Project Report: Autonomous Solar Powered Boat

Word Count: 6726 (Body text as reported by `texcount`)

John Hodge 20518201

Supervisor: Professor Thomas Bräunl
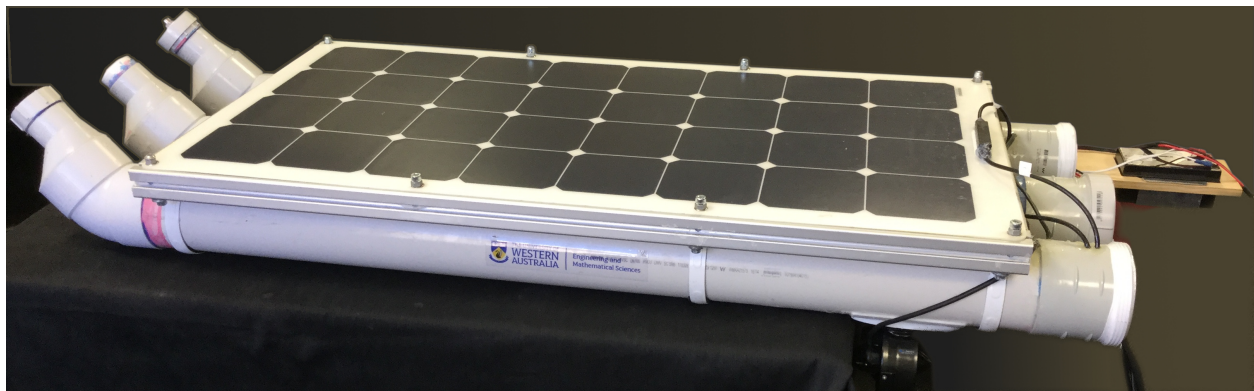
October 28, 2017

Abstract

This project is a side-project to the Autonomous Underwater Vehicle project and aims to create an autonomously guided, near-unlimited range water craft from mostly off-the-shelf components. Inspiration for this came from the SeaCharger craft, which was a small team's hobby project to create a solar powered vessel capable of reaching Hawaii from California (which it eventually succeeded at, and almost managed to reach New Zealand before suffering rudder failure).

This report covers the process of designing the propulsion, power supply, and control systems for the boat, including the specifics of the design and the various dead ends and stalls that occurred. The final design (as of writing) is a differential-drive three hulled boat made from PVC piping, an aluminum top frame, and propelled by two powerful brush-less motor based thruster pods. This design has been tested and can float, propel itself through the water, perform basic navigation between GPS waypoints, and report its position and status via a 3G modem.

Due to time constraints (from various delays during development), although the craft can operate semi-autonomously, it still requires work before it can be sent on a multi-day mission (as the 3G telemetry is not rigorously tested, the control logic needs to be tuned for a sea vessel, and the battery failsafe is not yet implemented). This report concludes with a list of future work, both the required work for the craft to be able to run on its own, and optional extensions for data collection and expanded self-monitoring.

Contents

1 Introduction

The objective of this project (and hence this report to document) is to build a solar powered and autonomous boat using off-the-shelf components, with the vague ultimate goal of producing a craft that can cross an entire ocean on its own. The inspiration for the project came from the SeaCharger project, which was a custom built craft that managed to sail from California, to Hawaii, and onwards to within 300 nautical miles of New Zealand before suffering rudder (and later propeller) failure. According to the SeaCharger website, the aim of that project was "to be the first unmanned surface vehicle (USV) to cross an ocean on solar power alone." [6] and although it came very close to succeeding (within 100 miles of New Zealand controlled waters), it failed at this task. With luck, this project will be able to succeed where it failed. Given the mode in which the SeaCharger failed, one of the design constraints given was to use differential steering (removing the potential for the rudder plane or the rudder linkage to fail). While this does add a second motor to the mix (and the added power draw, risk of motor failure, and risk of power imbalance) motors are vastly less complex than a lightweight rudder and the lever arm required to move it.

1.1 Literature Review

SeaCharger   As stated above, the specified inspiration for this project was the SeaCharger, a solar-powered autonomous boat built primarily by Darren McMillan in California as a hobby project. According to the website, the entire build process (from concept to release from a Californian beach) took over three years, a process that involved design and construction of the hull, thruster, autopilot, and telemetry formats. This fully custom build model allowed the final design to be exceptionally efficient, capable of maintaining a speed of around 1 m/s throughout the day and almost all night (when running of its 500 Watt-hours of Lithium-iron-phosphate batteries).

