ISSS Journal of Micro and Smart Systems
https://doi.org/10.1007/s41683-018-0031-5

ORIGINAL ARTICLE

Demonstration of control and inspection by circumferentially guided ultrasonic waves using novel Remotely Operated Underwater Vehicle

Shagun Agarwal[1] · Tanuj Jhunjhunwala[1] · NT Saikiran[1] · Prabhu Rajagopal[1]

Received: 18 May 2018 / Revised: 28 September 2018 / Accepted: 8 October 2018

Abstract
This article presents a cost efficient semi-autonomous remotely operated underwater vehicle for inspecting water-submerged pipeline networks using circumferential guided ultrasonic waves. Robot navigation is implemented using inertial measurement unit and pressure sensor, with motion control achieved through proportional–integral–differential controller specially tuned to take angled turns. The robot is equipped with pneumatic grippers for clamping on-to a pipe and uses thrusters to move along its length. The paper describes the electronics and control design of this mechanism and presents results from practical experimental trials and ultrasonic measurements.

Keywords Non destructive evaluation · ROV · Ultrasonic testing

Introduction

Underwater pipeline inspection operations are precarious and require highly skilled professionals. Regular inspections of this kind are necessary for submerged pipelines, large tank floors, etc. for monitoring their structural health. While basic visual observation using remote cameras is very useful, novel methods using magnetic probes, pressure testing and ultrasonic probes are being developed to predict structural failures well ahead through signal processing.

One of the more popular methods for pipeline inspections is pigging, where a appropriately sized spherical or cylindrical device, called a pipeline inspection gauge (or 'pig'), is propelled through the interior of the pipe through manipulation of the flow pressure or mechanical pulling (Bubar 2011). These devices can serve multiple purpose of cleaning and sealing the pipeline along with inspection using on-board sensors. However, pigs need to be inserted into the pipeline through custom made launching stations and removed from the pipeline through receiver stations. This operation not only exposes the pipeline to surrounding atmosphere, it is also a safety concern (Combe and Hair 2011). These devices are generally not compatible with pipelines using multiple internal diameters, or having sharp bends, nor can they move between multiple branches of the pipeline networks. Such location are structurally most sensitive and need regular check-up. Also, any pig equipped with sensors for inspection would require an on-board power source and would not be able to communicate with outside world once inserted into the pipeline, making its operation period very limited. Hence there is much interest in development of alternate methods for inspecting submerged pipeline networks.

Set in this context, this paper presents a tethered unmanned remotely operated underwater vehicle (ROV). ROVs are robots capable of swimming freely underwater and perform various tasks as directed by the operator. They are widely implemented on mobile platforms for collecting oceanographic and geophysical data. The applications of underwater ROVs mainly include visual inspection of ship hull, seismic surveys, and so on, refer Fossen (1994). This ROV is equipped with sensors for navigation, control and external ultrasonic inspection of submerged pipes. Our concept vehicle explores the combined processing power of a cost effective microprocessor Raspberry-Pi (Raspberry Pi Foundation 2012) and basic micro-controller Arduino Mega (Barrett 2010). It has the ability to dive and manoeuvre to an underwater pipeline network, hover over the required pipe, use a novel mechanical design of

This work was supported by Innovative Students Projects - Indian Institute of Technology Madras.

✉ Prabhu Rajagopal
prajagopal@iitm.ac.in

[1] Indian Institute of Technology Madras, Chennai, India

Published online: 23 October 2018

pneumatic gripper to move along the pipe's axis using its surge thruster. It is also equipped with cameras providing live feed for visual inspection of the pipes and PID controller. The control loops have been specially tuned to make angled turns such as L-turns and Z-Turns for smooth movement over branches in a pipeline network. The fault inspection is carried out using circumferential guided ultrasonic waves. They are generated due to superimposition of bulk ultrasonic waves from the solid boundaries and can hence travel along the circumference of a pipe (Rose and Nagy 2000; Na and Kundu 2002). The presence of a structural fault will scatter the wave causing a change in the received signal which can be used for fault detection.

This paper describes the mechanical design of the hull and the gripper mechanism. It presents the details of the electronics and communication framework for handling the large amount of data using affordable electronics. The paper also presents results from sensor-based PID controller tuned for angled turns. It finally presents the method and results for fault detection in a pipe using ultrasonics guided waves from probes mounted on this ROV.

Vehicle design

Mechanical design of the hull

The mechanical design of the ROV is minimalistic and aimed at reducing complexity in control through symmetry. The central hull is a hollow aluminium cylinder of length 0.42 m, internal diameter of 0.19 m and outer diameter of 0.20 m. It is capped on either side by 3D printed hemispherical domes of radius 0.12 m, giving it low drag along pertinent direction of movement. The water sealing is achieved through face sealing using two O-rings, with tested ability to withstand pressures up-to 50 m of water depth. Figure 1a, b show the front and side view of the ROV. It has 3 translational degrees of freedom (dof), i.e., surge along x-axis, sway along y-axis and heave along z-axis; and 3 rotational dof, i.e., roll about x-axis, pitch about y-axis and yaw about z-axis. However, out of these only 4 dof are controllable through the action of the actuators—surge, heave, pitch and yaw.

