Advanced Robotics, Vol. 17, No. 4, pp. 339–358 (2003)

Full paper

The concept and research of a pipe crawling rescue robot

ZHELONG WANG* and ERNEST APPLETON

School of Engineering, University of Durham, Durham DH1 3LE, UK

Received 19 June 2002; accepted 31 July 2002

Abstract—This paper presents the control algorithm and design of a pipe crawling robot which can be used for the purpose of earthquake rescue and pipeline maintenance. The robot is designed to be able to intelligently alter its body shape to fit the pipe or tunnel-like voids within rubble. The paper introduces a simulation to test how the robot alters its body shape to fit voids. The control algorithm uses a look-up table method combined with the method of least squares to predict the shape of the robot under the influence of actuators. The paper also presents the design of both the hardware and software systems of the robot, and laboratory experiments on the robot body module. Computer simulation results by using MATLAB and the experimental results indicate the feasibility of the robot body shape change control algorithm proposal.

Keywords: Pipe robot; control algorithm; method of least squares; deformable body; rescue robot.

1. INTRODUCTION

1.1. A pipe crawling robot

International pipeline systems used for the transmission of oil, gas and water are growing in age, and some installations have already been in operation beyond the service life they had originally been designed for. Inspection of short pipes is an important task faced by many industries. The deterioration of the inner surface of pipes is especially common in refineries and steam plants. It is therefore of ever-increasing importance that pipeline operators are provided with the means to accurately and reliably inspect their pipelines, and obtain the information needed for decision making regarding safe operation, rehabilitation and repair [1]. The use of automatic pipe inspection could reduce the downtime and manpower required for the pipe inspection process. This highly demanding requirement has resulted in the development of various kinds of pipe robots.

*To whom correspondence should be addressed. E-mail: zhelong.wang@durham.ac.uk

1.2. Earthquake rescue robots

Earthquakes are unfortunately frequent happenings and very dangerous natural phenomena. In almost every major earthquake many victims are buried under collapsed buildings, bridges, roadways, etc. It is very difficult to rescue these people, who may well be injured, hungry and weak. In addition, the structural conditions under the rubble can be complex, dangerous and unknown. Thus, there is a primary need to explore these conditions and determine the victim's location and their condition. The matter is also urgent as it is important to rescue those victims as soon as possible. As a result, rescue robots have been developed to undertake these tasks. In an article by Tokuda *et al.* [2], a prototype CUL (Carry and power assist robot for Unspecified Landform) which will be used for earthquake rescue missions has been described. The paper also describes results of experiments for a feasibility study. The experimental results show the effectiveness of the CUL rescue robot proposed.

2. PROBLEM PRESENTATION

2.1. The shortcomings of existing pipeline robot design

Pipeline robots are mentioned in a large number of papers. For example, a robot system named KARO [3] (Entwicklung eines flexible einsetzbaren Robots zur intelligenten sensor-basierten Kanalinspektion) is equipped with intelligent multi-sensors. These allow automatic and reliable detection of damage location, its type and size, and give superior performance compared with the majority of CCTV-based systems. The system has been developed for the protection of groundwater and soil against contaminating materials and liquids. Toyomi and Koichi [4] have developed a micro-inspection robot for 1-inch diameter pipes. The robot can undertake visual inspections in pipes and also retrieve small objects. Glen and Devon [5] describe a robot that undertakes automatic pipe inspection and has been proved to be a suitable alternative to current pipe inspection techniques. The advantage of this system is that it utilizes established mechatronic principles to produce a low-cost device capable of detecting inner pipe defects. However, all these robots can only work in pipes of fixed diameters and cannot work if the wall of the pipe is broken or badly damaged, if the pipe diameter varies or if the pipe collapsed partially.

2.2. A Brush robot

A pipe crawling robot has been developed at the Center of Industrial Automation and Manufacturing (CIAM), School of Engineering, University of Durham. The robot utilizes a unique, innovative and patented traction system. The principle of the drive system is simple. That is, if a brush with a diameter slightly larger than the bore of a pipe is inserted into a pipe, its bristles are swept back at an angle. Under this condition, it is easier to push the brush forwards though the pipe than it is to pull it backwards. Thus, if two brush are interconnected by a reciprocating cylinder, then, by cycling the cylinder, it is possible for the robot to crawl along the pipe [6]. The drive system has been awarded patents on a near worldwide basis. More details about the drive system are illustrated in Stutchbury [6] and Han [7]. Most of the robots developed in Durham are powered by pneumatics and grip the wall of the pipe by means of many bristle clusters, hence named Brush robots. Just like other pipeline robots mentioned earlier, a Brush robot can only work in pipes within a limited diameter range although, unlike many of the other traction systems, the bristle mechanism is capable of dealing with broken on partially collapsed pipes.

2.3. An improved Brush robot for earthquake rescue and pipeline maintenance

This report outlines some improvements on the established Brush robot principle, and adapts it to work for the purpose of earthquake rescue and in severely damaged or broken pipelines. Thus, it is designed to be able to alter its body shape to fit the variable void shapes in a collapsed building or different diameter pipes whose walls might be broken or in a bad condition. To realize these functions, a sensor system for detecting the hole shape and a control system for altering the robot body shape is necessarily equipped. For the purpose of rescue, a CCD camera needs to be equipped to see the conditions in the hole. A CO_2 detector is used for detecting whether victims are alive or not and for locating their positions. A microphone is equipped for a victim to communicate with rescue personnel. In addition, an air hose will be carried by the robot for conditions of air deficiency.

