RoboCon SRS

Project PavneetKaurMukar_CSCI568Project (Requirements Management)

Printed by Pavneet Kaur Mukar

May 8, 2017, 7:58:22 PM EDT

Configuration Pavneet_Main_Project Initial Stream

Configuration Type Local Configuration

Component Pavneet_Main_Project

Table of Contents

1 Introduction

This document describes the high level characteristics and function of a proposed robotic convoy system called **RoboCon.**

1.1 Purpose

The purpose of this document is record the decisions agreed by the RoboCon stakeholders. It will further be served as a basis for detailed system requirements specification [SRS] and documentation.

1.2 Scope

RoboCon consists of a number of robots that form a convoy and follow a leader robot to a specified destination. Usually the number of robots used in the convoy are minimum four. It consists of mainly two parts:

1. One robot called as the *leader robot* is provided with the path to follow in the form of series of spatial coordinates called *waypoints.*
2. All the rest of the robots are *follower robots,* which used on-board sensors to track the robot immediately ahead of it in the convoy and then follow them to the destination.

This autonomous convoy has applications in transportation, inventory management, automated farming, house cleaning and many more.

Eg. A set of trucks could be placed in a convoy to transport goods. A human driver could drive the leader truck and the rest of them could be automated.

Another example could be in house cleaning. The leader can be given destination area for cleaning and then the rest of the robots can follow. The leader could clean, follower robots can mop, cleanse etc the area.

1.3 Definitions

Guide : The RCU tracked and followed by a follower. A guide may be the convoy leader or a follower. All followers have exactly one guide.

Follower : An RCU that does not know the waypoints of the route; uses on-board sensors to follow another RCU immediately preceding in the convoy.

Leader : The RCU that is provided with the route waypoints. The leader proceeds first in the convoy.

Tail : The RCU immediately behind a given RCU in the convoy. An RCU in the guide of its tail.

Waypoint : A set of spatial coordinates indicating a position that the convoy must pass through on its way to the destination of the route.

Kanban is a process management system for tracking the status of work items. They are captured on a Kanban “board”. This board includes set of process steps (“bins”). All the work items are moved between steps as their status changes. The process steps, task types, work items, user roles, etc. are customizable.

Thus overall recording of the system workflow can be managed easily.

1.4 Acronyms

This section provides definitions and lists all used acronyms as well as references

1.4.1 GPS : Global Positioning System
1.4.2 ACU : Administrative Control Unit
1.4.3 GUI : Graphical User Interface
1.4.4 IR : Infrared
1.4.5 LAN : Local Area Network
1.4.6 OS : Operating System
1.4.7 OTS : off-the-shelf
1.4.8 PC : Personal Computer
1.4.9 RCU : RoboCon Unit
1.4.10 RS: RoboCon System
1.4.11 Algo: algorithm

1.5 References

1. IEEE Software Engineering Standards Committee, “IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications”
2. http://www.cse.chalmers.se/~feldt/courses/reqeng/examples/srs_example_2010_group2.pdf
3. https://capstone.cs.ucsb.edu/team_docs_14/SRS/SRS_Let_My_People_Code

[format for SRS is referenced]

2 Overall Description

2.1 Product perspective

RoboCon will provide a demonstration and experimentation platform for CS568 Systems image processing and artificial intelligence algorithms. It will:

- Allow CS568 Systems to conduct internal evaluation of alternative algorithms in different operational scenarios.
- Provide a platform for CS568 Systems to showcase the performance of new algorithms to customers.

It consists of two types of robots, using which destination can be achieved easily:

1. Leader Robot
2. Follower Robot

To achieve the success of the product, there are mainly 3 operational modes[discussed later], which are as follows:

1. Convoy Formation
2. Movement
3. Recharge

The real life implementation is in transport system, surgerical system, inventory management, cleanliness system and many more.

2.2 Product Usage Scenario

RoboCon is easy in terms of implementation and can be considered useful if implemented effectively. It can be used in various different aspects like

- Transportation
- Inventory Management
- Automated farming and many more

*The two many high-level **usage scenarios** are*:

1. **Experimentation-** It will be used by CS568 Systems algorithm developers to experiment with different image processing and artificial intelligence algorithms. The algorithm under test will be implemented in a component that plugs into the RoboCon software. The RS will be configured for a particular activity like the number of robots being used, the start and destination locations etc.As the experiment takes place the RS will record the performance metrics like batter usage, total travel time, number of messages exchanged, number of robots used etc. This data can then be used to evaluate the algorithm's performance.

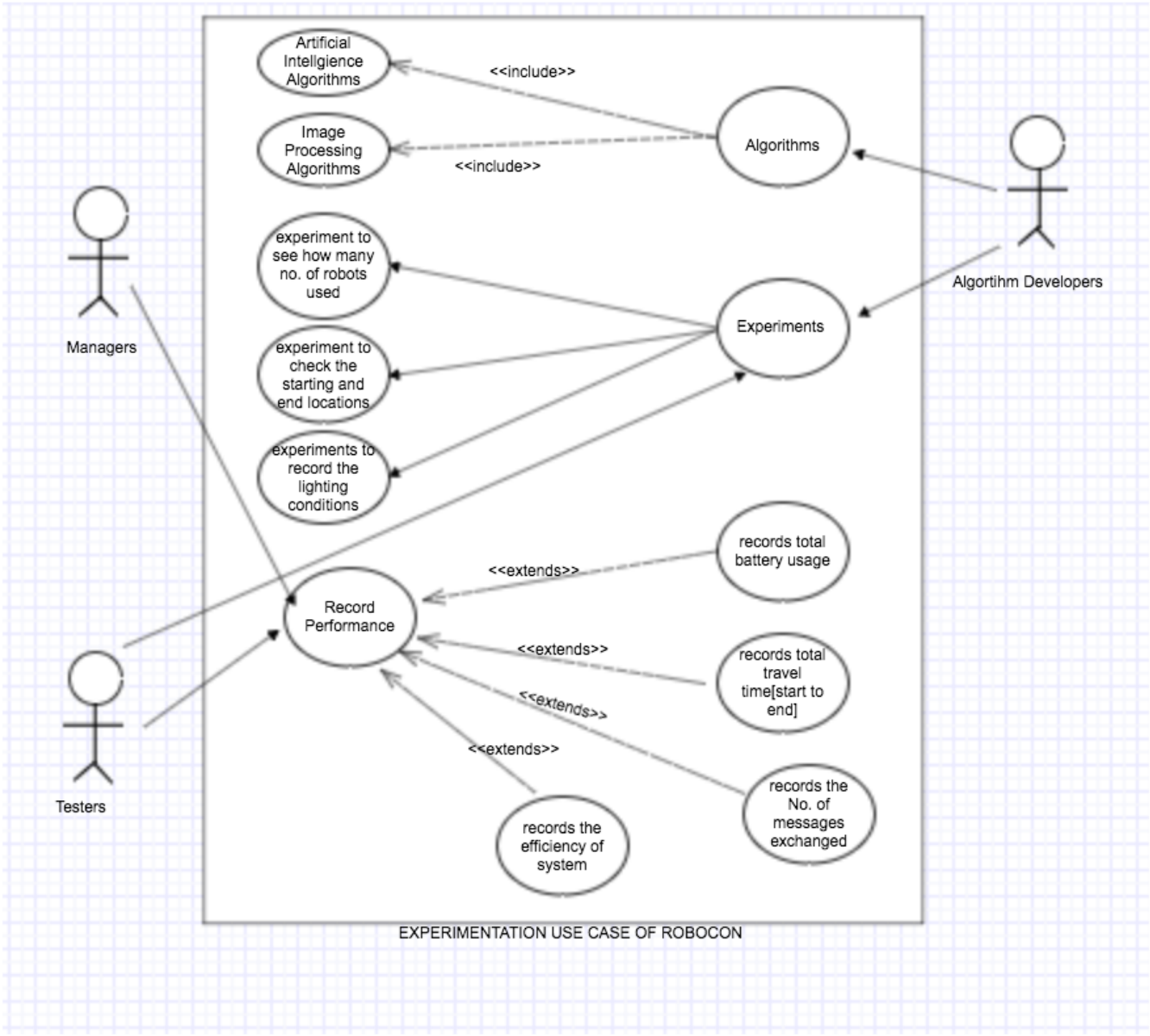
2. **Demonstration-** It will be used to demonstrate the algorithms to external customers.In this, the best performing algo is plugged in the RoboCon Software and a comparison is made with other competitors algo and explains why CS568 System algo is much better than them.

