Generation of Configuration Space Obstacles: The Case of Moving Algebraic Curves†

Chanderjit Bajaj and Myung-Soo Kim

Department of Computer Science,
Purdue University,
West Lafayette, IN 47907.

Abstract

We present algebraic algorithms to generate the boundary of planar configuration space obstacles arising from the translatory motion of objects amongst obstacles. Both the boundaries of the objects and obstacles are given by segments of algebraic curves.

1. Introduction

Using configuration space, (*C–Space*), to plan motion for a single rigid object amongst physical obstacles, reduces the problem to planning motion for a mathematical point amongst "grown" configuration space obstacles, (the points in *C–Space* which correspond to the object overlapping one or more obstacles). For example, a rigid polygonal object translating and rotating in the plane can be represented as a point moving in 3-Dimension *C–Space*, Lozano-Perez and Wesley (1979). The technique thus relies in efficiently generating the boundary of *C–Space obstacles*. Numerous applications such as part machining, mold design, and part assembly etc., also exist where the *C–space* approach proves useful, Adamowicz and Albano (1976), Freeman(1975), Lozano-Perez (1983), Tiller and Hanson (1984), Bajaj and Kim (1986). Nevertheless the only efficient algorithms known for generating *C–Space obstacles* have been for polygonal objects and obstacles; using methods for efficiently computing convex hulls for convex polygonal objects and obstacles, Lozano_Perez (1983), and recently efficient convolution algorithms for simple polygonal shapes, Guibas, Ramshaw and Stolfi (1983), Guibas and Seidel (1985). In this paper we characterize and generate the algebraic curve boundary of the *C–Space obstacles*, arising from the translatory motion of objects amongst obstacles whose boundaries are defined by segments of algebraic curves.

The main contributions of this paper are as follows. In § 3 we characterize the boundary of *C–Space obstacles* for general planar curved objects moving with only translation. They are related to the convolution of the boundaries of object and obstacle (*Convolution*) as well as a certain outer envelope of curves of the moving object (*O–Envelope*), generated by reversing the object with respect to a reference point and then allowing the reference point to move on the physical obstacle. Only for convex shaped objects and obstacles are the boundaries of the *C–space obstacle*, the *O–Envelope* and the *Convolution* exactly the same. The objects and obstacles that we consider have arbitrary shape with boundaries consisting of segments of algebraic curves and are represented by a boundary representation model discussed in § 2. In § 4 we give algebraic algorithms to generate the curve segments and vertices of the *Convolution* of the boundary of object and obstacle. Crucial too here is the internal representation of algebraic curves, i.e., whether they are parametrically or implicitly defined‡. We present algorithms for both these internal representations. In §5 we deal with curve singularities that arise in the *Convolution* and give methods by which to remove these singularities and thereby obtain the boundary of the *C–space obstacles*.

† Research supported in part by NSF grant DCI-85 21356.

2. Algebraic Boundary Model

In a general boundary representation, an object with algebraic boundary curves consists of a list of *peels*. An object may have internal holes and peels which correspond to them are termed "hole" peels. Each *peel* in turn consists of the following:

(1) A face bounded by a single oriented cycle of edges. (The area bounded by the cycle of edges is infinite for a "hole" peel and finite otherwise.)

(2) A finite set of directed edges, where each edge is incident to two vertices. Each edge also has a curve equation, represented either in implicit or in parametric form.

(3) A finite set of vertices usually specified by Cartesian coordinates.

The curve equation for each edge is chosen such that the direction of the normal at each point of the edge is towards the exterior of the object. For a simple point on the curve the normal is defined as the vector of partials to the curve evaluated at that point. For a singular point on the curve we associate a range of normal directions determined by normals to the tangents at the singular point. Further the edges are oriented such that the interior of the object is to the left when the cycle of edges is traversed. Straightforward assumptions are also made, e.g., edges are non-singular except at vertices, and edges are strictly convex (slopes of tangents are strictly increasing along the edge), strictly concave (slopes of tangents are strictly decreasing) or line segments. Such conditions are easily met by adding extra vertices to the boundary.

3. C-space Obstacles, Convolution and Envelopes

Let A be a moving object with its reference point at the origin and B be a fixed obstacle in the 2-dimensional real Euclidean plane R^2. Both A and B are modeled by the above boundary representations and assumed to be without 'holes'. Further non-regularities such as dangling edges and isolated vertices are also not permitted. The *C–space obstacles* that we construct are also regularized in this fashion and are modeled by the above

‡ A unit circle is implicitly given as $x^2+y^2-1=0$ and in parameteric form as $x=(1-t^2)/(1+t^2)$ and $y=2t/(1+t^2)$

boundary representation. For the sake of notation and preciseness in our usage we make the following distinctions. We denote $Int(A)$ as the interior of A and $Bdr(A)$ as the boundary of A. Note that $A = Int(A) \bigcup Bdr(A) = Cl(A)$ = closure of A by regularity. Further, the exterior of A is denoted by $Ext(A) = A^c$ (the complement of A) $= R^2 \sim A$, where the set difference $P \sim Q = \{p \in R^2 \mid p \in P \text{ and } p \notin Q\}$. Note that $Int(A)$ and $Ext(A)$ are open sets.

Throughout we consider object A to be free to move with fixed orientation. In this case configuration space is also 2-dimensional. We fix a reference point on A and denote A_p to be A located in R^2 with its reference point at the point $p \in R^2$. We also have $d(p,q)$ as the Euclidean distance between p and q; $NB_\varepsilon(p) = \{q \in R^2 \mid d(p,q) < \varepsilon\}$ = ε–neighborhood around a point p; $-A = \{-p \mid p \in A\}$ = Minkowski inverse, $A \pm B = \{p \pm q \mid p \in A \text{ and } q \in B\}$ = Minkowski sum and difference.

