Conceptual Design Review

Team H

PhoeniX

UAV-AGV Fire-fighting System

Team Members:
Akshit Gandhi
Shubham Garg
Parv Parkhiya
Zhihao Zhu (Steve)

Sponsors:
Oliver Kroemer
Sebastian Scherer

Advisors:
Dimitrios Apostolopoulos
John Dolan

Submission Date: 8th December 2018

Table of Contents

2. Project Description

On 8th October 1871 a small barn in Chicago caught fire because of unknown reason and what followed was a conflagration lasting 3 days that killed up to 300 people and made 100,000 residents homeless. In aftermath of this Great Chicago Fire, Chicago and many other cities updated and implemented better fire safety code. One and half century later, we are still learning to tackle the fire.

In year 2011 in US alone, following table shows the devastating number of fire incidents and their negative impact.

Table 1 Reported Fire Incidents in 2011 in US

Outside (Forest)	686,000
Structure (Building)	484,500
Vehicle	219,000
Total	1,389,500

[1]

Table 2 Fire Loss

Damage	Structure	Outside	Vehicle	Total
Property (Billion USD)	9.7	0.616	1.4	11.7
Civilian Injuries	15,635	675	1,190	17,500
Civilian Deaths	2,640	65	300	3,005

[1]

Time is of the essence. In the early stage of fire, it may be possible to put out a fire using just a bucket of water or single CO_2 cylinder. Usually fire starts small but spreads rapidly and if it gets its gigantic form, it may take hundreds of people and days to extinguish.

An autonomous multiagent system with navigation, perception capabilities and mechanism to deploy fire extinguishing material at best can extinguish the fire before it can spread and at very least can act as first response collecting information about surroundings (map) and location of fire which human firefighters can use to make better judgements.

PhoeniX team proposes an multiagent heterogeneous UAV (Unmanned Aerial Vehicle) & AGV (Automated Ground Vehicle) firefighting system that given a fire alarm signal can autonomously search for fire inside the building as well as surrounding and deploy the extinguishing material to put out fire.

3. Use Case

North Oakland is a Pittsburgh community located near the world-famous Carnegie Mellon University. At the night of 11th Nov 2018, a three-floor building caught fire. Ten minutes within the receipt of the fire notification, fire fighters reached an open area (we refer as base station) near the target building, and they put PhoeniX firefighting system on the ground and send an initiation signal to the PhoeniX system. The system becomes active and 3 UAVs takes-off from the station and an AGV also drives towards the building.

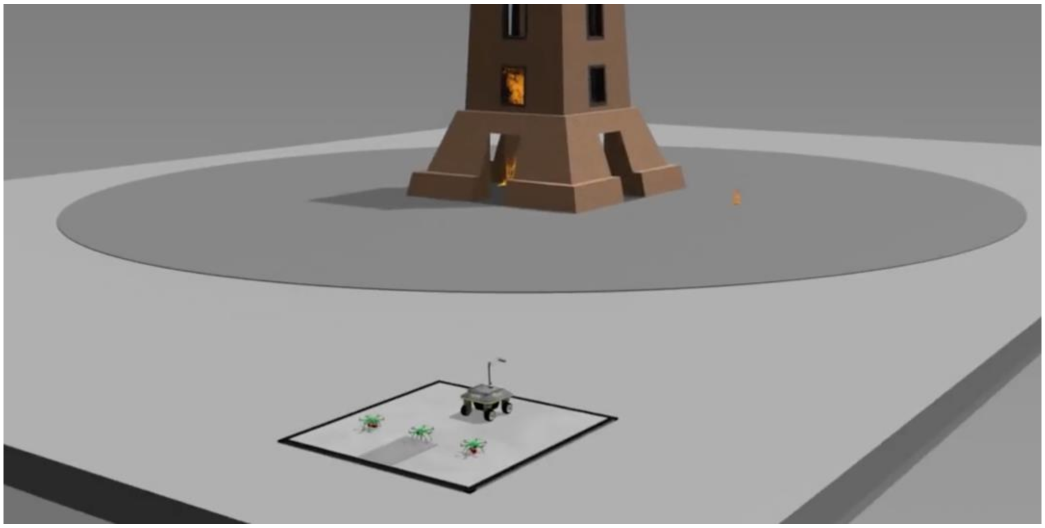

Figure 1 PhoeniX system at base station

All the robots coordinate and collaborate to optimally explore the surrounding by avoiding obstacles while creating a map of the environment. UAV-2 detects fire in the building at two locations: ground floor and 1st floor, it shares that information with other systems.

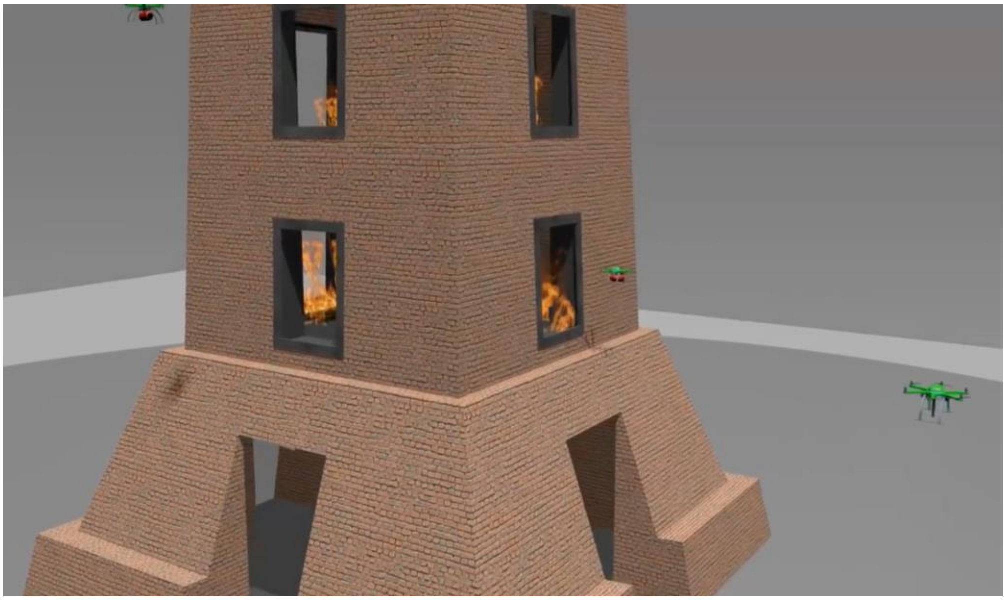

Figure 2 UAV flying towards fire location

The system divides the task of extinguishing fire at those two locations. The AGV is assigned the task of extinguishing fire at the ground floor and 2 UAVs are assigned the first floor. While the 3rd UAV is still exploring to find potential fire locations. The AGV uses sweeping strategy to extinguish fire whereas the UAVs use some different mechanism to extinguish fire depending on the fire location.

Figure 3 UAV extinguishing fire

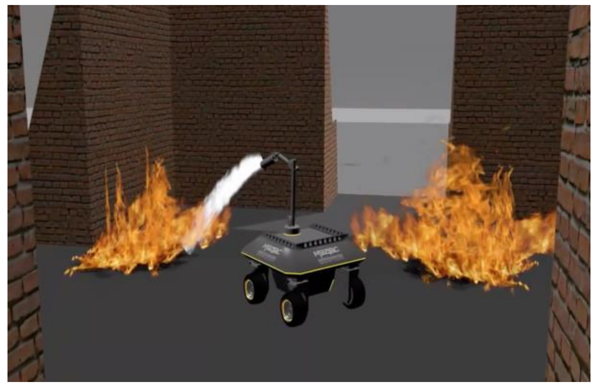

Figure 4 AGV extinguishing fire

Every robot monitors their fire extinguishing progress. AGV reports that it has successfully extinguished the fire. When the 1st and 2nd UAVs are out of the firefighting material, they request help from the 3rd UAV. The third UAV comes and extinguishes the fire. After ensuring that there is no more fire in the building the UAVs lands back at the station along with the AGV driving back.