For example: It might demonstrate how CS568 System algo travels to it destination with minimum numbers of messages being exchanged with the follower robots and so on.

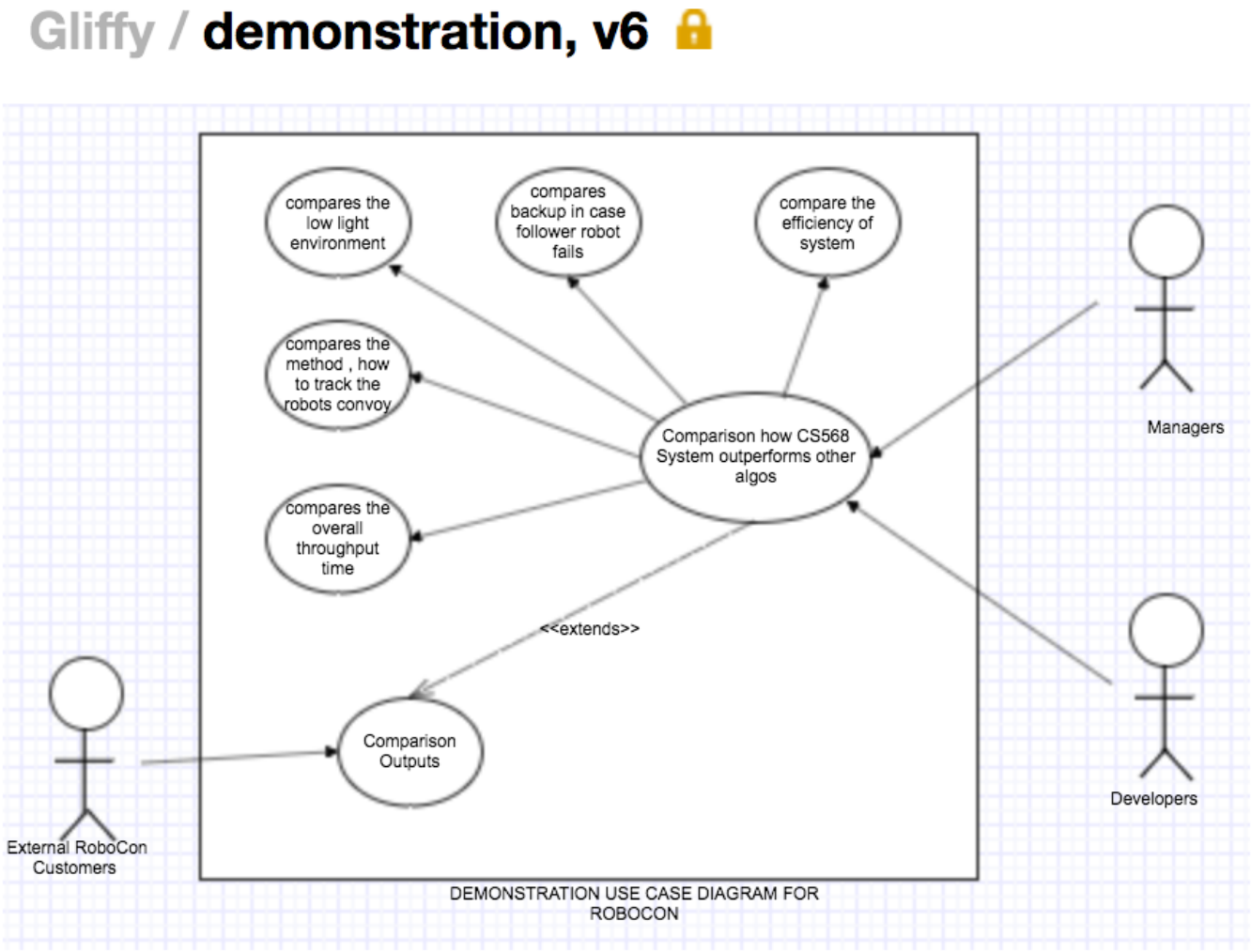
2.3 Use Cases

2.3.1 *Use Case for Experimentation*

Gliffy / **experiment, v6**



2.3.2 *Use Case for Demonstration*



2.4 Design and Implementation Constraints

In the usage of the system , there are many different constraints. They are as follows:

1. While designing the system, keep in mind the ACU is build properly and all the controllers as well as other RCUs can connect with ACU effectively.
2. Wifi also poses as a constraint when GPS is used to locate the position of the robots.
3. The IR transmitter should be linked with the receiver. They work in sync and should be paired otherwise individually they won't function effectively.
4. If camera is to be used, effective video functionality should be inculcated.
5. RCU being assembled should be done from off-the-shelf components.
6. Need to use Linux Operating System.
7. The cost effectiveness should be kept in mind and only then the number of robots

needed should be designed.

8. The time of delivery of the product should also be noted and implementation must be done according to the system. Best way to record this is by using Kanban tool.

