INSPECTION OF SURFACES WITH A MANIPULATOR MOUNTED ON A CLIMBING ROBOT

Carsten Hillenbrand⋆ and Karsten Berns⋄

University of Kaiserslautern, Department of Informatics, Robotics Research Lab
Gottlieb-Daimler-Straße, 67663 Kaiserslautern, Germany
⋆cahillen@informatik.uni-kl.de ⋄berns@informatik.uni-kl.de

Speaker: Berns K., Robotics Research Lab, +49 (631) 205 26 56, berns@informatik.uni-kl.de
Topic: Research and Development
Key words: parallel kinematics, robot, inspection, climbing robot,

Abstract

The non-destructive inspection of large bridges with autonomous systems is still an unsolved problem. One of the main difficulties is to develop a very flexible platform, which is able to move at horizontal and vertical areas and can inspect the surface. For precise inspection of concrete walls it is essential to have a manipulator to perform a 2D scan with adequate sensors. In this paper a climbing robot with a manipulation arm will be presented. The platform consists of a vacuum system for the adhesion and omni directional driven wheels for the locomotion. The manipulator base on parallel kinematic techniques and is able to make scans with different types of inspection sensor. For controlling the platform and the manipulator there are coupled algorithm implemented to generate adequate inspection trajectories. The paper ends with test results, done with a manipulator mounted on a vehicle which drives on the floor.

1 Introduction

Almost 90 % of all motorway bridges in Germany are made from reinforced concrete or prestressed concrete. The total number of these bridges are 32 300, the total length is 1197 km. The averaged age is 33 years. To maintain the stability, safety of traffic and durability of these bridges regular inspections are prescribed by law. In Germany the test procedures and inspection intervals are fixed in the German standard DIN 1076. According to this standard inspections have to be performed every three years, main inspections have to be performed every six years and inspections after special occurrences have to be performed after earthquakes, crashes with vehicles, fires and other unusual stresses. To perform the inspection tasks complicated access devices are needed. The aim of our research is the development of a service robot which is moving on the structure and is carrying different sensor types. The inspection of the surface can be done remote-controlled in the same way doing it manually with existing technologies. It is expected that inspection robot will:

− reduce costs of inspections by replacing costly access devices and by avoiding hidden costs caused by closing of traffic lanes
− improve the objectivity and reproducibility of inspections
− enable inspectors to test larger areas of the structures during normal inspection time
− improve the working conditions of bridge inspectors

In literature are different types of climbing robots which can be used for this application. They can be classified by the adhesion system. There exists under pressure system [14], magnetic systems [2], mechanical gripper [1], vertical force generator with air propeller [9] and the gecko principle which practice the van der Waals adhesion [3]. Another kind of classification is the locomotion system. There are the three different types of wheel driven robots [7], chain driven robots (FHG/IPA: Raccoon) and legged machines [8]. For climbing on concrete surfaces the vacuum system driven by wheels seems to be the best solution, because of fast and continuous motions. Furthermore the mechanical construction keeps simple.

Our mechanical concept of the climbing robot itself consist of a big vacuum camber which is slipping over the concrete (Figure 1). Three omni directional driven wheels are pushing the robot over the surface. More details for the vacuum system can be found in published papers [4][6][13].

For the inspection of the surface different sensors can be used. The cover meter is the most common method to detecting rebars in the concrete. It is working with an alternating magnetic field. Another sensor is an impulse...
radar which detects holes inside the concrete. These were made for wires inside of prestressed concrete bridges. And of course a camera which must detect cracks of 0.2mm width. Such cracks are indicating an overload of a bridge. An overview of sensor technologies which can be used to inspect concrete walls and which are adapted to be installed in a climbing robot is introduced in [12].

![Fig. 1. The climbing robot with the vacuum system and the drive](image)

For the inspection task of the climbing robot some requirements must be fulfilled for the manipulator and the platform.

- The manipulator must be able to move in lines
- Different inspection sensor should be changed by a tool changer
- The construction of the manipulator is restricted by the design concept of the climbing machine
- Because of the characteristic of the inspection sensors electrical components and metal should be very far away
- The weight of the manipulator should be very low to reduce disturbances caused by torque an forces to robot vehicle
- A control strategy must be implemented to coordinate motions of manipulator and mobile platform for the inspection process.