The hull itself was constructed from fiberglass (moulded over a foam prototype, then sanded and waxed to the final shape), and was a mono-hull design with a large keel and mounting on the top for two 100W solar panels. The thruster was a completely custom build that (according to the designer) took many months of design iteration to complete [5], where the final design was a very low kV (speed per unit voltage) rating brushless motor connected to a speed reduction belt, then to a magnetic coupling that transmitted the rotation to the propeller on the other side of the watertight seal.

The SeaCharger was finally launched from Half Moon Bay, California on the 13th of June 2016, destined for Hawaii, and managed to complete the crossing in 41 days. Upon reaching Hawaii, and lacking an affordable way of getting the craft back to the mainland, Darren chose to set the craft loose again, this time aimed at New Zealand. The craft made decent progress despite some unfavorable currents and a potential point for an autopilot crash (integer overflow in the 32-bit internal timestamp), however as the journey wore on the progress slowed (very likely to an accumulation of fouling) until the rudder failed and the craft became uncontrollable. After drifting for nearly two months, SeaCharger was rescued by the crew of a container ship and finally deposited in Auckland, and now resides in the local maritime museum.

The story of the SeaCharger is definitely due inspiration, but also serves as a guide of what potential issues could arise for a small craft alone on the ocean.

MicroTransAt Competition   The inspiration for the SeaCharger was the MicroTransAt challenge [1] - a race across the Atlantic ocean for autonomous boats. Started in 2010, the challenge has seen thirteen different vessels (all sailboats) attempt to cross the Atlantic ocean (between Ireland and Canada), to varying degrees of success. To this date, no craft registered with the competition has successfully completed the trip (with the closest being the craft "SB Wave" in this year's competition making it over half way from Canada to Ireland before suffering antenna failure, attempting to return home, and being picked up by a fishing boat).

Since all of the vessels that have been registered in the eight years that this competition documents have been sailboats, there isn't much that can gleamed from the vessel designs, but the various notes about the autopilots (and their handling of communication failures) could prove to be useful examples of what to do (and what to avoid).

"ArduBoat" The chosen autopilot board and software - AuduPilot - is primarily designed for airborne vehicles (planes and multi-rotors), but has a variant that is designed for ground vehicles called "Rover". Although it's designed for operation on the ground, there is official support for using the same code to drive both differential and rudder-based water craft. [3] This official support provides a large source of prior experience with the controller upon which to draw when various issues are encountered (with many answers being just a simple web search away).

2 Design Process

The design of the boat revolves around three major fixed aspects, it must run off a single (approximately) 100W solar panel, it must use differential steering (instead of a rudder), and it must have the hull be made out of PVC piping (i.e. not use expensive/complex fiberglass moulding). Most of these restrictions are based on cost (both financial and manufacturing complexity), with the single solar panel being chosen from an interpretation of a comment on the SeaCharger build blog [7] that stated that only one of the two solar panels was needed (the second being for redundancy), and the hull because PVC piping is a simple way to get a water-tight vessel in which to place the control electronics. The drive requirement, however, comes from avoiding the fate that befell the SeaCharger. Without a rudder (and associated linkage) to break, the craft should be less likely to break due to wear (or due to interactions with sea creatures).

Although much of the design work was performed during the first half of the year, the lack of precise measurements for many components meant that there was still a substantial amount of work to be performed even into September, with the first still-water test happening in early October. At the time of writing, the "prototype" design has been tested in controlled conditions, but not exposed to open water conditions.

2.1 Hull and Superstructure

Although the hull and structural design is not the focus of this report, it's worth touching on the design process to provide context for the electronic design (and the progress thereof). The primary point of contention throughout the first half of the project was about the frame holding the hulls together (and supporting the flexible solar panel). The dimensions of this frame (and hence the approximate dimensions of the entire vessel) was fully dictated by the size of the solar panel, but the material was a matter of some contention for the mechanical side of the team. The frame was finally decided to be made out of extruded aluminum section (due to an offer by Critical Room Solutions to construct the frame for free). This provided a very sturdy and versatile frame, but was quite heavy.

The original hull design was two sections of 100mm PVC piping, with bows made from 30 degree angle pipe, and capped screw caps at both ends. However, later density calculations showed that this arrangement (combined with the mass of the frame and solar panel) was too heavy, and such the bow screw caps were replaced with smaller (and lighter) caps, a third (empty) hull added, and some reinforcing removed from the frame.

2.2 Propulsion

The most important question on the electronics side of the project is "what sort of propulsion method should be used", as this dictates how far and how long the boat can go on a day's worth of solar charge. The SeaCharger used a thruster pod constructed from a fiberglass hull, brushless aircraft motor, a magnetic coupling, and a model aircraft propeller. While this arrangement was likely very efficient and well tuned to the vessel, it was also complex to build (and beyond what could be done by students working part-time). The prop was also found to have been lost when the SeaCharger was retrieved.

