*Scientia Iranica*, Vol. 13, No. 1, pp 83–90

SCIENTIA IRANICA

# A Minimum Route for Machine Tool Travel

**M.A. Rahbary**[1]

This paper presents an algorithmic approach to solving the problem of excessive travel in C.N.C. machine tools, by introducing an efficient method to compute the shortest path between given sets of points (origin and destination) in an $R^2(x, y)$ plane. When a work piece is located (as an obstacle) between sets of points, it is proved that the optimum path between these points would be formed by sequences of connected straight line segments whose intermediate end points are vertices of an appropriate polygonal (closed control barrier). The case of one origin, one destination and a set of barriers is considered. This method is computationally efficient.

## INTRODUCTION

The study of numerically controlled machine tool systems indicates that there are many elements which play a major role in increasing cycle times. Since manufacturing efficiency is directly influenced by machining cycle time, the attention given to its reduction has always been great. One of the main elements, which contributes to the problem of higher cycle time, is the inefficient travel of cutting tools.

The central objective of this paper, therefore, is the development and initial implementation of an algorithm for planning a suitable path for cutting tools to travel, which results in a reduction of the cycle time for manufacturing a part on a machine tool. The problem of finding an optimum path between given points in the presence of barriers has received some attention in the past, but there have been very few publications of the findings. Some of these algorithms are proven to be computationally inefficient or are difficult to implement for a particular case.

Of these published algorithms, the oldest is in [1] where the authors studied the problem of displacement of an autonomous vehicle on Mars and the latest is in [2] where the author investigated strategies of cutter path optimization. Shkel and Lumelsky [3] investigated the problems of finding the shortest smooth path between two points in the plane for robotics applications. Vaccaro [4] discussed the problem of routing an urban vehicle. The routing was considered independently, but the barriers were also represented by line segments.

Wang Dahl, Pollock and Woodward [5] investigated the shortest pipeline layout between two points on a ship (with polygon barriers). Larson and Li [6] allowed the polygons to be convex or non-convex, but only the rectilinear ($L_i$) norm was considered. Viegas [7] developed an $O(n^3)$ algorithm for the generation of the shortest-paths network in the context of evaluating pedestrian displacements from home to public facilities in towns, using Euclidean distance.

1. *Department of Manufacturing Engineering, Tabriz University, Tabriz, I.R. Iran.*

## ASSUMPTIONS AND PROBLEM FORMULATION

It is desired to route a tool tip from a specified start position to another position within a two-dimensional work space, avoiding interferences with barriers placed in between the positions. If one or more paths exist, as usually is the case, the shortest route must be found.

The following assumptions are made:

1. Only the case of a single origin and destination in the presence of a single barrier and in the plane $R^2$ is considered. However, the technique is extendable to more than one barrier and sets of origins and destinations. It could also be defined to work in a three-dimensional work space;
2. All boundaries (of the space and of the barrier) are composed of straight-line segments;
3. It is required to find the shortest distance between any origin and destination, such that these paths do not cross any barrier;
4. Barriers are convex or non-convex closed polygons;
5. No points of origin or destination are located inside or on the barriers;
6. The route should not intersect any barrier.

## METHOD

The principal problem of finding an optimum path between a set of points (origin and destination) is broken down into a sequence of minor or local problems, which can be dealt with one at a time.

A set of origins, $O = \{o_1\}$, and destinations, $D = \{d_1\}$, in the plane $R^2$, and a set of barriers to travel, $B = \{b_1\}$, are introduced. Before proceeding with any calculation of an optimum path for any point, it is, first, necessary to reduce the problem to a network or matrix representation. To do this, it is required to define control lines drawn parallel to the boundaries of the barrier in question, in order to define the control barrier. Control points on the control barrier are the only defined points through which an optimum path could pass.

Geometrically, a 'control point' is defined to be the intersection of two straight lines that are adjacent members of the set of straight lines forming the super scribing polygon (control polygon or barrier) around a piece of equipment. The sides of this polygon are a prescribed perpendicular clearance distance (say $d$) from their respective parallel equipment boundaries.

Figure 1 illustrates the generation of such control points $A$, $B$, $C$, $D$, $E$, $F$, $G$ and $H$ for a polygon.

The following notation is used: The barrier (defined by control points) denoted '$b$' (control barrier), is defined by the list of its vertices, $V_b = \{V_0, V_1, V_2, V_3, \cdots V_m\}$, where $(m + 1)$ denotes the number of vertices of the barrier. This list corresponds to successive vertices from one end of the barrier to the other.