4. System-Level Requirements

Since all our performance and non-functional requirements have been derived from the functional requirements and the objective tree (attached in the Appendix), the table below depicts a one-to-one mapping between the mandatory requirements.

4.1. Mandatory Requirements

Table 3 Mandatory Performance Requirements

Requirement ID	Requirement Description	
F.R.1	Take-Off and Land from base station	
	M.P.1	Land within 5 m radius from center of base station
F.R.2	Plan Trajectory	
	M.P.2	Explore 50 m x 60 m x 20 m environment with greater than 60% coverage in 10 minutes or less
F.R.3 F.R.4	Create real-time map	
	Localize itself in the environment	
	M.P.3	Accumulate less than 5 m drift for every 100 m of distance travelled
F.R.5	Traverse desired trajectory	
	M.P.4	Maximum error between desired and actual trajectory should be less than 1 m

F.R.6	Avoid collision with obstacles and other UAVs/AGV	
	M.P.5	Keep 2 m minimum distance between system and fire
F.R.7	Detect Fire	
	M.P.6	Achieve fire detection AUC (Area under curve) of ROC (Receiver Operating Characteristic) of 0.65
	M.P.7	Detect fire from 2.5 m away
F.R.8	Localize and Monitor Fire	
	M.P.8	Localize fire with less than 1 m error
F.R.9	Deploy material strategically	
	M.P.9	Carry 1 kg of extinguishing material each
	M.P.10	Deposit 40% deployed extinguishing material on the target area of minimum 0.5 m x 0.5 m
F.R.10	Coordinate between different UAVs & AGV	
	M.P.11	Reliable communication within 25 m

Table 4 Mandatory Nonfunctional Requirements

Requirement ID	Requirement Description
M.N.1	Fit in the size of 1.2m x 1.2m x 0.5m (UAV)
M.N.2	Fit in volume of 1.7m x 1.5 m x 2m (AGV)
M.N.3	Feature propeller guards
M.N.4	Feature kill switch for safety
M.N.5	Feature User Interface
M.N.6	Maintainable with easily replaceable components like motor, battery, ESCs etc.
M.N.7	Resist wind speed up to 2-3 knots
M.N.8	Interoperate with other MBZIRC team's systems by the means of functional modularity

4.2. Desirable Requirements

Table 5 Desirable Requirements

Requirement Type	Requirement ID	Requirement Description
Performance	D.P.1	Dock to refill the extinguishing material when it is below 10% level within 5 minutes
	D.P.2	Dock for battery recharge/battery replacement when it is below 20% level within 5 minutes
	D.P.3	Detect humans trapped inside the building with 60% accuracy
	D.P.4	Notify authorities about the location of people trapped inside to plan rescue mission within 45 seconds of human detection
Non-functional	D.N.1	Perform non-overlapping tasks (mapping, extinguishing, etc.)
	D.N.2	Create a common global map by merging maps from different systems
	D.P.3	Portable (weight, compact size/form factor within 0.5m x 0.5m x 0.25m)
	D.N.3	Economical (system costs under $7000)
	D.N.4	Scalable (in terms of manufacturing)

5. Functional Architecture

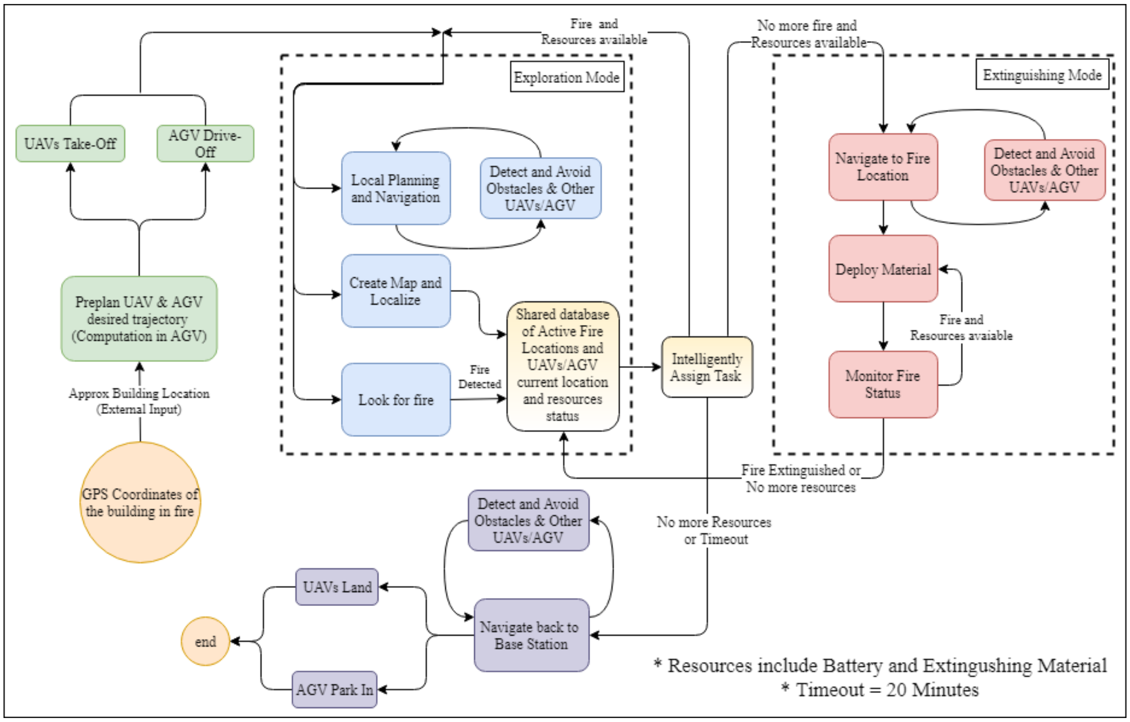

Figure 3 Functional Architecture of PhoeniX Fire Fighting System

The functional architecture of the PhoeniX firefighting system has been depicted in above figure. It captures all the functionalities we derived from the functional requirements and the objectives tree. The operation of the system begins when an operator (person/system) triggers a start signal to the system which will be in the form of the approximate GPS location of the building under fire. Given this start signal the UAVs take off and the AGV drives off following a trajectory precomputed by the AGV onboard and sent to the UAVs within 2 minutes of the system initiation. While they travel towards the fire location, the systems start to map the environment while avoiding any obstacles. This map will be used by the system to plan path from point A to point B.

While the systems are exploring the environment, they will also keep on checking for any potential fire locations by using the fire detection subsystem. Once a system identifies a location with fire, it will inform other systems by adding the location of the fire in a shared database. If there are fire extinguishing tasks pending in the database, an intelligent task assigner/scheduler will command a system with enough extinguishing resources & battery to navigate to the fire location and extinguish fire. So, once the systems will receive the co-ordinates of the fire location and they shall autonomously navigate in the environment while avoiding obstacles and once they reach the proximity of fire, they will orient themselves in an appropriate position to extinguish fire. Now the system will deploy the extinguishing payload using a specific determined strategy and update the database when they recognize that they have extinguished the fire. So once every potential fire location inside the building has been extinguished or a timeout of 20 minutes has passed, the system shall return to the base station. The system may also return to base if it runs out of the extinguisher material or battery.