3. RESEARCH METHODOLOGY

3.1. Outline

The key point for a Brush robot is to produce enough friction to drive itself and carry the necessary loads. For the purpose of earthquake rescue, the robot should be able to pass through holes and cracks in the ruins made by the earthquake. However, the holes and cracks in the ruins usually have irregular shapes and sizes. Thus, the robot must be able to alter its body shape and fit the hole shape to produce enough frictional force. To realize the requirement, it is essential to install a sensor system on the robot's head, which is used to roughly detect the hole and crack shapes. In the design of this Brush robot, four groups of actuators need to be installed around the robot's body. These actuators will change the robot's body shape according to signals from the sensor system. During this procedure a control system will recognize the sensor's signals, performing calculations based on the signal information and sending appropriate commands to change the actuators. Using a CCD camera mounted on the head of the robot, the conditions in the holes and cracks can be investigated visually. A manual control function is also required to deal with exceptional circumstances in the case of failure of the robot automatic control function. A software system based on a PC is also required to run data communication and control, store data, etc. More specifically, the software is required to be able to record the robot's routes and the hole shapes at different travel stages.

3.2. Robot working mechanism theory

3.2.1. The mechanism theory of an old Brush robot. Before building a real robot model, it is necessary to do some computer simulations to prove the feasibility of the control algorithm. Movement of a Brush robot is achieved by the utilization of curved bristle as the means of propulsion and support, as illustrated in Fig. 1.

When the cylinder opens, the leading brush, offering lower resistance because of the bristle curvature, moves forward easily; the trailing brush, because of its higher resistance to backward forces, remains stationary. However, when the reverse happens, i.e. the cylinder closes, the leading brush remains stationary, whereas the trailing brush, now offering low resistance, is pulled forward.

Based on this theory, the resultant traction depends entirely on the bristle mechanism set-up and can be illustrated in the following way. Considering a single bristle for the purpose of simplicity, when a bristle is put into a pipe, and because of its effective lateral dimension, it is bent by the pipe wall, there will be a perpendicular force P acting at the tip of the bristle, as shown in Fig. 2. When moving the core of bristles, traction F should equal μP. The projection of a bristle in the direction of the y-axis is marked as h. The length of a bristle is l. The chord between the two tips of the bristle is expressed as L. In the thesis [6], Stutchbury gives the conclusion that the optimum angel between the bristle and pipe wall should be between 30° and 40° to achieve the best traction F, although this angel will vary depending upon a number of factors, e.g. lubrication. In Ref. [7], Han gives us the relation between h and l:

$$\frac{h}{l} = 2\frac{E(\alpha)}{Ql} - L. \tag{1}$$

Note:

$$E(\alpha) = \frac{\pi}{2}\left[1 - \left(\frac{1}{2}\right)^2 \sin^2\frac{\alpha}{2} - \left(\frac{1*3}{2*4}\right)^2 \frac{\sin^4\frac{\alpha}{2}}{3} - \left(\frac{1*3*5}{2*4*6}\right)^2 \frac{\sin^6\frac{\alpha}{2}}{5} - \cdots\right],$$

and

$$Ql = \frac{\pi}{2}\left[1 + \left(\frac{1}{2}\right)^2 \sin^2\frac{\alpha}{2} + \left(\frac{1*3}{2*4}\right)^2 \frac{\sin^4\frac{\alpha}{2}}{3} + \left(\frac{1*3*5}{2*4*6}\right)^2 \frac{\sin^6\frac{\alpha}{2}}{5} + \cdots\right].$$

The value of h can be obtained if l and α are known. The thesis by Han gives a table which indicates the results obtained by applying (1), as shown below in Table 1.

3.2.2. The mechanism theory of an improved Brush robot. A newly improved Brush robot used the same brush mechanism as the old one, but the structure of

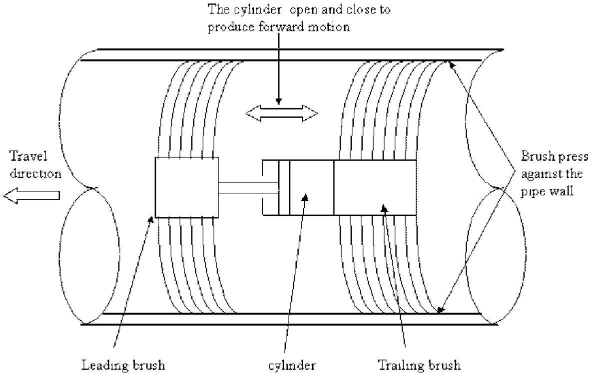


Figure 1. Brush robot motion principle.

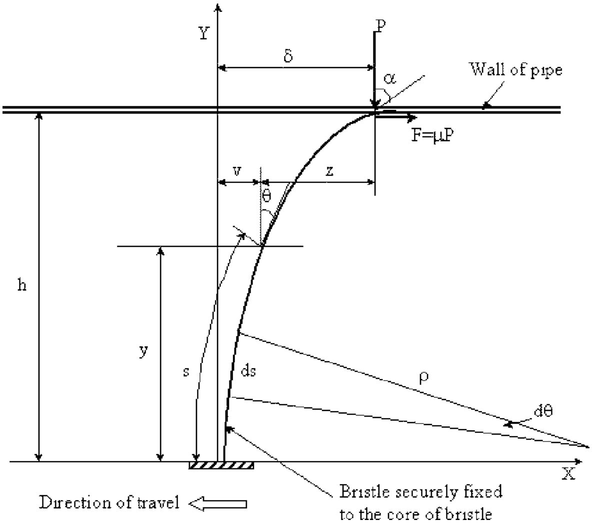


Figure 2. Bristle mechanism diagram.