## PROPERTIES OF SHORTEST PATHS NETWORK

A few concepts and some theories are first introduced:

1. A point is said to be visible [8] from a point if and only if, the straight line segment joining these two points crosses no barrier. To solve the problem of inter-visibility as a practical matter, it is important to describe a way to determine the visibility of one point from another. Here, a straightforward technique follows. Consider the situation in Figure 2. The segment $[P_1 \quad P_2]$ joining points $P_1$ and $P_2$, is a boundary or a control line. It can be said that the line joining $P_1$ and $P_2$ consists of all points $(x, y)$, such that:

$$x = \alpha x_{p1} + (1 \quad \alpha) x_{p2},$$

$$y = \alpha y_{p1} + (1 \quad \alpha) y_{p2}, \tag{1}$$

where:

$$0 \leq \alpha \leq 1.$$

Similarly, the line joining $i$ and $j$ consists of all points $(x, y)$, such that:

$$x = \beta x_i + (1 \quad \beta) x_j,$$

$$y = \beta y_i + (1 \quad \beta) y_j, \tag{2}$$

where:

$$0 \leq \beta \leq 1.$$

Solving Equations 1 and 2 simultaneously for $\beta$ and $\alpha$ gives:

$$\beta = \frac{(x_{p2} \quad x_j)(y_{p1} \quad y_{p2}) \quad (y_{p2} \quad y_j)(x_{p1} \quad x_{p2})}{(x_i \quad x_j)(y_{p1} \quad y_{p2}) \quad (y_i \quad y_j)(x_{p1} \quad x_{p2})},$$

$$\alpha = \frac{(x_{p2} \quad x_j)(y_i \quad y_j) \quad (y_{p2} \quad y_j)(x_i \quad x_j)}{(x_i \quad x_j)(y_{p1} \quad y_{p2}) \quad (y_i \quad y_j)(x_{p1} \quad x_{p2})}.$$

For $i$ and $j$ not to be visible, both conditions: $0 < \alpha < 1$ and $0 < \beta < 1$ must be true.

For all other values of $\alpha$ and $\beta$, $i$ and $j$ are visible. If:

$$(x_i \quad x_j) * (y_{p1} \quad y_{p2}) \quad (y_i \quad y_j) * (x_{p1} \quad x_{p2}) = 0,$$

the lines are parallel, there is no intersection and $i$ and $j$ are visible.