6. Trade Studies

6.1. System-level trade Study

We have done a system level trade study between human firefighters, sensor based traditional sprinkler system and AGV + UAV collaborative firefighting. We ranked all attributes on a scale from 1 to 5, with 5 being the most favorable. Of all the attributes, response time and life risk are the most critical aspects of the fire extinguisher system. Sensor based traditional sprinkler system has response time between 7sec to 33sec [2], AGV+UAV can respond within a minute based on the base station's location while firefighters can take around four minutes to respond. Human firefighters have huge life risk but at the same time they are much more robust in extinguishing fire as compared to the autonomous systems. Fire loss is minimum with human firefighters as they can extinguish fire at large scale as they have a large extinguishing material carrying capacity but the UAV + AGV autonomous system has limited payload carrying capacity. AGV+UAV firefighting system can be placed at multiple remote locations which is not easily accessible to human firefighters. Similarly, sensor-based sprinkler system can't be installed everywhere.

Table 6 System level Trade Study

Criteria	Weight (in %)	Human Firefighters	Sensor based fire sprinkler system	UAV + AGV
Response Time	20	3.5	4.5	4.0
Life Risk	20	2.5	4.0	4.5
Fire loss	15	4.5	2.5	4.0
Robustness	15	4.5	4.0	3.5
Availability	15	3.0	3.5	4.0
Complexity	10	4.5	4.0	3.5
Setup cost	5	3.0	4.5	4.5
Total	100	3.6	3.8	**4.0**

6.2. Component-level trade Study

For AGV+UAV firefighting system, we did four different component level trade studies. Based on the trade-off between weight and localization accuracy, we found 2D LiDAR (depth map), stereo camera (3D map) and IMU (odometry data) forms the best combination for localization on AGV. Due to weight constraints, 2D LiDAR doesn't seems feasible on UAV. Also, it increases the software, hardware complexity and power consumption of UAV. So, localization on UAV will done using stereo camera and IMU. Tilted rotor Hexacopter [3] will be most appropriate for our system because of its stability with external disturbances and robust attitude control. Communication between different UAVs and AGV is also crucial for our task. So, based on the trade study we found Wi-Fi [4] as the best reliable & easy to use mode of communication.

Table 7 Trade Study on UAV Platform

Criteria	Weight (in %)	Hexacopter	Hexacopter with tilted rotor	Quadcopter
Payload Capacity (M.R.3)	20	2.5	2.2	1.2
Reliability/Safety of structure (M.N.3)	10	3	3	1.7
Power/Thrust (M.R.3)	10	2.7	2.7	2
Stability (M.N.6)	15	2.4	4.4	2.8
Ease of Maintenance (M.N.5)	10	2.7	2.4	3
Cost	5	2	2	2.7
Control (M.N. 6)	20	3.8	4.2	3.3
Size (M.N.1)	10	3.4	3.2	4.5
Total	100	2.853	**3.111**	2.55

Table 8 Wireless Communication Trade Study

Criteria	Weight (in %)	Wi-Fi	Bluetooth	Cellular Technology	Zigbee
Reliability (F.R.11)	10	4.4	3.1	4.2	4.2
Power Consumption (M.R.3)	20	4	4.2	3.9	4.7
Signal Penetration	10	3.5	3	4	4
Operating Range (M.R.1)	20	4	2.6	3.5	3.7
Live Internet Connectivity	5	3	0	2.5	0
Additional HW Requirement	5	3.8	4	3.2	4
Ease of Integration	25	4.5	3.4	3.2	3.6
Location Dependency	5	1.2	2	2	3
Total	100	**4.2**	2.44	3.345	3.03

Table 9 Mapping and Localization Software Trade Study [5],[6],[7],[8]

Criteria	Weight (in %)	ORB SLAM 2	Distributed ORB SLAM	Visual SLAM (DSO)	SVO (Semi-Direct Visual Odometry)
Speed (M.R.1)	15	3.5	3.5	3	4
Reliability (M.R.2)	25	4	3.5	2	2
Integration Simplicity	25	4.5	3	2.5	2.5
Stereo Support	10	5	2.5	5	0
IMU support (M.R.2)	10	0	0	5	0
Multiagent Support (F.R.11)	10	0	5	0	0
Density of point cloud (M.R.2)	10	2	2	3	2
Total	100	**3.35**	3.1	2.875	1.925

7. Cyber-Physical Architecture

Below figure shows our cyber-physical architecture which maps the flow of data and energy between components and subsystems based on the results on our trade studies.

1. Input: GPS coordinates of the target location is being sent to the System.

2. UAV/AGV System: Based on the trade studies, we have finalized combination of stereo camera, IMU, GPS as mapping sensors on UAV and an additional 2D LiDAR is added on AGV for better localization. Different mobile robots communicate via Wi-Fi link.

3. Exploration Mode: Mapping sensors will generate an occupancy grid map for the obstacle avoidance and planning local trajectory. Each mobile robot will run YOLO [9] fire detection classifier for fire detection.

4. Scheduler: Each mobile robot will detect fire and update the shared database. Based on the fire location in shared database, Scheduler will assign fire extinguishing task to different robots based on their locations from the fire.

5. Extinguishing Mode: Based on the global fire location, mobile robots will do visual servoing [12] towards the assigned fire location, monitor and extinguish fire.

