P-Controller as an Expert System for Manoeuvring Rudderless Sail Boats

Naisan Benatar*, Omer Qadir†, Jenny Owen‡, Paul Baxter§

*Department of Computer Science, University of Nottingham. nxb@cs.nott.ac.uk
‡Department of Computer Science, University of York. jowen@cs.york.ac.uk
†Department of Electronics, University of York. oq500@ohm.york.ac.uk
§School of Systems Engineering, University of Reading. p.e.baxter@reading.ac.uk

Abstract

Sailing robots provide a useful platform for investigations into autonomous robotics. Typically, sail boats involve one or more sails and a rudder. Recently, a two fixed-sail boat with no rudder has been proposed. In this paper, a P-controller is proposed for the control of this sail boat. It is shown that the heading of the boat can be controlled using this simple controller, where the update rate is an important factor. Upon this basis, extensions for increasing the fault tolerance of the system are discussed as a means of increasing the autonomy of the sail boat - an important aim in the use of sail boats as autonomous marine research platforms.

I. Introduction

The exhibition of autonomy, or more particularly of autonomous behaviours, is a desired goal for much work in robotics. Autonomy in this sense implies the ability of the robotic agent to operate for extended periods of time without human control. This in turn implies the ability to react to the environment, adapt to changes and be able to maintain the integrity of the agent. The design of autonomous sailing boats provides an ideal opportunity for the exploration of these issues. Being designed to operate for lengthy periods of time at sea, these robotic agents must be able to operate without explicit human control. It should be able to complete its set task (or tasks), whilst maintaining energy levels in an environment (the sea) which is highly variable.

Unmanned sailing boats can be expected to be used for tasks such as as oceanography, wildlife monitoring, and weather data collection. Such tasks can place many kinds of strains on a boat which is expected to spend long amounts of time in what is a hazardous environment. If the control system for the sailing mechanism is designed such that it uses minimal power, solar energy can be used to power the motors that drive the sails and various other electronics. Since sailing uses a renewable energy source (namely wind) a sail boat is very well suited for such tasks. In addition, the ability to overcome various hardware malfunctions, be they temporary (such as an overheated motor) or permanent (such as a broken sail), is desirable given the aim of autonomous operation over extended periods.

Typically, sailing boats operate with a sail and a rudder. With this design both sail and rudder must be used in coordination to achieve the desired course. Recently, a two fixed sail boat with no no rudder has been proposed. Control of the boat must be achieved by coordinating the two sails for propulsion and turning. This also introduces a possibility for fault tolerance which is not possible in the sail-and-rudder setup, as errors in one sail may, to a certain extent at least, be compensated for by the other sail. In this paper, a computational architecture for the control of this sail boat setup is proposed. Extensions to this controller are also discussed which aim to take advantage of this inherent possibility for fault tolerance. Details of the hardware design may be found in the following section and in [1].

Section II provides an overview of the novel dual-sail and rudderless boat. Section III describes the structure of the experiment and contains a brief outline of the controller that was implemented. Section IV describes the experimental results and section V discusses them. Finally, extensions to this basic controller are discussed in section VII, particularly with regards to improving the fault tolerance of the system.

II. Background

Sailing is an activity that requires a great deal of training. To become an expert and direct the boat in the desired direction, human beings typically need to go through a great deal of practice. Good sailors develop a *feel* for the boat which are subtle markers in the way the boat responds to inputs.

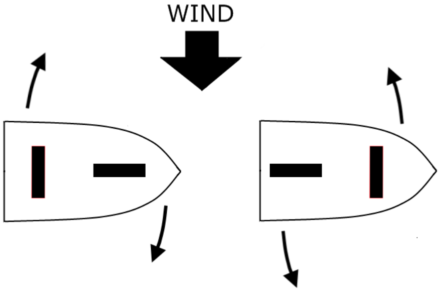

Fig. 1. Turning the sail boat using sails only. (a) turning the boat away from the wind by increasing the resistance to the wind of the front sail, and decreasing that of the rear sail; (b) the opposite sail configuration leads to the boat turning into the wind

Most sailors would be hard-pressed to describe the whole process in terms of logic rules and would resort to generalisations although texts such as [2] attempt to do this.

