Proceedings of the Fourteenth Annual Early Career Technical Conference
The University of Alabama, Birmingham ECTC 2014
November 1 – 2, 2014 - Birmingham, Alabama USA

DESIGN RATIONALE FOR *SUBZERO*

Bennett R Stedwell, Ada A Odey, Kevin McFall Phd
Southern Polytechnic State University
Marietta, GA, United States of America

ABSTRACT

Southern Polytechnic State University's *SubZero* is a littoral-class autonomous underwater vehicle (AUV) built by undergraduate members of the SPSU AUV Team. The vehicle has been continuously modified and enhanced over the past several years, however the current configuration is a complete redesign from past vehicles, the product of a ten-month development cycle. The vehicle was designed almost completely using three-dimensional CAD and simulation in Dassault Systemes' Solidworks design software. Among the new design's features are redesigned camera housings, a new main housing, and a hexagonal, riveted 6061 aluminum exoskeleton to provide structural security while minimizing weight. *SubZero* is equipped with two cameras for challenge recognition and maneuvering computer vision tasks, a pressure sensor for active depth control, and an inertial measurement unit for orientation control.

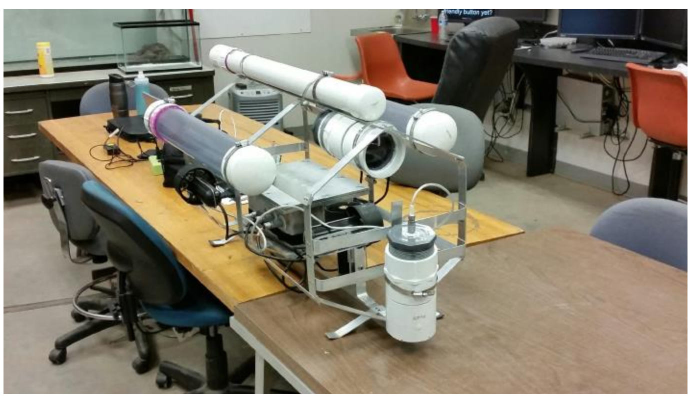

Figure 1. *SubZero*

OPERATING SYSTEM AND LANGUAGES

To avoid unnecessary complication, a software stack is used to provide communications and interfacing with the sensors and cameras. This allowed for additional time to be spent addressing the challenges rather than perfecting the utilities. These functions were provided by Robot Operating System (ROS). As explained on ROS's website, the Robot Operating System is a set of software libraries and tools that assist in building robot applications. ROS possesses open source drivers, state-of-the-art algorithms, and other powerful developer tools. ROS has a six month release cycle; throughout the development cycle, new versions were released and thus had to be updated on *SubZero*, Figure 1. Fortunately, little code had to be rewritten between each update. Using ROS narrowed down the choices of operating system for the on-board embedded computer (ePC). ROS can be run on a variety of Linux distributions and Windows, however only one distribution is fully supported; Ubuntu. Code is not developed on the embedded PC (ePC); instead personal laptops or lab computers are used, which have ROS installed, to develop any software for *SubZero*. One reason for this decision was the concern that doing many compiles on the ePC could wear out the flash memory on the compact flash (CF) card, which is its only means of storage. It was decided to run Ubuntu on both the development and on-board computers, but with each installation customized for its purpose. The development installation has a desktop environment (Cinnamon), a browser, many editors, and other useful tools. The embedded PC installation boots directly into the terminal with the option to start a graphic user interface (GUI) for testing, and little extra software aside from ROS and the code, all of which must fit on an 8 gigabyte compact flash (CF) card. The ROS libraries in use are C++ and Python. Initially, it was decided to write in C++ for the speed of using native code, as the operating system itself is written in C++. After additional consideration, the code was switched to Python to maximize performance capabilities when vision processing tasks are running. When the code was ported to Python, there was no evidence of any performance lost on the other systems.

SOFTWARE ARCHITECHTURE

A closed layer approach is used to design the ROS package. This architecture was chosen for its ease of comprehension for those students new to computer programming. The four layers in the architecture are: Control, Decision, Calculating, and Device. Figure 2 shows the premise of the software stack. The nodes in each layer can only interact with nodes from the layers directly above or below, as well as nodes from its own layer.