Conditions on the propulsion selection were a target speed of 1.3m/s (which is the speed of the faster ocean currents), and the cruise power draw has to be sufficiently low to allow it to run for most of the day (ideally the motor would be able to run constantly, but that turns out to require a VERY low power draw with the available storage). Ideally, the hull design would be used to determine the drag (and hence the

required propulsive power to meet the speed target), but at the time the motors were being selected, the hull design was far from complete, so the primary condition was low power draw.

The power draw was originally calculated based on the assumption that the craft could be driving constantly, and that it could store sufficient power to run overnight (and through a dark day). This was determined by using established ratios of panel rating to daily generation [2, p. 4] to get 400Wh/day for our 100W panel. Splitting this full day's generation across 48 hours (accounting for a full day with zero generation) and then splitting between the two motors arrives at an average power draw of 4W per motor, a number which guided motor selection (400Wh/day $\div$ 48H $\div$ 2 = 4.16W))

Various options were considered for the propulsion, ranging from a fully custom build, to embedding a pump in the hull, and finally getting an off-the-shelf thruster pod.

Custom Build   Building a custom thruster pod (or arrangement within the hull) would allow building a solution that would exactly fit the boat's requirements, BUT would also require precise selection of components, constructing a housing, and a method of sealing the shaft as it leaves the hull. This complexity in construction makes the option of a custom build not the best choice for a time-limited project.

Embedded Pump   Another option is to take a waterproof pump, and configure it to push a stream of water out the back. Minimal modification of the pump allows us to assume that it is running to its rated specifications, and avoid extra work. However, this option is a little less efficient overall (due to losses in piping), and runs into unknown situations with controlling the pump speed.

Off-the-shelf   The final (and eventually chosen) option was using the BlueRobotics thruster pods, an off-the-shelf underwater solution intended for underwater vehicles. These are rated to quite high power, but have a peak power efficiency at 6W load. On top of this match for power usage, the motors come with their own speed controllers, have a thick cowling, and are designed to handle minor ingestion of particulate matter.

2.3   Power Storage

All solar-powered devices require some form of power storage, for the boat simplicity was the primary goal. To this end we acquired four 70mm tall/70mm wide sealed lead-acid batteries to provide power during times of darkness, using a number that would allow even distribution of weight across both hulls and between the bow and stern. Lead-acids are very simple to charge (compared to higher-density solutions like lithium-polymer batteries), so have cheap off-the-shelf solar charger units available. These four batteries each provide a rated 3.3amp-hours of storage, for a total of 13.2 amp-hours (or 158.4 Watt-hours), which is substantially less than half the generation capacity of the solar panel. It is possible to just add more batteries to the boat (as they're all arranged in parallel) but batteries are heavy, and the so any added weight needs to be offset by added displacement.

From testing while connected to a bench supply, the entire control equipment draws a passive 300mA at 12 volts (with no motors connected). This passive draw can be used to determine the approximate amount of time the craft will be able to endure with no solar power (without powering off completely and stopping reporting). First check is to check the rated discharge time of the batteries, which in this case is 20 hours (meaning that they're rated to supply $3.3 \div 20 = 0.165A$ for 20 hours), to ensure that the combined draw is not more than the rated draw (as the batteries cannot supply the full capacity when drained faster than this). Since the total 300mA draw is less than half the combined 20 hour average (660mA) we should be able to get close to the full rated capacity of these batteries and continue running in the dark for a total of $13.2 \div 0.3 = 44$ hours (nearly two whole days, note that even in overcast conditions the panel will be producing some charge).

2.4 Autopilot

A boon to development was the availability of the ArduPilot autopilot software and the associated "ArduCopter 2.8" board (which after some searching, appears to be a clone of the open-sourced "ArduPilot Mega" board), including a ATMega2560 microcontroller, magnetic compass, and 6-axis inertial sensor. This avoided needing to both write an entire control suite for the board, and design a board on which to run the control logic. However, documentation on the specifics of this board were few and far between (both due to its clone nature, and the deprecation of the source design), which made determining the minutia of the design challenging (e.g. determining how to access an unused UART for telemetry, and finding connectors for the various sockets).

The ArduPilot software comes with a variant designed for controlling ground vehicles (as opposed to planes and multirotor copters), which has been shown to work well for water-born vehicles. This also supports controlling a skid-steer vehicle, which means that the boat can be driven with just configuration changes (no code changes required). Sadly, the core control logic is where the easy work stops, as the codebase does not support far-roaming vehicles like this project, and the "rover" codebase being slightly out of date compared to most of the supporting PC software. To account for the deficiencies, a several changes had to be made to the ArduPilot source code - most of which were simple once fully investigated, but still took many lengthy compile/test/diagnose cycles.

