KfK 4120
August 1986

Automatic Path-Planning for a Multilink Articulated Boom Within the Torus of a Fusion Reactor

D. Smidt
Institut für Reaktorentwicklung

Kernforschungszentrum Karlsruhe

KERNFORSCHUNGSZENTRUM KARLSRUHE
Institut für Reaktorentwicklung

KfK 4120

Automatic Path-Planning for a Multilink Articulated Boom Within the Torus of a Fusion Reactor

D. Smidt

Kernforschungszentrum Karlsruhe GmbH, Karlsruhe

Kernforschungszentrum Karlsruhe GmbH
Postfach 3640, 7500 Karlsruhe 1

ISSN 0303-4003

Automatische Bahnplanung für ein mehrgliedriges Trägersystem im Torus eines Fusionsreaktors

Zusammenfassung

Für Wartungsarbeiten im Torus eines Fusionsreaktors wird ein Manipulator durch ein mehrgliedriges Trägersystem an seinen Einsatzort gebracht. Das Trägersystem wird als "articulated boom" bezeichnet. Trägersysteme dieser Art haben vier bis fünf Glieder und bewegen sich in der Mittelebene des Torus. Sie sind kinematisch redundant und haben einen sehr begrenzten Bewegungsspielraum.

In dieser Art werden automatische Verfahren für das kollisionsfreie Anfahren beliebiger Positionen innerhalb der Reichweite durch das letzte Glied beschrieben. Sie schließen das Ein- und Ausfahren des Trägersystems in bzw. aus dem Torus ein. Mit Hilfe der CAD-Simulation kann eine Tabelle sicherer Konfigurationen aufgebaut werden, die durch einen Feinpositionierungsalgorithmus ergänzt wird.

Abstract

For in-torus maintenance of fusion machines a manipulator is conveyed to the working area by a multilink-transporter, also called "articulated boom". Systems of this type have in general four to five links and move in the midplane of the torus. They are kinematically redundant and have a very restricted working space. In this paper automatic methods for the collision free approach of any position of the final joint within the reach of the transporter are presented, including insertion and removal. By automatic teach-in with the CAD-simulation a table of safe configurations can be generated and supplemented by a fine-positioning algorithm.

1. Introduction

For maintenance purposes within a torus of a fusion reactor an endeffector or manipulator will be conveyed to its working area by a transporter. In most cases the transporter is a multilink system, also called "articulated boom". It is mounted on a trolley by which it can be inserted into and removed from the torus. The axes of the joints are parallel to one another and so the transporter always moves in the torus midplane. Positions outside the midplane will only be reached by movement of the endeffector or manipulator. As an example Fig. 1a) shows the articulated boom for the JET (Joint European Torus), Fig. 1b) depicts a typical working position of the transporter within the torus.

The JET transporter consists of four links with five joints. The fifth joint carries the endeffector. As movement in a plane requires only two degrees of freedom to reach a certain target position, the system is kinematically undefined. The redundant degrees of freedom are required to avoid collisions between transporter and torus. However, these constraints do not completely resolve the problem of kinematic uncertainty.

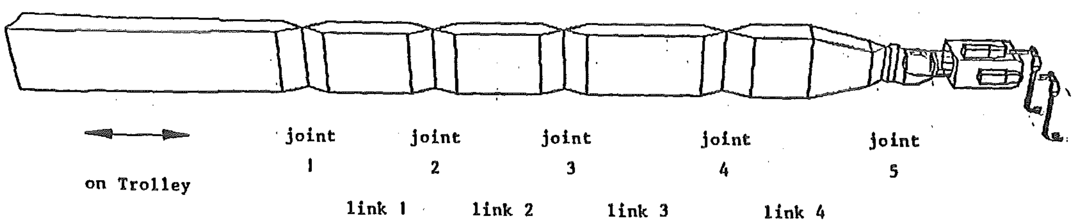


Fig. 1a): Transporter (articulated boom) with endeffector for JET (Joint European Torus)

Originally each joint was to be controlled separately by the operator, who observes the transporter with one or more videocameras. This is rather difficult, and therefore our group has developed a method /2/ which gives the operator a CAD image coupled with the transporter showing him a clear display of the whole transporter-scenario. The CAD model of the articulated boom is coupled to the movement of the transporter. Collision warning is generated on the basis of a CAD environment model.

The permitted angular positions of the joints for moving the endeffector step by step to the target position can be generated by a teach-in and stored for further use. However, such teach-in is a lengthy procedure. It must be supplemented by an interpolation procedure for fine positioning or continuous movement (e.g. for continuous driving along a surface for any cleaning or treatment). If the teach-in could be done automatically, it would be faster and more independent from human error.

