

Aquaticus Game Mechanics

August 2018

Michael Benjamin, mikerb@mit.edu

Michael Novitzky, novitzky@mit.edu

Paul Robinette, paulrobi@mit.edu

Department of Mechanical Engineering, CSAIL

MIT, Cambridge MA 02139

1	Aquaticus Game Mechanics Overview	1
1.1	The Aquaticus Playing Field	1
1.2	Basic Rules of Aquaticus	1
1.3	Aquaticus Implementation	3
1.4	Aquaticus Platforms	4
1.5	Aquaticus Autonomy	5
1.6	Robot and Human Communications	5
1.7	The Aquaticus Simulation Environment	6
1.7.1	Simulation Mode: Single User	7
1.7.2	Simulation Mode: Multi-User	7

1 Aquaticus Game Mechanics Overview

The competition is described here in terms of the playing field environment, the rules of engagement, how certain components are implemented in software, a description of the platforms, inter-vehicle communications and the simulation environment for training and development. The competition is based on capture-the-flag. The flags are physical buoys in the water, but flag "grabbing" is done virtually through software. Likewise, tagging between vehicles is also done virtually through software as discussed in Section 1.3.

1.1 The Aquaticus Playing Field

The Aquaticus field is a nearly 1.3 square kilometer rectangular area right off the docks of the MIT Sailing Pavilion. See Figure 1. It is 160 meters long and 80 meters wide. Each team has a home half containing its own flag about 20 meters from the end of the field, roughly centered in the other direction. Around each flag is the home team "zone", which has implications during the competition explained in Section 1.2. The flag is a small colored buoy that is visible to humans roughly when they are within the zone containing the flag. The zone around the flag is not demarcated in any way. The boundary of the entire playing field may or may not be demarcated by a lines of small buoys. The edge of the playing field closest to the dock allows for a small buffer zone between the dock and the field. This is for the safety of the competitors.

1.2 Basic Rules of Aquaticus

The basic goal of the Aquaticus competition is to capture the enemy flag and safely return to one's own zone with the flag. The basic components are depicted in Figure 2. Here we describe what

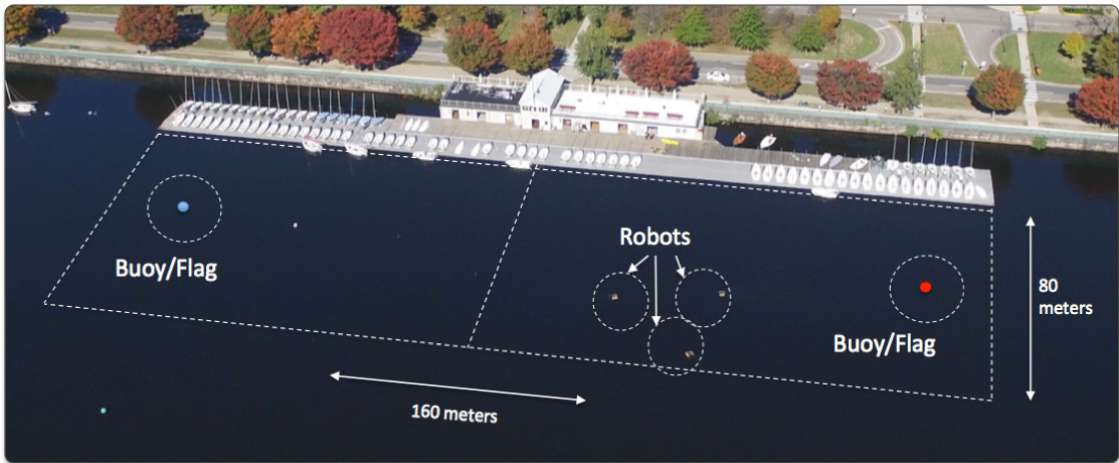


Figure 1: **Aquaticus**: The playing field situated at the MIT Sailing Pavilion on the Charles River.

constitutes a flag capture, and what constitutes a safe return, and what happens to the game upon a successful flag capture. The conceptual field is shown in Figure 2. Note each team consists of two human and two robotic platforms that begin the competition anywhere in their home zone. The Blue Team Zone is also known as the Blue Team’s *home zone*. The Blue Team side of the field is also known as the Blue Team’s *home field*.

