数理解析研究所講究録
第872巻 1994年 65-78

Discrete Geometry and Davenport-Schinzel Sequence

Keiko Imai

Department of Information and System Engineering,
Faculty of Science and Engineering, Chuo University

Abstract In computational geometry, a lot of knowledge in combinatorial geometry is required in order to analyze the combinatorial structure of geometric problems and algorithms presented to solve the geometric problems. Conversely, some recent problems posed in computational geometry give a new direction to the study of discrete and combinatorial geometry. One of such subjects is the Davenport-Schinzel sequence. Although the Davenport-Schinzel sequence is a combinatorial concept, it is important in computational geometry because theres are many algorithms in computational geometry which can be analyzed by the theory of Davenport-Schinzel sequences. Davenport-Schinzel sequences are strongly connected with the lower envelope of a set of functions. In this paper we survey the theory of Davenport-Schinzel sequences and how to apply the theory to geometric problems.

1. Introduction

The lower envelope of a set of n functions is an important concept in computational geometry, and particularly, there are its many applications to motion planning, visibility, ray shooting and geometric optimization problems. For example, consider the nearest neighbor problem of moving points $p_1(t)$, $\cdots$, $p_n(t)$ with parameter t in the plane. The problem is to compute the sequence of points that are nearest to some certain point, say p_1. The sequence is called the nearest neighbor sequence. The problem of computing the nearest neighbor sequence can be formulated as the following lower envelope problem:

$$f(t) = \min_{i \neq 1} d(p_1(t), p_i(t)),$$

where $d(p_1(t), p_i(t))$ is the Euclidean distance between $p_1(t)$ and $p_i(t)$. Note that the graph $y = f(t)$ is the lower envelope of graphs of n functions $y = d(p_1(t), p_i(t))$. For functions of a single variable, the combinatorial and algorithmic analysis of the lower envelope has been studied as a Davenport-Shinzel sequence.

For positive integers n and s, a sequence of integers $U = (u_1, u_2, \cdots, u_m)$ is an (n, s) *Davenport-Schinzel sequence* (an (n, s) DS sequence for short), if it satisfies the following conditions:

(1) $1 \leq u_i \leq n \quad (i = 1, \cdots, m)$

(2) $u_i \neq u_{i+1} \quad (i = 1, \cdots, m-1)$

(3) There do not exist $s+2$ indices $1 \leq i_1 < i_2 < \cdots i_{s+2} \leq m$ such that $u_{i_1} = u_{i_3} = u_{i_5} = \cdots = a$, $u_{i_2} = u_{i_4} = u_{i_6} = \cdots = b$ and $a \neq b$.

We refer to s as the *order* of the sequence U and $|U| = m$ is called the *length* of the sequence U. For example, for $n = 5$, $(1,2,3,4,5)$ is $(5,1)$ DS sequence of length 5 and $(1,2,3,4,5,4,3,2,1)$ is $(5,2)$ DS sequence of length 9. $\lambda_s(n)$ is the maximal length of (n,s) Davenport-Schinzel sequence:

$$\lambda_s(n) = \max\{|U| : U \text{ is an } (n,s) \text{ DS sequence}\}.$$

At a first glance, it seems that this definition of the DS sequence is artificial in a sense. However, the DS sequences provide a combinatorial characterization of the lower envelope of the set of continuous functions. Therefore DS sequences have been studied as an important concept in aspects of applications in computational and combinatorial geometry, and have produced satisfactory results in computational geometry.

This paper surveys the theory of DS sequences and its applications in computational geometry. In the next section, the strong connection between the DS sequence and the lower envelope is reviewed and we summarize the known facts of the DS sequence and the lower envelope. Sections 3 and 4 deal with some applications of DS sequence. In the section 3, dynamic Voronoi diagram and generalized dynamic Voronoi diagrams can be analyzed by using the DS sequence and applications of these dynamic diagrams to geometric fitting problems are also presented. In the section 4, some other applications to motion planning and geometric optimization problem are touched upon.

2. Lower Envelope of n functions and Davenport-Schinzel Sequences

Let $\Gamma = \{f_1, f_2, \cdots, f_n\}$ be a set of n continuous functions of a single variable x, where each f_i is defined over the reals. Assume that each pair of the functions in Γ intersect in at most s points. For the set of the functions Γ, define the lower envelope $f(x)$ by

$$f(x) = \min_i f_i(x).$$

The graph of $f(x)$ consists of connected portions of the graphs of the functions f_i. Consider the indices sequence of the lower envelope f, and we call it the *lower envelope sequence*. The sequence is nothing but an (n,s) DS sequence, thus the number of connected components of $f(x)$ is at most $\lambda_s(n)$. Moreover, it is known that, for any (n,s) DS sequence U, there is a set of functions Γ such that its lower envelope sequence