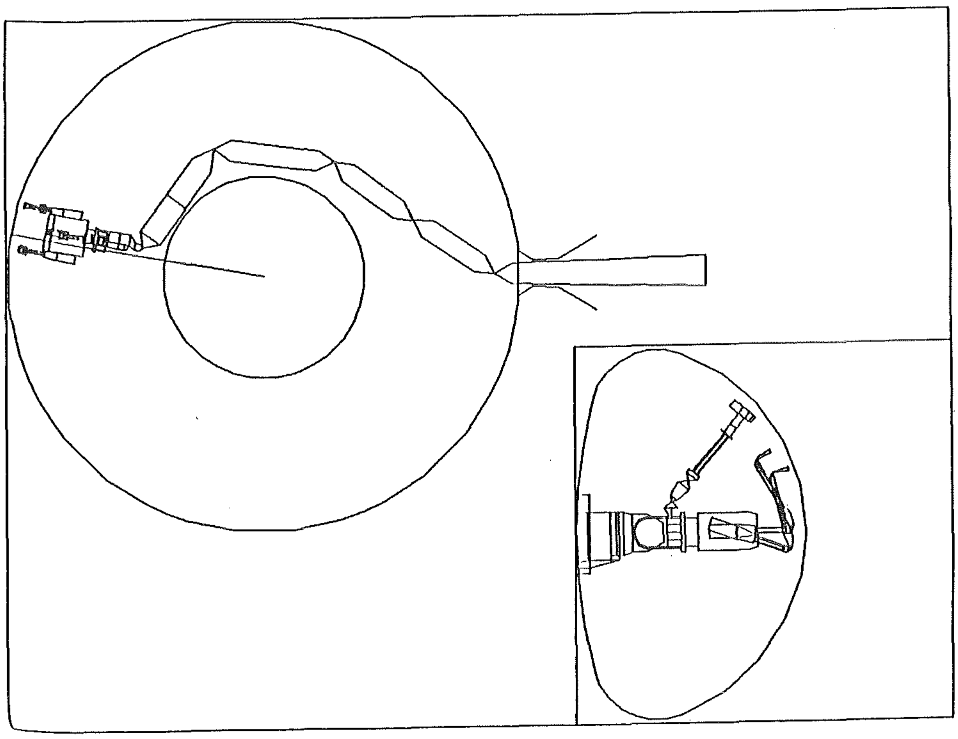


Fig. 1b): Articulated boom inserted into the torus
(orthographic representation)

In this connection it is helpful that with the given geometry the endeffector can be controlled independently, when the required position for joint No. 5 has been reached. The control of the endeffector is not subject of this report but shall be treated in later research.

In this report I will describe an automatic method for the teach-in, for the path-planning of the transporter. It is supplemented by an interpolation algorithm. Within the range of the boom joint No. 5 can be moved to any position without collision. This is done for two principal operations, namely

a) Insertion of the transporter into the torus to full length (respectively vice versa the removal of the transporter)

b) Approach to the target position of joint No. 5

2. Collision free Insertion and Removal of the Transporter

2.1 Collision Detection

To detect a collision or penetration between two objects is a basic problem of all CAD-methods and quite a number of solutions is available. In this work the torus and the transporter, moving in a plane, are coded as scenarios of polygons each modelled as a "quadtree". Object representation by quadtrees (octrees for the case of three dimensions) is in use since some years /3 - 5/. At edges or corners of objects the range of definition is subdivided into four quadrants (or eight octants). If necessary this subdivision can be recursively continued until a minimum scale is reached. Tree structures generated by this method permit in a simple way set operations as negation, intersection, and union. If the two trees generated from the torus and from the transporter have a not vanishing intersection, collision of the objects is indicated.

For this report I have implemented a quadtree method reported by Ayala et al. /6/. In this algorithm edges need not be resolved to the minimal scale, but are directly included into the tree with their line-parameters. Only corners require complete resolution. The knots of the tree, representing the quadrants of a region, are of the type

W = white, free area
B = black, inside of object
X = contains just one straight edge, characterized by the parameters of the line equation
G = grey, contains more than one edge and will be subdivided until minimum scale is reached or until it is described by W, B, X only.

A tree with four W-quadrants represents the empty set. For the torus the quadtree has to be generated only once, for the transporter this has to be done for each new configuration. With a Pascal program on the Intel 80286/8087 processors this takes about 100 secs for the torus configuration, about 20 secs for the transporter configuration.

The intersection operation is comparatively fast. If the intersection set is not empty, it delivers the origin coordinates XS and YS of the minimum scale quadrant finally encountered. Having the property X or B in quadrants of minimal scale with XS and YS the position of the collision at the torus and at the transporter and the evasive motion of the transporter can be determined. In general a minimal scale of 1/256 of the total display width was sufficient for the required accuracy.

