Development of Sensitive Skin for a 3D Robot Arm Operating in an Uncertain Environment*

Edward Cheung and Vladimir Lumelsky

Yale University, Department of Electrical Engineering
New Haven, Connecticut 06520

Proceedings of 1989 IEEE International
Conference on Robotics and Automation
Scottsdale, Arizona, May 1989

Proceedings of 1989 IEEE International
Conference on Robotics and Automation
Scottsdale, Arizona, May 1989

1

2

3

4

5

6

* Supported in part by the National Science Foundation Grant DMC-8712357, and by a grant from the North American Philips Corporation.

Abstract

Sensor-based robot motion planning in an uncertain environment, especially for autonomous vehicles, has attracted much attention recently. Its natural extension to 3D arm manipulators calls for a capability to sense an approaching obstacle by any point of the robot body. This paper describes an on-going research project in our laboratory on the development of a sensitive skin and its control scheme for a 3D robot arm. The skin consists of hundreds of active infra-red proximity sensors that cover the whole arm body. Details of the skin design are presented, along with the computer hardware and operating logistics, and the sensor data interpretation algorithm that connects this subsystem to the global motion planning subsystem.

Introduction

The problem of the sensor-based control for an autonomous vehicle operating in an unknown environment has been a topic of active research in recent years (see, e.g. [1-3]). A natural extension of this problem is to collision-free motion of three-dimensional (3D) robot arm manipulators operating in workspace with unknown obstacles. One approach here is to cover the surface of the robot arm with an array of obstacle sensitive sensors. Obstacles appearing from any direction can then be detected, and appropriate action taken to avoid collision. This approach had been found successful in the preliminary 2D feasibility study [4]. In that version, only two degrees of freedom were used, and only those sides of the arm that could interact with obstacles during planar motion were covered with sensors. It was shown, specifically, that with no previous knowledge about the obstacles and using only sensor data, the arm was capable of moving to a given target position if this were possible, or conclude in finite time that no path existed if such were the case.

This paper describes the next stage in this project and is concerned, specifically, with design of what we call the sensitive skin of the robot and its incorporation into the control and 3D global motion planning subsystems. The skin is composed of many individual optical proximity sensors that are organized in the form of a grid on the surface of the arm. The skin covers all the areas of the arm that might in principle come in contact with obstacles. This includes practically the whole arm body, including the wrist. The density of sensors on the skin is chosen such that no obstacles can approach the arm without being sensed by the sensors. No actual contact with objects in the environment takes place during the operation, as the sensors use reflected light to sense obstacles; the sensing distance of each sensor is from 3 to 10 inches. In the current implementation, 500 sensors blanket the robot arm, a General Electric P5 robot. The motion planning system that controls the skin operates with the arm major linkage - that is, with the first three arm joints. The remaining wrist joints are not controlled directly, although the collision-free motion of the body of the wrist is still guaranteed.

The sensitive skin and the hardware and software that control it present the lower level subsystem of the motion planning system. Also on this level is a module that plans small steps at every moment when the arm is moving toward its target position in free space or is maneuvering in 3D space around an obstacle. In the latter case, the step planning algorithm must guarantee that, first, no collision with the obstacle takes place, and second, a "contact" with the obstacle is maintained at the end of the step. The terms "contact" and "contact point" here refer to the situation when an obstacle is within the sensitivity range of one or more skin sensors.

Below, we first describe the sensor system, then give a brief review of the computer system that controls it, and finally discuss the step planning method. For lack of space, no proofs of statements related to the step planning algorithms are given here. Also, no details of the global motion planning strategy to be used with the described system are provided; for those, see [5].

The Sensor System

Each sensor in the 500 sensor array consists of a pair of elements; one element emits infra-red (IR) light, whereas the other element receives the reflected light. The light is amplitude modulated to enhance immunity to other IR sources, and to other sensors. The latter characteristic is important for the case when the light of one sensor's transmitter falls on the receiver of a sensor on the other link. The links on the articulated arm that are covered by sensors are l_1, l_2 , and the wrist, Figure 1. Sensors are time multiplexed to prevent cross-talk between sensors on the same link. Also, sensors on each link modulate the transmitted light at different frequencies to prevent cross-talk between links. This way, light from the transmitter of a sensor is detected by only its associated receiving section.

The output of the sensor is proportional to the amount of reflected light returning to the receiver, and provides for an analog indication of obstacle proximity. This output signal is affected by the color, shape and size of the obstacle.