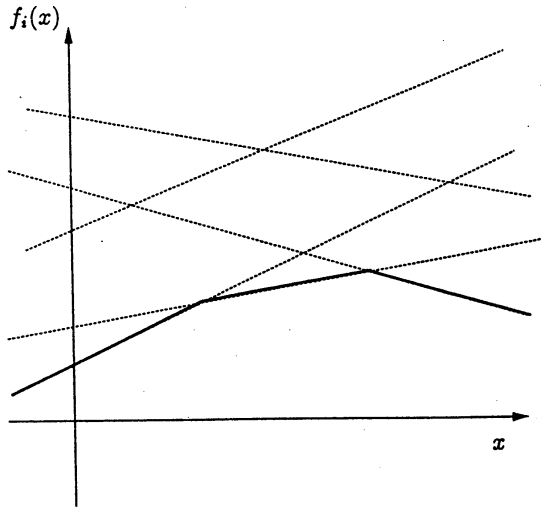

Fig.1 Lower envelope of n lines: $\lambda_1(n) = n$

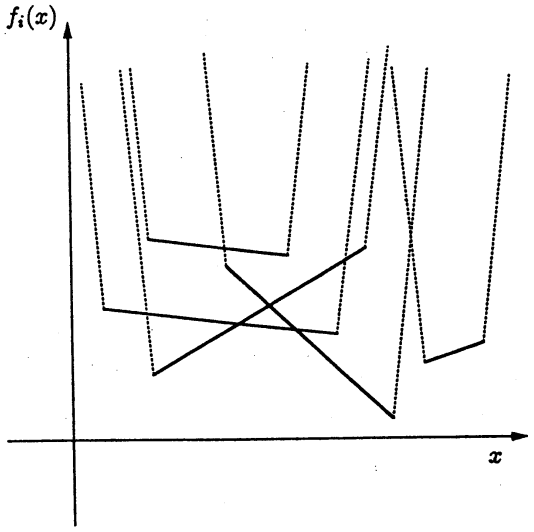

Fig.2 Lower envelope of n segments: $\lambda_3(n) = \Theta(n\alpha(n))$

is U. Therefore the maximal length of lower envelope sequences is equal to the maximal length of (n, s) Davenport-Schinzel sequences $\lambda_s(n)$.

Thus, estimating the maximal length of DS sequences is equivalent to computing the lower envelope sequence of a set of functions which intersect each other in at most some fixed number of points. We will trace briefly the history of analyzing DS sequences before giving examples of applications of DS sequences.

The problem of estimating $\lambda_s(n)$ has originally been posed by Davenport and Schinzel [6]. They were interested in a linear differential equation with constant coefficients of order d. Let $f_1(x)$, $\cdots$, $f_n(x)$ be n distinct solutions of the equation. They considered the upper envelope of the solutions and reduced the problem of estimating the number of connected portions of the upper envelope to the combinatorial problem of DS sequences.

It is easy to show that $\lambda_1(n) = n$ (See Fig.1) and $\lambda_2(n) = 2n - 1$. For $\lambda_3(n)$, Davenport and Schinzel showed that $\lambda_3(n) = O(n \log n)$ [6], and it was improved to $O\left(n \frac{\log n}{\log \log n}\right)$ by Davenport [5]. For higher order DS sequences, the first non-trivial upper bound was $\lambda_s(n) = O(n \exp\{C\sqrt{\log n}\})$ [6], where C depends on only on s, and it was improved to $O(n \log^* n)$ by Szemerédi [20], where the $\log^* n$ is the smallest $i \geq 1$ for which $k_i \geq n$, where $k_1 = 2$ and $k_{i+1} = 2^{k_i}$ $(i \geq 1)$. Attallah raised the DS sequence again in dynamic computational geometry and generalized the lower envelope problem for n partially-defined functions [3] as follows: Let f_1, $\cdots$, f_n be partially defined functions, such that each function f_i is defined and continuous over

some connected interval I_i, and such that each pair of them intersect in at most s points. In this general case, the lower envelope consists of at most $\lambda_{s+2}(n)$ connected portions [3]. This result shows that the combinatorial complexity of the lower envelope of n line segments in the plane is $O(\lambda_3(n))$ (Fig.2). After the rediscovery, the problem of estimating $\lambda_s(n)$ has been one of the main topics in computational geometry. Hart and Sharir proved that $\lambda_3(n) = \Theta(n\alpha(n))$ [7], where $\alpha(n)$ is the functional inverse of Ackermann's function. The function $\alpha(n)$ grows very slowly and tends to infinity with n, thus it was shown that DS sequences are non-linear for $s \geq 3$. Their proof was based on an equivalence between DS sequences with $s = 3$ and sequences of certain operations performed on arbitrary rooted trees, called generalized path compressions. The path compression has complexity $\Theta(n\alpha(n))$ [21]. Soon after this paper, Sharir proved that $\lambda_s(n) = O(n\alpha(n)^{O(\alpha(n)^{s-3})})$ for $s \geq 4$ [16] and $\lambda_{2s+1}(n) = \Omega(n(\alpha(n))^s)$ for $s \geq 2$ [17]. The facts that are known until now are as follows (See [1]):

