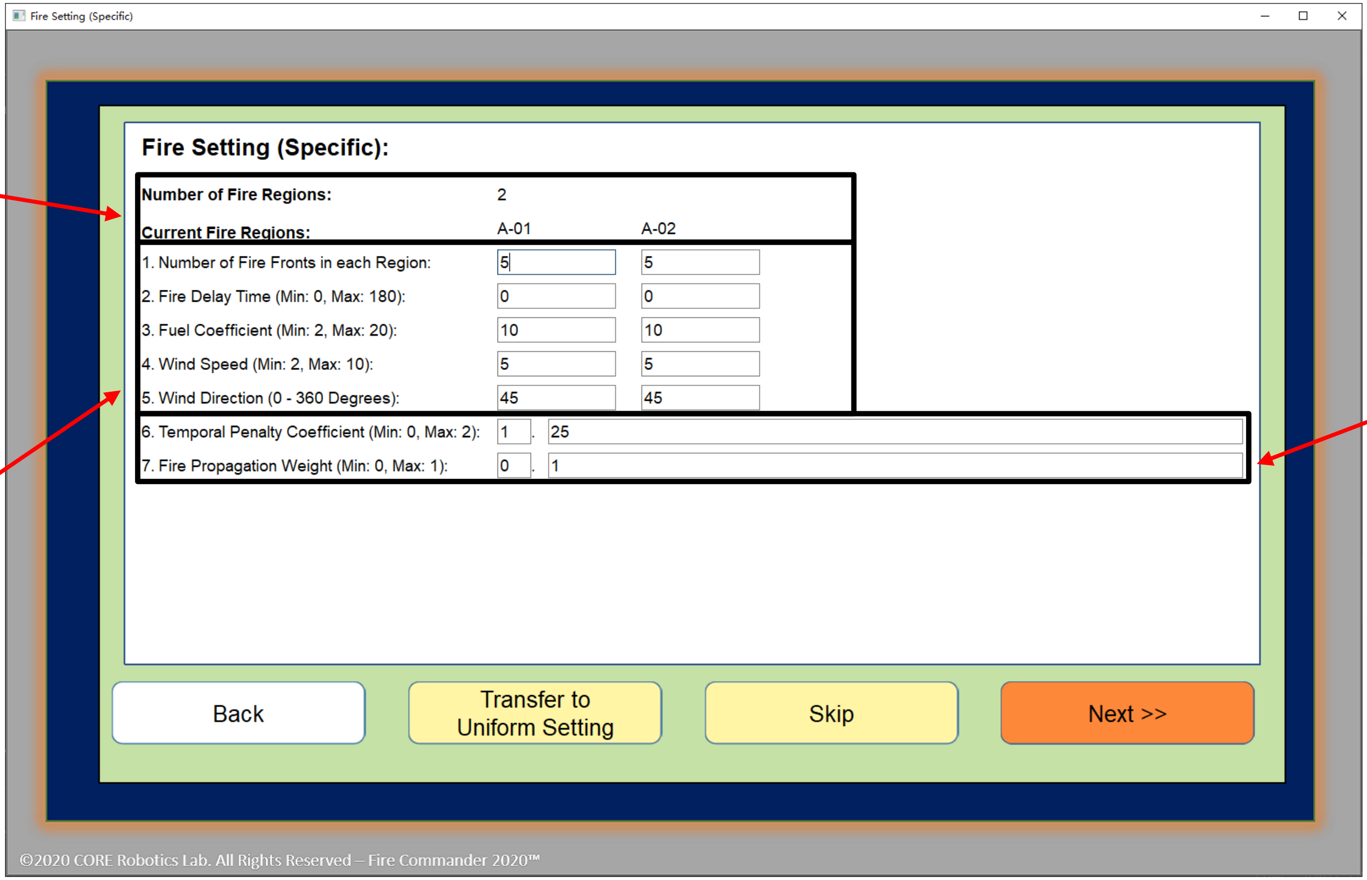
- **Location Setting:** The same as the target setting
- **Parameter Setting:** Uniform (Homogenous Setting) / Specific (Region-wise Setting)

Info Display

- Fire Region Number
- Coordinates on the Grid Map

Fire Setting

- Fire Front Number in Each Region
- Delay Time
- Propagation: Fuel Coefficient, Wind Speed and Direction



Score Parameter

- Parameters used to compute the negative reward introduced by the wildfire propagation

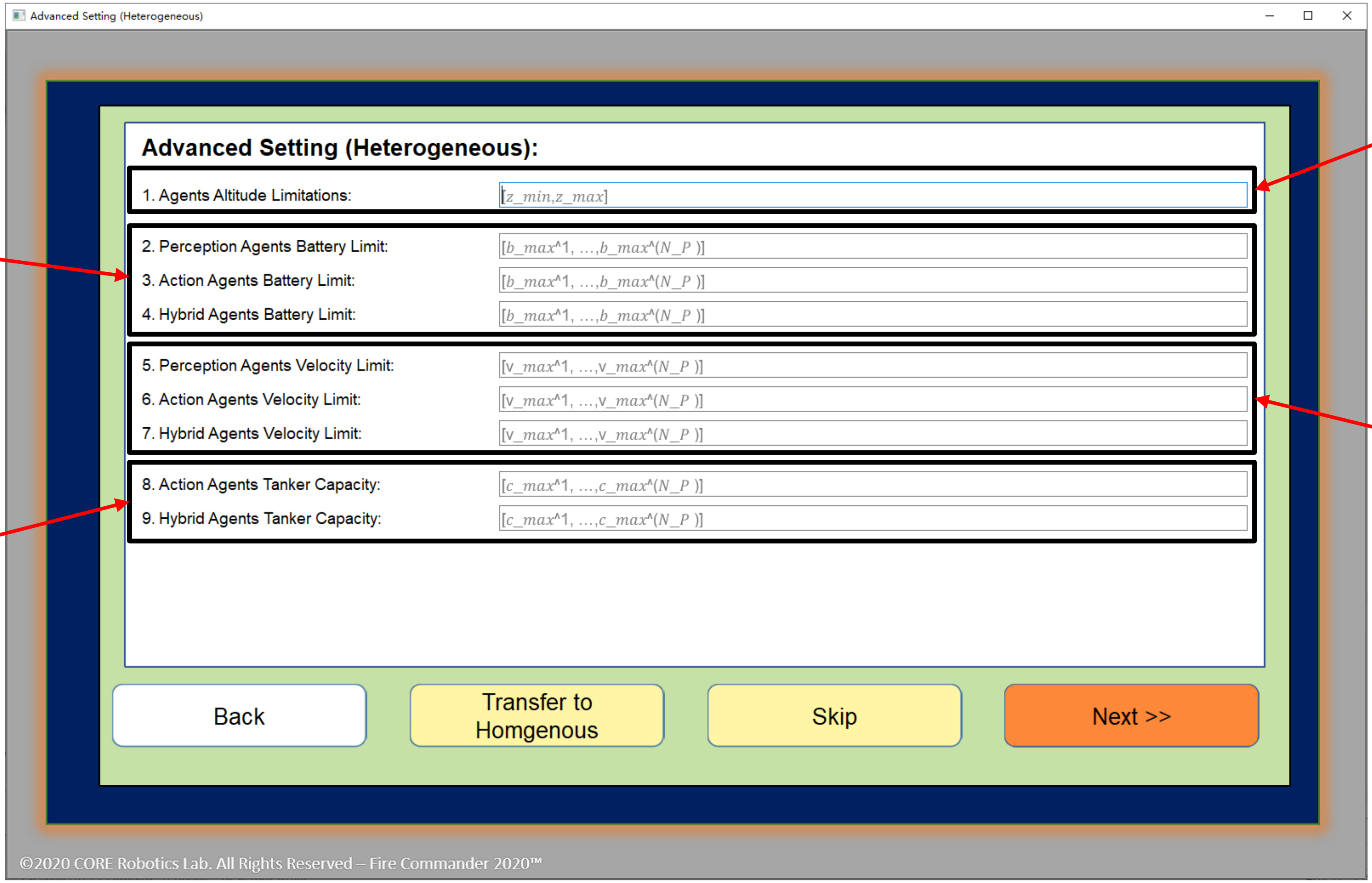
Agent Setting Page

Battery Limit

- Determine the battery capacity
- Consumption During Waiting: 0.05 / Iteration
- Consumption During Flight: 0.1 / Iteration

Water Capacity

- Only for the action and hybrid agent
- Determine the time number that the agent could put out the fire in the given region



Agent Flight Height

- From Minimum Height to Maximum Height

Velocity

- Determine the maximum velocity for each agent
- Generally, the step size of the agent equals to the velocity. However, when the distance between the goal and the agent is small enough, the agent will directly move to the goal

Agent Setting Page

- **Standard Input List Format**
  - E.g. There are 1 perception and 2 hybrid agents in the environment
  - **Homogenous**
    - Height Limitations: [10, 100]
    - Perception Battery Limit: [500]
    - Hybrid Battery Limit: [500]
  - **Heterogenous**
    - Height Limitations: [(10, 100), (10, 100), (10, 100)]

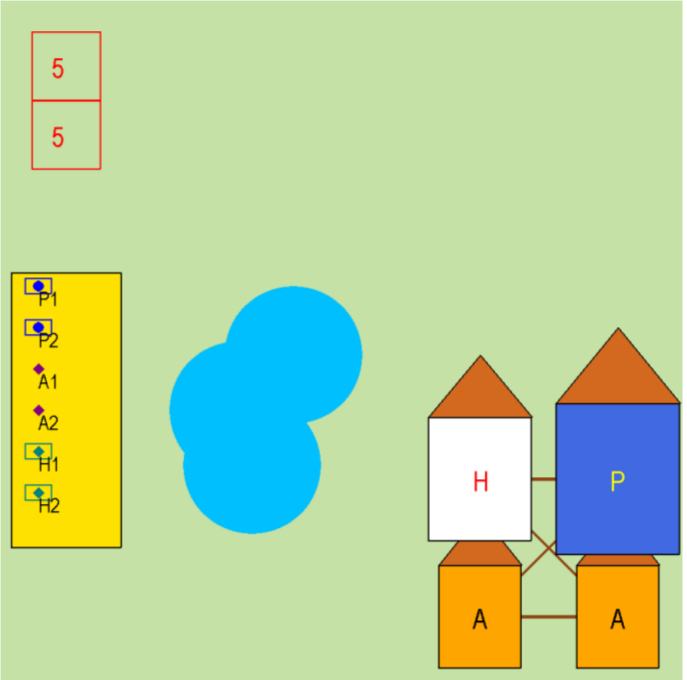
      Follow this order: 1st Perception, 2nd Perception… 1st Action, 2nd Action ... 1st Hybrid, 2nd Hybrid …
    - Perception Battery Limit: [500]
    - Hybrid Battery Limit: [500, 500]
- **Note:** If the user leave several inputs empty and wants to generate the scenario, the program will automatically fill the list with the default value

