The Use of 3-D Sensing Techniques for On-Line Collision-Free Path Planning

V. Hayward, S. Aubry, and Z. Jasiukajc
McGill University
Montreal, Quebec, Canada H3A2A7

Abstract

This paper discusses the state of the art in collision prevention for manipulators with revolute joints, showing that it is a particularly computationally hard problem. Based on the analogy with other hard or undecidable problems such as theorem proving, we propose an extensible multi-resolution architecture for path planning, based on a collection of weak methods. Finally, we examine the role that sensors can play for an on-line use of sensor data.

1. Introduction

Automated collision prevention for robot manipulators is an essential feature seldom available, even in its simplest forms, in today's robotic systems. The term "collision prevention" will refer to collision detection and collision-free path planning. This paper presents current issues in collision prevention for manipulator robots with revolute joints in relation to the use of sensors.

Given the difficulty of the problem, only partial solutions have been found. Furthermore, we believe that it is particularly constraining to have to assume that robotic systems will operate in perfectly pre-modeled environments, as do all the currently existing industrial systems. As a consequence of the "perfect model" approach, many largely unsolved issues immediately arise: uncertainty representation and assessment, time-varying environments, computational and storage complexity.

We will rather advocate an "imperfect model" approach that will lead us to the consideration of a multi-level system heavily relying on sensory information and using multi-resolution algorithms. Instead of elaborating a theoretically exact method, and then deriving the computational and storage complexity in order to design a computer system to implement it, we will rather present methods that only partially solve the problem, but whose performance improve as the computing power is increased. Studying the problem of collision path-finding in connection with the available sensing techniques also provides some insight into the solution.

2. Current Methodologies for path planning

Experimental research in path-planning in the context of mobile robots has been fo far more successful because of the reduced complexity of the two-dimensional case. However, many concepts developed in the two-dimensional case do not extend readily to the three-dimensional case. For example, the conceptually attractive "configuration space approach" fails to extend easily to the case of manipulators with revolute joints, whereas it applies very nicely to the case of mobile robots. The reason is that the Cartesian space, in which obstacles—or free-space—are described, maps very awkwardly into the configuration space. In case of a manipulator, the configuration space is equivalent to the joint space. The mapping is highly non-linear and occurs between spaces of different dimensionalities. The computational complexity becomes unmanageable beyond three joints for the problem of mapping the Cartesian space scene into joint space as well as for searching the resulting graph. The approach is limited to three joints (Lozano 1986, Gouzène 1984) even if recursive decomposition schemes are utilized (Faverjon 1985).

Another widely adopted approach (Khatib 1986) is of local nature and consists of the computation of a artificial field of potential increasing near obstacles, and globally decreasing toward a goal position. Pseudo-forces are then included in the low-level motion servo computations of the manipulator. As a result, the manipulator is controlled to move away from obstacles and toward the goal position. Unfortunately, the method breaks down in obstacle configurations that create local minima. Computational complexity also precludes attempts to enlarge the scope of this method. The great attraction of this idea is the possibility to use sensor data directly for the computation of the potential instead of an a priori model. Similar schemes can be formulated in kinematic terms, that is in terms of velocities, instead of forces. The problem of local minima can be largely eliminated by the use of redundant manipulators: the trajectory of the end-effector can be completely specified and the redundant linkages used to avoid obstacles using only local information (Maciejewski 1985). Operation Research has been also considered helpful to attack the complexity problem (Grechanovsky 1983).

The methods of the last category resort to limit the class of tasks being planed. For example, the range of motions can be restricted to pick and place operations (Brooks 1983b). Similarly, the range of obstacles can be restricted, for example to pillars shapes (Luh 1984).

3. Proposed Ideas

We would like to suggest a few new ideas to the problem of collision prevention, in the view of their use in an on-line control system. We mean by that, trajectory generation techniques that allow the computation of collision-free trajectories in the same amount of time as require by the motion of general purpose robots, using limited computational resources. At this point, we would like to draw an analogy with the problem of theorem proving in computational logic. This problem is undecidable, that is to say, if the proposition submitted to the system is in fact false, the result cannot be obtained in a finite number of steps.