2.2 Strategy for the removal of the transporter from the torus

The task of moving an object through an area cluttered with obstacles from a start to a target position is the central problem for the next generation of manipulating systems. Here Lozano-Perez, Brooks, Brady and other workers at the Artificial Intelligence Laboratory of the MIT have done pioneer work /7 - 11/. However, since their work refers to simple moving objects or conventional roboters in their typical environment, it could not be applied to the present problem. The transporter problem is characterized by a large number of joints, a very restricted workspace and the two-dimensionality which facilitates the solution. Therefore a new method had to be developed.

In the case considered here the transporter enters the torus by the upper opening and moves either into the right or into the left-hand side (Fig. 2). All links must be controlled in such a way that the fully inserted position can be reached without collisions. One can easily see that insertion and removal can be understood as the same procedure in opposite direction. Also the insertion and removal on the right-hand side of the torus are mirror-symmetrical to the respective procedures on the left-hand side. Therefore, one needs only to determine one of the four processes in-out and right-left and to generate a table of collision-free intermediate positions. This solution then can respectively be transferred to the other operations. For this purpose I use the pull-out from the fully inserted starting position. It is the only hand-made configuration. Pulling is reasonable since the transporter behaves kinematically as a chain and pulling a chain requires less additional constraints than pushing. By this, beginning with the shoulder, stepwise link after link is oriented parallel to the input-output-channel and pulled out. The other links follow according to the kinematic conditions as defined by the Denavit-Hartenberg-matrices. In case of a collision with the torus an evasive motion of the transporter is generated. This algorithm will be described in the following section.

2.3 Algorithms for the removal of the transporter (generation of a table of configurations)

The starting position is depicted in Fig. 2. The transporter is fully inserted on the right-hand side of the torus. The joints are numbered from 1 to 4 from shoulder to wrist, the respective angles are α_1 to α_4. The fifth joint, carrying the endeffector, is controlled independently by local criteria and is not considered here. The links have lenths l_1 to l_4. Stepwise a link is rotated by an angle $\Delta\alpha$, which has a predefined value, until it is parallel to the exit channel.

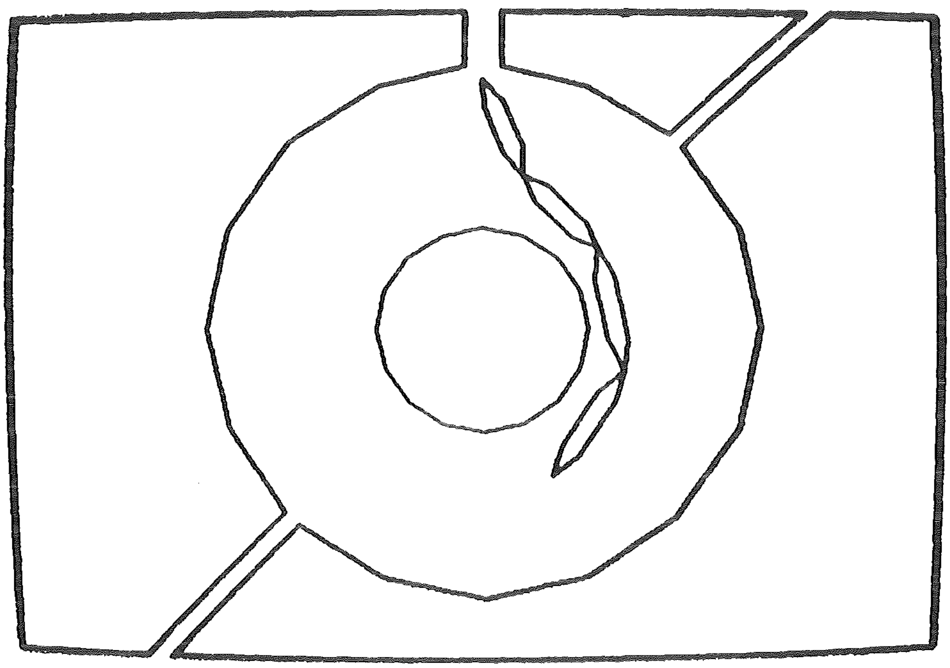


Fig. 2: Boom fully inserted at the right-hand side (endeffector omitted) This and all subsequent figures are screen-photographs. The torus consists of 4 polygons. The 45-degree gaps are planned as channels for the insertion of narrow objects. They are not used in the present work.

```
begin
i: = 1;
  While i < 4 do
  begin
    While α_i <>3*π/2 do
    begin
      α_i: = α_i-Δα;  (*rotation of link i to get parallel to trolley*)
      new kinematic computation of transporter
      build new quadtree for transporter
      collision test with torus-tree
        if collision then
          begin
          intersection operator generates intersection coordinates xs, ys
          from xs, ys get closest joint j
          also determine wether collision with inner or outer
          wall of torus
           → sign of movement to avoid collision
          if j < = i+1 then j: = j+1; ( *only joints > i may be changed* )
          repeat
          α_j: = α_j + sign *Δα
          until not collision
        end if;
      end while; (α_i = 3 *π /2)
      (*now similarly pull out link No. i+1*)
     i: = i+1;
    end while;
  end procedure.
```

Algorithm for generation of collision free positions at removal of the boom the torus

All configurations without collision are stored. Using $\Delta\alpha = \pi/16$ results in 27 configurations which are generated in about 4 minutes. This is quick compared to a teach-in controlled by the operator.