The ROV is loaded with dead weights on the bottom side of the hull to lower its centre of gravity to make it roll stable. Overall, the ROV is slightly positively buoyant, i.e., it would normally float on water surface. This has been done so that it would resurface automatically in case of any failure. However, even in this floating condition the ROV would have enough weight to submerge all its thrusters underwater. The front and back heave thrusters are mounted on the central hull end plates and are covered by the domes. The domes, however, have been designed to allow free passage of vertical flow of water. The right and left surge thrusters have been mounted on the central hull through a pre-deigned mount.

The thrusters used are Seabotix BTD150 rotary blade propellers which have operating voltage of 18 V DC and provide thrust of 14.3 N in forward and 13.8 N in reverse directions. The ROV has 2 transparent camera housings made from acrylic cylinder and attached to central hull using face sealing. The top camera housing has provision for mounting a pressure sensor so as to minimise the effect on the readings from thrusters. An ultrasonic probe holder made from standard propylene is mounted in the front to give direct access to the pipe surface for inspection. The slot for the probe is tilted at 15.5° to ensure correct angle of incidence for generation of circumferentially guided ultrasonic waves. The description regarding the choice of this angle is presented in Sect. 4.1.

Design of the pipe gripper mechanism

The pipe gripper actuation is a four bar mechanism actuated by a pneumatic piston. Two such grippers are mounted on the bottom surface of the hull of the ROV. Each group of gripper consists of two grippers which are controlled by two pistons respectively. Figure 2 shows the mechanism with its links and pin joints. To obtain the best optimised solution (for the given hull design), the following constraints and assumptions were taken:

- r_1 varies between 0 and 215 mm
- θ_1 varies from 0° to 180°
- $\Delta\theta 1 = 18°$. The deviation is sufficient for clamping and unclamping.
- $r_2 = 150$ mm. It limited by the standard available piston.
- $\Delta r_2 = 40$ mm. It is limited by the standard available stroke length of piston
- The clearance angle between the hull face and the piston is positive

The value of r_3 can is hence a function of r_1 and θ_1, which is given by

$$r_3(r_1, \theta_1) = \sqrt{\frac{\cos(\theta_1)(r_1^2 - (r_2 + \Delta r_2)^2) - \cos(\theta_1 + \Delta\theta_1)(r_1^2 - r_2^2)}{\cos(\theta_1 + \Delta\theta_1) - \cos(\theta_1)}} \quad (1)$$

$$\cos(\theta_2) = \frac{r_2^2 + r_3^2 - r_1^2}{2 r_2 r_3} \quad (2)$$

Based on the mentioned ranges for r_1 and θ_1, the optimal values of the variables required to satisfy the design

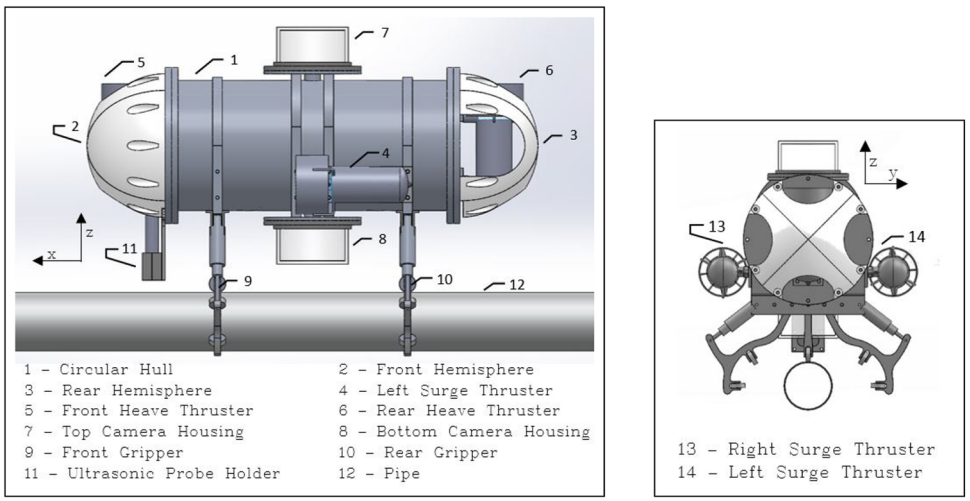

Fig. 1 Schematic of the hull of novel pipeline external inspection ROV

(a) side view **(b)** Front view

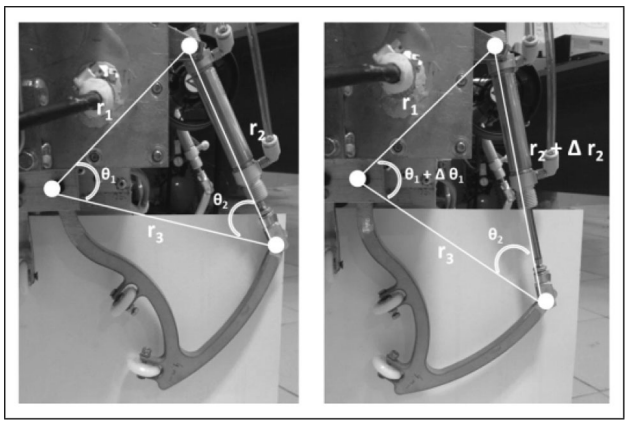

Fig. 2 Pipe gripper mechanism with its links and pin joints in open and close state

specification for this ROV were found to be $r_1 = 136\,\text{mm}$, $\theta_1 = 57°$, $r_3 = 173\,\text{mm}$ and $\theta_2 = 49°$.