Pre-View Page

- **Pre-View**
  - Display the static objects in the scenario, including all kinds of targets, agent base and lakes
  - For agents, mark their initial positions on the agent base
  - For the fluctuate elements like the wildfire region, mark the scope of their initial position, with the number of new fire front generated at each moment

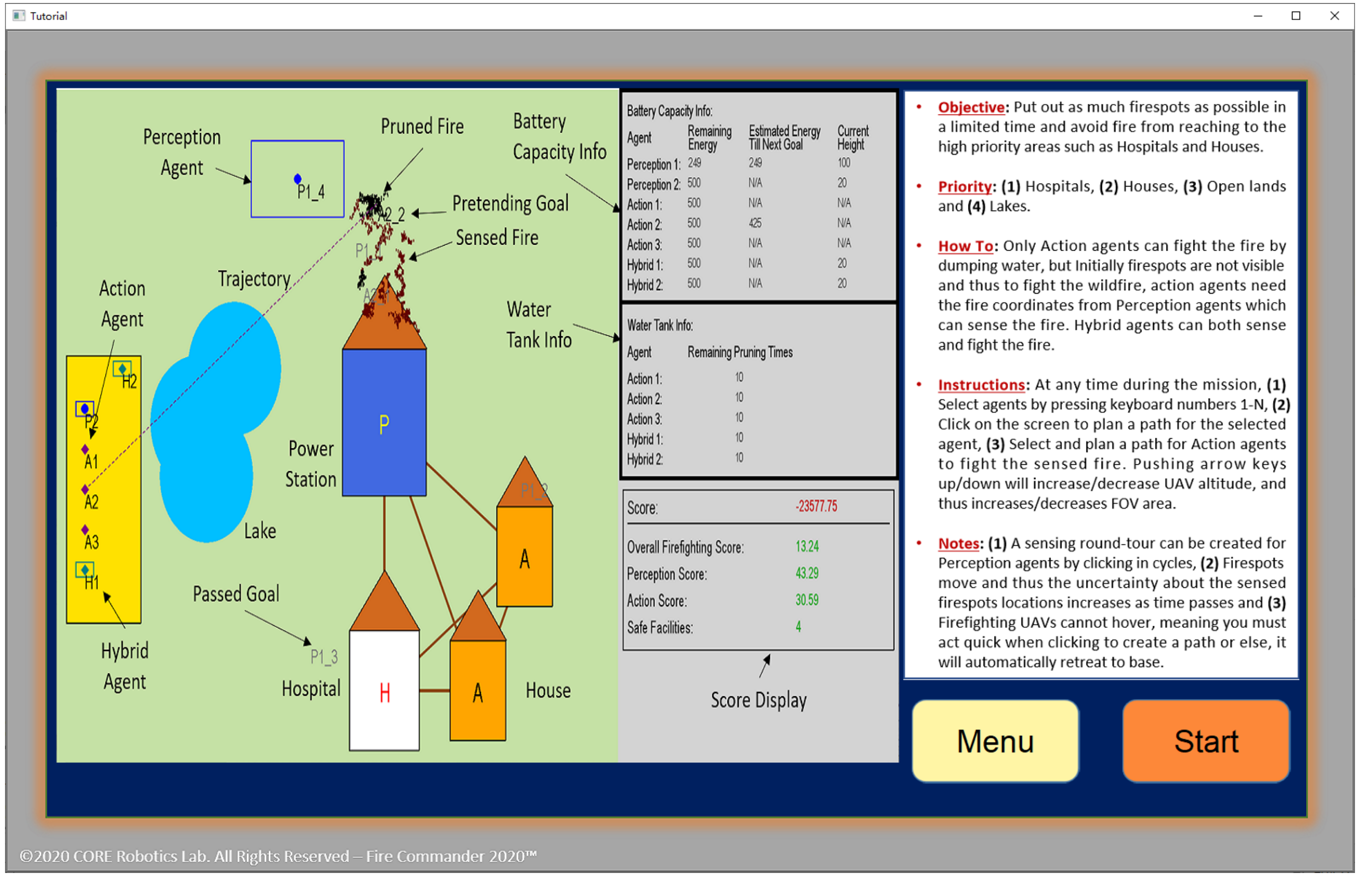
- **Grid Map**

- **Pre-View**



Tutorial Page

- **Tutorial**
  - Instruct the control policy of the simulation environment
  - Offer the returning to the menu function to enable the re-design option before simulation begins



Score Display Page

General Evaluation

Positive Reward

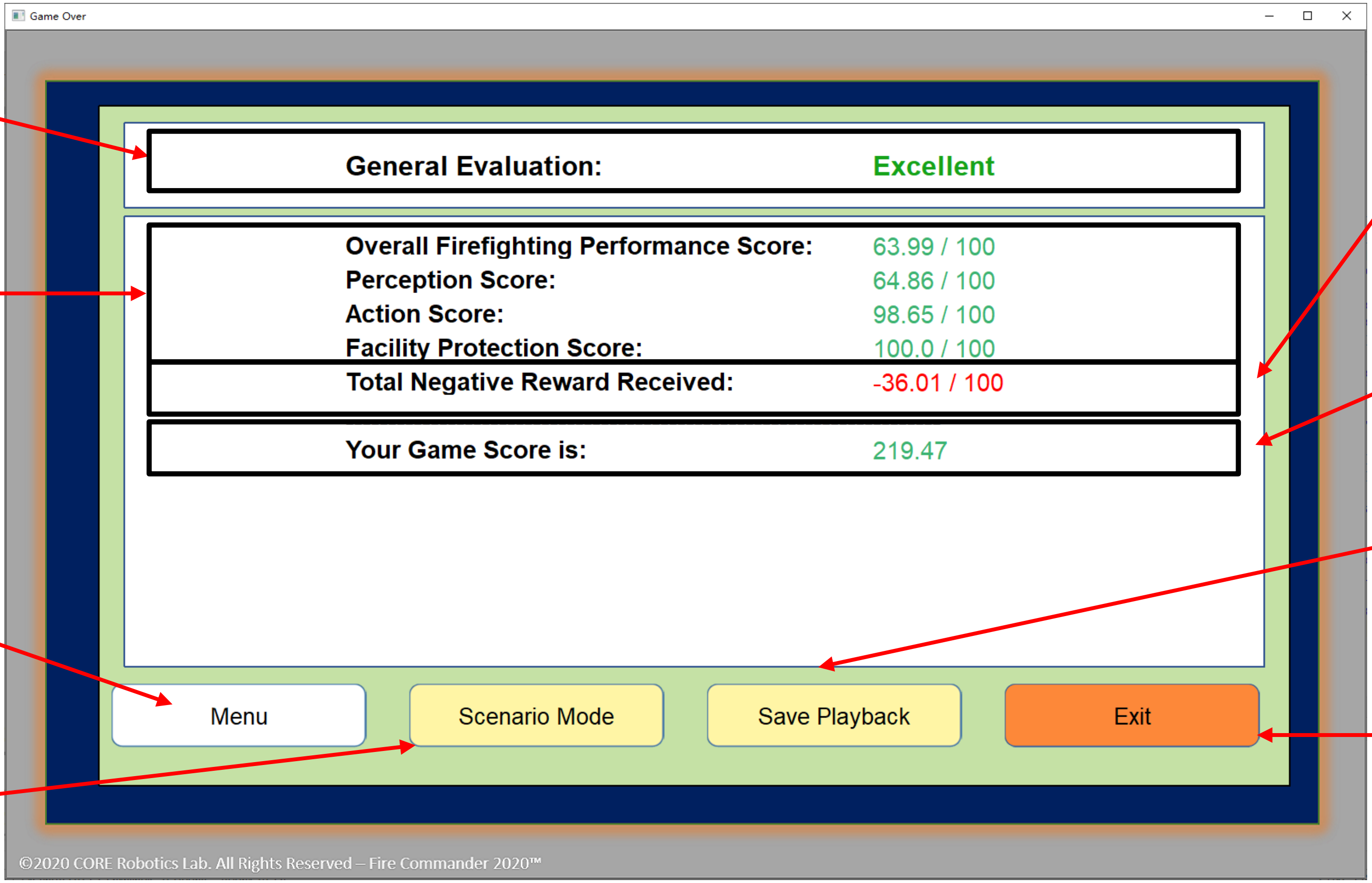
- Overall Firefighting Performance
- Perception Score
- Action Score
- Facility Protection Score

Menu

Back to welcome page, restart the simulation

Scenario Mode

Back to scenario mode page



Negative Reward

- Total Negative Reward Ratio: Final Online Total Negative Reward / Expected Negative Reward

Final Score

Save Playback

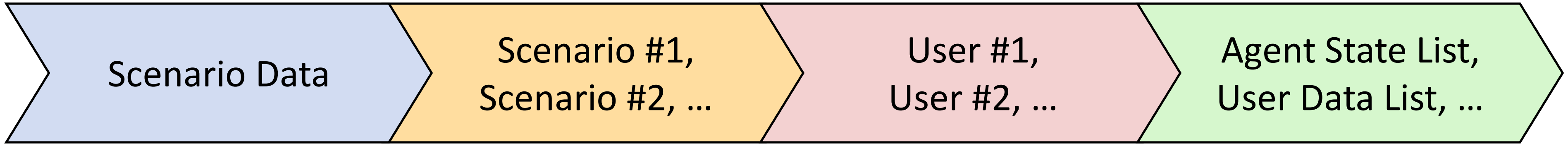
Animation Reconstruction

Exit

Directly exit the GUI program without reconstructing simulation

Scenario Mode

- **Motivation**
  - Collect and store user data under a finite number of scenarios in a standardized way
    - The structure of dictionary to store the user data:



  - Collect the user data under different scenarios for better control policy generation
    - Based on the setting of agents, targets and wildfire regions, separate all the scenarios into easy, moderate and hard scenarios, collect the user data in a gradual way
    - Each scenario has some parameters common with others to avoid the extreme result, as well as some unique ones to test the user’s reaction to some certain conditions
    - Add the practice scenario to help users familiarize the environment, avoid the occurrence of invalid data

