AGILE CLIMBING ROBOT DESIGN FOR NDT INSPECTION

Z. ZHAO and G. SHIRKOOHI School of Engineering, London South Bank University, 103 Borough Road, London SE1 0AA United Kingdom

Wall climbing robots are extremely useful for deployment in hazardous environments and operation at great heights and other hard to reach space. However, these robots usually require large and heavy suction pads and associated cylinders for supporting the payload and effectual movement during climbing and inspection process and are very difficult to miniaturise. A new type of robot mechanism has been developed where movements are powered by electric motors and hence can be made on a much smaller scale and with the promise of high speed of operation.

Keywords: Agile robot, suction pads, electric motors, miniaturisation.

1. Introduction

Over the years, a great number of mobile robots have been successfully developed at the Centre for Automated and Robotic Non-Destructive Testing (NDT), London South Bank University (LSBU), UK. Figure 1 shows three inspection robots developed for climbing flat walls and ceiling e.g. large petrochemical storage tanks and ship hulls.

Later, the centre carried out many projects funded by the European Commission. For example, one of the authors has worked on the development of a mobile robot NT systems which can inspect many nozzle joint geometries in radioactive environment as shown in Figure 2, on 860 mm diameter primary circuit coolant pipe. Figure 2(a) shows this robot for weld inspection in lab tests. Figure 3 shows another mobile robot RobAir developed in the centre for the aircraft fuselage inspection.¹ All of these robots have the common properties that the movement is actuated by pneumatic cylinders. The advantages of using pneumatic cylinders are high power to weight ratio and quick movement output. Among the disadvantages are difficulties in the design parameters for miniaturisation and achieving reduction in the size of robots, because of the bulk size of cylinders, and difficulties to make precision control due to the nature of the operation of the pneumatic cylinders. This paper will discuss the problems

associated with the previous version of the mobile robot design and presents the criteria for the design of a new type of climbing robot which overcome the problems and shortcomings of the previous one.

2. Analysis of the problems associated with the previous robot.

Using linear cylinders as actuators means that the robot should follow a complex sequence of actions to make a move. Take the Fig 2(b) robot as an example. Here the y-axis represents the front and x-axis represents the right. If the robot wants to move forward, the robot need to switch on the vacuum valves alternatively between the front and back suction cups and the left and right



Figure 1 Three climbing robots develop by LSBU in the early years

suction cups to keep the robot adhere to the working surface. At same time, it also needs the leg to lift up or put down to coordinate the move. Figure 4 should then extract codes of the control software which show how to move the robot forward. It can be concluded that such kind a robot needs to go through a complex sequence of movements' control and time delay in between to make it move. It will make the fast and stable movement impossible. RobAir has similar design and has the same problems.

2



3

Figure 2 Nuclear Power Plant Robot able to inspect many nozzle joint geometry's in radioactive environment shown on 860 mm diameter primary circuit coolant pipe.

Large size also means heavy weight. To make the robot able to climbing using vacuum force, special suction cups have been designed for these projects to increase the area of the suction pad and to make it able to accommodate curved surfaces.

For example, RobAir robot uses 36 specially made cups. To arrange these cups to adapt to different kinds of working surface is a sizeable challenge. A stable multi support point structure has been developed to deal with this problem in the RobAir project.¹ Even with special design, large size is still very difficult to accommodate for different types of surface. For example, if the robot with the length of 2a works in a curved surface with the radius of curvature of r, the difference in the legs will be at least $h = r - \sqrt{(r^2 - a^2)}$ compared with the robot