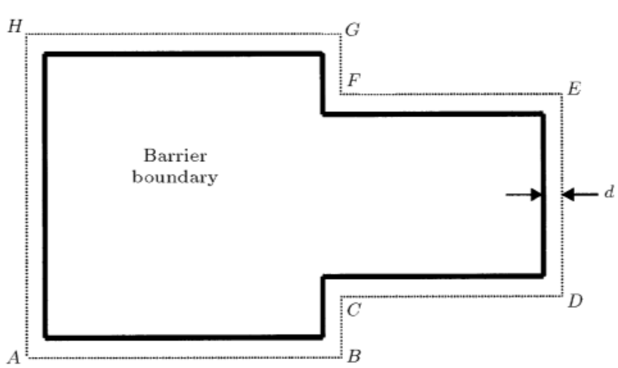


**Figure 1.** Control points for a closed polygon.

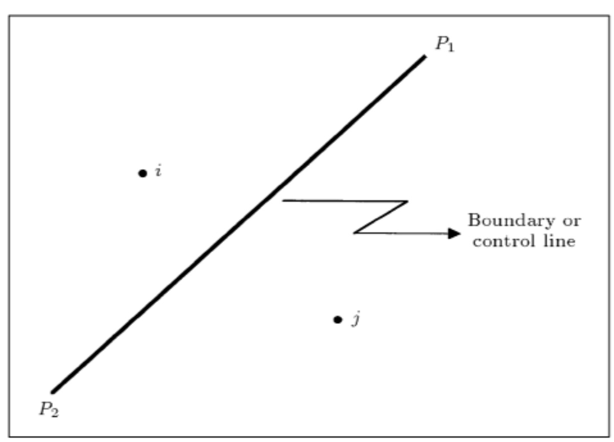


**Figure 2.** Geometry for visibility calculation.

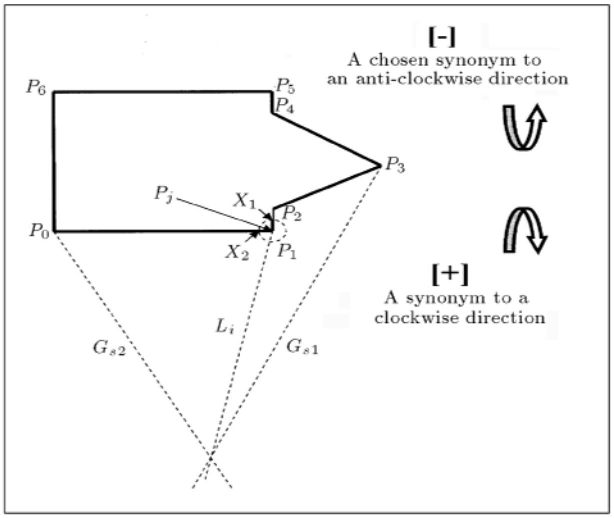


**Figure 3.** Locally and globally supporting lines of a barrier.

2. The line $L_i$ (see Figure 3) is locally supporting the barrier, $b$ (closed polygon), because:
   - (i) $L_i$ contains one vertex at least of $b(P_1)$;
   - (ii) All the points belonging to the intersection of an arbitrarily small circle, $C$, of radius "$r$" centered at point $P_j$, lie on one of the two closed half-planes defined by $L_i$, subject to the direction in which the shortest path is considered. If the negative ( ) direction is considered, then, the intersection points should lie on, or to, the left of the half-plane, or else to the right of the half-plane.

   $L_i$ is a locally supporting line of the barrier at $P_1$. The intersection points $x_1$ and $x_2$ are entirely located to one side of the line $L_i$, (left half-plane);
3. The line $G_{s1}$ is globally supporting barrier $b$ (closed polygon), because:
   - (i) The line $G_{s1}$ contains one vertex at least of barrier $b$ $(P_3)$;
   - (ii) The line $G_{s1}$ satisfies the Case (2) and, therefore, locally supports the barrier, $b$, at $P_3$;
   - (iii) All the vertices of the barrier, $b$, lie entirely on, or to one side of the line. This condition is also subject to the direction of scanning.

   $G_{s1}$ is a globally supporting line of the barrier, $b$, at $P_3$ and also a locally supporting line at $P_3$ (scanning in a ( ) direction), where the vertices ($P_0$, $P_1$, $P_2$, $P_3$, $P_4$, $P_5$, $P_6$), lie entirely to one of two sides defined by the segment $G_{s1}$. The line, $G_{s2}$, is also globally supporting the barrier, $b$, at $P_0$ (scanning in a ( ) direction). From the situation discussed above, it can be said that in any $x/y$ plane ($R^2$) and in the presence of a single point and a barrier, there can only be two globally supporting lines to that barrier ($G_{s1}$, $G_{s2}$), as in Figure 3.

### Theorem

When distances are measured with Euclidean distance, any optimum path is composed of connected straight line segments, where;

(a) $x_o$ is an origin;

(b) $x_d$ is a destination;

(c) $v_j$ with $j = 0, 1, 2, \cdots, k$ is a vertex of the barrier $b(j)$, such that the shortest path connecting $x_o$ and $x_d$ intersects the barrier at its vertices and each intersection point is the second point of the line segment, which, in turn, is a supporting line of the barrier at that intersection point.

### Proof

As a shortest path consists of a straight line between two mutually visible points, consider three consecutive line segments $[x_o, v_{j+1}]$, $[v_{j+1}, v_{j+2}]$ and $[v_{j+2}, x_d]$, as illustrated in Figure 4, on the shortest feasible path. The path is assumed to be nonlinear, otherwise $v_{j+1}$ and $v_{j+2}$ would be irrelevant and may be deleted from the path.

Now, consider points $x_1'$, $x_1''$, $x_2'$ and $x_2''$ at an arbitrarily small distance, $(r > 0)$, from $v_{j+1}$ and $v_{j+2}$, respectively. Consider another path, which is composed of $[x_0, x_1']$, $[x_1', x_1'']$, $[x_1'', x_2']$, $[x_2', x_2'']$ and $[x_2'', x_d]$, which is shorter than the path composed, $[x_o, v_{j+1}]$, $[v_{j+1}, v_{j+2}]$ and $[v_{j+2}, x_d]$. This is true because of the triangle inequality. So, the shortest path, which consists of segments of $[x_o, x_1']$, $[x_1', x_1'']$, $[x_1'', x_2']$, $[x_2', x_2'']$ and $[x_2'', x_d]$, cannot be a feasible one and must be deleted. The earlier path, with $x_1'$ located on the segment $[x_o, v_{j+1}]$ and $x_1''$ on $[v_{j+1}, v_{j+2}]$, is

