On the Size of a Generalized Voronoi Diagram for Convex Polyhedra in d-dimensions[1]

Abhi Dattasharma
Department of Computer Science and Automation
Indian Institute of Science, Bangalore 560 012, India
e-mail: abhi@chanakya.csa.iisc.ernet.in

1. Introduction

Voronoi diagram is a well known tool in Computational Geometry and it has extensive applications in several areas. Classical Voronoi diagram was defined for finite point sets in two dimensions Extensions of this diagram to higher dimensions and non-point sets exist, but all such extensions deal with either point sets in higher dimensions or relatively simple objects; and it appears that the size of such a generalized Voronoi diagram for non-point convex polyhedra in d-dimensions is yet to be conclusively established [1].

A generalized Voronoi diagram was proposed in [2,3] in the context of motion planning in three dimensions for a convex polyhedron M with non-empty interior moving among convex, pairwise interior disjoint polyhedral obstacles O_is with non-empty interiors. It was shown in [3] that the size of a generalized Voronoi diagram for non-point convex polyhedra in three dimensions is given by $O(n^2Q^2l^2)$, where n is the total number of two facets on the obstacles, Q is the number of obstacles and l is the number of two facets on the moving polyhedron.

This paper extends the result of [3] to d-dimensions. We show that in d-dimensions, the size of the generalized Voronoi diagram is given by $O(f(d)v^dQ^{d-1}v_M^d)$ where $f(d)$ is a singly exponential function of d, v is the total number of vertices on the obstacles, Q is the total number of obstacles and v_M is the total number of vertices on the moving object. Our proof uses a recursion technique.

2. Preliminaries

In this section, we briefly describe some important definitions. More details can be found in [4].

Definition 2.1 A set S is said to be *polyhedral* if S can be written as a finite union of convex polyhedra, i.e., $S = \bigcup_{i=1}^{n} P_i$, where each P_i is a convex polyhedron and n is finite.

Definition 2.2 Suppose X_i, $i = 1, \ldots, n$ are polyhedral sets. Let E_i be the set of all open 1-faces of X_i and V_i be the set of all 0-faces of X_i. Then we call the set $S = \bigcup\{E_i \bigcup V_i\}$ the *wireframe* of $\{X_i\}$.

The following definition is taken from a paper by Leven and Sharir [5].

Definition 2.3 Let $x \in R^d$. The *M-distance of a set A from x* is defined as the minimum expansion required of M when v_{ref} is placed at x such that M "just touches" A. Formally,

$$d(x; A) = \inf\{\lambda : (x + \lambda M)\bigcap A \neq \phi, \lambda \geq 0\}$$

If $x \in A$ then $d(x; A) = 0$. For convenience, we write $d(x; O_i)$ as $d_i(x)$.

Definition 2.4 Let O_i be an obstacle. Then *the cell associated with* O_i, C_i is the set

$$\{x \in R^d : d_i(x) \leq d_j(x) \ \forall j \neq i, j \in 1, \ldots, Q\}$$

Physically, this is the set of all points in R^d from where M is closer to O_i than any other obstacle. It is not difficult to see that each cell is polyhedral.

Definition 2.5 Let $x \in R^d \setminus \bigcup O_i$ and consider $(x + d_i(x)M)$. Clearly $(x + d_i(x)M)$ touches O_i. Then, by convexity of O_i and M, a unique open facet o_i of O_i is being touched by a unique open facet o_m of M. We call the ordered pair (o_i, o_m) as the *touch description associated with the touch.*

Definition 2.6 For the touch description associated with a touch t we define *loss of degrees of freedom* $ldof(t)$ as $(d + 1)-$ the number of free variables in the set of linear equations which describe the touch.

---

[1]This research was supported in part by DST-ONR research project on Modelling, Analysis and Control of Discrete Event Systems under grant N00014-93-1017

Consider an $x \in R^d$. Suppose x is such that exactly k obstacles $O_1, \ldots, O_k$ are equidistant from x and no other obstacle is as close as any of these k obstacles. Then when M is placed with v_{ref} on x and expanded by $d_1(x)$, there are exactly k touches, say $t_1, \ldots, t_k$. Each one of these touches t_i has one touch description (o_i, o_{m_i}) associated with it.

Definition 2.7 We call the list $(o_1, o_{m_1}, \ldots, o_k, o_{m_k})$ as the *type of touch* T at x.

