Largest Similar Copies of Convex Polygons in Polygonal Domains

Taekang Eom ✉
Department of Computer Science and Engineering,
Pohang University of Science and Technology, South Korea

Seungjun Lee ✉
Department of Computer Science and Engineering,
Pohang University of Science and Technology, South Korea

Hee-Kap Ahn ✉
Department of Computer Science and Engineering, Graduate School of Artificial Intelligence,
Pohang University of Science and Technology, South Korea

Abstract

Given a convex polygon with k vertices and a polygonal domain consisting of polygonal obstacles with n vertices in total in the plane, we study the optimization problem of finding a largest similar copy of the polygon that can be placed in the polygonal domain without intersecting the obstacles. We present an upper bound $O(k^2n^2\lambda_4(k))$ on the number of combinatorial changes occurred to the underlying structure during the rotation of the polygon, together with an $O(k^2n^2\lambda_4(k)\log n)$-time deterministic algorithm for the problem. This improves upon the previously best known results by Chew and Kedem [SoCG89, CGTA93] and Sharir and Toledo [SoCG91, CGTA94] on the problem in more than 27 years. Our result also improves the time complexity of the high-clearance motion planning algorithm by Chew and Kedem.

2012 ACM Subject Classification Theory of computation → Computational geometry

Keywords and phrases Polygon placement, Largest similar copy, Polygonal domain

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2021.19

Related Version *Full Version*: `https://arxiv.org/abs/2012.06978`

Funding This research was partly supported by the Institute of Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (No. 2017-0-00905, Software Star Lab (Optimal Data Structure and Algorithmic Applications in Dynamic Geometric Environment)) and (No. 2019-0-01906, Artificial Intelligence Graduate School Program(POSTECH)).

1 Introduction

Finding a largest object of a certain shape that can be placed in a polygonal environment has been considered as a fundamental problem in computational geometry. This kind of optimization problems arise in various applications, including the metal industry where we want to find a largest similar pattern containing no faults in a piece of material. There is also a correspondence to motion planning problems [12, 13, 17] and shape matching [10].

In the *polygon placement problem*, we are given a container and a fixed shape, and want to find a largest object of the shape that can be inscribed in the container. There are various assumptions on the object to be placed, the motions allowed, and the environment the object is placed within. In many cases, the container is a convex or simple polygon, possibly with holes. Typical shapes are squares, triangles with/without fixed interior angles, and rectangles with/without fixed aspect ratios. For the motions, we may allow translation or both translation and rotation, together with scaling. When scaling is not allowed, the problem is to find a copy of a given object under translation or rigid motion that can be inscribed in a container [7, 5]. When both translation and scaling are allowed, the objective becomes to find a largest homothetic copy of a given object that can be inscribed in a container [11, 15]. When rotation is allowed, together with translation and scaling, the problem is to find a largest similar copy of a given object that can be inscribed in a container and it becomes more involved; it may require to capture every change occurring to the underlying structure during the rotation of the polygon, and therefore the complexity of the algorithms may depend on the total number of the changes.

In this paper, we consider the polygon placement problem under translation, rotation, and scaling. We aim to find a largest similar copy of a given convex polygon P with k vertices that can be inscribed in a polygonal domain Q consisting of polygonal obstacles with n vertices in total. This problem has been considered fairly well for many years. See Chapter 50 of the Handbook of Discrete and Computational Geometry [12].

The earliest result was perhaps the SoCG'89 paper by Chew and Kedem [8]. They considered the problem and gave an incremental technique for handling all combinatorial changes to the Delaunay triangulation of the polygonal domain Q under the distance function induced by the input polygon P during the rotation of P. They gave an upper bound $O(k^4 n\lambda_4(kn))$ on the number of combinatorial changes, together with a deterministic $O(k^4 n\lambda_4(kn)\log n)$-time algorithm, where $\lambda_s(n)$ is the length of the longest Davenport–Schinzel sequence of order s including n distinct symbols. A few years later, the bound was improved to $O(k^4 n\lambda_3(n))$ by the same authors, and thus the running time of the algorithm became $O(k^4 n\lambda_3(n)\log n)$ [9].

Toledo [21], and Sharir and Toledo [19] studied this problem (they called this problem *the extremal polygon containment problem*) and applied the motion-planning algorithm [13] to solve this problem. They gave an algorithm with running time $O(k^2 n\lambda_4(kn)\log^3(kn)$ $\log\log(kn))$ that uses the parametric search technique of Megiddo [16].