$$\lambda_1(n) = n,$$

$$\lambda_2(n) = 2n - 1,$$

$$\lambda_3(n) = \Theta(n\alpha(n)),$$

$$\lambda_4(n) = \Theta(n2^{\alpha(n)}),$$

$$\lambda_{2s}(n) = O(n2^{(\alpha(n))^{s-1}(1+o(1))}) \quad \text{for } s > 2,$$

$$\lambda_{2s+1}(n) = O(n2^{(\alpha(n))^{s-1}(1+o(1))\log\alpha(n)}) \quad \text{for } s \geq 2,$$

$$\lambda_{2s}(n) = \Omega(n2^{\frac{1}{(s-1)!}(\alpha(n))^{s-1}}) \quad \text{for } s > 2,$$

$$\lambda_{2s+1}(n) = \Omega(n(\alpha(n))^s) \quad \text{for } s \geq 2.$$

These results show that $\lambda_s(n)$ $(s \geq 3)$ is slightly super-linear in n for any fixed s.

Consider the problem of constructing the lower envelope of n continuous functions over the reals such that each pair of them intersect in at most s points. The algorithm is based on a straightforward application of the divide-and-conquer paradigm.

Step 1. Partition the functions into two sets of equal size.

Step 2. Find the lower envelope of each set recursively.

Step 3. Merge the two envelopes.

The algorithm produces the merge of the two envelopes in time proportional to the number of intervals in which one of the n functions attains the envelope. The algorithm computes the lower envelope in $O(n\lambda_s(n)\log n)$ time, and it is easy to apply to the case of n partially defined functions. In that case, the lower envelope of n partially

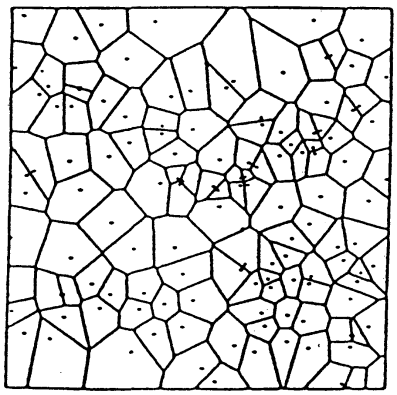

Fig.3. Voronoi diagram

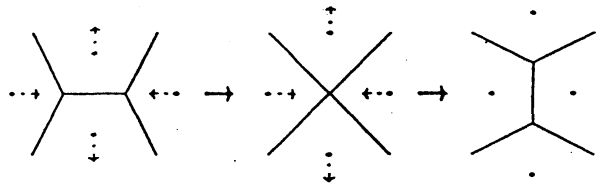

Fig.4. Topological change of Voronoi diagram

defined functions has $O(n\lambda_{s+2}(n))$ connected portions, thus an $O(n\lambda_{s+2}(n)\log n)$ time algorithm can be obtained. Moreover, Hershberger improved the time complexity to $O(n\lambda_{s+1}(n)\log n)$ time [8].

Consider the nearest neighbor sequence problem for moving points along lines in the plane. For i, let $f_i(t)$ be $d(p_1(t),p_i(t))^2$, and each pair of $f_i(t)$ intersect at most twice. The function $\tilde{f}(t) = \min_{i\neq 1} f_i(t)$ can be considered instead of $f(t) = \min_{i\neq 1} d(p_1(t),p_i(t))$. By using results of the DS sequence, it can be shown that the maximal length of the nearest neighbor sequence for moving points along lines in the plane is $O(n)$ and we obtain the sequence in $O(n\log n)$ time.

3. Voronoi Diagrams for Moving Points

In this section, we consider the Voronoi diagram for moving points. The problems of estimating the combinatorial complexity of dynamic Voronoi diagrams and analyzing algorithms for constructing them are deeply related to the problem of the lower envelope problem and DS sequences. We studied this problem in [9,10].

The most fundamental Voronoi diagram is the Euclidean Voronoi diagram for n points in the plane. For a set of n points p_i $(i = 1,\cdots,n)$ in the plane, define the Voronoi region $V(p_i)$ of p_i by

$$V(p_i) = \bigcap_j \{p \mid d(p,p_i) < d(p,p_j)\},$$

where $d(p,p_i)$ denotes the Euclidean distance between two points p and q in the plane. The boundaries of $V(p_i)$ $(i = 1,\cdots,n)$ form a planar skeleton, which is called the Voronoi diagram for n points (Fig.3).

Recently, the Voronoi diagram for moving objects has been shown to be useful in motion planning and geometric optimization problems. Figure 4 illustrates how the

Voronoi diagram for points changes when the points move dynamically. As shown in this figure, the topology of the Voronoi diagram changes according to the movement. The dynamic Voronoi diagram represents such topological changes.