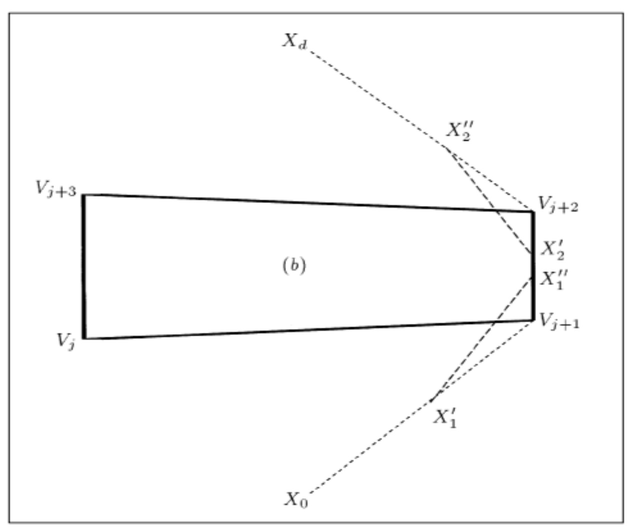


**Figure 4.** Shortest path and barrier $b$ for the proof of the theorem.

entirely positioned on one half defined by the line $[x_o, v_{j+1}]$ and the vertices of the barrier, $b$, are also located on that half.

So, one can say that the line segment $[x_o, v_{j+1}]$ is a globally supporting line of barrier $b$ at point $V_{j+1}$. With the same principle, $[x_d, v_{j+2}]$ is a globally supporting line of the barrier $b$ at point $v_{j+2}$. One can also say that the only feasible shortest path connecting $x_0$ to $x_d$ is a path which consists of the straight line segments $[x_o, v_{j+1}]$, $[v_{j+1}, v_{j+2}]$ and $[v_{j+2}, x_d]$.

## GENERAL SOLUTION

### Initialization

Select the total of $N_b$ control points and let $(X_{ib}, Y_{ib})$ be the coordinate of the $i$th control point where;

$$i_b = 1, 2, 3, \cdots, N_b.$$

By convention, $o_b$ is assigned as the origin and $d_b$ as the destination. Points $o_b$ and $d_b$ are said to be visible if a straight line joining them does not intersect the boundary of the control lines (visibility test theorem).

If $o_b$ and $d_b$ are visible, then, one can define the path which is composed of a straight line, $[o_b, d_b]$, and the problem is solved. Otherwise, compute the distance, $r_{ib}$, from $o_b$ and $d_b$ to every point $(P_{ib})$ of $X_b$, where $X_b$ is associated with the set of vertices of the barrier:

$$X_b = \{P_1, P_2, P_3, \cdots, P_{nb}\},$$

$$R_{ob} = \{r_{1b}, r_{2b}, r_{3b}, \cdots, r_{mb}\},$$

$$R_{db} = \{r_{1b}, r_{2b}, r_{3b}, \cdots, r_{mb}\}.$$

The set, $R_{ob}$, is associated with the set of distances which are related to $o_b$ and vertices in $X_b$ and the set, $R_{db}$, is related to $db$ and vertices in $X_b$. Let $r_o$ be the minimum distance associated with $o_b$ and $r_d$ with $d_b$. Now, find the corresponding vertices of these minimum members from set $X_b$ and let them be defined as $P_0$ and $P_d$.

### Scanning Phase

Now, precede scanning in clockwise and anti-clockwise directions for $o_b$ and $d_b$, respectively, starting from the barrier vertex corresponding to the minimum distances $(r_o, r_d)$ and their respective points ($P_o$ and $P_d$). Compute the globally supporting points (see sets $G_{ob}$ and $G_{db}$). In either direction, for both $o_b$ and $d_b$. Let:

$$G_{ob} = \{P_i\},$$

and:

$$G_{db} = \{P_i\}.$$

If sets $G_{ob}$ and $G_{db}$ shared a common member (let $P_i$ be the common member) in their sets, then compute that point. Connect $o_b$ to $P_i$ and $d_b$, where $o_b$ and $d_b$ must be inter-visible via $P_i$. The set should be computed as:

$$S_b = \{o_b, P_i, d_b\}.$$

If the condition mentioned for the sets was not satisfied, proceed with scanning.

The scanning should start in both directions (clockwise and anticlockwise), from the first globally defined point and ending at the destination point, $d_b$.