For these two running times, $O(k^4 n\lambda_3(n)\log n)$ and $O(k^2 n\lambda_4(kn)\log^3(kn)\log\log(kn))$, the latter one is asymptotically smaller for large k ($k > n$) while the former one is asymptotically smaller for small k.

There is a randomized algorithm by Agarwal et al. [2] that finds a largest similar copy in $O(kn\lambda_6(kn)\log^3(kn)\log^2 n)$ expected time using the parametric search technique of Megiddo [16]. Agarwal et al. [1] also considered a special case of the problem for finding a largest similar copy of a given convex k-gon contained in a convex n-gon and gave an $O(kn^2\log n)$-time algorithm. Very recently, there were results on two variants. Given a set of n points in the plane, Bae and Yoon [6] gave an $O(n^2\log n)$-time algorithm for finding a largest square that contains no input point in its interior, but contains input points on every side or on three sides. Lee et al. [14] gave an algorithm for finding a largest triangle with fixed interior angles in a simple polygon with n vertices in $O(n^2\log n)$ time.

However, no improvement to the worst-case time bounds by Chew and Kedem, and Sharir and Toledo has been known for the polygon placement problem.

New result. We present an upper bound $O(k^2 n^2\lambda_4(k))$ on the combinatorial changes, which directly improves the worst-case time bound for the algorithm to $O(k^2 n^2\lambda_4(k)\log n)$. This improves upon the previously best known results by Chew and Kedem [9] and Sharir and Toledo [19] in more than 27 years.

Compared to the combinatorial bound $O(k^4 n\lambda_3(n))$ by Chew and Kedem, our bound is $o(k^3 n^2\log^* k)$ while their bound is $O(k^4 n^2\alpha(n))$, because $\lambda_4(k) = o(k\log^* k)$ [20] and $\lambda_3(n) = \Theta(n\alpha(n))$. Therefore, our algorithm outperforms theirs for both k and n. Compared to the time bound $O(k^2 n\lambda_4(kn)\log^3(kn)\log\log(kn))$ by Sharir and Toledo [19], our running time outperforms theirs for both k and n without resorting to parametric search, because $n\lambda_4(k)\log n = o(\lambda_4(kn)\log^3(kn)\log\log(kn))$. Thus our result improves upon the best deterministic result for the problem introduced in Chapter 50 of the Handbook of Discrete and Computational Geometry [12].

Compared to the randomized algorithm using parametric search by Agarwal et al. [2], the worst-case running time of our algorithm, without resorting to parametric search, is asymptotically smaller than their expected running time when $k = O(\log^4 n)$ while it is unclear how to compare the two worst-case running time and the expected running time for larger k.

There is some correspondence between the combinatorial complexity and motion planning problems [9, 12]. In the *high-clearance motion planning,* the goal is to find the path of a convex polygonal robot P contained in a polygonal domain Q from an initial position to a final position while remaining "as far as possible" from the boundaries of Q throughout translations and rotations of P. Chew and Kedem [9] gave an $O(k^4 n\lambda_3(n)\log n)$-time algorithm for high-clearance motion planning, where k and n are the numbers of vertices of P and Q, respectively. Since the running time is dominated by the number of combinatorial changes, our result improves the running time to $O(k^2 n^2\lambda_4(k)\log n)$.

Our result provides some insight for improving the time bounds for some other combinatorial problems with moving objects, for instance the Voronoi diagrams and Delaunay triangulations for moving points under various convex distances [3, 18].

Our improvement may seem marginal compared to previously best ones. However, this is the first and only result that pushes the (worst-case) complexity barrier for the last 27 years. We conjecture that the tight bound is $\Theta(k^2 n^2)$, though we do not have a proof yet. Our result has factor $O(\lambda_4(k))$ to the conjecture, and it could be a stepping stone to closing the gap.

Overview of techniques. We achieve the improved upper bound by carefully analyzing the combinatorial changes in the edge Delaunay triangulation of Q (to be defined later, shortly eDT) while rotating P, and by reducing the candidate size to consider for the changes. Our strategy follows the approach of Chew and Kedem [9], which consists of two parts: Counting the combinatorial changes in eDT for a constant k, and then counting the combinatorial changes with respect to k.