Table 1.
Spreadsheet for the relation between h and l

α (deg)	P/P_{Euler}	δ/l	h/l
10	1.004	0.1116	0.9923
20	1.016	0.2193	0.9698
30	1.035	0.2588	0.9324
40	1.062	0.4221	0.8787
60	1.152	0.5930	0.6973
90	1.392	0.7925	0.4189

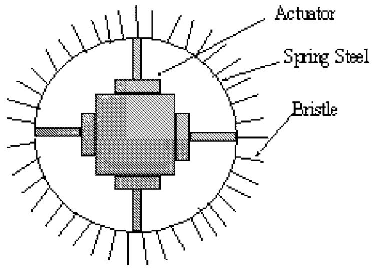


Figure 3. The structure of one robot body module.

its body module has been modified. The old Brush robot's body module is a solid steel cylinder mounted with hundreds of steel bristles. The steel cylinder is the robot body core of a fixed diameter and the body core cannot do any change to its physical shape. However, in the improved Brush robot, a thin steel strip circle replaces the old robot body cylinder, as illustrated in Fig. 3. Inside the strip circle, four actuators connect the strip circle by joining the actuator end points with the strip circle. Like the old robot body module, hundreds of steel bristles are mounted on the surface of the strip circle. The new, improved robot body module with such a mechanism can alter its shape by actuators pushing in and out.

3.3. Control algorithm for the new Brush robot

To make an improved Brush robot to be able to alter its body shape and fit the voids, a hybrid control algorithm based on a look-up table method and the method of least squares is developed. The method that the robot use to decides how to alter its body shape to fit the hole is reference to a data file, stored in a table. The table is composed of data files and each file presents one calculation result obtained by using ABAQUS software. Figure 3 illustrates the structure of one robot body module. Each robot body module is composed of four actuators, a thin spring steel strip circle and hundreds of spring steel bristles mounted on the surface of the strip. The end point of each actuator is connected to the strip so that the shape of the steel strip circle can be deformed by the actuators pushing in and out. If several hundred points on the strip circle are marked, the position of each point can be recorded as a pair of coordinates. The strip circle shape can be uniquely represented if all points' coordinates can be known. The shape of the strip circle can be acquired from coordinates of those points when actuators push in/out and the strip circle is deformed by such pushing. In fact, here the shape of the strip circle represents the actual shape of the robot body module. These shapes are related to actuator loads and consequently deflections that are put on the robot body in the direction of the x-axis and y-axis independently. Thus, the robot body module shape will be changed with varied actuator loads and deflections. In the mean time, the coordinates of the points on the robot body module will be changed because of the robot body deflection. The coordinates of those points can be calculated and predicted by using ABAQUS, when actuator loads are already known. This array of point coordinates presents the robot

Table 2.
A data file stores an array of point coordinates

	$POINT_1$	$POINT_2$	$\cdots$	$POINT_n$
X_i	X_1	X_2	$\cdots$	X_n
Y_1	Y_1	Y_2	$\cdots$	Y_n

body module shape under such actuator loads. If the actuator loads are changed, a new array of point coordinates can be acquired and this means that the robot body module will assume a new shape. Thus by changing the actuator loads, which are input variables in the calculation by ABAQUS, many arrays of point coordinates can be acquired. Each of them uniquely represents a robot body module shape. These arrays of point coordinates can be stored in data files and each data file is used as a record to put into a table. As a result, this table includes many robot body module shapes under different actuator loads. Similarly a hole shape can also be uniquely expressed as an array of coordinates of points around the hole wall. The robot's control algorithm is required to be able to find the most appropriate shape to fit the hole shape from those data files in the table.

For example, Table 2 expresses an array of point coordinates stored as a data file, which is a record in the robot body shape table. A $POINT_i$ $(i = 1, 2, \ldots, n)$ means a pair of coordinates of a point on the robot body after the robot body is deflected.

$POINT_i$ is expressed as (X_i, Y_i) in the meaning of the coordinates, which is a point on the robot body. The coordinate (x_i, y_i) is expressed as the coordinate of a point on the hole wall. d_i is the difference between a point on the robot body and its corresponding target point on the hole wall along the same direction. If the robot body module shape could fit the hole shape very well, that means that each d_i should be as small as possible. To realize this, the robot control algorithm needs to find the smallest D, which is the sum of the squared distance d_i. Finally, the control algorithm will search all data files in the robot body module shape table and find the best data file to minimize D, which is calculated by using (3). l is the optimal length of the bristle, which is a constant and can be know by using a spreadsheet in [6]. n is the total number of points around the robot body module.

$$d_i = \sqrt{x_i^2 + y_i^2} - \sqrt{X_i^2 + Y_i^2} - l, \quad (i = 1, 2, \ldots, n), \tag{2}$$

$$D = \sum_{i=1}^{n} \left[\sqrt{x_i^2 + y_i^2} - \sqrt{X_i^2 + Y_i^2} - l \right]^2, \quad (i = 1, 2, \ldots, n). \tag{3}$$

3.4. Computer simulation of the control algorithm

To test the feasibility of the control algorithm presented above, a number of computer simulations have been performed using MATLAB. The simulation results show that the control algorithm proposed will be effective. Figures 4–6 are based

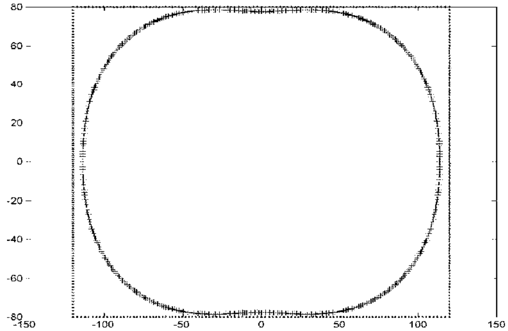


Figure 4. Robot body module fits a square: '+' indicates a point of the robot body and '·' indicates a point of the hole's wall.

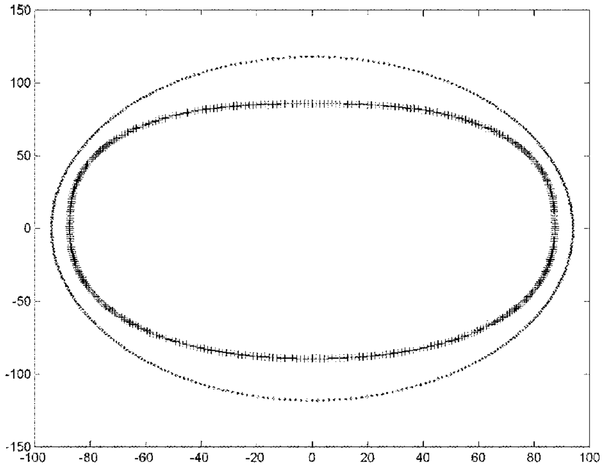


Figure 5. Robot body module fits an ellipse: '+' indicates a point of the robot body.