2.5 Stakeholder Goals

There are various different stakeholders and thus each category has different goal. These are as follows:

1. **RoboCon Development Team:**
    a. Responsible for building the RoboCon System
    b. Main Goal: To complete the project on time and within budget successfully.
2. **CS568 Systems Algorithm Developers(RoboCon Users):**
    a. They are the primary users of the system
    b. Main Goal:
        i. Be easy to extend with new image processing and artificial intelligence algorithms
        ii. Be simple to configure for different experiments
        iii. Provide rich and meaningful metrics of algorithm performance
3. **CS568 Systems Management:**
    a. They plan and fund the overall development of RoboCon
    b. Main Goal:
        i. Low maintenance costs over the time of the system
        ii. System can be used to provide compelling demonstrations to customers of the value of RoboCon algorithms
4. **CS568 System Customers:**
    a. They view the demonstration that showcase the performance of the algorithms
    b. Main Goal: Gain the understanding of how the algorithms can be incorporated into their own products for optimum effectives for the customers

2.6 Stakeholder Requirements

There are different stakeholders in different projects and the main aim as well as requirements for these stakeholders vary. They are defined as follows:

- RoboCon Development Team Requirements:
    - As they are responsible for building the main system. They focus on completion of the system.
    - On- time completion of the system
    - The man hours input should be done according to the time constraints.

- The overall budget of the system should not exceed.
- The user requirements mentioned should be fulfilled.
- All the functional requirements should operate effectively.
- Easy deployment of the system should be there.
- All the information required to build the system must be captured before hand.
- The SRS should be complete, consistent, correct, unambigious, traceable, manageable, portable and many more.

· CS568 System algorithm developers Requirements (RoboCon users) :

- They are the primary users of the system. Main focus is the application part.
- The system should be easy to understand.
- The system should be easy to extend with different algorithms.
- The system should not only limit to image processing and artificial intelligence algorithms.
- The system should be configured easily.
- The system should provide rich and meaningful metrics of the algorithm preferred.
- The system should effectively tell which algorithm is more efficient.
- The system should be implemented easily such that if any changes need to be done, the user can do it easily without external help.

· CS568 System Management Requirements:

- They plan and fund the overall system. So their requirements are based mainly on money saving.
- The system should be implemented within the budget.
- The system should not need a huge amount of money in the maintenance phase.
- The system should be done with the minimum amount of funds.
- Extra functionalities should be implemented which crosses the budget.
- They need to separate funds for agile implementation as well.
- The system should provide compelling demonstrations to customers of value.
- The system should have a huge visual impact.

· CS568 System Customers Requirements:

- They view the demonstrations that showcase the performance of the system algorithms.
- The system should be easily incorporated into individual product for better value perspective to the customers.
- The system should be implemented on day to day basis, for maximum

utilization.

- The system should not be limited to one algorithm only, thus its application should cover a huge kin of aspects.
- The system should be able to retrieve all the investment made in the initial stages of the system.

3 Specific Requirements

3.1 System Descriptions

3.1.1 System Components

RoboCon consists of multiple robots called **RoboCon Units(RCUs)** and one **Administrative Control Unit(ACU)**.

1. Each RCU consists of a robotic hardware platform and a software controller:
    a. RCU hardware platform will be assembled from off-the-shelf(OTS) components.
    b. RCU Software controller will be custom built by CS568 Systems.
2. The ACU consists of a standard PC and an administrative application.

3.1.1.1 RCU Hardware Platform

Each RCU has :

- An iRobot Create mobile programmable robot. This robot has wheels and a motor which can turn and move the robot. It is responsible for the movement of the robots. It also has a front bumper sensor that detects when the robot ran into an obstacle.
- An eBox 3854 computing running Fedora Linux. This box contains all computer components like processors, memory, flash storage etc. It also has a wireless local area network (LAN) adapter.
- Video Camera
- Global Position System(GPS) receiver
- Infrared Receiver
- Infrared Transmitter

RoboCon assembled RCUs

Ebox- 3854

iRobot example of a vacuum cleaner

3.1.1.2 RCU Software Controller

Each RCU is controlled by a Controller Software component. It runs on eBox with Linux as it OS.

- This controller provides core logic which does the implementation of the convoy behaviour. It consists of the following:
  - Receives sensor data from RCU sensors
  - Analyzes sensor data to determine the actions to be taken by RCU like moving forward, backward, turning, switch on/off camera etc
  - Sends commands to RCU sensors and actuators to perform the actions
  - Sends messages to other Controller instances to coordinate convoy activities
  - Records the performance metrics
  - Transmits metrics to Admin Component
- Each controller communicates with the camera, GPS, WiFi, IR etc using the drivers.

3.1.1.3 Administrative Control Unit

- The ACU is a standard PC and an Admin software application.
- The admin provides a graphical user interface {GUI} for performing various management functions.

- The Admin can:
    - Allow users to specify experiment and demonstrative configurations
    - Upload configurations to Controllers
    - Receive performance data and display in charts and tables

3.1.2 Operation Modes

RoboCon has mainly **3 modes of operation**. They are as follows:

1. **Convoy Formation:** In this RCU are placed in staging area and assemble themselves into a convoy system. They have a leader, follower, guide etc.
2. **Movement:** In this RCUs travel along a route in a convoy formation to reach their destination place. At times obstacles are encountered. They are discussed in next section.
3. **Recharge:** In this the RCUs individually charge their batteries while the rest of the convoy is in halt stage.

3.1.2.1 Convoy Formation

- In this stage, all the RCUs are in the staging area.
- Each RCU will function as either a leader or a follower. Each follower trails directly behind either the leader or another follower. Thus the convoy is a single file.
- The RCU immediately in front of a follower is called **FOLLOWER'S GUIDE**. Each follower has only 1 guide.
- RCU immediately behind an RCU is called **RCU'S TAIL**. Each RCU have only 1 tail, except the last RCU having no tail.
- All the RCUs position in a single file. Each follower uses on-board sensors to locate its guide.
- When the follower has locates its guide, it sends a message to the leader indicating that it is ready to switch to movement mode.
- Then, when all the followers have indicates the leader that they are ready for the switch, the leader broadcasts the message to all the followers indicating that the convoy is switching to movement mode.

3.1.2.2 Movement

- In movement mode, the leader uses an on-board GPS device to find its location and determine a path to the next waypoint.
- When a waypoint is reached, the leader computes the path to the next waypoint.
- If the leader encounters an obstacle in its path, it invokes an obstacle-avoidance algorithm.
- This algorithm temporarily take the leader off the path to the next waypoint and navigate around the obstacle.

- Each follower does not know the location of the waypoints, and must instead track and follow its guide.
- Each follower has three different ways to track its guide:
    - it can track the guide using a camera
    - it can track the guide using an IR sensor
    - it can request the guide's location over a WiFi link, use the GPS to get its own location, and compute a path to the guide's location.

NOTE:

- If a follower is using camera following, and the area becomes too dark, the follower can automatically switch to GPS or IR following.
- GPS following does not work indoors, while IR following does not work at large distances.