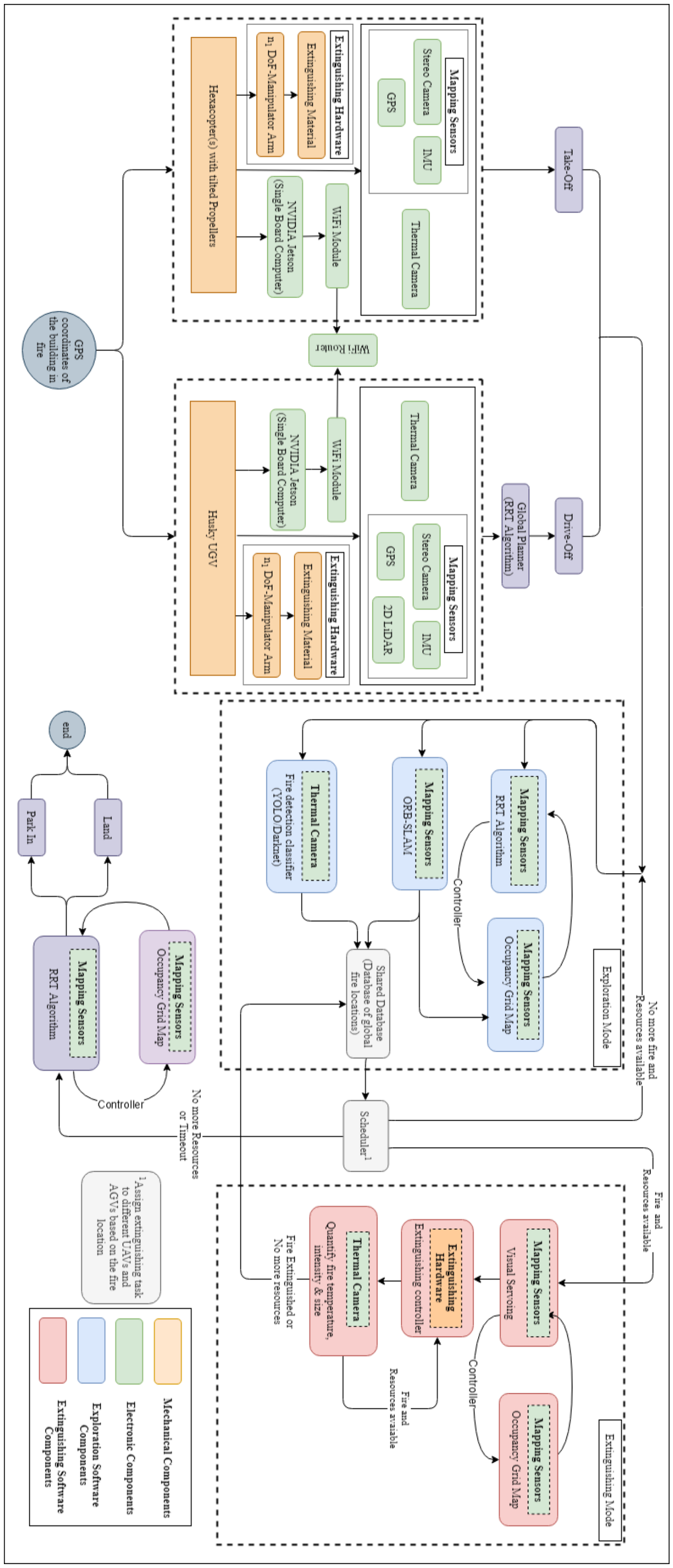

Figure 4 Cyber-Physical Architecture

8. Subsystem Descriptions

8.1. Hardware Subsystems

8.1.1. UAV Subsystem:

While the design of the base UAV (tilted rotor hexacopter) is provided by the sponsors (Sebastian Scherer and Oliver Kroemer), the firefighting task necessitates to modify existing design to integrate thermal and stereo camera sensors as well as mechanism for attaching and deploying payload (extinguishing material) at the target. For low level control in deploying mechanism, microcontroller and actuators would be required and for processing thermal image data, single board computer (Nvidia Jetson) would be required. Integrating these modules would requires us to redesign power distribution board as well. UAV hardware subsystem will handle modifying existing design to handle above mentioned requirements. UAV hardware system will include assembling various parts to build the UAV (under guidance of Air Lab, CMU). It will also provide basic user interface along with a kill switch.

Fallback:

In case titled rotor hexacopter is not able to generate necessary thrust to lift enough extinguishing material, standard hexacopter configuration would be used. In unfortunate event of non-repairable UAV breakdown near project end, commercial pre-built UAV could be used or take inspiration from the UAV being used for the Darpa SubT challenge.

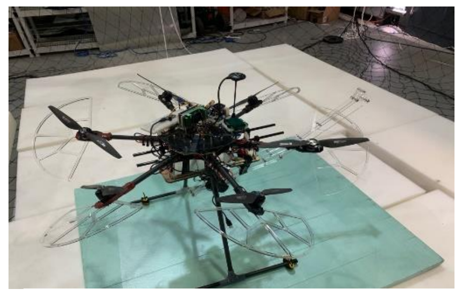

Figure 5 Tilted Hexacopter, Air lab

8.1.2. AGV Subsystem:

Pre-built base AGV system (Husky) is provided by the sponsors but like UAV system certain modifications are required to do firefighting task. Major difference compare to UAV is that there are no power constraints or significant payload constraints which would allow us to integrate LiDAR sensors for robust collision avoidance along with stereo and thermal camera sensors and extra extinguishing material. Design for attaching and deploying extinguishing material at target would be quite different and would require more Degree of Freedom since yaw and height of AGV can't be changed. Single Board Computer as well as Wi-Fi Base station would also needs to be integrated. AGV Subsystem will take of designing and physically integrating parts together. It will also provide basic user interface with kill switch.

Fallback:

In case of non-repairable, non-replaceable breakdown of AGV, UAV will handle the fire extinguishing task at ground floor as well. (Inverse is not true since AGV can't go to the higher floors as far as the MBZIRC test site is concerned)

Figure 6 Husky AGV

8.2. Software Subsystems

There are some key differences between UAV and AGV system but for the most part software subsystem developed for one can be used in other with minor modification. Considering time and resources available for MRSD project, it doesn't seem feasible to create separate software subsystem for UAV and AGV. This implies that all the software solutions would be generic such that it can work on both UAV and AGV but may not be optimum individually.

8.2.1. Simultaneous Localization and Mapping (SLAM) Subsystem:

Since exact 3D map of the outdoor or indoor of the environment is not provided, we need to create 3D map of the environment which would allow Path Planning Subsystem to avoid collision and Navigation Control Subsystem to generate better control signals. While distributed multi-agent SLAM would lead to better mapping and localization, our UAV/AGV system will not do distributed SLAM due to added complexity and reliance on high communication bandwidth. UAV and AGV do communicate about detected fire location but both can act and operate independently and are not handicapped in case of communication failure. SLAM Subsystem will provide "real time" map construction and localization in the environment using stereo camera sensor and will be computationally efficient to run on UAV as well.

Fallback:

Visual SLAM relies heavily on features present in the scene and lack of which can lead to breaking of SLAM during the run. Such failure is fairly probable. In such cases, system would rely on GPS sensor data to localize till SLAM re-initializes detecting features.

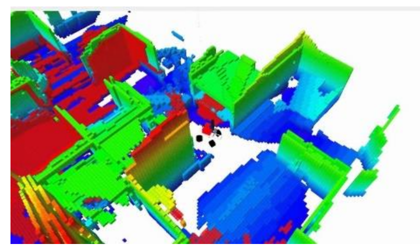

Figure 7 3D point cloud created by SLAM

8.2.2. Path Planning Subsystem:

PhoeniX firefighting system is essentially a heterogeneous multiagent system. While approximate idea about location of building would be known to the system, exact fire location in the building is not known. UAV and AGV both need to collaboratively search for fire in the environment efficiently i.e. we don't want both systems to search overlapping area. Path planning subsystem will provide both global path planning as well as local path planning. Global planning will include high level exploration path for both UAV, AGV in such way that they jointly achieve high coverage while exploration (Frontier based exploration Algorithm). Location planner involves taking cues from global planner and point cloud map from mapping subsystem to produce path that detours obstacles location to avoid any collision. (RRT Algorithm [10]) Path planning subsystem will provide its output of desired trajectory as way points to Navigation Control subsystem.

Fallback:

A low-level control fail safe will be included in AGV that uses 2D LiDAR and overwrites path planning output if that control can potentially cause collision.

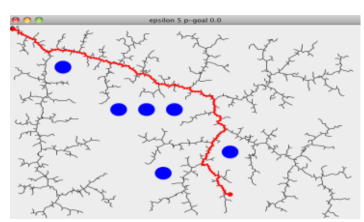

Figure 8 RRT for path planning

8.2.3. Navigation Control Subsystem:

Based on mode of operation, Navigation Control Subsystem takes input from Path Planning Subsystem or Fire Localization Subsystem and will create control signals to go to desired waypoint or follow desired trajectory. These control signals would be generated in closed loop fashion. Navigation controller would take localization data from SLAM Subsystem and will merge it with GPU and IMU data to get accurate localization (feedback) to generate control signals that corrects deviation from desired trajectory. Navigation control subsystem will communicate directly with low level motor controllers.

Fallback:

Proper execution of this subsystem is essential for safety as well as completion of the task. Any error/anomaly observed during execution of Navigation Control Subsystem should lead to emergency landing/stopping to avoid any damage.

Figure 9 Pixhawk Jetson Interface

8.2.4. Collaboration Subsystem:

Collaboration Subsystem will provide communication link to transfer vital information such as active fire location, other system’s resources status. This information can be extremely useful for high level decision making of what to do. (Explore, Extinguish or Return) Communication link would enable our system to fight the fire in collaborative manner. Using this communication, UAV/AGV can ask for help in extinguishing fire if fire is too large. Collaboration Subsystem will create a shared database containing active fire location and would assign tasks based on priority queue.

Fallback:

Collaboration effects efficiency of fighting fire but our system by design (UAV/AGV both has fire detection sensor as well as extinguishing material) can handle lack of communication or collaboration. In case of communication failure, each system will act independently would still try to search and extinguish fire.

