

Applied and  
Computational  
Mathematics  
Division

NISTIR 4340

---

Center for Computing and Applied Mathematics

---

*An Expected Linear  
3-Dimensional Voronoi Diagram  
Algorithm*

*J. Bernal*

*June 1990*

---

**NIST** United States Department of Commerce  
National Institute of Standards and Technology



# **AN EXPECTED LINEAR 3-DIMENSIONAL VORONOI DIAGRAM ALGORITHM**

**J. Bernal**

**U.S. DEPARTMENT OF COMMERCE  
National Institute of Standards  
and Technology  
Center for Computing and Applied  
Mathematics  
Applied and Computational Mathematics  
Division  
Gaithersburg, MD 20899**

**June 1990**



**U.S. DEPARTMENT OF COMMERCE  
Robert A. Mosbacher, Secretary  
NATIONAL INSTITUTE OF STANDARDS  
AND TECHNOLOGY  
Dr. John W. Lyons, Director**



# AN EXPECTED LINEAR 3-DIMENSIONAL VORONOI DIAGRAM ALGORITHM

Javier Bernal

Applied and Computational Mathematics Division  
National Institute of Standards and Technology  
Gaithersburg, MD 20899

*Key Words and Phrases:* algorithm; computational geometry; expected complexity; expected linear; expected time analysis; Voronoi diagram.

## ABSTRACT

Let  $S$  be a set of  $n$  sites chosen independently from a uniform distribution in a cube in 3-dimensional Euclidean space. In this paper, an expected  $O(n)$  algorithm for constructing the Voronoi diagram for  $S$  together with numerical results obtained from an implementation of the algorithm are presented.

## 1. INTRODUCTION

Consider a set  $S = \{p_1, \dots, p_n\}$  of  $n$  points (to be called *sites* in the following) in the Euclidean space  $E^3$ , and let  $d(\cdot, \cdot)$  denote the Euclidean distance. The *Voronoi diagram* for  $S$  is a sequence  $V(p_1), \dots, V(p_n)$  of convex polyhedra covering  $E^3$ , where for each  $i$ ,  $i = 1, \dots, n$ ,  $V(p_i)$ , the *Voronoi polyhedron of  $p_i$  relative to  $S$* , is defined by

$$V(p_i) \equiv \bigcap_{j=1, j \neq i}^n \{p \in E^3 : d(p, p_i) \leq d(p, p_j)\}.$$

The Voronoi diagram has played an important role in computational geometry for a long time, and several algorithms have been devised and implemented for constructing it in two and higher dimensions (see Bentley, Weide and Yao (1980), Bowyer (1981), Brostow, Dussault and Fox (1978), Brown (1979), Dwyer (1988), Finney (1979), Green and Sibson (1978), Lee and Schachter (1980), Maus (1984), Ohya, Iri and Murota (1984), Seidel (1986), Shamos (1978), Shamos and Hoey (1975), Tanemura, Ogawa and Ogita (1983), Watson (1981), Witzgall (1973)).

Assume the sites in  $S$  have been chosen independently from a uniform distribution in a 3-dimensional cube. In this paper we present an expected  $O(n)$

algorithm for constructing the Voronoi diagram for  $S$  that is a consequence of proofs and results in the companion paper Bernal (1990). Numerical results obtained from a Fortran implementation of the algorithm are also presented.

## 2. TERMINOLOGY

Let  $S = \{p_1, \dots, p_n\}$  be a set of  $n$  sites in  $E^3$  chosen independently from a uniform distribution in a cube  $R$ . With  $m$  defined as the *floor* of  $n^{1/3}$ , i. e. the largest integer less than or equal to  $n^{1/3}$ , assume as in Bentley, et al. (1980) that  $R$  has been divided into  $m^3$  equal-sized cells. Given a site  $q$ , define *the 1<sup>st</sup> layer of cells that surrounds  $q$*  as the collection of cells that contain  $q$ . Inductively, given  $k \geq 1$ , assume that the  $k^{\text{th}}$  layer of cells that surrounds  $q$  has been defined. Define *the  $(k+1)^{\text{th}}$  layer of cells that surrounds  $q$*  as the collection, possibly empty, of cells that have one or more points in common with cells in the  $k^{\text{th}}$  layer, and that do not belong to the first  $k$  layers.

Let  $lcell$  and  $vcell$  represent, respectively, the length and volume of each cell.

Given numbers  $c, c', c'', 0 < c \leq c', c'' \geq 1$ , define  $LG(n)$  and  $LG'(n)$  as the floors of  $c \cdot \log n$  and  $c' \cdot \log n$ , respectively, and assume  $n$  is large enough so that  $LG(n) > 2$  and  $2^{3/2} \cdot c'' \cdot LG'(n) \leq 2^{-1} \cdot n^{1/3}$ .

Let  $\hat{k}$  denote the largest integer  $k$  for which

$$2^{k/2} \cdot c'' \cdot LG'(n) \leq 2^{-1} \cdot n^{1/3}.$$

It follows from the assumptions on  $n$  that  $\hat{k} \geq 3$ .

Set  $LG_0(n)$  equal to  $LG(n)$ , and  $LG_k(n)$  equal to  $LG'(n)$  for each  $k, k = 1, \dots, \hat{k} - 2$ .