3.1.2.3 Recharge

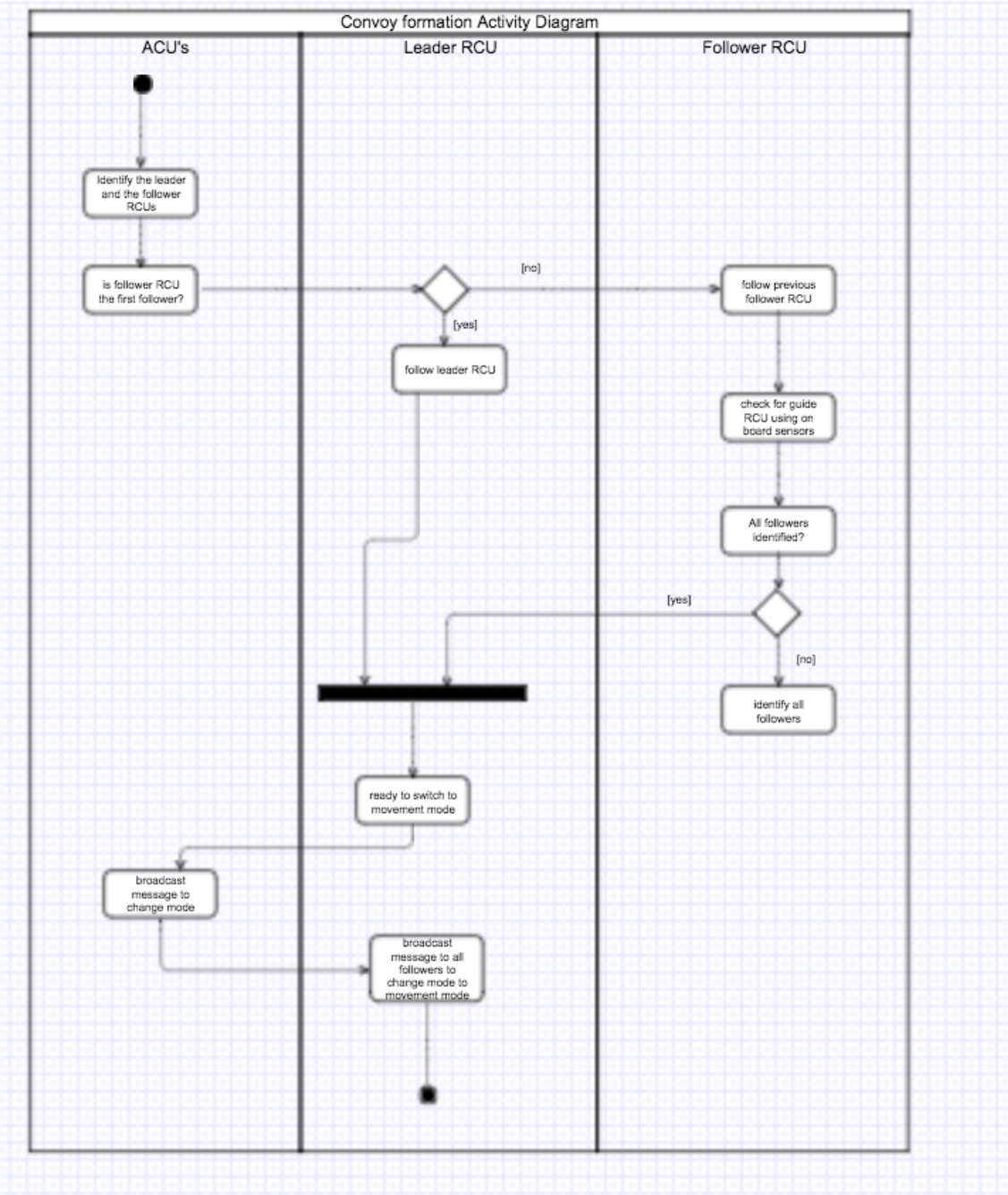
The convoy may encounter charging stations in the route. And thus the following steps should be done:

- If the convoy passes within 10 meters of a charging station, each RCU will check its remaining battery power.
- If any RCU has less than 20% battery remaining, it requests that all RCUs in the convoy stop temporarily. The convoy then switches to recharge mode.
- The RCUs that have low battery then visit the charging station one-by-one and recharge their batteries.
- When an RCU finishes charging, the RCU will assume its position in the convoy and notify all other RCUs that it is ready for the convoy to switch back to movement mode and continue moving.
- Once all RCUs with low battery have finished charging, the convoy switches back to movement mode.