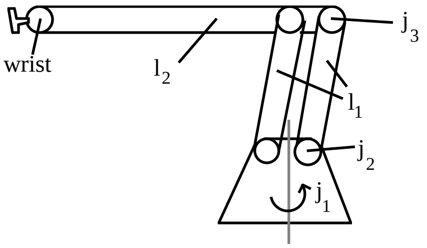


Figure 1. Sketch of robot arm

The infra-red light is modulated at 135 kHz for link l_2 and 67.5kHz for link l_1, and transmitted by the infra-red emitting diode (IRED). The two frequencies are kept at a constant 2:1 ratio so that light emitted from one link's sensor does not interfere with the sensor of the other link.

The detection and processing of the reflected light is done in the detection circuit, Figure 2. There is one detection circuit per link. First, the reflected light is converted to a current by a PIN diode, Figure 2. Both selectors S1 and S2 select adjacent optical component pairs, and consist of several analog multiplexers cascaded together. Each analog multiplexer chip selects one of 16 analog signals, and is controlled by a four bit input. It also features an 'inhibit' input, so that any number of analog signals can be selected by connecting the output of several chips together, and then applying a logic 'low' to the 'inhibit' input of one chip, and a logic 'high' to the rest of the chips [6].

The output signal of S2 is amplified by the operational amplifier (op-amp) A1. The high pass filter consisting of op-amp A2 and associated components remove a large portion of the ambient room lighting at DC and 60 Hz. The transmitted signal is then synchronously detected by op-amp A3 to demodulate the high frequency signal of the LED down towards DC. Lastly, the signal is filtered by the third order Butterworth filter composed of op-amps A4 and A5 [7].

The synchronous detector consists of op-amp A3 and associated components and functions by alternating its gain from +1 to -1 depending on whether the junction transistor Q_1 at its noninverting input is on or off. This switching is in step with the flashing of the IRED, causing frequencies at the odd harmonics of the switching frequency to be highly rejected. This results in a high degree of optical isolation between the sensors of each link.

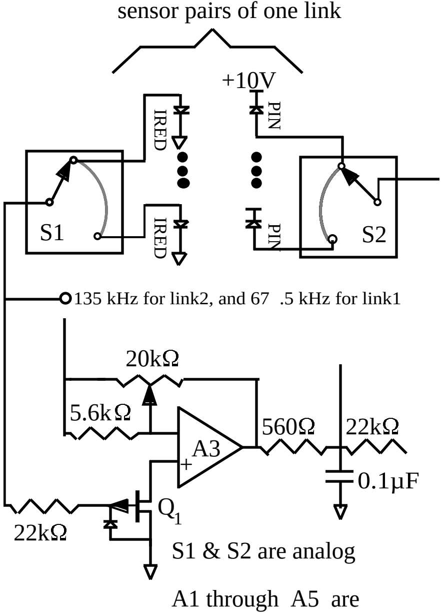


Figure 2. Schematic of the detector circuit for one link

The sensor that we used before in [4] also had a synchronous detector to filter out undesirable components in the received signal, but the detection was accomplished by a different circuit. With the new circuit, as many as 300 sensor pairs share a single detector circuit in our current sensor system; in contrast, only 16 sensor pairs shared one detector circuit in [4]. The new detector circuit is more compact, making it more suitable for use with the sensor system since the electronics can occupy little space

The optical components (LED and PIN diodes) are mounted on the surface that constitute the sensitive skin covering the robot arm. The surface is made of a novel material manufactured from a capton compound, and is only 0.0085" thick. It is cladded with two layers of copper, one on each side, and is very flexible. Physically, the material behaves similar to light gauge sheet metal; electrically, it behaves similar to a double sided printed circuit board. It is processed as an ordinary copper cladded board, and the optical components are soldered onto it. This material thus provides for both electrical connection to all the electronic components, as well as for structural support. The skin is made from three sheets of this material, the largest being a section 2 feet by 4 feet that covers link l_2 and the wrist, and two sheets that cover the two parts of link l_1, Figure 3. Electronics that controls the optical components, Figure 2, are contained on a card between the skin and the robot arm.

