Preparation of GUSS, the Autonomous Tugboat, for the 2011 AUVSI ASVC Competition

Thomas C. Furfaro, Janine L. Mask, and Joseph Marquardt

***Abstract*— For the fourth consecutive year, Florida Atlantic University (FAU) is participating in the Autonomous Surface Vehicle Competition (ASVC), organized by the Association for Unmanned Vehicle Systems International (AUVSI). Student members of the Department of Ocean and Mechanical Engineering have spent the Spring 2011 semester updating and rebuilding the *GUSS* unmanned surface vehicle. This same platform has been used for the last two ASVCs hosted by AUVSI. This year, the competition poses new challenges and requirements of participants, leading the FAU team to perform an extensive retrofit of *GUSS*'s control hardware, vision system, and navigation algorithms, and to add on extensive auxiliary systems to solve specific competition challenges. This paper outlines the new system design with respect to the competition obstacles and requirements.**

I. MEET GUSS

The competition rules are in similar spirit to years past, with the same hard requirements for vehicle size, weight, powerplant, and failsafe design. Accordingly, the basic hullform used by FAU in competitions past is again being used as the starting point in vehicle design (Table 1).

Parameter	Value
Length Overall	1.37 m
Length at waterline	1.26 m
Beam	0.58 m
Maximum beam	0.58 m
Draft	0.23 m
Molded depth of hull	0.10 m
Displacement	24 N
Length to beam ratio	2.7

Table 1: Hullform Characteristics

The vehicle, *GUSS* (Figure 1), resembles a twin-drive escort tug in shape. It is propelled by two azimuthing drive units, located at the estimated longitudinal hydrodynamic center. The drives have a 180° operating angle, with this range being centered on an axis parallel to the transverse plane. The result of such a configuration is the ability to perform pure-sway maneuvering by directing the drives perpendicular to the vessel centerline (Figure 2).

Figure 1: GUSS

The drives are each powered by a brushed DC motor via a right-angle transmission. A pair of 12 V, 12 Ah lead-acid batteries provide storage. The pair of motors are driven by a single motor controller, interfacing to the command computer via remote-control standard pulse width modulation (PWM) signals. The azimuth angles are controlled independently, each by a single high-torque servomotor. Thus, with full articulation and modulated thrust capabilities, four separate control inputs are required.

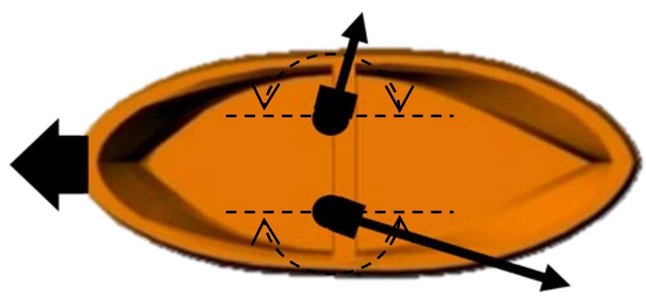

Figure 2: Propulsion System – Control actuation degrees of freedom are thruster RPM and thruster azimuth angle on both port and starboard

It is here where similarities to previous years' iterations of *GUSS* end. All other elements of the vehicle have undergone drastic redesign. High-level changes include a completely new machine vision system, new control hardware and remote supervisory control (RSC) architecture, and totally new auxiliary systems.

II. Machine Vision

The CMUcam3 camera pair used on *GUSS* in past years was upgraded to a pair of Logitech Pro 9000 webcams. Stereo vision with these higher-resolution cameras will improve buoy localization. Coupled with an additional CPU dedicated entirely to image-processing, the new vision system will allow more advanced processing techniques to be performed at a higher throughput than the previous system. Processing is based around the free, open-source OpenCV library. This library, originally developed by Intel and now maintained by Willow Garage, provides extensive image-processing tools in C and C++, with specific emphasis on real-time tasks.

Blob tracking, edge tracking, template matching and depth perception by stereo vision are the necessary functions of *GUSS*'s camera system for this year's competition. Blob tracking is the most important of the four camera functions because it applies to all parts of the course. It is done by converting the incoming video stream into the HSV color space. A range of color values that statistically defines the signature of a target is found through testing. The real image is analyzed to find areas that fall within this range, and then a bounding circle is found that captures the highest concentration of color. This is the estimate of the target location (Figure 3). This is used to locate the starting gate and channel marker buoys, as well as a secondary estimator for target location.