3.2 Activity Diagrams

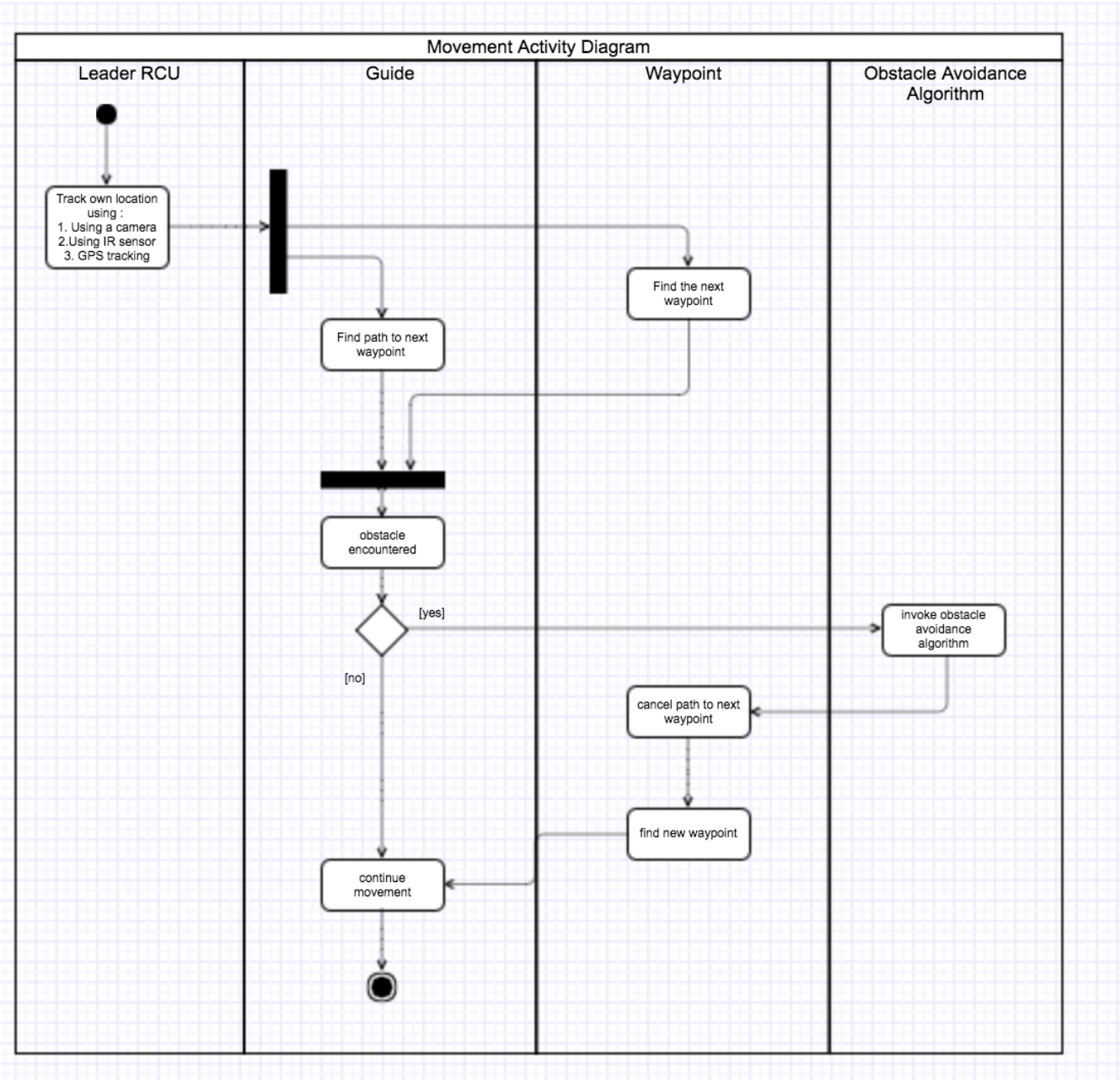
3.2.1 Convoy formation mode

Gliffy / **formation activity diagram, v1**



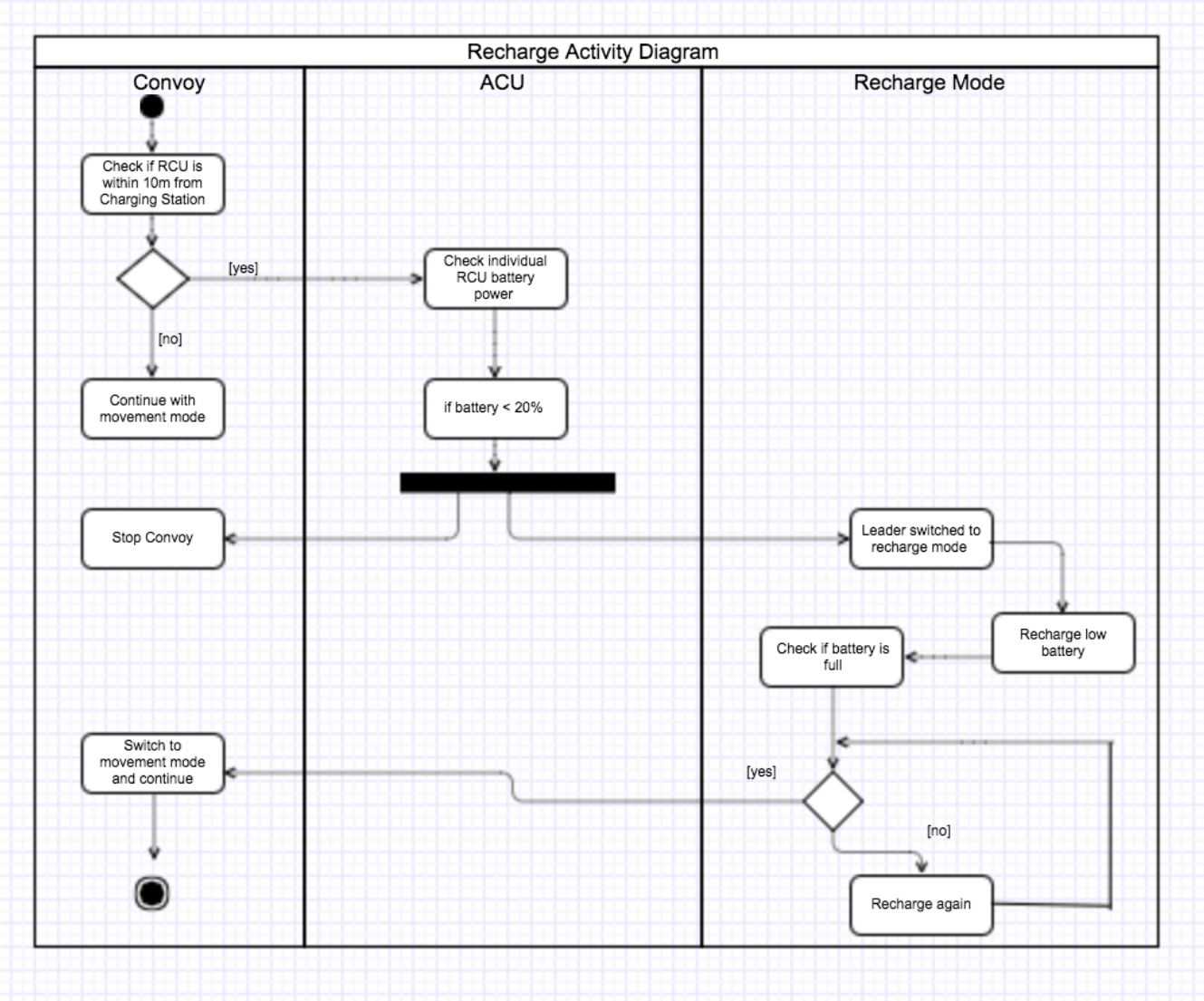