Figure 10 UAV-AGV Wireless communication for Collaboration

8.2.5. Fire Detection and Localization Subsystem:

As the name suggests, goal of this subsystem is to detect fire from the thermal image. To accurately create bounding box around fire in thermal image, image classifier needs to be trained. Thermal Image Dataset needs to be procured and darknet needs to be trained. Once classifier is available, it needs to be implemented in single board computer of UAV and AGV. Once integrated, Fire detection Subsystem keeps looking for fire and when it detects fire and it looks at output of SLAM Subsystem, to approximate location of fire in global map. Finally, Fire Detection and Location Subsystem shares this fire location to Extinguishing Fire Subsystem as well as Collaboration system so that other system can also become aware of that fire.

Fallback:

In case of failure in procuring thermal image dataset of fire, simple temperature based heuristic approach on thermal images can be employed to detect fire.

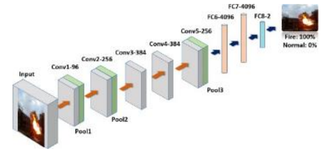

Figure 11 Convolution Neural Network for fire detection

8.2.6. Fire Extinguishing Subsystem:

Once extinguishing task is assigned after detecting the fire. Fire extinguishing subsystem becomes active. In first step, it orients UAV/AGV with respect to fire by providing waypoints to Navigation Control Subsystem using Visual Servoing method. E.g. if system is far away from the fire, waypoints that leads the system to fire are provided. In second step, it provides control signals to microcontroller which in turn activates the actuators of extinguishing deploying mechanism to point it directly at fire. Finally, it gives signal to deploy the material and provides alert to Navigation Control Subsystem so that it can handle recoil and payload change. It keeps monitoring the fire status using thermal image data to stop deployment if fire is extinguished. It also keeps track of how much extinguishing material is available using load sensor.

Fallback:

If system is unable to extinguish fire. It can still provide vital map data and location of active fire in it to the human firefighters that in turn can extinguish the fire using the generated resources.

Figure 12 Fire extinguisher

9. Project Management

9.1. Work plan and tasks

A 3-level Work Breakdown Structure is created based on the defined subsystems of the project where subsystem contains multiple tasks to meet the requirements of the final firefighting system. Integration, testing and project management branches are also included since they would also require significant work.

Table 10 Hardware task breakdown

Hardware Design	1. Finalize base part list for UAV and sensors for both UAV, AGV 2. Design mechanism to deploy extinguishing material at target 3. Design arm for UAV that attaches extinguishing material deploying mechanism with UAV (with or without DoF) 4. Design arm for AGV that attaches extinguishing material deploying mechanism with AGV (with n-DoF) 5. Design/Choose UAV Power Distribution Board 6. Design/Choose AGV Power Distribution Board
Hardware Integration	1. Procurement of Parts 2. Fabricating/Procuring Power Distribution Board 3. Assembling base UAV system with help of AirLab, CMU 4. Integrate sensors with UAV 5. Integrate sensors with AGV 6. Building deploying mechanism 7. Building UAV arm with microcontroller integrated

	8. Integrating deploying mechanism with arm and arm with UAV 9. Integrating deploying mechanism with arm and arm with AGV
Hardware Testing	1. Kill switch testing 2. UAV manual flight test 3. AGV manual drive test 4. Stereo Camera testing with Nvidia Jetson 5. Thermal Camera testing with Nvidia Jetson 6. 2D-LiDAR testing with Nvidia Jetson 7. Pixhawk testing with Nvidia Jetson 8. Microcontroller testing with Nvidia Jetson 9. Testing UAV arm in deploying material at target 10. Testing AGV arm in deploying material at target 11. Testing wireless communication

Table 11 Software task breakdown

Simultaneous Localization and Mapping Subsystem	1. Installing libraries dependencies in Nvidia Jetson 2. Installing and integrating ORB-SLAM2 in ROS pipeline 3. Integrating odometry data coming from GPU and IMU sensors to improve SLAM result
Path Planning Subsystem	1. Implementing global primitive paths that takes system to area of interest (building) from paths that achieve high coverage 2. Integrating implemented frontier-based exploration planner in ROS pipeline (where to search after reaching area of interest) 3. Integrating implemented RRT based local planner that takes 3D maps from SLAM and gives collision free paths
Navigation Control Subsystem	1. Establish MAVROS link from Nvidia Jetson to Pixhawk 2. Provide low level desired waypoints/trajectory to Pixhawk 3. Provide IMU, GPU data from pixhawk to SLAM subsystem and feed updated localization from SLAM subsystem to Pixhawk 4. Create collision avoidance failsafe
Collaboration Subsystem	1. Establishing Wi-Fi communication link between UAV and AGV 2. Creating shared database that can store active fire location, system location and available resources 3. Design and Implement optimum task assignment algorithm
Fire detection and Localization Subsystem	1. Collect dataset of thermal camera 2. Training Darknet classifier (offline) 3. Running forward pass of Darknet in Nvidia Jetson 4. Implement module to Localize fire based on current system location and 3D map
Fire Extinguishing Subsystem	1. Implement/Integrate basic visual servoing method to orient system close to fire 2. Establish serial communication from Jetson to Microcontroller 3. Implement controller to orient AGV arm to point at fire 4. Implement controller to orient UAV arm to point at fire 5. Implement signal to deploy material while monitoring fire

Software Integration	1. Getting familiar with ROS (bonding and action server) 2. Implement code that initializes various ROS nodes 3. Implement main function that does high level decision making 4. Provide any missing link between various subsystem

Table 12 System Level Integration and Testing task breakdown

System Level Integration and Testing	1. Integrating & testing Fire detection subsystem and Extinguishing Fire system 2. Integrating & testing Path Planning subsystem and Navigation Control subsystem 3. Integrating & testing Navigation Control subsystem and Fire Localization subsystem 4. Integrating & testing Collaboration subsystem with Fire Detection, Navigation Control & Fire Extinguishing Subsystem 5. Entire system testing

Table 13 Project Management task breakdown

Project Management	1. Document design and implementation 2. Manage budget 3. Arrange meetings 4. Address unforeseen challenges 5. Obtain various permissions

9.2. Schedule

Below is Gantt chart for Spring Semester where various tasks are scheduled based on their dependencies and milestones.