Let  $f_i, i = 1, \dots, 6$ , represent the facets of  $R$ , and let  $\Pi$  denote  $\cup_{i=1}^6 f_i$ , i. e. the boundary of  $R$ .

Given a point  $x$  in  $E^3$  and a closed subset  $W$  of  $E^3$ , define  $\text{dist}(x, W)$  as the minimum value of  $\|x - w\|$  for  $w$  in  $W$ , where  $\|\cdot\|$  represents the 3-dimensional Euclidean norm.

From the assumptions on  $n$ , several nonempty subsets of  $R$  can be defined as follows:

$$\begin{aligned} R_{-1} &\equiv \{x \in R : \text{dist}(x, \Pi) \geq l_{\text{cell}} \cdot \text{LG}(n)\}. \\ R_0 &\equiv \{x \in R : l_{\text{cell}} \cdot 2 \leq \text{dist}(x, \Pi) < l_{\text{cell}} \cdot \text{LG}(n)\}. \\ R_{\hat{k}} &\equiv \{x \in R : \text{dist}(x, \Pi) < l_{\text{cell}} \cdot 2^{-\hat{k}+2}\}. \end{aligned}$$

For each  $k$ ,  $k = 1, \dots, \hat{k} - 1$ ,

$$R_k \equiv \{x \in R : l_{\text{cell}} \cdot 2^{-k-1} \leq \text{dist}(x, \Pi) < l_{\text{cell}} \cdot 2^{-k+2}\}.$$

For each  $i$ ,  $k$ ,  $i = 1, \dots, 6$ ,  $k = 0, \dots, \hat{k} - 2$ ,

$$R_k^i \equiv \{x \in R_k : \text{dist}(x, f_j) \geq l_{\text{cell}} \cdot 2^{k/2} \cdot c'' \cdot \text{LG}_k(n), j = 1, \dots, 6, j \neq i\}.$$

It follows from these definitions that the sets  $R_k$ ,  $k = -1, \dots, \hat{k}$ , are pair-wise disjoint nested regions of the cube  $R$ , and

$$R = \bigcup_{k=-1}^{\hat{k}} R_k.$$

The significance of these regions for our purposes can be summarized as follows.  $R_{-1}$  is essentially that region of the cube  $R$  obtained by subtracting the outermost  $\text{LG}(n)$  layers of cells of  $R$  from  $R$ . From Bentley, et al. (1980), the Voronoi polyhedron of a site in  $R_{-1}$  can be constructed in expected constant time.  $R_0$  is essentially that region of  $R$  obtained by subtracting from the outermost  $\text{LG}(n)$  layers of cells of  $R$  the outermost two layers.  $R_k$ ,  $k = 1, \dots, \hat{k}$ , are regions of  $R$  whose union is essentially that region of  $R$  composed of the outermost two layers of cells of  $R$ , and whose thicknesses correspond to the terms of the geometric series expanded to the first  $\hat{k} - 1$  terms together with the remainder.  $R_k^i$ ,  $i = 1, \dots, 6$ ,  $k = 0, \dots, \hat{k} - 2$ , are subsets of  $R_k$ ,  $k = 0, \dots, \hat{k} - 2$ , respectively, defined in such a way that as intimated in Bernal (1990), due to their positions relative to the boundary of  $R$  and the geometric series aspect of  $R_k$ ,  $k = 1, \dots, \hat{k} - 2$ , for a properly selected value of  $c''$  the Voronoi polyhedra of sites in these regions can be constructed in expected linear time. They are also defined in such a way that due to the definitions of  $\hat{k}$ ,  $R_{\hat{k}-1}$  and  $R_{\hat{k}}$ , and the geometric series aspect of  $R_k$ ,  $k = 1, \dots, \hat{k}$ , the expected number of sites in  $\bigcup_{k=0}^{\hat{k}} R_k \setminus \bigcup_{i=1}^6 \bigcup_{k=0}^{\hat{k}-2} R_k^i$  is small enough that the Voronoi polyhedra of these sites can also be constructed in expected linear time even under the worst possible circumstances.

Given a site  $q$  in  $R_{-1}$ , let  $v$ ,  $v'$ ,  $v''$  and  $v'''$  be vertices of  $R$  for which  $v' - v$ ,  $v'' - v$  and  $v''' - v$  are all perpendicular to one another, and for each  $j$ ,  $j = 0, \dots, 8$ , and each  $m$ ,  $m = 0, \dots, 4$ , define a point  $r_{jm}$  by

$$\begin{aligned} r_{jm} &\equiv q + ((v' - v) \cdot \cos(j\pi/4) + (v'' - v) \cdot \sin(j\pi/4)) \cdot \sin(m\pi/4) \\ &\quad + (v''' - v) \cdot \cos(m\pi/4). \end{aligned}$$

In addition, for each  $j$ ,  $j = 1, \dots, 8$ , and each  $m$ ,  $m = 1, \dots, 4$ , let  $U_{jm}$  be the cone that is the convex hull of the rays  $q\vec{r}_{j-1,m-1}$ ,  $q\vec{r}_{j,m-1}$ ,  $q\vec{r}_{j-1,m}$ , and  $q\vec{r}_{jm}$ , and if within the first  $\text{LG}(n)$  layers of cells that surround  $q$ , for each  $j$ ,  $j = 1, \dots, 8$ , and each  $m$ ,  $m = 1, \dots, 4$ , there exists a site  $s_{jm}$ ,  $s_{jm} \neq q$ , such that  $s_{jm}$  belongs to  $U_{jm}$ , say that  $q$  is *closed* and that  $s_{jm}$ ,  $j = 1, \dots, 8$ ,  $m = 1, \dots, 4$ , *render  $q$  closed*. As shown in Bentley, et al. (1980), the Voronoi polyhedron of a closed site can be constructed in expected constant time.