Definition 2.8 Consider a type of touch T. By definition T is a $2k$-tuple $(o_1, o_{m_1}, \ldots, o_k, o_{m_k})$. The *loss of degrees of freedom associated with the type of touch T,* $ldof(T)$ is defined as the sum of the loss of degrees of freedom for each touch description (o_i, o_{m_i}) associated with the touch t_i, i.e. $ldof(T) = \sum ldof(t_i)$.

As in [3], we make certain generic assumptions [6] on the relative orientations of the obstacles. We give here only one of those which we will require later.

Assumption Let $k \geq 1$ and consider any k distinct touches, each touch being described by a touch description $t_i, i = 1, \ldots, k$. Then the set of all points where exactly these k touches (and no other) are maintained is either empty or a $(d+1) - (\sum ldof(t_i))$ dimensional manifold. A set having negative dimension is taken as the null set.

We use the name *skeleton* for the Voronoi diagram which is formally defined as:

Definition 2.9 The *skeleton of* $R^d \setminus \bigcup O_i$ is the wireframe of $\{C_i\}_{i=1}^{i=Q}$ where C_i is as in definition 2.4 and wireframe is as in definition 2.2.

3. Size of Voronoi diagram

In this section, we prove the main results. Because of lack of space, detailed proofs are omitted from this paper. Instead, we try to give the motivation by means of an example; we believe this will help understanding the underlying idea behind the proof. The main results are then stated in forms of theorems. Full details can be found in [4].

We will use the following notations. n_i^k will denote the number of k dimensional facets of obstacle O_i. n_i will denote the maximum of n_i^k over all k, i.e., $n_i = \max_k n_i^k$. n will denote the sum of n_i over all i, i.e., $n = \sum_i n_i$. Similarly, l^k will denote the number of k dimensional facets of the moving object M and l will denote $\max_k l^k$.

Proposition 3.1 Consider only two obstacles O_i and O_j. Then the total number of k dimensional facets in the final set formed by the cell boundaries of C_i and C_j is $O(d^3 n_i n_j l^2)$.

Proposition 3.1 gives a bound on the number of k dimensional facets when only two obstacles are present. However, in a general scenario, the number of obstacles is more than 2. Therefore we must derive a bound for the same when Q obstacles are present.

Our counting process uses an incremental technique. We start with only two obstacles and then use proposition 3.1 to derive the bounds on the size of the k dimensional facet sets. Next, we introduce *only* a third obstacle and find a bound on the *extra* k dimensional facets added because of this new obstacle. This process is repeated; i.e., obstacles are added *one by one* and we find a bound on the number of *extra* k dimensional facets added each time.

The above process is finite by virtue of independence. By independence, we know that not more than d obstacles can contribute a point to an edge. Thus it is sufficient if in the counting process we consider presence of only d obstacles to derive a bound on the size of k dimensional facet sets $\forall k \geq 1$. Other obstacles can affect these sets, but in a different manner. They can further subdivide these sets, and that can be taken care of separately. Also, since each vertex belongs to the closure of an edge and each edge contains at most two vertices, the size of the 0-dimensional facet set is the same as 1-dimensional facet set.

First we consider the presence of only two obstacles, and use proposition 3.1 to derive the bounds on the size of k dimensional facet sets. Next we introduce a third obstacle. This third obstacle creates new k dimensional facets by pairing with the two obstacles already present, and also by intersection between existing $(k+1)$ dimensional facets and newly generated $(k+1)$ dimensional facets. Thus the additional k dimensioanl facets generated are counted and listed down. Next we introduce the fourth obstacle, and consider the generation of k dimensional facets as before. This process goes on till we complete the addition of the d-th obstacle. Then we sum up the terms corresponding to each k dimensional facet which gives us a bound.

Since we are interested in a bound and not an exact count, we use certain bounding approximations to make the calculation simpler. Suppose E_r^k is a bound that we develop on the number of extra k dimensional facets added when the r-th obstacle is introduced. In the process of counting, we find that $E_r^k > E_s^k \; \forall s < r$ and $E_r^k \leq E_r^s \; \forall k > s$. That is, E_d^1 is the biggest term of all E_r^k, and thus it is sufficient if we can compute E_d^1; then, $E_r^k \leq E_d^1 \; \forall k, 1 \leq k \leq d-1$ and $r \leq d$. This is the idea used for the counting. We illustrate the idea informally using a five dimensional example.

Consider a five dimensional space. Suppose only five obstacles are present. Then the facet dimensions we need to consider are $0 \leq k \leq 4$. As already mentioned, 0-dimensional facet set size is the same as 1-dimensional facet set size; so it is sufficient to consider only $1 \leq k \leq 4$.