Remote control To facilitate testing both on the bench and in the water, the craft should be controllable from a ground station (i.e. a PC connected to the telemetry channel). The Rover codebase only supported manually overriding the control signals for the motors, while all of the current PC software sent more abstract control commands. This was fixed by changing the firmware to have basic support for the more abstract message (as this would permit any version of the PC software, instead of requiring a specific build on the tester's machine).

Battery failsafe The software only natively supports "landing" and returning home when the battery reaches low charge, but for a long-range system like the boat the correct behavior is to stop driving and drift until the battery recharges back to a sufficient level. Implementation of this is complicated further by the battery charge level being determined based on reported current draw since the device was powered on (not a level sensor or battery voltage). To correctly handle the case where the battery is charging and discharging custom code needs to be written that determines charge level from the battery voltage alone, and only dives when the battery is (near) full.

Telemetry support As expanded upon below, the stock software only supports a continuous (and high frequency) telemetry feed to the "ground station" (with a minimum update rate of 1Hz), which is too much data for a 3G connection, and far too much for a satellite modem (not to mention in the wrong format). This required some non-trivial additions to the code to allow operation out of range of the high bandwidth radio.
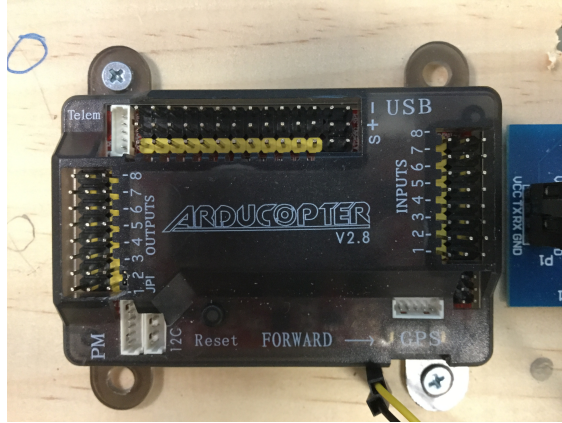
Figure 1: ArduCopter 2.8 Board

2.5   Telemetry

Since the intended use of the ArduCopter board is as a autopilot for a limited-range autonomous vehicle, it comes with an (external) serial radio (using a 433MHz radio to transmit a full-duplex serial stream at 57,600 baud) to communicate with the ground controller. While this radio is perfectly useful for short-range operation, to be able to be tracked and controlled no matter where it is the craft needs to be able to use a long-range radio of some form. The original plan was to use a satellite modem (using the Idridium network) to send infrequent (hourly) position reports, and check for mission updates. However, this idea was exchanged for a 3G modem designed for remote access to RS232-based equipment. This modem removed the need for writing and debugging both a communications protocol, and communicating with the satellite modem, and the cost of reducing craft's control range.

With the new modem available, the existing telemetry code (which uses a protocol called "MAVLink") could be leveraged, and combined with the modem's in-built TCP/IP stack sending the transmitted messages to a remote server, a basic tracking server can be written. The ArduPilot source however still needed to be modified, as the update message rate is only configurable in 1Hz increments (which leads to several hundred megabytes per month of data). To reduce this rate to something more reasonable (e.g. an update every 10s) some custom code was used that disabled the built-in periodic updates and replaced the with a long-period burst of messages.

2.6   Wiring

Tying all of the parts together is two planks of pine with the various controllers screwed and zip-tied down, and holes cut out for the batteries. This allows the electronics to be easily inserted and removed for maintenance and testing, as well as keeping wiring (and circuit boards) away from the bottom of the hull (preventing failure in the unlikely case of a leak). Connecting the various components (autopilot, modem, radio, ESCs, solar controller, ...) is a collection of lightweight signal wire and relatively heavy-gauge power wire. All cables entering the hulls are fully insulated and sealed with a combination of epoxy and silicone. To allow complete removal of the racks without needing to remove this sealing, all of the external cables have distinct and keyed connectors to both allow removal, and to ensure that the cables are always connected correctly.

For signal cabling, the wire gauge is nearly inconsequential (with standard hookup gauges being well sufficient for the currents involved), however for the power rails this is no longer the case. For the batteries, a 10A rated cable is used to connect between the racks and the connection between the hulls (two dual-conductor cables with keyed bullet connectors cut in half and spliced with an insulated cable between the hulls), with 5A rated cable connecting from that bridge to each battery. This is suitable because each of the

motors is rated to 130W at 12V (and should never be run at full power, the autopilot has been configured to never reach that point) and the solar panel controller is soft-fused at 20A (directly limiting the current from/to the batteries). Connecting the solar controller to both the panel and the motor ESCs is some very heavy gauge wire (likely rated to 15A, but is of an unknown exact gauge).