The overall pneumatic circuit had 4 double acting pistons which were controlled by 2 solenoid valves. A compressed air reservoir with a volume of 2 litres was used. Initially, the reservoir will be filled with compressed air up to a pressure of 7 bars. The exhaust of the solenoid valve open into the main hull. Therefore, the actuation of the grippers will lead to gradual drop in the reservoir pressure and a marginal increase in the hull pressure. Using the ideal gas law and assuming an isothermal and isochoric process, it was found that the reservoir can support 22 cycles of clamping and un-clamping at 1 m depth of submersion. With the increase in water depth, the pressure required from the clamping operation will increase, thus reducing the possible number of cycles. This gripper mechanism was tested on a model pipe underwater as shown in Fig. 3, where the number of clamping cycles were confirmed.

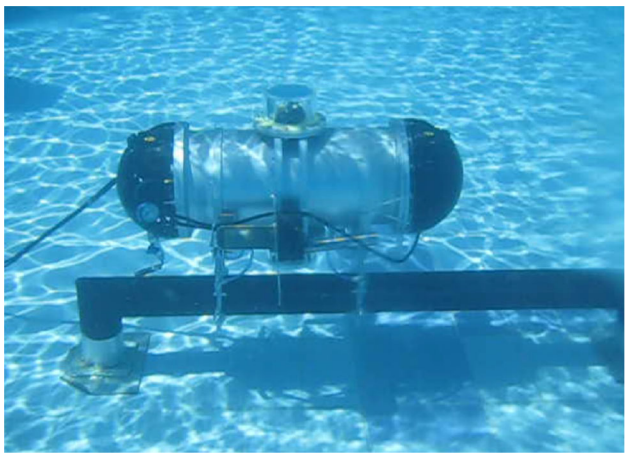

Fig. 3 Snapshot of the ROV gripper holding onto a fixed pipe for inspection

Electronics

The electronics in the ROV can broadly be classified into power circuit and logic circuit. The power distribution is controlled by custom made *Execution Board* which takes an input of 24 V DC and converts it to 12 V DC, 5 V DC and 3.3 V DC using three LM2678 Voltage Regulators. The thrusters are powered using VNH5019A-E fully integrated H-bridge motor driver chip each of which is soldered on individual detachable motor driver board thus allowing easy replacement in case of any overload failure. Each motor driver board is equipped with ACS712 current sensor and a resistors based voltage sensor to control power output to each thruster. In addition to this, the *Execution Board* is equipped with opto-transistor circuits for controlling solenoid valves for pneumatic grippers and has inputs for auxiliary sensors like pressure-based depth sensor.

Logic board is the control centre of the ROV. It uses micro-controller Arduino Mega for controlling the motor drivers and solenoid valves. It receives inputs from depth sensor and current & voltage sensors on each motor driver and relays the necessary information to Raspberry-Pi through serial communication. Depth sensor used is pressure-based Omega engineering PX309-200A5V. It can measure absolute pressure of up-to 14 bar and hence water-depth of up-to 130 m.

Logic board also houses Raspberry-Pi B+, a 700 MHz micro-processor with 512 MB of RAM, which is the central processing and communication unit. It communicates with the off-board system via Ethernet cable, receives the necessary instructions and relays real-time information regarding ROV heading, speed, sensors' values, actuators' status and on-board camera feed. It is connected with Arduino Mega, Inertial Measurement Unit (IMU), two USB cameras and ultrasonic pulser through USB ports. The heading of the ROV is measured using inertial measurement unit VectorNav VN-100S. It provides real-time angular acceleration, linear accelerations and orientations about the three axes. Visual feedback is obtained using two USB cameras, one mounted in the upper camera-housing to obtain the view ahead of the ROV from top and second one is mounted in the bottom camera-housing to obtain a direct view at the surface being accessed by the ultrasonic probe.

The two main boards are placed one above the other, mounted on a board housing and then connected to all the sensors. This housing is 3D printed and prevents relative motion of the electronics inside the central hull. All the sensors, actuators and Ethernet connections have been located on one side to allow easy access while opening the hull as shown in Fig. 4.

The user input for ROV navigation is given using a regular USB gaming joystick connected to the off-board system. Operating system used in off-board system is Debian-based Ubuntu and on Raspberry-Pi is Debian-based Raspbian.

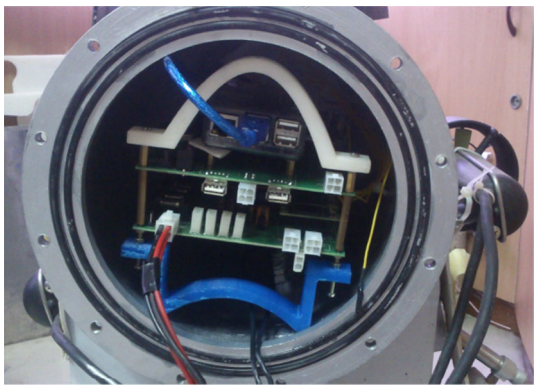

Fig. 4 Snapshot of board housing placed inside central hull

Communication

The ROV is executing a number of parallel programs, for running multiple independent communication channels as shown in Fig. 5, hence utilizing the full processing potential of the micro-processor. All data exchange over Ethernet cable is done using Lightweight Communications and Marshalling (LCM) (Huang and Olson 2010), which is a data assembling and passing tool, compatible with multiple programming languages, allows custom data structures, has low latency and works in real-time. LCM is also used for data exchange between the parallel programs running on Raspberry-Pi.

