Preprints of the Third International
Symposium on Experimental Robotics
Kyoto, Japan, Oct. 28-30, 1993

A robotic cell for deburring of polygonal objects

J-P Merlet, N. Mouly, J-J. Borrelly, P. Itey

INRIA Sophia-Antipolis, BP 93, 06902 Sophia-Antipolis Cedex,France

Abstract

A robotic cell for deburring planar polygonal objects is described. The object comes from a conveyor and arrives in a random position on a parallel manipulator. Its center of mass is located through the measurements of a 6-componants force sensor, it is then grasped and a force-feedback scheme is used to fix the object on a special plate. Then a probing algorithm is used to discover the location of the vertices of the object with a minimal number of measurements. The coordinates of the vertices are then used to build an ideal reference model of the object which is fed to a force-feedback scheme which perform the deburring of the object.

1. Introduction

Surface following with force-feedback is an important robotics task useful for many applications: grinding, polishing, deburring. In this kind of tasks the tip of the grinding tool has to apply a constant force on an object and follow its contour with a velocity as close as possible from a given constant value. In most cases surface following is only a 2D problem as the tip of the grinding tool may be reduced to a point moving in a known plane.

Therefore many researchers have addressed this problem [1],[2]. Most of these works emphasize the problem of stability of the force-feedback scheme due to the high gain in the loop. Clearly stability is an important issue as soon as there is contact between the robot and the object. It has been shown that stability is deeply dependent on the sampling rate of the system (which must be the highest possible) and on the mechanical stiffness of the coupling of the robot and the surrounding.

The sampling rate is in general fixed for a given hardware. Interesting results have been obtained by modifying the stiffness of the robot, for example by using a micro-macro manipulators as described in [3],[4],[5],[6] in which case a parallel manipulator is used as a wrist.

But for the special case of the surface following problem stability may also be improved if some approximate model of the shape of the contour is known. This can be done by a learning algorithm [7]: during a first experiment no model of the object is known but a force-feedback scheme enables to follow the contour. Forces and position of the end-effector are recorded during the experiment. Then the position of the end-effector for which the force was in a given range are selected to construct a reference model of the contour (for example by computing splines whose control points are the selected positions). Using this learning algorithm experiments can be performed with a reference model and this model may even be refined by using the data of further experiments. The drawback of such an approach is that the learning algorithm is rather slow and the use of splines to build the reference model may be not appropriate in some case, for example for polygonal objects. In that case the reference model may be defined by the list of the coordinates of the vertices of the object.

2. Probing algorithm

2.1. Principle

The problem of determining the locations of the vertices of a polygon using a sensing device which gives local information on the shape of the contour is known under the name of *probing* [8]. Researchers involved in computational geometry have addressed this problem especially to design a sensing strategy enabling to find all the vertices of the object with a minimum number of measurements.

We have decided to use the probing algorithm described by Boissonnat [9]. In this algorithm the measurement starts from a point and is done in a given direction called the *ray* of the measurement. The sensing device is able to determine the coordinates of the intersection of the ray with the object together with the normal of the object at this intersection point and the full process is called a *probe.* The following assumptions are made: the object has no collinear edges, the coordinates of a point U belonging to the interior of the object are known and the operator is able to define a circle C_e centered in U which does not intersect any part of the object.

The sensing strategy is now described. For the first probe a random point M_1 on C_e is chosen and the ray is the line M_1U, which insure that the probe will determine a point on the object. Together with the normal this point enables to determine the support line of an edge E_1 of the object. The second probe has as origin M_2, a point on C_e taken as the opposite point of M_1 with respect to U. This second probe enables to determine the support line of a second edge E_2 of the object. Let P_3 be the intersection point of the support lines of E_1, E_2. The third probe starts from a point M_3 intersection of the circle and the line P_3U and its ray is directed along P_3U.

For this third probe two cases may occur (figure 1):

- E_1, E_2 are adjacent and therefore P_3 is a vertice of the object. The probe will detect that the contact point is the intersection of the support line and that the left and right normals are the normal to E_1 and E_2. From this fact the algorithm deduces that P_3 is a vertice of the object.
- E_1, E_2 are not adjacent: the contact point will be different from P_3 and the support line of a new edge E_3 of the object is discovered.