In the following our manipulator concept is introduced with the above mentioned requirements. Also the control approach will be described which allows a adequate coordination of the manipulator and a wheel driven platform for solving different inspection tasks.

2 The mechanical construction of the manipulator

In literature one can find different solutions for manipulators able to move in 2D space like scare and cartesian robots. Based on the requirements like precision of the manipulator, stiffness, large workingspace and lightweight construction, a parallel kinematic is selected with two degree of freedom (figure 2). Additionally rotating and lifting units of the TCP is installed, which allows adaptation of the sensor box due to the surface. Totally the manipulator has four degree of freedom and is developed to scan the local area around the climbing robot. Most important for such a manipulator is that the center of gravity is closed to the center of the robot case. Otherwise high torques generated by the manipulator must be balanced by the drive system of the robot permanently.

A solution is to use round guides with two driven sledges which realise the motions in the x,y plane. Therefore, on each sledge arm segments are mounted with passive bearings. They are merged together at the head. The left arm is fixed connected to the head and the right one is connected with a joint. Such a solution has the advantage, that the space around the center of the robot is still usable for other robot components. The disadvantage is that the implementation of the motion control algorithm are more complex because of the coupled drives for movement in x or y direction. To lift and rotate the inspection sensor there are two additional drives in the head. One ball-screw nut is moving the stick up and down and a spline nut is rotating the stick. At the bottom of the stick were the TCP is placed, the inspection sensor is mounted. Unfortunately both joints are coupled too. If the sensor for example shut rotate without changing the distance to the surface, both drives have to be controlled in parallel. The motors are not place in the head, because of the electromagnet influence to some inspection sensors (magnetic field sensor). They are fixed close to the sledges in the arm segments and are connected via toothed belt (installed inside the arm segment) to the head.
The head is produced out of light plastic and equipped with position sensors HEDS-9040 from Agilent for both joint (Figure 4). The arms segments are made of composite material (glass fiber) and includes guide roller for the belts. The arm segments are bended to keep the space in the center of the robot free. The round guide is build up like a linear bearing but with an radius of 300mm and is available from THK. The drives of the sledges are realized with belts, because cogwheels with diameters of 600mm are too heavy and have rough cogs. Therefore, the belt is placed with the back side in a guideway and the drive is bringing the belt out and back to the guideway. During this operation one cogwheel is driven by a motor (Figure 5+ 6). Additionally a separate position sensor EM1-0-127 from US Digital is placed in the back of each sledge. For the future it is planed to use a special coded signature to get an absolute position value if the drive is moving a short distance. Inside of the base ring there is space for dragchains to realise the electrical connections between the robot case an the controlling electronic, which is placed on the sledge.

The selected DC motors, the gear reduction and the operating interval for each joint are be listed in the following table. The position of the sledges is described in degree, with means the angel inside the round guide with origin in the center of the robot. The mechanical realisation fulfills the above mentioned requirement. Different orientated line-scans inside the workspace can be done (Figure 3). In futures a tool change mechanism can be added to the stick at the head of the manipulator and bags round the robot case can be created to save different sensors. The disadvantages of coupled joints must be compensated by a adequate control architecture. In the following a short introduction into the soft- and hardware architecture on the climbing robot is given.
<table>
<thead>
<tr>
<th>Joint</th>
<th>Motor</th>
<th>gear reduction</th>
<th>operating interval</th>
<th>Max Velocity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Right Sledge</td>
<td>Faulhaber 2342</td>
<td>18,04+43=775,72</td>
<td>[98°, -110°]</td>
<td>8.8°/s</td>
</tr>
<tr>
<td>Rotation Head</td>
<td>Faulhaber 2224</td>
<td>66</td>
<td>[360°, 360°]</td>
<td>7.5°/s</td>
</tr>
<tr>
<td>Left Sledge</td>
<td>Faulhaber 2342</td>
<td>18,04+43=775,72</td>
<td>[110°, -98°]</td>
<td>8.8°/s</td>
</tr>
<tr>
<td>UpDown Head</td>
<td>Faulhaber 2224</td>
<td>0,0189mm / Rotation</td>
<td>[0mm 80mm]</td>
<td>2mm/s</td>
</tr>
</tbody>
</table>