1. In the first part, we analyze the combinatorial changes for a fixed k. We consider a family of functions defined for each vertex and edge of P, and compute their lower envelope. Since there are $O(k)$ vertices and edges of P, we compute $O(k)$ lower envelopes. We show that the complexity of each lower envelope is $O(n)$. Then we compute the breakpoints on the lower envelope of the lower envelopes. To bound the number of combinatorial changes in eDT, we consider a placement of a scaled copy of P such that a vertex of Q and a vertex of P are in contact, which we call a *hinge*. We show that the number of breakpoints on the lower envelope defined for each hinge is $O(n)$. Since there are $O(n)$ hinges for a constant k, the number of combinatorial changes in eDT for θ increasing from 0 to 2π is $O(n^2)$.
2. In the second part, we analyze the combinatorial changes to eDT with respect to k. A combinatorial change to eDT corresponds to a quadruplet of pairs, each pair consisting of an element of Q and an element of P touching each other in some placement of a scaled copy of P simultaneously. To count the quadruplets inducing combinatorial changes to eDT, we consider the triplets of such pairs and define a function for each triplet implying

the size of the scaled copy of P defined by the triplet, satisfying the followings: For the lower envelope L of the functions, a combinatorial change corresponds to an intersection of two such functions appearing in L. That is, every combinatorial change to eDT occurs at a breakpoint on the lower envelope of the functions. So, the complexity of the lower envelope bounds the number of combinatorial changes that occur during the rotation of P. We reduce the complexity bound on the lower envelope by classifying the combination of pairs for the quadruplets.

While this high-level strategy may appear similar to the previous one [9], there are a few major differences and difficulties in improving the bound. In the first part, we improve upon the previous bound $O(n\lambda_3(n))$ by Chew and Kedem as follows. We partition the family of functions to subfamilies such that the functions in the same subfamily have the same domain length, and therefore the complexity of their lower envelope becomes linear to the number of functions [6, 14]. Thus, the total upper bound is improved to $O(n^2)$.

In the second part, instead of the quadruplets considered by Chew and Kedem, we consider the triplets of pairs only and show that the functions for the triplets, in their lower envelope, give us an upper bound on the number of the combinatorial changes to eDT. These functions must reflect the placement of a scaled copy of P in Q as well as the scaling factor. We define functions satisfying this requirement and show that every combinatorial change to eDT occurs at a breakpoint on the lower envelope of the functions. There are $O(k^3n^2)$ such functions and two functions intersect each other at most four times. Thus, the complexity of the lower envelope of the functions is $O(\lambda_6(k^3n^2))$ as the lower envelope corresponds to a Davenport-Schinzel sequence of order 6. To reduce the bound, we classify the functions into types based on the combinations of pairs defining the functions, and show that any two functions belonging to the same type intersect each other less than four times. By applying the partition method in the first part and the classification on the functions above, we show that the complexity of the lower envelope becomes $O(k^2n^2\lambda_4(k))$.

Due to the limit of space, the proofs of some lemmas and corollaries are given in the full version of the paper.

2 Preliminary

A Davenport–Schinzel sequence is a sequence of symbols in which the frequency of any two symbols appearing in alternation is limited. A sequence of symbols is a Davenport–Schinzel sequence of order s if it has no alternating subsequences of length $s+2$. We use $\lambda_s(n)$ to denote the length of the longest Davenport–Schinzel sequence of order s that includes n distinct symbols.

We use some properties related to Davenport–Schinzel sequences in analyzing algorithms. Let $\mathcal{F} = \{f_1, \ldots, f_n\}$ be a collection of n partially-defined, continuous, one-variable real-valued functions. The points at which two functions intersect each other in their graphs and the endpoints of function graphs are called the *breakpoints*. If any two functions of $\mathcal{F}$ intersect each other in their graphs at most s times, the lower envelope of $\mathcal{F}$ has at most $\lambda_{s+2}(n)$ breakpoints [4]. We introduce some technical lemmas that are used in Section 3.