Scenario Page

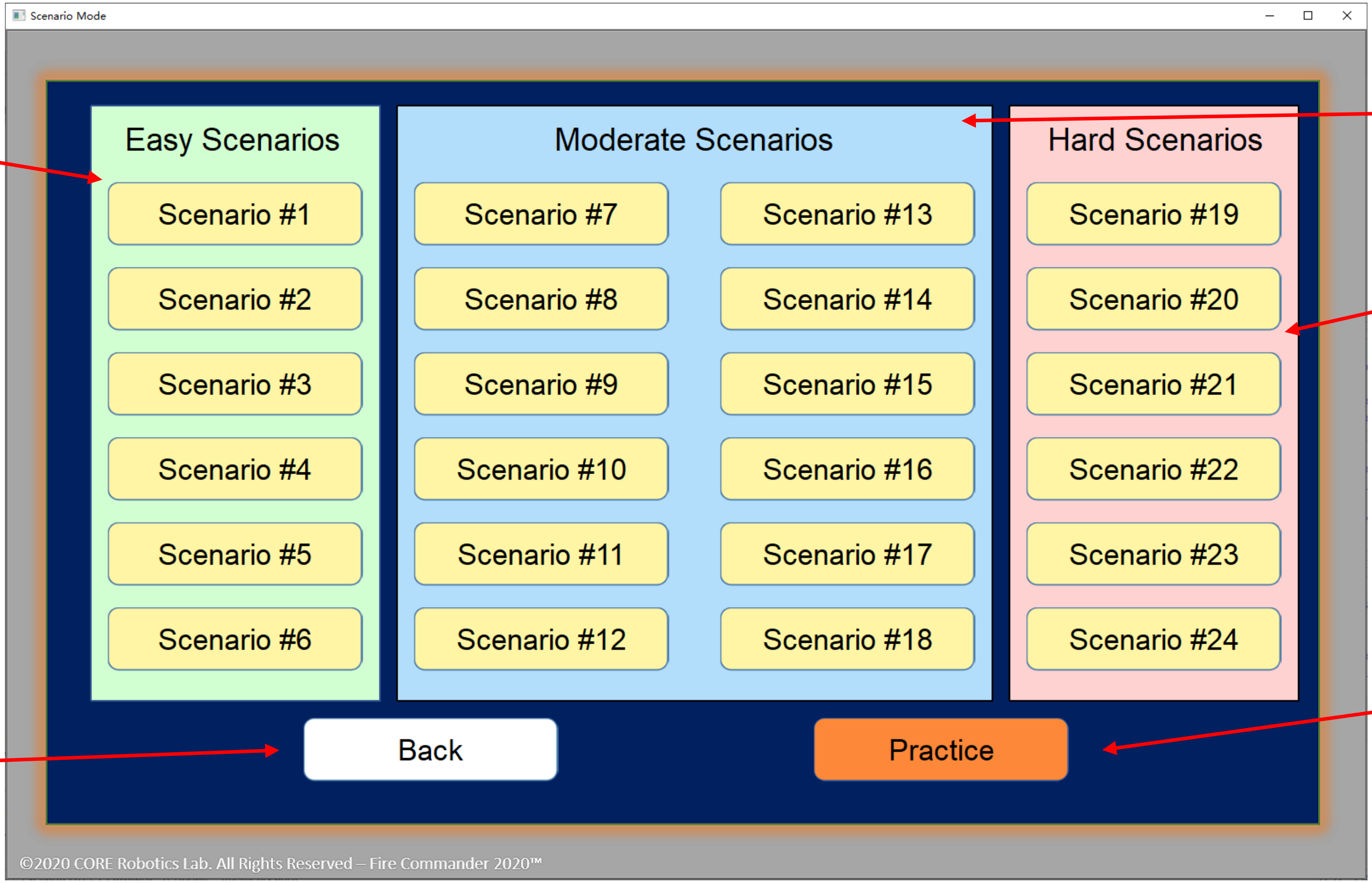
Easy Scenarios (6)

Scenario Button

- Enter the tutorial page when the user presses, then start simulation with built-in parameters
- Store the user data during simulation, calculate the final score

Back

Back to welcome page



Moderate Scenarios (12)

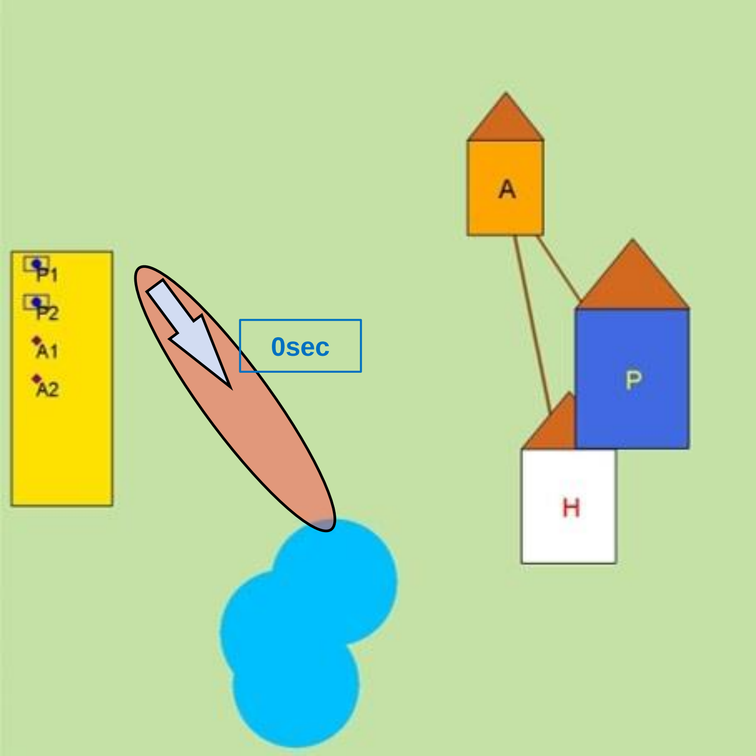
Hard Scenarios (6)

Practice Mode

- Enter the tutorial page when the user presses, then start a simple simulation without storing the user data
- Back to scenario page after simulation

Practice Scenario

- **Motivation**
  - Help the users familiarize the environment setting and operation
  - Only incorporate the simultaneous score calculation and display, disable the online data storage function
  - Back to scenario page for formal simulation trial when the simulation ends



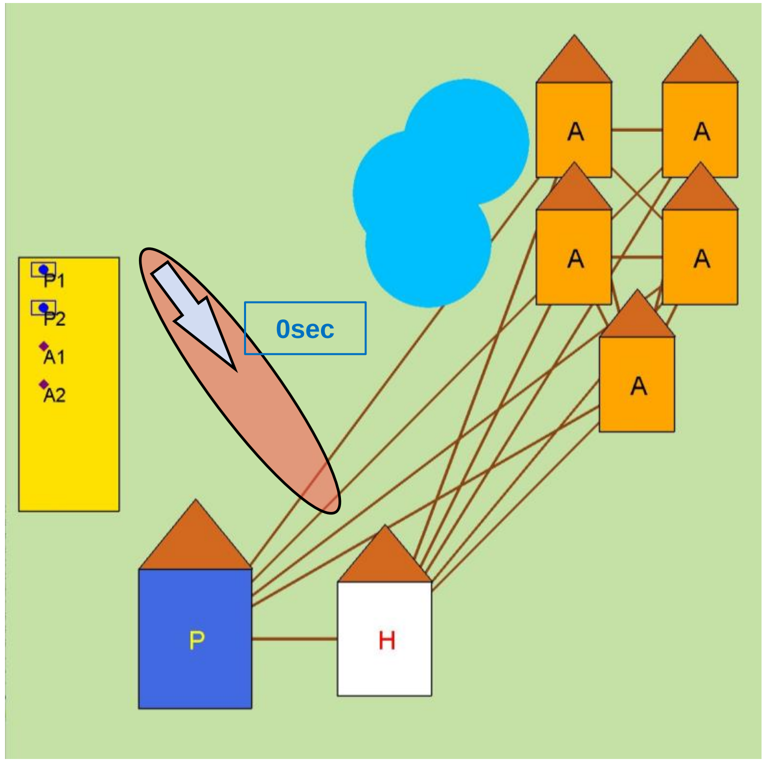
Targets:
- House: 1
- Hospital: 1
- Power Station: 1
- Lake: 1

Agents:
- Perception: 2
- Action: 2

Fire (1 Region):
- Number of Firespots in Each Region: 15
- Fire Delay: 0
- Fuel Coefficient: 10
- Wind Speed: 5
- Wind Direction: 45°

Easy Scenario

- **Scenario #1**

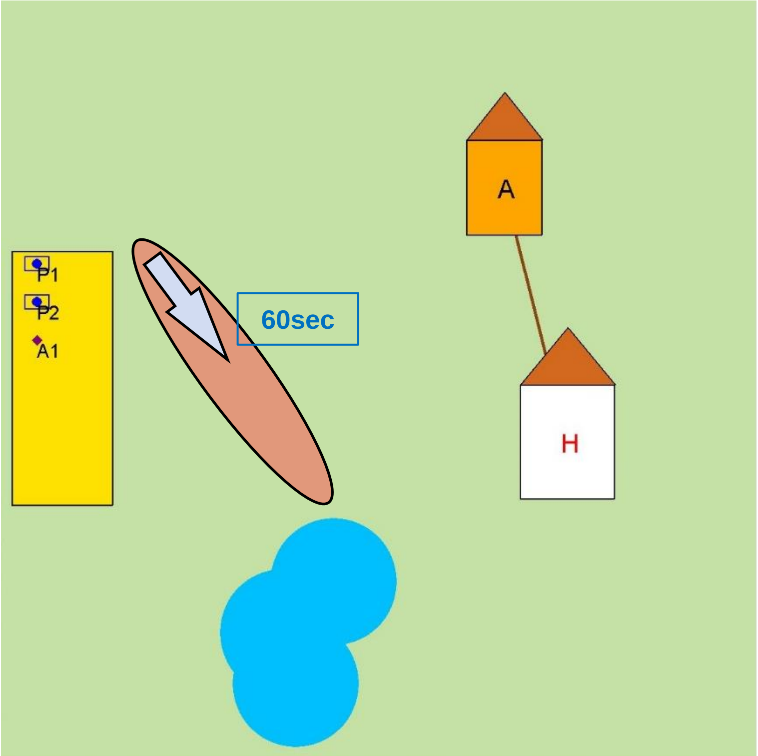


Targets:
- House: 5
- Hospital: 1
- Power Station: 1
- Lake: 1

Agents:
- Perception: 2
- Action: 2

Fire (1 Region):
- Number of Firespots in Each Region: 10
- Fire Delay: 0
- Fuel Coefficient: 10
- Wind Speed: 5
- Wind Direction*: 45°