For each facet  $f$  of  $R$ , let  $H(f)$  represent the plane that contains  $f$ , and for each site  $q$ , let  $T^f(q)$  represent the point in  $f$  that is the perpendicular projection of  $q$  onto  $f$ .

Given  $i$ ,  $k$ ,  $1 \leq i \leq 6$ ,  $0 \leq k \leq \hat{k} - 2$ , and a site  $q$  in  $R_k^i$ , let  $v$ ,  $v'$  and  $v''$  be vertices of  $R$  in  $f_i$  for which  $v' - v$  is perpendicular to  $v'' - v$ , and for each  $j$ ,  $j = 0, \dots, 8$ , define a point  $t_j$  in  $H(f_i)$  by

$$t_j \equiv T^{f_i}(q) + (v' - v) \cdot \cos(j\pi/4) + (v'' - v) \cdot \sin(j\pi/4).$$

In addition, for each  $j$ ,  $j = 1, \dots, 8$ , let  $O_j$  be the octant in  $H(f_i)$  that is the convex hull of the rays  $T^{f_i}(q)\vec{t}_{j-1}$  and  $T^{f_i}(q)\vec{t}_j$ , and if within the first  $2^{k/2} \cdot \text{LG}_k(n)$  layers of cells that surround  $q$ , for each  $j$ ,  $j = 1, \dots, 8$ , there exists a site  $q_j$  such that  $\text{dist}(q_j, f_i) < l_{\text{cell}} \cdot 2^{-k}$  and the ray  $q\vec{q}_j$  intersects  $O_j$ , say that  $q$  is *octant-closed* and that  $q_j$ ,  $j = 1, \dots, 8$ , *render  $q$  octant-closed*.

Given  $i$ ,  $k$ ,  $q$ ,  $v$ ,  $v'$ ,  $v''$  as above, let  $v'''$  be a vertex of  $R$  for which  $v''' - v$  is perpendicular to  $v' - v$  and  $v'' - v$ , and for each  $j$ ,  $j = 0, \dots, 8$ , and each  $m$ ,  $m = 2, 3$ , define a point  $r_{jm}$  by

$$r_{jm} \equiv q + ((v' - v) \cdot \cos(j\pi/4) + (v'' - v) \cdot \sin(j\pi/4)) \cdot \sin(m\pi/4) \\ + (v''' - v) \cdot \cos(m\pi/4).$$

In addition, for each  $j$ ,  $j = 1, \dots, 8$ , let  $U_j$  be the cone that is the convex hull of the rays  $q\vec{r}_{j-1,2}$ ,  $q\vec{r}_{j,2}$ ,  $q\vec{r}_{j-1,3}$ , and  $q\vec{r}_{j,3}$ , and if within the first  $2^{k/2} \cdot \text{LG}_k(n)$  layers of cells that surround  $q$ , for each  $j$ ,  $j = 1, \dots, 8$ , there exists a site  $s_j$ ,  $s_j \neq q$ , such that  $s_j$  belongs to  $U_j$ , say that  $q$  is *cone-semiclosed* and that  $s_j$ ,  $j = 1, \dots, 8$ , *render  $q$  cone-semiclosed*.

Given  $q$  as above, say that  $q$  is *semiclosed* if it is octant-closed and cone-semiclosed. As intimated in Bernal (1990), for a properly selected value of  $c''$  the construction of Voronoi polyhedra of semiclosed sites is of expected complexity acceptable for our purposes.

Finally, given sites  $p$  and  $q$ , say that  $p$  is a *Voronoi neighbor of  $q$  relative to  $S$*  if  $V(p)$  and  $V(q)$  have a facet in common.

### 3. THE ALGORITHM

In this section we present the algorithm in the form of a procedure called VORNOI. The algorithm and its expected complexity follow from proofs and results in the companion paper Bernal (1990).