Figure 3 The RobAir climbing robot platform.¹

<pre>on the surfaces/ fb&AnLe(LOCKON); 42 (); / Lock the front and back ankle and delay d1()time*/ fb%acua(GM); 24(); 41(); // Switch on the front and back suction cup vacuum and and delay d1()+d2()time*/ iFdg(GM); // Eff and light side a suis cylinder move to "Locg" position?/ moveRev(); 24(); // Left and right side a suis cylinder move to "Back" position?/ iFdg(GM); // Eff and right side suis cylinder move to "Back" position?/ iFdg(GM); 24(); // Left and right side suis cylinder move to "Back" position and delay d1()time*/ iFdg(GM); 24(); // Left and right side suis cylinder move to "Back" position and delay d1()time*/ iFdg(GM); 24(); // Left and right side s asis cylinder move to "Back" position and delay d1()time*/ iFdg(GM); 24(); // Left and right side suis cylinder move to "long" position and delay d1()time which press the front and back position iFdg(GM); 24(); // Loft show the left and right suitor our vacuum*/ iFdg(GM); 24(); // Loft show to the left and right suiton cup vacuum*/ iFdg(GM); 24(); 21(); // Eless the front and back side suis cylinder move to "back" moveTwo iFdb(GM); 24(); // Left and how to diac side suis cylinder move to "back" // iFdg(GM); 24(); 24(); // Left and right side suis cylinder move to "back"/ iFdb(GM); 24(); 24</pre>		<pre>seck_move_command(void) /*/ Climber move Forward, Backward, Left and Right 3/4/00*/</pre>
<pre>diable_ints(); tmpDotF=orf - 75; tmpDotF=orf - 75; //After Initalized () for the forward() if (0) forWarms(OFF) folge(UF) itAnkie(LOCBON);*/ //After Initalized () if () // is that and init side originate parallel to y axis drive the main body move to the front end and delay time determined by d1() function*/ integ(UF);d1();// is test and right side y and the sake and delay d1() time*/ integ(UF);d1();// is test and right side a sake y index move to "short" position and delay d1() time*/ integ(UF);d1();// is test and right side a sake originate move to "inon" position and delay d1() time*/ integ(UF);d1();// is test and right side a sake originate move to "inon" position and delay d1() time*/ integ(UF);d1();// is test and right side a sake originate move to "inon" position*/ if Washet(IOCONF); d1();// Suite in tight side a sake originate move to "inon" position*/ if Washet(IOCONF); d1();// Suite in tight side a sake originate move to "inon" position*/ if Washet(IOCONF); d1();// Suite in tight side y index move to "inon" position*/ if Washet(IOCONF); d1();// Suite in tight side y index move to "inon" position*/ if the d1();// is test and right side y index move to "inon" position*/ if the d1();// is test and right side y index move to "inon" position*/ if the d1();// is test and right side y index move to "inon" position and delay d1() time*/ if the d1();// is the d1 right side is and right side y d1() time*/ if the d1();// is the d1 right side is and right makes () index move to "inon" position and delay d1() time*/ if the d1();// d1();// is the d1 right side y d1() time with press the front and back is a side cylinder move to "inon" position and delay d1() time*/ if the d1();// d1();/ is the and right sake y cylinder move to "inon" position and delay d1() time*/ if the d1();// d1();/ is the and right sake y cylinder move to "inon" position and delay d1() time*/ if the d1();// d1();/ d1();/ the d1() time with the side y d1() time*/ if the d1();/ d1();/ d1();/ d1();/ the d1();/ the d1() time with test or d1()</pre>	int	t tmpRotF, tmpRotB;
<pre>tmpDotFectF - 75; tmpDotFectF - 75; tmpDotFectF - 75; /*After Initialization: lf*Gous(O)):leg(DN)fb*Gous(OFT)fbLeg(UF)!rAnkle(LOCKON);*/ if (commad[3] =* 0;*] / 'Move forward/ moveFed(); d() / 'Lock of the start set set of the start o</pre>	11	((command[1]=='M') && (command[2]=='0') && (command[3]=='V') && (command[4]=='E'))
<pre>tmpDotFectF - 75; tmpDotFectF - 75; tmpDotFectF - 75; /*After Initialization: lf*Gous(O)):leg(DN)fb*Gous(OFT)fbLeg(UF)!rAnkle(LOCKON);*/ if (commad[3] =* 0;*] / 'Move forward/ moveFed(); d() / 'Lock of the start set set of the start o</pre>	(
<pre>tmpDotB=cold - 75 / /*After initialization: if'scald(0)) if'scald(07) fbleg(UD) if'Ankle(LOCKON);*/ if (command(5) == '*'){ / 'Move forward'/</pre>		disable ints();
<pre>/*dfree Initialization: irWess (00) irUeg (00) irVesca (07) irbleg (00) irVekte (LOCON);*/ if (commad(5) == *?!('Wess forward/ moveFwd(); d(); /* Left and right side cylinder parallel to y axis drive the main body move to the front end moveFwd(); d(); /* Left and right side cylinder parallel to y axis drive the main body move to the front end integ(UT);/* Left and right side cylinder parallel to y axis drive the main body move to the front end integ(UT);/* Left and right side cylinder move to "hort" position and delay d()(iime*/ fbbe(GCONO); d()); /* Look the front and back axis end the and elay d()(ime*/ fbbe(GCONO); d()); /* Look the front and back axis or to "hort" position and delay d()(ime*/ ifWeaka(00);d(); /* Look the front and back axis or to "hort" position and delay d()(ime*/ ifWeaka(00);d(); /* Look the front and back axis or to "hort" position"/ moveEv(1); d(); /* Look the front and back site or to "hort" position"/ ifWeaka(00);d(); /* Look the front and back site or to "hort" position"/ ifWeaka(00);d(); /* Look the front and back site or to "hort" position"/ ifWeaka(00);d(); d(); /* Look the front and back site or to "hort" position"/ ifWeaka(00);d(); d(); /* Look the front and back site or to "hort" position"/ ifWeaka(00);d(); d(); d(); d(); d(); d(); d(); d()</pre>		tmpRotF=rotF - 75;