3 Hard- and Software architecture

The hardware consists of several parts as shown in Figure 7. The digital signal processor unit (circuit board) consists of a DSP (Freescale 56F803) which is cooperating with a logic device (CPLD - Altera EPM70256). Additionally there are current supply element, CAN-bus element and amplifiers for the motors. This digital signal processing unit is able to preprocess different types of encoders and controlling up to two drives. Several DSP units are connected via CAN-bus to an embedded PC which is placed inside the robot case. The calculation of movements of the kinematics is done on this Linux-PC.

![Fig. 7. The structure of the hard- and software](image1)

![Fig. 8. The controller structure for one drive](image2)

The manipulator control is implemented with the help of the Modular Controller Architecture (MCA). MCA is a modular, network transparent and real-time capable C/C++ framework for controlling robots (see [10] for details). MCA is conceptually based on modules and edges between them. Modules may be organized in module groups, which allow the implementation of hierarchical structures. The tools MCAGUI and MCAbrowser allow the interaction with the system. MCAGUI is used for developing GUIs for embedded systems. Embedded systems often have no graphical or textual output. It communicates only via messages which were send over buses. The tool MCAGUI examines these messages and delegates them to the specific components of the user specific GUI. If the user wants to send specific messages he can use different graphical components of the MCAGUI. Every input component reacts to a user input and automatically sends the message via bus. The tool MCAbrowser is used to show the structure of running MCA programs including parameters and control flow. A special aspect of the framework is the automatic support of the network as a communication platform. Every program which was developed with the help of the framework automatically communicates over the network. A special class is responsible for the communication with the digital signal processor.

The controller architecture for the manipulator and omnidirectional vehicle drive consists of independent close loop controller on the DSP’s. The controlling inputs are beside the position of each drive additionally the velocity. That’s necessary to manage the coupled joints and different position /velocity sensor places. This will avoid bucking motions. The kinematic calculation module is running on the PC and is transforming working space coordinates for position and velocity to joint or drive coordinates.
4 The close loop control

For the motor control of the platform and the manipulator a controller is presented in Figure 8. The controller itself consists of a cascaded closed loop velocity controller and a closed loop position controller. The velocity controller is an advanced PI-Controller, where the proportional value for desired and actual inputs are separated. This type is called ReDus unit and has the advantage, that only with the proportional element frictions of the motors can be compensated and the desired input can be reached without having a difference to the actual input. The integral element is only responsible to compensate external forces to the joint. This concept makes the controller quicker and more stable [11]. The superposed position controller is combined with an anticipation element. That means, that known velocity of the path calculation is an direct input to the speed controller. Differences in the position must be compensated by the position controller. No difference between the position actual and desired input must be build up to get a higher desired speed. The drag distance between desired and actual position value is shorter. This type of controller support different mounted positions of velocity and position sensor. Certainly this works only, if the desired values from the topper modules are correct and are coming continuously.

5 The kinematic model of the manipulator

To control the described manipulator, it is important to solve the direct and inverse kinematic problem. To calculate the direct Kinematic problem homogeneous coordinates can be used. One matrix (4x4) is transforming the coordination system of the tool center point (TCP - \((x, y, z, 1)_{TCP}\)) to the robot coordination system (\((x, y, z, 1)_{r}\)) by rotating and translating. The matrix can be generated by using the Denavit-Hartenberg convention by generating a matrix for each joint [5]. The joint and the names of the variables can be seen in figure 10.

\[
\begin{pmatrix}
 x \\
 y \\
 z \\
 1
\end{pmatrix}_{Robot} = \begin{pmatrix}
 \cos(\theta_1) - \sin(\theta_1) & 0 & R \cdot \cos(\theta_1) \\
 \sin(\theta_1) & \cos(\theta_1) & R \cdot \sin(\theta_1) \\
 0 & 0 & 1 \\
 0 & 0 & 0 & 1
\end{pmatrix} \cdot \begin{pmatrix}
 \cos(\theta_2) - \sin(\theta_2) & 0 & L \cdot \cos(\theta_2) \\
 \sin(\theta_2) & \cos(\theta_2) & L \cdot \sin(\theta_2) \\
 0 & 0 & 1 \\
 0 & 0 & 0 & 1
\end{pmatrix} \cdot \begin{pmatrix}
 x \\
 y \\
 z \\
 1
\end{pmatrix}_{TCP}
\]