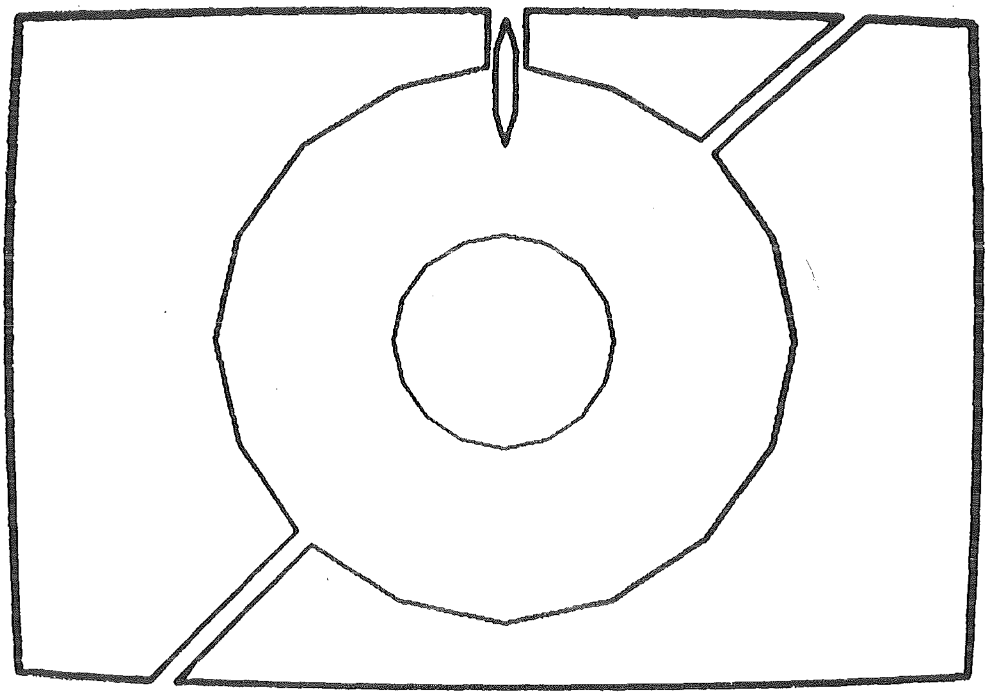


Fig. 3: Articulated Boom removed except link No. 4

3. Collision free Movement to a Target Position

3.1 General procedures

The required target position is entered into the system as a radius R and angle φ. Joint No. 5 must be transferred to this position. The orientation of the endeffector again is not considered here.

The approach of a position is executed in three steps:

1. Full insertion of the transporter on the required side

2. Coarse approach of position according to an automatically generated table of collision free positions

3. Fine positioning

Step No. 1 has been described in the preceeding chapter. The remaining steps will subsequently be detailed.

3.2 Algorithm for the coarse positioning (generation of a table of configurations)

A table of collision free configurations, which later can be used in both directions and in both sides, is generated by a pulling operation. The chain will be pulled from the fully inserted position on the right by turning away link No. 1 until it nearly touches the outer wall of the torus on the left-hand side. Afterwards link No. 2 can be turned in the same direction if necessary. This is shown in Fig. 4 for a sequence of steps. In all other respects the processes are analog to the one described in chapter 2.3. After each step the system is examined for collision and moved away if necessary. The generated collision free configurations of the α_i are attached to the corresponding angular position φ of link No. 5 in the torus. Once the table is available, for any start and target position φ the closest entries in the table have to be found. Between them they define the sequence of configurations which has to be passed.

In principle this automatic teach-in results in moving of the joints 1 and 2 and pulling the chain along the inner wall of the torus to the target positon. For safety reasons the inner radius can be increased compared to the real value to obtain additional distance.