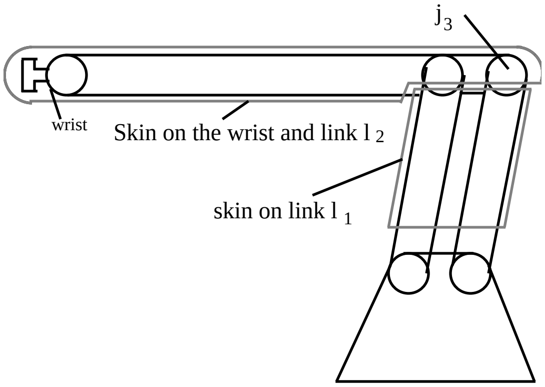


Figure 3. Sketch of the sensitive skin on the arm.

One challenge in covering a robot arm with a sheet of material is the covering of the joints. The human skin stretches and contracts, while maintaining its sensing capabilities, a feature that is difficult to imitate in a sheet of material that incorporates electronics and sensors. Different types of joints, such as revolute or prismatic, may require different solutions. Our solution for the General Electric P5 robot is shown in Figure 3 where the part of the skin covering link l_2 and the wrist has an opening in the sensor material at joint j_3. The sheet of the skin that covers the part of link l_2 near the joint slides above a similar sheet that covers link l_1.

Another challenge in designing the flexible skin relates to the wires that control the skin sensors and run along the arm. These cables have to sustain flexing and must not protrude from the arm, to avoid the possibility of collision. One way to contain the problem is to have as few wires running across the joints as possible. From this standpoint, in a system with 500 sensors, it is not desirable to have a multiple of 500 wires connecting the sensor system. There is a conflicting requirement, however: the system needs to be able to address each sensor individually, because we wish to know where obstacles are occurring with respect to the robot arm.

To meet these requirements, a serial clocking scheme is incorporated into the addressing mechanism. As was mentioned before, the circuitry shown in Figure 2 is contained on a card located between the sensor skin and the robot link. In addition to this circuit, a nine bit binary counter is also contained on the card. On every increment of this counter, a different sensor pair is enabled and made to sense for the presence of an obstacle. The counter is clocked by the microprocessor that processes the sensor data, and is reset by a 'high' pulse on the same signal line that clocks the counter. The result is that a virtually infinite number of sensors can be addressed in a pseudo random fashion using only one signal wire.

As a result of this serial data i/o (input and output), the sensor link is monitored through only five wires, laid across the joint connecting links l_1 and l_2. These include the following signals: sensor select in, analog proximity data out, IRED modulation frequency, and power supply.

Computer Hardware

The computers that control the sensitive skin and motion planning subsystems are T800 Transputers from the Inmos Corporation. Each is a 32 bit machine with an on-chip floating point processor and 4 high speed serial direct memory access (DMA) lines. The computing power of this machine is about 1 million scalar floating point operations per second (1 MFlop). In the current implementation, two such micros are used: one for handling sensor and robot interface, and the other for handling user interface and path planning. The development station is an IBM compatible containing a Transputer that runs the editor, compiler, and the Transputer network server. Supervising the whole system is a MicroVax station which does necessary documenting and graphics displays of the experimental data.

Step Planning Algorithm

The overall path planning system is hierarchical, consisting of a global planner that determines the overall strategy, and the local step planner that is used for sensor data interpretation and for planning the smallest steps that constitute the arm's global motion. The objective of step planning is to either move the arm toward the target position in free space along a prespecified trajectory (e.g., a straight line for the end effector), or maneuver it around an obstacle, while keeping in close proximity to it. The former mode takes place when the skin senses no obstacles, whereas the latter mode takes place when at least one skin sensor senses an obstacle.

Since controlling motion in free space is quite straightforward, we will concentrate now on step planning in the vicinity of obstacles. We present this process in terms of the configuration space of those joints that are being controlled. When the sensor system informs the global planner that it encountered an obstacle, the latter prescribes the arm to move along the intersection curve between the obstacle and a certain plane. Since the intersection curve itself is unknown, it is the task of the step planner to sense it and to maneuver the arm along the curve.