<pre>if (command(5) = **) { /*Move forward*/ moveFwd(); d(1); /* Left and right side cylinder parallel to y axis drive the main body move to the front end and delay time determineds by d() function*/ fbkAkle(LOCKONF); d(1); /* Lock of front and back makle and delay d()time*/ lifteg(UF); d(1); /* Lock of the front and back makle and delay d()time*/ fbKAkle(LOCKONF); d(1); /* Lock of the front and back makle and delay d()time*/ ifteg(UF); d(1); /* Lock of the front and back makle and delay d()time*/ fbKakle(LOCKONF); d(1); /* Lock of the front and back makle and delay d()time*/ ifteg(UF); d(1); /* Lock on the front and back makle and delay d()time*/ ifteg(UF); d(1); /* Lock on the front and back socion copy request and and delay d()time*/ ifteg(UF); d(1); /* Lock on tight side to axis cylinder move to "Bonce" position*/ ifteg(UF); d(1); /* Lock on the front and back socion copy request and delay d()time*/ ifteg(UF); d(1); /* Lock and right side suiton copy relaxes vaccum and delay d()time*/ ifteg(UF); d(1); /* Form and Back side continner to "Bonce" position*/ ifteg(UF); d(1); /* Form and Back side continner to "Bonce" position*/ ifteg(UF); d(1); /* Form and Back side continner to "Bonce" position*/ ifteg(UF); d(1); /* Form and Back side ratis cylinder move to "Bonce" position and delay d()time*/ ifteg(UF); d(1); /* Lock the left and right side ratis cylinder move to "Bonce" position and delay d()time*/ ifteg(UF); d(1); /* Form and Back side ratis cylinder move to "Bonce" position and delay d()time*/ ifteg(UF); d(1); /* Cont the not right side ratis cylinder move to "Bonce" position and delay d()time*/ ifteg(UF); d(1); /* Stath and right side ratis cylinder move to "Bonce" position and delay d()time*/ ifteg(UF); d(1); /* Stath on the left and right suctor cy wacuum*/ ifteg(UF); d(1); d(1);</pre>		tmpRotB=rotB - 75 ;
<pre>noveFwd(); d(); /* left and right side cylinder parallel to y axis drive the main body move to the front end and delay time determined by d() function' fbbakle(LOXOFF); d(); /* Lock off front and back ankle and delay d()(time*/ ifteq(TD); d(); /* Front and back side z axis cylinder move to "hort" position and delay d()(time*/ fbleg(ID); d(); /* Lock on the front and back ankle and delay d()(time*/ fbbakle(LOXODF); d(); /* Lock the front and back ankle and delay d()(time*/ fbbakle(LOXODF); d(); /* Lock the front and back suction cap vacuum and and delay d()(time*/ fbbakle(LOXODF); d(); /* Lock the front and back suction cap vacuum and and delay d()(time*/ if'asian(OFT); d())/ but is and right side suction cap vacuum and delay d()(time*/ if'asian(OFT); d())/* Left and right side suction cap vacuum and delay d()(time*/ if'asian(OFT); d())/* Left and right side suction cap vacuum and delay d()(time*/ if'asian(OFT); d())/* Left and right side suction cap vacuum and delay d()(time*/ if'asian(OFT); d())/* Left and right side suction cap vacuum and delay d()(time*/ if'asian(OFT); d())/* Left and right side suction cap vacuum and delay d()(time*/ if'asian(OFT); d())/* Left and right side suction cap vacuum*/ fbleg(UT); d())/* Left and right side is a six cylinders move to "hort" position and delay d()(time*/ if'asian(OFT); d()); d())/* Left and right side is a six cylinders move to "hort" position and delay d()(time*/ if'asian(OFT); d()); d())/* Left and right suction cap vacuum*/ fbleg(UT); d()(); d() the left and right suction cap vacuum*/ fbleg(UT); d()(); d()(); f'act and make six e suct so cap position cap vacuum*/ fbleg(UT); d()(); d()(); f'act and hack siz e six cylinders move to "hort" position and delay d()(time*/ if'asian(OFT); d()(); d()(); f'act and hack siz e six cylinders move to "hort" position and delay d()(time*/ if'asian(OFT); d()(); d()(); f'act and hack siz e six cylinders move to "hort" position and delay d()(time*/ if'asian(OFT); d()(); d()(); f'act and right side cylinders move to "hort" posi</pre>		/*After Initialization: lrVacua(ON)lrLeg(DN)fbVacua(OFF)fbLeg(UP)lrAnkle(LOCKON);*/