Live video streaming from on-board cameras to off-board system is done using GStreamer (Toma et al. 2011), which is a popular open-source media handling library, through Ethernet cable over channel *E4*. User instructions regarding heading set-points and sensor setting from off-board system to Raspberry-Pi are sent over *E1* using LCM. Off-board system receives real-time information from Raspberry-Pi regarding ROV heading, thruster power and sensor readings over *E2* and ultrasonic readings over *E3*, both using LCM.

IMU communicates with Raspberry-Pi over channel *U3* using FTDI-VCP protocol, whereas the ultrasonic probe does so using FTDI-D2XX protocol over *U4*, both through

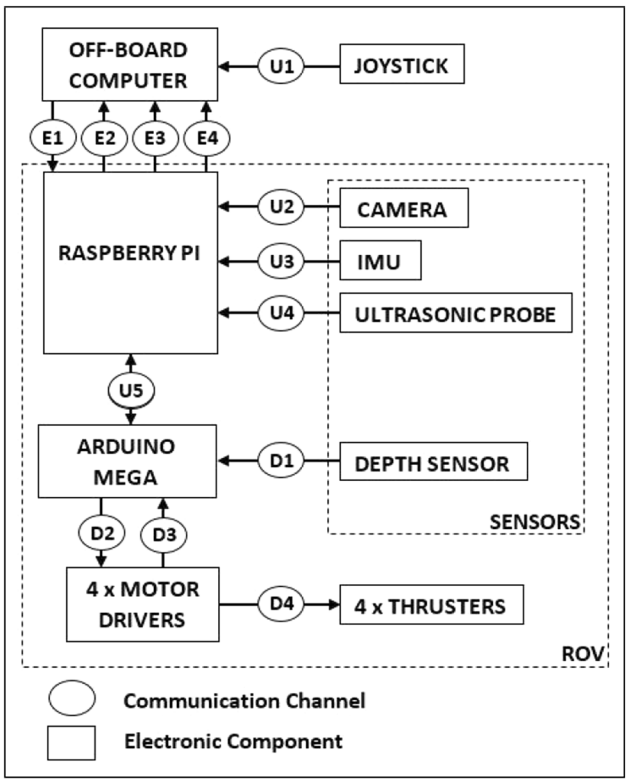

Fig. 5 Schematic of parallel communication channels implemented for pipescan ROV

USB ports. All remaining USB port-based communications are done using serial protocol.

Control

The equations for 6-DOF rigid body dynamics of an underwater ROV as given in Fossen (1994, Ch. 2) are

$$M\dot{v} + C(v)v + D(v)v + g(\eta) = \tau_E + \tau \tag{3}$$

where $v = [u, v, w, p, q, r]^{\mathrm{T}}$, with $[u, v, w]$ as linear and $[p, q, r]$ as angular velocities of the body-fixed origin about the body fixed X, Y, Z axes, respectively. The 6×6 inertia matix $M = M_{\mathrm{RB}} + M_A$, where M_{RB} is the rigid body inertia and M_A is the added inertia matrix due to the hydrodynamic forces. Similarly, the 6×6 centripetal and Coriolis force matrix $C(v) = C_{\mathrm{RB}}(v) + C_A(v)$, where C_{RB} and C_A are respectively the rigid body and hydrodynamic components. $D(v)_{[6\times6]}$ is the total hydrodynamic damping matrix, and $g(\eta)_{[6\times1]}$ correspond to the restoring forces and moments due to weight and buoyancy. The environment forces and moments due to waves, wind and currents are given by τ_E and the actuator forces and moments for this ROV with 4-thrusters would correspond to $\tau = [\tau_X, 0, \tau_Z, 0, \tau_Q, \tau_R]^T$.

To build a complete control model for the ROV, one would have to determine the large number of unknown coefficients in $M, C(v), D(v), \tau_E$ through rigorous experimental, analytical (through strip theory) or numerical analysis. This work does not focus on the development of the dynamic model. The objective here is to demonstrate the concept of using traditional ultrasonic sensors on existing ROV models for defect detection. Therefore, to achieve this objective this work has implemented a minimalistic ROV control based purely on the error between set-point and measured-signal with error minimisation using a PID algorithm as described below. This algorithm would not provide adequate control in presence of strong environmental forces, such as cross-currents; however, it is sufficient for demonstration of defect detection along with open-source communication and affordable electronics architecture.

If $u(t)$ is the system response, the complete control law using PID (Ogata 1990) can be written as

$$u(t) = K_P\, e(t) + K_I \int_{t-\Delta t}^{t} e(t)\,\mathrm{d}t + K_D \frac{\mathrm{d}}{\mathrm{d}t} e(t) \tag{4}$$

where K_P, K_I and K_D are coefficients corresponding to the proportional, integral and derivative terms and $e(t)$ is the error at time t between the control variable and its set-point value. Figure 6 shows the block diagram of control loop executed in this ROV. There are a total of 8 separate PID control loops running, 4 for controlling the yaw, surge, pitch and heave degrees of freedom (dof); and 4 for controlling the power output to each thruster. The dof control loops are executed in Raspberry-Pi and thruster control loops in Arduino Mega.