Whereas most boat consist of one or more sails and a rudder, the sail boat that is used for the current study uses two rigid sails and no rudder. The control of this boat must thus be achieved through the manipulation of the sail angles only. In fig. 1, two scenarios are displayed which illustrate the means of turning the boat either away from or in to the wind, respectively. Furthermore, in order to drive the boat forwards, the sails must be aligned with respect to the wind such that a forward force is exerted (fig. 2). The control of the angle and propulsion of may thus be achieved. These principles are summarised in the following points, and are formalised in section III(C):

- Forward motion is optimized by rotating the sails so that they present approximately 22° as angle of attack to the wind.
- Heading is modified by altering the *centre of effort* of the boat. This is also achieved by rotating the sails. Fig. 1 illustrates.
- Sailing directly into the wind is not possible – while sailing directly downwind is extremely difficult. As illustrated in fig. 3, both the upper and lower shaded areas indicate boat bearings with respect to the wind which must be avoided. The upper region is 90° while the lower one is 60°.

Static control mechanisms are not sufficient for the control of sail boats due to the dynamic nature of the environment where factors like winds, tides and current can all change suddenly and without warning. It is also possible for components to malfunction, for example the motors on the sails overheating due to excessive sail movement. The final control system implemented on the boat needs to take this in to account and include some means of fault tolerance to handle it.

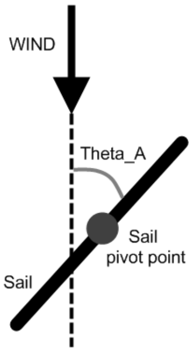

Fig. 2. The desired angle of both sails with respect to the wind for forward movement of the sail boat, the angle of attack ($Theta_A$ = $22deg$)

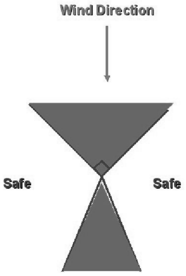

Fig. 3. Based on the wind which directions are possible for the boat to move in

III. Experiment Setup

A. Task and Performance Metric

As an initial proof of concept, the objective is for the craft to maintain a heading relative to wind, for a given period of time. We have chosen 5 minutes as a starting point, however if the boat gets stuck, for example in a corner of the lake, then we will terminate the run prematurely.

If the desired heading with respect to absolute North is denoted by H_d and the actual heading with respect to absolute North is denoted by H_a, then the error in heading (θ_e) can be obtained by equation 1.

$$\theta_e = H_d - H_a \tag{1}$$

θ_e can be used as a performance metric to minimize the cumulative error over a given period of time.

To compare separate experiments against each other where different numbers of readings could have been taken we will use the mean value of θ_e over the length of the experiment. The value of θ_e will be stored to a log file during each run of the main control loop. We will label this value $\overline{\theta_e}$, with values closer to 0 being preferable.

B. Hardware Architecture

The power constraints present in the boat mean that there are limited hardware resources available. A PIC micro-controller is used to monitor the sensors and control motor outputs. The boat includes a wind-sensor which returns wind direction with respect to the boat along with the wind speed. A compass which returns the current heading with respect to absolute north is also present. The boat uses solar panels to recharge on-board batteries which drive the PIC, the sensors and sail motors. There are 2 sails each with its own individual motor.

Although the software design is suitable for implementation on a PIC and can easily fit on the micro-controller, the code is actually implemented on an EEPC for ease of programming and testing as well as providing storage for logging. Commands are sent between the PIC and the EEPC on a RS-232 connection using a simple ASCII protocol. This eases experimentation by allowing modification of the control code in the field and helps logging as the logged data does not need to be transferred elsewhere - which can be problematic with unreliable wireless connections.

The EEPC is placed inside the boat along with a wifi access point that provides allow wireless communication with the shore.

A block design of the layout is illustrated in fig. 4.

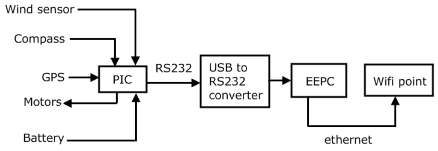

Fig. 4. Hardware Architecture

C. Software Architecture

An overall block level architecture of the software is presented in Fig. 5.