<pre>and delay time determined by d1() function*/ fbakke(LOCKOF); d1() /* Lock of front and back make and delay d1()time*/ lifeg(UF);d1();/* Lock of front and back system verto "short" position and delay d1()time*/ fbleg(UF);d1();/* Fort and back make and delay d1()time*/ integ(UF);d1();/* Store and back make and back make and delay d2()time*/ integ(UF);d1();/* Store on the front and back make and delay d2()time*/ integ(UF);d1();/* Store on the front and back make and delay d2()time*/ integ(UF);d2();/* Store on the front and back make and delay d2()time*/ integ(UF);d1();/* Store on the front and back make and delay d2()time*/ integ(UF);d1();/* Store on the front and back make and delay d2()time*/ integ(UF);d1();/* Lock and right side rais cylinder move to "Short" position*/ integ(UF);d1();/* Lock and right side cylinder move to "Short" position*/ integ(UF);d1();/* Lock and right side cylinder move to "Short" position*/ integ(UF);d1();/* Lock and right side rais cylinder move to "Short" position*, integ(UF);d1();/* Lock the d1; right side rais cylinder move to "Short" position*, integ(UF);d1();/* Lock the d1; right side rais cylinder move to "Short" position*, integ(UF);d1();/* Store and the side rais cylinder move to "Short" position*, integ(UF);d1();/* Store and the side rais cylinder move to "Short" position*, integ(UF);d1();/* Store and back side rais cylinder move to "Short" position*, integ(UF);d1();/* Store and back side rais cylinder move to "Short" position*, integ(UF);d1();/* Store and back side rais cylinder move to "Short" position and delay d1()time*, integ(UF);d1();/* Store and back side rais cylinder move to "Short" position*, integ(UF);d1();/* Store and Back side rais cylinder move to "Short" position*, integ(UF);d1();/* Store and Back side rais cylinder move to "Short" position and delay d2()time*, integ(UF);d1();/* Store and Back side rais cylinder move to "Short" position and delay d2()time*, integ(UF);d1();/* Store and Back side rais cylinder move to "Short" position and delay d2()time*, inte</pre>		<pre>if (command[5] == 'F') { /*Move forward*/</pre>
<pre>tbackle(LOXOFF): d1():/ Lock off front and back ankle and delay d1()tme*/ lifeg(UD):d1():/ Ett and right side I axis cylinder move to "hont" position and delay d1()tme*/ fbleg(UD):d1():/ Ett and right side I axis cylinder move to "hont" position and delay d1()tme*/ fbleg(UD):d1():/ Ett and right side I axis cylinder move to "hont" position and delay d1()+d2()tme*/ lifeg(UD):// Ett and right side I axis cylinder move to "hont" position"/ integ(UD):d1():/ Ett and right side I axis cylinder move to "hont" position"/ integ(UD):d1():/ Ett and right side I axis cylinder move to "hont" position"/ moveRev():d1():/ Ett and right side I axis cylinder move to "hont" position"/ moveRev():d1():/ Ett and right side I axis cylinder move to "hont" position integ(UD):d1():/ Ett and right side I axis cylinder move to "hont" position integ(UD):d1():/ Ett and right side I axis cylinder move to "hont" position integ(UD):d1():/ Ett and right side I axis cylinder move to "hont" position fbleg(UD):d1():/ Ett and right side I axis cylinder move to "hont" position integ(UD):d1():/ Ett and right side I axis cylinder move to "hont" position and delay d1()tme*/ irkAtle(EOXOFF):d1():/ Ett and right side I axis cylinder move to "hont" position and delay d1()tme*/ irkAtle(EOXOFF):d1():/ Ett and right side I axis cylinder move to "hont" position and delay d1()tme*/ irkAtle(EOXOFF):d1():/ Stock the left and right suction cup vacuum*/ fbleg(UD):/ Ettor(I):/ Stock the left and right suction cup vacuum*/ fbleg(UD):/ Ettor(I):/ Stock the left and right suction cup vacuum*/ fbleg(UD):/ Ettor(I):/ Stock and back is axis cylinders move to "hont" position and delay d1()tme*/ irkAtle(EOXOFF):d1():/ Stock the left and right suction cup vacuum*/ fbleg(UD):/ Ettor(I):/ Stock and back is axis cylinder move to "hont" position and delay d1()tme*/ movePud():d2():/ Ettor and Take is a axis cylinder move to the stock and back y d1()tme*/ movePud():d2():/* Ett and right side cylinder parallel to y axis move to the front ent*/ } </pre>		moveFwd(); d1(); /* Left and right side cylinder parallel to y axis drive the main body move to the front end
<pre>integ(TD);d1();/* Left and right side is axis cylinder move to "short" position and delay d1()time*/ fbleg(DD);d2();/* Fort and back side is axis cylinder move to "short" position and delay d1()time which press the front and back p on the surface*/ fblack(DDCOOR);d2();/* Lock the front and back ankle and delay d2()time*/ fblack(DDCOOR);d2();/* Lock the front and back ankle and delay d2()time*/ fblack(DDCOOR);d2();/* Lock the front shoft back surface cop vecous and and delay d1()+d2()time*/ fblack(DDCOOR);d2();/* Lock the front shoft back surface cop vecous and and delay d1()+d2()time*/ if/back(DDCOOR);d2();/* Left and right side vecous and shoft surface vecous and delay d1()+d2()time*/ if/back(DDCOOR);d2();/* Left and right side vecous cop vecous and delay d1()+d2()time*/ if/back(DDCOOR);d2();/* Left and right side vecous cop vecous and delay d1()+d2()time*/ if/back(DDCOOR);d2();/* Left and right side vecous cop vecous and delay d1()time*/ if/back(DDCOOR);d2();/* Left and right side vecous cop vecous and delay d1()time*/ if/back(DDCOOR);d2();/* Left and right