The problem of interpreting the data from 500 proximity sensors is handled by a method that presents an extended 3D version of the 2D algorithm described in [4]. Input data to the step planning algorithm is provided by the sensor system; the location of each sensor that senses an obstacle relative to the arm body is determined with a look-up table. In contrast to tactile systems, such as [8], the arm does not rub against the obstacle, but slides instead along the obstacle's surface, while keeping at some distance from it. Using the sensor data and the step planning algorithm described next, surface contour following occurs in a collision free fashion. For each sensor that contacts an obstacle, the conditions necessary to slide about that sensor are determined. Specifically, the local normal of the tangent plane at the contact point between the robot and the obstacle in configuration space is found. Motion in this tangent plane is considered to be the safest motion possible. If more than one sensor are simultaneously in contact with obstacles, more than one tangent plane appear, and a choice must be made as to which of these planes is the safest. New local normals are repeatedly calculated after each motion of the robot arm.

Since only the first three degrees of freedom of the robot are used for active control, the motion of the robot can be represented by a point automaton that operates in the three-dimensional configuration space $(\Theta_1, \Theta_2, \Theta_3)$ where Θ_i are the angular joint variables. Correspondingly, any obstacle in the work space has its image in the configuration space. Suppose that the arm is in contact with an obstacle. This corresponds to the automaton being positioned on the surface of that obstacle's image in the configuration space. A tangent plane can be defined at the point of contact between the robot and the obstacle. Now the automaton should move along this tangent plane to minimize the likelihood of collision with the obstacle. If the size of the step is chosen so that after its completion the arm is still positioned within the area that is currently known (based on the sensor data) to be free of obstacles, the motion is guaranteed to be collision-free. This tangent plane, an essential factor in surface following, is defined by its normal vector and can be found from the location of the obstacle with respect to the arm.

For every sensor that encounters an obstacle, a local normal is calculated. Thus, if several sensors are detecting obstacles, a bundle of local normals will be generated. Choosing a specific direction of motion among the generated contact planes is done by the global path planning algorithm, which is not discussed in depth in this paper. The corresponding tangent planes are intersected with the plane prescribed by the global planner. The resulting tangent lines can then be treated similar to the two-dimensional case considered in [4].

The procedure for local normal generation described below assumes robot links of zero thickness (Figure 4). In spite of this, the step planning algorithm can be used with real robots. The reason for that is that the sensor system provides an analog proximity reading. This information is then used to achieve better contour following of the obstacle, similar to the rotation of the calculated local tangent used successfully in [4]. By using the sensor readings continuously, as the obstacle is being followed, the local normal can be adjusted to compensate for the approximations used during the local normal calculation process.

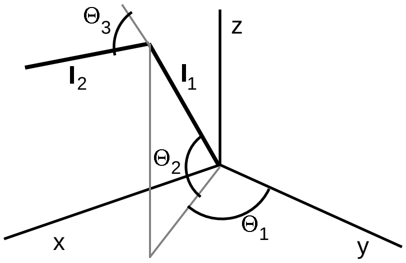


Figure 4. Sketch of robot arm

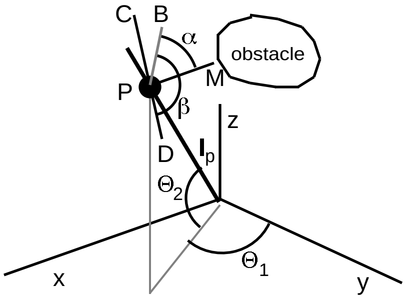


Figure 5. Type I obstacle

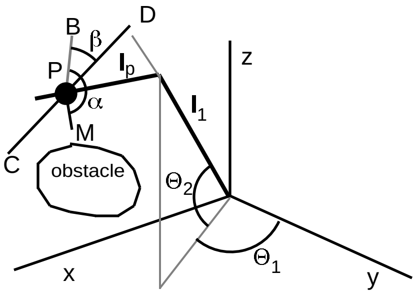


Figure 6. Type II obstacle

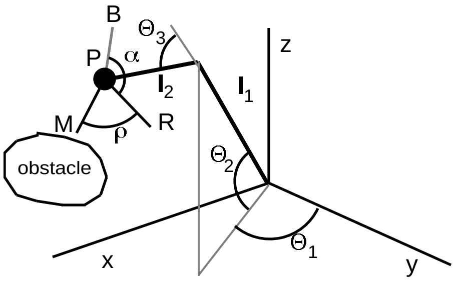


Figure 7. Type III obstacle

We turn now to the procedure for computing the normal vector of an individual tangent plane. Depending on their location with respect to the arm, obstacles can be grouped in three categories easily recognized by the sensor system: Type I, Type II, and Type III. Hereafter, the following substitutions will be made to enable a more compact notation

$$c_i = \cos \theta_i \text{ , } s_i = \sin \theta_i \text{ , } c_{(i-j)} = \cos(\theta_i - \theta_j);$$

$$s_{(i-j)} = \sin(\theta_i - \theta_j) \text{ , } i = 1,2,3;$$

$$c_\alpha = \cos \alpha \text{ , } s_\alpha = \sin \alpha, c_\beta = \cos \beta \text{ , } s_\beta = \sin \beta.$$