3.2.2 Movement mode

Gliffy / **movement activity diagram, v3**



3.2.3 Recharge mode

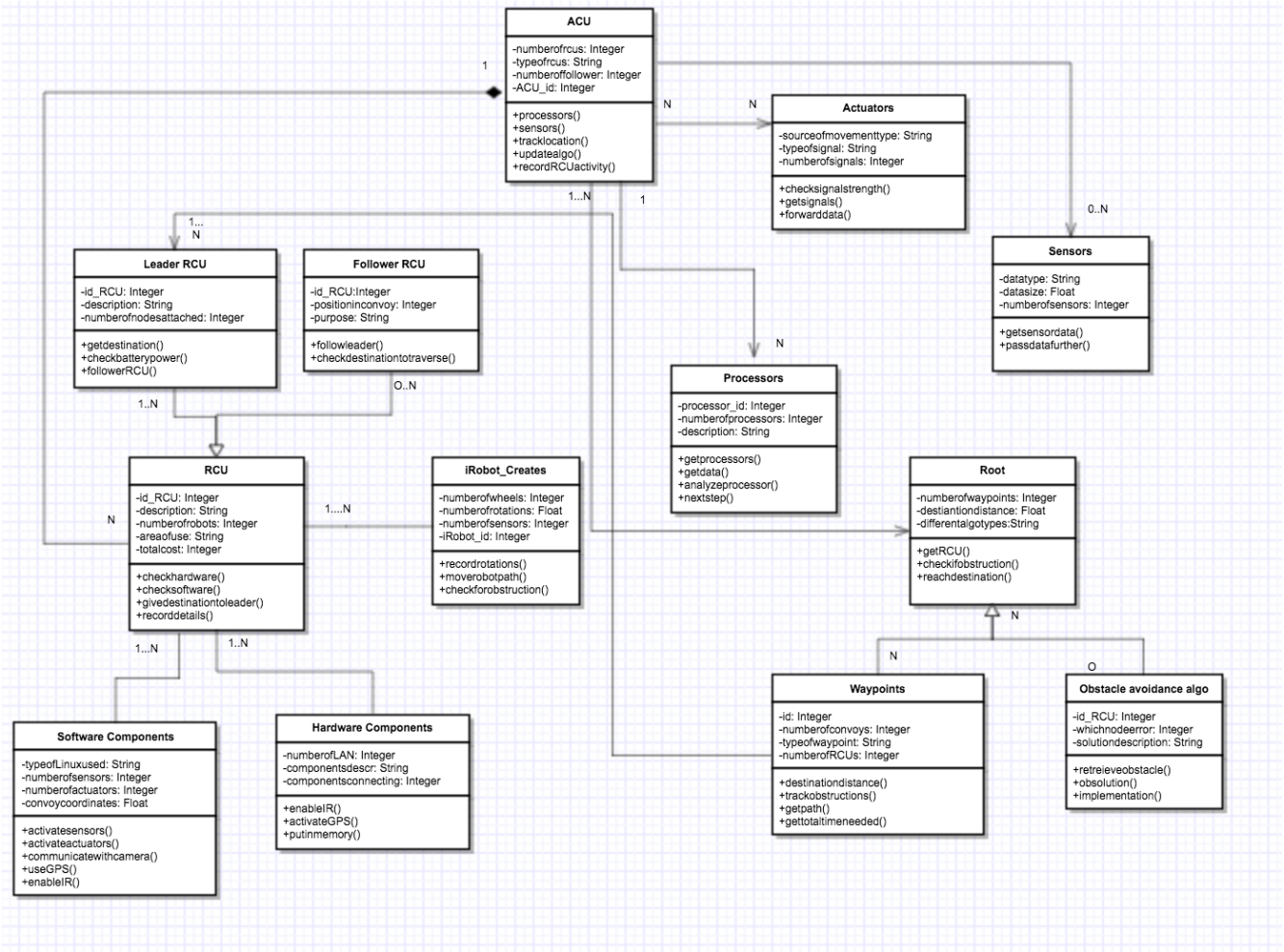
Gliffy / **recharge activity diagram, v2**