side vecous cop vecous and delay d1()time*/ if/back(DDCOOR);d2();/* Left and right sufficient and back sufficient cop vecous and delay d1()time*/ if/back(DDCOOR);d2();/* Lock the left and right sufficient cop vecous and delay d1()time*/ if/back(DDCOOR);d2();/* Lock the left and right sufficient cop vecous and delay d1()time*/ if/back(DDCOOR);d2();/* Rout had back size cop vecous and delay d2()time*/ if/back(DDCOOR);d2();/* Rout had back size cop vecous and delay d2()time*/ if/back(DDCOOR);d2();/* Rout had back size cop vecous and delay d2()time*/ if/back(DDCOOR);d2();/* Rout had back size cop vecous and delay d2()time*/ if/back(DDCOOR);d2();/* Rout had back size cop vecous and delay d2()time*/ if/back(DDCOOR);d2();/* Rout had back size cop vecous and delay d2()time*/ if/back(DD);d2();/* Left and right side cylinder pareallel to y axis move to the front end*/ } </pre>		and delay time determinded by dl() function*/
<pre>fbleg(E0); d2();/* Front and back side z axis cylinder move to "long" poistion and delay d2()time which press the front and back put on the surface*/ fbbackle(LOCKEN); d2(); /* Lock the front and back axile and delay d2()time*/ fbVeca(GN); d2(); /* Lock the front and back surtice our yacuum and and delay d1()+d2()time*/ lifeg(GN); d2(); /* Lock and right side uxile cylinder move to "long" position*/ lifeg(GN); d2(); /* Lock and right side uxile cylinder move to "long" position */ lifeg(GN); d2(); /* Lock and right side uxile cylinder move to "long" position*/ lifeg(GN); d2(); /* Lock and right side uxile cylinder move to "short" position and delay d1()+d2()time*/ lifeg(GN); d2(); Lock and right side cylinder move to "short" position and delay d1()time*/ lifeg(GN); d2(); Lock the left and right side axis cylinders move to "short" position and delay d1()time*/ lifeg(GN); d2(); Lock the left and right side is axis cylinders move to "short" position and delay d2()time which press the front and back p on the surface*/ lifeg(GN); d2(); Lock the left and right suction cup vacuum*/ fbLeg(GN); fCO; d2(); d2(); for toon the lack side side side side succer/ lifeg(GN); fCO; d2(); d2(); for toon the lack side side side succer/ lifeg(GN); fCO; d2(); d2(); d2(); for toon the lack side side side side succer/ lifeg(GN); d2(); d2(); d2(); d2(); d2(); for cont and back side side side side side side side side</pre>		<pre>fbAnkle(LOCKOFF); dl(); /* Lock off front and back ankle and delay dl()time*/</pre>
<pre>on the surface*/ fbMkHe(LOCKON); 42(); * Lock the front and back ankle and delay d2()time*/ fbWacua(GM); 42(); 4(); * Lock the front and back suction cup vacuum and and delay d1()+d2()time*/ fbWacua(GM); 42(); 41(); * Justch con the front and back suction cup vacuum and and delay d1()+d2()time*/ ifteg(UD); * d1(); * Lock and tight side r axis cylinder move to "Loop" position*/ moveRev(); 42(); * Lock and Tapks taide r axis cylinder move to "Mont" position*/ itteg(UD); 42(); * Lock and tight side r axis cylinder move to "Mont" position and delay d1()time*/ itteg(UD); 42(); * Lock and Tapks taide r axis cylinder move to "Mont" position and delay d1()time*/ itteg(UD); 42(); * Lock and Tapks taide r axis cylinder move to "Mont" position and delay d1()time*/ itteg(UD); 42(); * Lock the left and tight side r axis cylinder move to "Iong" position and delay d1()time which press the front and back side r axis cylinder move to "Iong" position and delay d1()time which press the front and back side r axis cylinder move to "Iong" position and delay d1()time which press the front and back side r axis cylinder move to "Iong" position and delay d1()time which press the front and back side r axis cylinder move to "Iong" position and delay d1()time which press the front and back side r axis cylinder move to "Iong" position and delay d1()time which press the front and back side r axis cylinder move to "Iong" position and delay d1()time which press the front and back side r axis cylinder move to "Iong" position and delay d1()time which press the front and back side r axis cylinder move to "Iong" position and back side r axis cylinder move to "Iong" position and delay d1()time which press the front and back side r axis cylinder move to "Iong" position and delay d1()time which press the front and back side r axis cylinder parallel to y axis move to the front end*/ } </pre>		<pre>lrLeg(UP);dl();/* Left and right side z axis cylinder move to "short" position and delay dl()time*/</pre>