First consider only two obstacles, O_1 and O_2. Then by proposition 3.1, the size of each dimensional facet set is $O(n_1 n_2 d^3 l^2)$. For the sake of convenience, we drop the order notation, and the terms d^3 and l^2 in the following analysis.

Now consider the introduction of a third obstacle O_3. Then clearly, extra k dimensional facets will be introduced. Also, a new k dimensional facet can now be introduced in one of the two following ways: i) by *direct generation* because of combinations of facets between (O_1 and O_3) and (O_2 and O_3) and ii) by intersection of one existing $(k+1)$ dimensional facet and one newly generated $(k+1)$ dimensional facet.

Consider $k = 4$. They are generated only directly and the extra number is $n_1 n_3 + n_2 n_3$ (note that this is just a bound. It contains the terms d^3 and l^2 too). For convenience in counting, we replace this by the bound $E_2^4 + (n_1 n_3 + n_2 n_3)$, which we call E_3^4. Consider $k = 3$. There directly generated 3-dimensional facets add $(n_1 n_3 + n_2 n_3)$ facets which we again replace by the bound $E_2^4 + (n_1 n_3 + n_2 n_3)$. Now consider intersection of 4-dimensional facets. It is easy to see that argument similar to that used for polygon intersection in 3-dimensional case [3] holds and the number of 4-dimensional facet intersections is $(E_2^4 + n_1 n_3) + (E_2^4 + n_2 n_3)$. We call this total number of extra 3-dimensional facets added as E_3^3.

Now the following observations can be made.

1. $E_r^4 > E_m^4 \; \forall m < r$. This is clearly true, as whenever a new obstacle O_r is added; the term E_r^4 becomes $E_{r-1}^4 + (n_1 + \cdots + n_{r-1})n_r > E_{r-1}^4$ and then by simple recursion the result follows.

2. $E_3^k \geq E_3^m \; \forall k < m$. This also is easy to see, as clearly $E_3^4 < E_3^3$ and E_3^2 and E_3^1 are the same as E_3^3.

Now consider the introduction of the fourth obstacle, O_4.

Then $E_4^4 = E_3^4 + (n_1 + n_2 + n_3)n_4$.

The 3-dimensional facets are introduced as before by one of the following ways: i) direct generation and ii) intersection of one old and one newly generated 4-dimensional facet. Direct generation number is E_4^4. But while considering the intersection of 4-dimensional facets we cannot use the arguments similar to that used for three obstacles case, as now the polytopes may be non-convex. However, observe the following. Suppose a non-convex polytope P_2 of dimension $\bar{d}$ is being intersected by another polytope P_1 of dimension $\bar{d}$. Then P_1 cannot generate more than one $(\bar{d}-1)$ dimensional facet in P_2 as it is truncated by the polytopes sharing the boundary of P_2. Let us take an example to illustrate the idea. Suppose a two dimensional non-convex polytope P_2 is being intersected by a two dimensional polytope P_1. Suppose the intersection can create three distinct edges e_1, e_2, e_3. Suppose both e_1 and e_2 are newly contributed edges. But the section $\bar{e}$ between e_1 and e_2 is not a new edge and that violates the convexity of the obstacles, which is a contradiction. Thus only one of e_1, e_2, e_3 can exist as a newly created edge. Therefore we need to repeat our arguments over only one possible intersection. Then a similar argument to that used for the three dimensional case holds, and the number of three dimensional facets generated by four dimensional facet intersection is $(E_3^4 + n_1 n_4) + (E_3^4 + n_2 n_4) + (E_3^4 + n_3 n_4)$.

We calculate E_4^2 as follows. They are generated by i)direct generation ii)intersection of two three dimensional facets, one existing and one newly generated and iii)intersection of three four dimensional facets, at least one of which is newly generated and at least one is existing. i) is easily calculated as $(E_3^4 + (n_1 + n_2 + n_3)n_4)$. ii) is calculated as before: (number of existing three dimensional facets $+n_1n_4$) + (number of existing three dimensional facets + n_2n_4) + (number of existing three dimensional facets + n_3n_4). Number of existing three dimensional facets can be enumerated as $E_2^3 + E_3^3$. Since $E_3^3 > E_2^3$, number of existing three dimensional facets is bounded by $2E_3^3$ and thus ii) is counted as $(2E_3^3 + n_1n_4) + (2E_3^3 + n_2n_4) + (2E_3^3 + n_3n_4)$.