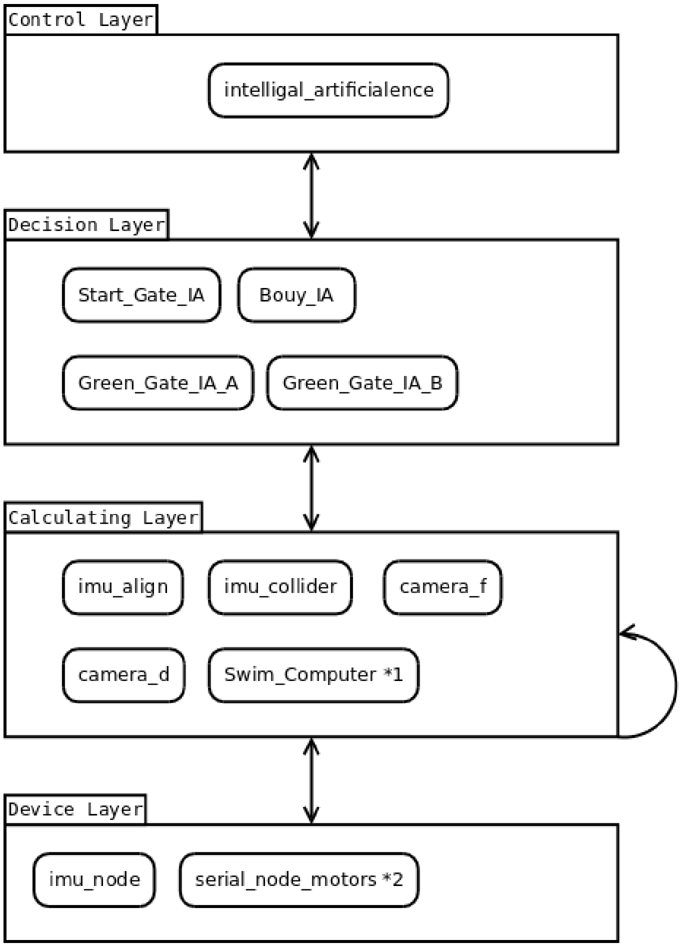

Figure 2. Software Architecture

CONTROL LAYER

The control layer houses only one node: Intelligal Artificialence (IA). This node's purpose is to run the mission objectives that are defined in the decision layer.

DECISION LAYER

The decision layer is where *SubZero*'s task decisions are made. Each task has its own node in this layer; including the intermediate tasks such as re-covering from the previous activity. This approach simplifies updating code and streamlines the review and merging process in the content management system.

CALCULATING LAYER

This layer handles all of the heavy number crunching functions, such as locating the start gate in a frame. Located here are the visual recognition code, the nodes for collision detection, and the swim computer.

DEVICE LAYER

This layer contains software from the ROS community that pulls the data from the hardware and converts it into a usable format. The device layer contains the rosserial nodes used to communicate with the two Arduino Unos, one to control the six thrusters and the other to read data from various sensors.

SWIM COMPUTER

After consulting with colleagues on SPSU's Aerial Robotics Team (ART), they suggested the use of a "swim computer". Unmanned Aerial Vehicles (UAVs) have an on board flight computer which is used to stabilize the aircraft. ART recommended that an actual flight computer board be placed in the vehicle, which could then be sent basic commands to handle stabilization and correcting for drift. However time constraints proved prohibitive to implementing this change, so "swim computer code" was instead built into the software.

CAMERA SENSOR SUITE

The camera housings are constructed from 3" PVC piping, two PVC slip fittings, two PVC slip to threaded fittings, two threaded PVC end caps, and finally two Lexan lenses. For a single housing, the pipe is cut to a length of 8.89 cm, a slip fitting and a slip to threaded fitting is glued in place using PVC glue. While the glue sets, the lens is cut to a diameter of 8.80 cm. Once the glue is fully cured, the lens is placed in the open end of the slip fitting and held in place with silicon.

The internal structure of *SubZero*'s camera housings are constructed from 0.47 cm thick polyethylene, and two 90° steel elbow joints. Two pieces of polyethylene are cut using the same method as the Lexan lens; one is cut with an outside diameter of 8.80 cm and an inside diameter of 3.00 cm. The second piece is a 5.08 x 7.62 cm rectangle. The two pieces are connected using the elbow joints and the circular segment is epoxied inside the threaded end cap. The last three steps are: securing the USB camera to the rectangular section using double sided tape, inserting a Fischer connector into the back of the end cap, and finally attaching the USB wires to the connector; which runs between the ePC inside the main housing and the mounted camera housing. These steps are exactly repeated for the second camera housing.