Raspberry-Pi collects the information regarding current ROV heading, depth and user instructions for heading and depth. Based on this information, the 4 dof PID loops evaluate the thrust needed to be generated by each of the 4 thrusters and relay this information to Arduino Mega. The micro-controller receives the power output status to each thruster through the current and voltage sensors on each motor driver and based on the thrust requirements given by Raspberry-Pi it evaluates the PWM (pulse width modulation) for each motor controller. This finally produces the required motion for the ROV which is fed-back using the IMU and depth sensors; hence completing the overall control loop.

The tuning for all PID loops is done manually through multiple trials. Some popular alternate tuning methods are Ziegler–Nicholas method and Cohen-Coon method. For manual tuning, initially a P-only control executed to obtain a suitable value for K_P. This control will have a steady state error which is then corrected by a suitable K_I value the $I(t)$ term. The oscillations are then minimised by K_D parameter in the $D(t)$ term. The control loop for thrusters were tuned first to minimise response time as due to the high underwater resistance an overshoot is very unlikely. Next, the surge and yaw control loops are tuned with moderate response time with emphasis on minimising any oscillations. Finally, the pitch and heave control loops are tuned which need to account for the large upward buoyancy force due to the positively buoyant nature of the ROV.

Results and discussion

Defect detection using circumferentially guided ultrasonic waves

Ultrasonic guided waves arise from superposition of bulk ultrasound reflected internally from structural boundaries (Rose and Nagy 2000). Guided waves have attracted wide attention for rapid defect screening in recent years due to ability for scanning large sections from a single transducer location. For pipe inspection, circumferentially guided waves are attractive for scanning the entire pipe cross-section from a single probe location.

The experiment for defect inspection was carried out on a submerged mild steel pipe of outer diameter 90 mm and thickness 1 mm. The frequency of the ultrasonic signal was 1 Mhz with probe focal length of 20.3 mm. From Fig. 7, obtained using DISPERSE software (Pavlakovic et al. 1997), it can be observed that the S_0 mode is less dispersive

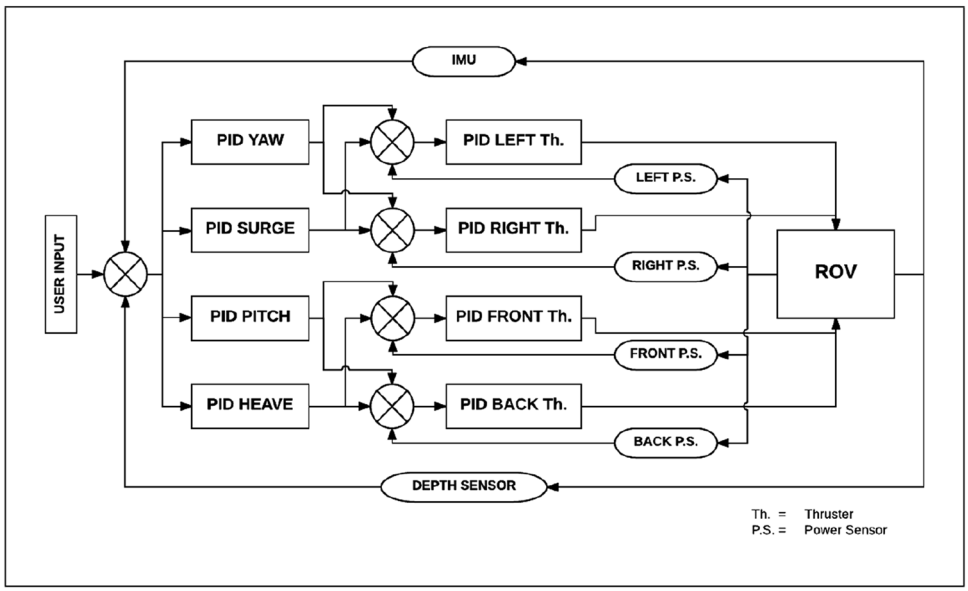

Fig. 6 Block diagram of control loop in the pipescan ROV

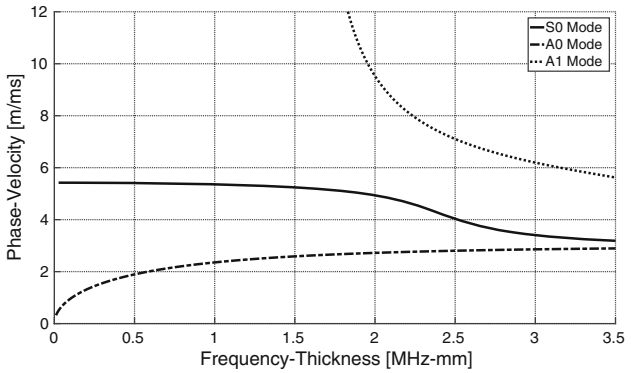

Fig. 7 Plot obtained using DISPERSE software, showing phase velocity vs frequency thickness for a mild-steel plate immersed in water