One also needs the following distinctions

(1) $A_{\bar{p}}$ is *free* from $B \iff A_{\bar{p}} \bigcap B = \varnothing$

(2) $A_{\bar{p}}$ *collides* with $B \iff Int(A_{\bar{p}}) \bigcap Int(B) \neq \varnothing$

(3) $A_{\bar{p}}$ *contacts* with $B \iff A_{\bar{p}} \bigcap B \neq \varnothing$ and $Int(A_{\bar{p}}) \bigcap Int(B) = \varnothing$ (Note that these conditions imply $Bdr(A_{\bar{p}}) \bigcap Bdr(B) \neq \varnothing$.)

(4) $CO(A,B) = C\text{-}space$ obstacle due to A and $B = \{\bar{p} \in R^2 \mid A_{\bar{p}} \bigcap B \neq \varnothing\}$.

(5) $O\text{-}Envelope(-A,B)$ = Outer envelope due to $-A$ and $B = \{\bar{p} \in R^2 \mid \bar{p} \in Bdr((-A)_p)$ for some $p \in Bdr(B)$, and $\bar{p} \notin Int((-A)_q)$ for any $q \in B\}$ (Having $q \in B$ as opposed to $q \in Bdr(B)$ implies that only the outer envelope is considered.)

(6) $Convolution(Bdr(-A), Bdr(B))$ = Convolution of $Bdr(-A)$ and $Bdr(B) = \{\bar{p} \in R^2 \mid \bar{p} = p - q$ where $p \in Bdr(B)$ and $q \in Bdr(A)$ and B has an outward normal direction at p exactly opposite to an outward normal A has at $q\}$

We now note the following.

Theorem 3.1 : $CO(A,B) = B - A$

Proof : Lozano-Perez and Wesley (1979). □

From the above Theorem and our prior definitions we obtain,

Corollary 3.2 : (1) $CO(Int(A), Int(B)) = Int(B) - Int(A) = B - Int(A)$ (This is an open set)
(2) $A_{\bar{p}}$ is *free* from $B \iff \bar{p} \in Ext(CO(Int(A), Int(B))) = Ext(Int(B) - Int(A))$
(3) $A_{\bar{p}}$ *collides* with $B \iff \bar{p} \in Int(CO(Int(A), Int(B))) = Int(B) - Int(A)$
(4) $A_{\bar{p}}$ *contacts* with $B \iff \bar{p} \in Bdr(CO(Int(A), Int(B))) = Bdr(Int(B) - Int(A))$

We next obtain the following important characterizations,

Theorem 3.3 : $Bdr(CO(Int(A), Int(B))) = O\text{-}Envelope(-A,B)$

Proof : ($\subseteq$) : Let $\bar{p} \in Bdr(CO(Int(A), Int(B)))$, then $A_{\bar{p}}$ *contacts* with B, (*Corollary* 3.2 (4)), and $\exists\, p \in Bdr(A_{\bar{p}}) \bigcap Bdr(B)$. Since $p - \bar{p} \in Bdr(A)$, we have $\bar{p} - p \in Bdr(-A)$ and $\bar{p} \in Bdr((-A)_p)$ for $p \in Bdr(B)$. Further $\bar{p} \notin Int((-A)_q)$ for any $q \in B$. Assuming the contrary, if $\bar{p} \in Int((-A)_q)$ for some $q \in B$, then $\bar{p} \in B - Int(A) = Int(B) - Int(A) = Int(CO(Int(A), Int(B)))$, (contradiction).
($\supseteq$) : Let $\bar{p} \in O\text{-}Envelope(-A,B)$, then $\bar{p} \in Bdr((-A)_p)$ for some $p \in Bdr(B)$, and $\bar{p} \notin Int((-A)_q)$ for any $q \in B$. Equivalently, $p \in Bdr(A_{\bar{p}}) \bigcap Bdr(B)$ and $q \notin Int(A_{\bar{p}})$ for any $q \in B$. This implies $A_{\bar{p}} \bigcap B \neq \varnothing$ and $Int(A_{\bar{p}}) \bigcap Int(B) = \varnothing$. Hence, $A_{\bar{p}}$ contacts with B. □

Theorem 3.4 : $Bdr(CO(A,B)) \subset O\text{-}Envelope(-A,B) \subset Convolution(Bdr(-A), Bdr(B))$

Proof : (1) Using Theorem 3.3 we show $Bdr(CO(A,B)) \subset Bdr(CO(Int(A), Int(B)))$: For any $\bar{p} \in CO(A,B)$, $A_{\bar{p}} \bigcap B \neq \varnothing$, equivalently $\bar{p} \in Cl(CO(Int(A), Int(B)))$, (*Corollary* 3.2 (2)). Hence, $CO(Int(A), Int(B)) \subset CO(A,B) \subset Cl(CO(Int(A), Int(B)))$ and $Cl(CO(A,B)) = Cl(CO(Int(A), Int(B)))$. Since $Int(CO(Int(A), Int(B))) \subset Int(CO(A,B))$, we have $Bdr(CO(A,B)) \subset Bdr(CO(Int(A), Int(B)))$.
(2) $O\text{-}Envelope(-A,B) \subset Convolution(Bdr(-A), Bdr(B))$: For any $\bar{p} \in O\text{-}Envelope(-A,B) = Bdr(CO(Int(A), Int(B)))$, since $A_{\bar{p}}$ contacts with B at some $p \in Bdr(B)$, $A_{\bar{p}}$ has an outward normal direction at p which is opposite to an outward normal direction B has at p. For $q = p - \bar{p} \in Bdr(A)$, we have $\bar{p} = p - q$ and B has an outward normal direction at p exactly opposite to an outward normal A has at q. Thus $\bar{p} \in Convolution(Bdr(-A), Bdr(B))$. Also see Guibas, Ramshaw, and Stolfi (1983). □

