BRCORP
TECHNOLOGY | INNOVATION | INDUSTRY

5th World Conference on
Applied Sciences, Engineering & Technology
02-04 June 2016, HCMUT, Vietnam

Research on design and heading control of remotely operated vehicle (ROV)

DUY ANH NGUYEN[1], HOANG TON NGUYEN[1], PHUNG HUNG NGUYEN[2]
[1]Dept. of Mechatronics, Faculty of Mechanical Engineering, HCM City University of Technology, Vietnam
[2]Faculty of Navigation, HCM City University of Transport, Viet Nam
Email: duyanhnguyen@hcmut.edu.vn

Abstract: ROVs (Remotely Operated Vehicle) play an important role in underwater activities. One of the most important problems in controlling ROV is the movement of ROV in the horizontal plane because of high complexity and 6-degree-of-freedom motion. In this paper, we focus on designing and prototyping a mini-ROV platform, which had four thrusters for horizontal movement and two thrusters for vertical movement. Then, the control algorithms will be built in order to control the motions of the platform on a horizontal plane with the following criteria: The desired speed must be ranging from 0.2 to 0.5 m/s and the maximum tolerance of ROVs direction, which is checked using a compass and a 6-axis gyroscope, must be lower than 2 degrees. Finally, the results of some experiments are shown to analyse the effectiveness of the algorithms.

Keywords: *Diving robot, PID controller, Remotely Operated Vehicle (ROV), Heading control*

Introduction:

Remotely operated underwater vehicles (ROVs) play an important role in a number of underwater missions. The potential uses for ROVs include: Scientific (oceanography, geology, geophysics, etc.), environmental (waste disposal monitoring, wetland surveillance, etc.), commercial (oil and gas, submerged cables, harbors, etc.), military (tactical information gathering, smart weapons, etc.) and other applications where their endurance, economy and safety can replace human divers.

Underwater vehicles that use differential thrust for surge and yaw motion control have the advantage of increased maneuverability. Unfortunately, such vehicles usually don't have thrusters or actuators to control the lateral movements. Hence, they fall into the underactuated vehicle category. The purpose of this research is to design and prototype a mini-ROV model could move flexibly in the horizontal plane. The platform is required to have some basic features: It has to be able to reach a predefined depth and move in the desired direction (forward, backward, left or right) with the speed ranging from 0.2 to 0.5 m/s; the maximum difference between the desired and the actual direction must be about 2^o using a compass and a 6-axis gyroscope. In this paper, the control algorithms are built for the motions of platform on a horizontal plane and some experiments are conducted to check the effectiveness of the control algorithms.

Materials and Methods:

In this project, the experimental method was used to test the control algorithms. A platform of the ROV was designed and manufactured to conduct experiments and evaluate the effectiveness of the control algorithms. The ROV's model consists of a platform, a computer control interface and a tether. The platform had four thrusters for horizontal movement and two thrusters for vertical movement; all of horizontal thrusters were arranged so that the α angle was 45^o (see Figure 1). To move along the surge (X) axis, the platform used either the (1) (2) or (3) (4) thrusters; to move along the sway (Y) axis, the ROV used either the (1) (4) or (2) (3) thrusters and two vertical thrusters were used to help it move along the heave (Z) axis. The platform had a pressure sensor to get the information on the depth of the robot, a compass for measuring angle around the yaw axis and a 6-axis gyroscope was used to find out the angle around the pitch axis. The added weight and buoyancy distribution make sure that the ROV always return back to the zero pitch and zero roll state. RS485 communication was used for transmitting signal between the platform and the computer. The authors also researched and manufactured a waterproof structure for DC motors with encoder feedback. This structure enables the electric motor to work normally underwater.

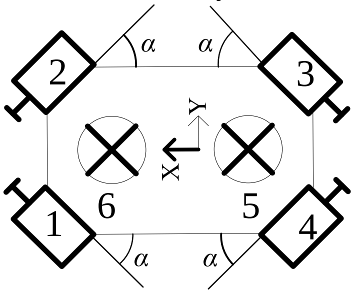


Fig. 1. ROV's principle diagram.