in the frequency thickness region of interest of 0.5 Hz-mm–1.5 Hz-mm. Therefore, the fundamental symmetric Lamb mode S_0 was chosen for the experiments. At frequency-thickness value of 1 MHz-mm, the expected speed of wave in mild speed pipe is in the range of 5600 ms^{-1}. Based on this expected velocity, the ideal angle of incidence for creating a 90 degree angle of refraction was calculated to be 15.5 degrees using Snell's law. The slot in the probe holder is tilted at this angle to ensure appropriate incidence of the ultrasonic wave. The response signal is received at 50 MHz sampling rate which ensures that frequencies up-to 25 MHz are captured. However, for the purpose of detecting 1 Mhz S_0 mode response the raw signal was processed through a Chebyshev type-1 band pass filter of lower passband frequency of 0.5 MHz and higher pass band frequency of 1.5 MHz. Also, to identify the regions of prominence in the obtained time series, the signal envelope is evaluated using analytical signal obtained through the Hilbert transform of the series. Figure 8a shows the filtered response and its envelope for a signal obtained from a defect-free region and Fig. 8b shows the same for a region with a defect. The shown region with a defect has a 4 mm hole drilled at the bottom of the pipe.There are three prominent responses to be identified from Fig. 8a, marked by labels $T1$, $T2$ and $T3$.

$T1 = 11$ μs is the response from first reflection of the incident signal from the top of the pipe surface. It should be received in every case irrespective of presence of a defect, as long as the defect is not at the point of incidence. It can clearly be observed at the same position even in Fig. 8b for the defect region, thus validating both signals in successful incidence of the ultrasonic wave.

$T2 = 23.6$ μs is a spurious response obtained from an axial feature and can be ignored in the current analysis.

$T3 = 60$ μs is the response from the guided wave which has travelled the full circumference of the pipe. Hence the time taken for the wave to travel the circumference of the pipe is $T3 - T1 = 49$ μs. Based on the 28.27 mm circumference of the pipe, the speed of the sound wave obtained is 5770 ms^{-1} which is accurate up-to 3% of the required speed. This confirms that the wave has indeed travelled the full circumference of the pipe. It is to be noted that this response is missing from the defect region signal because the wave could not travel the full circumference as it was dispersed by the presence of the defect. Hence it can be used for effectively identifying the presence of a major defect in the pipe in real time.

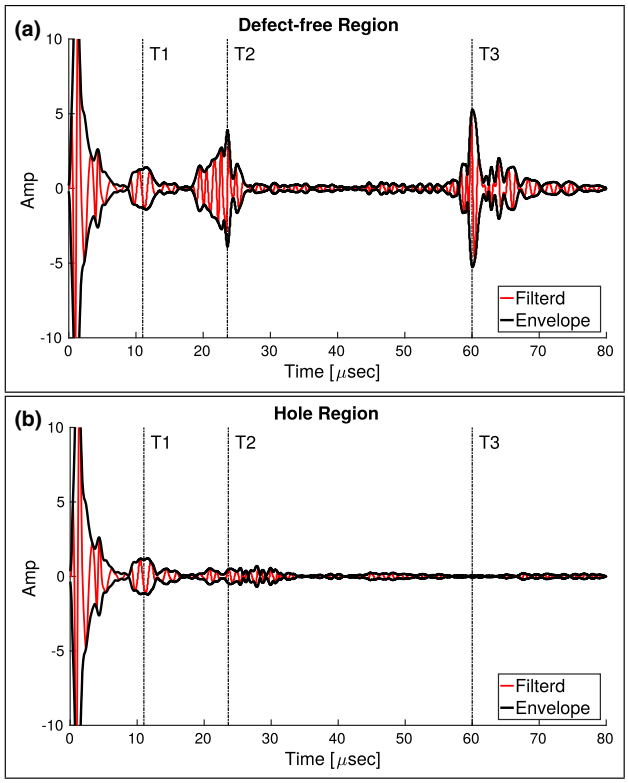

Fig. 8 Plots showing the detection of a hole defect in the pipe using the circumferential guided ultrasonic waves; **a** Plot showing filtered signal and its envelope obtained from defect free region; **b** Plot showing filtered signal and its envelope obtained from the region with the hole

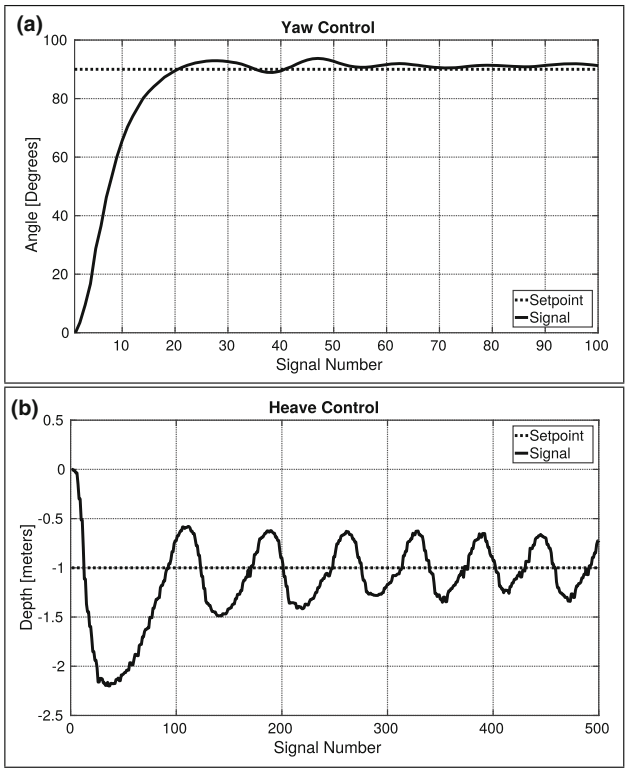

Fig. 9 Plots showing PID tuning outputs for steady state setpoints; **a** Plot showing PID tuning output of yaw control; **b** Plot showing PID tuning output of heave control

Control

The tuning of control loops and test experiments were conducted in a wave flume of water depth 2.5 m and width of 4 m; in calm water conditions.