<pre>tbhakle(LOCKN); d2(); /* Lock the front and back walke and delay d2()time*/ fbbmass(00); d2(); d1() /* 50xich on the front and back sociclo oup vacuum and and delay d1()+d2()time*/ lifeg(D0); /* Loft and right side r axis cylinder move to "Boher" position*/ lifeg(D0); d2(); /* Loft and right side r axis cylinder move to "Boher" position*/ lifeg(D0); d2(); /* Loft and right side r axis cylinder move to "Boher" position*/ lifeg(D0); d2(); /* Loft and right side r axis cylinder move to "Boher" position*/ lifeg(D0); d2(); /* Loft and right side r axis cylinder move to "Boher" position and delay d1()time*/ lifeg(D0); d2(); /* Front and Back side r axis cylinder move to "Boher" position and delay d1()time*/ lifeg(D0); d2(); /* Loft and right side r axis cylinder move to "Boher" position and delay d1()time vi- lifeg(D0); d2(); /* Loft and right side r axis cylinder move to "Boher" position and delay d1()time vi- lifeg(D0); d2(); /* Loft and right side r axis cylinder move to "Boher" position and delay d1()time vi- lifeg(D0); d2(); d2();</pre>		fbLeg(DN); d2();/* Front and back side z axis cylinder move to "long" position and delay d2()time which press the front and back p
<pre>fbVacua(00);d2();d1();/* Suitch on the front and back suction "up vacuum and and delay d1()+d2()time*/ irideg(00);/* Left and right side suits cylinder move to "Long" position*/ irideg(00);/* Left and right side suits cylinder move to "Biostr" position*/ moveRev(); d2():/* Left and right side suits cylinder move to "Biostr" position*/ irideg(00);/* Left and right side cylinder parallel to y axis move to the back end and delay d1()time*/ irideg(00);/* Left and right side suits cylinders move to "short" position and delay d1()time*/ irideg(00);/* Cont and Back side i axis cylinders move to "short" position and delay d1()time*/ irideg(00);/* Cont and Back side i axis cylinders move to "short" position and delay d1()time*/ irideg(00);/* Cont in the sufface/ irideg(00);/* Cont in the sufface/ fbLeg(00);/* front and Back side i axis cylinders move to "short" position and delay d1()time which press the front and back p irideg(00);/* front and Back side i axis cylinders move to "short" position for fbLeg(00);/* front and Back side i axis cylinders move to "short" position for fbLeg(00);/* front and Back side i axis cylinders move to "short" position for fbLeg(00);/* front and Back side i axis cylinders move to "short" position and delay d2()time*/ fbLeg(00);/* front and Back side i axis cylinders move to "short" position and delay d2()time*/ fbLeg(00);/* front and Back side i axis cylinders move to "short" position and delay d2()time*/ fbLeg(00); d2();/* Left and right side cylinder parallel to y axis move to the front end*/ }</pre>		
<pre>lifeg(DB);/* left and ight side i axis cylinder move to "Long" position*/ lifeg(DB);/# (J);/* left and right side value coll compare left and delay d1()+d2()time*/ lifeg(UB); d2();/* left and right side coll compare life(UB); d2(); lime*/ lifek(ELOCONTF); d1();/* left and right side coll compare life(UB); d2(); lime*/ lifek(ELOCONTF); d1();/* left and right side coll compare life(UB); d2(); lime*/ life(UB); d1();/* left and right side coll compare life(UB); d2(); lime*/ life(UB); d1();/* left and right side coll compare life(UB); d2(); lime*/ life(UB); d1();/* lott and right side coll compare life(UB); d2(); lime*/ life(UB); d1();/* lott and right side coll compare life(UB); d2(); lime*/ life(UB); d1(); d2(); d2(); d2(); d2(); d2(); compare life(UB); d2(); lime*/ life(UB); d1(); d2(); d2(); d2(); compare life(C); d2(); lime*/ life(UB); d1(); d2(); d2(); d2(); life(C); d2();</pre>		
<pre>irVscis(OFF): d2();d1();/' left and right side suction cups release vacuum and delay d1()+d2()/time*/ irLeg(UF);d2();/L left and right side aris of pinder more to "Short" position*/ moveRev();d2();/L left and right side of the side side side side of the side side side side side side side sid</pre>		
<pre>lifeg(UD); d();// if(and right side 'axis cylinder move to "Bhort" position*/ movEvp(); d();/+ if(and right side cylinder parallel to y uxis move to the back end and delay d()time*/ lickakie(LOCKONF); d();/+ laft and right side or links is cylinder move to "long" position and delay d()time*/ lickakie(LOCKONF); d();/+ long to the side axis cylinder move to "long" position and delay d()time*/ lickakie(LOCKONF); d();/+ long to the side axis cylinder move to "long" position and delay d()time*/ lickakie(LOCKONF); d();/+ long to the side axis cylinder move to "long" position and delay d()time*/ lickakie(LOCKONF); d();/+ long to the left and right subtract if the side control to the left and right side cylinder move to "long" position // fileg(UD);/* form and back side c axis cylinder move to "long" position // fileg(UD);/* form and back side c axis cylinder move to "long" position // fileg(UD);/* form and back side c axis cylinder move to "long" position // fileg(UD);/* form and back side c axis cylinder move to "long" position // fileg(UD);/* form and back side c axis cylinder move to "hort" position // fileg(UD);/* form and back side c axis cylinder move to "hort" position // fileg(UD);/* form and back side c axis cylinder move to "hort" position // fileg(UD); d();/* Lift and right side cylinder parallel to y axis move to the front end*/ }</pre>		