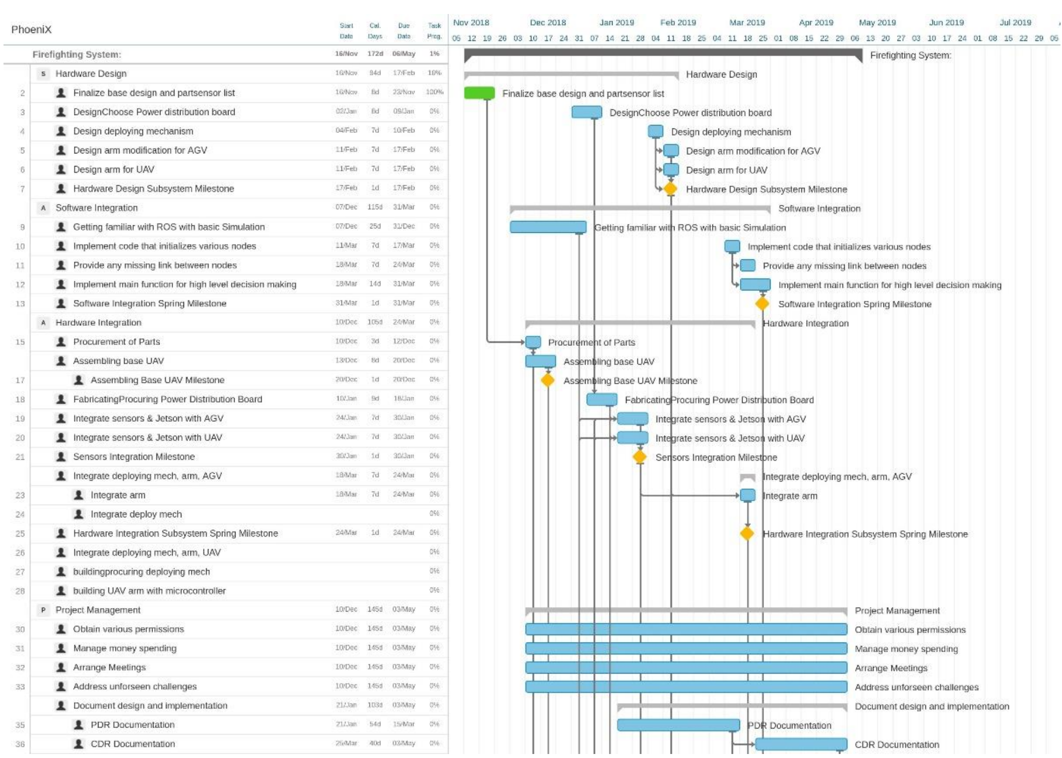

Figure 13 Spring Gantt Chart- I

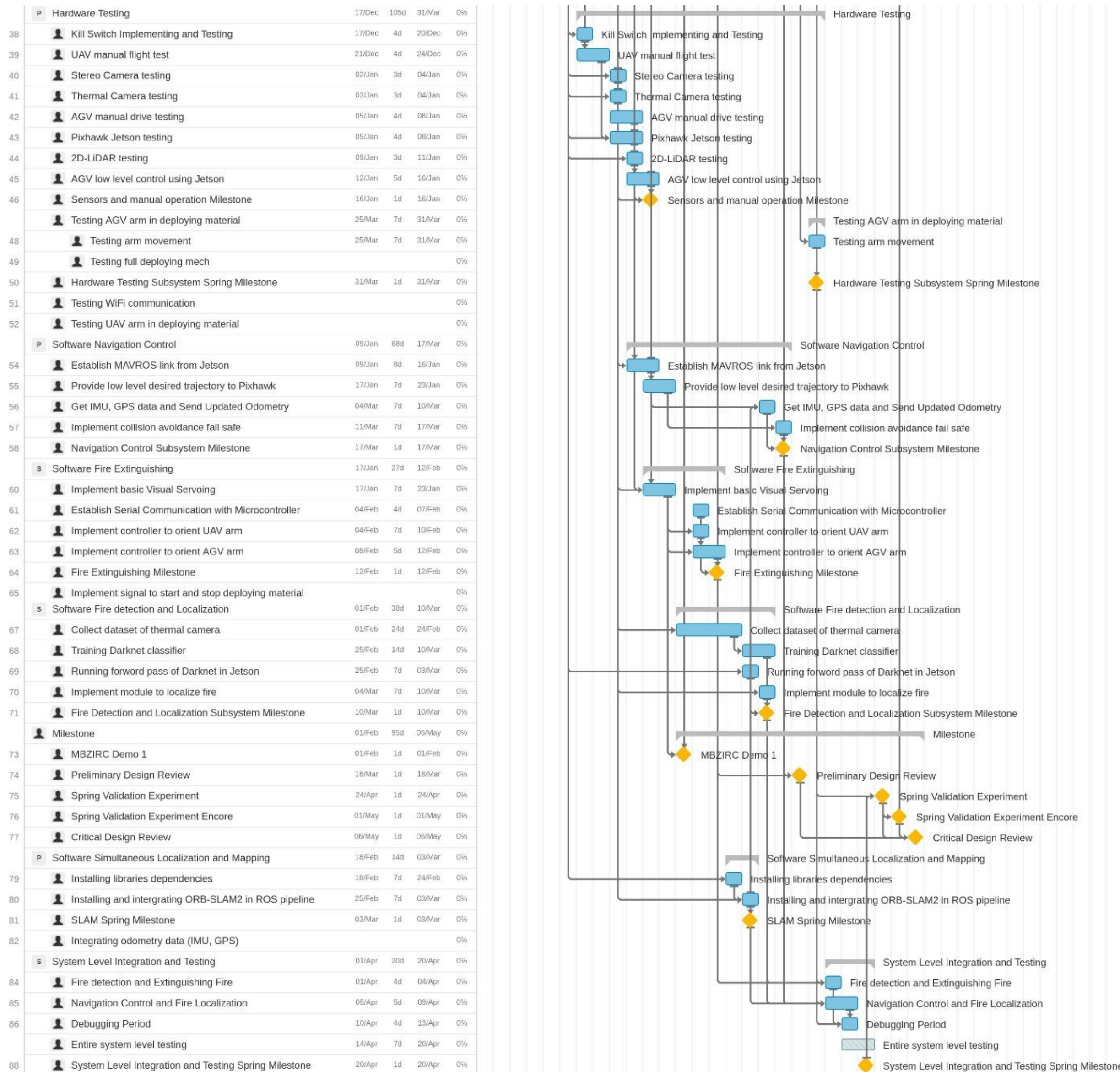

Figure 14 Spring Gantt- II

Below are the milestones for spring and fall semesters.

Table 14 Spring 2019 Milestones

Milestone	Date
Assembling base UAV	December 20
Sensors and manual operation testing	January 16
Sensors Integration	January 30
MBZIRC Demo 1 [11]	**February 1**
Fire Extinguishing Subsystem	February 12

Hardware Design Subsystem	February 17
SLAM Subsystem (Spring)	March 3
Fire Detection and Localization Subsystem	March 10
Navigation Control Subsystem	March 17
Preliminary Design Review	**March 18**
Hardware Integration Subsystem (Spring)	March 24
Hardware Testing Subsystem (Spring)	March 31
Software Integration Subsystem	March 31
System Level Integration and Testing (Spring)	April 20
Spring Validation Experiment	April 24
Spring Validation Experiment Encore	May 1
Critical Design Review	May 6

Table 15 Fall 2019 Milestones

Milestone	Date
SLAM Subsystem (Fall)	September 8
Path Planning Subsystem	September 15
MBZIRC Demo 2	September 28
Collaboration Subsystem	October 15
Mid Sem Report (MRSD)	October 22
Hardware Integration (Fall)	November 7
Hardware Testing (Fall)	November 14
System Level Integration and Testing (Fall)	November 28
Fall Validation Experiment	December 1

9.3. System Validation Experiments

The spring validation experiment will showcase some of our rudimentary hardware and software components. As we begin the project in spring 2019, we will be able to demonstrate the software which might not be fully integrated with our hardware and some of it will be in the form of individual component testing or simulation. The fall validation experiment will demonstrate the final product conforming to all the requirements promised by the team.

9.3.1. Spring Validation Experiment

Table 16 SVE: Test 1 & Test 2

Test 1	Description: Fire detection classifier and UAV controls test
Location: NSH Level B / "TBD"	Equipment: Phoenix UAV, heated plate of minimum size 0.5 m x 0.5 m.
Test Process: 1. PhoeniX UAV will take off from a base station of 1m x 1m. 2. The UAV will follow a predefined trajectory till it reaches within 2.5 m radius of the location of fire (heated plate). 3. The UAV will then detect the simulated fire from a minimum of 2 m distance (line of sight): Forms basis for verification criteria 1. 4. UAV will orient itself in the direction in front of fire (heated plate) by maintaining minimum 1 m distance. 5. UAV will point a laser pointer on the target fire location to simulate the extinguishing mechanism: Forms basis for verification criteria 2. 6. The UAV will now follow another predefined trajectory back to the base station. 7. The UAV will now land within 5 m radius of the base station: Forms basis for verification criteria 3.	Verification Criteria 1. UAV successfully detects fire from a minimum of 2 m distance. 2. UAV points a laser pointer within the heated plate of size 0.5 m x 0.5 m. 3. UAV successfully lands within 5 m radius of the base station center.