The housings are attached to the exoskeleton so that one faces forwards and the other faces downwards. The downward camera is used to find the direction markers on the bottom of the pool which leads between the various competition tasks. The forward camera is used for computer vision targets such as the start gate.

For much of the vision code development cycle, FireWire cameras were used. These FireWire cameras have benefits and drawbacks. They do yield images at a high resolution. However, the cameras would randomly return a distorted image; the image would be divided in half and switched (right side on the left and vice versa), one half inverted and in gray scale, with the other half oriented correctly and in proper color. Since *SubZero* relies so heavily on a video feed from the cameras, it was decided to switch to a pair of Logitech USB web cams. This switch resolved all image distortion issues.

Figure 3. Pressure Sensor Suite

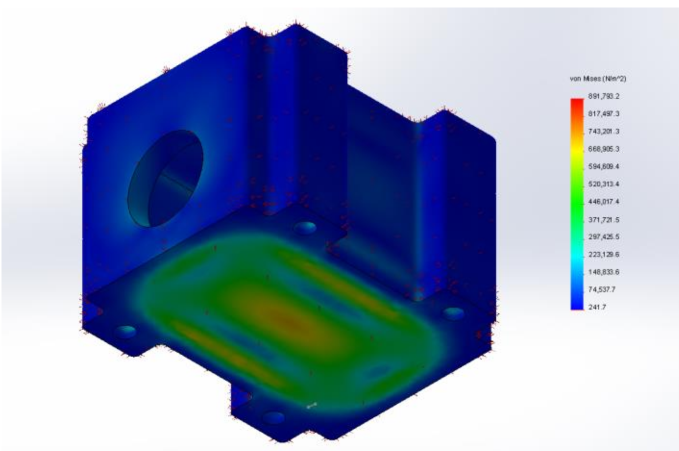

Figure 4. Pressure Sensor Suite Pressure Simulation

DESIGNED MAIN HOUSING

At the beginning of the year the mechanical lead demonstrated a sketch which showed how a cylindrical housing would be able to hold all required electrical components in the current main housing. The cylindrical housing consists of clear

PVC tube and two machined aluminum end caps. The end caps are designed in SolidWorks so that they can be made by a computer numerical control (CNC) machine at a very high level of precision. Aluminum was chosen for the end caps for its relatively low weight in comparison to other metals, as well as the high rate of heat transfer it would provide to the surrounding water. To take advantage of this the ePC is mounted directly to one of the end caps. The other end cap is designed to allow for multiple water-proof Fischer connectors to pass and connect to the electronics inside the housing. There are a total of 16 waterproof connectors intended to pass through the end cap into the housing. The two end caps utilize an O-ring channel so that the O-rings will be able to compress against the side wall of the tube, forming the watertight seal. The advantage of designing the parts in SolidWorks was that it became possible to utilize the simulation and analysis suites available to simulate the maximum pressure that the new main housing could sustain before being destroyed.

PRESSURE SENSOR SUITE

In order for *SubZero* to perceive its own depth, it needs to know the water pressure surrounding it. To do this, *SubZero* uses an Omega PX26 Series Pressure Transducer wired to the Adruino Uno. The pressure transducer is encased in its own housing to reduce the risk of a complete vehicle flooding. Since the pressure transducer has to be exposed to the surrounding water, there is a higher chance of the housing flooding. The housing, Figure 3, is milled from a 5.71 x 6.98 x 8.89 cm aluminum billet. The housing has two holes; one for the water side of the sensor, and the other for a four pin Fischer connector. This connector leads back to the main housing, and in turn the Arduino Uno; again passing through a four pin Fischer connector. While the housing is designed to handle depths of 18 m, Figure 4, it has only been tested to 3.66 m.

INERTIAL MEASUREMENT UNIT

In addition to the pressure sensor and cameras, there is an Inertial Measurement Unit (IMU). The IMU allows the vehicle to its current orientation, velocity, acceleration, and compass bearing. The orientation is based around rotation in the three primary axes and translation in the primary planes: Pitch, Roll, and Yaw, and dead reckoning in the planar motion. s of 18 m, Figure 4, it has only been tested to 3.66 m.