- **Scenario #2**



Targets:
- House: 1
- Hospital: 1
- Power Station: 0
- Lake: 1

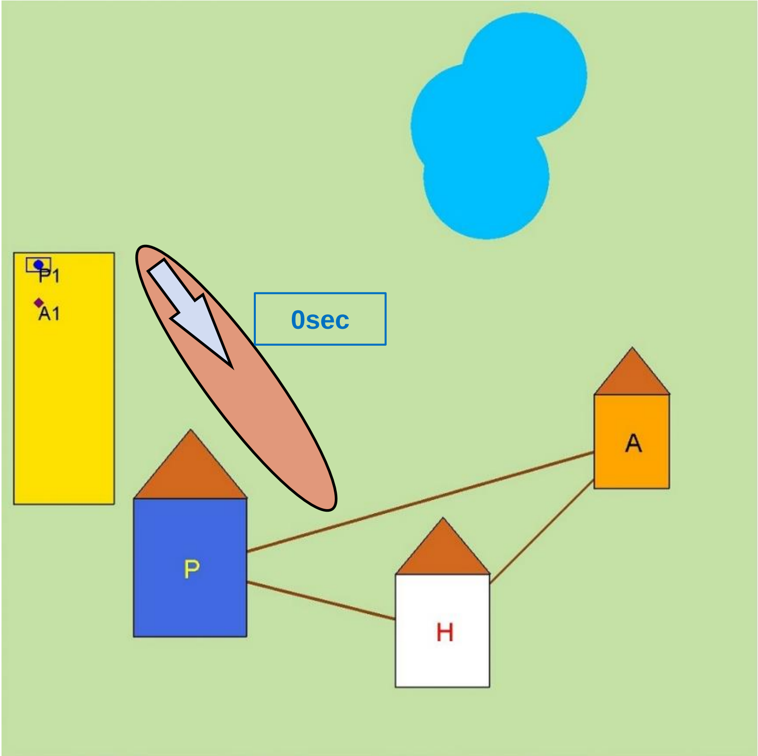
Agents:
- Perception: 2
- Action: 1

Fire (1 Region):
- Number of Firespots in Each Region: 15
- Fire Delay: 60
- Fuel Coefficient: 15
- Wind Speed: 3
- Wind Direction: 45°

***Note:** The wind direction is the counterclockwise angle between the up-to-down axis on the left of the environment and the wind direction arrow

Easy Scenario

- **Scenario #3**

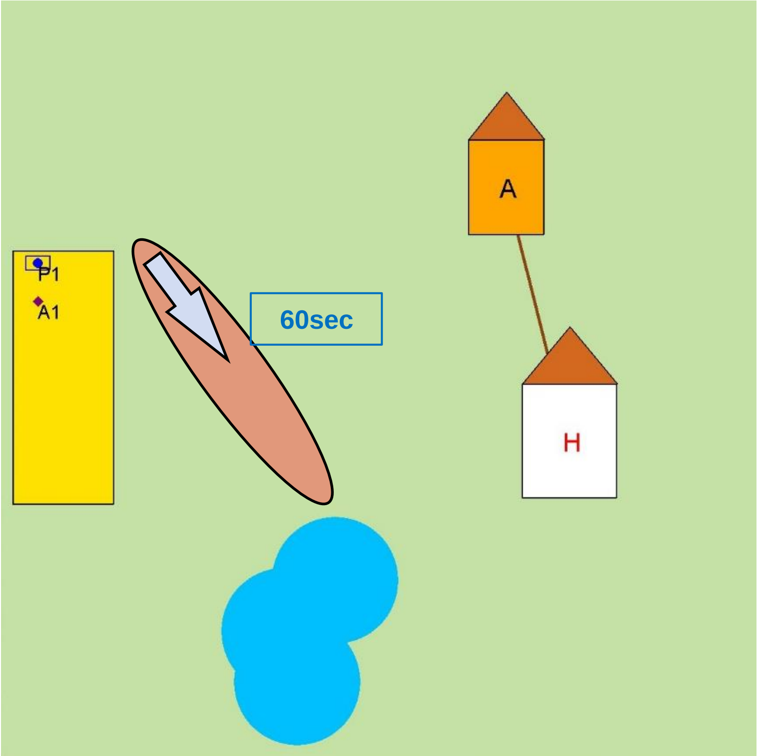


Targets:
- House: 1
- Hospital: 1
- Power Station: 1
- Lake: 1

Agents:
- Perception: 1
- Action: 1

Fire (1 Region):
- Number of Firespots in Each Region: 5
- Fire Delay: 0
- Fuel Coefficient: 15
- Wind Speed: 5
- Wind Direction: 45°

- **Scenario #4**



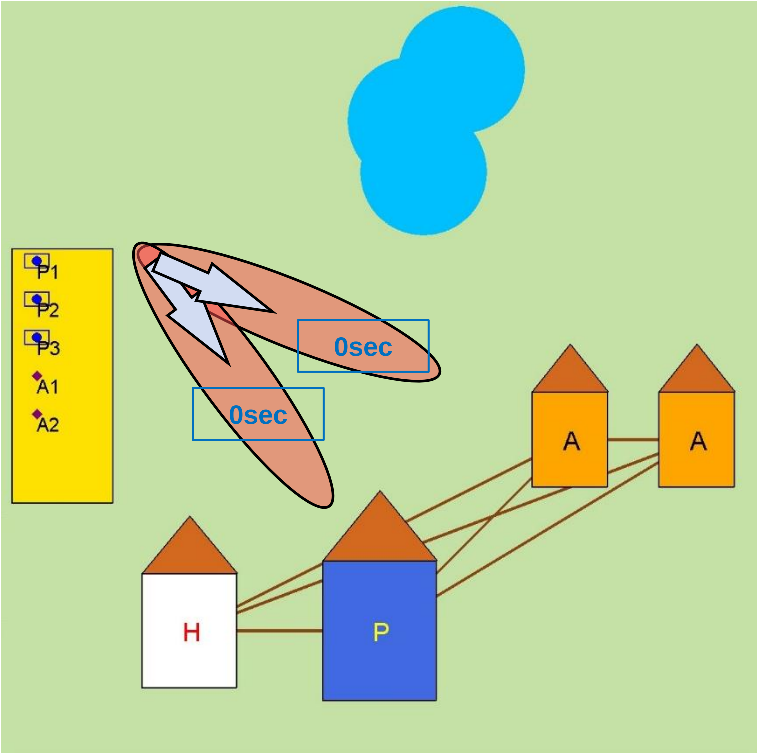
Targets:
- House: 1
- Hospital: 1
- Power Station: 0
- Lake: 1

Agents:
- Perception: 1
- Action: 1

Fire (1 Region):
- Number of Firespots in Each Region: 12
- Fire Delay: 60
- Fuel Coefficient: 5
- Wind Speed: 3
- Wind Direction: 45°

Easy Scenario

- **Scenario #5**

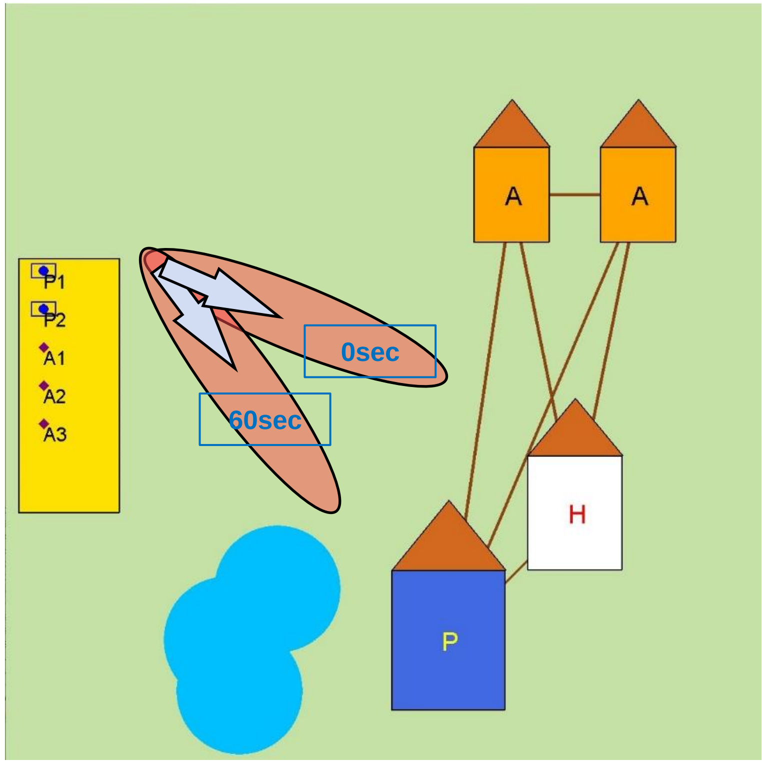


Targets:
- House: 2
- Hospital: 1
- Power Station: 1
- Lake: 1

Agents:
- Perception: 3
- Action: 2

Fire (2 Region):
- Number of Firespots in Each Region: 3, 8
- Fire Delay: 0, 0
- Fuel Coefficient: 5, 10
- Wind Speed: 5, 3
- Wind Direction: 45°, 15°