The search space to explore in order to prove a proposition can become arbitrarily large. If the proposition under study in indeed provabie, efficient methods in theorem proving attempt to use powerful heuristics to reduce the search space. These heuristic methods are called "weak methods" because they do not guarantee success, but are likely to converge in most interesting cases. If the proposition is false, this conclusion may not be reachable in a finite number of steps, and one has to resort to cut the search at some arbitrary point and to assume that the proposition probably is false.

In path planning, there are many heuristics available, and we suppose a limited amount of available computations, hence the architecture described below, based on a collection of weak methods.

3.1. Computational Architecture

We require the system to be extensible in the sense defined by Brooks (Brooks 1986). As researchers are devising new methods to calculate collision free trajectories, we would like to be able to integrate these advances while causing a minimum of disturbances to existing and working parts of the system.

The following diagram illustrates the design concept of an extensible architecture. The question of whether each of the methods will reside on one or several processors is of little importance. What is is important is to design them as peers such as they can accept the same input and produce the same output formats. The crucial point is not to attempt to parallelize the computations of one particular method, because we know that many of them require exponential times to execute, but to parallelize the methods between them so that we obtain a natural selection of the most appropriate for the situation at hand.

The computations should be done at a least two levels. The top-level searches for collision-free trajectories, using any of the methods available. Input to the high-level is data obtained from global sensor measurements as discussed in the section "Sensors," or from pre-determined models obtained from data-bases. The lowest level is a local collision detector that also uses either sensor information or pre-determined model information. We will require an efficient collision detector to certify the proposed trajectories, or use on-line proximity sensor mounted on the arm to locally modify the trajectories as they are executed. In the later case, we take the chance that the motion may never terminate.

3.2 A Variety of Heuristics: Archetypical Motions

The study of robot motions shows that given an approximate description of a robot's environment, and given the initial and final configurations, robot motions can be classified into classes of archetypical motions. A preliminary analysis shows that collision-free motions bear a strong relationship with the structure of the workspace. This relationships can be exploited to built a system that infers plausible motions. In this framework, the problems under study are:

- Classification of obstacles according to the influence that their shape might have on the motions: small, large (with respect to the robot), compact, elongated, flat, etc... Interesting simple cases are: spheres, infinite cylinders, half spaces, and holes.

- Classification of the relations between these obstacles with respect to the robot elements: proximity, position with respect to the elbow, under, above, on the left, on the right, etc... Inference will occur on criteria of this kind.

- Classification of typical motions: retraction, extension, sweep, wrist re-orientation. The consequences of these motions are explicit: if joint No 1 turns in the positive direction and the arm is stretched, then end-effector sweeps on the left; if the arm retracts and is in the elbow up configuration, then the end-effector moves inward, and the elbow moves upward, etc...

Once the scene and the robot attitude are encoded in terms of facts and rules, motions can be generated by automatic inference.

GLOBAL SENSORS or CAD DATABASE

v

MULTI-RESOLUTION SCENE DESCRIPTION

v

METHOD-1 METHOD-2 METHOD-3 ... METHOD-n

v

CERTIFICATION and SELECTION, or LOCAL ON-LINE MODIFICATION

3.3 Joint Decoupling

Joint decoupling is another way to attack the problem. The observation of certain collision avoiding motions reveals that motion planning can be performed by planning the motions of the joints independently. During such a motion, in the *reference frame* attached to each link of a robot, all the obstacles appear as moving obstacles. The task consists for each link to plan a one-dimensional trajectory in its own coordinate frame with a time-varying environment. We know from (Kant 1984 and 1986) that such a planning is possible, by planning the velocity along a predetermined path. This algorithm finds solutions in a large number of cases, when the priority among the set of joints is adequately determined. The problem is formaly equivalent to moving multiple objects as in (Erdman 1986).

3.4 Piece-Wise Trajectory Decomposition

Another heuristic method can be described as follows. If the arm is to move from point A to point B, a trajectory is generated at the first iteration using a very simple scheme: a linear joint interpolated motion between A and B, for example. The trajectory is then verified. In case of collision, an intermediate knot point is generated by the closest non interfering position. The initial segment is then split and the process recursively iterated on the sub-segments.

3.5 The generate-test-refine architecture

We have just listed three powerful heuristics to reduce the search space of the problem. There exist others. We can augment the power of these heuristics by feeding back to a motion planner information provided by the collision detector in case of the failure of a plan, or information provided by a merit estimator, in case of success. The system is left interating during the allocated time period, the last best solution being retained.