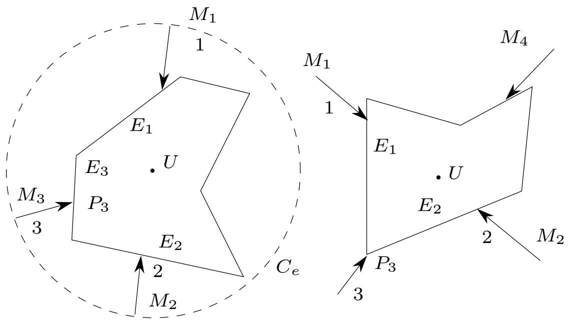

Figure 1. After completing the two first probes the third probe enables either to determine a vertice of the object (right) or a new edge (left).

In the first case the full shape of the object between E_1, P_3, E_2 has been discovered and the fourth probe will start from M_4 the opposite point to M_3 and the ray will be directed toward P_3. This insure that a new edge E_3 will be discovered. In both cases a list of pair of edges is build which contains the edges for which the intersection point of the support line has not been tested as a potential vertice: for example after the third probe this list is $\{ (E_1, E_3), (E_2, E_3) \}$. As soon as a pair is used to perform a probe it is removed from the list and the list is updated according to the result of the probe. When the list is empty all the vertices of the object have been determined. Clearly for a convex polygone only two probes are necessary to discover a vertice and therefore in that case the algorithm find all the vertices of a n-vertices polygon with $2n$ probes.

In some case for non-convex polygons a probe may not discover a vertice nor a new edge but it can be shown that in that case we can compute a probe which will discover either a new vertice or a new edge. Therefore for non-convex polygon it may happen that more than $2n$ probes are necessary but it may be shown that for a n-vertices polygon at most $3n-3$ probes will be necessary.

2.2. Probing with a force sensor

The measurement device used in our cell is based on a force sensor and the tip of the robot moves along the ray of the probe.

The force sensor is able to detect the contact between the tip of the robot and the object: therefore the coordinates of the contact point in the robot frame is known. In order to determine the normal to the object at the contact point the following procedure is used: as soon as a contact has been detected the robot performs a small backward motion along the ray, then a small motion along the perpendicular to the ray in the left direction and then a motion in a direction parallel to the ray toward the object. A new contact point is found and the robot backtrack until it lie again on the ray and a similar procedure is used now on the right side of the ray to find a new point on the edge. After this procedure the coordinates of three points on the edge are known which enable to determine the support line of the edge with a good accuracy. Although this procedure will not work with very small edges in practice no problem have been encountered. Figure 2 shows two examples of experimental probing of complex polygons.

3. The cell

3.1. Constitution

Our robotic cell (figure 5) is constituted of three manipulators:

- a parallel manipulator called the "left hand" which receive the object on its end-effector plate in a random position but with an approximatively fixed orientation. This manipulator has a traction-compression force sensor in each of its 6 legs and through the 6 measurement of these sensors the external forces and torques acting on the end-effector may be computed.
- an IBM SCARA robot which perform the probing and then the surface following of the object.

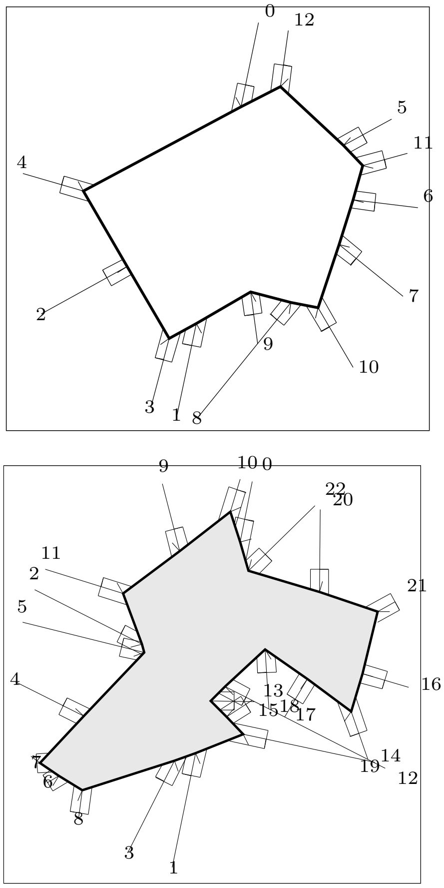

Figure 2. Two examples of application of the probing algorithm on complex polygons.

During these operations the object is fixed on a table lying in the vicinity of the robot and the force informations are given by a 6 componants force sensor mounted below the table.

- a 6 d.o.f. serial link manipulator AID which will pick the object on the end-effector of the left hand and transfer it on the table of the SCARA robot.