Any segment $(P_j, P_k)$, where $P_j$ and $P_k$ represent the start and end points of a globally defined line, may be handled by the following procedures (now the tests are being carried out on the segments of the barrier itself, $P_j$ and $P_k$ are the vertices of the barrier):

1. Start, for example, with $P_j$ in an anticlockwise direction. Take the segment $(P_j, P_{j+1})$ and carry out the globally supporting lines test. If $P_{j+1}$ passes the test, insert it in the list of live vertices (set $S_{1b}$);
2. Go to the next vertex (say in the anticlockwise direction, $P_{j+1}$) and carry out Step 1. Continue the procedure until one of the segment's end points is also a member of the set, $G_{db}$. Now, complete $S_{1b}$ by computing the last point. The set, $S_{1b}$, represents the first feasible path;
3. Continue the steps above in a clockwise direction, starting with $P_j$. Complete $S_{2b}$ by computing all the barrier's vertices which pass the test;
4. $S_{1b}$ and $S_{2b}$ hold two different sets of paths which connect $o_b$ to $d_b$. Using a Euclidean distance calculation, select the set which produces the optimum path.

The path with the minimum value is the shortest feasible path which connects $o_b$ to $d_b$ and satisfies the conditions for an optimum path.

## EXAMPLE OF HOW THE ALGORITHM WORKS

Consider now a small example, using Euclidean distance, in Figure 5. Select the the control points as starting with barrier vertices (0-7), destination $d_b$ and origin $o_b$. Register the Cartesian coordinates of these points, as presented in Table 1. To check the visibility of $o_b$ and $d_b$ an algorithm is executed for visibility presented earlier. If these points are inter-visible, terminate the process, or else, resume with the following procedures:

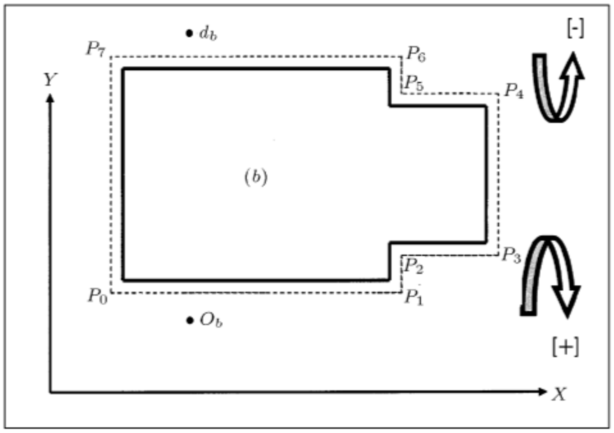


**Figure 5.** Control geometry for the example.

**Table 1.** Coordinate for the points of the example.

| Point | No. | *X* | *Y* |
|---|---|---|---|
| Barrier vertices | $P_0$ | 4 | 6 |
| | $P_1$ | 9 | 6 |
| | $P_2$ | 9 | 7 |
| | $P_3$ | 11 | 7 |
| | $P_4$ | 11 | 9 |
| | $P_5$ | 9 | 9 |
| | $P_6$ | 9 | 10 |
| | $P_7$ | 4 | 10 |
| Origin | - | 6 | 5 |
| Destination | - | 6 | 11 |

1. Compute the Euclidean distance of $o_b$ and $d_b$ from all the vertices of barrier $b$ (Table 2);
2. Select the vertex corresponding to the smallest element in both lists ($P_0$ and $P_7$, respectively);
3. Scan for $o_b$ in an anticlockwise ( ) direction, starting with the smallest element in Table 2, as found in the above step ($P_0$). Note that the condition for globally definable lines for a ( ) direction is not the same as for a (+) direction, (see the definition of globally defined lines in the previous section);
4. Segment $[o_b \quad P_0]$ does not satisfy the conditions for a globally supporting line and, therefore, is deleted. $[o_b \quad P_0]$ satisfies the conditions and, therefore, the segment $[o_b \quad P_1]$ should be registered. Proceed scanning in the opposite direction (+). The line, $[o_b \quad P_1]$, satisfies the condition and, so, should be computed. The completed set should be computed as:

$$G_{ob} = \{P_1, P_0\}.$$

5. Resume the scanning operation, starting with point $d$ and carry out Steps 2 to 4. The completed set should be computed as:

$$G_{db} = \{P_7, P_6\}.$$

6. Sets $G_{ob}$ and $G_{db}$ do not have a common element and scanning should be resumed. Move to point $P_1$, ($P_1 \varepsilon G_{ob}$) and scan in the ( ) direction, starting with the segment $[P_1, P_2]$. $[P_1, P_2]$ is a segment connecting two vertices of the barrier and is subject to globally supporting lines definition tests:
   a) At $[P_1, P_2]$; 7:$[P_1, P_2]$ does not satisfy the condition for a globally supporting line;
   b) At $[P_2, P_3]$; 8:$[P_2, P_3]$ supports the conditions for a globally supporting line and should be computed;
   c) At $[P_3, P_4]$; 9:$[P_3, P_4]$ satisfies the condition for a globally supporting line at $P_4$;
   d) At $[P_4, P_5]$; 10:$[P_4, P_5]$ does not satisfy the conditions for a globally supporting line;
   e) At $[P_5, P_6]$; 11:$[P_5, P_6]$ satisfies the conditions for a globally supporting line at $P_6$;
   f) At $[P_6]$; 12:$P_6$ is a supporting vertex, which is also a member of $G_{db}$. So, $P_4$ and $d_b$ are visible via the vertex $P_6$. So, compute the set $S_1 = \{o_b, P_1, P_3, P_4, P_6, d_b\}$, which comprises the line segments $[o_b, P_1]$, $[P_1, P_3]$, $[P_3, P_4]$, $[P_4, P_6]$ and $[P_6, d_b]$.

The sweep is terminated and the segments which complete the path are registered. The sweep from $o_b$ towards $d_b$ could be achieved on either side of the barrier, $b$ (clockwise ( ) or anticlockwise (+) directions). Now, proceed in the opposite direction, which, in this case, is the clockwise (+) direction and repeat procedures 2-13. The result of this sweep can be shown as the set $S_2 = \{o_b, P_o, d_b\}$ and the line segments $[o_b, P_o][P_o, P_7]$ and $[P_7, db]$.

Set $S_1$ and $S_2$ represent the two different patterns which define the shortest path to connect $o_b$ to $d_b$

**Table 2.** The distances for the example.

| Points No. | Distance (Origin) | Distance (Destination) |
|---|---|---|
| $P_0$ | 2.23 | 5.38 |
| $P_1$ | 3.16 | 5.38 |
| $P_2$ | 3.60 | 5.00 |
| $P_3$ | 5.38 | 6.40 |
| $P_4$ | 6.40 | 5.38 |
| $P_5$ | 5.00 | 3.60 |
| $P_6$ | 5.83 | 3.16 |
| $P_7$ | 5.38 | 2.23 |

without crossing the barrier in either direction. One of these sets ($S_1$ and $S_2$) contains the optimum route. By comparing the two, $S_2$ with the segments $[o_b, P_o]$, $[P_o, P_7]$ and $[P_7, d_b]$, can be seen to be the optimum.

## IMPLEMENTATION IN MANUFACTURING

Computer Numerical Control (CNC) machine tools are amongst the most important and most complex machines in the manufacturing world. The increasing complexity of engineering components requires an increase in safety, productivity, system reliability, greater operator satisfaction and a significant reduction in machine down-time in the implementation of these machines. The introduction of the algorithms (shortest path and visibility) in this paper is aimed at producing a method which could be implemented in manufacturing to reduce the machining cycle time by increasing the speed and accuracy of the operations.

Although these algorithms have many implementations, three stages have been introduced, where they can significantly improve speed and accuracy in CNC machine tool operations.

It must be emphasized that such techniques have not been used in any of the modern CAD/CAM systems, due to complexity in the C.N.C. operating environment. The method introduced makes the implementation of such an approach computationally possible, as shown in this paper.

These algorithms could be used in finding a suitable path between any two spatial points in a continuously changing environment. This changing environment could be due to alteration in a component's geometry or in the positions of any moveable part of the machine tool. This concept is best described in Figure 6.

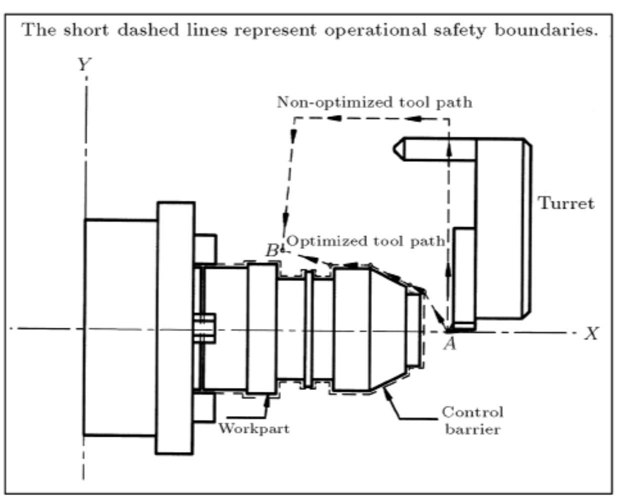


**Figure 6.** Non-optimized and optimized tool path between points $A$ and $B$.

The algorithms could also be used in optimising the indexing position during the cutting operations. The position control for indexing operations would be based on safety and speed. Figures 7a and 7b describe the concept.