The TCP server provides a communication path with a client, running on a laptop on the shore, which allows modification of parameters on the fly so that multiple experiments can be performed quickly. The logging system is used during off-line evaluation. The Hardware box represents APIs that are used to read from sensors or actuate motors. The COMMS box performs hardware abstraction and integrates all the other modules. The P-controller implements the equations 2 and 3 to rotate the sails.

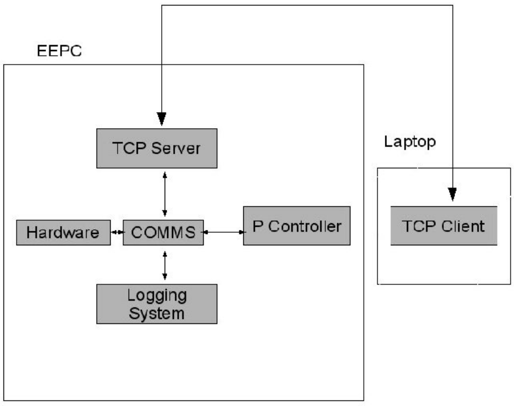

Fig. 5. Software Architecture

$$M_f = W - (22 + (P \times \theta_e)) - H_a \qquad (2)$$

$$M_b = W - (22 - (P \times \theta_e)) - H_a \qquad (3)$$

M_f and M_b are the angles for the front and the back motors respectively. P is the gain for the P-Controller. θ_e and H_a are as defined by equation 1. W is the wind direction with respect to absolute North. θ_e, M_b and M_f need to be corrected for roll-over errors so that they are a continuous variable and do not jump from -359 to 0. The value 22 was determined as the angle of the sail with respect to the wind fig. 2 to enable forward movement of the boat, provided that the boat bearing with respect to the wind was in the safe regions (as shown in fig. 3). In order to isolate errors in wind sensor measurements, it is also possible to measure the wind direction independently at the test site. This can then be transmitted to the control code using the TCP server and wireless communication. Similarly, the destination heading may also be transmitted using the TCP server.

IV. Results

At the test site the wind direction was measured as being at a bearing of 310° relative to magnetic North. This was transmitted to the control code and fixed for the duration of the experiment due to software problems with the sensor which we did not have time to correct. Destination heading H_d was arbitrarily chosen as 220° relative to magnetic north. These bearings are shown in fig. 6 superimposed above our test lake.

In the first experiment P is set to be 0.8 and the motors are updated at a frequency of 0.83Hz fig. 7 shows the results of plotting the heading error θ_e (equation 1) over time. The downward spikes in the graph are measurement errors in the sensor readings

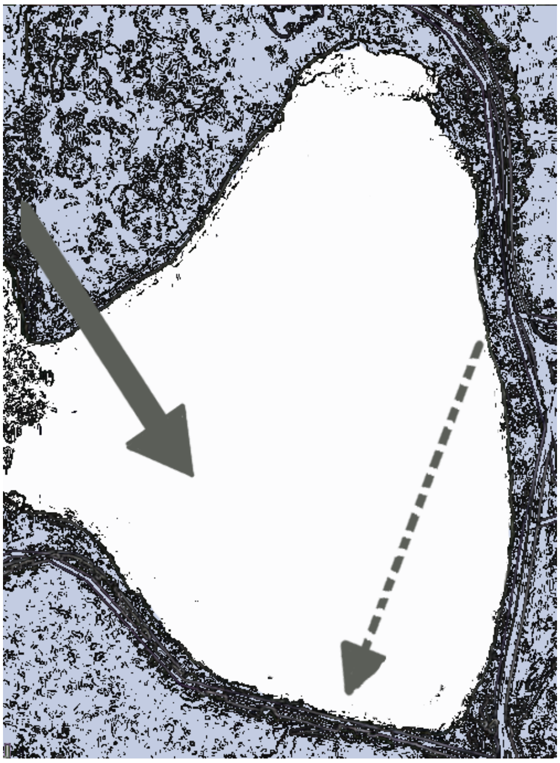

Fig. 6. The wind direction and desired bearing relative to our testing lake. The solid arrow indicates wind direction and the dashed arrow our desired heading.