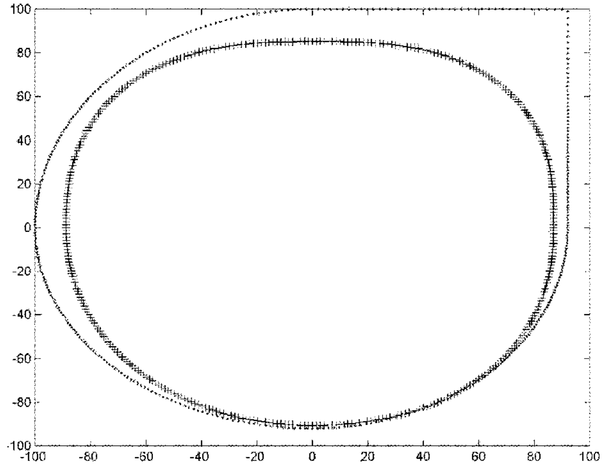


Figure 6. Robot body module fits an irregular shape: '·' indicates a point of the hole's wall.

on some simulation results. In these figures, the hole shape is expressed by an array of interconnected '·' symbols and the robot body module shape is expressed by an array of interconnected symbols '+'. Also each '·' symbol represents a point on the hole wall and each '+' symbol represents a point on the robot body module. In Fig. 4, the hole shape is a rectangle and the robot body curve fits the hole well. In Fig. 5, the hole shape is an ellipse and the robot body curve is similar to the hole shape, but it is relatively smaller than the hole. The bristle around the robot body can deal with the little difference by elastic deflection. Figure 6 shows the robot body tested to fit an irregular hole shape and most of the robot body module curve can fit the hole by elastic deflection of the bristles; however, the right upper corner cannot be fitted by the robot body module curve. To achieve a better fitting, further work needs to be carried on making a non-symmetric robot body.

From the simulation results above, the robot body module could alter its shape to fit some basic geometric shapes and simple irregular shapes. The control algorithm is basically proved to be feasible. More complicated irregular shapes cannot be fitted well by using the current structure of the robot body module. This could be solved by making a non-symmetric robot body module and using more actuators, which will enable robot body module to change into more complex shapes to fit various hole shapes.

3.5. Robot control system

3.5.1. Control module diagram. It is envisaged that a control program will run in a PC and that a sensor system will send back signals about the constantly changing hole shape. After processing these signals, the PC makes the control decisions and sends control commands to every control module. Then the control modules will control the movements of the actuators according to those control commands. Figure 7 displays that a robot control system that includes two layers of control. One is the control from a PC to each control module; the other is the control from each control module to its corresponding actuator. In addition, a sensor system

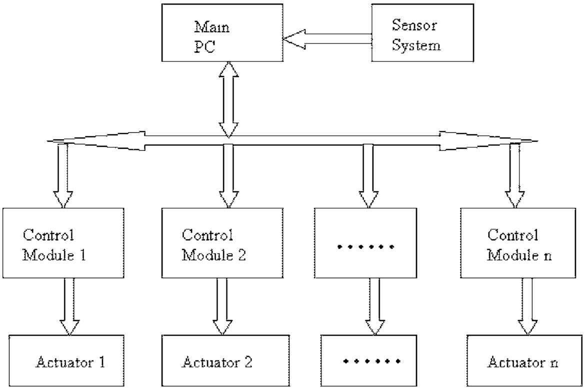


Figure 7. Control module diagram.

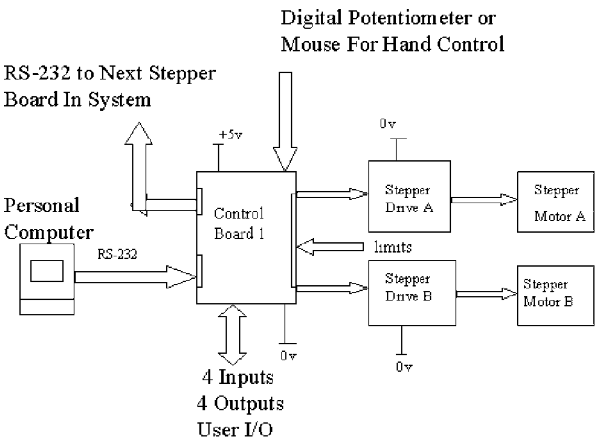


Figure 8. Control board diagram.

communicates with the PC to collect the information of the hole shape and sends the information to the PC.

3.5.2. Control board diagram. It is proposed to make a prototype of this robot using eight stepper motor control boards, 16 stepper motor drive boards, 16 stepper motors and one PC to run the control software. In Fig. 8, a scheme is drawn to explain the connections between these modules. Each control board can control two stepper motors and every stepper motor needs to be driven by a drive board. Four stepper motors are needed for each robot body module, so that a four-body-part Brush robot needs 16 stepper motors in all.

4. EXPERIMENTS

This Brush robot is composed of the same four modules. Thus, the laboratory experiment focuses on one robot body module.

4.1. Laboratory experiment on a robot body module

The robot body module experimental device is shown in Fig. 9. The devices include a strain gauge sensor, robot body module, robot actuator control box, DC power supply and PC. The strain gauge sensor is used to detect the void shapes. The sensor includes 12 fingers equipped with strain gauges. The strain gauge can be used to detect the deflections of the fingers. The touching point coordinates of the fingers on the void wall can be worked out by these deflections. With these coordinates, a spline algorithm can estimate the whole void shape.