Consider n points $p_i(t) = (x_i(t), y_i(t))$ parametrized by t in the plane (t may be sometimes regarded as time), where $x_i(t)$, $y_i(t)$ are functions of t , which are polynomials or rational functions of t . The degrees of these polynomials and rational functions are assumed to be independent of n . To simplify our discussion, it is also assumed that these functions are different from one another, $p_i(t) \neq p_j(t)$ for any t and $i \neq j$ and any four points become cocircular for at most s distince values of t . s is independent of n . For fixed t , the Voronoi diagram for n points $p_i(t)$ is defined as usual, and the problem is to construct the Voronoi diagrams for some range of t .

As is well known, the Voronoi diagram for $p_i(t)$ for fixed t is the orthogonal projection of the lower envelope of functions $g_i(x, y) = ((x - x_i(t))^2 + (y - y_i(t))^2$ of two variables x , y $(i = 1, \cdots, n)$. In a similar way, the problem of constructing the dynamic diagram may be regarded as computing the lower envelope of n functions

$$f_i(x, y, t) = (x - x_i(t))^2 + (y - y_i(t))^2$$

of three variables x , y , t $(i = 1, \cdots, n)$. Define $f(x, y, t)$ by

$$f(x, y, t) = \min_{i=1,\cdots,n} f_i(x, y, t).$$

Now the problem is to compute $f(x, y, t)$.

For this function $f(x, y, t)$, the *minimum diagram* is a subdivision of (x, y, t) -space such that, with each region, a function f_i attaining the minimum in the definition of f for any point in the region is associated. This is nothing but the projection of the pointwise minimum of these functions onto (x, y, t) -space. The intersection of this diagram with the plane $t = t'$ is the Voronoi diagram for $p_i(t')$. We call this minimum diagram the *dynamic Voronoi diagram.*

3.1. Analysis of the Dynamic Voronoi Diagram

By the assumptions on $p_i(t)$, each region of the minimum diagram of f consists of a maximal connected 3-dimensional set of points at which the minimum is attained by a function f_i . The faces, edges and vertices of the subdivision consist of points at which the minimum is attained simultaneously by two, three and four, respectively, functions.

It is easy to see the following properties of the intersections of the trivariate functions $f_i(x, y, t)$ by solving the simultaneous equations each of which defines a function

$f_i(x,y,t)$ in 4-dimensional Euclidean space $\mathbf{E}^4$. In the sequel, variables x, y, t of f_i will be often omitted.

Lemma 1. (1) $f_i(x,y,t) = f_j(x,y,t)$ is a connected surface in $\mathbf{E}^4$ for each $i \neq j$.
(2) For each triple i, j and k of distinct indices, if $p_i(t)$, $p_j(t)$ and $p_k(t)$ are not collinear for all t, $f_i = f_j = f_k$ is a continuous curve with parameter t. When there exists a t such that $p_i(t)$, $p_j(t)$ and $p_k(t)$ are collinear, the curve is discontinuous at this t. The curve may be discontinuous at a constant number of points.
(3) For four distinct indices i, j, k and l, $f_i = f_j = f_k = f_l$ consists of at most s points. □

A trivial bound on the number of vertices on this minimum diagram is $O(n^4)$, but this is loose as shown below.

Let i and j be distinct two indices, and fix them. For each $k \neq i,j$, we define new functions $g_k^+(t)$ and $g_k^-(t)$ as follows. Let q_k be the center of the circumscribed circle of $p_i(t)$, $p_j(t)$ and $p_k(t)$, and d_k the distance between q_k and the middle point of $p_i(t)$ and $p_j(t)$. Let $l_{ij}(t)$ be the oriented line passing $p_i(t)$ and $p_j(t)$ in this order. See Fig.5.

Case 1: $p_k(t)$ is on the right side of the oriented line $l_{ij}(t)$.

$$g_k^+(t) = \begin{cases} +d_k & (q_k \text{ is on the right side of } l_{ij}(t)) \\ -d_k & (q_k \text{ is on the left side of } l_{ij}(t)) \end{cases}$$

$$g_k^-(t) = +\infty$$

Case 2: $p_k(t)$ is on the left side of the oriented line $l_{ij}(t)$.

$$g_k^+(t) = +\infty$$

$$g_k^-(t) = \begin{cases} -d_k & (q_k \text{ is on the right side of } l_{ij}(t)) \\ +d_k & (q_k \text{ is on the left side of } l_{ij}(t)) \end{cases}$$