Essentially, the algorithm consists of three steps. Let  $n$ ,  $S$ ,  $R$ ,  $R_{-1}$ ,  $R_k^i$ ,  $i = 1, \dots, 6$ ,  $k = 0, \dots, \hat{k} - 2$ , be as defined in the previous section. In the first step, the Voronoi polyhedra of sites in  $R_{-1}$  are constructed as suggested in Bentley, et al. (1980). Given a site in  $R_{-1}$ , a geometrical procedure is available for constructing in expected constant time the Voronoi polyhedron of the site. Thus, the first step of the algorithm has expected linear complexity. In the second step, the Voronoi polyhedra of sites in  $R_k^i$ ,  $i = 1, \dots, 6$ ,  $k = 0, \dots, \hat{k} - 2$ , are constructed as intimated in Bernal (1990). Given a site in  $\cup_{i=1}^6 \cup_{k=0}^{\hat{k}-2} R_k^i$ , a geometrical procedure that generalizes the one used in the first step is available for obtaining a subset of  $S$  that contains all of the Voronoi neighbors relative to  $S$  of the site. This is done in such a way that as implied in Bernal (1990), the expected time involved in obtaining all such subsets for all such sites is bounded above by  $O(n^{2/3} \cdot (\log n)^4)$ . Thus, since an  $O(k \cdot \log k)$  procedure is also available for computing the intersection of  $k$  half-spaces in 3-dimensional space (see Preparata and Muller (1979)), a computation can be carried out that shows that the second step of the algorithm has at most expected  $O(n^{2/3} \cdot (\log n)^5)$  complexity. Finally, in the third step, the Voronoi polyhedra of sites in  $R \setminus ((\cup_{i=1}^6 \cup_{k=0}^{\hat{k}-2} R_k^i) \cup R_{-1})$  are constructed. As shown in Bernal (1990), a procedure is available for obtaining for each site in this region a subset of  $S$  that contains all of the Voronoi neighbors relative to  $S$  of the site. This is done in such a way that as implied in Bernal (1990), the expected time involved in obtaining all such subsets for all such sites is bounded above by  $O(n^{2/3} \cdot (\log n)^4)$ . Thus, since the  $O(k \cdot \log k)$  procedure used in the second step for computing the intersection of  $k$  half-spaces is also available in this step, it can be shown in a manner similar to the one used for the second step that the third step of the algorithm has also at most expected  $O(n^{2/3} \cdot (\log n)^5)$  complexity. Therefore, the entire algorithm has expected linear complexity.

In the following we list and describe, in the order of their first appearance in procedure VORNOI, functions and procedures used as primitives in that procedure.

**FLOOR( $x$ ):** For a positive real number  $x$  computes the largest integer less than or equal to  $x$ .

PRTION( $R, R', m, P$ ): Creates a partition of a cube  $R$  into  $m^3$  equal-sized cells, and then reduces it to those cells that intersect a region  $R'$  of  $R$ .  $P$  will contain the data structure that describes the reduced partition.

CEASGN( $S, P, A$ ): Using data structure in  $P$  obtained from PRTION procedure, assigns each site in a set  $S$  to a cell that contains the site in the partition associated with  $P$ , and for each cell in the partition creates a list of those sites assigned to the cell. The corresponding data structure will be contained in  $A$ .

RGASGN( $S, R', S', n', B'$ ): Locates and orders those sites in a set  $S$  that are contained in a region  $R'$ .  $S'$  will be the set of ordered sites,  $n'$  will be the number of sites in  $S'$ , and for each  $h$ ,  $h = 1, \dots, n'$ ,  $B'(h)$  will be the  $h^{\text{th}}$  site in  $S'$ .

CLTEST( $P, A, q, \text{LG}(n), \text{flag}, Q$ ): Using data structures in  $P$  and  $A$  obtained from procedures PRTION and CEASGN, tests whether a site  $q$  (assumed to be in  $R_{-1}$ ) is closed. The test consists of searching at most the first  $\text{LG}(n)$  layers of cells that surround  $q$  in the partition associated with  $P$  and  $A$  for sites  $s_{jm}$ ,  $j = 1, \dots, 8$ ,  $m = 1, \dots, 4$ , assigned to cells in these layers that render  $q$  closed. As soon as  $q$  is found to be closed  $\text{flag}$  is set equal to 1 and sites  $s_{jm}$ ,  $j = 1, \dots, 8$ ,  $m = 1, \dots, 4$ , that render  $q$  closed are placed in  $Q$ . Otherwise after  $\text{LG}(n)$  layers have been searched and  $q$  has not been found to be closed  $\text{flag}$  is set equal to zero.

POLYHD( $q, Q, V$ ): Given a set  $Q$  of sites and a site  $q$ , constructs the Voronoi polyhedron  $V$  of  $q$  relative to  $Q \cup \{q\}$  through an  $O(k \cdot \log k)$  worst-case algorithm for constructing the intersection of  $k$  half-spaces (see e. g. Preparata and Muller (1979)).

BNDIST( $q, V, d$ ): Computes the maximum distance  $d$ , possibly infinite, from a site  $q$  to the boundary of a polyhedron  $V$ .

SEARCH( $P, A, q, d, Q$ ): Using data structures in  $P$  and  $A$  obtained from PRTION and CEASGN procedures, given a site  $q$  searches layers of cells that surround  $q$  in the partition associated with  $P$  and  $A$  for sites assigned to cells in these layers within a distance  $d$  from  $q$ .  $Q$  will contain the sites found during this search.

VNEISV( $V, q, S, N$ ): Given a site  $q$  in a set  $S$  and a polyhedron  $V$  such that  $V$  is the Voronoi polyhedron of  $q$  relative to  $S$ , identifies from  $V$  those sites

in  $S$  that are Voronoi neighbors relative to  $S$  of  $q$ . On input  $N$  will contain for each site in  $S$  a list, possibly empty, of known Voronoi neighbors relative to  $S$  of the site obtained from previous executions of VNEISV. During the execution of VNEISV,  $N$  will be updated so that on output for each site that is a Voronoi neighbor relative to  $S$  of  $q$ ,  $q$  will appear in the list of known Voronoi neighbors relative to  $S$  of the site.