The position control of the tool tip could also be achieved by implementing the algorithms for this purpose. The position of the tool tip is continuously monitored in reference to the next position and is computed according to the safety margin created around the work-piece. This concept is also described in Figures 8a and 8b.

Figure 9 shows how the complete system can be implemented and used as an extension of the existing CAD/CAM systems.

Figure 9 represents an optimization system which has been created using the technique introduced in this paper. It is aimed to be used in conjunction with a

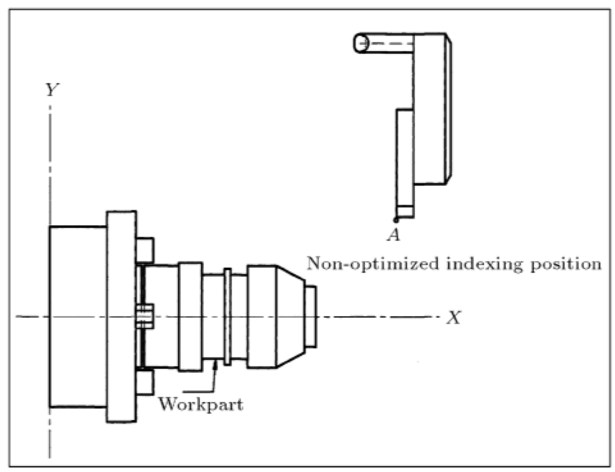


**Figure 7a.** Non-optimized indexing position for upper turret before moving to point $A$.

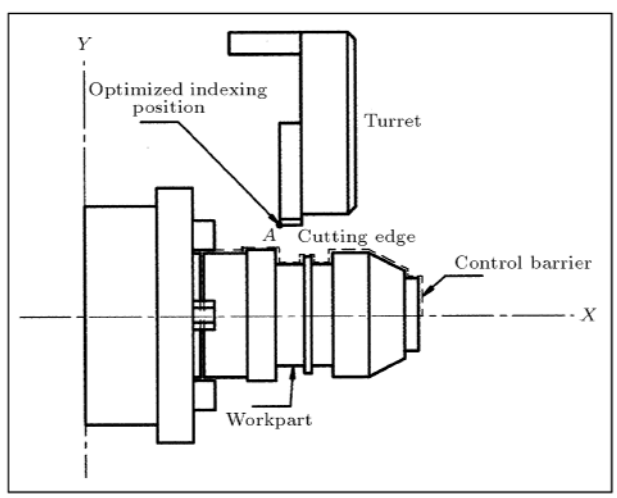


**Figure 7b.** Optimized indexing position for upper turret before moving to point A.

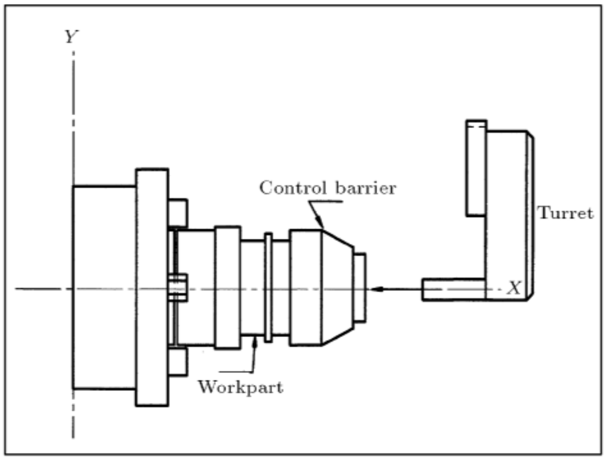


**Figure 8a.** Non-optimized position of tool tip prior to a cutting operation.

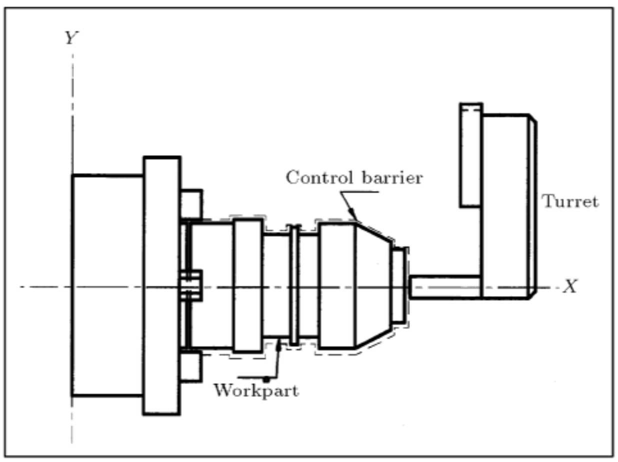


**Figure 8b.** Optimized position of tool tip prior to a cutting operation.

modern CAD/CAM system and as a separate module. This system offers functions such as:

1. Simulation of N.C. operations using an N.C. program received from a CAM system;
2. Automated work piece contour generation and update using the technique introduced in this paper;
3. Automated collision detection and avoidance;
4. Cycle time reduction, using the tool tip travel optimization techniques introduced.