3.5 Good Collision Detectors

Of what precedes, we require a good quality collision detector, that is to say, one that does not require exponential nor polynomial times to perform and one that uses multi-resolution algorithms. This problem has been examined in (Hayward 1986). One approach is to perform the modeling the robot in terms of control points scattered on its boundary. Collision detection can be then performed by showing that all the control points are in free-space. (note that there is no need to worry about rotations). A multi-resolution system can then be easily obtained.

The quality of the result augments with the allocated running time and the CPU power. Methods for generating multi-resolution control points are indicated in (Bhan 1986). Octree encoding methods provide very naturally for multi-resolution algorithms, however, we have other schemes under consideration because octree make no use of the coherence that might be present in a scene and therefore can lead to great inefficiencies.

4. Sensors

"Model Building Sensing" is used to gather *global* three-dimensional information from the environment. In a robotic context, the sensors perform a "surveying" function, providing information to be used by the path planning module. This is quite different from the on-line uses of sensors in which the local environment is continuously sampled so as to avoid crashes. In particular, model building sensors must operate *over a wider range than their servoing counterparts.*

4.1 Global Sensors versus Proximity Sensors

The chosen sensor must be either a proximity sensor attached to a "roving" arm or it must be capable of acquiring three-dimensional information at a distance. In the first case, the accuracy is limited only by that of the manipulator. However, control problems are likely to crop up for complex environments where concavities abound. Such problems arise because the environment is not known a priori: in fact, the environment is difficult to explore precisely because it is not known! Consequently, such a process is likely to be a slow one.

We contend that such proximity methods are only advisable when the task environment is so intricate that spatial considerations prevent larger apparatus such as those we describe below to conveniently operate. Suppose for example that we wish to model the bottom of a narrow, oblong cavity inside a given object. We can safely assume that our robot arm can indeed penetrate the cavity and orient itself within it, since otherwise there would be little point in modeling it. Using the very manipulator which is to perform the robotic task is then the most direct way to model the task environment.

4.2 Acquiring 3-D Information

Techniques developed for acquiring three-dimensional information at a distance are still the preferred answer to automated model-building in most cases. These techniques can be either photometric or telemetric.

4.2.1 Photometric Techniques

Photometric techniques attempt to infer distance from photographic images. But such images map intensity, an extrinsic characteristic of the three-dimensional world, onto the the two-dimensional plane along the lines of a perspective projection. The task of recovering the correct interpretation for a given image is then a formidable one since it requires that the perspective ambiguity (lines map into points) be resolved from the intensity cue alone; formally, this task consists of inverting an illumination-reflectance operator I which maps the three-dimensional scene to the image plane. The so-called "shape-from" techniques attempt to perform that difficult inversion using a combination of analytical work (Horn 1968; Horn 1975; Ferrie 1986; Levine, O'Handley and Yagi 1973), and of higher-level cognitive processes (Rosenburg, Levine and Zucker 1978; Bajcsy and Lieberman 1976; Shirai 1973; Marr 1976). These methods have been much investigated in part for their similarity to human visual processing, but also because they do not require sophisticated optical hardware.

4.2.2 Telemetric Techniques

In contrast with photometric techniques, telemetric techniques usually require specialized hardware but are much easier to analyze in return and therefore constitute a much preferable means for automatic three-dimensional scene acquisition. The goal here is to build "range images": a range image maps the distance of the closest point in the scene to every node of an orthographic grid the size of that scene. These images are usually constructed by monitoring patterns of points (Hasegawa 1982; Ishii and Nagata 1976), lines (Oshima and Shirai 1979: Sato and Inokuchi 1985), or grids (Potmesil 1979) of light which are successively projected onto the scene and reflected to a sensor located at or near the light emitting device (often a laser). Either positional analysis of the returning rays or time-of-flight discrimination can now be used to infer the range of the closest obstacle. In the first case, simple geometrical relationships relating emitted and returned rays yield the sought distance in a process called *triangulation*. In the second case, the time taken by light rays to travel from and back to the emitting laser source allows us to calculate that same distance. Needless to say, the practicability of the latter method is limited by the very sophisticated electronics that the enormous speed at which light travels requires (Lewis and Johnston 1977; Nitzan, Brain and Duda 1977.)