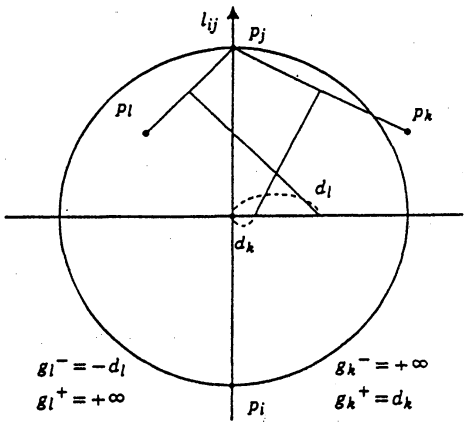

Fig.5. Definitions of g_k^+ and g_k^-

Case 3: $p_k(t)$ is on $l_{ij}(t)$, where this case occurs for at most a constant number of t. If $p_k(t)$ is on the line segment connecting $p_i(t)$ and $p_j(t)$, we define $g_k^+(t) = g_k^-(t) = -\infty$. Otherwise, $g_k^+(t) = g_k^-(t) = +\infty$.

For $g_k^+(t)$, $g_k^-(t)$, further define functions $g^+(t)$, $g^-(t)$ and $g'(t)$ as follows:

$$g^+(t) = \min_{k \neq i,j} g_k^+(t), \qquad g^-(t) = \min_{k \neq i,j} g_k^-(t),$$

$$g'(t) = \max\{g^-(t) + g^+(t),\ 0\}.$$

Geometric implications of these definitions are given as follows.

Lemma 2. (1) For t, suppose $g^+(t) \neq \pm\infty$ (resp. $g^-(t) \neq \pm\infty$) is attained by a function $g_k^+(t)$ (resp. $g_k^-(t)$). Then, any point $p_l(t)$ lying on the right (resp. left) side of $l_{ij}(t)$ is not contained inside the circumscribed circle of $p_i(t)$, $p_j(t)$ and $p_k(t)$.
(2) For t, suppose $g'(t) > 0$. If $g^+(t) = g_k^+(t) \neq +\infty$ or $g^-(t) = g_k^-(t) \neq +\infty$, any point $p_l(t)$ is not contained inside the circumscribed circle of $p_i(t)$, $p_j(t)$ and $p_k(t)$. □

Graphs of g^+, g^- and g' are composed of maximal connected portions of graphs of g_k^+, g_k^-, $g_k^+ + g_l^-$ and 0. As usual, define the combinatorial complexity of these functions to be the maximum number of such maximal connected portions. Also, call t' *an intersecting value* of a function (g^+, g^-, g') if the functions (g_k^+, g_k^-, $g_k^+ + g_l^-$, 0) attaining its minimum or maximum at $t'-t_\epsilon$ and $t'+t_\epsilon$ are different for sufficiently small t_ϵ. The number of intersecting values is nearly equal to the combinatorial complexity. This complexity can be evaluated as follows.

Lemma 3. The combinatorial complexity of g^+, g^- and g' are $O(\lambda_{s+2}(n))$. These functions can be computed in $O(\lambda_{s+1}(n)\log n)$ time.

Proof: Each g_k^+ may be discontinuous at most a constant number of times from Lemma 1(2). Any two functions among g_k^+ intersect at most s points by Lemma 1(3). Hence, the combinatorial complexity of g^+ is $O(\lambda_{s+2}(n))$ [3]. For g^-, similar. Any two functions among g_k^+, g_k^-, $g_k^+ + g_l^-$ and 0 intersect at most constant times. Then, the combinatorial complexity of g' is within that of g^+ and g^- by a constant factor. The time complexity follows from [8]. □

Theorem 1. The dynamic Voronoi diagram has the combinatorial complexity of $O(n^2 \lambda_{s+2}(n))$, and can be computed in $O(n^2\lambda_{s+1}(n)\log n)$ time and $O(n)$ space.

Proof: Suppose that one of the intersections, whose number is at most s, of f_i, f_j, f_k and f_l in $\mathbf{E}^4$ corresponds to a vertex on the minimum diagram of f in $\mathbf{E}^3$, and t-coordinate of this point is t'. There exists a circle on which $p_i(t')$, $p_j(t')$, $p_k(t')$ and $p_l(t')$ lie, and, for the other indices h, $p_h(t')$ is not enclosed in the circle. Considering configurations of four points $p_i(t')$, $p_j(t')$, $p_k(t')$ and $p_l(t')$, there exist three cases to consider:

(a) $p_k(t')$ and $p_l(t')$ are on the right side of the oriented line $l_{ij}(t')$;

(b) $p_k(t')$ and $p_l(t')$ are on the left side of the oriented line $l_{ij}(t')$;

(c) one of $p_k(t')$ and $p_l(t')$ is on the right side of $l_{ij}(t')$ and the other on the left side.

In the cases (a), (b) and (c), t' is the intersecting value of g^+, g^- and g' (in fact, in (c), it is a solution of the equation $g^+ + g^- = 0$), respectively. Therefore, a vertex on the minimum diagram and the surface defined by $f_i = f_j$ correspond to an intersecting value of g^+, g^- or the solution of $g^+ + g^- = 0$. Conversely, each of such intersecting values corresponds to a unique vertex in the minimum diagram.