INERTIAL BASED NAVIGATION

Inertial based navigation is a tool that appears to have seen a recent increase in implementation due to recent advances in micro-electromechanical systems (MEMS) technology. The basic principles of its operation are that it uses a combination of three accelerometers and three rate gyroscopes oriented orthogonally to each other, to measure linear and angular accelerations in the three reference frame axes, say *X,Y, Z* although these letters are completely arbitrary. Given that these micro-sensors effectively measure the effects of forces, which by Newton's second law is a mass times its acceleration, it follows that the accelerations may be integrated once to obtain velocity data, and twice to obtain position data. If a reference position and orientation is known, the IMU will output changes to this position(ΔV) and orientation($\Delta\theta$) at some fixed sampling rate in seconds, which may be then used by control software to ascertain how far the vehicle moved from the and how its orientation has changed from the reference orientation. The error in accelerometer and angular rate measurements being computed by the IMU manifests itself as a position drift proportional to the square of time (NEST, 2007). This error may be reduced by combining IMU data with other instruments, although *SubZero*'s missions are such that this is not necessary.

EXOSKELETON

A significant factor in the creation of *SubZero*'s body structure, Figure 5, is the fact that it is almost completely built from the ground up from scratch. The first step was to gather information and brainstorm ideas; research was conducted into the standard types of raw materials that are available for immediate purchase on the market. A local home improvement and construction retailer became the source for many of these materials. This option proved expensive, so an alternate source was sought to provide bulk materials at a discount for a student competition team. A supplier of this kind was not found until later in the competition season. With a good idea of the types of raw materials available, each member of the team was tasked with submitting a rough exoskeleton structure design for the competition. Two of the best submitted designs were combined into a hybrid design combined the best features of both; a standing hexagonal prism design with mounting points for hardware components and housings. Once the structural design had been finalized, the next step was to order the tools and materials to construct a prototype. The prototype exoskeleton allowed the team to see flaws and improve the efficiency of the manufacturing process. For example, it was discovered that brake press bending machines are more accurate and better suited to preform bends, while a Chinese pipe bending press can perform larger-degree bends.

Figure 5. Exoskeleton

The team is permitted access by the school to a small milling machine, this allowed for the both time consuming and accurate placement of each mounting hole on the structure, followed by the four hole patterned joints. When this joint was tested it was found to be unnecessarily robust, and so a two-hole pattern was adopted. It was deemed necessary to build a prototype to validate the structural integrity of the design change, so a single ring was fabricated to perform deformation and impact testing which proved successful. With a solidified manufacturing plan, materials, and practice with the new equipment, the team proceeded to final exoskeleton build. Three types of raw materials are used to construct the final exoskeleton. 1" x 1/16" thick aluminum bar stock, selected for use as light weight housing support structure, possesses the advantage of being inexpensive and easy to form. This is used to secure the camera housings and motors to the main frame. The three hexagonal structural rings are comprised of 1" x 1/8" bar stock. It was calculated that the increase in thickness of the metal provided an increase in stiffness, preventing excessive deflection while under the load of the components. The lower T-beam, of identical thickness to the hexagonal rings, formed the crucial lower structure for the frame and permitted the attachment of lighter supports directly to it, including metal "feet" to support the entire exoskeleton upright on a table. An advantage to the current exoskeleton frame is the versatility to add additional components with minimal changes in weight; it is simple to change component configurations, and the frame itself weighs seven pounds. The use of aluminum rivets was a topic of debate among the team, with good reasons for and against their use. They are lighter and tougher than a bolt of the same material, while being lower in cost. However, compared to bolts which need only washers, lock washers and bolts to be a fastener, rivets require specialized tools and training to properly install. It was decided that the structural benefits outweighed the costs of learning to use and properly install rivets, and so they were implemented as *SubZero*'s primary exoskeleton fasteners.

BALLAST TANKS

The primary design criterion for the ballast tanks is to be inexpensive and easy to build; Polyvinyl chloride (PVC) fits both of these requirements. To build the tanks, two 3" Polyvinyl chloride (PVC) pipes are cut to a length of 78.74 cm and a 3" PVC end cap are glued to the ends of each pipe. This completely seals the tanks against water leaking into the tanks. There is also a third tank made from a 2" PVC pipe and end caps, which has a length of 82.23 cm. The combined positive buoyancy from these three tanks is enough to counteract the weight of the entire vehicle and remain positively buoyant. All three tanks are attached to *SubZero*'s frame using two standard 4" steel hose clamps per tank. As a safety feature, the vehicle is positively buoyant so that it floats to the surface in the event of a power failure. It was through trial and error that the correct pipe lengths were found.

While the current ballast tanks work for the current vehicle, the next version will not use PVC piping, but instead use two thin walled aluminum pipes. The next iteration will be lighter weight, function at a greater depth, and have smaller over all dimensions while still supporting the same amount of weight.