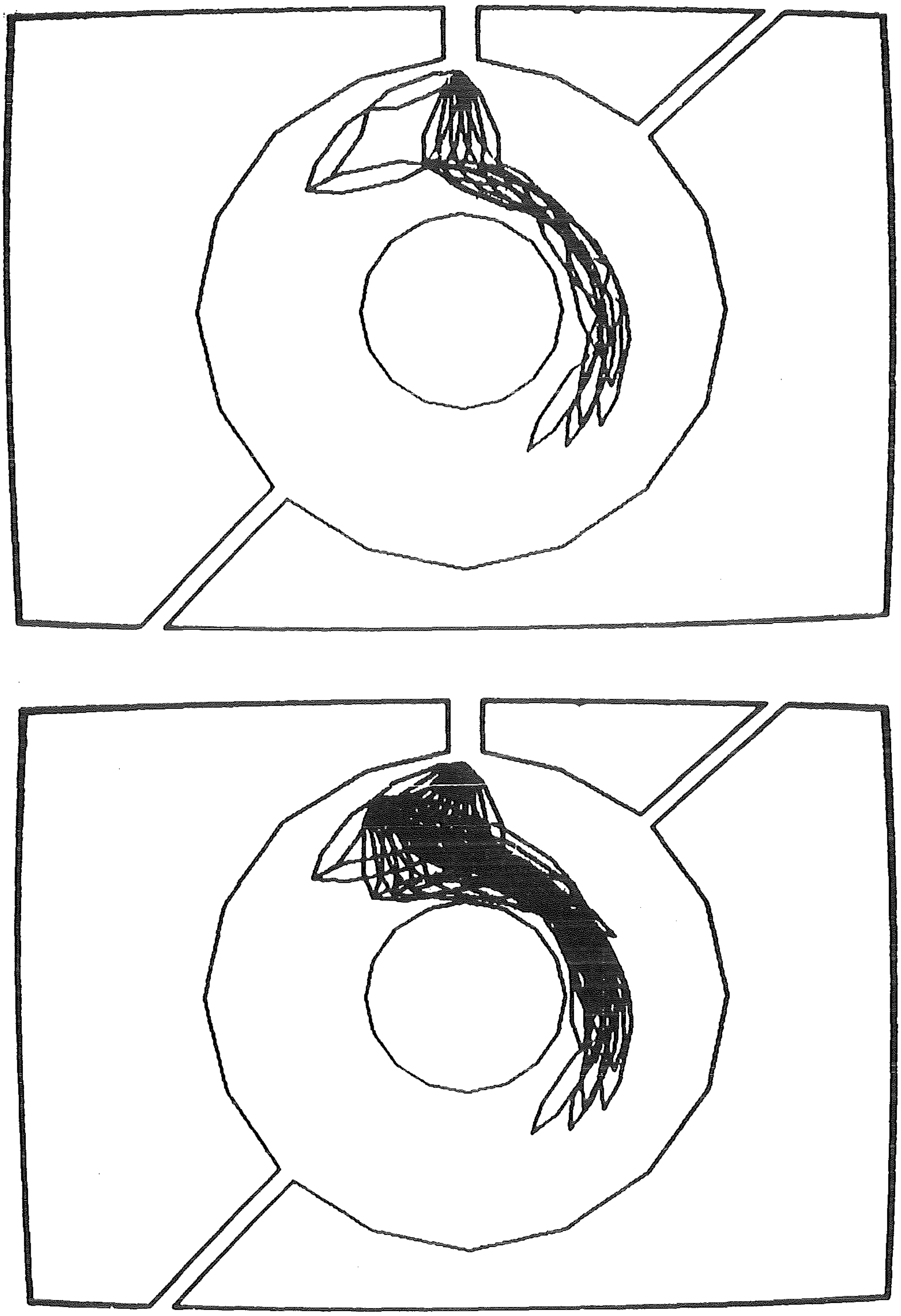


Fig. 4: Series of intermediate positions to reach the target configuration visible as the leftmost boom position
a) Sequence turning link No. 1 to the left
b) After link No. 1 is in the final position, link No. 2 is turned, collisions are repaired by turning other links