For the motors, this connection is done using 3-pin XLR connectors (more commonly seen in audio equipment), that have the advantage of being both sturdy and rated to the maximum currents used by the motors. These connectors, however, are quite large, and in the first version of the racks the solar panel controller was position such that the motor connectors would sit right on top of it. This turned out to not be a workable arrangement, as the connectors did not fit on top of the solar controller. Thankfully, there was empty space towards the bow where the solar panel controller could be positioned with only a little bit of extra work.
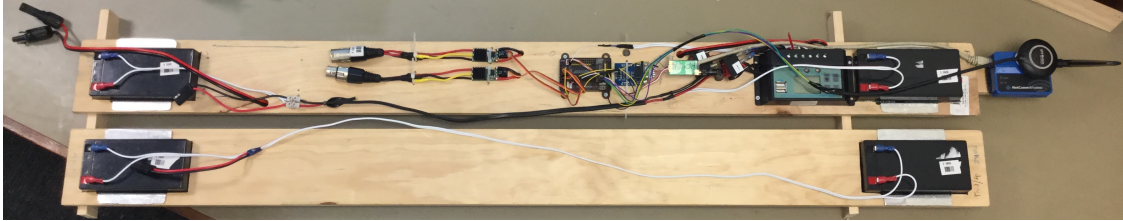

Figure 2: Final electronics racks

3   Design Retrospective

Due to the design process being slightly ad-hoc and iterative, progress was hampered by the process of acquiring parts and construction of the superstructure (which was needed to know where parts would fit and how long the interconnecting cables had to be). Despite the delays, there's a working basis for future expansion completed. There are however still some things that could have been done better the first time around, battery capacity being one of them. The original design process paid no attention to the capacity of the batteries (assuming that the solar panel would be the limiting factor in driving endurance), so ended up with a battery capacity well below a day's worth of generation.

4   Testing

The boat has gone through various stages of testing throughout the design process, from initial tests of the autopilot in isolation, to hull tests in a bathtub, through to planned tests on the Swan river.

4.1   On the bench

Isolated Controller   To test the controller's original configuration (ensuring that the skid-steer mode was working), an external controller (RaspberryPi) was used to generate servo-style signals to connect to the controller's remote-control input channels (before the firmware modification to support PC control was added). An oscilloscope was used to determine that although the signals generated by the Raspberry Pi were very inaccurate (with jitter of around 10%) the controller was able to interpret those and correctly asserted the motor output.

Solar panel controller   At the UWA open day (where the boat was on display, albeit before it was sufficiently sealed to go in the water) the solar panel was tested to ensure that it and the batteries worked with the controller (as this was the first opportunity to place the full system in the sun since it was built). This test appeared successful, with the controller reporting power flow from the panel to the batteries.

Full system bench testing   Since the lab has south-facing windows, there is a lack of sunlight to safely operate the full setup (without draining the batteries). To be able to test the entire system without needing to move to a sunny location, provision was made to connect an bench power supply to the main 12V rail (between the solar panel controller and the ESCs). This had the added benefit of presenting a rough indication of current draw under various conditions.

Motor Testing   As the motors are not designed for running out of the water, only short tests were performed with the motors connected (primarily to determine that the motors did turn, and that they turned in the correct direction). It's worth noting that due to the airflow around the motors out of the water, it was incorrectly determined that they were turning the correct direction in the original tests.

Control Logic Tests   After determining that the motors worked, subsequent control tests could be performed by just observing the reported output values from the controller. This allowed testing the navigation logic by placing waypoints at various locations ahead and behind the vessel's reported position, and watching the motor outputs change.

4.2   UWA Pool - Take 1

The first real test of the entire system was performed on a cold and windy Tuesday morning at the UWA pool (which was apparently set up for water polo, so was relatively clear of obstructions). The test uncovered a few small issues with the control systems, notably that the throttle was reversed when running with remote control. The craft also suffered a slight brown-out when commanded to go to full throttle (likely due to a spike in motor current causing the battery voltage to drop low enough to trigger an autopilot reboot). A test of automatic mode was also performed, which turned out to be a mixed failure... the craft did attempt to navigate, but appeared to have left and right switched which cause it to turn away from its waypoint (and into the water polo goal).