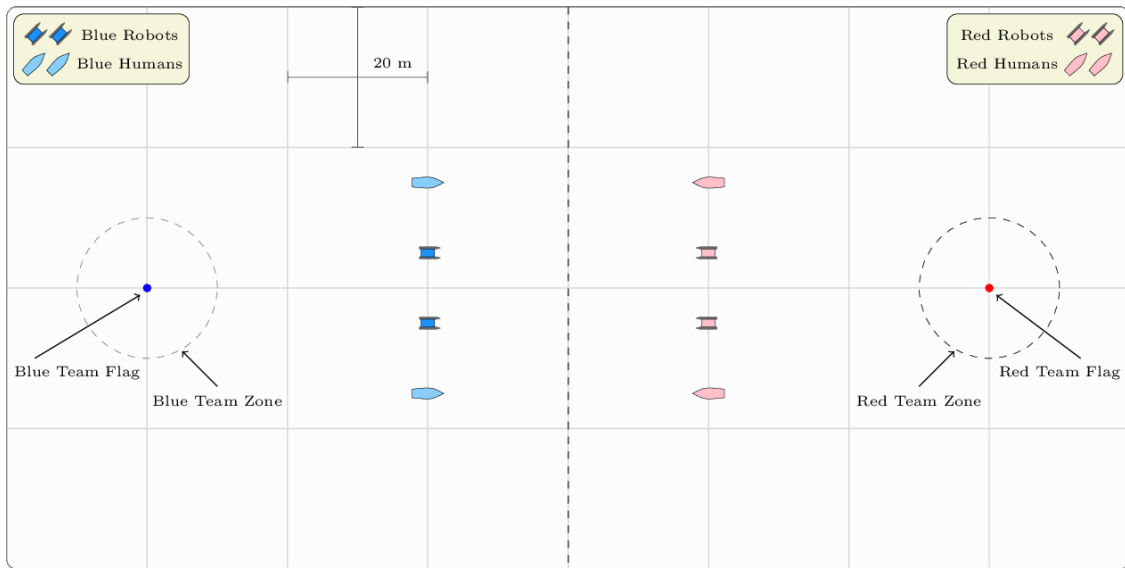


Figure 2: **Aquaticus**: The playing field is 160 x 80 meters with each team owning one half of the field. Each team has a flag and a home zone roughly 10 meters in diameter around the flag. Each team is comprised of two humans and two robots.

A vehicle may capture the enemy flag by entering into the roughly 10 meter zone around the enemy flag and declaring a *flag grab*. The mechanics of a grab are discussed in Section 1.3. A grab is granted if it is close enough to the flag, i.e., in the zone, and if the grabbing vehicle is not currently

tagged, and the flag has not already been grabbed by another teammate.

A vehicle may be *tagged* by a human or robot from the other team if (a) the tagging vehicle is within 10 meters of the tagged vehicle, (b) the tagging vehicle is in its home side of the field, (c) the tagged vehicle is on the enemy side of the field, and (d) the tagging vehicle is not itself currently tagged. Once a vehicle is tagged, it must return to its home zone to be un-tagged. In the meanwhile it may not tag or grab a flag. A tag may only be applied to the closest enemy vehicle where there is more than one taggable enemy vehicle. In this way, teammates can work together, with one protecting the other to absorb tags. A tagging vehicle must wait a period of time (default 30 seconds) after making a tag before it can make another tag. If a vehicle for whatever reason exits the playing field it is also automatically in the tagged state.

Once a vehicle has grabbed a flag, a "safe return" to its home zone is accomplished if it is not tagged prior to entering the roughly 10 meter home zone around its own flag. If it is tagged, the flag automatically is (virtually) returned to its home position. If the flag is successfully returned, the score is noted, and the enemy flag is returned to its home position. During a competition a score may be accomplished several times.

1.3 Aquaticus Implementation

Aquaticus is implemented in part through software components that make the issues of tagging and grabbing viable. Clearly we don't want vehicles physically tagging each other, and furthermore, the implementation of a physical grab of a flag would be difficult for a robotic platform, and would prevent the notion of an automatic return of the flag - we do not want any officials on the field tasked with resetting flags. For both tagging and flag-grabbing, a software module is run on a shoreside computer, the flag manager and the tag manager. These modules, at all times, are consuming position reports sent by the vehicles over WiFi. The managers know all the vehicle locations, and they know the flag locations.