4. Picking of the object

4.1. Locating an object on the end-effector

Let $(O, \mathbf{x}, \mathbf{y}, \mathbf{z})$ be a frame $\mathcal{R}_l$ attached to the base of the left hand and $(C, \mathbf{x_r}, \mathbf{y_r}, \mathbf{z_r})$ be a frame $\mathcal{R}_l^r$ attached to the end-effector. By controlling the 6 links lengths the location of C and the orientation of the end-effector with respect to $\mathcal{R}_l$ can be modified at will. The posture of the end-effector of the left hand is assumed to be known and the forces and torques acting on the end-effector in the frame $\mathcal{R}_l$ can be computed.

For picking an object it is essential to determine the location of its center of mass (which may not be coincident with its geometrical center e.g. for non-homogeneous objects). Let M_x, M_y be the torques acting on the end-effector around the x, y axis, measured with respect to point C, F_z the force acting along the $\mathbf{z}$ axis and let x_m, y_m be the componants in $\mathcal{R}_l$ of the vector $\mathbf{CG}$ where G denotes the center of mass of the object. As the only force acting on the end-effector is the weight of the object we get:

$$M_x = -mgy_m \quad M_y = mgx_m \quad F_z = -mg \tag{1}$$

and therefore:

$$x_m = -\frac{M_y}{F_z} \quad y_m = \frac{M_x}{F_z} \tag{2}$$

As the location of C in $\mathcal{R}_l$ is known it is now easy to determine the coordinate of the center of mass in $\mathcal{R}_l$. Therefore the force and torques measurements enable to compute the picking point of the object in $\mathcal{R}_l$.

4.2. Calibration procedure

Let $(O_A, \mathbf{x}, \mathbf{y}, \mathbf{z})$ be a frame $\mathcal{R}_a$ attached to the base of the AID robot. As the picking operation will be performed by this robot and as we have determined the picking point in a frame $\mathcal{R}_l$ linked to the left hand we have to determine the location of O with respect to $\mathcal{R}_a$ i.e. the componants of $\mathbf{O_A O}$.

Once again this operation is done by using the force measurements. Assume that the tip of the end-effector of the AID robot touch the end-effector of the left hand as some point P. Using the force measurements we are able to determine the componants of $\mathbf{OP}$. Using the direct kinematics of the AID we are also able to compute the componants of $\mathbf{O_A P}$. As $\mathbf{O_A O} = \mathbf{O_A P} - \mathbf{OP}$ we are theoretically able to compute $\mathbf{OO_A}$. As the force measurements may be noisy the procedure is repeated for a dozen of points and a least-square algorithm is used to determine $\mathbf{O_A O}$.

4.3. Picking

As soon as an object is put on the effector of the left hand the force along the $\mathbf{z}$ axis indicates that an object is present and the location of the center of mass is computed. If the picking point is in the reachable workspace of the AID it moves over the picking point, the end-effector being directed toward the object. In the opposite case the left hand moves toward the AID base until the picking point lie in the workspace of the AID.

The gripper is then opened and the AID moves toward the picking point along the $\mathbf{z}$ axis. This motion is stopped when a contact is detected by the force sensor of the left hand (therefore the height of the object has no importance). The gripper is then closed and the robot moves along the $\mathbf{z}$ axis with the object in the gripper. The correct gripping of the object is verified as the force along the **z** axis must go back to zero. In case of failure the procedure is repeated.

5. Transfer of the object and fixation on the table

As soon as the object has been picked the robot AID moves toward the table near the SCARA and put the object on the it. At this point the left hand and AID are free to receive and transfer a new object. Now the SCARA has to pick the object on the table. But during the transfer of the part toward the table and its setting its orientation may have changed. As the object will be fixed on the table through an insertion process its orientation shall be approximatively constant. Therefore the object has to be picked and its orientation corrected. This process begins as soon as the presence of an object is detected by the force sensor below the table. Once again the force sensor of the SCARA table is used to determine the location of the center of mass of the part in the SCARA frame (a calibration procedure has been previously performed to found the position of the center of the force sensor frame in the SCARA frame). Then a simple "pushing" algorithm [10] is used to correct the orientation of the object. A force-feedback scheme which has been described in [11] is then used to insert the pins of the object in some holes of the table.

6. Surface following

As soon as the object is fixed on the table a probing is completed and the force-feedback scheme can be used. A desired velocity V_d and a desired force F_d are given together with a direction for the deburring (e.g. clockwise). As soon as there is contact between the grinding tool and the object the use of the reference model enables to determine on which edge of the object the contact point is located together with the external normal unit vector $\mathbf{N}$ and tangent unit vector $\mathbf{V_T}$ of this edge . Note that the tangent vector is chosen according to the direction indicated for the deburring.