The differences between the above entities are various kinds of redundant vertices and curve segments. There are primarily five classes of redundancies or singularities that may arise, (a) dangling edges (b) isolated vertices (c) coincident edges (d) intersecting edges and (e) self-intersecting edges, (see Figure 1 (a)–(e)). In Figure 1–(a) there are four dangling edges (dotted lines). These dangling edges are in the *Convolution*, but not in the *O–Envelope* or in the *C–space* obstacle boundary (bold lines). The moving object is a small square. In Figure 1 (b)–(e) the moving object is a circle. In Figure 1–(b) there is an isolated vertex which is in the *Convolution* and in the *O–Envelope*, but not in the *C–space* obstacle boundary. In Figure 1–(c) there is a coincident edge which is in the *Convolution* and in the *O–Envelope*, but not in the *C–space* obstacle boundary. In Figure 1–(d) there is an intersection point of two edges. At this intersection point the configurations of $A_{\bar{p}}$ and B change from *contact* to *collide* or from *collide* to *contact*. Parts of both these intersecting edges (dotted) are in the *Convolution* but in the *C–space* obstacle boundary. In Figure 1–(e) there is a self-intersecting edge. The triangular loop resulting from this self-intersection is in the *Convolution*, but not in the *O–Envelope*. The redundancies (b)–(d) are closely related with the intersections and self-intersections of edges. An isolated vertex may result from a degenerate self-intersecting edge and a coincident edge may result when two edges intersect over a curve segment not just at a common point. The boundary representation of the *C–space* obstacle we construct has no singularities except at its vertices. The differences between the *Convolution*, the *O–Envelope*, and the *C–space* obstacle boundary of Theorem 3.4 are made more precise in Theorem 3.6 below. But before that we consider an important special case.

In the special case when both A and B are convex, both the set containments of Theorem 3.4 become equalities. This follows from the properties of convexity. In particular we use the following simple fact. For convex A and B, if $A_{\bar{p}}$ and B have opposite outward normal directions at $p \in Bdr(A_{\bar{p}}) \bigcap Bdr(B)$, then there is a common supporting line L_p such that $A_{\bar{p}}$ and B are on opposite sides of the line L_p, Kelly and Weiss (1979).

Theorem 3.5 : For convex A and B, we have $Bdr(CO(A,B)) = O\text{-}Envelope(-A,B) = Convolution(Bdr(-A), Bdr(B))$

Proof : Using Theorem 3.4, all we need to show is $Convolution(Bdr(-A), Bdr(B)) \subset Bdr(CO(A,B))$ for convex A and B. Suppose $\bar{p} \in Convolution(Bdr(-A), Bdr(B))$. We first show $\bar{p} \notin Ext(CO(A,B))$. If $\bar{p} \in Ext(CO(A,B))$, then $\exists\, \varepsilon > 0$ such that

$(A_{\bar p}+NB_\varepsilon(0)) \cap B = \emptyset$ and $Cl(A_{\bar p}) \cap Cl(B) = \emptyset$. Hence, $\bar p \notin Bdr((-A)_p)$ for any $p \in Bdr(B)$, (contradiction), and so $\bar p \notin Ext(CO(A,B))$. Now, we show $\bar p \notin Int(CO(A,B))$. Since $\exists\, p \in Bdr(A_{\bar p}) \cap Bdr(B)$ such that $A_{\bar p}$ and B have opposite outward normal directions at p, a common supporting line L_p separates $A_{\bar p}$ and B. For any $\varepsilon > 0$, let e be an outward normal vector to B at p such that $\| e \| = \varepsilon$ and e is orthogonal to L_p, then $A_{(\bar p+e)}$ and B are separated by the banded region bounded by $L_{(\bar p+e)}$ and L_p, and so $A_{(\bar p+e)} \cap B = \emptyset$. Hence, $\bar p \notin Int(CO(A,B))$. Thus $\bar p \notin Int(CO(A,B)) \cup Ext(CO(A,B))$ implies $\bar p \in Bdr(CO(A,B))$. □

This then leads to the following observations. For convex A and B there exist no singularities in the boundary of the *Convolution*, *O−Envelope* or *C−space* obstacles (except possibly at the vertices). It also suggests a natural method for handling *non−convex* object and obstacle shapes. One first obtains a convex decomposition consisting of union of convex pieces and then generates the *C−space* obstacle as the union of *C−space* obstacles for convex object and obstacle pairs. Such convex decompositions are possibly for planar polygonal objects, see Chazelle (1980). However not all objects with algebraic curve boundaries permit decompositions consisting of the union of convex pieces – for example an object with an inward circular (concave) arc on its boundary. Hence alternate methods of dealing with non-convex objects become important. The method we suggest here deals with non-convex objects directly.

For general objects and obstacles with algebraic curve boundaries we first generate the $Convolution(Bdr(-A),Bdr(B))$ complete with redundancies. Then on systematically removing redundancies one obtains the boundary of the *C−space* obstacle $Bdr(CO(A,B))$. The next Theorem helps characterize this procedure.

Theorem 3.6 : (1) $(Convolution(Bdr(-A),Bdr(B))) \sim (O\text{-}Envelope(-A,B))$ is the set of all the vertices and edge segments of $Convolution(Bdr(-A),Bdr(B))$ such that for points $\bar p$ on these vertices and edge segments $A_{\bar p}$ *collides* with B.
(2) $(O\text{-}Envelope(-A,B)) \sim (Bdr(CO(A,B)))$ is the set of all the isolated vertices and coincident curve segments of $O\text{-}Envelope(-A,B)$.