The control algorithm for the platform was built after the mechanical model was completed. The working principle of this entire system is described as follows: Firstly, the platform need to keep in a state of zero pitch. This means the rotation angle around the surge (Y) axis of the platform always close to 0^o. A PID-Pitch controller was used to control the angle around the Y axis to keep it near zero. The input of the PID-Pitch controller is the rotation angle of the ROV in respect to the Y axis and the output is (%) PWM that increase or decrease accordingly for the 2 vertical thrusters.

After the platform achieves stability (state of zero pitch), it can then proceed to go underwater, move along the predefined direction (forward, backward, left or right). In this paper, the motions of the platform in the horizontal plane are the focus. Hence, the yaw angle is an important information. When the robot moves forward along predefined direction, it means the platform needs to move forward and keep its yaw angle as a predefined constant. A PID-Yaw controller was used to control the platform's movement so that the difference between the reference direction and the actual direction is near zero. The input of the PID-Yaw controller is the error between the reference yaw angle and the actual yaw angle; the output is (%) PWM that increase or decrease for 2 horizontal thrusters to move the platform forward. The control algorithms for the backward, left and right motion along the predefined direction are also built similar to this algorithm.

In summary, the control system includes a controller to balance the ROV in horizontal plane and other controllers so that ROV can move forward, backward, left or right along the predefined direction. Control algorithm will decide the thruster's operations in order for ROV to move. Below is the scheme of the PID controllers and the flowchart of control algorithm for the system:

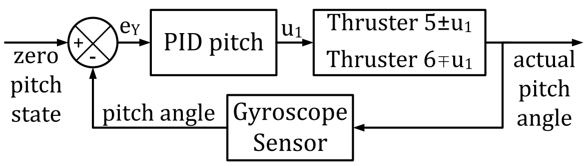


Fig. 2. PID controller keeps the robot in balance.

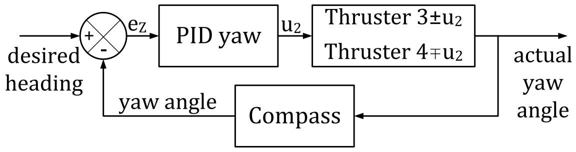


Fig. 3. PID controller to ROV moves forward along the predefined direction.

The main program for the control system:

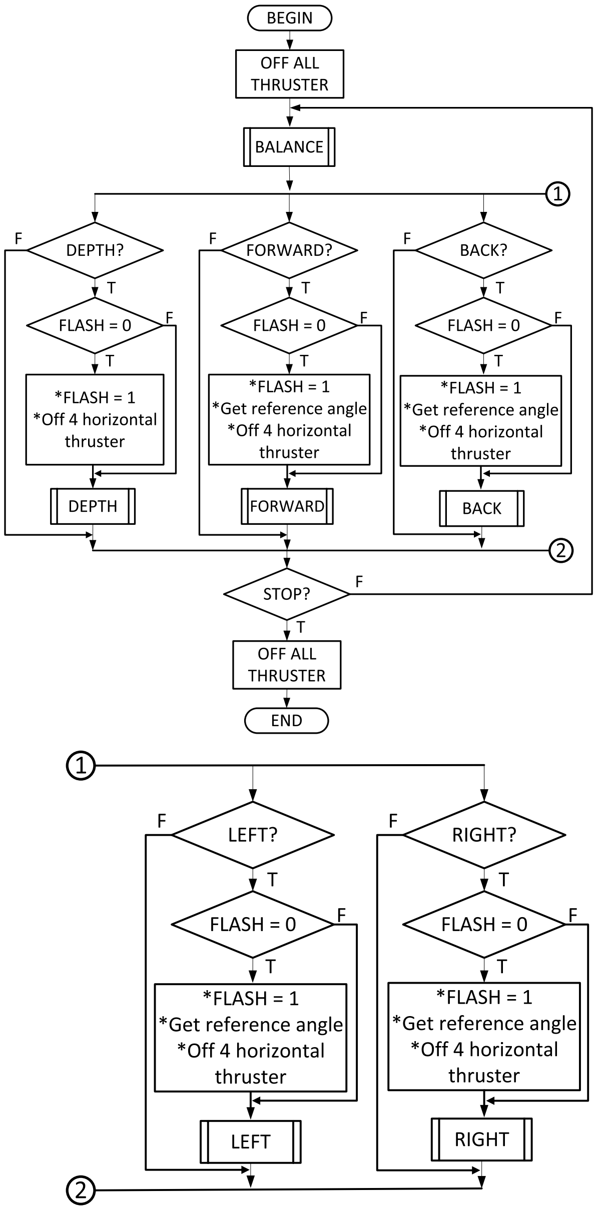


Fig. 4. ROV's main algorithm

The subroutine for ROV move along the z axis, and move forward, backward, left or right along the certain direction, which will be presented below:

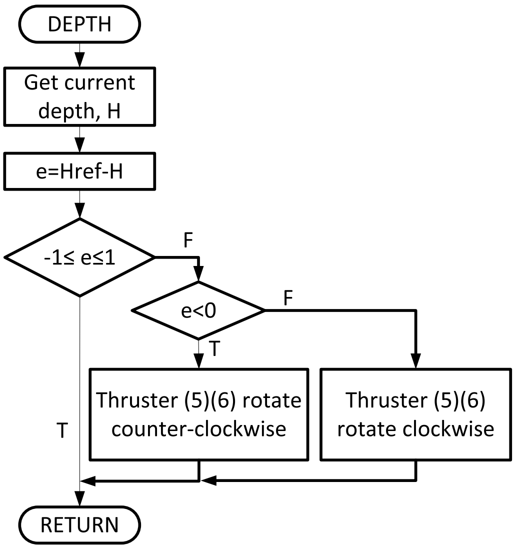


Fig. 5. Algorithm for ROV moves along heave axis

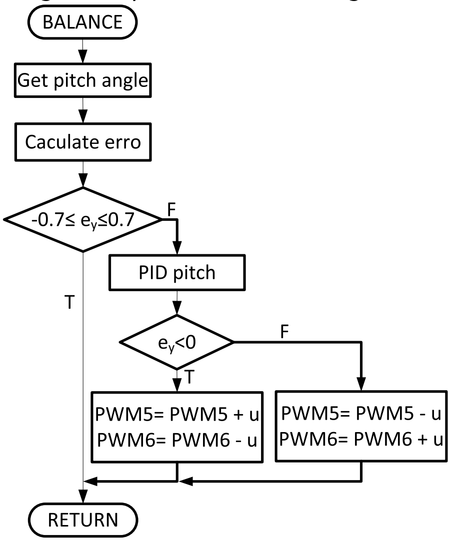


Fig. 6. Algorithm for ROV keeps balance

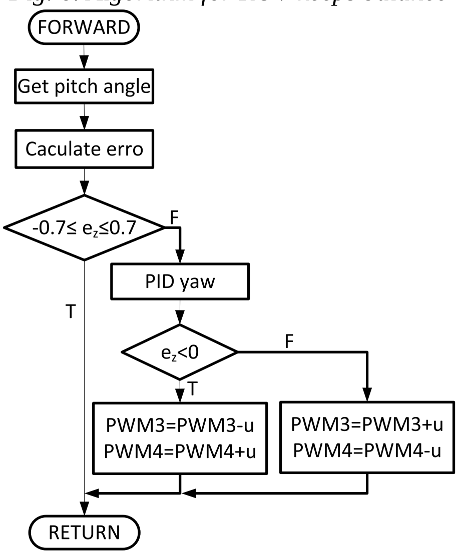


Fig. 7. Algorithm for ROV move forward along the surge axis

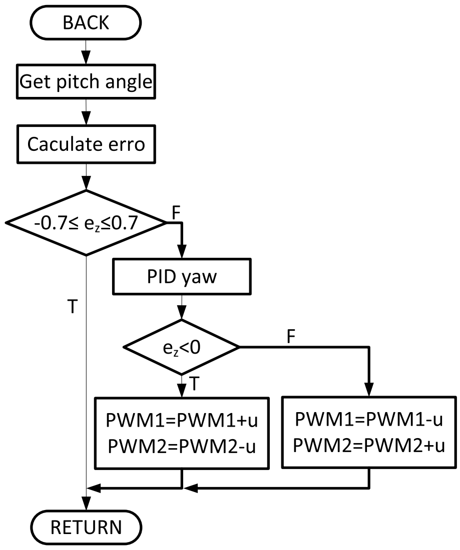


Fig. 8. Algorithm for ROV move back along the surge axis

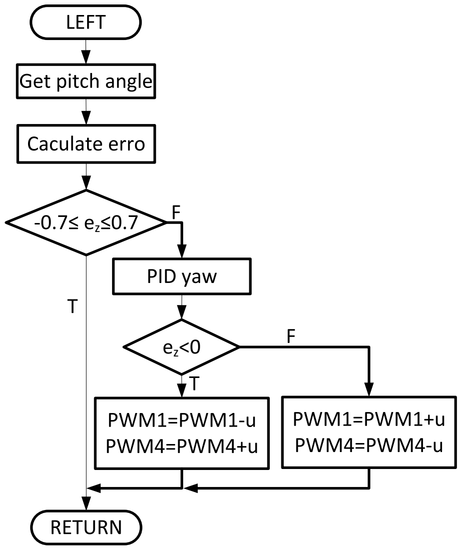


Fig. 9. Algorithm for ROV moves along the positive direction of the sway axis

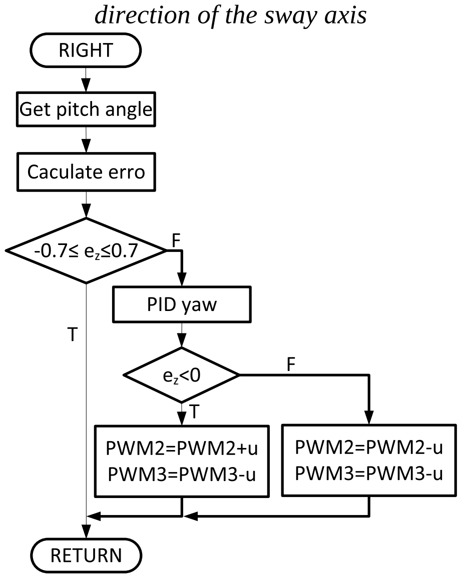


Fig. 10. Algorithm for ROV moves along the negative direction of the sway axis

The experiments were conducted to evaluate the algorithms and during the entire tests, the desired depth is set to 0.5 m below water surface. When the platform is balance, it would move forward, backward, left and right along the predefined direction. The experimental results will be presented in the following section.

Results and Discussion:

The experiment results for testing control algorithms (algorithm for ROV moving forward, backward, left or right along the predefined direction) are shown in this section. In these experiments, the platform was initially located at the origin with a 30^o yaw angle. The platform would move forward (or backward or left or right) and keep the yaw angle near the reference yaw angle (30^o). The sensor data from the platform will be read once every 50ms.

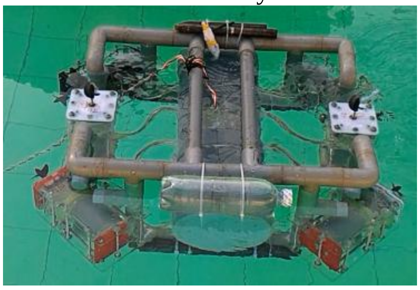
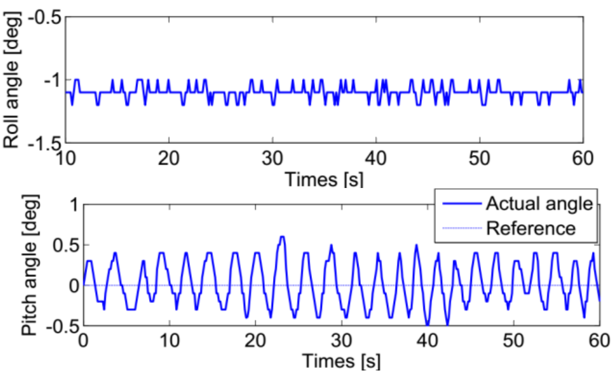


Fig. 11. Experimental Platform

The data of roll angle, pitch angle and yaw angle were collected to evaluate the balance of the platform underwater and the capability of the robot to move along the predefined direction in a horizontal plane.