- **Scenario #6**



Targets:
- House: 2
- Hospital: 1
- Power Station: 1
- Lake: 1

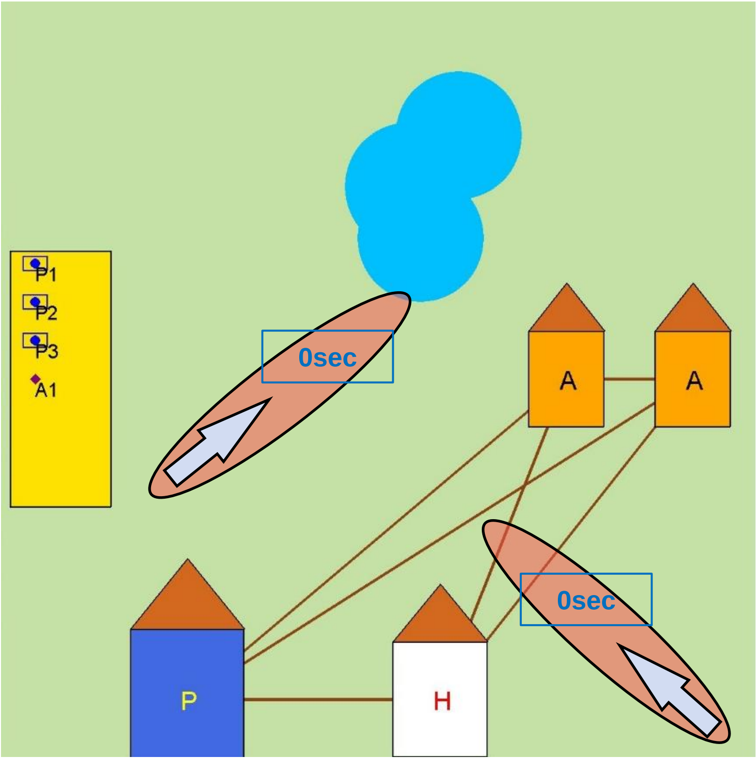
Agents:
- Perception: 2
- Action: 3

Fire (2 Region):
- Number of Firespots in Each Region: 5, 3
- Fire Delay: 60, 0
- Fuel Coefficient: 10, 10
- Wind Speed: 5, 5
- Wind Direction: 45°, 15°

Esmaeil Seraj, Xiyang Wu, Matthew Gombolay

Moderate Scenario

- **Scenario #7**

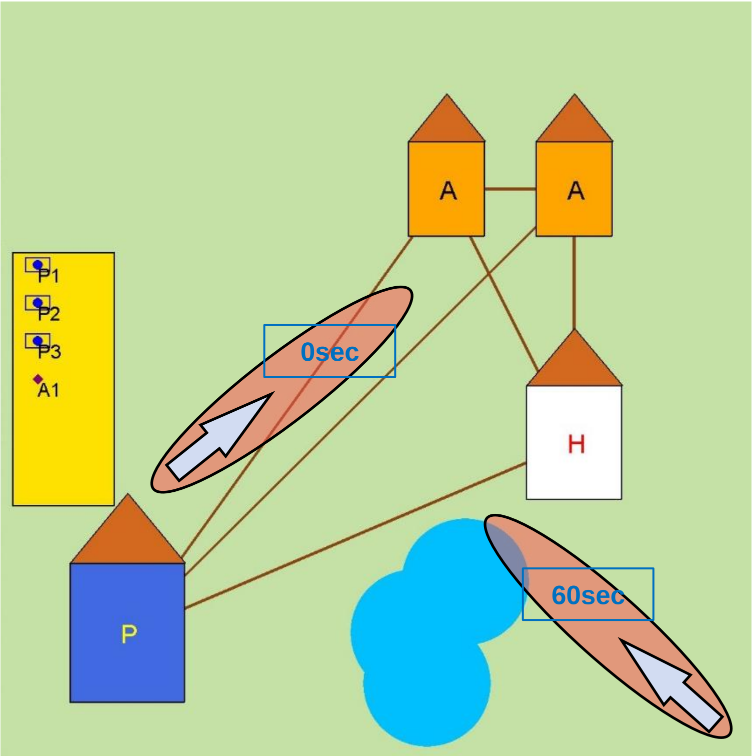


Targets:
- House: 2
- Hospital: 1
- Power Station: 1
- Lake: 1

Agents:
- Perception: 3
- Action: 1

Fire (2 Region):
- Number of Firespots in Each Region: 3, 3
- Fire Delay: 0, 0
- Fuel Coefficient: 5, 5
- Wind Speed: 5, 5
- Wind Direction: 135°, 225°

- **Scenario #8**



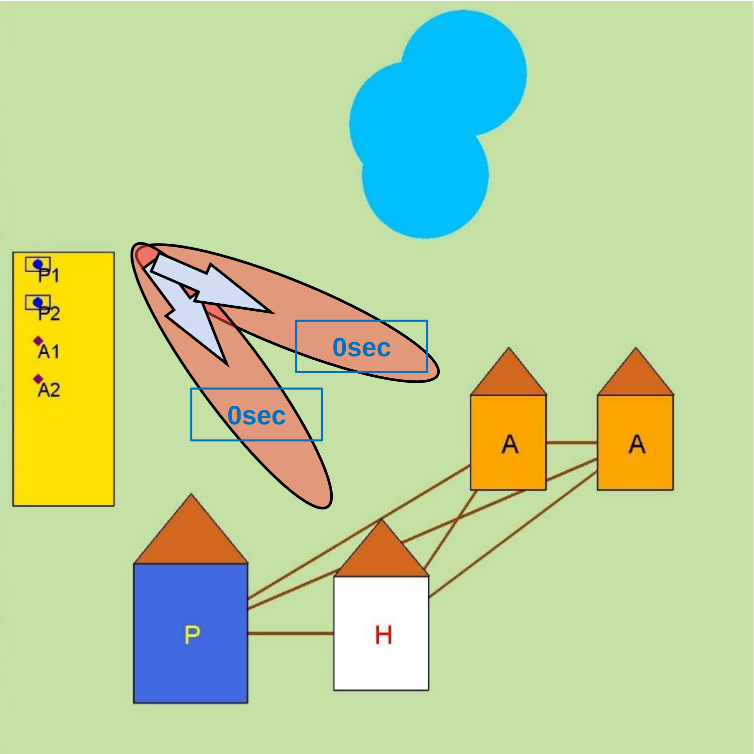
Targets:
- House: 2
- Hospital: 1
- Power Station: 1
- Lake: 1

Agents:
- Perception: 3
- Action: 1

Fire (2 Region):
- Number of Firespots in Each Region: 5, 7
- Fire Delay: 0, 60
- Fuel Coefficient: 3, 3
- Wind Speed: 10, 10
- Wind Direction: 135°, 225°

Moderate Scenario

- **Scenario #9**

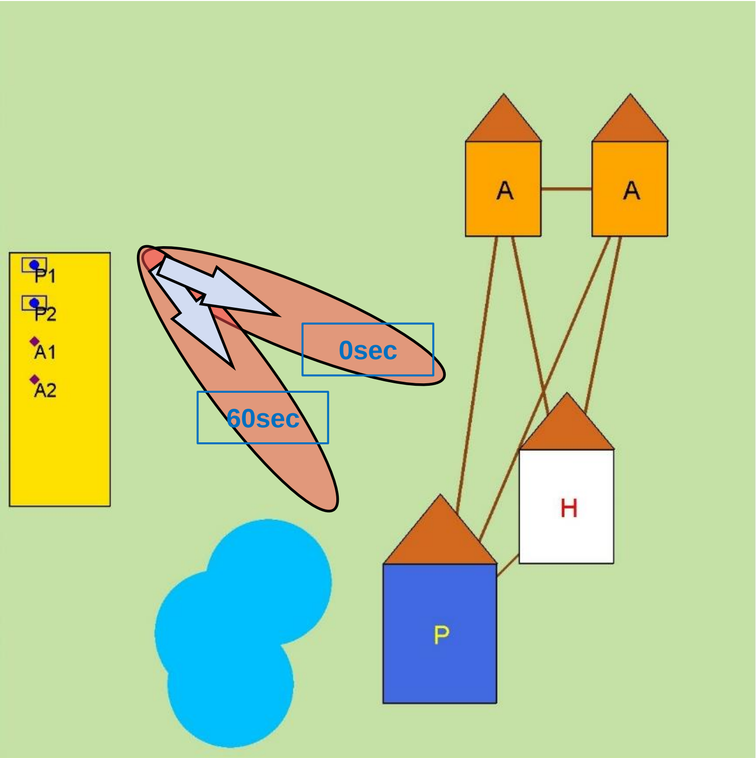


Targets:
- House: 2
- Hospital: 1
- Power Station: 1
- Lake: 1

Agents:
- Perception: 2
- Action: 2

Fire (2 Region):
- Number of Firespots in Each Region: 5, 5
- Fire Delay: 0, 0
- Fuel Coefficient: 10, 10
- Wind Speed: 5, 5
- Wind Direction: 45°, 15°

- **Scenario #10**



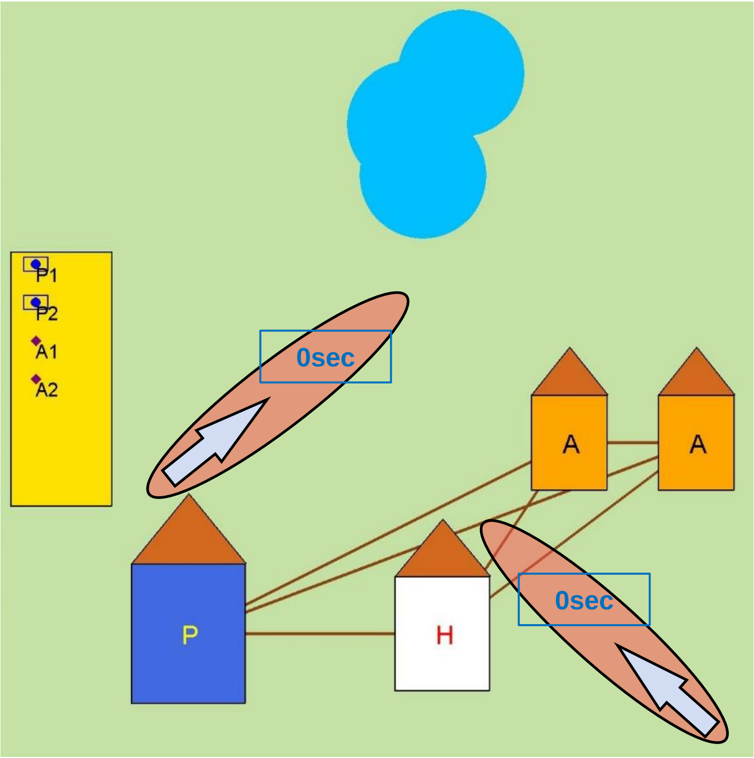
Targets:
- House: 2
- Hospital: 1
- Power Station: 1
- Lake: 1

Agents:
- Perception: 2
- Action: 2

Fire (2 Region):
- Number of Firespots in Each Region: 5,5
- Fire Delay: 60, 0
- Fuel Coefficient: 10, 10
- Wind Speed: 5, 5
- Wind Direction: 45°, 15°