A proportional controller is used to adjust the contact force by generating a velocity $\mathbf{V_N}$ along the normal $\mathbf{N}$ of the object which is computed as:

$$\mathbf{V_N} = k_f(F_m - F_d)\mathbf{N} \tag{3}$$

where k_f is a constant positive gain and F_m the measured force. The amplitude of this velocity is thresholded to V. Then a velocity $\mathbf{V_T}$ along the tangent of the contour $\mathbf{T}$ is computed as:

$$\mathbf{V_T} = \sqrt{V^2 - ||\mathbf{V_T}||^2}\ \ \mathbf{T} \tag{4}$$

Near the vertices of the object the velocity is reduced to minimize the value of the change of the force: indeed contact may be lost at sharp corner (but in that case the reference model indicates in which direction the robot has to move in order to find the object) or big increase in force may occur at acute corner as the sampling rate of the force measurement is finite.

At the start of the experiment the tip of the end-effector is put at some fixed distance from an edge of the object and then the robot moves toward the object in the direction of the closest edge as determined by the reference model. As soon as contact is detected by the force sensor the force-feedback scheme is used. The task is completed and will be stopped as soon as the tip of the end-effector has moved along an edge different from the starting edge and is now at a small distance from the start point.

Experimental results of a deburring process are shown in figure 3 (followed contour) and in figure 4 (magnitude of the contact force). In this experiment the velocity of the robot was 1 cm/s, the desired force was 2.5 N and the average contact force is about 2.506 N with a mean-square error of 0.355 (which is close from the mean-square of the sensor noise) i.e. very close from the desired value.

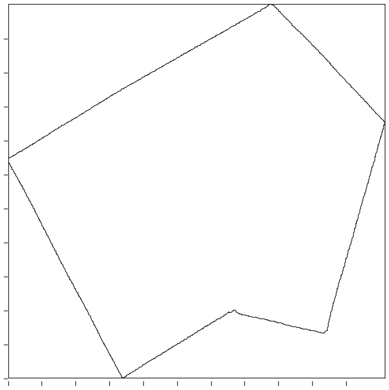

Figure 3. Result of the contour following task: followed contour.

It may be seen that at some point there was important changes in the force signal which occur mainly when the robot moves in the vicinity of the vertices of the polygon.

7. Hardware

The AID and the left hand are controlled through a special robot controller based on a VME bus using two 68030 CPU boards which perform the servoing of the robots. Force conditioning is done by a special interface connected to the bus. The sampling rate of the force sensor is about 72 Hz with a sensibility of approximatively 50 gr (the force sensor was initially designed to enable the measurement of mass of about 100 kg). Motions orders are read from a memory board and the force measurement are written on this memory board. This memory is shared through a special PTVME board by a SUN workstation which compute the desired motion according to the force measurement which are read from the shared memory and write them on the memory.

Another robot controller is used to control the SCARA robot and acquire the force measurement from the table sensor. This controller use three 68020 CPU boards: one is used for the servoing of the robot, a second one manage the real-time aspect of the controller and the third one is used to execute a program written in a C-like robot language. Specific primitives for this language may be written in C on a SUN workstation and downloaded through a RS232 link in the memory of the controller.

The SUN workstation may modify at running time the behavior of the program through the use of specific primitives of the robot language which enable to read and write data on the memory of the controller through the serial link. Due to the low speed of this link the possibility is mainly used to get data from the controller in order to monitor the execution of the task. The two SUN workstations work independently. The force sensor is an AICO with a sampling rate of about 200 Hz and a force sensibility of about 5gr.

The probing algorithm is managed by the SUN workstation devoted to the SCARA robot and the program running on the SCARA controller just receive motion orders from the workstation, stop this motion as soon as a contact is detected and send the current position of the robot to the workstation which compute the motion necessary to perform the next probe. This is a rather slow procedure as all the information goes through the serial link and the average time for a probe is 7s. The surface following task is much more faster as this task is fully under the control of the SCARA controller, the serial link being used only to send position and force data to the workstation for debugging and recording purposes.