VNEIGT( $q, S, N, Q'$ ): Given a site  $q$  in a set  $S$ , and  $N$  as described for VNEISV, produces from  $N$  a set  $Q'$  that will contain the known Voronoi neighbors relative to  $S$  of  $q$ , if any, since the last execution of VNEISV.

SCTEST( $P, A, q, \text{LG}_k(n), H(f_i), \text{flag}, Q'', Q'''$ ): Using data structures in  $P$  and  $A$  obtained from procedures PRTION and CEASGN, tests whether a site  $q$  (assumed to be in  $R_k^i$ ) is semiclosed. The test consists of searching at most the first  $\text{LG}_k(n)$  layers of cells that surround  $q$  in the partition associated with  $P$  and  $A$  for sites  $q_j, s_j, j = 1, \dots, 8$ , assigned to cells in these layers that render  $q$  octant-closed and cone-semiclosed, respectively. As soon as  $q$  is found to be semiclosed  $\text{flag}$  is set equal to 1, sites  $s_j, j = 1, \dots, 8$ , that render  $q$  cone-semiclosed are placed in  $Q''$ , and points  $q'_j, j = 1, \dots, 8$ , are placed in  $Q'''$ ; where for each  $j, j = 1, \dots, 8$ ,  $q'_j$  is the intersection of  $q\bar{q}_j$  and  $H(f_i)$ , where  $q_j, j = 1, \dots, 8$ , are sites that render  $q$  octant-closed. Otherwise after  $\text{LG}_k(n)$  layers have been searched and  $q$  has not been found to be semiclosed  $\text{flag}$  is set equal to zero.

HALFSP( $q, H, C$ ): For a site  $q$  and a plane  $H, q \notin H$ , computes the closed half-space  $C$  that contains  $q$  and that is determined by the plane parallel to  $H$  that contains  $(T(q) + q)/2$ , where  $T(q)$  is the point in  $H$  that is the perpendicular projection of  $q$  onto  $H$ .

MAXDST( $q, Q''', d''$ ): Given a site  $q$ , and a finite set of points  $Q'''$ , computes the maximum distance  $d''$  between  $q$  and the points in  $Q'''$ .

MAXVAL( $d', d''$ ): Computes the maximum of two numbers  $d'$  and  $d''$ .

The outline of VORNOI follows. Here  $T$  is the output variable. For each  $h, h = 1, \dots, n$ , if in some ordering of  $S, q_h$  is the  $h^{\text{th}}$  site in  $S$  then  $T(q_h)$  will be the Voronoi polyhedron of  $q_h$  relative to  $S$ . All other arguments act as input variables and are as defined in the previous section.

```

procedure VORNOI( $S, R, n, \hat{k}, \text{LG}(n), \text{LG}_0(n), \dots, \text{LG}_{\hat{k}-2}(n),$ 
 $R_{-1}, H(f_1), \dots, H(f_6), R_0^1, \dots, R_0^6, \dots, R_{\hat{k}-2}^1, \dots, R_{\hat{k}-2}^6, T$ )
  begin

```

```

m := FLOOR( $n^{1/3}$ );
PRTION( $R, R, m, P$ );
CEASGN( $S, P, A$ );
RGASGN( $S, R_{-1}, S_{-1}, n_{-1}, B_{-1}$ );
for  $h := 1$  until  $n_{-1}$  do
  begin
     $q_h := B_{-1}(h)$ ;
    CLTEST( $P, A, q_h, LG(n), flag, Q$ );
    if ( $flag = 1$ ) then
      begin
        POLYHD( $q_h, Q, V$ );
        BNDIST( $q_h, V, d$ );
         $d := 2 \cdot d$ ;
        SEARCH( $P, A, q_h, d, Q$ )
      end
    else  $Q := S \setminus \{q_h\}$ 
      POLYHD( $q_h, Q, V$ );
      VNEISV( $V, q_h, S, N$ );
       $T(q_h) := V$ 
    end
  end
 $S' := S \setminus S_{-1}$ ;
 $R' := R \setminus R_{-1}$ ;
for  $k := 0$  until  $\hat{k} - 2$  do
  begin
     $m := FLOOR(2^{-k/2} \cdot n^{1/3})$ ;
    PRTION( $R, R', m, P$ );
    CEASGN( $S', P, A$ );
    for  $i := 1$  until 6 do
      begin
        RGASGN( $S', R_k^i, S_k^i, n_k^i, B_k^i$ );
        for  $h := 1$  until  $n_k^i$  do
          begin
             $q_h := B_k^i(h)$ ;
            VNEIGT( $q_h, S, N, Q'$ );
            SCTEST( $P, A, q_h, LG_k(n), H(f_i), flag, Q'', Q'''$ );
            if ( $flag = 1$ ) then
              begin
                 $Q := Q' \cup Q''$ ;
                POLYHD( $q_h, Q, V$ );
                HALFSP( $q_h, H(f_i), C$ );
                 $V := V \cap C$ ;
                BNDIST( $q_h, V, d'$ );
              end
            end
          end
        end
      end
    end
  end