Moderate Scenario

- **Scenario #11**

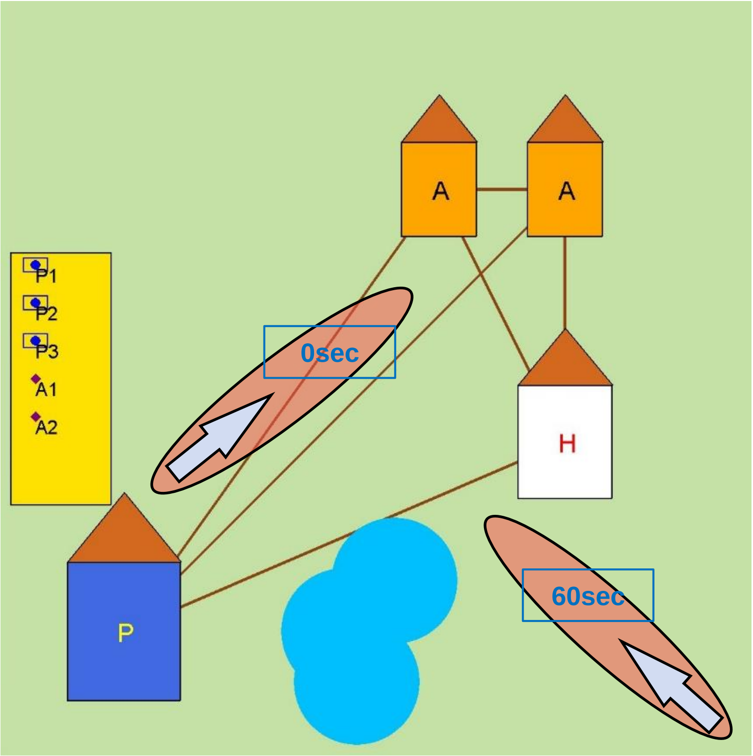


Targets:
- House: 2
- Hospital: 1
- Power Station: 1
- Lake: 1

Agents:
- Perception: 2
- Action: 2

Fire (2 Region):
- Number of Firespots in Each Region: 5, 5
- Fire Delay: 0, 0
- Fuel Coefficient: 5, 5
- Wind Speed: 5, 5
- Wind Direction: 135°, 225°

- **Scenario #12**



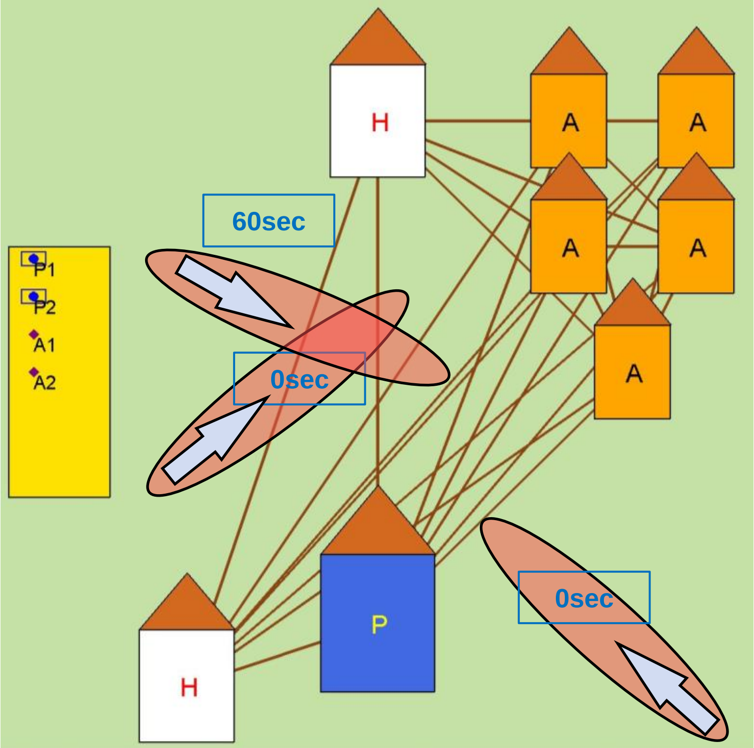
Targets:
- House: 2
- Hospital: 1
- Power Station: 1
- Lake: 1

Agents:
- Perception: 3
- Action: 2

Fire (2 Region):
- Number of Firespots in Each Region: 3, 10
- Fire Delay: 0, 60
- Fuel Coefficient: 5, 10
- Wind Speed: 5, 10
- Wind Direction: 135°, 225°

Moderate Scenario

- **Scenario #13**

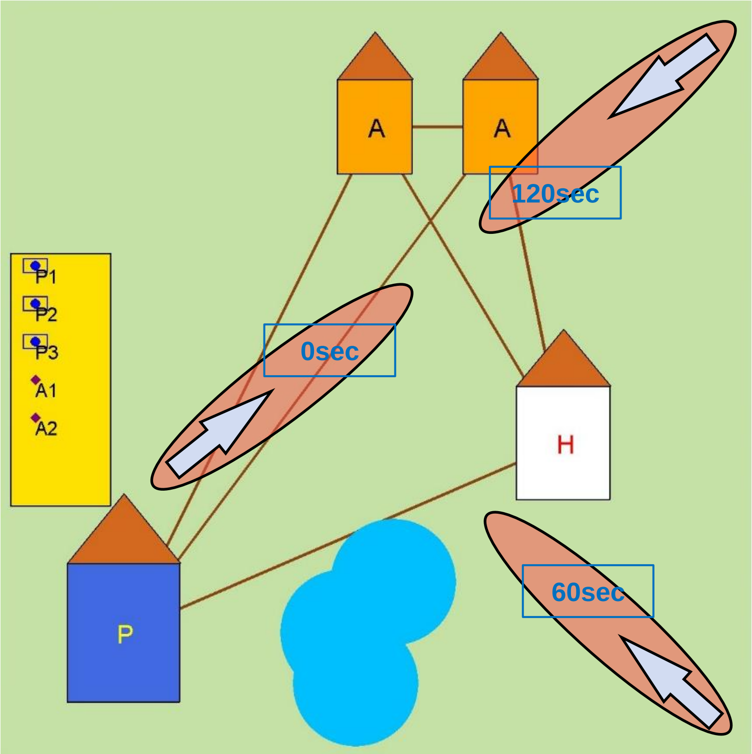


Targets:
- House: 5
- Hospital: 2
- Power Station: 1
- Lake: 0

Agents:
- Perception: 2
- Action: 2

Fire (3 Region):
- Number of Firespots in Each Region: 3, 3, 3
- Fire Delay: 60, 0, 0
- Fuel Coefficient: 10, 10, 10
- Wind Speed: 5, 5, 5
- Wind Direction: 75°, 135°, 225°

- **Scenario #14**



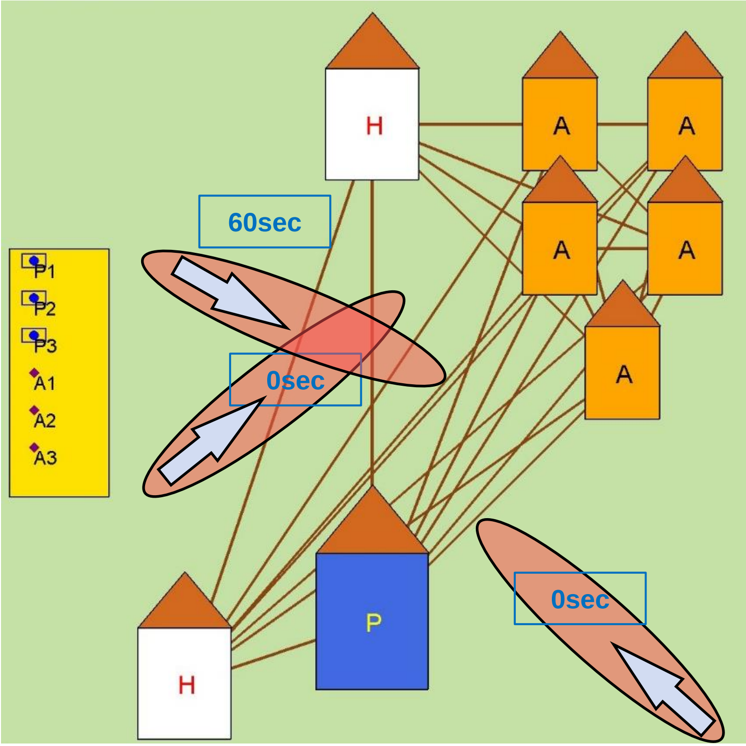
Targets:
- House: 2
- Hospital: 1
- Power Station: 1
- Lake: 1

Agents:
- Perception: 3
- Action: 2

Fire (3 Region):
- Number of Firespots in Each Region: 5, 5, 5
- Fire Delay: 0, 120, 60
- Fuel Coefficient: 10, 10, 10
- Wind Speed: 5, 5, 5
- Wind Direction: 135°, 315°, 225°

Moderate Scenario

- **Scenario #15**

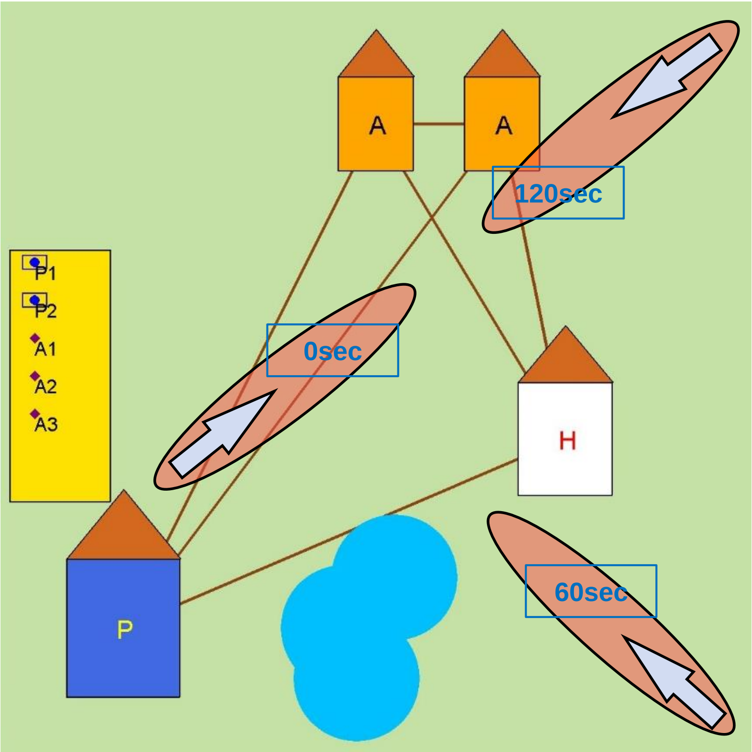


Targets:
- House: 5
- Hospital: 2
- Power Station: 1
- Lake: 0

Agents:
- Perception: 3
- Action: 3

Fire (3 Region):
- Number of Firespots in Each Region: 3, 5, 7
- Fire Delay: 60, 0, 0
- Fuel Coefficient: 10, 10, 10
- Wind Speed: 3, 5, 10
- Wind Direction: 75°, 135°, 225°