An alternative time-of-flight method uses sound waves instead of light rays because of their more manageable speed. Although simple in principle, the method suffers from various engineering problems such as the need for frequent recalibration. the difficulty experienced in focusing sound waves, as well as their hard-to-model reflective properties.

In summary, the "safest" and most accurate methods of acquiring distance information seems at present to be triangulation. However, one should not discount ultrasonic time-of-flight methods which are already commercially available. Further, many researchers believe that laser time-of-flight methods will soon present itself as the most viable method since it offers in theory the greatest absolute accuracy. The interested reader should refer to the excellent review by Jarvis (Jarvis 1983) for further reading on range acquisition techniques.

5. Conclusion

In this paper, we have presented an overview of methods related to the collision prevention for manipulators with revolute joints. It has been shown that it is a difficult problem in its generality and we have proposed a computational architecture based on an analogy with an another domain of Artificial Intelligence.

6. Acknowledgements

The ideas in this paper were initially formulated when the first author was developing a Cartesian-based collision detector while at CNRS (France). and by the second author while studying 3D sensing techniques at McGill University. Further contributions are due to conversations with Kamal Kant at McGill University, and to an inspiring lecture delivered by R. A. Brooks.

7. References

Ahuja, N. et al. 1980. Interference detection and collision avoidance among three dimensional objects. *First Ann. Nat. Conf. on Artificial Intelligence*, Stanford, CA.

Ahuja, N., Nash, C. 1984 (May). Octree representation of moving objects. *Computer Vision, Graphics, and Image Processing*, Vol 26, No 2.

Bajcsy, R., Lieberman, L. 1976. Texture gradient as a depth cue. *Computer Graphics & Image Processing*, Vol. 5, 1976, pp. 52-67.

Bham, B, Chih-Cheng, H. 1986 (October). Computer aided geometric design based 3-D models for machine vision. *Proceeding of the Eighth International Conference on Pattern Recognition, IAPR*, Paris, France, pp. 107-110.

Boyse, J. W. 1979. Interference detection among solids and surfaces. *Communications of the ACM*, Vol 22, No 1.

Brooks, R. A. 1983*a* (March). Solving the find-path problem by good representation of free space. *IEEE Trans. on Systems, Man, and Cybernetics*, Vol 13, No 3.

Brooks, R. A. 1983*b* (Winter). Planning collision-free motions for pick-and-place operations. *International Journal of Robotics Research*, Vol 2, No 4.

Brooks, R. A. 1986 (March). A robust Layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, Vol. RA-2, No. 1, pp. 14-23.

Cameron, S. 1985(March). A study of the clash detection problem in robotics. *IEEE Second Conference on Robotics and Automation*, St. Louis, Missouri. pp. 488-496.

Canny, J. 1984. Collision detection for moving polyhedra. AI Memo No 806, MIT, October 1984.

Erdman, M., Lozano-Pérez, T. 1986 (May). On Multiple Moving Objects. *Massachusetts Institute of Technology Artificial Intelligence Laboratory*, A. I. Memo 883.

Faverjon, B. 1984 (June). Obstacle Avoidance Using an Octree in the Configuration Space of a Manipulator. *IEEE First Conference on Robotics*, Atlanta.

Ferrie. F.P. 1986 (March) Reconstructing and interpreting the 3D shape of moving objects. *Ph.D Thesis, Department of Electrical Engineering, McGill University*, Montréal.

Gouzène, L. 1984 (Winter). Strategies for solving collision-free trajectories problems for mobile and manipulator robots. *International Journal of Robotics Research*, Vol 3, No 4.

Grechanovsky, E., Pinsker, I. Sh. 1983. An algorithm for moving a computer-controlled manipulator while avoiding obstacles. *Proc. Eighth IJCAI*. 807-813. Karlsruhe, W. Germany.

Hasegawa, T. 1982 (May). An interactive system for modeling and monitoring a manipulation environment. *IEEE Transactions on Systems, Man and Cybernetics*, SMC-12, pp. 250-258.

Hayward, V., Paul R. P. 1984 (June). Introduction to RCCL: a robot control "C" library. *IEEE First International Conference on Robotics*, Atlanta.

Hayward, V. 1986 (April). Fast collision detection scheme by recursive decomposition of a manipulator workspace. *IEEE Conference on Robotics and Automation*, San Francisco, CA, pp. 1044-1049.