## CONCLUSION

The algorithms described have been successfully programmed in 'C++'. The method is based on finding vertices corresponding to the control barrier, which would be the only feasible vertices able to form the segments of the optimum path. The algorithm is particularly designed to suit a CNC dynamic environment,

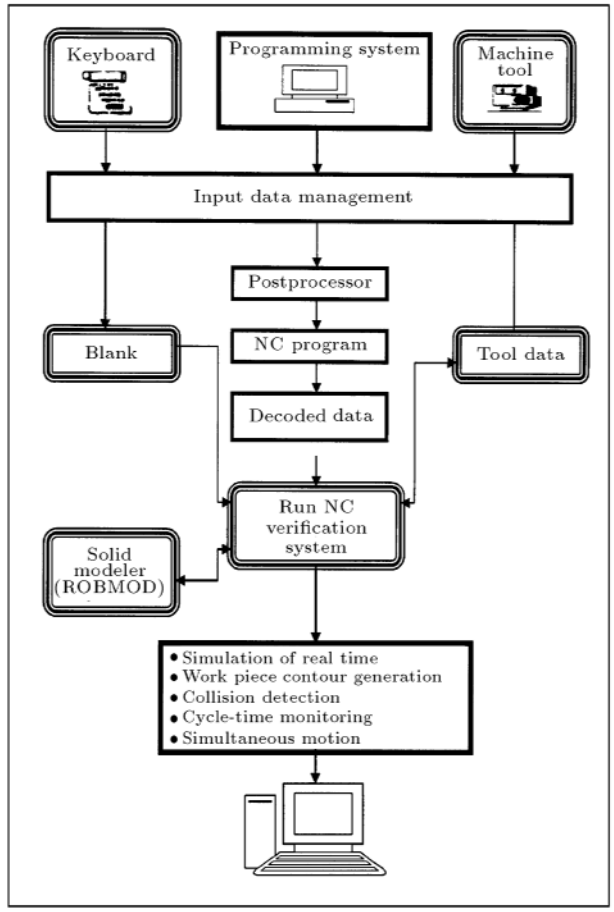


**Figure 9.** Data flow in the dynamic verification system implementing the algorithm.

but, at the same time, its generality has been preserved by presentation as an independent algorithm.

The advantage of this algorithm is that it can be easily developed, computerized and implemented for practical purposes. Also, the system supports the case of one barrier and one set of origins and destinations in the plane, $R^2$, as is the case in any machining operation.

It must be noted that the current and modern CAD/CAM systems have never implemented similar techniques, due to the complex dynamic environment presented by C.N.C. operations.

## REFERENCES

1. Kirk, D. and Lim, L. "A dual-mode routing algorithm for an autonomous roving vehicle", *IEEE Transactions on Aerospace and Electronic Systems*, **6**, pp 290-294 (1970).
2. Toh, C.K. "Cutter path strategies in high speed machining of hardened steel", *Materials & Design*, Article in Press (2004).
3. Shkel, A.M. and Lumelsky, V. "Robotics and autonomous systems classification of the Dubins set 1",

*Robotics and Autonomous Systems*, **34**(4), pp 179-202 (31 March 2001).

4. Vaccaro, H. "Alternative techniques for modeling travel distance", MA. Thesis in Civil Engineering, Massachusetts Institute of Technology, USA (1974).

5. Wang Dahl, G., Pollock, S. and Woodward, J. "Minimum trajectory pipe , J. "Minimum trajectory pipe routing", *Journal of Ship Research*, **18**, pp 46-49 (1994).

6. Larson, R. and Li, G. "Facility locations with the Manhattan metric", *The Presence of Barriers to Travel. Operations Research*, **31**, pp 652-669 (1983).

7. Viegas, J. "Capacity expansion in public facility networks", Doctoral Thesis in Civil Engineering, Institute Superior Tecnico, Lisbon, Portugal (1984).

8. Goldman, A.J. "A theorem on convex programming", Paper presented at *MAA Meeting*, Annapolis, MD, USA (1963).