- **Scenario #16**



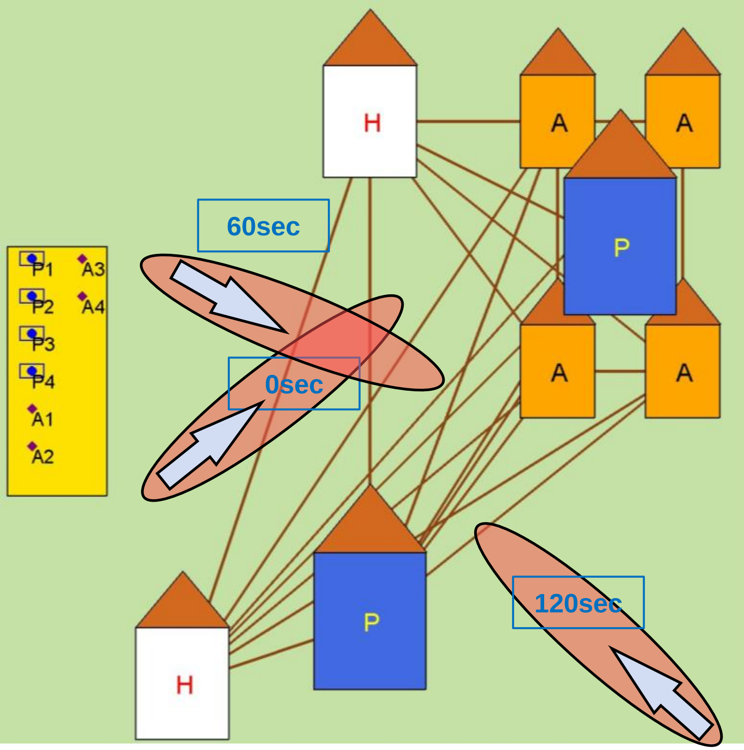
Targets:
- House: 2
- Hospital: 1
- Power Station: 1
- Lake: 1

Agents:
- Perception: 2
- Action: 3

Fire (3 Region):
- Number of Firespots in Each Region: 3, 5, 7
- Fire Delay: 0, 120, 60
- Fuel Coefficient: 3, 5, 10
- Wind Speed: 5, 5, 5
- Wind Direction: 135°, 315°, 225°

Moderate Scenario

- **Scenario #17**

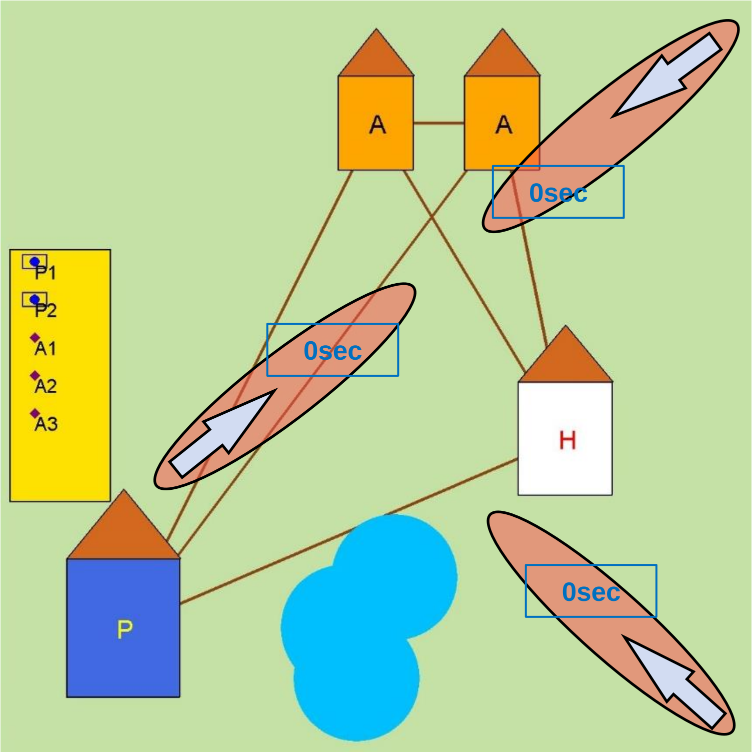


Targets:
- House: 4
- Hospital: 2
- Power Station: 2
- Lake: 0

Agents:
- Perception: 4
- Action: 4

Fire (3 Region):
- Number of Firespots in Each Region: 5, 5, 5
- Fire Delay: 60, 0, 120
- Fuel Coefficient: 10, 10, 10
- Wind Speed: 5, 5, 5
- Wind Direction: 75°, 135°, 225°

- **Scenario #18**



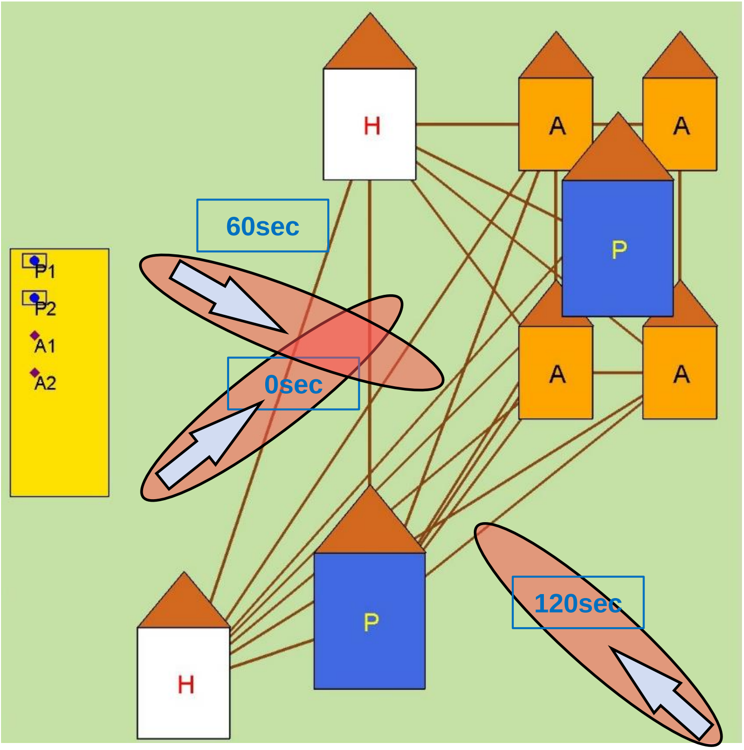
Targets:
- House: 2
- Hospital: 1
- Power Station: 1
- Lake: 1

Agents:
- Perception: 2
- Action: 3

Fire (3 Region):
- Number of Firespots in Each Region: 3, 3, 5
- Fire Delay: 0, 0, 0
- Fuel Coefficient: 5, 5, 10
- Wind Speed: 3, 3, 5
- Wind Direction: 135°, 315°, 225°

Hard Scenario

- **Scenario #19**

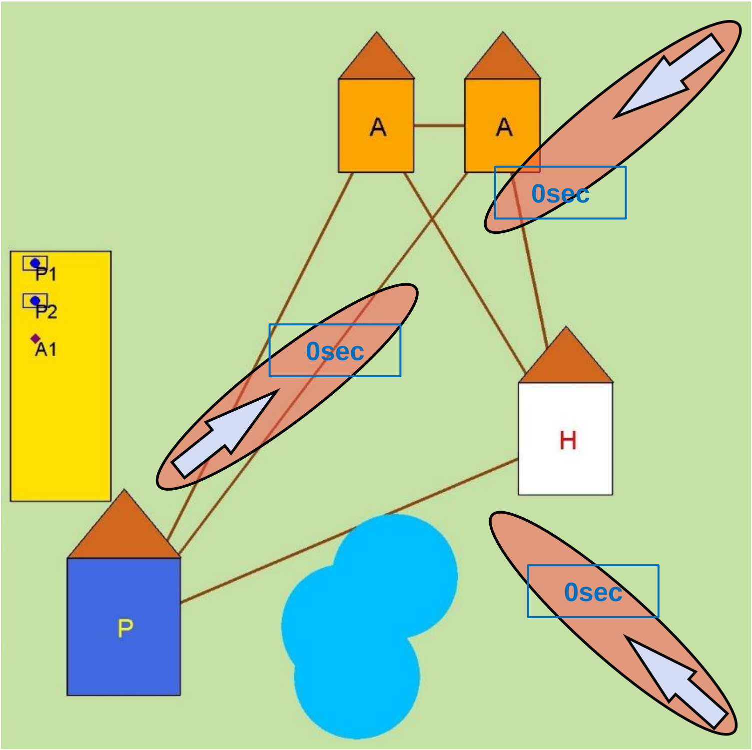


Targets:
- House: 4
- Hospital: 2
- Power Station: 2
- Lake: 0

Agents:
- Perception: 2
- Action: 2

Fire (3 Region):
- Number of Firespots in Each Region: 5, 5, 5
- Fire Delay: 60, 0, 120
- Fuel Coefficient: 10, 10, 10
- Wind Speed: 5, 5, 5
- Wind Direction: 75°, 135°, 225°

- **Scenario #20**



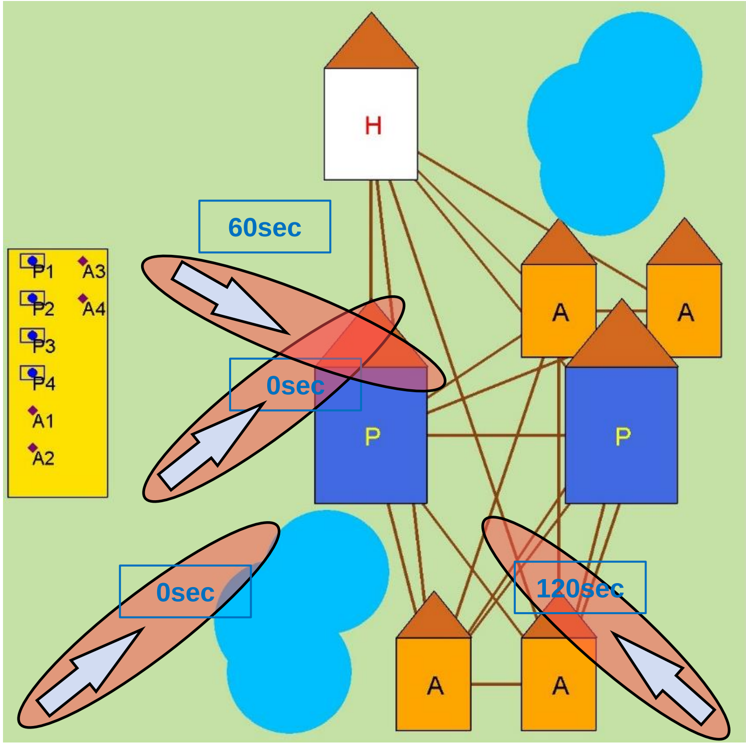
Targets:
- House: 2
- Hospital: 1
- Power Station: 1
- Lake: 1