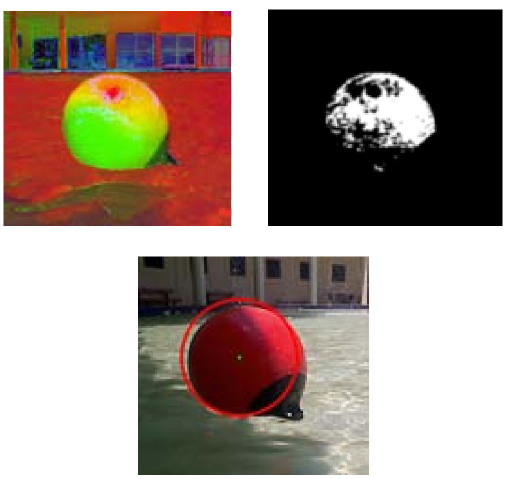

Figure 3: Blob tracking example using OpenCV

In the total system, the camera processor and cameras themselves behave as slave devices to the command computer. A cross-over ethernet link between the two allows the control processor, i.e. the master, to dictate which type of search mode the CPU should be operating in, while receiving results across the same connection.

III. Control, Navigation, and Command Hardware Improvements

For the 2010 competition, changes in the way that a team's in-water time was calculated dictated a new, remote supervisory control system. The team needed to be able to instantaneously take control of the vehicle during autonomous operations. This would allow the team to cut out transition time between runs, by forcefully driving *GUSS* to the dock to launch for the next attempt. The system that was implemented met the requirements and worked, but it was not a long term fix.

This year, a completely new remote control interface was designed to give team members the instantaneous control needed, while providing extensive failsafe mechanisms to prevent accidents from occurring during automated operations. This is performed by using an intelligent switch that decides under which conditions to permit autonomous or human control. Practically, a small microprocessor monitors remote control communications, radio signal status, and control computer outputs, and logically switches between human and computer control.

Furthermore, improved navigation sensors have been integrated in the vehicle subsystems this year, including both a new compass (the TCM5 from PNI Corporation) and a new inertial measurement unit (Motion Pak II IMU from Systron Donner). Coupled together with the existing GPS unit, the navigation suite should lend itself to more accurate position and orientation estimates by the software navigator.

All of these changes are being implemented on a newly designed and fabricated motherboard, a 5 layer printed circuit board that interfaces power and signal electronics for all sensory, communication, and command and control systems on the vehicle. The new system is much more elegant than the previous years' implementation, and is fabricated in a more permanent fashion, hopefully providing a backbone for future work with this, or other similar vehicles.

Lastly, tachometers were installed on each drive unit to monitor individual drive speed. This allows for an on-board speed controller to modulate motor controller command inputs in order to maintain shaft-speed setpoints. Overall, this approach lends itself to more consistent high-level controller performance, due to the sub-controller virtually guaranteeing that high-level commands will be maintained by the plant.

IV. Auxiliary Systems

Obstacles new to the 2011 competition are driving auxiliary system requirements in unprecedented directions. Beyond the usual buoy channel and water-gun elements, new target identification and localization obstacles, as well as a payload retrieval challenge, inspire the creation of new *GUSS* add-ons.

A. Water gun

As in competitions past, there is a mission component that requires the autonomous system to identify and localize a target of specified geometry, and fire upon it with a water jet. This year, the target is actually a specific area cut out of a large printed poster. *GUSS* is outfitted with a 350 GPH pump operated by a transistor switched by the motherboard. The output nozzle is rigidly fixed to the hull, so the entire vehicle will be moved to aim. The main navigation cameras will be used to locate the target area, and this direction information will be relayed back to the control computer which actually performs the aiming and firing operations.

B. Temperature Sensor

A different element of the mission requires the vehicle to visually or thermally identify and localize a target, and report the estimated location back to a judges station via a TCP message. The goal target is hidden among a field of 3 other decoys, which do not have the same visual signature (different graphic printed on each), and all of which are lower temperature than the target of interest. Thus, *GUSS* is being augmented with an infrared temperature sensor to measure target temperatures at a distance. This sensor will be mounted to a CMUcam3, which will attempt to target the right outline. The temperature sensor will then be read for surface temperature information. Thus a target which has been localized (by the CMUcam3 blob tracking algorithm plus the on-board navigator) and also has a higher temperature reading will be reported as the target of interest.