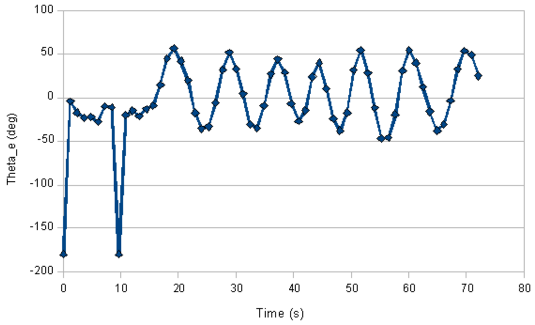

Fig. 7. The heading error as a function of time, with an update rate of 0.83Hz. A positive θ_e indicates the boat is clockwise of its desired heading and a negative θ_e is anti-clockwise. For this experiment P was 0.8.

and do not reflect the actual heading of the boat at that time. For the purposes of working out $\overline{\theta_e}$ they will be ignored.

It can be seen that the boat is oscillating around the desired heading but actually spends only a small amount of time pointing in the correct direction. The value of $\overline{\theta_e}$ for this experiment was $-7.65°$. The observed behaviour of the boat under these conditions was that it moved quickly when perpendicular to wind (on course) but would often turn towards the wind, slowing the boat down to a halt. Once stopped in this position, the force of the waves on the boat's hull were enough to turn the boat and push it into a region where it could harness the wind and reset its bearing.

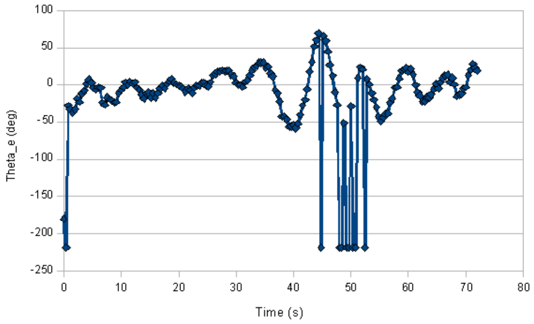

Fig. 8. P is 0.8 and each sensor reading is 0.4s apart.

In the second experiment P was maintained at 0.8 but the frequency of the motor position update was increased to 2.5Hz. The results for θ_e were observed, again over a 74 second time period and are presented in fig. 8.

The same oscillating behaviour was noted in this experiment, but it was much less severe, as is reflected in the graph. The value of $\overline{\theta_e}$ was $0.995°$.

Other experiments were also conducted where the value of P was varied, but this resulted in very little difference in behaviour.

V. Discussion

The experiments have shown that there are a number of factors which contribute to the behaviour of the boat - we hypothesize that the actuation frequency is one of the more important parameter of the motors that control sail movement. Our results indicate that a larger data set will be necessary to prove this point, and as such further experimental runs will be required.

This frequency is greatly affected by the fact that the control code was running on the EEPC. The EEPC had to communicate through a USB port to a RS232 converter which would transmit the data to the PIC which would then actuate the motors.

The robot platform used is an unstable system which needs to balance the forces exerted on two sails so that they are equal about the boat's centre of lateral rotation. It uses feedback from the environment to decide the next action. If the boat is unable to react to environmental changes in a timely manner, then the forces on the sails become unbalanced and the boat must compensate even more for the error in heading.

There were many imperfections in the physical design of the sail boat which were ignored in the experiment and whose effects turned out not to be negligible. For example the P-controller does not cater for the difference in sizes between the front and the rear sail, consequently the larger front sail is pushed slightly harder than the rear sail causing the boat to turn away from the wind. The boat then compensates for this by turning into the wind but the slow motor reaction meant that it does not always manage to correct itself before it is pointing too close to the wind. When this happens the boat slows to a halt and it needs to wait until it is pushed by the waves before it can catch the wind again. The correction in this manner is slow enough that the sail motors have plenty of time to react to the environmental feedback from its compass and can set itself on the right path. In addition, imperfections in the construction of the hull meant that the centre of lateral motion of the boat was not in the middle which also aggravated the oscillating behaviour.

Figs. 7 and 8 both reflect this behaviour since before the oscillation begins, the boat is pointing away from the wind and once the boat corrects for this, it begins a cycle of overcompensation for heading errors. With shorter time period between motor actuation an improvement can be seen but over time still tended to face towards the wind. Once it reached some threshold where it could no longer catch the wind, the oscillation process started. However with the increased motor command rate it was able to stop itself from turning as far from the goal as with the first experiment - also shown by comparisons of the values of $\overline{\theta_e}$ which are significantly smaller in the second experiment than the first.