Type I are those obstacles that obstruct link l_1, Figure 5. Since only link l_1 is obstructed, we temporarily ignore link l_2. Point P is the location of a sensor on the arm that is sensing an obstacle, and PM represents the sensing axis of that particular sensor, where PM is perpendicular to l_1. Line segment PB is perpendicular to l_1 and lies in the plane that contains the z-axis and link l_1. Let ß denote the angle between lines PB and PD, and l_p - distance between the origin and point P along the link l_1. Safe motion of the arm for a Type I obstacle in work space is in a direction perpendicular to l_1 and along the line CD. The local normal to obstacles of this type can be found using the following procedure.

Using forward kinematics, we find the vector of coordinates of P:

$$P = \begin{pmatrix} P_x \\ P_y \\ P_z \end{pmatrix} = l_p \begin{pmatrix} c_2 s_1 \\ c_2 c_1 \\ s_2 \end{pmatrix} \tag{1}$$

Differentiating P with respect to θ, denoted by $D_\theta P$, we can find the Jacobian [9]. Notice that because θ_3 does not enter in (1) we obtain a two-column Jacobian:

$$J_1 = D_\theta P = l_p \begin{bmatrix} c_2 c_1 & -s_2 s_1 \\ -c_2 s_1 & -s_2 c_1 \\ 0 & c_2 \end{bmatrix} \tag{2}$$

Using (2), find the motion of P in terms of the motion of joints θ_1 and θ_2:

$$\begin{pmatrix} dP_x \\ dP_y \\ dP_z \end{pmatrix} = J_1 \begin{pmatrix} d\theta_1 \\ d\theta_2 \end{pmatrix} \tag{3}$$

We wish to move point P of the arm along the line CD, Figure 5, perpendicular to PM and l_1. This corresponds to these two vectors being parallel to each other:

$$\begin{pmatrix} dP_x \\ dP_y \\ dP_z \end{pmatrix} \text{ parallel to } \overrightarrow{CD} \tag{4}$$

The directional vector of CD is:

$$\overrightarrow{CD} = \begin{pmatrix} -\left(s_\beta c_1 + s_2 c_\beta s_1\right) \\ s_\beta s_1 - s_2 c_\beta c_1 \\ c_2 c_\beta \end{pmatrix} \tag{5}$$

By substituting (5) into (3) and solving for $d\theta_1$, $d\theta_2$, $d\theta_3$, find the derivative of the joint vector which defines the motion needed to move point P along CD . Thus the direction of motion is defined by the vector:

$$\begin{pmatrix} d\theta_1 \\ d\theta_2 \\ d\theta_3 \end{pmatrix} = w \begin{pmatrix} -s_\beta \\ c_\beta c_2 \\ \lambda \end{pmatrix} \tag{6}$$

where λ and w are constants

The quantity λ in (6) can be arbitrary because in the case at hand link l_2 is not obstructed, and so it is free to move. Therefore, θ_3 is free to swing in any direction, and $d\theta_3$ can have any value. Instead of the angle β, the sensor system will actually provide another angle, α, which describes the line PM, the sensing axis, and relates to β as

$$\alpha = \beta - \pi/2 \tag{7}$$

Substituting (7) into (6), obtain:

$$\begin{pmatrix} d\theta_1 \\ d\theta_2 \\ d\theta_3 \end{pmatrix} = \begin{pmatrix} c_\alpha \\ s_\alpha c_2 \\ \lambda \end{pmatrix} \tag{8}$$

Finally, the normal to the collection of vectors described by (8) is:

$$\vec{n} = \begin{pmatrix} s_\alpha c_2 \\ -c_\alpha \\ 0 \end{pmatrix} \tag{9}$$

Thus, (9) is the local normal of the tangent plane at the point of contact with a Type I obstacle.