Proof : (1) We first prove $Convolution(Bdr(-A),Bdr(B)) \subset Cl(CO(Int(A),Int(B)))$. For any $\bar p \in Convolution(Bdr(-A),Bdr(B))$, $\bar p = p+q$ for some $p \in Bdr(B)$ and $q \in Bdr(-A)$. Then $p = \bar p - q \in Bdr(A_{\bar p})$ and $Bdr(A_{\bar p}) \cap Bdr(B) \neq \emptyset$. This implies $A_{\bar p} \cap B \neq \emptyset$, ($A$ and B are regular sets). Equivalently $A_{\bar p}$ is not *free* from B and $\bar p \in Cl(CO(Int(A),Int(B)))$, (*Corollary* 3.2). Thus $Convolution(Bdr(-A),Bdr(B)) \subset Cl(CO(Int(A),Int(B)))$. Next the equality of *Theorem* 3.3 implies that $[Convolution(Bdr(-A),Bdr(B)) \sim O\text{-}Envelope(-A,B)] \subset [Cl(CO(Int(A),Int(B))) \sim Bdr(CO(Int(A),Int(B)))] = Int(CO(Int(A),Int(B)))$. Thus the assertion follows from Corollary 3.2(3).
(2) Having no intersections, self-intersections or *collide* edges and vertices, $O\text{-}Envelope(-A,B) = Bdr(CO(Int(A),Int(B)))$ can only have (a) isolated vertices (b) coincident edges and (c) edges and vertices separating $Int(CO(Int(A),Int(B)))$ from $Ext(CO(Int(A),Int(B)))$. Types (a) and (b) arise from the absence of the boundaries of A and B in $CO(Int(A),Int(B))$. Isolated vertices arise from point "holes" of $Int(CO(Int(A),Int(B)))$. Coincident edges arise from two overlapping outer-envelope curves grown from two different edges of $Bdr(B)$ and lie between two regions of $Int(CO(Int(A),Int(B)))$. Types (a) and (b) are all in $CO(A,B)$ and also these are surrounded by $Int(CO(A,B))$. This implies that the vertices and edges of types (a) and (b) are in $Int(CO(A,B))$, but not in $Bdr(CO(A,B))$. Since $Ext(CO(Int(A),Int(B))) = Ext(CO(A,B))$, Types (c) are adjacent to $Ext(CO(A,B))$ and hence these are in $Bdr(CO(A,B))$. Thus $O\text{-}Envelope(-A,B) \sim Bdr(CO(A,B))$ consists of all the isolated vertices and coincident curve segments of $O\text{-}Envelope(-A,B)$, and nothing more. □

4. Generating Convolution of Object and Obstacles

Let $S \subset R^2$ be a set, $p \in Bdr(S)$ be a boundary point, and $C \subset Bdr(S)$ be a boundary curve segment. Then denote $N(S,p)$ to be the set of all unit outward normal direction vectors of S at p, and $N(S,C) = \bigcup_{p \in C} N(S,p)$. Note, for a singular vertex on the boundary we associate a range of outward normal directions determined by outward normals to the tangents at the singular vertex. For a given point $p \in Bdr(B)$, let the set of points $Ch(p)$ = the *characteristic set* of $p = \{ \bar p = p+q \mid q \in Bdr(-A)$ and $N(B,p) \cap N(-A,q) \neq \emptyset \}$. For a boundary curve segment $C \subset Bdr(B)$, the set of points $Ch(C) = \bigcup_{p \in C} Ch(p)$ is called the *characteristic set* of C. One can easily show that $Convolution(Bdr(-A),Bdr(B)) = (\bigcup_{C \in \Gamma_1} Ch(C)) \cup (\bigcup_{p \in \Gamma_2} Ch(p))$, where Γ_1 is the set of all boundary edges of $Bdr(B)$ and Γ_2 is the set of all boundary vertices of $Bdr(B)$.

Fortunately not all edge-edge, edge-vertex and vertex-vertex pairs between A and B contribute to the $Convolution(Bdr(-A),Bdr(B))$. Let $C_B \subset Bdr(B)$ and $C_A \subset Bdr(-A)$ be boundary edge segments, (p,N_p) be a pair such that $p \in Bdr(B)$ and $N_p \subset N(B,p)$. Further let (q,N_q) be a pair such that $q \in Bdr(-A)$ and $N_q \subset N(-A,q)$. Then we define *compatible* vertex and edge segment pairs between A and B which contribute to the *Convolution* as follows.

(1) C_B and C_A are *compatible* $\iff N(B,C_B) = N(-A,C_A)$

(2) C_B and (q,N_q) are *compatible* $\iff N(B,C_B) = N_q$

(3) (p,N_p) and C_A are *compatible* $\iff N_p = N(-A,C_A)$

(4) (p,N_p) and (q,N_q) are *compatible* $\iff N_p = N_q$

Straightforwardly from definitions we then note the following for the *Convolution* of the above *compatible* vertex and edge segment pairs,

Remark:

(a) $Convolution(C_A,C_B) = \{ \bar p = p+q \mid p \in C_B$ and $q \in C_A$, and $N(B,p) \cap N(-A,q) \neq \emptyset \}$

(b) $Convolution((q,N_q),C_B) = \{ \bar p = p+q \mid p \in C_B$ and $N(B,p) \cap N_q \neq \emptyset \} = C_B + \{q\}$

(c) $Convolution(C_A,(p,N_p)) = \{ \bar p = p+q \mid q \in C_A$ and $N_p \cap N(-A,q) \neq \emptyset \} = \{p\} + C_A$

(d) $Convolution((q,N_q),(p,N_p)) = \{ \bar p = p+q \mid N_p \cap N_q \neq \emptyset \} = \{p+q\}$

We now show how to generate $Ch(C)$ in § 4.1 and to generate $Ch(p)$ in § 4.2 for compatible vertex and edge segment pairs between A and B. The computation of compatible pairs is discussed in § 4.3 . Note that in the following subsections we assume that each of the boundary edges of A and B are strictly convex, strictly concave, or line segments as specified in § 2.