4.1.1. Robot software system. A computer program written in Visual C++ was developed for the robot prototype experiment to collect sensor signals, control

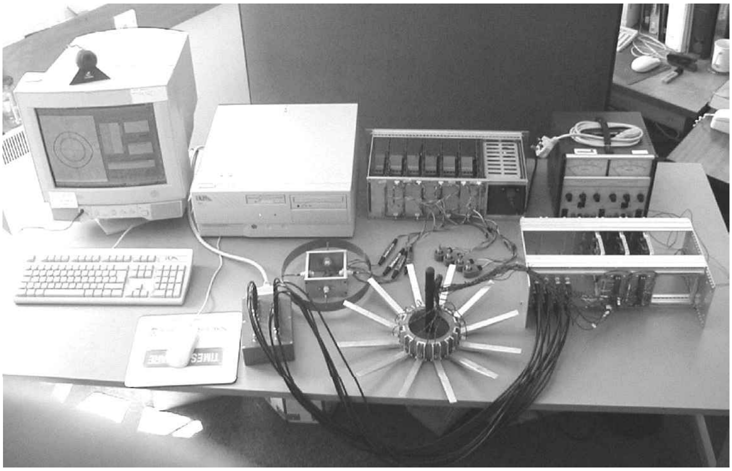


Figure 9. Robot experimental devices.

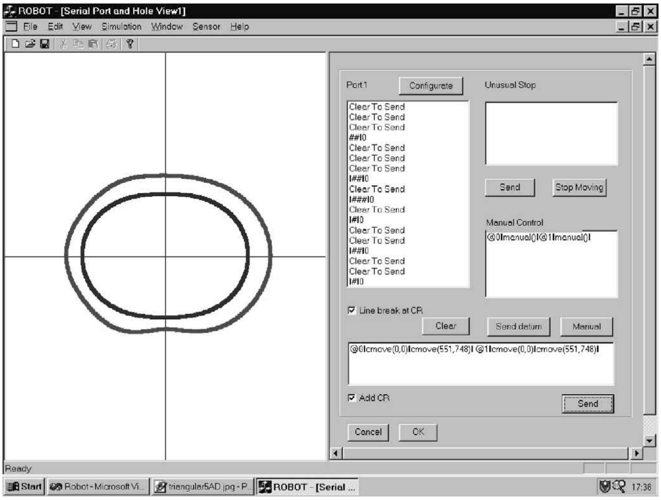


Figure 10. Robot software interface.

actuators and realize the shape change algorithm. Figure 10 illustrates the program interface. The left window shows how the robot body module will alter its body shape to fit the void. The black inner line shows the outline of the robot body module and the grey outer line shows the outline of the void wall. The right window displays the communication working status; the stepper motor working status and control commands sent to the stepper motors. Figure 11 shows the interface of the data acquisition program, which displays the data acquisition card information; 32 analog input channels in single ended mode and 16 analog input channels in differential mode. Only 12 channels in differential mode are used in this experiment since there are only 12 strain gauge output signals to be converted in this application.

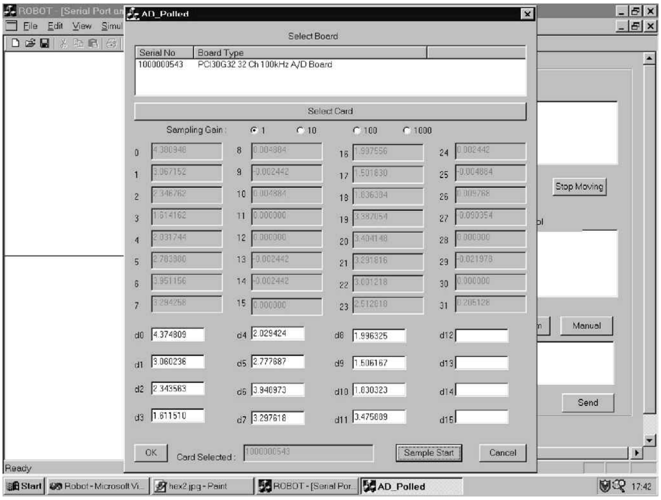


Figure 11. Data acquisition interface.

4.1.2. Eperimental procedure and results. The device shown in Fig. 9 is used in the experiment. To simulate the void shape and test the theory, a few wooden boxes were made in a few basic regular geometric shapes and an irregular geometric shape. Actual void shapes in a real environment are much more complex than such simple geometric shapes, but it is reasonable to simplify the problem at the initial stage of the robot development. Also, the experiment is designed to test that the robot body shape algorithm can work correctly. The sensor was only a prototype used to test the feasibility of robot shape change theory and needs further technical improvements to detect more complicated void shapes. Thus, complicated void shapes in real environments are not considered in this experiment. The regular geometry boxes in the experiment include a square, a rectangle, a triangle and a hexagon.

The experimental procedure is to put the robot sensor into the box which simulates a void shape. Then the data acquisition card collects the box shape information and sends it to the PC. The program running on the PC decides what shape the robot body module should change into to fit the box shape as closely as possible and how the actuators move according to the sensor information. After the program's decision, the program sends the control command to the control module via RS 232 serial communication and the control module will control the stepper motors, moving as much as the program decides. The robot body module then needs to be put into the box to see how the robot body module fits the box. The experimental procedure needed to be repeated for each box in different shape.

Figure 12 shows the robot sensor put into a square box to detect the box shape. Figure 13 shows how the program decides the robot body's change. The left window shows that the robot body module needs to be altered into a square. The right window shows the control commands sent to the control module. All stepper motors

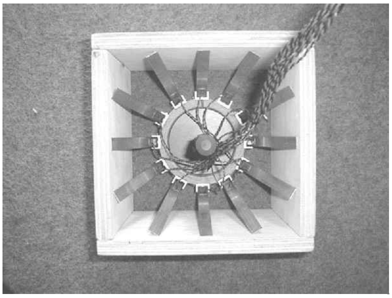


Figure 12. Robot sensor detects a square box shape.