Type II are those obstacles that obstruct link l_2, as shown in Figure 6. Line segment PB is perpendicular to l_2 and lies in the plane that contains the z-axis and link l_2; points P and M are as defined above; line PM is perpendicular l_2; line CD is perpendicular to the line PM and link l_2; α is the angle between PB and PM; and β is the angle between PB and CD. The arm can be moved safely (in work space) by moving the point P in the plane that contains l_2 and the line CD. Using forward kinematics, we can find the coordinates of point P:

$$P = \begin{pmatrix} \left(l_1 c_2 + l_p c_{(2-3)}\right) s_1 \\ \left(l_1 c_2 + l_p c_{(2-3)}\right) c_1 \\ \left(l_1 s_2 + l_p s_{(2-3)}\right) \end{pmatrix} \qquad (10)$$

Differentiating P with respect to θ, obtain the Jacobian of the arm, where

$$s_{(2-3)} = \sin(\theta_2 - \theta_3),\ c_{(2-3)} = \cos(\theta_2 - \theta_3)$$

$J_2 = D_\theta P =$

$$\begin{bmatrix} c_1 \left(l_1 c_2 + l_p c_{(2-3)}\right) & -s_1 \left(l_1 c_2 + l_p s_{(2-3)}\right) & s_1 l_p s_{(2-3)} \\ -s_1 \left(l_1 c_2 + l_p c_{(2-3)}\right) & -c_1 \left(l_1 c_2 + l_p c_{(2-3)}\right) & c_1 l_p s_{(2-3)} \\ 0 & \left(l_1 c_2 + l_p c_{(2-3)}\right) & -\ l_p c_{(2-3)} \end{bmatrix} \qquad (11)$$

The line segment PM is the normal of the tangent plane at the contact point P in workspace:

$$\overrightarrow{PM} = \begin{pmatrix} -\left(c_1 s_\alpha + s_1 s_{(2-3)} c_\alpha\right) \\ s_1 s_\alpha - c_1 s_{(2-3)} c_\alpha \\ c_{(2-3)} c_\alpha \end{pmatrix} \qquad (12)$$

Obtain two non-colinear vectors that are perpendicular to line PM. A convenient pair of vectors are, for example,

$$\overrightarrow{t_1} = \begin{pmatrix} 0 \\ -\ c_{(2-3)} c_\alpha \\ s_1 s_\alpha - c_1 s_{(2-3)} c_\alpha \end{pmatrix} \qquad (13)$$

$$\overrightarrow{t_2} = \begin{pmatrix} c_{(2-3)} c_\alpha \\ 0 \\ c_1 s_\alpha - s_1 s_{(2-3)} c_\alpha \end{pmatrix}$$

Find the direction in configuration space that will cause P to move in the direction of M:

$$J_2^{-1} \overrightarrow{PM} = \begin{pmatrix} s_\alpha l_p \\ 0 \\ c_\alpha \left(l_p c_{(2-3)} + l_1 c_2\right) \end{pmatrix} \qquad (14)$$

Point P can move anywhere in the plane spanned by t_1 and t_2. We can now use J_2 to find the directions of motion in configuration space that would keep the arm in this plane. The local normal to the tangent plane at the point P in configuration space is vector $\overrightarrow{n}$ which is normal to $\overrightarrow{p_1} = J_2^{-1} \overrightarrow{t_1}$ and to $\overrightarrow{p_2} = J_2^{-1} \overrightarrow{t_2}$. This vector is given as:

$$\overrightarrow{n} = \begin{pmatrix} s_\alpha \left(\frac{l_1}{l_p} c_2 + c_{(2-3)}\right) \\ -\ c_\alpha \left(\frac{l_1}{l_p} c_3 + 1\right) \\ c_\alpha \end{pmatrix} \qquad (15)$$

Thus, (15) presents the local normal of the contact plane in configuration space. One may ask why we couldn't have found (15) simply by transforming the vector in (12) to configuration space. After all, doesn't the normal to the tangent plane in workspace transform to the normal of the tangent plane in configuration space? The answer is no, and the reason for that is that the mapping from workspace to configuration space is nonconformal. In other words, angles between lines and planes are not preserved by the mapping.

However, the plane in the configuration space spanned by the vectors p_1 and p_2 is indeed the tangent plane in that space. Thus, the normal vector in the configuration space can be found directly, using vectors p_1 and p_2 - which produces vector n in (15). Any motion in the configuration space perpendicular to n is therefore safe.