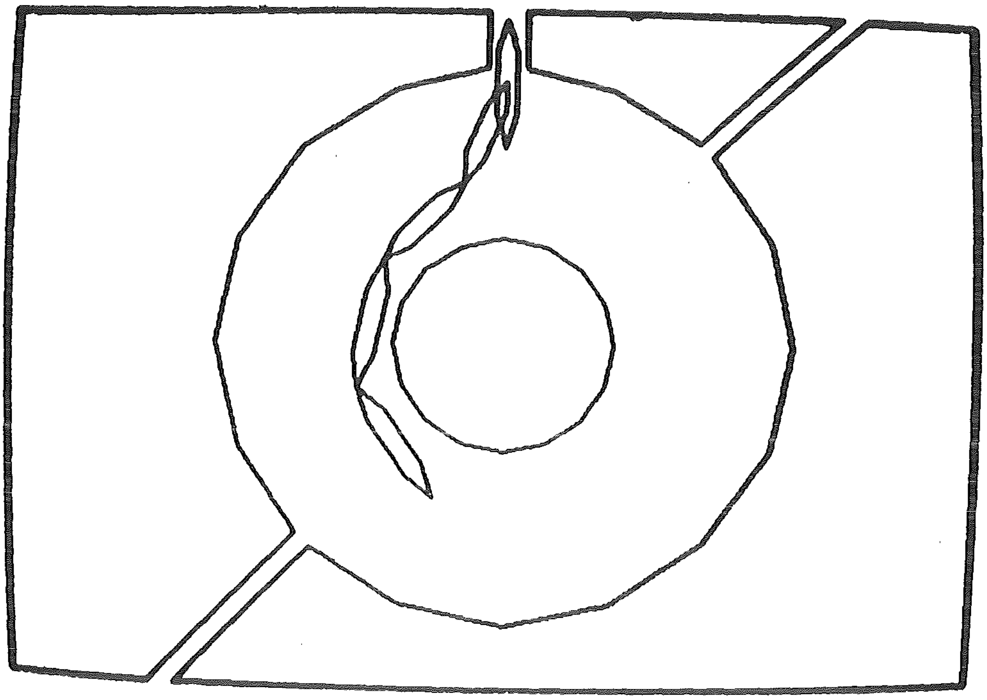


Fig. 5: Insertion and positioning on the left are based on the same table and a mirror transformation

3.3 Fine positioning

Using the procedures described in the preceding chapter the end joint 5 can be moved into the proximity of the required angular position within the torus. So far the radial position is random, but close to the inner wall of the torus. For the final approach to the target position a correcting algorithm has been developed, based on the Jacobian of the configuration.

Generally the cartesian coordinates x, y of joint 5 depend on the angles of the other links. With x_0 and y_0 as coordinates of the shoulder joint we have

(1) $x = f(\alpha_1, \alpha_2, \alpha_3, \alpha_4) + x_0$

(2) $y = g(\alpha_1, \alpha_2, \alpha_3, \alpha_4) + y_0$

Corrections in the positions x and y can be described as displacements dx and dy. While the change is reasonably small one gets

$$(3) \quad dx = \frac{\partial f}{\partial \alpha_1} d\alpha_1 + \frac{\partial f}{\partial \alpha_2} d\alpha_2 + \frac{\partial f}{\partial \alpha_3} d\alpha_3 + \frac{\partial f}{\partial \alpha_4} d\alpha_4$$

$$(4) \quad dy = \frac{\partial g}{\partial \alpha_1} d\alpha_1 + \frac{\partial g}{\partial \alpha_2} d\alpha_2 + \frac{\partial g}{\partial \alpha_3} d\alpha_3 + \frac{\partial g}{\partial \alpha_4} d\alpha_4$$

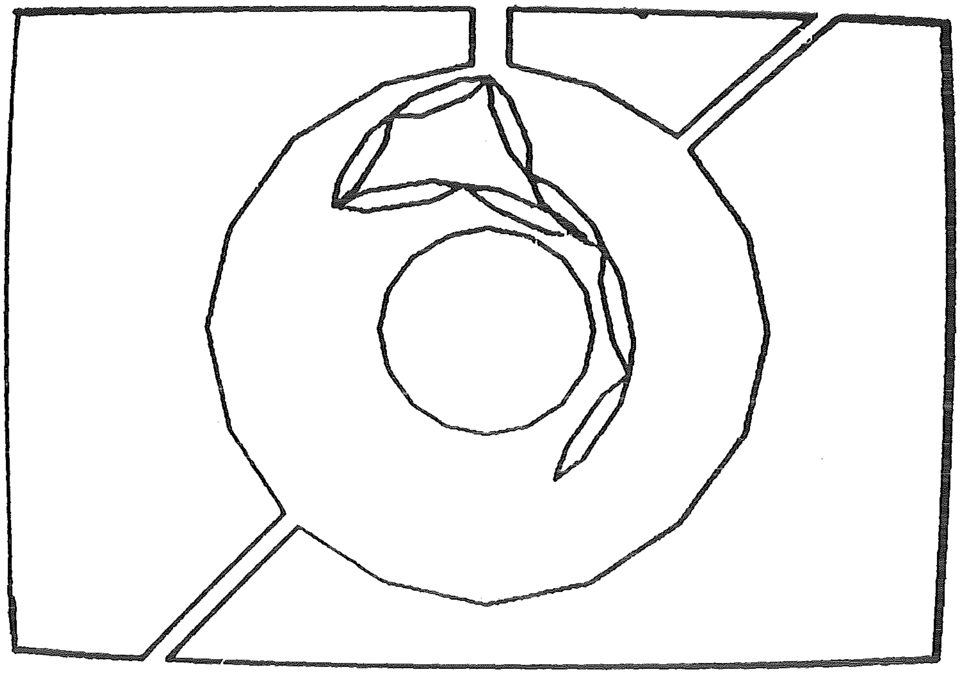
The $\frac{\partial f}{\partial \alpha_i}$ and $\frac{\partial g}{\partial \alpha_j}$ are known sin and cos functions.