The same can be done for the left arm. If the angles for each joint are known the position of the manipulator can be calculated by fitting the position \((0, 0, 0, 1)_{TCP}\) for the local coordination system of the tool center point (TCP). That’s the procedure for the direct kinematic. But if the angles must be calculated (inverse kinematic) the equation must be solved for each angle depending on its position. For such kind of planar manipulator it is easier to solve the inverse kinematic based on the geometry. The distance \(r_M\) and angle \(\alpha_M\) (figure 10) for the TCP with the origin in the middle of the robot can be calculated with

\[
r_m = \sqrt{x_m^2 + y_m^2} ; \quad \alpha_M = \arctan\left(\frac{y_m}{x_m}\right)
\]
With the angle difference of both sledges $\Delta \alpha_{LR}$, the angles for each sledge can be calculated by adding and subtracting the half of the value from the middle angle to the TCP.

$$\Delta \alpha_{lr} = 2 \cdot \arccos \left(-\frac{L^2 + R^2 + r^2}{2 \cdot R \cdot r_m}\right); \quad \alpha_l = \alpha_m + \frac{\Delta \alpha_{lr}}{2}; \quad \alpha_r = \alpha_m - \frac{\Delta \alpha_{lr}}{2}; \quad (3)$$

Combining the equations for the right arm segment, a closed formula can be found.

$$\alpha_r(t) = \frac{\arctan \left(\frac{y_m(t)}{x_m(t)}\right)}{\arccos \left(-\frac{L^2 + R^2 + x_m(t)^2 + y_m(t)^2}{2 \cdot R \cdot \sqrt{x_m(t)^2 + y_m(t)^2}}\right)} \quad \left(4\right)$$

This equation can also be used for calculating the velocity. Therefore, the derivation by the time must been determined, which is possible but is unsightly and will not be presented in detail. The direct and inverse kinematics is able to transform the global velocity and position into the joint values. The same must be done for the platform.

6 Kinematic model of the omni directional platform

For the locomotion system of the climbing robot three driven wheels are the best solution, because all adhesion forces can be used for pushing the wheel to the surface and three wheels needs no springs which will cause stability problems. The disadvantage of this drive is the increased effort for the control software. Therefore a kinematics module must be developed.

For the control of our omni directional platform a given trajectory the velocity vector $v_r$ and the rotation velocity $\omega_r$ of the robot $(r)$ is given as input. The rotation velocity and the rotation of each wheel is given as an output. Given parameters are the radius $R_w$ and the angle $\Phi_{wx}$ of the position of each $(x = 1, 2, 3)$ omnidirectional wheel $(w)$ (see figure 11 ). The task formally of the inverse kinematics is the calculation of the velocity $v_{w1,2,3,x}$ and the orientation $\phi_{w1,2,3}$ of each wheel as a function of the inputs $f(v_r, \omega_r)$. The path which the robot will drive can be described as a circle path with a pivotal point $P$. This point must be on the line that is perpendicular to the direction of the velocity vector $v_r$, because the given velocity represents the tangential direction of the circle in the kinematic point $K$.

Possible paths are dotted, but the right one is defined by the rotation velocity $\omega_r$. If the robot drives a certain time $\Delta t$ the driven distance is $s_m = \Delta t \cdot |v_r|$ and the angle of the segment of the circle is $\varphi_m = \Delta t \cdot \omega_r$, because the angular velocity of the robot itself is the same as the rotation velocity around the rotating point $P$. The radius for the path and the angle to the center point $P$ can be calculated by

$$r_p = \frac{s_m}{\varphi_m \cdot \left(\frac{180}{\pi}\right)} = \frac{\Delta t \cdot |v_r|}{\Delta t \cdot \omega_r} \cdot \frac{180}{\pi} = \frac{|v_r|}{\omega_r} \cdot \frac{180}{\pi}; \quad \varphi_p = \arctan \left(\frac{v_{r,y}}{v_{r,x}}\right) + 90^o \quad \left(5\right)$$