8. Conclusion

This experiment was designed for presentation during the IEEE Robotics Conference in Nice in 1992. Many researchers have seen it working (there was only one failure during the 25 presentations which have been made). We encounter mainly two kinds of problem: one was really a hardware problem and was due to the multiplicity of wires involved in this experiment. The other one was the problem of the management of the various tasks, their synchronization and dealing with the various events which may occur during the execution of the experiment (e.g. failure of a sub-task). One possibility to deal with this problem is to describe the task and the possible events in a synchronous language like ESTEREL which will generate a finite state automata which can be verified and deadlock discovered. Such an approach has been used successfully in simulation to manage an insertion task using a conveyor, two manipulators and the left hand [12],[13].

References

[1] Mason M.T. Compliant motion. In *in Robot , Motion-Planning and Control, Brady & al. Ed.*, Cambridge, 1982. The MIT Press.

[2] Planck G. and Hirzinger G. Controlling a robot's motion speed by a force-torque sensor for deburring problem. In *4th IFAC/IFIP Symp. on Information Control problems in manufacturing technology*, October, 26-28, 1982.

[3] Reboulet C. and Robert A. Hybrid control of a manipulator with an active compliant wrist. In *3rd ISRR*, pages 76–80, Gouvieux, France, October, 7-11, 1985.

[4] Khatib O. Inertial characteristics and dextrous dynamic coordination of macro/micro manipulator systems. In *7th CISM-IFToMM Symposium on Theory and Practice of Robots and Manipulators*, Udine, Italie, September 1988.

[5] Merlet J-P. Force-feedback control of parallel manipulators. In *IEEE Int. Conf. on Robotics and Automation*, pages 1484–1489, Philadelphia, April, 24-29, 1988.

[6] Reboulet C. and Pigeyre R. Hybrid control of a 6 d.o.f. in parallel actuated micro-macro manipulator mounted on a Scara robot. In *ISRAM*, volume 3, pages 293–298, Burnaby, July, 18-20, 1990. ASME Press Series.

[7] Merlet J-P. Manipulateurs parallèles, 3eme partie : applications. Research Report 1003, INRIA, March 1989.

[8] Cole R. and Yap C. Shape from probing. *J. Algorithms*, 8:19–38, 1987.

[9] Alevizos P., Boissonnat J-D, and Yvinec M. On the order induced by a set of rays. application to the probing of non-convex polygons. Research Report 927, INRIA, November, , 1988.

[10] Akella S. and Mason M. Posing polygonal object in the Plane by pushing. In *IEEE Int. Conf. on Robotics and Automation*, pages 2255–2262, May, 12-14, 1992.

[11] Merlet J-P. Use of C-surface based force-feedback

algorithm for complex assembly tasks. In *ISER*, Toulouse, June, 25-27, 1991.

[12] Coste-Manière E. Utilisation d'Esterel dans un contexte asynchrone : une application robotique. Research Report 1139, INRIA, December, , 1989.

[13] Coste-Manière E. and Faverjon B. A programming and simulation tool for robotics workcells. In *Int. Conf. on Automation, Robotics and Computer Vision*, Singapore, September, , 1990.

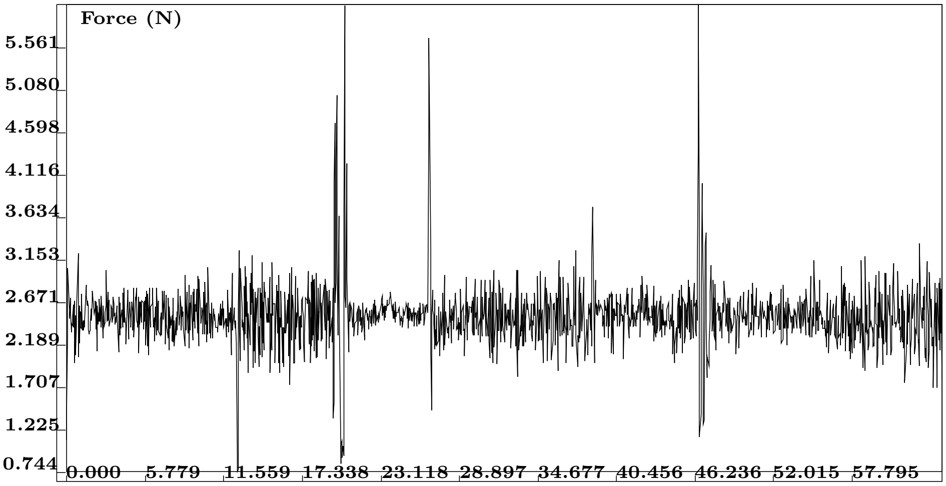

Figure 4. Result of the contour following task: measured force.

Figure 5. The robotic cell