As it is wellknown the Jacobian is the coefficient matrix of (3) and (4). From (3) and (4) the coefficients of maximal size $\frac{\partial f}{\partial \alpha_i}, \frac{\partial g}{\partial \alpha_j}$ are determined and neglecting all other coefficients one gets

$$(5) \quad dx = \frac{\partial f}{\partial \alpha_i} d\alpha_i$$

$$(6) \quad dy = \frac{\partial g}{\partial \alpha_j} d\alpha_j$$

For any desired displacements dx and dy a very simple equation for the change of some angles $d\alpha_i$, $d\alpha_j$ can be determined. This algorithm must be repeated until the target position has been reached. Fig. 6 shows an example. However, for certain configurations this method fails. As an example the configuration in Fig. 7a) results in maximal coefficients for joint 3 and 4. Since both links are oriented in the same direction towards the target position, the iteration will not converge. Therefore after a certain number of unsuccessful iteration steps the presently moved joints are stopped and the others moved. Now the target position is reached (Fig. 7b).



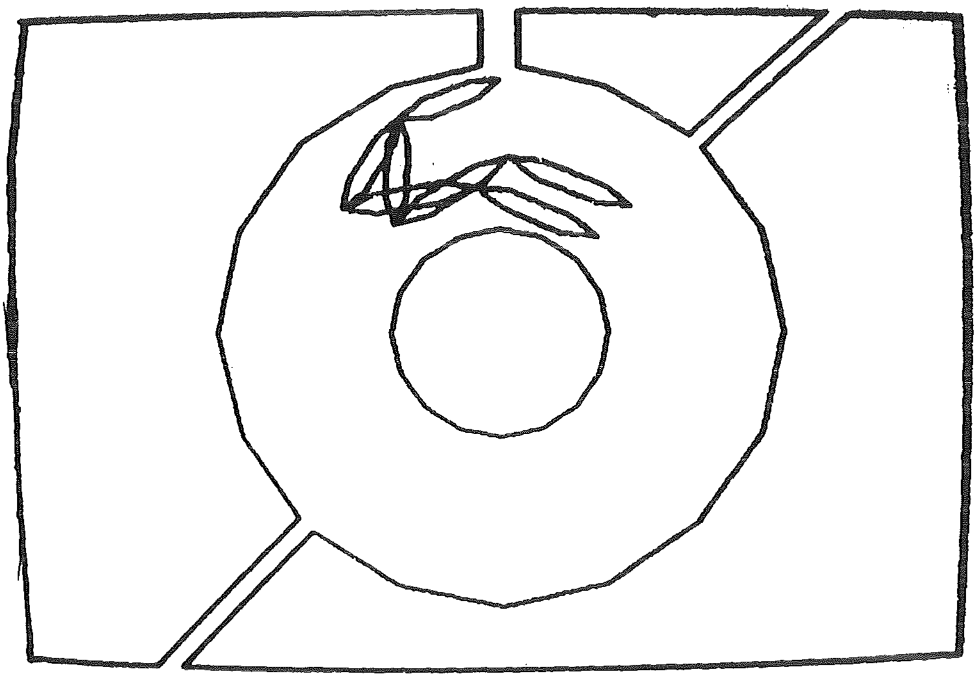


Fig. 6a): By coarse positioning (table) the transporter is moved to the position $\varphi = 45^{\circ}$

Fig. 6b): By fine positioning another radial position at $\varphi = 45^{\circ}$ can be approached without further collision control