Test 2	Description: PhoeniX UAV payload test
Location: NSH Level B / "TBD"	Equipment: Phoenix UAV, dead weight of 1.5 KG.
Test Process: 1. A team member will attach a 1.5 KG payload to the UAV. 2. The UAV will take off to height of 1 m with the payload attached: Forms basis for verification criteria 1. 3. The UAV will move 2 m forward: Forms basis for verification criteria 2. 4. The UAV will hover for 1 minute at this location. 5. The UAV will land on the same location: Forms basis for verification criteria 3.	Verification Criteria 1. UAV successfully lifts the payload. 2. UAV performs the desired movements (hover height, forward waypoint of 2m and final hover time). 3. UAV successfully lands with the payload attached.

Table 17 SVE, Test 3

Test 3	Description: PhoeniX AGV test
Location: NSH Level B / "TBD"	Equipment: AGV, heated plate of minimum size 0.5 m x 0.5 m.
Test Process: 1. PhoeniX AGV will drive off from a base station of 1.2 m x 1.2 m. 2. The AGV will follow a predefined trajectory till it reaches within 2.5 m radius of the location of fire (heated plate). 3. An obstacle will be placed in front of the AGV at some instants and it will stop as soon as it detects: Forms basis for verification criteria 2. 4. AGV will detect fire (heated plate) from minimum 2 m distance (line of sight): Forms basis for verification criteria 1. 5. AGV will orient itself in front of fire (heated plate) by maintaining minimum 1 m distance. 6. AGV will point a laser pointer on the target fire location to simulate the extinguishing mechanism: Forms basis for verification criteria 3. 7. The AGV will now follow another predefined trajectory back to the base station. 8. The AGV will now return back to base and park in within 1 m radius of the base station: Forms basis for verification criteria 4.	Verification Criteria 1. AGV successfully detects fire from a minimum of 2 m distance. 2. AGV successfully stops as soon as it detects obstacle and does not crash into it. 3. AGV points a laser pointer within the heated plate of size 0.5 m x 0.5 m. 4. AGV successfully parks in within 1 m radius of the base station center.

9.3.2 Fall Validation Experiment

Table 18 FVE, Test 1

Objective	Demonstrate that the PhoeniX firefighting system is capable of collaboratively extinguishing fire using UAV and AGV in a building or an equivalent simulated environment.
Sub-systems	● Global Trajectory Planner (Navigation) ● Local Planner with Obstacle Avoidance ● Visual Servoing ● Mapping ● Manipulator control ● Fire extinguishing
Location	Dummy MBZIRC test site / "TBD"
Equipment	● PhoeniX UAV, UGV both with extinguishing material ● Heated Plates ● Kill Switch ● Safety nets (if test location is indoors)

Table 19 FVE testing procedure, validation criteria

Testing Procedure	Validation Criteria
1. Operator will give the GPS location of the building in the form of an input to the PhoeniX firefighting system. 2. UAV and AGV will takeoff and drive off towards the known location of a structure/building containing potential fire spots. 3. During the movement of the systems they will create a real time map: Forms basis for verification criteria 3. 4. The systems shall continuously avoid obstacles in their way towards the structure by rerouting around the obstacles like other UAVs, AGV and the walls of the structure. 5. Systems will enter inside the building through the openings like windows and doors to detect fire. 6. When the systems detect fire locations, they will add its location in the shared database: Forms basis for verification criteria 5. 7. The same system or some other system shall then navigate to this fire location to extinguish fire. 8. The system will deploy extinguishing material on the simulated fire spot to simulate the extinguishing task: Forms basis for verification criteria 2. 9. Once all the fire locations have been extinguished the system shall come out of the building.	1. Maps 60% area of the actual test site ("TBD") within 10 minutes of operation. 2. Deposit 40% deployed extinguishing material. on the target area of minimum 0.5 m x 0.5 m 3. System accumulates less than 5m drift for every 100m distance travelled. 4. Localizes fire with less than 1 m error. 5. System is able to communicate with each other within a radius of 25 m. 6. System carried 1 KG of extinguishing material each.

9.4. Team Member Responsibilities

Following table contains the preliminary team responsibility matrix, where numbers 1 and 2 represent primary and secondary responsibilities respectively.

Table 20 Team Member Responsibilities

Responsibility	Parv	Akshit	Shubham	Steve
Hardware Design	2		1	
Hardware Integration		1	2	
Hardware Testing	1		2	
Navigation	1	2		
SLAM	1	2		
Robots Collaboration		2		1
Path Planning	2		1	
Extinguishing Fire			1	2
Fire Detection		2		1
Software Integration		1		2
Software Testing		2		1
Project Management	1		2	

9.5. Parts List and Budget

Table 21 Part List and cost

Subsystem	Component	Details	Unit cost	Qty	Total cost
Electronics	Miscellaneous Connectors	XT90 connector, Battery Straps, AS150/TX150 Connectors	$16.88	1	$16.88
	Batteries	GenAce “Tattu” 6S 16000mAh 15C	$651.00	1	$651.00
	Battery Monitor	Li-Po monitor	$10.99	1	$10.99
	Jetson TX2 Active heat sink	Measuring force on probe	$50.00	1	$50.00
	Nvidia Jetson TX2	Single Board Computer	$350.00	1	$350.00
	Carrier Board	Peripheral Board	$174.00	1	$174.00
	TeensyDuino Development Board	Slave microcontroller for manipulator arm	$85.00	1	$85.00
	FTDI cable	To handle communication between SBC and Flight Controller	$17.99	1	$17.99
	USB Hub	To power all the Vision System externally	$12.99	1	$12.99
UAV Platform	Hexacopter Frame	Tarot X6 (TL6X001)	$276.00	8	$276.00
	Connectors	Bullets connectors for ESCs,	$8.00	1	$8.00
	Brushless Motor	KDE4215XF-465	$148.8	8	$1191.00
	Propellers	14inch (P14 x 4.8 Prop)	$27	6	$162
	Shock-absorbing Foam Protective cover	For Landing Gear	$1.79	1	$1.79
	Electronics Speed Controller	KDEXF-UAS55HVC 55A+HV	$118.875	8	$951
	Flight Controller	Pixhawk 2	$330	1	$330
	Battery Eliminator	UBEC	$117.90	1	$117.90
Sensors	Stereo Camera	ZED	$385.00	1	$385.00
	FLIR Sensor	Thermal + RGB Camera	$379.00	2	$758.00
Wireless-Communication	Serial Wireless datalink (900MHz)	3DR Radio V2	$389.50	1	$42.90
	Wi-Fi Module	Dual Band N600 Wi-Fi Adapter	$30	2	$ 60
Total					$5652.14*

(* Sponsor is willing to provide the additional funding for the project)

9.6. Risk Management

The team has identified multiple risks related to design, schedule and resources. L and C stand for likelihood and consequence, on a scale from 1 (least) to 5 (most).