4.1. Growing edges — Generating segments of $Ch(C)$

For a given boundary edge $C \subset Bdr(B)$, suppose that $C_i \subset Bdr(-A)$ ($i = 1, ..., m$) are all the boundary edges such that $N_i = N(B,C) \cap N(-A,C_i) \neq \emptyset$. Also let $q_j \in Bdr(-A)$ ($j = 1, ..., n$) be all the boundary vertices with $N_{q_j} = N(B, C) \cap N(-A, q_j) \neq \emptyset$. Further let $C'_i \subset C$ and $\bar{C}_i \subset C_i$ be the *compatible* edge segments of type (1) with $N(B, C'_i) = N(-A, \bar{C}_i) = N_i$. Also let $C''_j \subset C$ be the *compatible* edge segments of type (2) such that $N(B, C''_j) = N_{q_j}$. One can easily show that $Ch(C) = (\bigcup_i Convolution(\bar{C}_i, C'_i)) \cup (\bigcup_j Convolution((q_j, N_{q_j}), C''_j))$. One can use Theorems 4.1–4.4 to compute $Convolution(\bar{C}_i, C'_i)$ while directly computing $Convolution((q_j, N_{q_j}), C''_j) = C''_j + \{q_j\}$ as simply translated edge segments.

Theorem 4.1 : Let $C_B \subset Bdr(B)$ be a segment of an algebraic boundary curve segment $f = 0$ with outward normal directions ∇f. Further let $C_A \subset Bdr(-A)$ be a segment of an algebraic boundary curve segment $g = 0$ with outward normal directions ∇g, and suppose that C_B and C_A are compatible. Then $Convolution(C_A, C_B)$ is the set of points $\bar{p} = (\bar{x}, \bar{y}) = p + q = (x + \alpha, y + \beta)$ such that

$$\begin{cases} f(x,y)=0 \text{ and } p=(x,y)\in C_B & (1) \\ g(\alpha,\beta)=0 \text{ and } q=(\alpha,\beta)\in C_A & (2) \\ f_x \cdot g_\beta - f_y \cdot g_\alpha = 0 & (3) \\ f_x \cdot g_\alpha + f_y \cdot g_\beta > 0 & (4) \end{cases}$$

Proof : Suppose $p = (x, y)$ and $q = (\alpha, \beta)$ satisfy (1)–(4). (1) and (2) implies that $p \in Bdr(B)$ and $q \in Bdr(-A)$, and (3) and (4) implies that the outward normal direction of B at p is the same as that of $-A$ at q. Hence, $\bar{p} = p + q \in Ch(p) \subset Ch(C_B)$ □

We use Theorem 4.1 as follows. First substitute $x = \bar{x} - \alpha$ and $y = \bar{y} - \beta$ in the above equations (1) and (3). Then one can obtain the implicit algebraic equation of the $Convolution(C_A, C_B)$ in terms of $\bar{x}, \bar{y}$ by eliminating α and β from the equations (1), (2) and (3). Elimination of variables can be performed by computing resultants on pairs of equations, see Collins (1971). For certain special forms of polynomials the generalized method of Sylvester as proposed by Dixon (1908) may be used for simultaneous elimination of two variables from three equations.

The proofs for the Theorems 4.2–4.4 below are essentially the same as that of Theorem 4.1.

Theorem 4.2 : Let $C_B \subset Bdr(B)$ be a segment of an algebraic boundary curve segment $f = 0$ with outward normal directions ∇f. Further let $C_A \subset Bdr(-A)$ be a segment of a parametric boundary curve segment $C(t) = (c_1(t), c_2(t))$ with outward normal directions $(c'_2(t), -c'_1(t))$, and suppose that C_B and C_A are compatible. Then $Convolution(C_A, C_B)$ is the set of points $\bar{p} = (\bar{x}, \bar{y}) = p + q = (x + c_1(t), y + c_2(t))$ such that

$$\begin{cases} f(x,y)=0 \text{ and } p=(x,y)\in C_B & (1) \\ q=(c_1(t),c_2(t))\in C_A & (2) \\ f_x \cdot c'_1(t) + f_y \cdot c'_2(t) = 0 & (3) \\ f_x \cdot c'_2(t) - f_y \cdot c'_1(t) > 0 & (4) \end{cases}$$

First substitute $x = \bar{x} - c_1(t)$ and $y = \bar{y} - c_2(t)$ in the above equations (1) and (3). Then one can obtain the implicit algebraic equation of the $Convolution(C_A, C_B)$ in terms of $\bar{x}, \bar{y}$ by eliminating t from the equations (1) and (3) by computing resultants.