Figure 9a shows result from tuned yaw control. It was tuned to have low overshoot and low settling time but has a relatively high initial rise time. Figure 9b shows a result from tuned heave control loop. Heave control is more challenging because the ROV is slightly positively buoyant creating an imbalance in the force required for moving deeper as compared to moving upwards. Due to this the control had to be tuned to be more aggressive thus creating more oscillations about the required set point.

The functioning of the complete control loop can be explained using the following example. Let the user require the ROV to take a 90^0 anticlockwise yaw turn. The user will provide an input to the off-board computer through joystick to change the yaw set-point by $+90^0$ which will be relayed to Raspberry-Pi and this will initiate a response from the yaw PID controller. Raspberry-Pi will receive the current heading of the ROV using its sensor, and based on the tuning of yaw PID controller it will produce a response in terms of thrust required from the two surge thrusters. This information will be relayed to Arduino Mega. The micro-controller will initiate individual power control loops for the two surge thrusters. It will receive current state of the thrusters using current and voltage sensors and produce required PWM response to generate sufficient thrust and trigger a turn. This turn will be measured by the IMU and the feedback will be given to main yaw PID controller on Raspberry-Pi hence closing the control loop. This process will hence ensure that a successful 90^0 turn has been made.

After tuning individual PID control loops shown in Fig. 6, a series of perturbations tests and successive L-Turns and Z-Turns tests were carried out to determine the long-term performance of the complete control loop.

Yaw perturbation test was carried out by fixing the ROV orientation set-point and then randomly disturbing it using a stick as shown in the time-lapse in Fig. 11a. It is evident from this sequence of images that the ROV is able to maintain its orientation in physical space despite external perturbations.

Figure 10a shows the result from successive L-Turns test. This type of turn is significant for smooth navigation along pipeline networks. The slow rise time tuning is evident from the small lag between set-point signal and

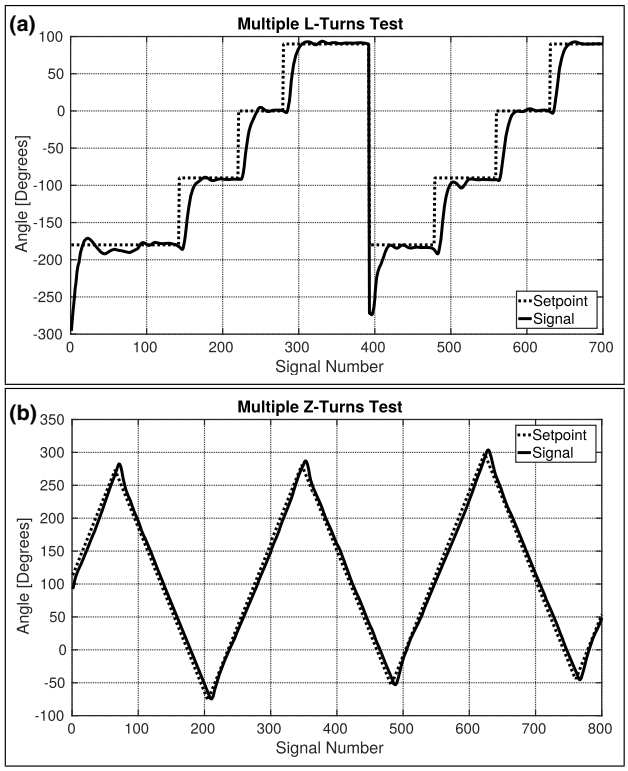

Fig. 10 Plots showing PID tuning outputs for angled turns; **a** comparison for successive L-turns, **b** comparison for successive Z-turns