Agents:
- Perception: 2
- Action: 1

Fire (3 Region):
- Number of Firespots in Each Region: 3, 3, 3
- Fire Delay: 0, 0, 0
- Fuel Coefficient: 5, 5, 5
- Wind Speed: 5, 5, 5
- Wind Direction: 135°, 315°, 225°

Hard Scenario

- **Scenario #21**

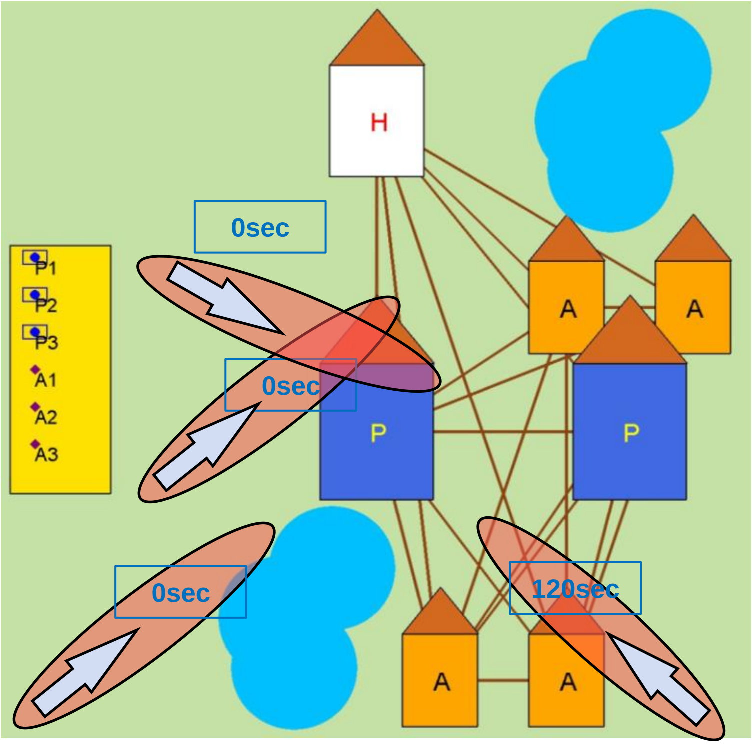


Targets:
- House: 4
- Hospital: 1
- Power Station: 2
- Lake: 2

Agents:
- Perception: 4
- Action: 4

Fire (4 Region):
- Number of Firespots in Each Region: 5, 5, 5, 5
- Fire Delay: 60, 0, 0, 120
- Fuel Coefficient: 10, 10, 10, 10
- Wind Speed: 5, 5, 5, 5
- Wind Direction: 75°, 135°, 135°, 225°

- **Scenario #22**



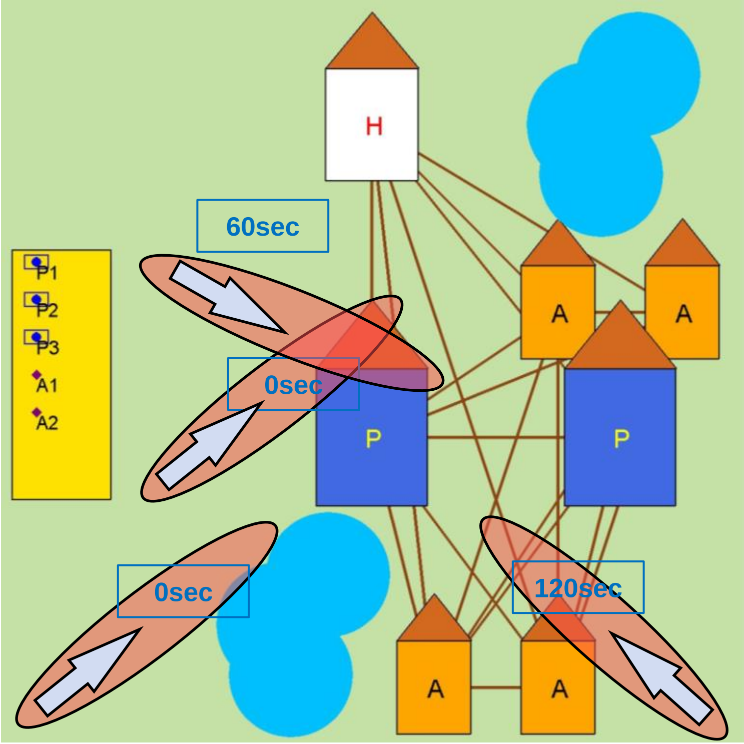
Targets:
- House: 4
- Hospital: 1
- Power Station: 2
- Lake: 2

Agents:
- Perception: 3
- Action: 3

Fire (4 Region):
- Number of Firespots in Each Region: 5, 5, 5, 5
- Fire Delay: 0, 0, 0, 120
- Fuel Coefficient: 5, 5, 10, 10
- Wind Speed: 5, 5, 5, 5
- Wind Direction: 75°, 135°, 135°, 225°

Hard Scenario

- **Scenario #23**

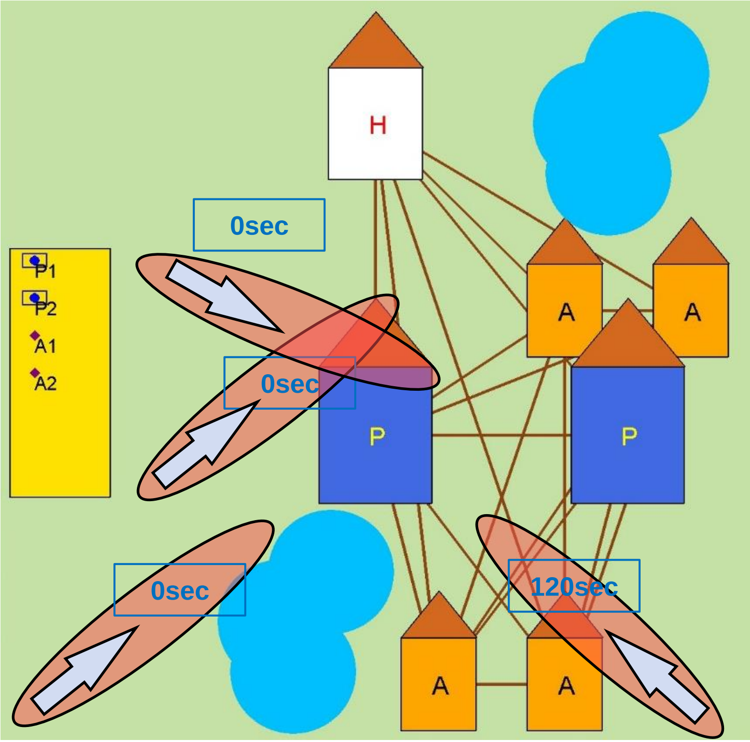


Targets:
- House: 4
- Hospital: 1
- Power Station: 2
- Lake: 2

Agents:
- Perception: 3
- Action: 2

Fire (4 Region):
- Number of Firespots in Each Region: 3, 3, 5, 7
- Fire Delay: 60, 0, 0, 120
- Fuel Coefficient: 3, 5, 3, 10
- Wind Speed: 5, 5, 5, 5
- Wind Direction: 75°, 135°, 135°, 225°

- **Scenario #24**



Targets:
- House: 4
- Hospital: 1
- Power Station: 2
- Lake: 2

Agents:
- Perception: 2
- Action: 2

Fire (4 Region):
- Number of Firespots in Each Region: 3, 5, 7, 8
- Fire Delay: 0, 0, 0, 120
- Fuel Coefficient: 8, 8, 8, 8
- Wind Speed: 3, 3, 3, 3
- Wind Direction: 75°, 135°, 135°, 225°

- **References:**

1. Esmaeil Seraj, Xiyang Wu, and Matthew C. Gombolay. “**Firecommander 2020**.” GitHub Repository, Available Online: https://github.com/EsiSeraj/FireCommander2020, 2020.

2. Seraj, Esmaeil, Andrew Silva, and Matthew Gombolay. "**Safe coordination of human-robot firefighting teams**." arXiv preprint arXiv:1903.06847 (2019).

3. Seraj, Esmaeil, et al. "**Adaptive Leader-Follower Control for Multi-Robot Teams with Uncertain Network Structure**." 2021 American Control Conference (ACC). IEEE, 2021.

4. Seraj, Esmaeil, et al. "**Heterogeneous Graph Attention Networks for Learning Diverse Communication.**" arXiv preprint arXiv:2108.09568 (2021).

5. Seraj, Esmaeil, Letian Chen, and Matthew C. Gombolay. "**A hierarchical coordination framework for joint perception-action tasks in composite robot teams**." IEEE Transactions on Robotics (2021).

6. Seraj, Esmaeil, and Matthew Gombolay. "**Coordinated control of uavs for human-centered active sensing of wildfires**." 2020 American Control Conference (ACC). IEEE, 2020.

7. Seraj, Esmaeil, and Matthew Gombolay. ”**Coordinated Control of UAVs for Human-Centered Active Sensing of Wildfires-Presentation Slides**.” Diss. Georgia Institute of Technology, USA, 2020.

8. Seraj, Esmaeil, Xiyang Wu, and Matthew Gombolay. "**FireCommander: An Interactive, Probabilistic Multi-agent Environment for Joint Perception-Action Tasks**." arXiv preprint arXiv:2011.00165 (2020).

9. Seraj, Esmaeil, Xiyang Wu, and Matthew Gombolay. "**FireCommander: An Interactive, Probabilistic Multi-agent Environment for Joint Perception-Action Tasks-Tutorial Slides**." Diss. Georgia Institute of Technology, USA, 2020.