Theorem 4.3 : Let $C_B \subset Bdr(B)$ be a segment of a parametric boundary curve segment $C(t) = (c_1(t), c_2(t))$ with outward normal directions $(c'_2(t), -c'_1(t))$. Further let $C_A \subset Bdr(-A)$ be a segment of an algebraic boundary curve segment $g = 0$ with outward normal directions ∇g, and suppose that C_B and C_A are compatible. Then $Convolution(C_A, C_B)$ is the set of points $\bar{p} = (\bar{x}, \bar{y}) = p + q = (c_1(t) + \alpha, c_2(t) + \beta)$ such that

$$\begin{cases} p=(c_1(t),c_2(t))\in C_A & (1) \\ g(\alpha,\beta)=0 \text{ and } q=(\alpha,\beta)\in C_A & (2) \\ c'_1(t) \cdot g_\alpha + c'_2(t) \cdot g_\beta = 0 & (3) \\ c'_2(t) \cdot g_\alpha - c'_1(t) \cdot g_\beta > 0 & (4) \end{cases}$$

First substitute $\alpha = \bar{x} - c_1(t)$ and $\beta = \bar{y} - c_2(t)$ in the above equations (2) and (3). Then one can obtain the implicit algebraic equation of the $Convolution(C_A, C_B)$ in terms of $\bar{x}, \bar{y}$ by eliminating t from the equations (2) and (3) by computing resultants.

Theorem 4.4 : Let $C_B \subset Bdr(B)$ be a segment of a parametric boundary curve segment $C(s) = (c_1(s), c_2(s))$ with outward normal directions $(c'_2(s), -c'_1(s))$. Further let $C_A \subset Bdr(A)$ be a segment of a parametric boundary curve segment $\bar{C}(t) = (\bar{c}_1(t), \bar{c}_2(t))$ with outward normal directions $(\bar{c}'_2(t), -\bar{c}'_1(t))$, and suppose that C_B and C_A are compatible. Then $Convolution(C_A, C_B)$ is the set of points $\bar{p} = (\bar{x}, \bar{y}) = p + q = (c_1(s) + \bar{c}_1(t), c_2(s) + \bar{c}_2(t))$ such that

$$\begin{cases} c'_1(s) \cdot \bar{c}'_2(t) - c'_2(s) \cdot \bar{c}'_1(t) = 0 & (1) \\ c'_1(s) \cdot \bar{c}'_1(t) + c'_2(s) \cdot \bar{c}'_2(t) > 0 & (2) \end{cases}$$

One can obtain the implicit algebraic equation of the $Convolution(C_A, C_B)$ by eliminating s and t from the equations $\bar{x} = c_1(s) + \bar{c}_1(t)$, $\bar{y} = c_2(s) + \bar{c}_2(t)$ and the above equation (1). Elimination of both variables can be performed by computing resultants on pairs of equations, or at times simultaneous elimination of two variables from three equations.

In the above Theorems we considered both the implicit and rational parametric internal representation of curves segments since not all algebraic curves have both representations, see Walker (1978). For the class of *rational* algebraic curves which have a rational parametric form, algebraic algorithms also exist for converting between the two representations. However their efficiency are limited to curves of low degree, see Abhyankar and Bajaj (1986a, b).

4.2. Growing vertices — Generating segments and vertices of $Ch(p)$

For a given boundary vertex $p \in Bdr(B)$, suppose that $C_i \subset Bdr(-A)$ $(i = 1, ..., m)$ are all the boundary edges such that $N_i = N(B, p) \cap N(-A, C_i) \neq \emptyset$, and $q_j \in Bdr(-A)$ $(j = 1, ..., n)$ are all the boundary vertices of type (4) with $N_{q_j} = N(B, p) \cap N(-A, q_j) \neq \emptyset$. Further let $\bar{C}_i \subset C_i$ be the *compatible* edge segments of type (3) with $N(-A, \bar{C}_i) = N_i$. One can easily show that $Ch(p) = (\bigcup_i Convolution(\bar{C}_i, (p, N_i))) \cup (\bigcup_j Convolution((q_j, N_{q_j}), (p, N_{q_j})))$. Since one has $Convolution(\bar{C}_i, (p, N_i)) = \{p\} + \bar{C}_i$ and $Convolution((q_j, N_{q_j}), (p, N_{q_j})) = \{p + q_j\}$, computing $Ch(p)$ is easy.

4.3. Obtaining $Convolution(Bdr(-A), Bdr(B))$

We first show how to obtain *compatible* vertex and edge segment pairs for which to generate the *Convolution*. Let $C_i \subset Bdr(B)$ and $p_i \in Bdr(B)$ $(i = 1, ..., m)$ be all the boundary edges and vertices of B. Also let $C'_j \subset Bdr(-A)$ and $q_j \in Bdr(-A)$ $(j = 1, ..., n)$ be all the boundary edges and vertices of $-A$. By adding more vertices if necessary, we can make each extreme angle of $N(B, C_i)$, $N(B, p_i)$, $N(-A, C'_j)$, and $N(-A, q_j)$, not to be an interior angle of any of these intervals, see Figure 2 (a)–(c). Let I_k $(k = 1, ..., l)$ be a sorted sequence of all disjoint angle intervals. Note that I_k is a single point interval for a linear edge. Take $C_i \subset Bdr(B)$ $(i = 1, ..., m_1)$, $p_{i'} \in Bdr(B)$ $(i' = 1, ..., m_2)$, $C'_j \subset Bdr(-A)$ $(j = 1, ..., n_1)$, and $q_{j'} \in$

$Bdr(-A)$ $(j' = 1, ..., n_2)$, to be all the boundary edges and vertices such that $I_1 = N(B, C_i) = N(B, p_{i'}) = N(-A, C'_j) = N(-A, q_{j'})$. Then, there are $m_1 \cdot n_1$ edge–edge convolutions $Convolution(C_i, C'_j)$, $m_1 \cdot n_2$ edge–vertex convolutions $C_i + \{q_{j'}\}$, $m_2 \cdot n_1$ vertex–edge convolutions $\{p_{i'}\} + C'_j$, and $m_2 \cdot n_2$ vertex–vertex convolutions $\{p_{i'} + q_{j'}\}$. After generating all these convolutions for I_1, we continue the same procedure for I_2, and so on. In Figure 2–(c), I_1 has 9 edge-edge convolutions, I_2 has 6 edge-edge and 3 vertex-edge convolutions, I_3 has one edge-vertex and 2 edge-edge convolutions, I_5 has only one line-vertex convolution, I_7 has 3 vertex-line convolutions, and so on. I_5 and I_7 are single point intervals.