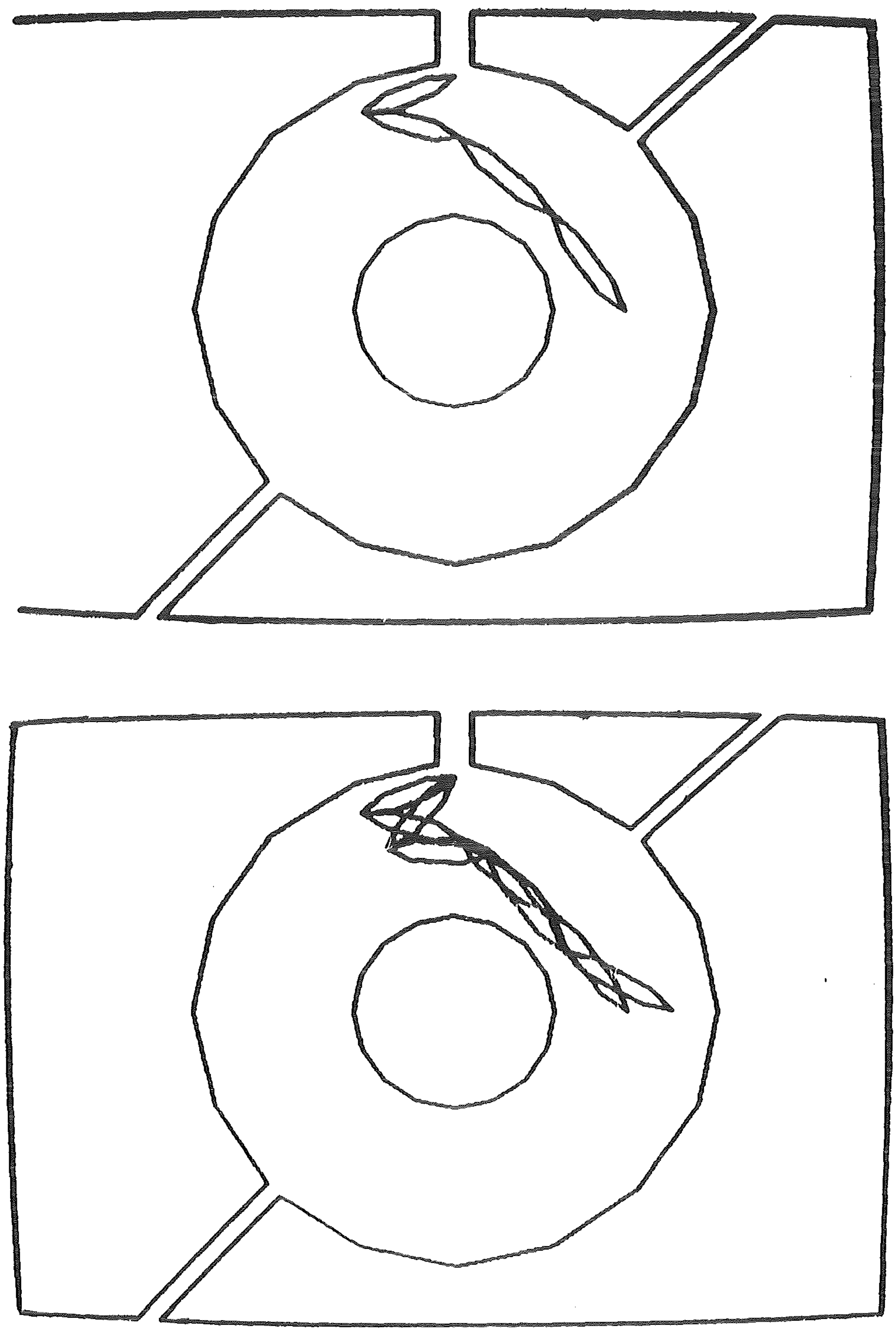


Fig. 7a): In this example the use of the Jacobian with maximal coefficients fails, if an outer radial position must be reached

Fig. 7b): By using other joints, it succeeds

The starting position for the fine adjustment is the end position of the coarse approach. According to the principle of the method this generally is close to the inner wall of the torus. Therefore the fine adjustment moves joint 5 generally outward and away from the nearest collision range. Therefore it is not necessary to test for collisions in connection with the fine adjustment. This results in a faster computation.

4. Future Work

The described method permits the automatic generation of the complete field of movements of the transporter in the torus. In a next step the respective steps shall be done for the endeffector to enable the system to do arbitrary and complex operations.

References

/ 1/ P.D.F. Jones, D. Maisonnier, T. Raimondi, Design and Operation of the JET Articulated Boom, 11th Symp. on Fusion Engineering, Austin, Texas, Nov. 1985

/ 2/ K. Leinemann, E.G. Schlechtendahl, Computer Graphics Support for Remote Handling Simulation and Operation, Proc. 32nd Conf. on Remote Systems Technology, New Orleans, June 3-7, 1984, Vol. 1, pp. 10-16

/ 3/ G.M. Hunter, K. Steiglitz, Operations on Images Using Quadtrees, IEEE Trans. on Pattern Analysis and Machine Intelligence 1, p.145-153, April 1979

/ 4/ G.M. Hunter, K. Steiglitz, Linear transformation of pictures represented by quadtrees, Computer Graphics and Image Processing, Vol. 10, 289-296, 1979

/ 5/ C.L. Jackins, S.L. Tanimdo, Oct-trees and their use in representing three-dimensional objects, Computer Graphics and Image Processing, Vol. 14, 249-270, 1980

/ 6/ D. Ayala, P. Brunet, R. Juan, I. Navazo, Object Representation by Means of Nonminimal Division Quadtrees and Octtrees, ACM Transactions on Graphics, Vol. 4, No. 1, 11-59, Jan. 1985

/ 7/ Tomas Lozano-Perez, Spatial Planning, A Configuration Space Approach, IEEE Transactions on Computers C-32, Febr. 1983, 108-120

/ 8/ Tomas Lozano-Perez, Automatic Planning of Manipulator Transfer Movements, IEEE Transactions on Systems, Man, and Cybernetics, SMC-11, No. 10, 1981, 681-698

/ 9/ J.M. Brady et al., Robot Motion: Planning and Control, MIT Press, Cambridge, MA, 1983

/10/ Rodney A. Brooks, Solving the Find-Path-Problem by Good Representation of Free Space, IEEE Transactions on Systems, Man, and Cybernetics, SMC-13, 1983, 190-197

/11/ Bruce R. Donald, Motion Planning with Six Degrees of Freedom, Thesis, Report AI-TR-791, MIT, Cambridge, MA