<pre>moveBev(): d2():// Left and right side cylinder parallel to y axis move to the back end and delay d1()time*/ irAnkie(EOKONF): d2():// Left and right side cylinders move to "short" postion and delay d1()time*/ iFileg(UP):d1():/* Front and Back side : axis cylinders move to "short" postion and delay d1()time*/ iFileg(UP):d2():/* Lott and right side : axis cylinders move to "short" postion and delay d2()time*/ iFileg(UP):d2():/* Lott and right side : axis cylinder move to "short" postion and delay d2()time which press the front and back p</pre>		
<pre>irknke(LCCCOPF); d2()/* Left and right ankles lock off and delay d2()time*/ Tbleq(DP);d1()/* Forn and Back side i xxis cylinders move to "ahort" position and delay d1()time*/ lifeg(DN); d2()/* Left and right side r axis cylinder move to "long" position and delay d2()time which press the front and back p on the surface*/ irknke(LCCCON); /* Lock the left and right suction cup vacuum*/ fbleg(DN); /* Fornt and back side i xxis cylinder move to "long" position*/ fbleg(DN); /* Fornt and back side i xxis cylinder move to "long" position*/ fbleg(DN); d2(); d2(); d2(); d2(); d2(); fornt and bids is axis cylinders move to "short" position and delay d2()time*/ movePwd(); d2(); /* Eft and right side cylinder parallel to y axis move to the front end*/ }</pre>		
<pre>fbleg(UP);d1();/* From and Back side is axis cylinders move to "short" position and delay d1()time*/ liFeg(DB); d2();/* left and right side is axis cylinders move to "long" position and delay d2()time which press the front and back p intAkle(DoCMD);/* Lock the left and right ankle*/ liFeg(DB);/* front and Back side : axis cylinder move to "long" position '' fbleg(UB);/* front and Back side : axis cylinder move to "long" position '' fbleg(UB);/* front and Back side : axis cylinder move to "long" position'' fbleg(UB);/* front and Back side : axis cylinder move to "long" position '' fbleg(UB);/* front and Back side : axis cylinder move to "long" position '' fbleg(UB); d2();/* front and Back side : axis cylinder ** movePud(); d2();/* Left and right side cylinder parallel to y axis move to the front end*/ }</pre>		
<pre>lifeg(D0); d2():/* Left and right side z axis cylinder move to "long" position and delay d2()time which press the front and back p on the surface*/ irAnkle(LOCKDN):/* Lock the left and right suction cup vacuum*/ fbleg(CDN):/* Front and back side z axis cylinder move to "long" position*/ fbleg(CDN):/* Coro and Back side z axis cylinder move to "long" hostion and delay d2()time*/ moveFwd(); d2():/* Coro and Back side z axis cylinders move to "hont" position and delay d2()time*/ moveFwd(); d2():/* Left and right side cylinder parallel to y axis move to the front end*/ }</pre>		
on the surfaces' irAnkle(LOXCNN);// Lock the left and right ankle*/ irYacua(ON);ad();/* Switch on the left and right suction cup vacuum*/ fbleq(BD);/* front and back side cuts sufjoinder move to "long" position*/ fblequa(ON); dd();/d();/* lock in the front and back suction cups vaccuum*/ fblequa(DD); dd();/* for an and Back side cut and poinder move to "thert" position and delay dd()time*/ moveFwd(); dd();/* Left and right side cylinder parallel to y axis move to the front end*/ }		
<pre>irAnkle(LOCKON);/* Lock the left and right ankle*/ IrYeaus(ON);/* Lock the left and right suction cup vacuum*/ fbleg(CN);/* Front and back side z axis cylinder move to "long" position*/ fbleg(CN);/* Front and back side z axis cylinder move to "short" position and delay d2()time*/ fbleg(CN); d2();/* Lort and Back side z axis cylinders move to "short" position and delay d2()time*/ moveFwd(); d2();/* Left and right side cylinder parallel to y axis move to the front end*/ }</pre>		
<pre>irTessa(00)/d2()/d1()/* Suitch on the left and right suitch out yecumar/ fbteg(UD)/* from and back side x xxis sylindes move to "long" position// fbtegma(OTT)/ d4()/d1()/*Raises the from and back suitchick coupy vecumar/ fbteg(UD) xd1()/* for cont and Back side x xxis (pinders move to "bhort" position and daisy d2()time*/ moveFwd()) d2()/* Left and right side cylinder parallel to y axis move to the front end*/ }</pre>		
<pre>fbleg(DN);/* Front and back side z axis cylinder move to "long" position*/ fbVacua(OFF); d2();d2();d2();d2();d2();d2();d2();d2();</pre>		
<pre>fbVacua(OFF); d2();d1();/*Release the front and back suction cups vaccum*/ fbLeg(UF); d2();/* Front and Back side z axis cylinders move to "short" position and delay d2()time*/ moveFwd(); d2();/* Left and right side cylinder parallel to y axis move to the front end*/ }</pre>		
<pre>fbleg(UP); d2();/* Front and Back side z axis cylinders move to "short" position and delay d2()time*/ moveFwd(); d2();/* Left and right side cylinder parallel to y axis move to the front end*/ }</pre>		fbLeg(DN);/* Front and back side z axis cylinder move to "long" position*/
<pre>moveFwd(); d2();/* Left and right side cylinder parallel to y axis move to the front end*/)</pre>		
}		
		<pre>moveFwd(); d2();/* Left and right side cylinder parallel to y axis move to the front end*/</pre>
if (command[5] == 'B') { /*/Move backward*/)