Having obtained all the vertices and curve segments of the $Convolution(Bdr(-A), Bdr(B))$, the next step is to connect these together with the correct topology. The topology of $Bdr(B)$ essentially induces a similar relationship between edges and vertices of $Convolution(Bdr(-A), Bdr(B))$. However to build the *Convolution* graph correctly, we also need to check for intersections and self-intersections of convolution edges. If two edges intersect or an edge has a self-intersection, a new vertex is created for the intersection point, and the new edges are connected with appropriate adjacencies. Intersecting edges $f = 0$ and $g = 0$ can be detected by either numerically solving $f = g = 0$ or algebraically via resultants. Alternatively for low degree curves one of f or g may be parameterized and the intersection computed by solving for the real parameter roots of the intersection, Abhyankar and Bajaj (1986 a, b). Self intersections and singularities on curves can be computed algebraically by simultaneously solving $f = f_x = f_y = 0$ where f_x and f_y are the x and y partials. Alternatively, singular points can be obtained by numerically traversing the edges of the *Convolution* graph and computing points where both f_x and f_y disappears. Coincident edges are special cases of intersecting edges and are merged into a single edge. Convolution edges which collapse into single vertices are special cases of self-intersecting edges. The elimination of redundant and singular edges and vertices is discussed in § 5. While generating the *Convolution* graph we tag each edge and vertex resulting from the above singularities appropriately.

5. Generation of C-space obstacles

Removing from the *Convolution* graph the vertices and edges on which the configurations of $A_{\bar{p}}$ and B are colliding, we obtain the *O–Envelope*, (*Theorem* 3.6 (1)). Further removing various isolated vertices and coincident edges from the *O–Envelope* one obtains the boundary of the *C–space* obstacle $Bdr(CO(A, B))$, (*Theorem* 3.6 (2)). The process of obtaining the boundary of the *C–space* obstacle is however more direct. The $Bdr(CO(A, B))$ we construct conforms to the boundary representation model of 2. We note that even if A and B are object models with single *peels*, $Bdr(CO(A, B))$ may consist of more than one *peel*, corresponding to "holes" in the *C–space* obstacle.

In generating the boundary of the *C–space* obstacle, first the *Convolution* graph is constructed as specified in §4 above. The intersecting and self-intersecting curve segments are broken up into edges with additions of new vertices at the intersecting or singular points. Next, a cleanup phase is initiated where redundancies such as isolated vertices, coincident edges and dangling edges are eliminated. These are either part of the *Convolution* or the *O–Envelope* or are formed while constructing the *Convolution* graph. For example in the *Convolution* graph construction phase, coincident edges of the *O–Envelope* redundant to the *C–space* obstacle boundary may get merged into single dangling edges. The final cleanup step in the $Bdr(CO(A, B))$ generation is then to eliminate the redundant edges and vertices which give rise to colliding configurations.

From the way the *Convolution* graph is constructed one can see that the relative configurations of $A_{\bar{p}}$ and B is constant on each edge (i.e., either *collide* or *contact* for each point $\bar{p}$ of the edge). Thus each convolution edge can be classified as either a *collide* edge or as a *contact* edge. To eliminate the redundant *collide* edges a vertex-by-vertex analysis needs to be done. First consider a vertex with only two adjacent edges. In this case, one can easily see that the edge types in the *Convolution* graph are either *collide–collide* or *contact–contact*. In the *collide–collide* case the common vertex is redundant, and in the *contact–contact* case the common vertex is in $Bdr(CO(A, B))$. Further one notes that as you follow a path of edges and vertices on which each vertex has only two adjacent edges, the whole path is either totally redundant or totally in $Bdr(CO(A, B))$. Thus one can classify the *simple* paths which have no more than two branches except at both end points, as redundant or non-redundant. In summary, a depth-first search on the *Convolution* graph with a vertex-by-vertex analysis as above allows one to delete all the redundant simple paths.

For vertices with more than two adjacent edges the problem of deciding redundant paths is slightly more complicated. Such high valence vertices arise either as a complex intersection of many curve segments or because of a self intersection singularity. One needs a distinct point on each of the various edges incident to the vertex for then a decision can be made as to whether the entire edge is either redundant or non-redundant. Generating distinct points on various branches of an algebraic curve emanating from a singularity can be quite difficult for high order and irregular singularities, see Walker (1978). However a local analysis about such high valence vertex points which yields distinct points on separate branches is always possible, Abhyankar (1983). Note that though this analysis applies to a singularity of a single algebraic curve with multiple branches at the singularity, for our purposes it can also be applied to the multiple edges arising from distinct intersecting curve segments. In this case one simply considers the product of all the distinct curve segments locally about the vertex point. By generating distinct points on the various edges incident on a high valence vertex, and checking for *collide* configurations of $A_{\bar{p}}$ and B, the redundant edges can be detected and removed.

6. Acknowledgement

We would like to thank Kyungho Oh for useful suggestions on envelopes and an anonymous referee for suggestions which helped improve the presentation of this paper.

7. References

Abhyankar, S. S., 1983
Desingularization of Plane Curves, *Proc. of the Symp. in Pure Mathematics*, 40, 1, 1-45.