Type III are those obstacles that obstruct the arm wrist or elbow, see Figure 7. Here, line segment PB is perpendicular to link l_2 and lies in the plane that contains the z-axis and link l_1; the plane containing points B, P, and R is perpendicular to link l_2. Notation: PR is the projection of PM onto the plane containing B, P, and R; the angle between BP and PR is α, and the angle between the plane BPR and PM is ρ. Finding the local normal here is similar to the case of a Type II obstacle, except now $l_p = l_2$ is substituted in the expressions for forward kinematics and Jacobian calculation. First, find the directional vector of PM:

$$\overrightarrow{PM} = \begin{pmatrix} -c_1\ s_\alpha c_\rho + s_1\left(c_{(2-3)} s_\rho - s_{(2-3)}\ c_\alpha c_\rho\right) \\ s_1\ s_\alpha c_\rho + c_1\left(c_{(2-3)} s_\rho - s_{(2-3)}\ c_\alpha c_\rho\right) \\ s_{(2-3)} s_\rho + c_{(2-3)}\ c_\alpha c_\rho \end{pmatrix} \qquad (16)$$

Find two non-colinear vectors, $\vec{t_1}$ and $\vec{t_2}$, each perpendicular to PM:

$$\overrightarrow{t_1} = \begin{pmatrix} 0 \\ -s_{(2-3)} s_\rho - c_\alpha c_{(2-3)} c_\rho \\ s_1 s_\alpha c_\rho + c_1\left(c_{(2-3)} s_\rho - c_\alpha s_{(2-3)} c_\rho\right) \end{pmatrix}$$

$$\overrightarrow{t_2} = \begin{pmatrix} -s_{(2-3)} s_\rho - c_\alpha c_{(2-3)} c_\rho \\ 0 \\ -c_1 s_\alpha c_\rho + s_1\left(c_{(2-3)} s_\rho - c_\alpha s_{(2-3)} c_\rho\right) \end{pmatrix} \qquad (17)$$

Finally, for a Type III obstacle the local normal to the tangent plane in the configuration space is:

$$\vec{n} = \begin{pmatrix} s_\alpha\left(\frac{l_1}{l_2} c_2 + c_{(2-3)}\right) \\ -c_\alpha\left(-\frac{l_1 s_3 s_\rho}{l_2 c_\alpha c_\rho} + \frac{l_1}{l_2} c_3 + 1\right) \\ c_\alpha \end{pmatrix} \qquad (18)$$

Handling Special Cases

Interestingly, the problem of singularities that so often plagues various robot motion control techniques, does not present a difficulty here. This is because, although the Jacobian is being used in the derivation of the normal vectors, its determinant can be factored out because only the direction of the normal vector, not its norm, is being used.

Another type of special case takes place when all the components of the normal vector n turns to zero. To consider this case, it is sufficient to assume that only one sensor is sensing an obstacle. First, we find the local normal of the obstacle in the special case of $\alpha = \pi / 2$. When this value of α is substituted into any of the expressions (9), (15), or (18), the resulting local normal has the form :

$$\vec{n} = \begin{pmatrix} u \\ 0 \\ 0 \end{pmatrix} \qquad (19)$$

where *u* is a constant; assume for the moment that $u \neq 0$. This local normal implies that safe motion of the arm relative to the obstacles in this situation must involve an arm incremental change of Θ_2 and Θ_3 only (see Figure 6). When $\alpha = \pm\ \pi / 2$ the plane that contains l_1 and l_2 is perpendicular to the line PM, and the arm can move in that plane without collision with the obstacle.

Another special situation develops, when the point P is on the z-axis: then, the following relationship between Θ_2 and Θ_3 holds:

$$c_2 = -\frac{l_p}{l_1}\ c_{(2-3)} \qquad (20)$$

In this case, the local normal is of the following form:

$$\vec{n} = \begin{pmatrix} 0 \\ u_1 \\ u_2 \end{pmatrix} \qquad (21)$$

where u_1 and u_2 are constants

The expressions for local tangent are always well defined except in the case when both of the above situations occur simultaneously, thus when :

$$c_2 = -\frac{l_p}{l_1}\, c_{(2-3)} \quad \text{and } \alpha = \pm \pi / 2$$

in which case the local normal becomes :

$$\vec{n} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \qquad (22)$$

Obtaining the null vector suggests that any incremental motion of the arm can be considered collision free. As already described above, an incremental change of Θ_2 and Θ_3 can be considered safe because the plane that contains l_1 and l_2 is perpendicular to PM. As for Θ_1, because point P lies on the z-axis, its motion is not affected by any incremental change of Θ_1. Thus, an incremental change of Θ_1 can be considered safe. By repeating the sensing and step planning phase after the incremental move, proper collision free motion can be achieved.