Later investigation in the lab exposed three separate issues that compounded to give the two observed behaviors. The motor output wiring was such that a "forward" signal would turn the propeller backwards (fixed by a simple wiring switch), the motors themselves were plugged in reversed (with the left plugged into the channel used for the right motor, and vice-versa), and the motor output configuration on the autopilot was configured to believe that the channels were "reversed" (meaning that to go forwards, the controller should send a minimum-length PWM signal).

Figure 3: Floating successfully in the pool, tethered to avoid needing to swim

4.3   UWA Pool - Take 2

A second test in the pool (a few weeks later) showed that the previous control issues have been rectified, with the craft working correctly under manual control, and automatic mode being able to (loosely) follow a selection of waypoints. Minor issues were seen with a consistent offset between the GPS and the map (with the map being 2.5 meters offset north-east from the GPS reading), and with the untuned control looped leading to a very wide turning circle and slightly unstable driving (wobbling along the path between the waypoints).

Since this test run was done on a very bright spring day (unlike the previous which was quite overcast), it was possible to confirm that the solar panel could charge the batteries, however the configured maximum throttle was possibly too high to allow the craft to both drive and charge (this was hard to determine, as the battery voltage fluctuates when under variable load).

This test also appeared to show some slight communications problems, both with the 433MHz radio (which had some infrequent disconnections, usually when the craft was at the far side of the pool), and with the 3G modem, which appeared to stop passing data partway through the time on the water (the TCP connection appeared to still be up, but there was no data being transmitted). This issue will have to be properly debugged and fixed before sending the craft fully beyond the range of the 433MHz radio.
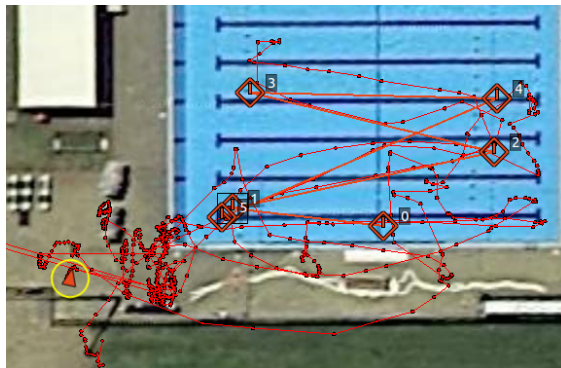

Figure 4: Programmed waypoints and actual track. Note offset between map and GPS

4.4   Swan River (future)

At the time of writing, only closed-water tests have been performed, however it's planned to repeat the same tests at Matilda Bay (on the swan estuary), and potentially let the craft go on a long-distance loop of the beach (taking it out of contact with the ground station). This will test the behavior of the craft in the presence of waves, currents, and how well it handles on longer trips.

4.5   Rottnest Crossing

The final stage of testing (planned for mid-November hopefully) is to set the craft off at the beach and send it to Rottnest island (which is around 23km off the coast) and meet it on the other end. This trip will likely take around 7 hours for the craft to complete (but this depends on its actual cruise speed, which is still to be determined). If it can complete that journey without incident, then it should be able to survive extended periods on the water.

5   Conclusion

The resultant design appears to be a suitable base for future expansion, capable of propelling itself, charging, and following waypoints on the water. There's a simple system setup to log position reports transmitted

over the 3G serial channel, and that same channel can be used to update the waypoint list.

The main remaining question is how far the craft can travel in a day (and before the batteries reach the cutoff point and it stops driving). Since there isn't an easy way to get hard data on the power draw in the water (and the hull drag), the easiest way to determine this is just to watch the battery level when the craft is on the water. The battery failsafe will ensure that the batteries do not completely run dry, and the solar panel should be able to charge the batteries even when under steam.

The craft definitely has room for improvement, but as a prototype build this has been successful.

5.1 Open Tasks

Due to the delays in building a testable craft, at the time of writing there are still some aspects of the design that have not been implemented (or sufficiently tested) that are required before the craft can operate beyond visual range, or continue to operate outside of sunlight.

3G Telemetry   As noticed in the second closed-water tests, the 3G modem seemingly stop passing data after a period. This needs to be investigated and fixed before the craft can be sent out of visual range (as without telemetry reports, finding the vessel would be very difficult)

Battery Cut-off   Custom code still needs to be written to correctly enable/disable the motor based on the available power level, without this the craft will continue to drive until the battery is exhausted (both causing a reboot when the power returns, and eventually damaging the batteries).