```

```

      d' := 2 · d';
      MAXDST(qh, Q''', d'');
      d'' := √2 · d'';
      d := MAXVAL(d', d'');
      SEARCH(P, A, qh, d, Q);
      Q := Q ∪ Q'
    end
  else Q := (S' ∪ Q') \ {qh}
    POLYHD(qh, Q, V);
    VNEISV(V, qh, S, N);
    T(qh) := V
  end
end
S' := S' \ ∪i=16 Ski
end
R' = R' \ ∪i=16 ∪k=0k-2 Rki;
RGASGN(S', R', S', n', B');
for h := 1 until n' do
  begin
    qh := B'(h);
    VNEIGT(qh, S, N, Q');
    Q := (S' ∪ Q') \ {qh}
    POLYHD(qh, Q, V);
    VNEISV(V, qh, S, N);
    T(qh) := V
  end
end
end

```

#### 4. NUMERICAL RESULTS

A Fortran implementation of the algorithm has been developed on a Control Data Cyber 205 at the National Institute of Standards and Technology. Table 1 shows the computing time per site in CPU seconds for the implementation when applied to eight randomly generated sets in a cube for 30 values of  $n$ . Table 2 shows the number of 0-dimensional faces per site of the Voronoi diagrams that were obtained with the implementation for the same sets and values of  $n$ . We note that the numerical results in Table 1 and Table 2 seem to confirm our theoretical results. We note with interest from the results in Table 2 that the expected number of 0-dimensional faces per site of a 3-dimensional Voronoi diagram seems to be increasing very slowly as  $n$  increases but appears to be bounded above by the expected number of 0-dimensional faces per site of a 3-dimensional Poisson-Voronoi tessellation

(approximately 6.768) (see Miles (1970)). Finally, we note that in the implementation of the algorithm the constants  $c$ ,  $c'$ ,  $c''$  used in the definitions of Section 2 were all set equal to 1. However, the implementation has been written so that it functions essentially as if they had been set equal to those values that render the implementation the most efficient. For example, the implementation has been written so that procedure CLTEST is also executed for sites in  $R \setminus R_{-1}$  during the construction of their Voronoi polyhedra. Doing this is essentially equivalent to enlarging  $R_{-1}$  to a region that renders the implementation the most efficient which in turn is equivalent to setting  $c$  equal to that value that produces the same effect.

#### ACKNOWLEDGMENTS

The author thanks Irma Cardona-Colon and Damaris Santana-Morant for providing programming assistance in various phases of the implementation of the algorithm. He also wishes to express his gratitude to Jim Lawrence for his criticism and suggestions on the paper.

#### BIBLIOGRAPHY

Bentley J. L., Weide B. W. and Yao A. C. (1980), Optimal expected-time algorithms for closest point problems, *ACM Trans. Math. Software* (4) **6**, 563-580.

Bernal J. (1990), On the expected complexity of the 3-dimensional Voronoi diagram, NISTIR 4321, National Institute of Standards and Technology, Washington, D. C.

Bowyer A. (1981), Computing Dirichlet tessellations, *Comput. J.* (2) **24**, 162-166.

Brostow W., Dussault J-P. and Fox B. L. (1978), Construction of Voronoi polyhedra, *J. Comput. Phys.* **29**, 81-92.

Brown K. Q. (1979), Voronoi diagrams from convex hulls, *Inform. Process. Lett.* (5) **9**, 223-228.

Dwyer R. A. (1988), Higher-dimensional Voronoi diagrams in linear expected time, Tech. Rep. CMU-CS-88-100, Dept. of Comput. Science, Carnegie-Mellon Univ.

Finney J. L. (1979), A procedure for the construction of Voronoi polyhedra, *J. Comput. Phys.* **32**, 137-143.

- Green P. J. and Sibson R. (1978), Computing Dirichlet tessellations in the plane, *Comput. J.* (2) **21**, 168-173.
- Lee D. T. and Schachter B. J. (1980), Two algorithms for constructing a Delaunay triangulation, *Internat. J. Comput. Inform. Sci.* (3) **9**, 219-242.
- Maus A. (1984), Delaunay triangulation and the convex hull of  $n$  points in expected linear time, *BIT* **24**, 151-163.
- Miles R. E. (1970), On the homogeneous planar Poisson point process, *Math. Biosci.* **6**, 85-127.
- Ohya T., Iri M. and Murota K. (1984), A fast Voronoi-diagram algorithm with quaternary tree bucketing, *Inform. Process. Lett.* (4) **18**, 227-231.
- Preparata F. P. and Muller D. E. (1979), Finding the intersection of  $n$  half-spaces in time  $O(n \log n)$ , *Theoret. Comput. Sci.* **8**, 45-55.
- Seidel R. (1986), Constructing higher-dimensional convex hulls at logarithmic cost per face, in *Proc. 18<sup>th</sup> ACM Sympos. on Theory of Computing*.
- Shamos M. I. (1978), Computational geometry, Ph.D. dissertation, Yale University.
- Shamos M. I. and Hoey D. (1975), Closest-point problems, in *Proc. 16<sup>th</sup> Ann. IEEE Symp. on the Foundations of Comp. Sci.*, pp. 151-162.
- Tanemura M., Ogawa T. and Ogita N. (1983), A new algorithm for three-dimensional Voronoi tessellation, *J. Comput. Phys.* **51**, 191-207.
- Watson D. F. (1981), Computing the  $n$ -dimensional Delaunay tessellation with application to Voronoi polytopes, *Comput. J.* (2) **24**, 167-172.
- Witzgall C. J. (1973), Algorithms for Euclidean neighbor relations, Boeing Document D180-17573-1, Boeing Aerospace Company, Seattle, Washington.