Case iii) is enumerated as follows. First we consider intersection of four dimensional facets, one pair at a time. There will be exactly two such pairs, each generating one three dimensional facet, which in turn will intersect to generate a two dimensional facet. But the number of three dimensional facets generated by four dimensional facet intersection is already calculated to be $3E_3^4 + (n_1 + n_2 + n_3)n_4$. These three dimensional facets in turn can generate at most the same order of two dimensional facets by intersection among themselves (by the argument given for case ii) of three dimensional facet generation) and thus case iii) is enumerated as $3E_3^4 + (n_1 + n_2 + n_3)n_4$.

Now we make the following observations.

1. $E_r^k \geq E_r^m \; \forall k < m$. This is easy to see. Clearly $E_r^3 \geq E_r^4$ as $E_r^4 = E_{r-1}^4 + (n_1 + \cdots + n_{r-1})n_r$ and $E_r^3 = E_r^4 + (r-1)E_{r-1}^4 + (n_1 + \cdots + n_{r-1})n_r$. Now, E_r^2 is calculated as the sum of directly generated two dimensional facets and intersection of existing three and four dimensional facets. with newly generated three and four dimensional facets. Therefore, $E_r^2 \geq E_r^3$. E_r^1 is calculated as directly generated + intersection of two and three dimensional facets, and so $E_r^1 \geq E_r^2$. This process can be made recursive which gives the result.

2. $E_r^k \geq E_m^k \; \forall r > m$. E_r^4 is monotone increasing. Therefore E_r^3 is, and therefore E_r^2 and in turn E_r^1.

By these two observations, we have that $E_d^1 \geq E_r^k \; \forall r = 2, \ldots, d, k = 1, \ldots, d-1$. Therefore it is sufficient if we calculate the term E_d^1. Also note that to calculate E_d^1 it suffices to find only the reverse diagonal lower triangular matrix of the structure.

Number of Obstacles	Facet dimension			
	1	2	3	4
2				E_2^4
3			E_3^3	E_3^4
4		E_4^2	E_4^3	E_4^4
5	E_5^1	E_5^2	E_5^3	E_5^4

This is true because E_5^1 is affected *only* by the number of existing 2, 3 and 4 dimensional facets. But $E_4^2 \geq E_3^2 \geq E_2^2$ and thus it is sufficient to have E_4^2. Similarly for E_4^3 and E_4^4. Again. E_4^2 is affected *only* by existing three and four dimensional facets and $E_3^3 \geq E_2^3$; so it is sufficient to have only E_3^3. A simple recursive argument then validates our claim.

Now let us calculate E_5^1. These one dimensional facets are generated in one of the following ways: i)direct generation ii)intersection of two two dimensional facets, one existing, one newly generated iii)intersection of three three dimensional facets,at least one existing and at least one newly generated and iv)intersection of four four dimensional facets, at least one existing and at least one newly generated. Case i) is enumerated as $E_4^4 + (n_1 + \cdots + n_4)n_5$. Case ii) is enumerated as $(3E_4^2 + n_1n_5) + (3E_4^2 + n_2n_5) + (3E_4^2 + n_3n_5) + (3E_4^2 + n_4n_5)$. Case iii) is enumerated as $(3E_4^3 + n_1n_5) + \cdots + (3E_4^3 + n_4n_5)$ (This is seen as before: the above expression determines the number of two dimensional facets generated by intersection of three dimensional facets, and each such two dimensional facet can create the same order of one dimensional facet among themselves). Case iv) is done similarly to give $4E_4^4 + (n_1 + \cdots + n_4)n_5$.

E_5^1 is a bound for the edge set created by 2-tuples, 3-tuples ,..., 5-tuples. Thus edge set size is $O(Q)(E_5^1|_{2-tuples} + \cdots + E_5^1|_{5-tuples})$. The term $O(Q)$ comes as every edge can be broken into two by every other obstacle. As seen before, vertex set size = edge set size.

By independence, at most d obstacles can contribute a point to an edge. Thus at most d two dimensional facets can share an edge. So size of two dimensional facet set is $O(d)e$ where e is the size of the edge set. Similarly, size of three dimensional facet set is $O(d-1)O(d)e$, or in general, the size of any k dimensional facet set is bounded by $O((d!)e)$.

Theorem 3.2 Suppose there exist Q obstacles $O_1,\ldots,O_Q, Q > d$. Then the total number of k dimensional facets in the final set formed by the boundaries of cells $C_1,\ldots,C_Q$ is $O((d!)Qe)$ where e is the size of the edge set when only d obstacles are present.