▶ **Lemma 1** (Lemma 14 of [14]). *Assume any two functions f_i and f_j of $\mathcal{F}$ intersect each other in their graphs at most once and each function f_i has domain D_i of length d. If there is a constant c such that $|\bigcup D_i| = cd$, then the lower envelope of $\mathcal{F}$ has $O(n)$ breakpoints.*

▶ **Lemma 2.** *Let $\mathcal{G} = \{g_1, \ldots, g_m\}$ be a collection of m partially-defined, piecewise continuous, one-variable real-valued functions. Let n be the total number of continuous pieces in the function graphs of $\mathcal{G}$. If any two continuous pieces intersect each other in at most s points, the lower envelope of $\mathcal{G}$ has $O(\frac{n}{m}\lambda_{s+2}(m))$ breakpoints.*

We introduce the *edge Voronoi diagram* and its dual, the *edge Delaunay triangulation* (eDT). The set S of sites consists of the edges (open line segments) and their endpoints in the polygonal domain Q. For a convex polygon P in $\mathbb{R}^2$ containing the origin in its interior, the P-distance from a point p to a point q is $d_P(p,q) = \inf\{\mu \mid q \in p + \mu P\}$. The edge Voronoi diagram is a subdivision of the plane into regions such that the points in the same region have the same nearest site under P-distance. See Figure 1(a).

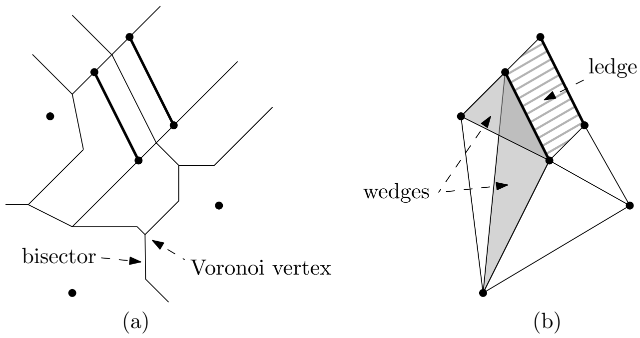

Figure 1 (a) The edge Voronoi diagram of sites, two open line segments (thick) and seven points, under P-distance when P is an axis-aligned square. (b) The edge Delaunay triangulation dual to the edge Voronoi diagram in (a).

The edge Voronoi diagram consists of *Voronoi vertices* and *Voronoi edges (bisectors)*. A point in the plane is a Voronoi vertex if and only if there is an empty circle defined by the P-distance centered at the point and touching three or more sites. The number of Voronoi vertices is linear to the complexity of the sites [15]. A Voronoi edge is a polygonal line that connects two Voronoi vertices. Each point on a Voronoi edge is equidistant from the two sites defining the edge under P-distance. The edge Voronoi diagram can be constructed in $O(kn \log kn)$ time and $O(kn)$ space [15], where k and n are the numbers of vertices in P and Q, respectively.

Just as the standard Delaunay triangulation is the dual of the standard Voronoi diagram, the edge Delaunay triangulation (eDT) is the dual of the edge Voronoi diagram. It has three types of *generalized edges*: *edges*, *wedges*, and *ledges*. An edge connects two point sites, a wedge connects a point site and a segment site, and a ledge connects two segment sites. See Figure 1(b). The edge Delaunay triangulation is a planar graph consisting of point sites, segment sites, generalized edges, and empty triangles. Since a ledge is a trapezoid or a degenerate trapezoid, eDT may not be a triangulation.

The edge Delaunay triangulation can be constructed by first building the edge Voronoi diagram and then tracing the diagram to determine the sites that define each portion of the Voronoi edges and vertices. The type of a generalized edge is determined by the sites defining the corresponding Voronoi edge.

2.1 The Algorithm of Chew and Kedem

We present a sketch of the algorithm by Chew and Kedem. Imagine we rotate P by angle θ in counterclockwise direction, and let P_θ be the rotated copy of P. A homothetic copy $\mathcal{P}_\theta$ of P_θ is said to be *feasible* if $\mathcal{P}_\theta$ is inscribed in Q. For the set S of the sites consisting of the edges and their endpoints in Q, let eDT_θ denote the edge Delaunay triangulation of the sites in S under P_θ-distance. For a face T of eDT_θ, we say $\mathcal{P}_\theta$ is *associated to* T if it touches every site defining T. For $\mathcal{P}_\theta$ associated to T, the set of the elements (vertices or edges) of

$\mathcal{P}_\theta$ touching the sites defining T becomes the *label of* T. See Figure 2(a). For a site s of S, the *label of* s is the set of elements of $\mathcal{P}_\theta$ touching s, for $\mathcal{P}_\theta$ associated to the faces incident to s. See Figure 2(b).