Table 22 Risk Management

Risk Id	Risk	Category	L	C	Mitigation Strategy	Risk Owner
R.1	Lack of availability of Test Site conforming MBZIRC Specifications	Schedule, Technical	4	4	1. Talk to Sebastian to at least create a dummy test site of smaller scale by 27 August 2019.	Akshit
R.2	No knowledge on the actual Fire simulation to be used in MBZIRC	Technical, Schedule	3	5	1. Get the sponsors (Oliver Kroemer) speak to the MBZ Committee (Lakmal) and get a resolution by December 21, 2018. 2. Procure and use Induction cooktops to simulate fire or some heating plate till the 'TBD' status is resolved by 15 February 2019.	Shubham
R.3	Lack of Data Set for training Fire Detection module	Technical, Schedule	3	4	1. Create the data set by capturing some videos using the UAV and train YOLO on that data starting January 15, 2019 to January 21, 2019 2. If the dataset is not enough then probably use some alternative approach like image segmentation based on color	Steve
R.4	Extra effort on repairing/maintaining the UAV and AGV	Schedule, Cost	5	2	Maintain a contingency reserve especially for the UAV like motors, propellers and ESC	Akshit
R.5	UAV built using parts list from AirLab does not have enough payload capacity	Schedule, Cost	3	4	1. Order new parts for the UAV, run calculations thoroughly for the new motors, prop and frame latest by January 15, 2019. 2. Use off the shelf any UAV from AirLab if possible, by asking Sebastian (or use some reserve parts used for the DARPA SubT challenge at AirLab).	Steve
R.6	Operator Safety from UAV with carbon fiber props	Personal	4	5	Use nets for the testing area even if the prop guards or the kill switch fail and maintain an operating procedure involving all preflight checks to ensure safety of personnel and the UAV	Parv

R.7	Underestimating UGV manipulator arm design and fabrication task	Schedule, Cost	2	3	1. Run simulations before buying parts, complete the task with the help of Oliver Kroemer and Anish at Air lab 2. Plan for the task with a buffer of 15 days in the winter break so that time is utilized efficiently 3. Buy off the shelf arm (latest by 15 March 2019) if budget permits or ask sponsor for any manipulator in Oliver's lab which can meet the requirements	Parv
R.8	ORB-SLAM on the UAV does not perform well or the implementation takes time	Schedule, Technical	1	4	1. Run tests in winter break and try to find solution if time permits in the break 2. Totally remove real time mapping from the requirements and use only GPS to localize and Visual Servoing on the UAV if we cannot solve issues by August 27, 2019.	Parv
R.9	Team Members lack of experience in certain fields	Schedule	3	2	1. Compensate for the learning time by consulting professors, sponsors at CMU 2. Take guidance from peers who have some idea in that field	Shubham
R.10	Data/Code Corruption	Technical	2	4	Always take a copy of the code/data, or distribute code on the cloud and share among team members	Steve

		R.2	R.6	
R.8	R.10	R.3, R.5	R.1	
	R.7			
		R.9		R.4

Likelihood v/s Consequence Matrix

Red = Critical
Yellow = Moderately Critical
Green = Less Critical

10. Reference

[1] National Fire Protection Association (https://nfpa.org)

[2] Fire Sprinkler System https://en.wikipedia.org/wiki/Fire_sprinkler

[3] Juan I. Giribet ; Ricardo S. Sanchez-Pena ; Alejandro S. Ghersin: Analysis and design of a tilted rotor Hexacopter for fault tolerance

[4] Wireless Communication https://www.electronicproducts.com/Computer_Peripherals/Communication_Peripherals/Bluetooth_vs_Wi_Fi_vs_ZigBee.aspx

[5] Titus Cieslewski, Siddharth Choudhary and Davide Scaramuzza: Data-Efficient Decentralized Visual SLAM

[6] Raul Mur-Artal and Juan D. Tard: ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras.

[7] Ji Zhang and Sanjiv Singh: Visual-lidar Odometry and Mapping: Low-drift, Robust, and Fast

[8] Ji Zhang and Sanjiv Singh: Visual–Inertial Combined Odometry System for Aerials Vehicles.

[9] YOLO object classifier: https://pjreddie.com/darknet/yolo/

[10] Steven M LaValle: Rapidly-Exploring Random Tree: A new tool for path planning

[11] MBZ-International Robotics Competition: https://www.mbzirc.com/

[12] Xiao-Jing Shen, Jun-Min Pan: A simple adaptive control for visual servoing

11. Appendix

11.1. Objective Tree

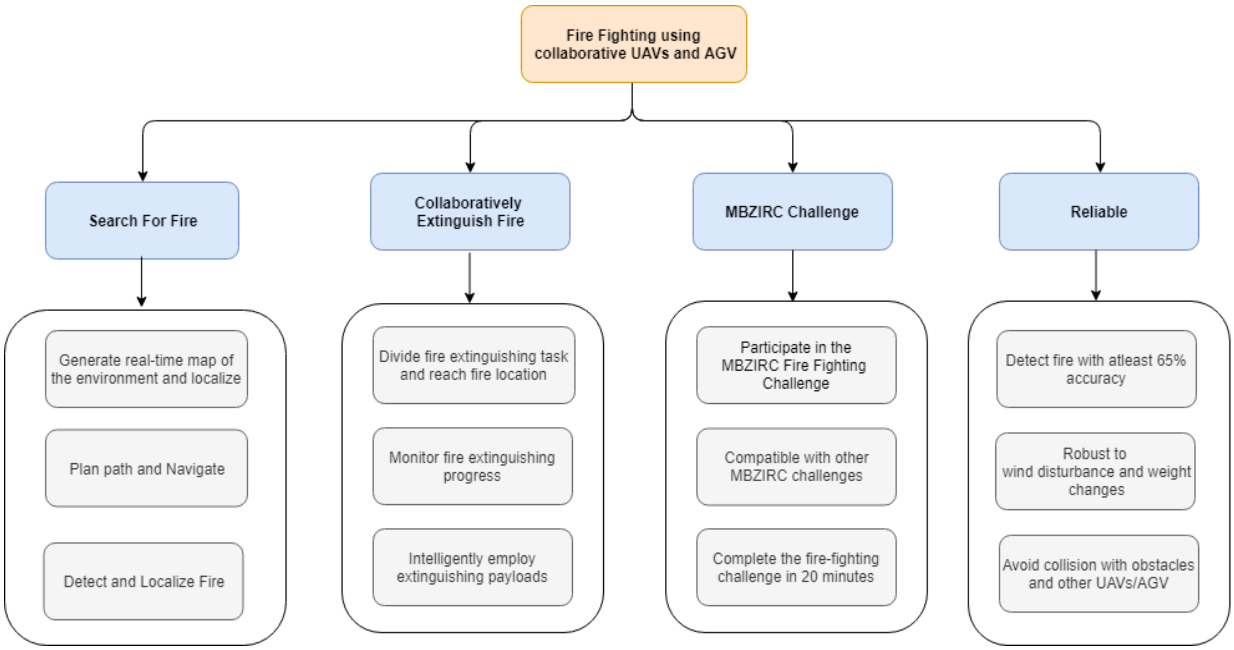

Figure 15 Objective Tree

11.2. Component level trade Study

Table 23 Single Board Computer Trade Study

Criteria	Weight (in %)	Odroid XU 4	Snapdragon Flight	Nvidia Jetson	Intel Aero
Cost	5	1.25	0.25	0.25	1
Power Consumption (M.R.3)	10	2.5	2	1.5	2
Size (M.N.2)	5	1.25	1	0.5	1.25
Peripherals (F.R.7)	5	1.25	1	2	1.25
Storage (M.R.4)	5	1.25	1.25	1.25	1.25
RAM (M.R.1)	10	1.5	2	2.5	1.5
Speed (M.R.1)	10	1.5	2	2.5	1.5
CPU (M.R.1)	10	1.5	2	2.5	1.5
GPU (M.R.1)	20	2.5	4	5	2
Community Support	20	3.75	1.25	4.5	1.25
Total	100	2.2	2.025	**3**	1.5375