Proof Since every edge can be broken into two by each of the other obstacles, the edge set size for Q obstacles is $O(Qe)$. Then by the discussion preceeding Theorem 3.2, the result follows. ∎

Theorem 3.2 gives a bound on the number of k dimensional facets in terms of e, the number of edges when only d obstacles are present. Next, we derive a bound on e using the ideas illustrated by the example.

Theorem 3.3 Suppose only d obstacles $O_1,\ldots,O_d$ are present. Then the total number of one dimensional facets in the final set formed by the boundaries of the cells $C_1,\ldots,C_d$ is $O(2^d(d!)^2d^3n^2Q^{d-2}l^2)$.

Proof We will give a short outline of the proof; the principle being the same as illustrated before in the five dimensional example.

In the following discussion, for the sake of convenience we write n_in_j in place of $O(d^3n_in_jl^2)$. Whenever we do specific operations on order notations, we mention that.

First consider the presence of only two obstacles O_1 and O_2. Then by proposition 3.1, the size of each of the k dimensional facet set is defined by these two obstacles and is n_1n_2.

Next we consider the introduction of a third obstacle say O_3. We count the number of extra $d-1$ dimensional facets added and extra $d-2$ dimensional facets added. We will use the notation E_r^k to denote the number of extra k dimensional facets added when the r-th obstacle is introduced. Then as in the previous discussion, $E_3^{d-1} = \sum_{i<j,1\le i,j\le 3} n_in_j$ and $E_3^{d-2} = (n_1n_2+n_1n_3)+(n_1n_2-n_2n_3)$.

In general, we find that $E_r^{d-1} = \sum_{i<j,1\le i,j\le r} n_in_j$ and $E_r^{d-2} = E_r^{d-1}+(r-1)E_{r-1}^{d-1}+(n_1+\cdots+n_{r-1})n_r, r\ge 3$.

We must find a similar recursive relation for any dimensional facet. Now observe the following. For any $k\in\{1,2,\ldots,d-3\}$; we need to sum up the extra $k+1, k+2,\ldots$ dimensional facets generated in every step to find the number of intersections they produce; and thus:

$$\begin{aligned}E_r^m = E_r^{d-1} &+ (r-1)E_{r-1}^{d-1}+(n_1+\cdots+n_{r-1})n_r\\ &+ (r-1)(r-2)E_{r-1}^{d-2}+(n_1+\cdots+n_{r-1})n_r\\ &+ \cdots\\ &+ (r-1)(r-2)E_{r-1}^{m+1}+(n_1+\cdots+n_{r-1})n_r\end{aligned}$$

where $r+m\ge d+1$. This final constraint is necessary, as when the r-th obstacle is added, the effect of $d-1$ dimensional facets can come down only upto the $[(d-1)-(r-2)]$ dimensional facets. Since we are counting the reverse diagonal lower triangular part of the matrix, we always satisfy the relation.

Observe that the expression of E_r^m has the term $(r-1)(r-2)$ associated with each $E_{r-1}^k, k\in\{d-2,\ldots,m+1\}$ except for E_{r-1}^{d-1} which has only $(r-1)$ associated with it. To make the recursion simpler, and since we are using order notations, we can replace $(r-1)E_{r-1}^{d-1}$ by $(r-1)(r-2)E_{r-1}^{d-1}$ and thus we will work with E_r^{d-2} replaced by the expression: $E_r^{d-2} = E_r^{d-1}+(r-1)(r-2)E_{r-1}^{d-1}+(n_1+\cdots+n_{r-1})n_r$ and similarly modifying the expression for E_r^m.

We claim that: $E_r^m = E_r^{m+1}+(r-1)(r-2)E_{r-1}^{m+1}+(n_1+\cdots+n_{r-1})n_r$.

The result can be proved easily by induction.

Then the number of one dimensional facets is $\sum_{2-tuples} E_2^1+\sum_{3-tuples} E_3^1+\cdots+\sum_{d-tuples} E_d^1$. The final form uses d-tuples as there are only d obstacles present by assumption. Also, $E_d^1\ge E_r^1\ \forall r\in\{2,3,\ldots,d\}$ and thus it is sufficient to find E_d^1 as in order notation it bounds every E_r^1.