Hence, the combinatorial complexity of the dynamic Voronoi diagram is $O(n^2 \lambda_{s+2}(n))$. By computing g^+, g^-, g' for any pair of $p_i(t)$ and $p_j(t)$, all the vertices on the dynamic Voronoi diagram can be listed in $O(n^2 \lambda_{s+1}(n) \log n)$ time and $O(n)$ space (Lemma 3). □

The lower envelope of functions of a single variable has been investigated as a DS sequence, and it is easy to see that the combinatorial complexity of the lower envelope of general algebraic surfaces in the d dimensional space is $O(n^d)$ and its lower bound is $\Omega(n^{d-1})$. Howerer, it has been conjectured that the combinatorial complexity of the lower envelope is only slightly larger than $O(n^{d-1})$. The dynamic Voronoi diagram for moving points is regraded as the lower envelope of n functions $f_i(x, y, t)$ of three variables, and our bounds of the dynamic Voronoi diagram are tight within $\log^* n$ factor, which beats bounds n^ϵ for any $\epsilon > 0$. Sharir surveyed the maximum combinatorial complexity of the lower envelope of general algebraic surfaces in [18], and improved the upper bound $O(n^d)$ to an almost tight upper bound for the complexity. More precisely, the paper presented that the combinatorial complexity of the lower envelope of n low-degree algebraic surfaces in the d dimensional space is $O(n^{d-1+\epsilon})$ for any $\epsilon > 0$, with the constant of proportionality depending on ϵ, on d, on the maximum number s of intersection points between any d-tuple of surfaces and also on the degree and shape of these surfaces and of their boundaries.

3.2. Generalized Dynamic Voronoi Diagrams

The Euclidean Voronoi diagram for fixed points in the plane is so useful in solving many geometric problems, for instance, the Delaunay triangulation, the minimum spanning tree problem, the largest empty circle problem, etc. Therefore, many generalizations have been performed, and the Voronoi diagrams based on the L_1 distance, L_∞ distance, etc., and further the Voronoi diagram for more complex objects such as segments, polygons, disks, etc., have been studied.

We applied the same technique presented in the section 3.1 to the higher-order and weighted Voronoi diagrams for moving points in the plane, and the dynamic Voronoi diagram for n circles in the Euclidean geometry and in the Laguerre geometry. The combinatorial complexity of the dynamic Voronoi diagram for moving circles in Euclidean and Laguerre geometry is $O(n^2\lambda_{s+2}(n))$ [12], and the combinatorial complexity for the dynamic m-th Voronoi diagram is $O(n^2 m\lambda_{s+m+3}(n))$ [11]. For constructing dynamic Voronoi Diagrams, we obtain an $O(n^2\lambda_{s+1}(n)\log n)$ time algorithm for moving circles in Euclidean and Laguerre geometry [12], and an $O(n^2 m\lambda_{s+m+2}(n)\log n)$ algorithm for the dynamic m-th Voronoi diagram [11]. Moreover, for the dynamic weighted Voronoi diagram, the combinatorial complexity is $O(n^2\lambda_{s+2}(n))$, and an $O(n^2\lambda_{s+1}(n)\log n)$ time algorithm for constructing it is obtained [13].

For the Voronoi diagram based on L_1 distance, Chew showed that the number of topological changes as the points move with constant velocity in the plane is $O(n^2\alpha(n))$ [4].

3.3. Related Minimax Geometric Fitting Problems

There are some applications of the non-weighted or weighted dynamic furthest Voronoi diagram to the geometric fitting problem between two sets of points. Although the furthest Voronoi diagram is different from the nearest one, the above arguments hold with a slight modification. The geometric fitting problem is a fundamental problem in pattern recognition and image processing. For example, it arises in an industrial robot attaching a pin-grid-array type LSI to a board by using visual sensors [14]. The robot tries to fit the pins of LSI package to corresponding patterns on the board. The patterns are a collection of disks or squares of the same size.

The non-weighted problem can be formulated as follows. Given two sets $S = \{s_j = (x_j, y_j) \mid j = 1, \cdots, n\}$ and $T = \{t_j = (u_j, v_j) \mid j = 1, \cdots, n\}$ of points in the plane such that s_j is associated with t_j, translate, rotate (or transform in a more complicated way) and/or scale the set S simultaneously so that the maximum of the L_2 (or L_∞) distances, according to each pattern, between t_j and the transformed s_j is minimized. For example, in the case that the patterns are disks, and translation and rotation are used as geometric operations, the problem is expressed as follows:

$$\min_{z, 0\le\theta<2\pi} \max_{j=1,\cdots,n} \|s_j e^{i\theta} - t_j - z\| ,$$