With the calculated pivotal point the velocity and the orientation of each wheel can be calculated using the geometrical formulas 6, 7 and 8 (see figure 12):
Fig. 13. Calculation of the wheel orientation

\[ v_{w1,2,3,x} = \left| \mathbf{v}_r \right| \cdot \frac{r_{w1,2,3}}{r_p} = \frac{\left| \mathbf{v}_r \right|}{v_{w1,2,3} \cdot \omega_r \cdot \frac{\pi}{180^\circ}} = \frac{r_{w1,2,3} \cdot \omega_r \cdot \frac{\pi}{180^\circ}}{\pi} \] (6)

\[ r_{w1,2,3} = \sqrt{R_w^2 + r_p^2 - 2R_w r_p \cos (|\Phi_{w1,2,3} - \varphi_p|)} \] (7)

\[ \varphi_{w1,2,3} = \arcsin \left( \frac{r_p}{r_{w1,2,3}} \cdot \sin (|\Phi_{w1,2,3} - \varphi_p|) \right) + \varphi_{w1,2,3} + 90^\circ \] (8)

Unfortunately, there are two special cases for this calculation. In the first case the robot is driving straight ahead without any rotation \((\omega_R = 0)\). The wheels are pointing at the same direction and have the same velocity, but the result for \(r_p\) from equation 5 is infinite and therefore not reasonable. By examining the equations there exists the solution 9.

\[ \omega_r = 0 \implies r_{w1,2,3} = 1 \implies v_{w1,2,3,x} = \left| \mathbf{v}_r \right| ; \varphi_{w1,2,3} = \varphi_p + 90^\circ \] (9)

In the second case the robot is only rotating around itself and there exists no velocity \((|\mathbf{v}_R| = 0)\). This means that the wheels are tangential to a circle with its origin in the middle of the robot \((P\) and \(K\) are at the same position). Also in this case a solution of the equations can be found:

\[ |\mathbf{v}_r| = 0 \implies r_p = 0 ; r_{w1,2,3} = R_w ; \frac{r_p}{r_{w1,2,3}} = 0 \]

\[ \implies v_{w1,2,3,x} = R_w \cdot \omega_r \cdot \frac{180^\circ}{\pi} ; \varphi_{w1,2,3} = \varphi_{w1,2,3} + 90^\circ \] (10)

With the presented equations the inverse kinematics can be calculated. For odometric informations the system needs additionally the direct kinematics to calculate the direction and velocity of the robot from the existing orientations and velocities of the wheel. As this problem is over-determined, it has to be solved by an optimization strategy. This is part of current research.

7 Test results

The mechanic of the manipulator was assembled to a vehicle with has complete omnidirectional functionality. The final locomotion system is under construction, so tests can be executed on the floor only. The controller and the mechanics are working well and after adjusting the controller parameter the manipulator and the omnidirectional locomotion system are working smooth. Some additional algorithm are implemented to keep the manipulator mechanism inside the working space. A line generator is planing the trajectory for the inspection sensor with constant velocity. The algorithm for the combined motions of the platform and manipulator were implemented and the results shows that scanning with combined motions are possible. That’s an efficient way to scan large areas very precise, because of correction the position with the manipulator during the motion of the vehicle and without stopping the scan process. An Overview about the robot can be seen in figure 14.
8 Summary

A manipulator was built up to realise the special tasks of scanning surfaces with inspection sensors. All requirements were considered which caused more complicated controller and software algorithm. But nowadays there is no reason to avoid more effort on the software side. The presented parallel kinematic was the best solution for this task and the calculations for the kinematics was clearly presented with geometrics methods. The next steps of our research is the integration of the drive and manipulator system into the seven chamber under pressure system. For the under pressure system new seals is under development. The control concept will be increased according to the detection of objects and forbidden areas. Also the user interface should be improved to allow a easy definition of the inspection task.

Acknowledgment

The research work presented in the paper is sponsored by "Stiftung Rheinland Pfalz für Innovation"

References