The following figures display the experiment results of the control algorithms when the platform move forward, backward, left and right along the predefined direction (desired yaw angle was 30^o) with the same PID controller parameters. The first graph displays roll angle of platform. The second graph displays the reference pitch angle (dashed line) and the actual pitch angle (solid line). The third graph displays the reference direction (dashed line) and the actual yaw angle (solid line) in degree. The PID controller parameters are tuned by trial and error and used in all of the experiments.

Figure 12 display the results for forward motion along the predefined direction with the yaw angle of 30^o:



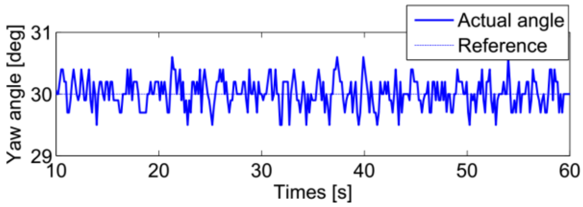


Fig. 12. Results for forward motion

The practice results showed that the roll angle values during the experiment were around -1.1^o. The pitch angle values were closed to the zero pitch state with the maximum error was 0.6^o. The pitch angle values converged to the desired direction (30 degree) with the maximum error was 0.6 degree.

Figure 13 display the results for backward motion along the predefined direction with the yaw angle of 30^o. It showed that the roll angle values were around -1^o, the pitch angle values were closed to 0^o (maximum error 0.6^o) and the yaw angle values converged to 30^o (maximum error 0.85^o).

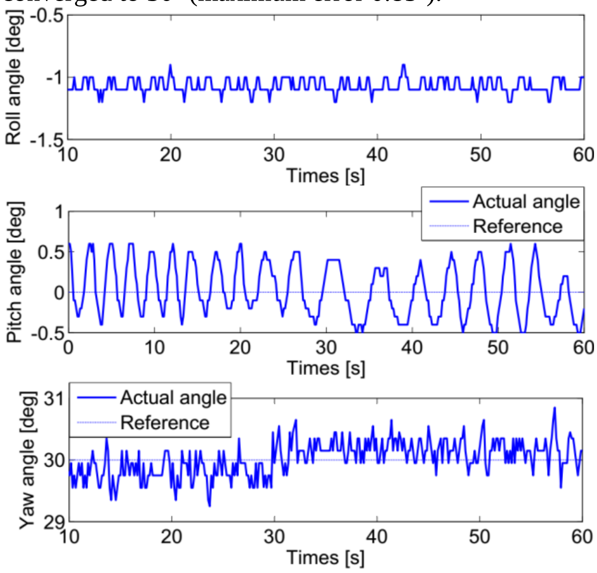
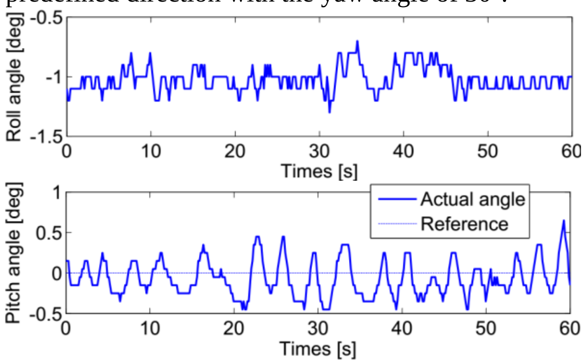


Fig. 13. Results for backward motion

The above results showed that the controllers were working well in controlling the platform to move along the surge axis. The practice for the motion left and right of robot were similarly conducted with the above PID parameters.