Abhyankar, S., and Bajaj, C., (1986a)
Automatic Rational Parameterization of Curves and Surfaces I: Conics and Conicoids, *Computer Aided Design*, (to appear).

Abhyankar, S., and Bajaj, C., (1986b)
Automatic Rational Parameterization of Curves and Surfaces II: Cubics and Cubicoids, Computer Science Technical Report CSD-TR-592, Purdue University.

Adamowicz, M., and Albano, A., (1976)
Nesting two-dimensional shapes in rectangular modules, *Computer Aided Design*, 2, 1, 27-33.

Bajaj, C., and Kim, M., (1986)
Generation of Configuration Space Obstacles II: The case of Moving Algebraic Surfaces, Computer Science Technical Report CSD-TR-586, Purdue University.

Chazelle, B., (1980)
Computational Geometry and Convexity, Carnegie-Mellon Tech. Report, CMU-CS-80-150.

Collins, G., (1971)
The Calculation of Multivariate Polynomial Resultants, *Journal of the ACM*, 18, 4, 515-532.

Dixon, A., (1908)
The Eliminant of Three Quantics in Two Independent Variables, *Proc. London Mathematical Society*, 2, 6, 468-478.

Freeman, H., (1975)
On the packing of arbitrary shaped templates, *Proc. 2nd USA-Japan Computer Conference*, 102-107.

Guibas, L., Ramshaw, L., and Stolfi, J., (1983)
A Kinetic Framework for Computational Geometry, Proc. 24th Annual Symp. on Foundations of Computer Science, 100-111.

Guibas, L., and Seidel, R., (1986)
Computing Convolutions by Reciprocal Search, *Proc. of 2nd ACM Symposium on Computational Geometry*, 90-99.

Kelly, P., and Weiss, M., (1979)
Geometry and Convexity, John Wiley & Sons, New York.

Lozano-Perez, T., and Wesley, M.A., (1979)
An algorithm for planning collision free paths among polyhedral obstacles, *Communications of the ACM*, 22, 560-570.

Lozano-Perez, T., (1983)
Spatial Planning: A Configuration Space Approach, *IEEE Trans. on Computers*, v.C-32, 108-120.

Tiller, W., and Hanson, E., (1984)
Offsets of Two-Dimensional Profile, *IEEE Computer Graphics & Applications*, Sept., 36-46.

Walker, R., (1978)
Algebraic Curves, Springer Verlag, New York.

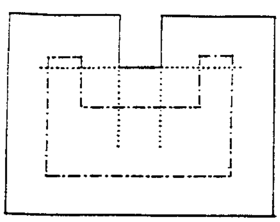

Figure 1–(a) dangling edges

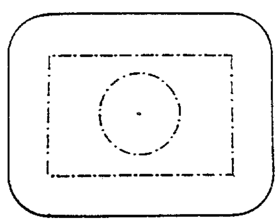

Figure 1–(b) isolated vertex

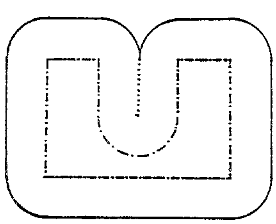

Figure 1–(c) coincident edge

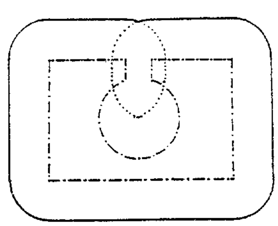

Figure 1–(d) intersection of edges

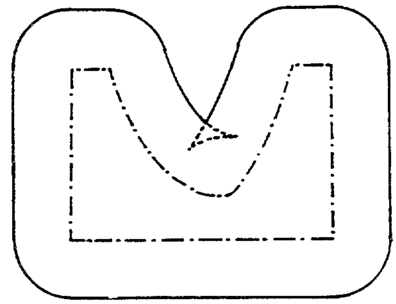

Figure 1–(e) self-intersection of edges

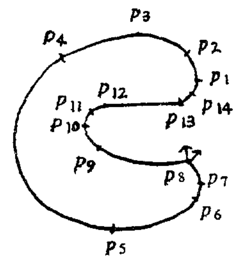

Figure 2–(a) obstacle

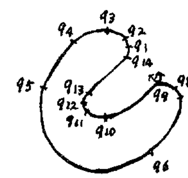

Figure 2–(b) reversed object

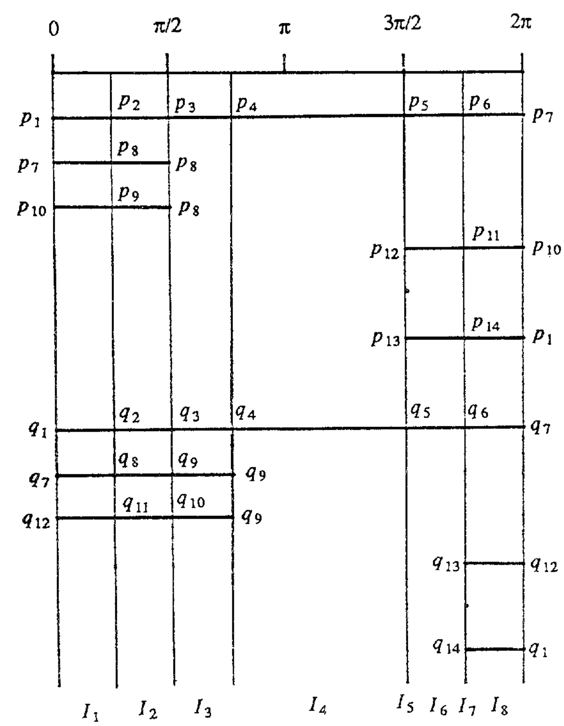

Figure 2–(c) compatible edges and vertices