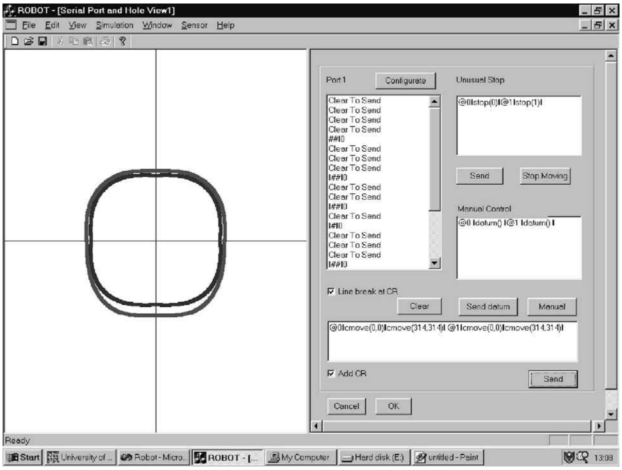


Figure 13. The most appropriate shape decided by shape change theory to fit a square box.

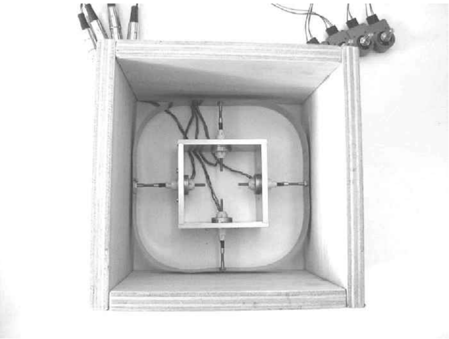


Figure 14. Robot body module fits in a square box.

move in, following the command of the control system. Figure 14 shows how the changed robot body module fits the square box.

Figure 15 shows the robot sensor in a rectangular box. The program obtained a rough box shape indicated by the grey outer line and decided the most appropriate

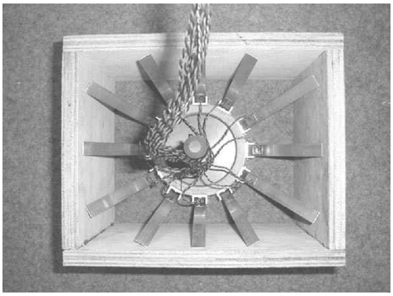


Figure 15. Sensor detects a rectangular box shape.

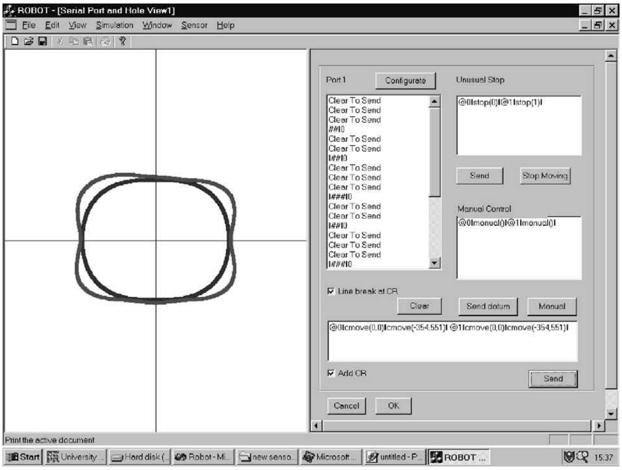


Figure 16. The most appropriate shape decided by shape change theory to fit a rectangular box.

robot body module shape indicated by the black inner line as illustrated in Fig. 16. Following the control commands, the stepper motors in the y-axis pushed in and the stepper motors in the x-axis pushed out. Figure 17 shows that the altered body module fits the rectangular box reasonably well.

Figure 18 shows the robot sensor in a hexagonal box. The program obtained a rough box shape indicated by the grey outer line and decided the most appropriate robot body module shape indicated by the black inner line as illustrated in Fig. 19. Following the control commands, the stepper motors in the y-axis pushed in and the stepper motors in the x-axis pushed out to produce an ellipse shape. Figure 20 shows that the changed robot body module fits the hexagonal box appropriately.

Figure 21 shows the robot sensor in a triangular box. The program obtained a rough box shape indicated by the grey outer line and decided the most appropriate robot body module shape indicated by the black inner line as illustrated in Fig. 22. Following the control commands, the stepper motors in both the y- and x-axis pushed out slightly to fit the box. Figure 23 shows that the changed robot body module fits the triangular box.

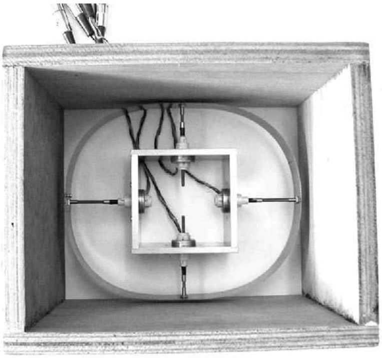


Figure 17. Robot body module fits in a rectangular box.

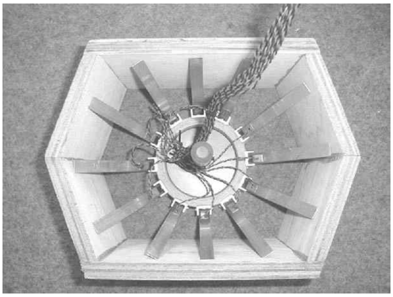


Figure 18. Robot sensor detects a hexagonal box shape.