Navigation Lighting   According to the international rules on maritime navigation [4] all sea-vessels operating at night or in reduced visibility conditions are required to display suitable lights. For this vessel (which falls under the category of a power-driven craft under 7m in length and with a speed less than 7kts) the lighting requirement is a single all-round white light. To be technically sea-worthy (and reduce the chance of a vessel colliding with it), our vessel will need to be fitted with such a light. Along with the required lighting, displaying a flag or other notice indicating the craft's purpose and contact information would likely be expedient (allowing the craft to be returned in the case of equipment failure).

5.2 Future Work

The boat has quite a bit of potential for future extension, from changes to the control logic through to fitting out as a monitoring platform.

Use attitude information   As the core controller is designed as an aircraft controller, it has a pitch/roll information that is currently unused by the control code. This information could be used to gain assistance from the ocean swell (effectively surfing), or to avoid capsizing by facing into oncoming waves when they get too large, and in the even of a capsize - to turn off the motors and indicate distress to the ground station.

Collate and present telemetry   The current version of the telemetry server just logs the position, and doesn't support sending updates to the boat. A future task is to take this logged information and display it in a user-friendly manner (and allow updating the mission while the craft is at sea).

Alternate power storage   As noted earlier, the boat's current power storage capacity is well underneath the daily generation of the solar panel, which means that the craft will most likely have to reduce power during the night (or cut off all together). Adding more SLA batteries, or even switching to a more complex power solution, would allow longer continuous operation, and may lead to an improvement in range.

Advanced power monitoring The only aspect of power that is currently being monitored is the battery charge level (through a basic resistor divider feeding into an analog input on the microcontroller). Adding other sensors (e.g. battery current, solar panel voltage) would aid in system monitoring (detecting actual motor load, and determining if the solar panel has been fouled). Another potential self-monitoring sensor would be a moisture sensor within the hull, allowing the craft to report if it has sprung a leak (it may not have anything it could do in that case, but such an indication could be useful for forensic proposes).

External sensing equipment As this craft is designed to be able to cross large stretches of water without human interaction, it is the ideal location to place various sensors to measure various aspects of the ocean surface. Options include a basic "CTD" (conductivity, temperature, and depth) sensor, wind speed sensor, or even a full-blown camera. However, with any sensor comes the requirement to extract the recorded data, which ideally would be done by sending over the telemetry channel, which would require integration with the autopilot board (which may be separate to the data collection board).

6   User Manual

The following is a "user"'s guide for the boat as it currently stands, covering the various tasks that a maintainer or user may have to perform that may not be obvious. Software instructions are based on the APMPlanner2 software (which is cross-platform, and was used as the primary method of communicating with the controller) however any MAVLink/ArduPilot compatible software suite should work.

6.1   Building and uploading firmware

The firmware (and some helper tools) are at the time of writing stored on the author's github account (at http://github.com/thepowersgang/fyp-boat), to build the firmware it is suggested that a liunx system be used, with the following packages installed (ubuntu/debian names): `avr-gcc`, `make`, `git`. The firmware makefile first needs to be told to download and prepare the ardupilot source code (using the 'make init' command). Once that is done, the source can be compiled by running 'make', which will apply the project-specific patches to the source and compile it into a .hex file that can be written to the controller. If changes are made to the source code, the command 'make repatch' can be used to re-generate the patch files ready for commit.

To upload the built firmware, any tool capable of writing to an AVR chip via USB will do. The APMPlanner2 software supports doing this under the "Initial Setup" tab, by selecting "Install Firmware" and then "Custom". Once uploaded to the board, the configuration may need to be re-applied in the "Config/Tuning" tab.

6.2   Configuration of the autopilot

The ArduPilot firmware presents a large collection of configurable parameters, only a few of which need to be changed for the boat's operation. A working copy of the configuration is stored in the repository, and can be loaded from the "Config/Tuning" tab, but for completeness's sake a list of the important parameters is below.

- `THR_MAX` Maximum throttle used in auto mode as a percentage of the maximum output value

- `CRUISE_SPEED` Nominal cruising speed (in kilometers per hour)

- `CRUISE_THROTTLE` Throttle level that should maintain cruise speed (percentage of maximum throttle)

- `BATT_VOLT_MULT` Multiplier from the reported voltage on the battery sensor input pin, to the actual battery voltage. (Information only, battery percentage only works with a supported battery controller)

6.3   3G Modem configuration

The 3G modem used in the current design is a NetCommWireless M2M serial modem (NTC-3000-03), which is designed for remotely accessing legacy equipment (but can also be used as a standard USB or serial 3G modem). To support this legacy equipment, it can be configured to connect to/expose a UDP or TCP server and forward serial data between the attached device and the internet-connected server/client.