By the above claim, we can express E_d^1 as $E_d^1 = E_d^2+(d-1)(d-2)E_{d-1}^2+(n_1+\cdots+n_{d-1})n_d$. This expression for E_d^1 can be further expanded using the claim (by expanding the terms E_d^2 and

E_{d-1}^2), and finally we get an expression for E_d^1 involving $E_k^{d-1}\ \forall k = 2, \ldots, d$. This final expression can be written as:

$$E_d^1 = O(2^d (d!)^2 [n_1 n_2 + (n_1 + n_2) n_3 + \cdots + (n_1 + \cdots + n_{d-1}) n_d])$$

The above sum is valid as we are using order notations. Using proposition 3.1, we find that the term within the square parentheses equals $O(d^3 l^2 \sum_{i<j, 1 \leq i,j \leq d} n_i n_j)$, and therefore E_d^1 is given by

$$E_d^1 = O(2^d (d!)^2 d^3 l^2 \sum_{i<j, 1 \leq i,j \leq d} n_i n_j).$$

Then summing over all possible combinations, we find

$$\sum_{2-tuples} E_d^1 + \sum_{3-tuples} E_d^1 + \cdots + \sum_{d-tuples} E_d^1 \quad = \quad O(2^d (d!)^2 d^3 n^2 Q^{d-2} l^2)$$

which proves the theorem. ■

Theorem 3.4 Suppose there exist Q obstacles $O_1, \ldots, O_Q$. Then the total number of k dimensional facets in the final set formed by the boundaries of cells $C_1, \ldots, C_Q$ is $O(2^d (d!)^3 n^2 Q^{d-1} l^2)$.

Proof Immediate from theorem 3.2 and theorem 3.3. ■

Remark In theorem 3.4, we have chosen $n = \sum_i n_i$, $n_i = \max_k n_i^k$ where n_i^k is the number of k dimensional facets of obstacle O_i. Now suppose the number of vertices of O_i is v_i. Then n_i is of size $O(v_i^{\lfloor \frac{d}{2} \rfloor})$. Therefore, $n = O((\sum_i v_i)^{\lfloor \frac{d}{2} \rfloor})$ [as $\sum_i v_i^{\lfloor \frac{d}{2} \rfloor} \leq (\sum_i v_i)^{\lfloor \frac{d}{2} \rfloor}$]. Let $v = \sum_i v_i$. then the size of each of the k dimensional facet sets is $O(2^d (d!)^3 v^d Q^{d-1} l^2)$.

l can also be expressed similarly. Suppose the number of vertices on M is v_M. Then $l = O(v_M^{\lfloor \frac{d}{2} \rfloor})$, and so the size of each k dimensional facet set is $O(2^d (d!)^3 v^d Q^{d-1} v_M^d)$.

4. Conclusion

In this paper, we have established a bound on the size of a generalized Voronoi diagram for non-point convex polyhedra in d-dimensions. Our proof uses a recursive technique which can find applications to other geometric problems as well.

The bound obtained in this paper is clearly not optimal: this can be easily seen by substituting $d = 3$ in theorem 3.4 and comparing the result with the three dimensional bound in [3]. It will be an interesting exercise to obtain a better bound, probably an optimal one: though there are reasons to believe that this question is hard.

Acknowledgement The author would like to thank Abhijit Das for his help in the proof of theorem 3.3 and Prof. S. S. Keerthi for his useful suggestions.

References

[1] V. Srinivasan, IBM T.J. Watson Research Center, New York, *Personal Communication.*

[2] A. Dattasharma and S. S. Keerthi, *"Translational Motion Planning for a Convex Polyhedron in a 3D Polyhedral World Using an Efficient and New Roadmap"*, Proc. of 5th Canadian Conf. on Comput. Geo., 1993.

[3] A. Dattasharma and S. S. Keerthi, *"An Augmented Voronoi Roadmap for 3D Translational Motion Planning for a Convex Polyhedron Moving Amidst Convex Polyhedral Obstacles"*, Theoretical Computer Science, To appear.

[4] A. Dattasharma, *"Structure and Computation of a Class of Generalized Voronoi Diagrams with application to Translational Motion Planning"*, Ph. D. Thesis, Department of Computer Science and Automation, Indian Institute of Science, Bangalore, India.

[5] D. Leven and M. Sharir, *" Planning a Purely Translational Motion for a Convex Object in Two-Dimensional Space Using Generalized Voronoi Diagrams"*, Disc. and Comput. Geo., Springer-Verlag, Vol 2, pp 9–31, 1987.

[6] S.S. Keerthi, N.K. Sancheti and A. Dattasharma, *" Transversality Theorem : A Useful Tool for Establishing Genericity"*, Proc. IEEE Conf. on Decision and Control, 1992.