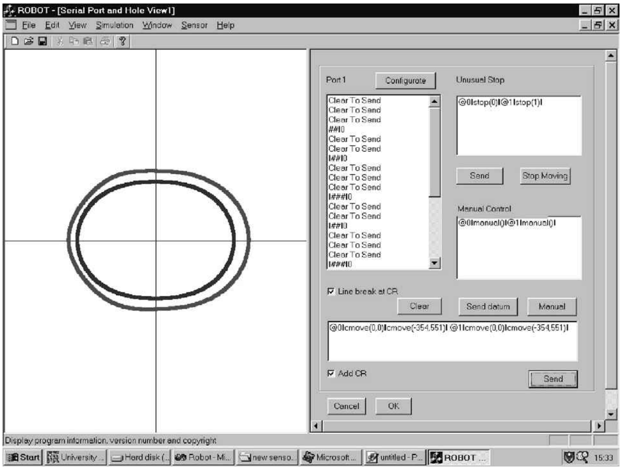


Figure 19. The most appropriate shape decided by shape change theory to fit a hexagonal box.

Figure 24 shows the robot sensor in an irregular shape. The program obtained a rough box shape indicated by the grey outer line and decided an appropriate robot body module shape indicated by the black inner line as illustrated in Fig. 25.

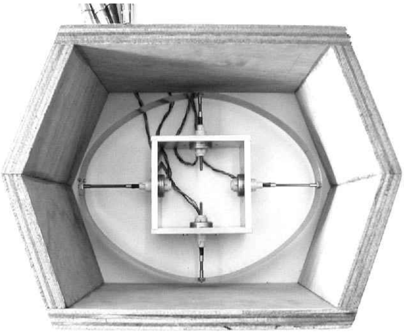


Figure 20. Robot body module fits in a hexagonal box.

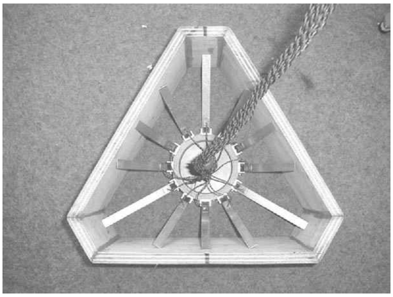


Figure 21. Robot sensor detects a triangular box shape.

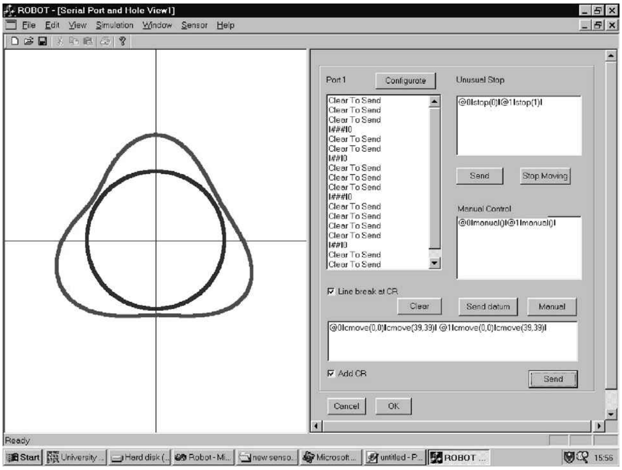


Figure 22. The most appropriate shape decided by shape change theory to fit a triangular box.

Following the control commands, the stepper motors in the y-axis pushed in slightly and the stepper motors in x-axis pushed out to fit the box. Figure 26 shows how the altered robot body module fits the irregular box.

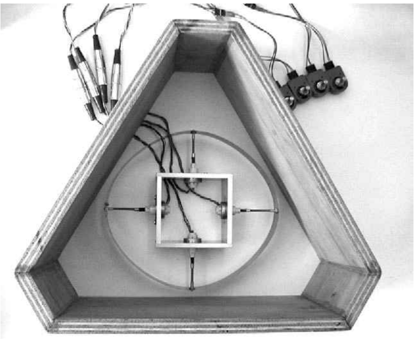


Figure 23. Robot body module fits in a triangular box.

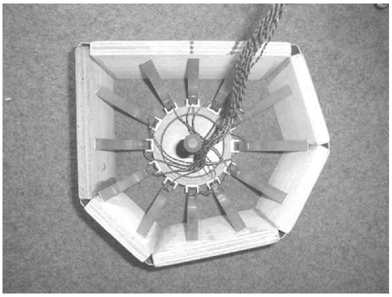


Figure 24. Robot sensor detects an irregular box shape.

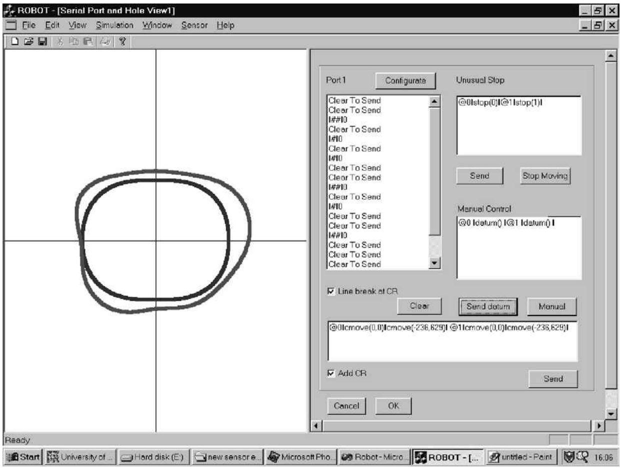


Figure 25. The most appropriate shape decided by shape change theory to fit an irregular box.

4.2. Conclusions of the experiment

In Fig. 23, the robot body module should change to a triangular shape to fit the box shape. However, this did not happen in the experiment. The reason is that the robot body module used in the experiment could only change its shape symmetrically in

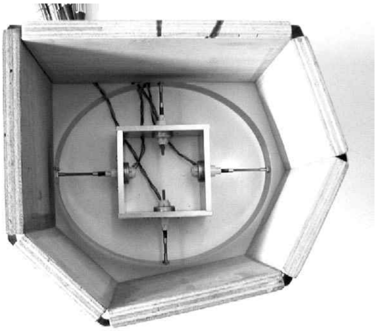


Figure 26. The altered robot body shape in an irregular box.