ELECTRICAL SAFETY

The Kill Switch is a magnetic reed switch similar to a door sensor for home security systems. When a magnet is brought close enough, the switch will change states. The switch is hooked up to the second Arduino Uno and a 30 A relay. When the switch changes state, the Arduino sends a signal to the ePC informing the vehicle of which state the motors are now in. The relay is connected between the thruster batteries and the motor drivers, thus when the switch is in the off position, no current can pass through the relay.

A magnetic switch was chosen in order to reduce the number of possible places the vehicle could leak. The magnet is attached to a blaze orange float for the safety diver to easily grab in the event of an emergency. Pulling the float will remove the magnet, which cuts the current to the motor drivers and putting the vehicle in a safe state for the diver to handle.

ELECTRICAL POWER DISTRIBUTION

Initially the power system was housed separately from all the electronics; however this was found to significantly increase the weight of *SubZero* but had little additional benefit. The current fully integrated setup is simple and organized, with the focus of keeping power system maintenance as easy as possible. Motor power is provided by two 14.8 V, 5 AH lithium-polymer batteries connected in parallel. The maximum voltage rating on the motors is 19 V, thus a parallel setup is used to stay under this limit while benefiting from an increased mission time. The power from the batteries moves through the kill switch, and then gets passed on to three motor drivers to individually power the motors. The main power for the computers is a 12 V, 7 AH lithium-iron-phosphate motorcycle battery. Originally all power was to be stepped down from the motor battery setup; however this would have required additional circuitry and thus another point of possible failure. The lithium-iron-phosphate battery powers the ePC, which then powers the rest of the electronics: the Arduino's, IMU, Camera, etc.

PROPULSION

The submarine utilizes six SeaBotix thrusters for maneuverability. They are brushed DC motors encased in a waterproof housing. They are able to produce a peak thrust of 2.9 kg force and are controlled by three Sabertooth motor drivers that communicate by simple serial. These drivers give users the ability to control the rotational speed and direction of the thrusters.

SUBZERO'S PERFORMANCE

The maximum depth achieved by *SubZero* is 3.66 m, which is the deepest part of the test pool. The maximum depth of the camera housings, which are the weakest components, is unknown because the housing is a replication of a field proven idea.

SubZero's vision processing is capable of finding two parallel vertical orange bars floating in the water and then guiding the vehicle towards the center point between the bars.

PLANS FOR NEXT YEAR

Currently each task is split in multiple files; the main logic has its own node, but any supporting logic, such as vision, is in another node in the calculating layer. In order to streamline the code, everything is placed in one file but have the supporting nodes import any necessary functions from the main node. Another plan is to change the ePC to a Hardkernal ODroid U3, which is the same computer ART uses. The U3 has comparable performance specifications as the current ePC, but with a significantly smaller footprint and electrical power draw. A major drawback to the U3 is it only has 2GB of RAM instead of the 4 GB found on the current ePC. Finally, an off the shelf RC flight computer will be placed in *SubZero* to handle vehicle drift and orientation.

Another step forward will be to finish the designed housing. While the current one works, the designed housing is significantly lighter and more compact which will allow for a short vehicle.

Finally, three things need to be done with the camera housings: CAD modeling, computer finite element analysis (stress testing) and finally field testing to verify the computer simulations.

ACKNOWLEDGMENTS

We would like to thank our former academic adviser, Professor Scott Tippens, for putting up with us throughout the year. It has been a rather chaotic year for the team, but he has helped us make as far as we have. We would also like to acknowledge the rest of our team: Douglas Allyson, Albert Cheng, Alex Evans, Andrew Geiman, Lorenza Hill, Taylor Martin, and Nikhil Ollukaren. Without them, we would not have been able to finish our vehicle. Additionally, we would like to acknowledge the SPSU Student Government Association and Alumni Foundation for giving us our budget for the year. Finally, we would like to thank Marietta Diver Supply Co., who allowed us to use their pool facilities and SCUBA equipment for the vehicle testing.

REFERENCES

Naval Engineering Support Team (NEST) (2007). *AUVSI/ONR Engineering Primer Document for the Autonomous Underwater Vehicle (AUV) Team Competition*. AUVSI Foundation. Retrieved October 2, 2014, from http://higherlogicdownload.s3.amazonaws.com/AUVSI/fb9a8da0-2ac8-42d1-a11e-d58c1e158347/UploadedImages/Support_Primer_r1.pdf