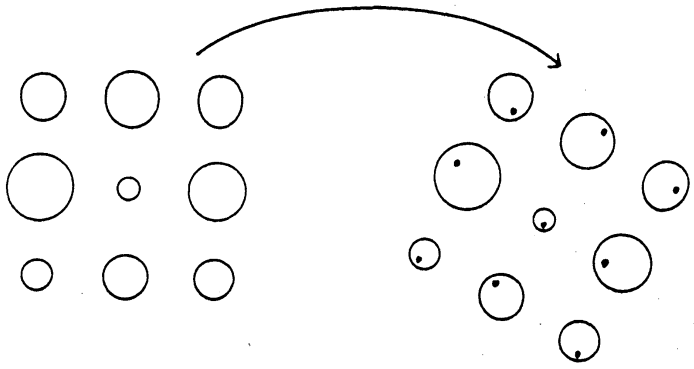

Fig. 6. Geometric fitting of distorted grid points with disks with different radii

where s_j and t_j are identified with complex numbers x_i+iy_i and u_i+iv_i, respectively, and θ is an angle $(0 \leq \theta < 2\pi)$ and the S is translated by making the origin to $z = x+iy$. $\|\cdot\|$ denotes the Euclidean norm. Using the furthest Voronoi diagram for moving points in the plane, this geometric fitting problem can be solved in $O(n^2\lambda_7(n)\log n)$ time in L_2 norm.

In a more general setting, the following variants of the fitting problem should be considered. (a) In the case that there are some points which cannot be put into the corresponding disks, minimize the number of such points. (b) In the case that the radii of disk patterns are different from one another, solve this non-uniform geometric fitting problem (Fig.6). The weighted Voronoi diagram for moving points can be used to solve the problem (b), since the radii of the disks may be considered as weights of their centers. Suppose that point s_j should be put into disk with center t_j and radius r_j in the pattern. By rotating the set S of points by θ and further translating it by z, s_j is the corresponding disk iff $\|s_j e^{i\theta} - t_j - z\| \leq r_j$. Then, all the points in S can be put into the corresponding disks iff the following value

$$\min_{z,0\leq\theta<2\pi} \max_{j=1,\cdots,n} \frac{1}{r_j}\|s_j e^{i\theta} - t_j - z\|$$

is less than or equal to 1.

4. Other Applications

In this section, we introduce some geometric problems and describe the results analyzed by DS sequences. Three applications to motion Planning are given by Sharir et al. [19], and Kedem et al. discussed the technical problem arising in such motion planning algorithms for a convex polygon in the plane [15]. The last one is the geometric optimization problems, which has application in geographical information processing [2].

4.1. Problem of Separating Two Simple Polygons [19]

Let P and Q be two disjoint simple polygons having m and n edges, respectively. The problem is to determine whether P and Q can be separated from one another by a sequence of translations, and, if so, to produce such a separating motion. Consider the Minkowski difference

$$K = P - Q = \{x - y \mid x \in P, y \in Q\}.$$

It is easy to see that P and Q can be separated by translations if and only if the origin O lies in the unbounded connected component C_∞ of the complement of K. In the worst case, the boundary of C_∞ may consist of $\Omega(mn)$. Sharir et al. obtained that their algorithm runs in $O(mn\ \alpha(mn) \log m \log n)$ time, which is close to a worst case lower bound on the number of translations needed to separate Q from P.

4.2. Polygon Containment Problem [19]

Let P be a convex polygon with k edges, and let Q be a closed polygonal region whose boundary consists of n edges. The problem is to determine whether P can be translated and rotated to a placement in which it lies inside Q. For the case in which P is convex but Q is an arbitrary polygonal region, Sharir et al. presented that there exist at most $O(kn\lambda_6(kn))$ free placements of P and an $O(kn\lambda_6(kn) \log n)$ time algorithm was given.

4.3. Collision-Free Motion of a Convex Polygon Moving Amidst Polygonal Obstacles [19]

Let P be a moving convex k-gon and Q a polygonal region in which P is free to move and whose boundary consists n edges. F is the three dimensional free configuration space, and the problem is to obtain a combinatorial representation of the boundary of F. Sharir et al. have done this by constructing an edge graph whose nodes are the one dimensional edges on the boundary of F. They showed that this edge graph has $O(kn\lambda_6(kn))$ vertices and edges, and that it can be computed in $O(kn\lambda_6(kn) \log kn)$ time.

4.4. Maximin Placement of Convex Objects in a Polygon [2]

The maximin placement of a convex polygon P inside a polygon Q is to place P inside Q, using translation and rotation, so that the minimum Euclidean distance between any point on P and any point on Q is maximized. For a fixed angle of the rotation, the feasible region of P inside Q is defined to be a set of points u such

that $P(u)$ is contained in Q. As the angle changes, the boundary of the feasible region changes. Using the dynamic Voronoi diagram for moving edges of the boundary of the feasible region, this problem can be solved in $O(m^4 n\lambda_{16}(mn)\log mn)$ time, where m and n are the numbers of edges of P and Q, respectively.