A vehicle *applies a tag* by declaring a tag request to the tag manager. For a robot, this is done by sending a formatted message over WiFi. For a human, this is done by speaking a tag request into a voice recognition system on the hardened laptop in the human platform, which converts it to the same formatted message to the tag manager over WiFi. The tag manager either grants or denies the tag request based on the vehicle locations and states of the vehicles described earlier. When a tag request is made, a tagging target is not specified. Instead the tag is applied to the *nearest* taggable enemy vehicle. A minimum interval of time between tag requests is enforced. Once a vehicle is tagged, the tag manager also posts a message locally to the flag manager, since the tagged or un-tagged state of the vehicle has ramifications to the flag manager logic, e.g., a tagged vehicle cannot grab a flag.

A vehicle *grabs a flag* by declaring a flag grab request to the flag manager. For a robot, this is done by sending a formatted message over WiFi. For a human, this is done by speaking a flag grab request into a voice recognition system on the hardened laptop in the human platform, which converts it to the same formatted message to the flag manager over WiFi. The flag manager either grants or denies the request based on the vehicle locations and their state, and also the state of the flag. The flag manager knows the vehicle location from position reports, and it knows the tagged state of a vehicle based on messages from the tag manager. The flag manager will generate a response to a vehicle requesting a flag grab indicating success or denial. If the grab is denied, the

reason is also sent.

Each of the human platforms are equipped with a hardened laptop computer along with a headset and microphone. Tag and flag grab requests are initiated via spoken commands. The same voice recognition system is used by the humans to generate messages of their own design to their robotic teammates. These messages typically are from a set of messages tied robot commands to either initiate or change a robot task, or request information from a robot teammate.

All vehicles are noting their GPS position, heading and speed at all times. Even the human vehicles are equipped with a GPS attached to the same hardened laptop used for processing voice commands. These vehicle reports are packaged into a single message and continuously broadcast from the vehicles to all other platforms as well as the shoreside. This is similar to an AIS in marine systems. This report is not only used by the tag manager and flag manager, but also used by all the robotic platforms for situational awareness to prevent collisions. Each robotic platform is running collision avoidance software based on the COLREGS to prevent collisions. Users do not have the option of overriding this on their robots.

1.4 Aquaticus Platforms

There are two types of platforms in Aquaticus. The human platform is a motorized kayak, called a MOKAI, shown in Figure 3. It is augmented with an on-board computer, WiFi, GPS and power. This is a commercial platform that runs on a gas engine and has a top speed of roughly 4 meters per second (9 miles per hour). It is not capable of going in reverse and currently is moving slightly forward when at full idle. It is load and presents a challenge for the voice recognition system, but voice recognition is achievable.



Figure 3: **The MOKAI:** The motorized kayak, MOKAI, serves as the platform for human teammates.

The robotic platform is a Heron M300 made by Clearpath Robotics, shown in Figure 4. It is augmented with a payload computer running GNU/Linux and the open source MOOS-IvP autonomy software from MIT. It is capable of a top speed of 1.8 meters per second (4 miles per hour). It is equipped with a GPS, compass, and WiFi.



Figure 4: **The Heron M300:** The Heron M300 unmanned surface vehicle serves as the robotic teammate.

1.5 Aquaticus Autonomy

The robot platforms are designed to support a payload autonomy computer with an interface to the robot actuators and sensors. The generic nature of this interface means that virtually any autonomy solution is supported as long as it abides by this interface. Loosely speaking, this interface accepts left and right thruster commands from the autonomy system. In return the vehicle provides position and heading information to the autonomy system. The autonomy system however may comprise plans, strategies and internal state in a manner designed by the human teammates. In practice the payload computer is running a fairly evolved set of autonomy software available through the MOOS-IvP open source project at www.moos-ivp.org.