$n$	<i>set 1</i>	<i>set 2</i>	<i>set 3</i>	<i>set 4</i>	<i>set 5</i>	<i>set 6</i>	<i>set 7</i>	<i>set 8</i>
$8^3$	0.2303	0.2452	0.2285	0.2457	0.2271	0.2275	0.2275	0.2049
$9^3$	0.2750	0.2770	0.2702	0.2835	0.2691	0.2751	0.2599	0.2524
$10^3$	0.3057	0.3175	0.2835	0.3177	0.3011	0.3034	0.2973	0.2876
$11^3$	0.3147	0.3243	0.2953	0.3158	0.3099	0.3098	0.3219	0.3218
$12^3$	0.3331	0.3347	0.3230	0.3347	0.3624	0.3028	0.3027	0.3262
$13^3$	0.3540	0.3393	0.3554	0.3558	0.3447	0.3663	0.3243	0.3324
$14^3$	0.3505	0.3543	0.3537	0.3739	0.3614	0.3904	0.3342	0.3481
$15^3$	0.3464	0.3569	0.3651	0.3622	0.3523	0.3773	0.3467	0.3592
$16^3$	0.3483	0.3550	0.3769	0.3636	0.3359	0.3760	0.3479	0.3650
$17^3$	0.3477	0.3523	0.3787	0.3608	0.3485	0.3736	0.3429	0.3654
$18^3$	0.3649	0.3569	0.3596	0.3711	0.3530	0.3528	0.3415	0.3650
$19^3$	0.3462	0.3521	0.3558	0.3586	0.3483	0.3447	0.3509	0.3473
$20^3$	0.3555	0.3437	0.3524	0.3623	0.3491	0.3378	0.3408	0.3481
$21^3$	0.3555	0.3544	0.3548	0.3531	0.3540	0.3430	0.3464	0.3480
$22^3$	0.3601	0.3553	0.3523	0.3447	0.3591	0.3388	0.3456	0.3432
$23^3$	0.3601	0.3526	0.3560	0.3438	0.3517	0.3297	0.3346	0.3432
$24^3$	0.3523	0.3466	0.3561	0.3409	0.3425	0.3290	0.3353	0.3442
$25^3$	0.3470	0.3391	0.3467	0.3346	0.3379	0.3206	0.3282	0.3368
$26^3$	0.3431	0.3430	0.3447	0.3270	0.3394	0.3105	0.3321	0.3308
$27^3$	0.3359	0.3428	0.3322	0.3306	0.3361	0.3189	0.3270	0.3263
$28^3$	0.3316	0.3377	0.3366	0.3225	0.3291	0.3150	0.3210	0.3272
$29^3$	0.3263	0.3339	0.3179	0.3173	0.3276	0.3116	0.3202	0.3208
$30^3$	0.3304	0.3224	0.3273	0.3184	0.3235	0.3107	0.3208	0.3220
$31^3$	0.3082	0.3243	0.3264	0.3153	0.3191	0.3156	0.3177	0.3188
$32^3$	0.3148	0.3139	0.3079	0.3030	0.3151	0.3259	0.3064	0.3226
$33^3$	0.2988	0.3300	0.3000	0.3096	0.2982	0.3036	0.3064	0.3065
$36^3$	0.3060	0.3040	0.3003	0.3102	0.2941	0.3037	0.2981	0.3072
$39^3$	0.2856	0.2871	0.2904	0.2953	0.2939	0.2908	0.2920	0.2957
$42^3$	0.2863	0.2813	0.2822	0.2891	0.2842	0.2866	0.2901	0.2774
$48^3$	0.2740	0.2664	0.2676	0.2759	0.2686	0.2654	0.2667	0.2648

Table 1: Computing time per site.