Figure 4 Extract codes from the control software of nuclear plant inspection robot shown in Fig2

working on a flat surface. It is clear that the smaller "a" would make the robot easy to work on different types of surfaces.

3. The New Robot Design

3.1. Structure Design

To overcome the disadvantages of pneumatic cylinder driven climbing robots, we have designed a new range of climbing robot which are actuated by electric motors. The new design uses rotation to drive robot to move forward or backward. The robot size can be very small and will be much faster and more



Figure 5 Different working surface requires different length of climbing robot legs.

4



Figure 6 The 3D view and side view of the new robot mechanism.

agile compared with the cylinder driven climbing robots.

The overall new robot mechanism is shown in Figure 6. Fig 6(a) is the 3D image of the mechanism. Fig 6(b) is the side view and it shows the large suction cup (FC100P 100 mm diameter) installed.

Four small suction cups (ZPT-UC-50-6-A10, 50mm diameter) are installed to the side legs. The size of the robot platform is 150x108 mm which is much smaller compare with RobAir (518x518mm). There are two degrees, one is the rotation of the legs and one is rotation of the central suction cups. The rotation of the legs will drive the side cups to make a circular movement parallel to the robot main body platform. It will drive the robot move forward or backward. The rotation degree of the central part will be sued to change the direction of the



Figure 7 Examples of the new robot movement sequences.



Figure 8 Control diagram.

robot movement.

6

Figure 7 shows the movement sequences of the newly designed robot vehicle. In Fig 7(a), the side cups vacuum switched on and suck to the surface, the motor driven side leg will move the robot platform forward to Fig. 7(b); continue rotating until the central cup contact the surface. Stop the rotation of side legs and switch on the vacuum of central cup, then switch off the side cups vacuum and rotating the side leg in the same direction.

The platform will stop to move and the leg rotation will cause the side suction cups to move circularly through the status of Fig. 7(d), (e), (f), (g) until (h) when the side cups will touch the surface, then stop the leg rotation and switch on the side cups vacuum, then release the central cup, further rotating the side leg will drive the robot platform to move forward like described earlier. If the robot wants to change the direction, this can be achieved when the central cup is stuck to the working surface. It could be when the robot is in the status like Fig 7(e) or during the leg movement in the air.

3.2. Control system design

The control system (Fig. 8) is based on author's work on the Robtank² and RobAir¹ projects. The on-board DC motor drivers and I/O modules developed in

RobTank project³ are used to control the motors and the valves, and at same time to receive the feedback from sensors. The communication between the host computer and the on-board modules is via RS485 connections. It has been proved to be successful in the RobTank and RobAir projects and also for this system. The advantage of this design has been discussed in paper.¹⁻³

3.3. The new Design

Figure 9 shows the photo taken from different sides of the first prototype of



Figure 9 Photos of the new designed robot in different views.

this robot. The central suction cap has the diameter of 100mm and the side cups each have a diameter of 50 mm. This means that the suction force of side cups and the central cups are almost same. The lifting forces are displayed in table 1, from PIAB's website.⁴ The first prototype weighs less 3kg. It incorporates the robot payload range of 10 - 30 kg which is very good compared with the robots

Performance — lifting forces				
		1		
20 -kPa	137 N	176 N		
60 -kPa	284 N	318 N		
90 -kPa	377 N	420 N		

Table 1 PIAB suction cup FC100P performance table.⁴

mentioned earlier. The tests show that the robot can walk very well on a flat surface and surfaces with small curve.

4. Conclusion

The design of a new structure mechanism climbing robot that uses the DC motor combine with the rotation leg to replace the traditional pneumatic cylinder actuator can make the robot to have a miniature size, which further simplify the moment sequence and able to work in different type of working surface. It can used to replace human being carrying NDT sensors for inspection work in hazard environment. The author are working on develop the portable NDT device which can be carried by this robot. The further improvement of the leg part of this robot is under the way.

References

- Janzhong Shang, Tariq P Sattar, Shuwo Chen and Bryan Bridge Design of a climbing robot for inspecting aircraft wings and fuselage, *Industrial Robot*, 34(6):495-502 • October 2007
- Sattar T.P., Zhao Z., Feng J., Bridge B., Mondal S., Chen S., (2002) Internal In-service Inspection of the floor and walls of Oil, Petroleum and Chemical Storage Tanks with a Mobile Robot, *Proc. Of 5th International Conference* on Climbing and Walking Robots and the Support Technologies for Mobile Machines, ISBN 1 86058 380 6, 2002, pp 947-954, Professional Engineering Publishing Ltd. UK
- 3. Shuwo Chen, Jianzhaong Shang, Zhanfang Zhao Novel Solutions to Design Problems of Industrial Climbing Robots, *Proceedings of the 8th International Conference on Climbing and Walking Robots and the Support Technologies for Mobile Machines, CLAWAR 2005*, London, UK, September 13-15, 2005
- <u>https://www.piab.com/Products/suction-cups/shape/flat-concave/fc---flat-concave-20-150-mm/0103297/</u> (accessed 1st May 2017)

8