5. Conclusions

In this paper, we have described the connection between DS sequences and the lower envelope of a set of functions, illustrated how to apply the DS sequence to analyzing the geometric algorithms by taking the dynamic Voronoi diagram as an example, and reviewed some other applications of the DS sequence to motion planning and geometric optimization problems.

Investigations in computational geometry and in combinatorial geometry have been inseparable and will progress with interacting each other from now on. The DS sequence is one of the examples which show the strong connection between two research areas.

References

[1] P. Agarwal, M. Sharir and P. Shor: Sharp Upper and Lower Bounds on the Length of General Davenport-Schinzel Sequences. *Journal of Combinatorial Theory, Series A*, Vol. 52 (1989), pp.228–274.

[2] H. Aonuma, H. Imai, K. Imai and T. Tokuyama: Maximin Location of Convex Objects in a Polygon and Related Dynamic Voronoi Diagrams. *Proceedings of the 6th Annual ACM Symposium on Computational Geometry*, 1990, pp.225–234.

[3] M. J. Atallah: Some Dynamic Computational Geometry Problems. *Computers and Mathematics with Applications*, Vol.11 (1985), pp.1171–1181.

[4] L. P. Chew: Near-Quadratic Bounds for the L_1 Voronoi Diagram of Moving Points. *Proceedings of the 5th Canadian Conference on Computational Geometry*, 1993, pp.364–369.

[5] H. Davenport: A Combinatorial Problem Connected with Differential Equations II, *Acta Arithmetica*, Vol.17 (1971), pp.363–372.

[6] H. Davenport and A. Schinzel: A Combinatorial Problem Connected with Differential Equations, *American Journal of Mathematics*, Vol.87 (1965), pp.684–689.

[7] S. Hart and M. Sharir: Nonlinearity of Davenport-Schinzel Sequences and of Generalized Path Compression Schemes, *Combinatorica*, Vol.6, No.2 (1986), pp.151–177.

[8] J. Hershberger: Finding the Upper Envelope of n Line Segments in $O(n \log n)$ Time. Information Processing Letters.

[9] H. Imai and K. Imai: Dynamic Voronoi Diagrams for moving objects. *Proceedings of International Computer Symposium*, 1990, pp.600–606.

[10] K. Imai: Voronoi Diagrams for Moving Points and its Applications (in Japanese). *Transactions of the Japan Society for Industrial and Applied Mathematics*, Vol.1, No.2 (1991), pp.127–134.

[11] K. Imai and H. Imai: Higher-Order Dynamic Voronoi Diagrams and it Applications (in Japanese). *Memoirs of Mathematical Research Institute*, Kyoto University, Vol.790, 1992, pp.222–228.

[12] K. Imai and H. Imai: On Weighted and Higher-Order Dynamic Voronoi Diagrams (in Japanese). *Thechnical Notes of IPSJ*, AL–29–4 (1992), pp.25–32.

[13] K. Imai and H. Imai: On Weighted Dynamic Voronoi Diagrams. *Proceedings of the 6th Karuizawa Workshop on Circuits and Systems*, 1993, pp.103–108.

[14] K. Imai, S. Sumino and H. Imai: Geometric Fitting of Two Corresponding Sets of Points. *Proceedings of the 5th Annual ACM Symposium on Computational Geometry*, 1989, pp.266–275.

[15] K. Kedem, M. Sharir and S. Toledo: On Critical Orientations in the Kedem-Sharir Motion Planning Algorithm for a Convex Polygon in the Plane. *Proceedings of the 5th Canadian Conference on Computational Geometry*, 1993, pp.204–209.

[16] M. Sharir: Almost Linear Upper Bounds on the Length of General Davenport-Schinzel Sequences. *Combinatorica*, Vol.7, No.2 (1987), pp.131–143.

[17] M. Sharir: Improved Lower Bounds on the Length of Davenport-Schinzel Sequences. *Combinatorica*, Vol.8 (1988), pp.117–124.

[18] M. Sharir: Arrangements of Surfaces in Higher Dimensions: Envelopes, Single Cells, and Other Recent Developments. *Proceedings of the 5th Canadian Conference on Computational Geometry*, 1993, pp.181–186.

[19] M. Sharir, R. Cole, K. Kedem, D. Leven, R. Pllack and S. Sifrony: Geometric Applications of Davenport-Schinzel Sequences. *Proceedings of the 27th IEEE Symposium on Foundations of Computer Science*, 1986, pp.77–86.

[20] E. Szeméredi: On a Problem by Davenport and Schinzel. *Acta Arithmetica*, Vol.25 (1974), pp.213–224.

[21] R. E. Tarjan: Efficiency of a Good but not Linear Set-union Algorithm, *Journal of Association for Computing Machinery*, Vol.22 (1975), pp.215–225.