C. Payload Retrieval

Possibly the most difficult task this year is the payload retrieval challenge. For the 2011 competition, a large, raised platform holds a brightly colored tennis ball. The minimum size specified for the dock, 3' x 4', makes a system that is contained entirely water-side very difficult to implement. This is especially difficult for smaller vehicles such as GUSS. Thus, an agent is being designed to deploy from the water-bourne vehicle to the dock, retrieve the payload, and return again to the mothership. The FAU team is using a popular robotics system, the LEGO Mindstorms kit, to rapidly construct and integrate a system capable of performing such a task. The Mindstorms robots provide many advantages to the developer, because that basic servo control and sensor interfacing can be performed within minutes of opening the box. The largest downside is that the systems are not in any way inherently waterproof, so extra caution must be taken to protect the subsystem from failures related to water intrusion.

V. Control Algorithm

The high-level software architecture for the FAU vehicle is that of an event-driven finite state machine. The software is constantly in a specific mode, looking to accomplish a simple task, such as go to a coordinate or follow a heading, while a higher mission-control loop is monitoring global variables for important changes that might dictate a change in action. For example, a timeout countdown expiring will, in almost all cases, dictate an advancing of current goals from the present target. Also, other events, such as changes in goal proximity or camera target recognition, will trigger a change of controller setpoints. A basic overview of what this decision process looks like can be seen in Figure 4.