Conclusion

The sensor-based motion planning system presented here senses obstacles by emitting amplitude modulated infrared light and then sensing the reflected signal. Five hundred of such sensors have been incorporated in the sensitive skin that covers the surface of our arm manipulator. The problem of interpreting data from this many sensors is handled by the step planning algorithm, which transforms the sensor data into local tangent normals in the configuration space. Once this information is available, a global path planning algorithm can then be used to move the arm through free space, or to follow the contour of obstacles that are encountered.

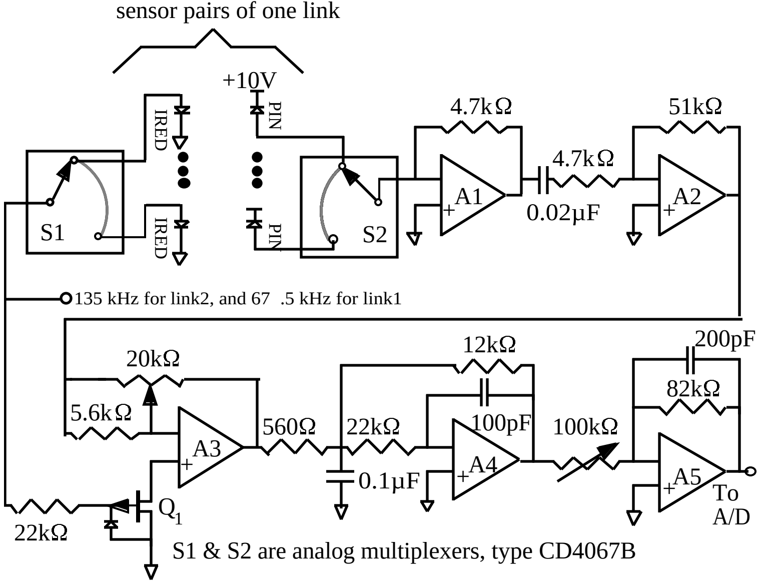
The sensor system, and the sensor interpretation algorithms are the main contributions of this paper. Together with a global path planning algorithm not presented here, they present the major components of a system that allows a robot arm to operate in a complex environment with unknown obstacles. Assembly of such a system is presently underway in the Robotics Laboratory of Yale University.

Acknowledgment

The authors would like to thank Thom Warmerdam from Philips Laboratories in Briarcliff Manor, NY, who suggested the idea of the switching synchronous detector. They also wish to thank the reviewers for all their suggestions and comments.

References

1. A.Elfes, Sonar Based Real-World Mapping and Navigation, IEEE Journal of Robotics and Automation, June 1987.
2. A. Fenn, R.Brooks, MIT Mobile Robots - What's next, Proc. 1988 IEEE Conference on Robotics and Automation, Philadelphia, PA, April 1988.
3. D. Kriegman, E.Triendle, T.Binford, A Mobile Robot: Sensing, Planning, and Locomotion, Proc. 1987 IEEE Conference on Robotics and Automation, Raleigh, NC, March 1987.
4. E. Cheung and V. Lumelsky, Motion Planning for Robot Arm Manipulators with Proximity Sensors, Proc. 1988 IEEE Conference on Robotics and Automation, Philadelphia, PA, April 1988.
5. V. Lumelsky, Effect of Kinematics on Motion Planning for Planar Robot Arms Moving Amidst Unknown Obstacles, IEEE Journal of Robotics and Automation, June 1987.
6. GE Solid State CMOS Data Book, GE Corporation, April 1988
7. Linear Data Book, National Semiconductor Corporation, 1982.
8. H. Hemami, Differential Surface Model for Tactile Perception of Shape and On-Line Tracking of Features, IEEE Journal of Systems, Man, and Cybernetics, March/April 1988.
9. R.P. Paul, Robot Manipulators: Mathematics, Programming, and Control, MIT Press, 1983.



S1 & S2 are analog multiplexers, type CD4067B

A1 through A5 are JFET operational amplifier, type LF347

Figure 2. Schematic of the detector circuit for one link