Configuring this modem is slightly challenging, due to some of the auto-magic ways it works. To reconfigure the modem once it has been configured for forwarding operation, connect at the configured baud rate (for the boat's modem, this is 57600 baud) and type three plus characters in a row, and wait for a second or so. After this, typing "AT" followed by a carriage return should result in "OK" being printed. Once in configuration mode, various options controlling the method of data transfer (UDP/TCP, client/server) and line discipline (baud rate, serial format, EOL/timeout behavior).

The current boat modem should be configured for 57600 baud 8N1 (8 data bits, no parity, 1 stop bit) serial, and connect to a TCP server on port 1516. There should be no EOL character set and the message timeout set to something small (e.g. 100ms). Note, when the baud rate is changed the device should be rebooted (using the save and reboot command), and the final command as part of configuration for use with the autopilot board must be the 'AT+PAD' command (which sets the mode). After this, the modem will reboot and forward all data to the connected server (and the '+++' sequence will be needed to re-enter configuration mode).

6.4   Attaching external power supplies

To attach an external power supply (for testing on the bench), ensure that the batteries are disconnected from the controller (the large battery switch in the center position) and that the main power switch is in the *on* ("II") position (meaning that the bare spade connector isn't connected). Attach the ground cable of the external supply to this bare spade connector and the 12V cable to the red wire poking out from the top of the solar controller. Next, turn on the desk supply and check that the voltage is around 12 volts, then turn the main power switch on the board to the "I" position, the ESC and autopilot boards should flash and start up.

6.5   Powering on and sealing

To power up and seal the boat, connect all of the batteries up to their power rails, connect the battery rails to the controller by selecting "I" on the large three-position switch (the solar panel controller should turn on and show a voltage between 12 and 13.5 volts), then slide the equipment racks into the hulls, attaching the battery bridge and motor connectors as the relevant cables slide into the hull (leave the solar panel connectors for last). Note, when the master rack reaches the end, the 3G modem antenna will need to be raised (this is easiest achieved by inserting a small flat object in the bow and pushing the rack forwards such that the antenna raises as the rack is pushed). Connect the solar panel connections, and then power on the control circuitry by switching the smaller switch at the back to the "II" position (the flashing of various lights should be visible down the tubes).

To complete the sealing, apply fresh plumbing tape to the various threads, and screw the caps back on the hull. The boat is now ready to be placed in the water, and commanded to enter "Auto" mode (this can be achieved either via the 3G modem interface, or via the short-range radio).

6.6   Configuring waypoints

The controller's route can be configured via either USB (directly to the controller) or the 433MHz radio using any MAVLink compatible ground control software. In APMPlanner2, changing waypoints is done using the "Flight Plan" tab, by clicking and dragging existing waypoints or using the buttons at the bottom of the waypoint list. Once the new waypoints are correctly arranged, click "Write" to send the new mission to the controller.

References

[1] The microtransat challenge. http://www.microtransat.org/.

[2] Consumer guide to buying household solar panels. https://www.solarchoice.net.au/blog/wp-content/uploads/Solar-Choice-Clean-Energy-Council-Solar-PV-Consumer-guide.pdf, 2011.

[3] "gmorph". Apm:rover new release also for boats. https://discuss.ardupilot.org/t/apm-rover-new-release-also-for-boats/8308.

[4] International Maritime Organization. *COLREGS - International Regulations for Preventing Collisions at Sea*, 2009.

[5] Damon McMillan. Seacharger build blog: Thruster. http://www.seacharger.com/build-blog/thruster.

[6] Damon McMillan. Seacharger oceangoing autonomous boat. http://www.seacharger.com/.

[7] Damon McMillan. Seacharger build blog: Solar panels arrive. http://www.seacharger.com/build-blog/solar-panels-arrive, May 2015.

Acknowledgements

Definitions

ESC   Electronic Speed Controller - A type of motor controller used for brush-less electric motors, typically takes a servo-style PWM signal (1-2 microsecond pulses with a period of 1-25ms) as input.

PWM   Pulse-Width Modulation - Digital signalling method involving sending periodic logic-high signals of varying duration (at a fixed period) to indicate a range of values.

PVC   PolyVinyl Chloride - Common plastic polymer, typically used in various forms of piping (notably plumbing and electrical).

Wh   Watt-Hour - A measure of electrical energy representing the amount of energy from a device operating at 100W for an hour, equal to 3600 joules.

SLA   Sealed Lead-Acid (battery) - A form of lead-acid battery that is completely sealed (not requiring top-up as many automobile batteries do).

TCP/IP   Transmission Control Protocol / Internet Protocol - Suite of data transmission protocols used to transmit data between two points on the internet.

3G   3rd Generation (cellular network) - A term used to refer to modern mobile telephone networks, capable of arbitrary data transmission as well as voice.