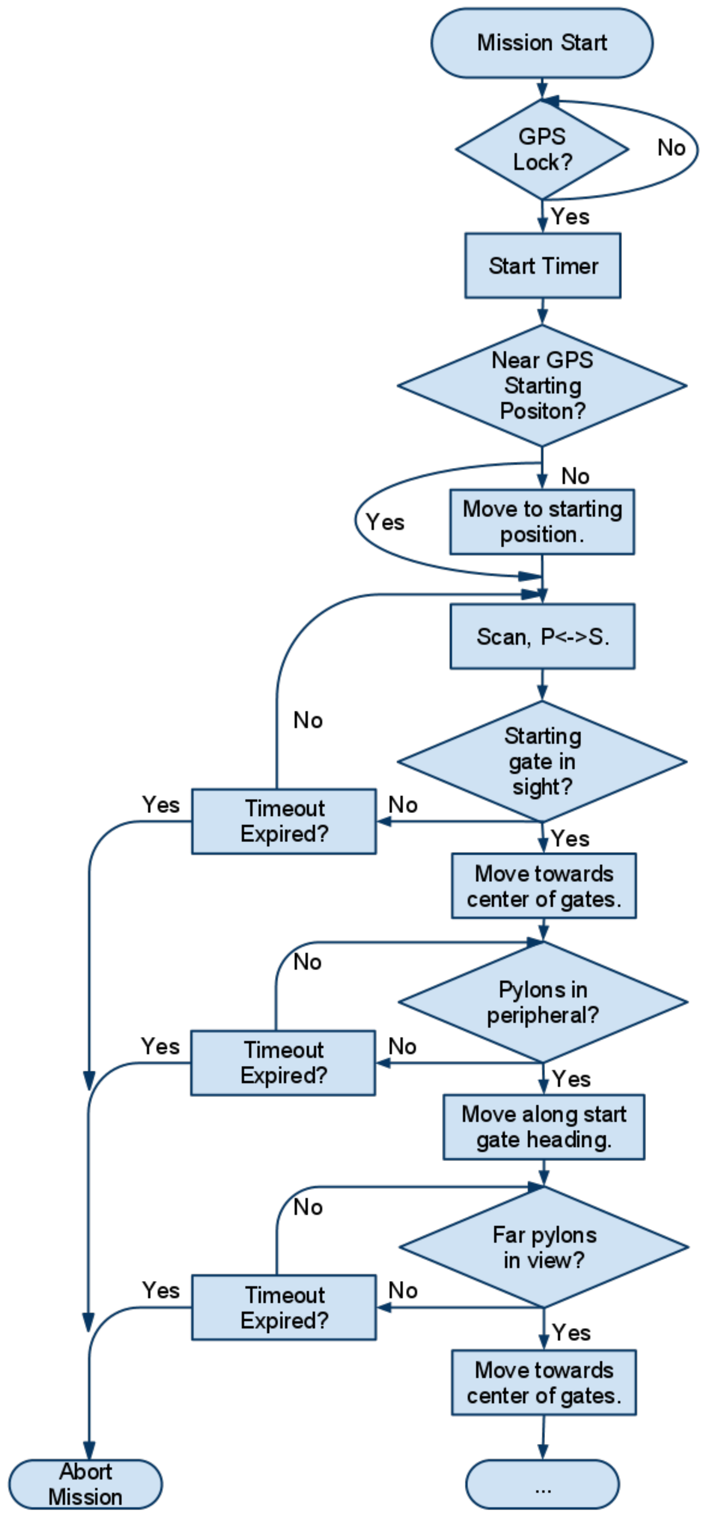

Figure 4: Mission control algorithm example

The middle tier controller has much more local goals, taking set-point information and maneuvering orders from the mission planner and feeding those into the control calculations. For example, these could be commands such as constant heading at constant speed, or track following at constant speed, or some sort of loitering with oscillating heading. The high-level goals are dictated by the mission planner, but the RPM and angle setpoints for the motors and drives, respectively, are set by the middle controller. This controller compares state estimates for position, speed, and orientation, with targets for the same parameters, and commands the plant according to some control law (Figure 5).

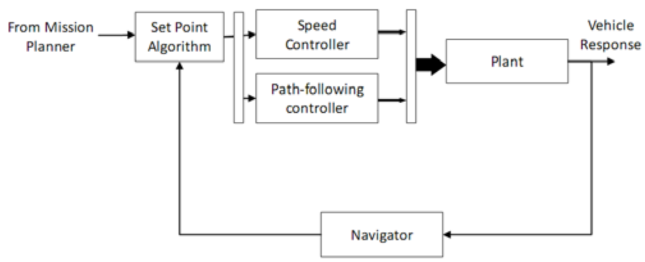

Figure 5: Mid-level controller, for speed maintenance and track following.

The lowest level controller does nothing but ensure that commanded setpoints for engine speed and thruster angle are reached and maintained. The mechanical system that drives the boat, while durable, is not always stationary or deterministic. That is, for example, over time a specific servo command angle might not correspond to constant drive angle, due to slippage in the linkage system.

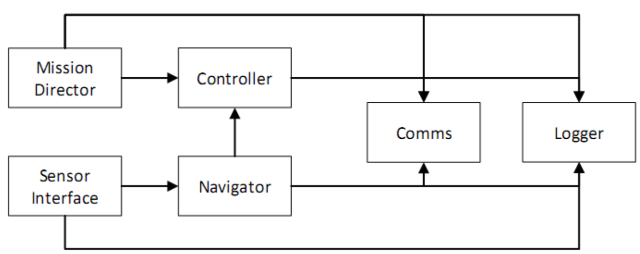

Figure 6: Thread inter-communication scheme.

In terms of actual software, this is all accomplished using multi-threading utilities from the GNU C Library. Multi-threading allows single core processors (such as that aboard *GUSS*) to appear, at least virtually, as multi-core machines. Threads split processor time according to priority and need, resulting in different threads apparently being performed simultaneously. This kind of parallel processing is hugely advantageous, lending itself to asynchronous events triggering different event handlers, such as un-synced serial or ethernet communications, or command inputs from a human operator. However, threads also come with their own programming hazards, primarily those dealing with re-entrancy. Non-reentrant software is inherently not thread safe, as collated threads may have unmitigated access to global variables, performing read and write operations at critical periods where errors in calculation might occur. The practical solution to this is the utilization of semaphores or mutexes, which are a series of locks that, when properly implemented, help protect critical section of code from being contaminated by intervening threads. The inter-thread communication layout is described in Figure 6.

VI. CONCLUSION

The newest iteration of the AUVSI ASV Competition presents unique challenges, driving year-to-year vehicle modification and improvement. During the run-up to the 2011 mission, Florida Atlantic University's team has given *GUSS* yet another retrofit. Changes to existing system include an improved remote supervisory control subsystem, a new machine vision configuration, and an upgraded navigation sensor suite. Entirely new systems include a water gun apparatus, an infrared temperature sensing device, and an independently autonomous LEGO Mindstorms robot for payload retrieval.

VII. ACKNOWLEDGEMENT

The team would like to first acknowledge the support of our sponsors, Oceaneering International Incorporated, and the Florida Engineering Foundation. Without sponsors, our team would be dead in the water. The team would also like to acknowledge the support of the FAU Seatech engineering staff, particularly the machinist, Luis Padilla, and the electronics lab staff, Ed Henderson, John Kielbasa, Tom Pantelakis, and Beatrice Gomez. Also, the team members themselves deserve recognition, for all of the time they have put in. They are: James Lovenbury, Mario Mirando, Ivan Bertaska, Will Valentine, José Alvarez, Joe Marquardt, Dylan Tilley, Brenton Mallen, Janine Mask and Thomas Furfaro. Finally, our faculty advisor, Dr. von Ellenrieder, also should be recognized.