Horn B.K.P. 1968 (May) Focusing. *M.I.T. Project MAC*, AI Memo 160.

Horn B.K.P. 1975 Obtaining shape from shading information, in *The psychology of computer vision* P.H. Winston (Editor), McGraw-Hill.

Ishii, M. and Nagata, N. 1976. Feature extraction of 3-D objects and visual processing in a hand-eye system using laser tracker, *Pattern Recognition*, Vol. 8 (1976), pp. 229-237.

Jarvis, R. A. 1983 (Mars). A perspective on range finding techniques for computer vision, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-5(2), pp. 122-139.

Kant, K., Zucker, S. W. 1984 (November). Trajectory planning in time-varying environments, I: TPP=PPP+VPP. Technical Report, TR-84-7R, Computer Vision and Robotics Laboratory, Department of Electrical Engineering, McGill University.

Kant, K., Zucker, S. W. 1986. Toward efficient trajectory planning: the path-velocity decomposition. *International Journal of Robotics Research*, Vol 5, No 4, to appear.

Khatib, O. 1986 (Spring). Real-time obstacle avoidance for manipulators and mobile robots. *The International Journal of Robotics Research*, Vol 5, No 1.

Knuth, D. E. 1973. *The Art of Computer Programming, Volume 1 / Fundamental Algorithms*, Addison-Wesley.

Levine, M. D., O'Handley, D. A., Yagi, G. M. 1973. Computer determination of depth maps, *Computer Graphics & Image Processing*, Vol. 2, pp. 134-150.

Lewis, R.A. and Johnston, A.R. 1977. A scanning laser range finder for a robotic vehicle, *Proceedings of the fifth International Joint Conference on Artificial Intelligence*, pp. 762-768.

Lozano-Pérez, T. 1986 (June). A simple motion planning algorithm for general robot manipulators. *Massachusetts Institute of Technology Artificial Intelligence Laboratory*, A. I. Memo 896.

Lozano-Pèrez, T. 1984 (November). An approach to automatic robot programming. USA-France Robotics Workshop, University of Pennsylvania.

Lozano-Pèrez, T. 1981. Automatic planning of manipulator transfer movements. *IEEE Transactions on Systems, Man and Cybernetics*, Vol SMC11, pp. 681-698.

Lozano-Pèrez, T. 1983 (February). Spatial-Planning: a configuration space approach. *IEEE Transactions on Computers*, Vol C32, No 2.

Luh, J. Y. S. 1984 (Spring). A scheme for collision avoidance with minimum distance traveling for industrial robots. *Journal of Robotic Systems*, Vol 1, No 1.

Maciejewski, A. A., Klein, K. A. 1985 (Fall). Obstacle avoidance for kinematically redundant manipulators. *The International Journal of Robotics Research*, Vol 4, No 3.

Marr D., Poggio, T. 1976 (June). Cooperative computation of stereo disparity, M.I.T., A.I. Lab, Memo No. 364.

Nitzan, D., Brain, A., Duda, A. 1977 (February). The measurement and use of registered reflectance and range data in scene analysis, *Proceedings of the IEEE*, pp. 206-221.

Oshima, M. and Shirai, Y. 1979. A scene description method using three- dimensional information, *Pattern Recognition*, Vol. 11, pp. 9-17.

Potmesil, M. 1979. Generation of 3D surface descriptions from images of pattern illuminated objects, *IEEE Conference on Pattern Recognition and Image Processing*, pp. 553-559.

Rosenburg, D, Levine, M.D. and Zucker, S.W. 1978 (November.) Computing relative depth relationships from occlusion cues, *Proceedings of the Fourth International Joint Conference on Pattern Recognition*, Kyoto, Japan pp. 765-769.

Sato, K., Inokuchi, S. 1985. Three-dimensional surface measurement by space encoding range imaging, *Journal of robotic systems*, 2(1), pp. 27-39.

Udupa, S. M. 1977 (August). Collision detection and collision avoidance in computer controlled manipulators. *Proceedings of the Fifth IJCAI*. MIT, Cambridge Massachusetts, August 1977.

Wong, E. K., Fu, K. S. 1986 (March). A hierarchical orthogonal space approach to three-dimensional path planning. *IEEE Journal of Robotics and Automation*, Vol RA-2, No 1.