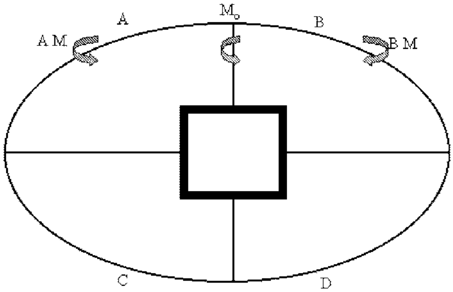


Figure 27. Symmetrical shape change.

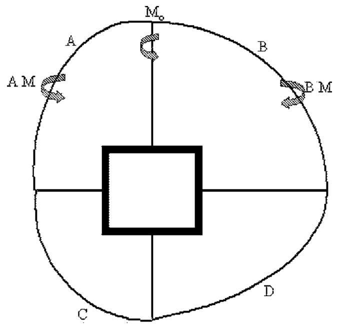


Figure 28. Non-symmetrical shape change.

both the x- and y-axis because of the limitations of its physical structure. In Fig. 27, the robot body module changes its shape symmetrically and produces the shape of an ellipse. The bending moment *AM* produced by part A and the bending moment *BM* produced by B are equal because the robot body module is in a symmetrical shape. Thus, the moment M_0 is zero and the axis of the stepper motor in the y-axis does not bend toward either part A or B.

In Fig. 28, the robot body module changes its shape non-symmetrically and tries to make an irregular shape. The bending moment *AM* and the bending moment *BM* are not equal and *AM* may be bigger than *BM* because of its bigger deformation.

As a result, the moment M_0 acting on the axis of the stepper motor is not zero any more and the axis of the stepper motor will be bent toward part A. Thus, the stepper motor may be blocked and cannot move any more because the stepper motor used in the experiment is not powerful enough to sustain this bending. Thus, the robot body module cannot change its shape to an anticipated non-symmetrical shape. This deficiency is caused by the linear stepper motor used in this experiment because the motor has small power and its axis could not sustain a big side force. However, the problem can be solved by choosing a more powerful actuator and revising the shape change mechanism.

The experimental results show that the robot sensor could roughly detect the box shape and the robot body module could change its shape symmetrically to fit the box as closely as it could. This experiment demonstrated that the robot body shape change algorithm was feasible and the control program worked correctly. The robot sensor needs further work to acquire more accurate void shape information in a real environment. In addition, the robot body shape change mechanism needs more powerful actuators and further work to be able to do non-symmetrical shape changing.

4.3. Future field experiment

The laboratory experiment on the robot body module proves that the given robot body shape change theory is feasible and the robot control system works correctly. The field experiment will be carried out in a quarry, since the geographical condition of a quarry may be suitable to simulate collapsed buildings after an earthquake has taken place. Different holes and cracks will be available in a quarry. This stage is to test how the robot works in a practical environment. A main problem that might be encountered is that side forces acting on the robot vertical actuators may be too large and may damage the actuators or make them work incorrectly. An original design in this experiment uses linear stepper motors as vertical actuators because they are easy to use and can be controlled precisely. If this problem happens, more powerful and robust actuators may be chosen, and the mechanism design for the vertical actuators may need to be revised to make the vertical actuators able to sustain the large side force. A non-contact laser sensor may be developed to substitute the current strain gauge sensor to acquire more accurate void shapes. Finally, the problems encountered in this experiment will be very helpful to improve the robot design, and make a more reliable and practical model in the future.

Acknowledgement

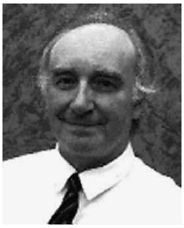
The authors would like to thanks the Audi Foundation for their sponsorship of this research.

REFERENCES

1. M. Beller and H. Schoenmaker, Pipeline inspection: a turnkey approach, Technology report, PII Ltd. (1999).
2. K. Tokuda, K. Osuka, and T. Ono, Concept and development of general rescue robot CUL, in: *Proc. IROS*, Kyongju, Leuven, Vol. 3, pp. 1902–1907 (1999).
3. H. B. Kuntze and H. Haffner, Experience with the developoment of a robot for smart multisensoric pipe inspection, in: *Proc. IEEE Int. Conf. on Robotics and Automation*, Leuven, pp. 1773–1778 (1998).
4. M. Toyomi, S. Koichi, K. Kimura and M. Hasegawa, Development of micro-inspection robot for small piping, *Int. J. Japan Soc. Prec. Engng.* **33** (2), 90–93 (1999).
5. G. Bright, D. Ferreira and R. Mayor, Automated pipe inspection robot, *Ind. Robot* **24** (4), 285–289 (1997).
6. N. W. Stutchbury, Design characteristics of a pipe crawling robot, PhD thesis, University of Durham (1999).
7. C. Han, The experiment investigation of a pipe robot, MSc thesis, University of Durham (1999).

ABOUT THE AUTHORS

Zhelong Wang is a PhD candidate in the School of Engineering at the University of Durham, UK. He received the BE and ME degrees from the Dalian University of Technology, China in 1996 and 1999. His research interests include robotics, intelligent control, mechatronic systems and evolutionary algorithm.



Ernest Appleton holds the Chair in Manufacturing Systems at the University of Durham. Up to January 1993 he was a Lecturer in Engineering at the University of Cambridge and a Director of the Caledonian Mining Co. Ltd. His current interests are in the areas of advanced manufacturing technology and automation, a field in which he has published a number of papers and books. Throughout his career he has operated at the academic–industrial interface, working with such companies as British Aerospace, IBM, British Gas, etc. He also has a keen interest in transferring appropriate technology into the medium and small company sector. He teaches in the area of Manufacturing Engineering and Manufacturing Management for the School of Engineering and Durham University Business School.