The autonomy system may be configured in a virtually limitless number of configurations by the user. This is true without the user writing any new code, but is especially true considering that any new software module is fair game. Variations may be achieved by (a) choice of behaviors comprising the autonomy system, (b) choice of configuration parameters for each behavior, (c) choice of autonomy modes and definition of which behaviors are associated with those modes, (d) the voice commands or other circumstances that may trigger a mode change, or (e) any other software running on the autonomy system that may automatically alter any of the above based on sensed events or other algorithms.

On one hand Aquaticus is designed to allow limitless software additions and configuration changes by the user to create a meritocracy and hopefully some surprises about what ends up being most effective. On the other hand, a design goal is to have a system that a new user can learn and use without any software or autonomy background. To achieve both objectives, a reasonably powerful robot autonomy mission configuration is provided as a default. This allows visitors and impromptu test-subjects to use the system with a short period of training that can be accomplished on the same day as one's first competition. It also allows a competitor or team to ponder completely novel strategies and have the time and means to change or extend the default autonomy configuration.

1.6 Robot and Human Communications

There are three types of communication between platforms, (a) human-to-human (b) robot-to-human and (c) robot-to-robot. Furthermore all platforms are in communication to the shoreside laboratory for both safety reasons as well as management of the competition. All communications is achieved by a combination of marine radio and WiFi.

Communications from a human to human teammate is accomplished over a commercial marine radio. This radio is activated by a push-to-talk setup in each human-operated platform. Each team uses a different marine channel. The same radio is used for comms to the shoreside to control operations, make game announcements and overall safety.

Communications between a human and robot is done with the same radio headset for the human, with a separate push-to-talk but directing voice to an on-board laptop. The laptop is running a voice recognition system trained and configure to recognize certain key words and phrases. These voice commands are then sent over WiFi to the robot to trigger changes in the robot autonomy state, or request information from the robot. Robot comms to a human also takes place over WiFi with the on-board laptop performing text-to-voice functions resulting in a computer-generated audio message in the human operator’s headset. The arrangement is shown in Figure 5.

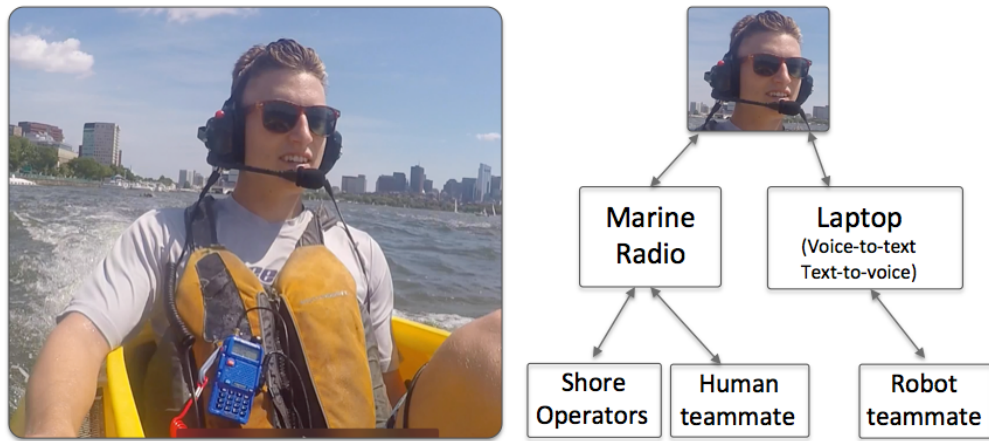


Figure 5: Communications from the human competitors is accomplished by commercial marine radio between humans and shore operators. Communications to a robot teammate is accomplished using a hardened laptop on board with the human. The computer is performing voice-to-text and text-to-voice operations.

A key design consideration for each team is the choice of autonomy modes implemented in each robot, and the corresponding verbal cues available to the human operators to affect mode changes. The environment will be noisy and the vocabulary will need to be chosen wisely to ensure reliable interpretation of verbal commands and queries.

1.7 The Aquaticus Simulation Environment

The Aquaticus simulation environment serves to develop the competition software and allows users to practice and become acquainted with the software and game strategy. Any number of robots and humans can be simulated. Typically a human in the simulation environment is equipped with a joystick to emulate the experience of driving the MOKAI platform. A snapshot from simulation is shown in Figure 6.