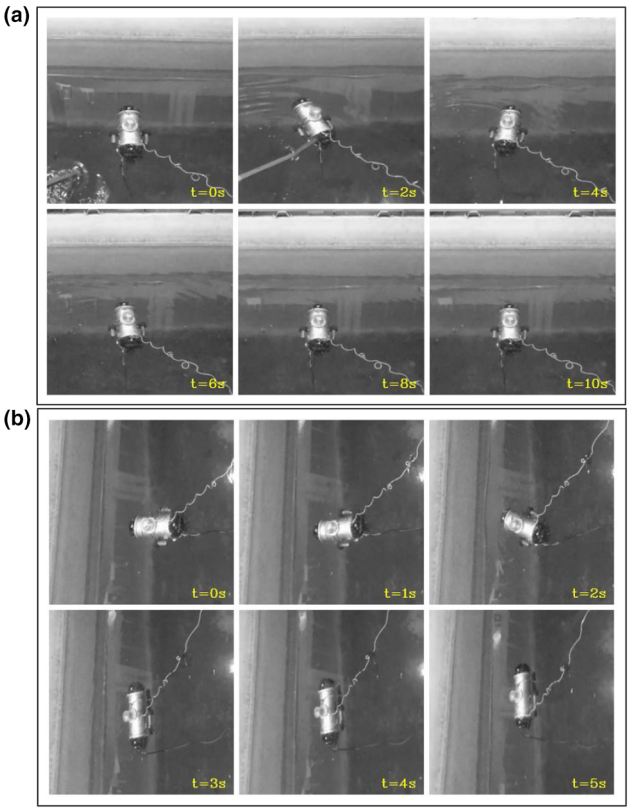

Fig. 11 Sequence of snapshots demonstrating the action of the tuned PID controller for **a** Yaw impulse-input and **b** L-angled turns

response signal, however, this has also prevented any sharp overshoots and maintained small settling time even for multiple turns. Figure 11b shows a sequence of images displaying results in physical space for a L-turn. Similar conclusion can be made from the results of successive Z-Turns in Fig. 10b.

Conclusion

The ROV was hence able to meet the primary objective. The novel design of the pneumatic gripper on the ROV hull is able to grip underwater pipes with reasonable strength. The ROV demonstrated that the combination of processing power of Raspberry-Pi and Arduino Mega, along with feedback sensors like IMU and pressure sensor can be used for controlling multiple degrees of freedom of motion simultaneously and hence provide a reliable navigation to a submerged pipeline network. It can successfully hover over pipe and grip it using the pneumatic grippers. The specially tuned motion control ensures that ROV can move along a pipe's length and take angled turns when necessary over the pipeline network. The affordable hardware has also been made to handle the large amount of data real-time from two mounted cameras for a live feed and ultrasonic probe. It has also successfully demonstrated the use of circumferential guided ultrasonic waves for detecting defects on pipeline surface.

This system hence overcomes the shortcoming of pigging as it can comfortably make sharp angled turns, provide live feedback of data and can run for long periods of time as it is externally powered. Also, it does not expose the pipeline to surrounding atmosphere and thus avoids any chance of contamination and addresses the safety concerns regarding insertion and removal of a pig device. In case more processing power is required, an arrangement using multiple Raspberry-Pi or using ones with better specifications, such as Raspberry-Pi 2 with 1 GB of RAM and a 900 MHz quad core processor; would be more effective as compared to available alternatives. The same system can be modified for inspecting large tank floors too.

This work has presented the proof of concept of using ultrasonics for defect detection using ROVs. With this basis, the future work will investigate complex problems such as identification of the defect, use of an array of ultrasonic sensors for simultaneous analysis, etc. Additionally, this work shows the ability of the communication and electronics architecture built using affordable and readily available microprocessor and sensors to handle the large amount of data associated with simultaneous

ultrasonic sensing and ROV control. This approach can be scaled for increased complexity in control and sensing.

Acknowledgements This project was funded by Innovative Student Projects initiative at Indian institute of Technology Madras. The facilities for underwater testing were provided by Ocean Engineering Department, IIT Madras.

References

Barrett SF (2010) Arduino microcontroller: processing for everyone! part II. Synth Lect Digit Circ Syst 5:1–244. https://doi.org/10.2200/S00283ED1V01Y201005DCS029

Bubar BG (2011) Pipeline pigging and inspection. In: Shashi Menon E (ed) Pipeline planning and construction field manual, Chap 15. Gulf Professional Publishing, Boston, pp 319–339. https://doi.org/10.1016/B978-0-12-383867-4.00015-3

Combe D, Hair D (2011) SPE 143748 problems with operational pigging in low flow oil pipelines. Society of Petroleum Engineers (September)

Fossen TI (1994) Guidance and control of ocean vehicles, vol 32. Wiley, Hoboken. https://doi.org/10.1016/0005-1098(96)82331-4

Huang AS, Olson E (2010) LCM: lightweight communications and marshalling. In: 2010 IEEE/RSJ international conference on intelligent robots and systems. pp 4057–4062. https://doi.org/10.1109/IROS.2010.5649358

Na WB, Kundu T (2002) Underwater pipeline inspection using guided waves. Journal of Pressure Vessel Technology 124(2):196. https://doi.org/10.1115/1.1466456

Ogata K (1990) Modern control engineering, 2nd edn. Prentice Hall PTR, Upper Saddle River

Pavlakovic B, Lowe M, Alleyne D, Cawley P (1997) Disperse: a general purpose program for creating dispersion curves. In: Thompson DO, Chimenti DE (eds) Review of progress in quantitative nondestructive evaluation, vol 16A. Springer, Boston, MA, pp 185–192. https://doi.org/10.1007/978-1-4615-5947-4_24

Raspberry Pi Foundation (2012) Raspberry pi - teach, learn, and make with Raspberry Pi. https://www.raspberrypi.org/. Accessed 5 Oct 2015

Rose JL (2004) Ultrasonic waves in solid media. Cambridge University Press, Cambridge, UK

Toma G, Schumacher L, De Vleeschouwer C (2011) Offering streaming rate adaptation to common media players. 2011 IEEE International Conference on Multimedia and Expo, pp 1–7, https://doi.org/10.1109/ICME.2011.6012181