Figure 14 display the results for left motion along the predefined direction with the yaw angle of 30^o:



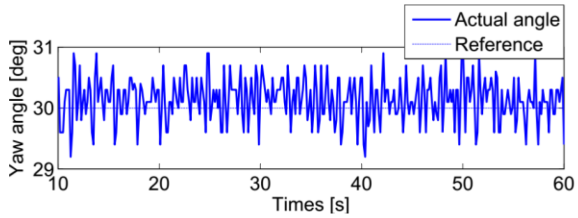


Fig. 14. Results for left motion

The experimental results showed that the roll angle values were around -1^{o}, the pitch angle values were closed to 0^{o} (maximum error 0.65^{o}) and the pitch angle values converged to 30^{o} (maximum error 1.3^{o}).

Figure 15 display the results for right motion along the predefined direction with the yaw angle of 30^{o}:

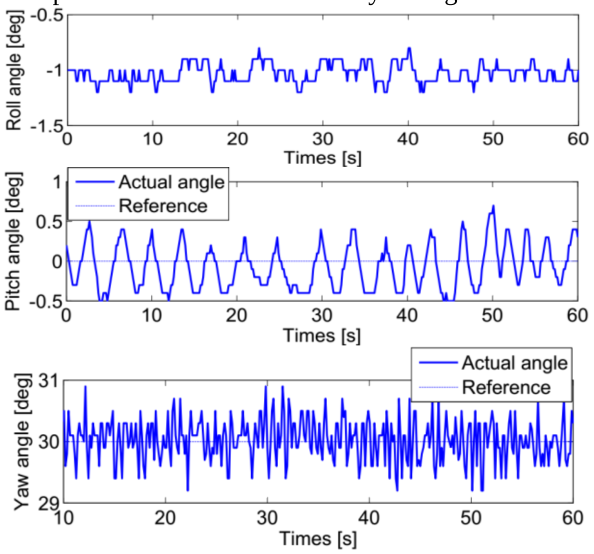


Fig. 15. Results for right motion

The experimental results showed that the roll angle values were around -1^{o}, the pitch angle values were closed to 0^{o} (maximum error 0.7^{o}) and the pitch angle values converged to 30^{o} (maximum error 1.3^{o}).

After experimenting the motion of the robot in 4 directions (forward, backward, left and right), the errors of pitch angles were smaller than 1^{o}, it means the balancing algorithms for the robot at a desired depth works well. In practice with the above PID parameters, the maximum error of the motion in the surge axis is acceptable (less than 1^{o}) while the motions in the sway axis is not (the PID controller parameters are not relevant). It shows that the water resistance in the surge and the sway is different. Therefore, the PID controller parameters for the motions in the surge and the sway should be different.

Conclusion:

The model was verified by experiments for the design of control algorithms. The results experiment showed that the controller was working well in controlling the robot to move forward and backward. The error was within a certain range that is acceptable (1^{o}). But the PID controller parameters for left and right motion along the predefined direction need to be tuned so that the platform could move along the sway axis better.

Currently, the controller heavily relies on the accuracy of the onboard compass to obtain information about the yaw angle. Unfortunately, the onboard compass exhibits high nonlinearity measuring yaw angle and is extremely vulnerable to its working environment. Therefore, filters and more accurate compass will be used for future work. Besides, to control ROV betters, another combination between PID and other intelligent or advanced controllers are preferred.

References:

[1] Robert D.Christ, Robert L.Wemli, "The ROV Manual-A User Guide for Remotely Operated vehicles", BUTTERWORTH HEINEMANN, ISBN: 978-0-08-098288-5, Oct. 2013.
[2] Huynh Thai Hoang – Advanced Control Theory, University of Technology, Ho Chi Minh City, 2014.
[3] M. Zhuang and D.P. Atherton, Automatic tuning of optimum PID controllers, IEEE Proceedings, Part D, 140(3):216-224, 1993.
[4] J.G. Ziegler and N.B. Nichols, Optimum settings for automatic controllers. Trans. ASME, 64:759-768, 1942.