3.3 Class Diagrams

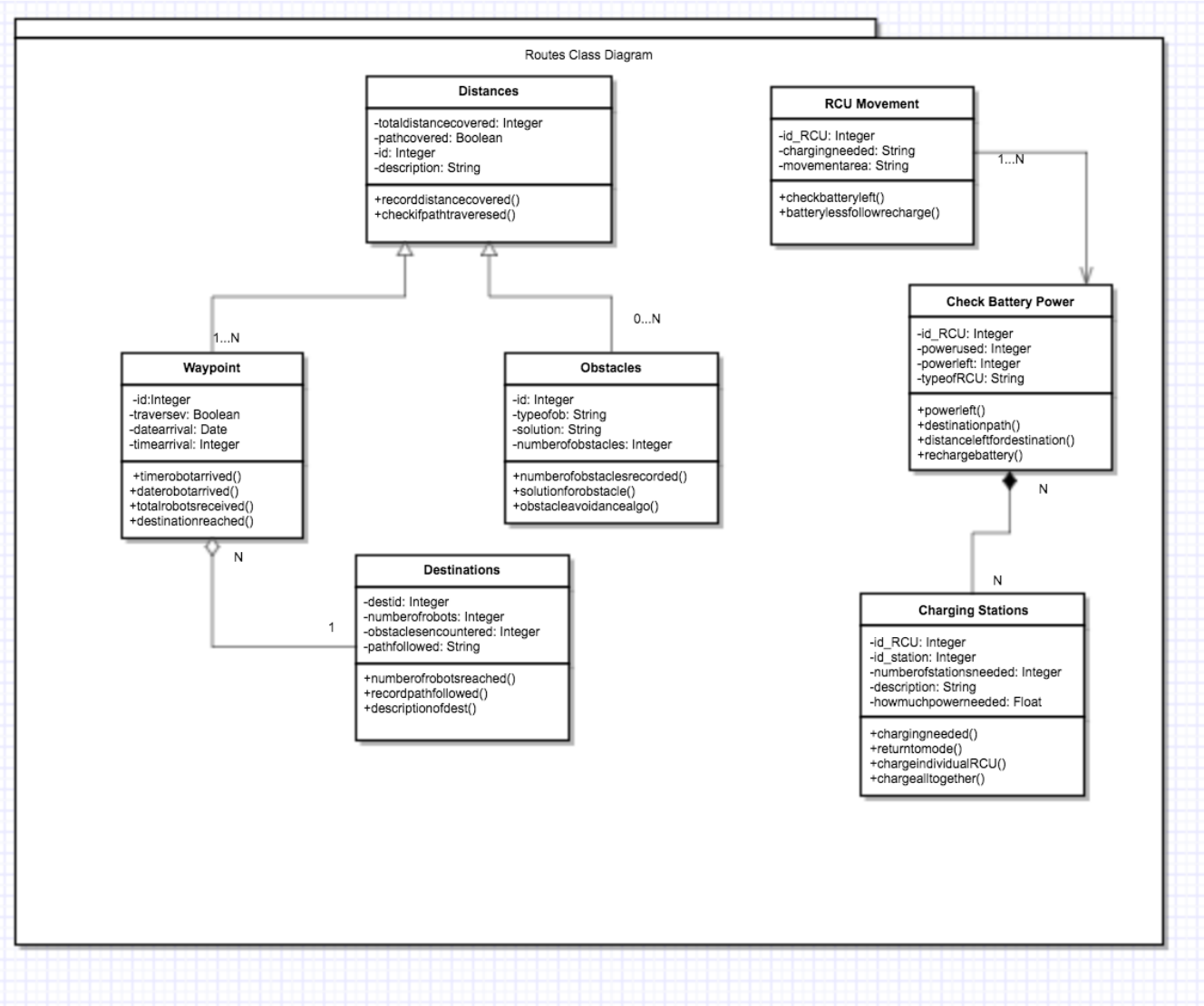
3.3.1 Architecture of RoboCon

Gliffy / **architecture class diagram, v2**



3.3.2 Routes of Robocon

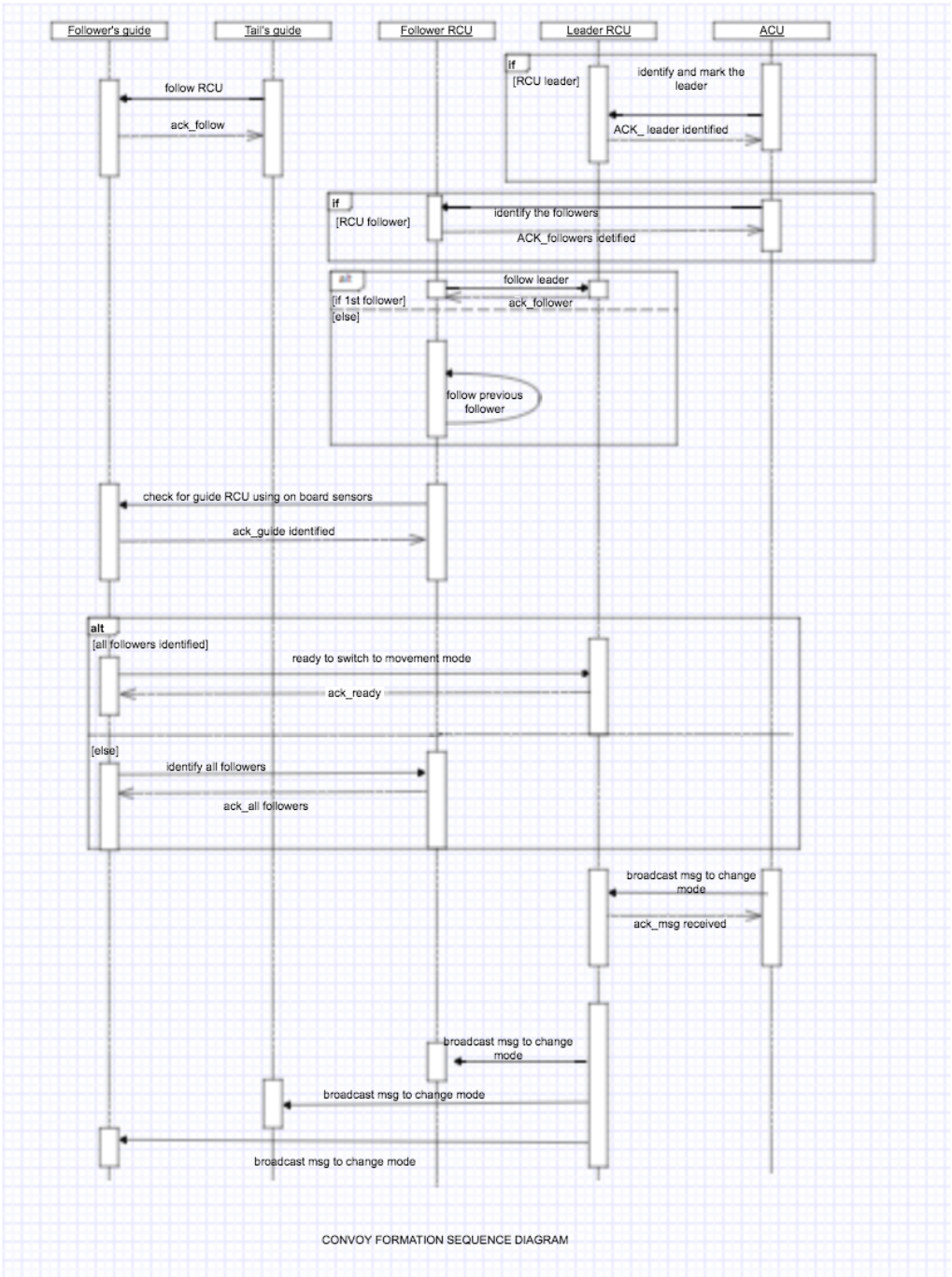
Gliffy / **route class diagram, v2**



3.4 Sequence Diagrams

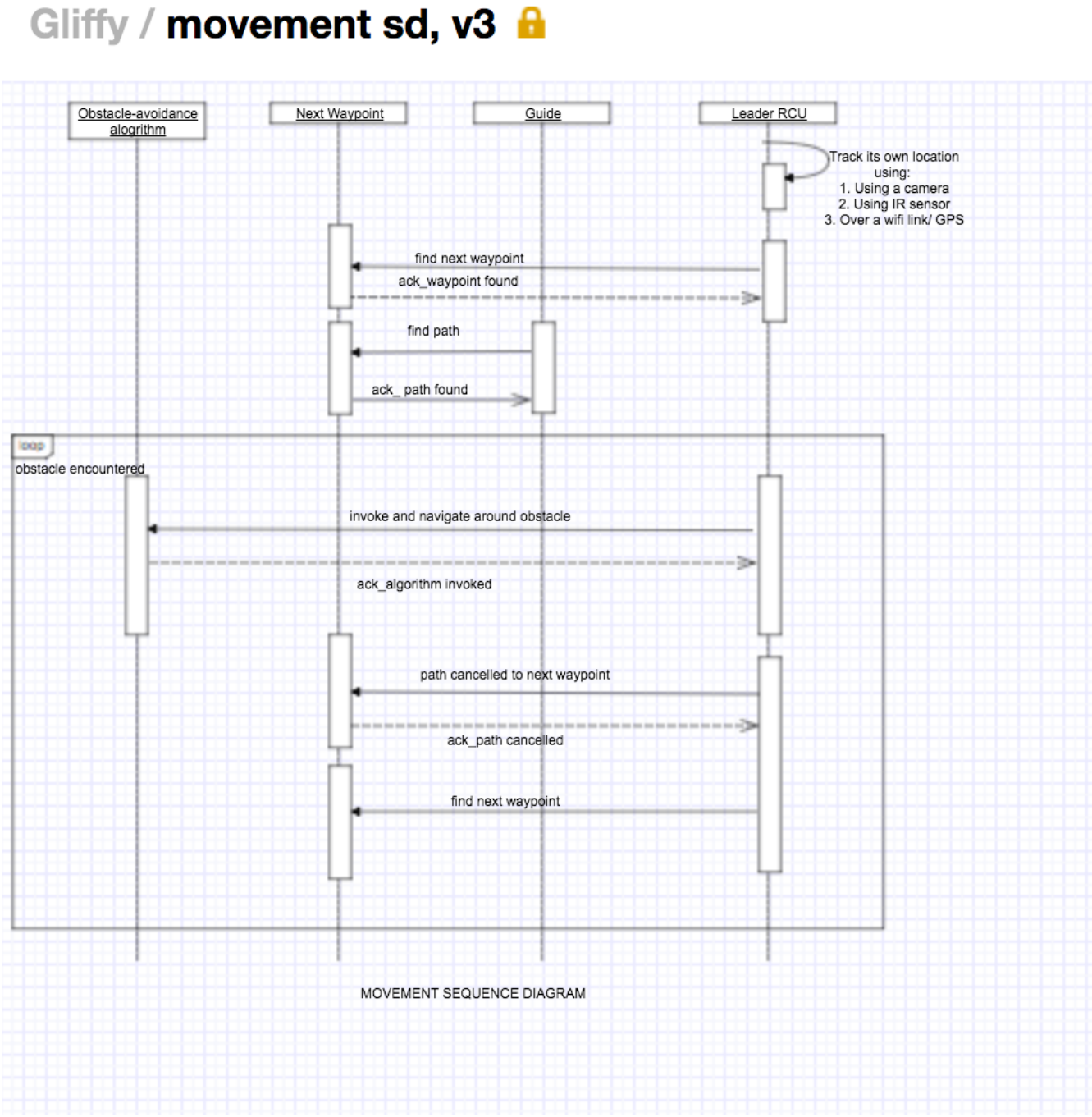
3.4.1 Convoy Formation sequence diagram

Gliffy / **formation sequence diagram, v5**



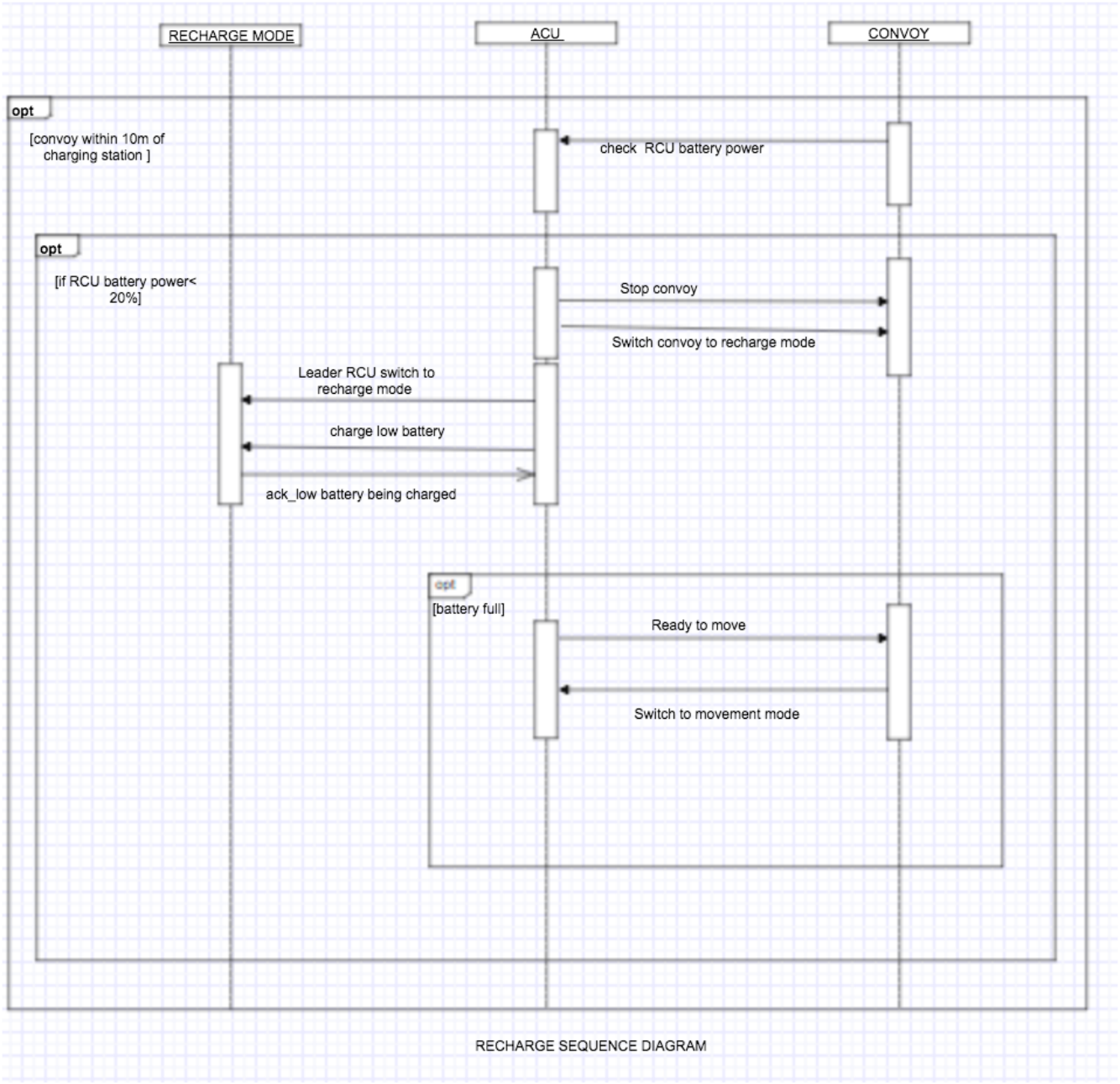
CONVOY FORMATION SEQUENCE DIAGRAM

3.4.2 Movement sequence diagram



MOVEMENT SEQUENCE DIAGRAM

3.4.3 Recharge sequence diagram of RoboCon



4 Other Non-functional Requirements

4.1 Performance Requirement

Project RoboCon needs to have the following performance requirements:

● The followers in the convoy formation mode, should either follow the leader if it is the first follower, else it should follow its previous follower.

● There is only 1 guide for each follower.

● If any obstacle is encountered while traveling, then obstruction- avoidance algorithm should be induced and the new pathways should not be discovered till the obstacle is handled successfully.

● If any RCU has a battery power less than 20%, then convoy should come to halt phase and should resume once the battery is recharged.

4.2 Safety Requirement

Project RoboCon must follow the following safety requirements:

- Both IR receiver and transmitter should be present; to avoid loss if either one is not present.
- ACU should be handled efficiently, else the system will shut down. Thus it should be altered with only when the developer has proper knowledge.
- Its application should not be done in places where there is direct money transfer system.
- If a follower breaks down or stops following the preceding RCU , an alternate solution should be provided.

4.3 Software Quality Attributes

Project RoboCon should have the following software quality assurance features:

1. Reusability
    a. It should be applicable in various aspects of life, having only 1 specific use might not be that productive
    b. It needs to be pretty easy to set up, otherwise no one will want to use it
2. Portability
    a. Deployment should be easy
    b. Installation should simple
3. Adaptability
    a. Should be adaptable to different areas
4. Usability
    a. Should be easy to use
    b. Simple to understand
5. Testability
6. Maintainability
    a. Should be easy to maintain
    b. Maintenance cost should not be high