The full operation of Aquaticus involves eight platforms, four autonomous and four human-controlled. This can present a challenge if the goal is to use the simulation to incrementally test software components and strategies. Users may at times gather to simulate in a group of four, but it is often useful to support subsets of the full competition, all the way down to a single user working alone on

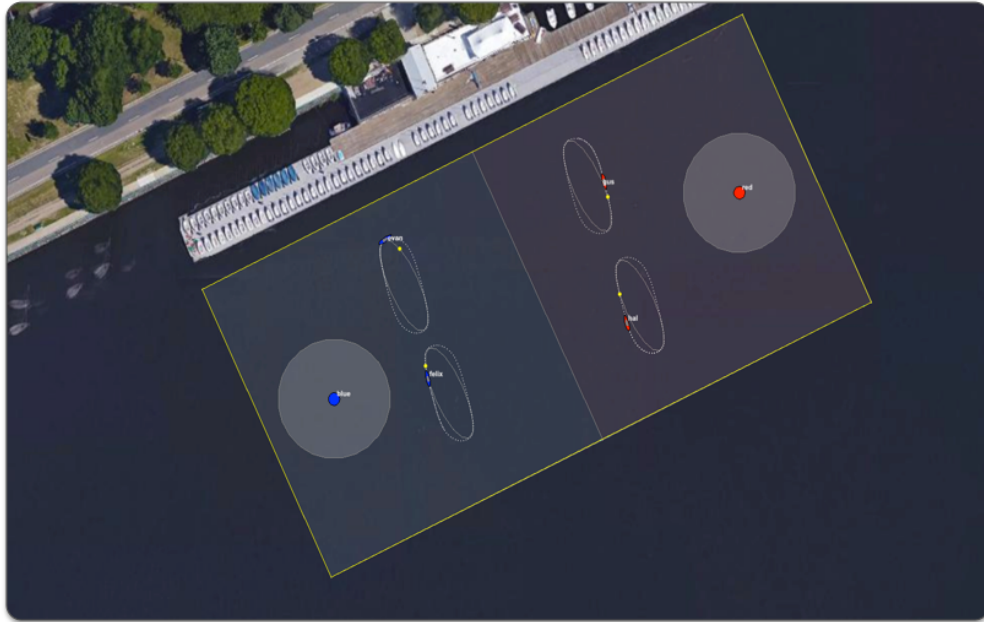


Figure 6: **Aquaticus Simulation:** The field of play is rendered with four platforms, a subset of the full 4-on-4 Aquaticus configuration. This field of play perspective is common across all variations of simulation. The GUI software itself is not specific to Aquaticus - it is available through the MOOS-IvP open source project. The particular configuration, chosen operation area, buoys and set of support applications, is specific to Aquaticus.

a laptop in silence. An overview of these modes is describe here:

1.7.1 Simulation Mode: Single User

In single user mode, typically the user is working with a subset of the competition, most likely from the perspective of a human with two robot teammates competing against three robot adversaries. The user controls the human platform by joystick connected to the simulation computer. There are two options for communication to the robot teammates. The user may have a headset and microphone and interact through the configured voice recognition environment, replicating the experience in MOKAI on the water. The user may also use a GUI pop-up display with all the possible voice commands represented by buttons. This is less realistic, but does allow for simulation in environments where the user may need to be quiet, or in situations where the headset is not working or available that day.

1.7.2 Simulation Mode: Multi-User

In multi-user mode, the user may be interacting with one or more other humans. These other humans may be teammates, adversaries or both. All humans are controlling their simulated vehicle via joystick, and interacting with their human and/or robotic teammates over a headset radio with voice-to-text and text-to-voice implemented on the simulation computer. In this setup, a *shoreside* computer is directing the operation display as shown in Figure 7. A key difference between the simulation user experience and the in-water experience is that the human has visual access to the tactical picture via the display. This enhances the situational awareness in terms of the position and

trajectory of teammates and adversaries. Another difference is that humans can typically overhear the conversation between other human adversaries in the room. This factor could be eliminated by separating the teams into different rooms.



Figure 7: **Aquaticus Multi-User Simulation:** Users are connected to the simulation with same headsets worn on the water. A simulated vehicle is under joystick control with functionality similar to the joystick control of the thrusters on the actual vehicle.