$n$	<i>set 1</i>	<i>set 2</i>	<i>set 3</i>	<i>set 4</i>	<i>set 5</i>	<i>set 6</i>	<i>set 7</i>	<i>set 8</i>
$8^3$	6.4043	6.3301	6.3496	6.3398	6.3789	6.4141	6.4141	6.3555
$9^3$	6.4595	6.3937	6.4033	6.4719	6.4225	6.4170	6.4115	6.4472
$10^3$	6.4710	6.4430	6.4610	6.4830	6.4610	6.4810	6.4440	6.5030
$11^3$	6.4936	6.5177	6.5297	6.5229	6.5289	6.5289	6.5177	6.5177
$12^3$	6.5758	6.5336	6.5706	6.5336	6.5156	6.5538	6.5538	6.5752
$13^3$	6.5772	6.5576	6.5544	6.5603	6.5872	6.5284	6.5690	6.5899
$14^3$	6.6148	6.5911	6.5652	6.6020	6.6148	6.6017	6.5860	6.6323
$15^3$	6.6216	6.6071	6.5609	6.6406	6.6308	6.6139	6.5961	6.6394
$16^3$	6.6394	6.5989	6.5972	6.6492	6.6316	6.6265	6.6445	6.6316
$17^3$	6.6585	6.6237	6.6076	6.6304	6.6381	6.6507	6.6640	6.6332
$18^3$	6.6408	6.6502	6.6487	6.6476	6.6626	6.6619	6.6493	6.6408
$19^3$	6.6553	6.6615	6.6545	6.6673	6.6602	6.6620	6.6606	6.6434
$20^3$	6.6583	6.6700	6.6524	6.6546	6.6621	6.6640	6.6610	6.6574
$21^3$	6.6622	6.6656	6.6628	6.6754	6.6758	6.6553	6.6634	6.6589
$22^3$	6.6767	6.6697	6.6595	6.6657	6.6802	6.6828	6.6665	6.6686
$23^3$	6.6869	6.6815	6.6615	6.6667	6.6763	6.6822	6.6694	6.6734
$24^3$	6.6811	6.6736	6.6748	6.6763	6.6778	6.6768	6.6782	6.6823
$25^3$	6.6797	6.6807	6.6799	6.6768	6.6733	6.6825	6.6845	6.6798
$26^3$	6.6758	6.6912	6.6880	6.6804	6.6912	6.6925	6.6907	6.6838
$27^3$	6.6815	6.6925	6.6803	6.6843	6.6888	6.6964	6.6901	6.6827
$28^3$	6.6824	6.6959	6.6893	6.6879	6.6969	6.7047	6.6959	6.6795
$29^3$	6.6953	6.6996	6.6899	6.6907	6.6975	6.7084	6.6962	6.6835
$30^3$	6.6991	6.7053	6.6976	6.6893	6.7071	6.7116	6.6984	6.6986
$31^3$	6.7004	6.7010	6.6961	6.7050	6.7007	6.7160	6.7073	6.6960
$32^3$	6.6978	6.7031	6.7079	6.7088	6.7041	6.7028	6.7031	6.7047
$33^3$	6.6955	6.7014	6.7113	6.7142	6.7017	6.7125	6.7148	6.7062
$36^3$	6.7185	6.7140	6.7090	6.7092	6.7166	6.7114	6.7128	6.7194
$39^3$	6.7193	6.7171	6.7188	6.7162	6.7144	6.7172	6.7164	6.7165
$42^3$	6.7160	6.7264	6.7205	6.7218	6.7226	6.7232	6.7219	6.7191
$48^3$	6.7262	6.7301	6.7221	6.7289	6.7303	6.7274	6.7244	6.7290

Table 2: Number of 0-dimensional faces per site.



# BIBLIOGRAPHIC DATA SHEET

1. PUBLICATION OR REPORT NUMBER NISTIR 4340
2. PERFORMING ORGANIZATION REPORT NUMBER JUNE 1990
3. PUBLICATION DATE

4. TITLE AND SUBTITLE

An Expected Linear 3-Dimensional Voronoi Diagram Algorithm

5. AUTHOR(S)

Javier Bernal

6. PERFORMING ORGANIZATION (IF JOINT OR OTHER THAN NIST, SEE INSTRUCTIONS)

U.S. DEPARTMENT OF COMMERCE  
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY  
GAITHERSBURG, MD 20899

7. CONTRACT/GRANT NUMBER

8. TYPE OF REPORT AND PERIOD COVERED

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS (STREET, CITY, STATE, ZIP)

10. SUPPLEMENTARY NOTES

DOCUMENT DESCRIBES A COMPUTER PROGRAM; SF-185, FIPS SOFTWARE SUMMARY, IS ATTACHED.

11. ABSTRACT (A 200-WORD OR LESS FACTUAL SUMMARY OF MOST SIGNIFICANT INFORMATION. IF DOCUMENT INCLUDES A SIGNIFICANT BIBLIOGRAPHY OR LITERATURE SURVEY, MENTION IT HERE.)

Let  $S$  be a set of  $n$  sites chosen independently from a uniform distribution in a cube in 3-dimensional Euclidean space. In this paper, an expected  $O(n)$  algorithm for constructing the Voronoi diagram for  $S$  together with numerical results obtained from an implementation of the algorithm are presented.

12. KEY WORDS (6 TO 12 ENTRIES; ALPHABETICAL ORDER; CAPITALIZE ONLY PROPER NAMES; AND SEPARATE KEY WORDS BY SEMICOLONS)

computational geometry; expected complexity; expected time analysis; linear algorithm; Poisson-Voronoi tessellation; Voronoi diagram

13. AVAILABILITY

<input checked="" type="checkbox"/>	UNLIMITED
<input type="checkbox"/>	FOR OFFICIAL DISTRIBUTION. DO NOT RELEASE TO NATIONAL TECHNICAL INFORMATION SERVICE (NTIS).
<input type="checkbox"/>	ORDER FROM SUPERINTENDENT OF DOCUMENTS, U.S. GOVERNMENT PRINTING OFFICE, WASHINGTON, DC 20402.
<input checked="" type="checkbox"/>	ORDER FROM NATIONAL TECHNICAL INFORMATION SERVICE (NTIS), SPRINGFIELD, VA 22161.

14. NUMBER OF PRINTED PAGES

16

15. PRICE

A02