Despite the oscillations, in both experiments the boat manages to move in the correct general direction, even if it is not necessarily on the exact desired bearing. However oscillations still represent a significant reduction in the efficiency of the boat - as it will slow down and require motor movement for correction - further work is needed in order to minimise these behaviour.

VI. Conclusions

This paper describes the motivation and implementation of a basic expert control system that is able to improve the sailing performance of a wind driven robot by the using sensor data to control changes in position to front and rear sails in order to minimise deviation from the given course. The effect of this is that the robot maintains a bearing for the duration of the experiment and while the chosen performance metric is far from optimal it is considerably better than anticipated given the time frame of the work.

We then discuss the future directions of the work including how the control system may be augmented with additional intelligence in order to give a more reactive system while also providing an efficient means of fault tolerance and to a degree, fault recovery. Such intelligent systems could include well understood and relatively mature approaches such as fuzzy logic possibly coupled with biologically inspired systems such as AIS and AES.

VII. Controller Extensions

Due to the investigatory nature of this paper there are many avenues of improvement that can be pursued to develop a fully autonomous sailing robot. Changes are required to take into account the imperfections in the boat – most notable – differences in sail area of the front and rear sails and asymmetrical weight distribution (centre of lateral motion). While this could be performed in software by compensating with weights when calculating the angles, a physical change would provide a more stable sailing platform overall.

Current existing work using fuzzy controllers such as [3], [4] and [5] has been shown to be quite suitable for such behaviour so that rudderless sail control can be used to augment the fault tolerance of such systems.

Another method that has been proposed is that of reactive and adaptive robotic systems, such as artificial neural networks (ANN's) or behaviour-based controllers that have been augmented with analogues of biological immune or endocrine systems in the hope of adding additional behavioural flexibility and fault tolerance as shown in [6] which applies such an algorithm to a security robot, [7] which explores the use of idiotypic selection in behaviour based robotics and [8] where immune inspired methods are applied to damage response and robotic self maintenance.

Using one of the above techniques is liable to provide a more robust and better performing robot, however the work required to implement and test such a system is not trivial so a choice will have to be made as to which is the one to use in future experiments.

Acknowledgements

We thank Mark Neal for use of his equipment and inspiration for the project and Colin Sauzé for help

with interfacing with the hardware.

References

[1] M. Neal. A hardware proof of concept of a sailing robot for ocean observation. *Oceanic Engineering, IEEE Journal of*, 31(2):462–469, April 2006.

[2] Bob Bond. *The Handbook Of Sailing*. Pelham Books, 2nd revision 2 edition, 1996.

[3] R. Stelzer, T. Proll, and R.I. John. Fuzzy logic control system for autonomous sailboats. In *Proc. IEEE International Fuzzy Systems Conference FUZZ-IEEE 2007*, pages 1–6, 2007.

[4] T.W. Vaneck. Fuzzy guidance controller for an autonomous boat. *Control Systems Magazine, IEEE*, 17(2):43–51, Apr 1997.

[5] Jaime Salon Jaime Abril and Oscar Calvo. Fuzzy control of a sailboat. *International Journal of Approximate Reasoning*, 16:359–375, 1997.

[6] Robert Oates, Julie Greensmith, Uwe Aickelin, Jonathan M. Garibaldi, and Graham Kendall. The application of a dendritic cell algorithm to a robotic classifier. In *Proceedings of the 6th International Conference on Artificial Immune Systems (ICARIS2007), Lecture Notes in Computer Science 4628, Santos, Brazil*, pages 204–215, 2007.

[7] Amanda Whitbrook, Uwe Aickelin, and Jonathan M. Garibaldi. Idiotypic immune networks in mobile robot control. *IEEE Transactions on Systems, Man and Cybernetics, Part B*, 37 (6):1581–1598, 2007.

[8] Mark Neal, Jan Feyereisl, Rosario Rascuna, and Xiaolei Wang. Dont touch me: Robot autonomy using an artificial innate immune. In *Proceedings of the 5th International Conference on Artificial Immune*, pages 349–361, 2006.