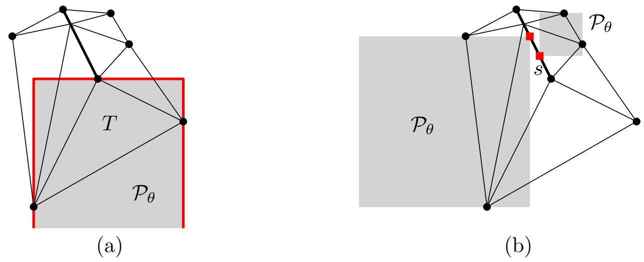

Figure 2 Labels of faces of eDT_θ and edge sites of S for an axis-aligned square P_θ. (a) The label of face T is the set of the edges (red segments) of $\mathcal{P}_\theta$, each containing a site defining T. (b) The label of edge site s is the set of the corners (red points) of $\mathcal{P}_\theta$'s lying on s.

Their algorithm classifies two possible types of changes, an edge change and a label change in eDT_θ while θ increases. In an edge change, a new generalized edge appears or an existing edge disappears. This change occurs when $\mathcal{P}_\theta$ touches four elements of Q, resulting in a flip of the diagonals in the quadrilateral formed by the four edges of eDT_θ. In a label change, the label of a face in eDT_θ changes. This occurs when two or more elements of $\mathcal{P}_\theta$ touch the same site, but the structure of eDT_θ may remain unchanged. Any edge or label change is called a *combinatorial change* to eDT_θ.

Their algorithm maintains a representation for eDT_θ while θ increases. It starts by constructing eDT_θ at $\theta = 0$. An edge change is detected by checking the edges of eDT_θ and a label change is detected by checking the faces of eDT_θ. For each generalized edge in eDT_θ, it determines at which orientation this edge ceases to be valid due to an interaction with its neighbors. For each face in eDT_θ, the algorithm determines at which orientation the label of this face changes. The algorithm maintains the edges and faces of eDT_θ in a priority queue, ordered by the orientations at which they change. At each succeeding stage of the algorithm, it determines which generalized edge is the next one to disappear or which face is the next one to have its label changed as θ increases.

For an edge change, a new edge appears in eDT_θ. Then the algorithm updates eDT_θ and the priority queue information on the new edge and its neighboring edges and faces (an edge change) and for the face and the edges incident to it (a label change). A priority queue can be implemented such that each operation can be done in $O(\log m)$ time, where m is the maximum number of items in the queue. Since there are $O(n)$ edges and faces in the queue at any time, each priority queue operation takes time $O(\log n)$.

For each event of a face T disappearing at θ_e, the algorithm finds the maximal interval $\mathcal{I} = [\theta_s, \theta_e]$ of θ such that T appears in eDT_θ. To find $\mathcal{I}$ in $O(1)$ time, it stores at T the orientation at which it starts to appear in eDT_θ. Then it computes the orientation $\theta^* \in \mathcal{I}$ that maximizes the area of each $\mathcal{P}_\theta$ that touches every site defining T simultaneously. Since $\mathcal{I}$ is maximal, $\mathcal{P}_\theta$ is feasible for every $\theta \in \mathcal{I}$ but not for any $\theta \notin \mathcal{I}$ sufficiently close to $\mathcal{I}$. Thus, the algorithm considers all orientations θ such that $\mathcal{P}_\theta$ is feasible and computes the placement and orientation of the largest similar copy of P. The area function of $\mathcal{P}_\theta$ can be computed in $O(1)$ time and there are $O(1)$ $\mathcal{P}_\theta$ that touch every site defining T simultaneously. Chew and Kedem gave an upper bound $O(k^4 n\lambda_3(n))$ on the number of combinatorial changes, and their algorithm takes $O(k^4 n\lambda_3(n)\log n)$ time [9].

3 The number of changes in eDT$_\theta$

We show that the number of combinatorial changes in eDT$_\theta$ during the rotation is $O(k^2n^2\lambda_4(k))$. This directly improves the time bound of the algorithm by Chew and Kedem to $O(k^2n^2\lambda_4(k)\log n)$. We analyze the number of combinatorial changes in eDT$_\theta$ for a constant k in Section 3.1, and then analyze the number of combinatorial changes with respect to k in Section 3.2 using the result in Section 3.1.

An ordered pair (A, B) is a *side contact pair* if A is a side of Q and B is a corner of P, and a *corner contact pair* if A is a corner of Q and B is a side of P. A homothetic copy $\mathcal{P}$ of P *satisfies* a contact pair (A, B) if B in $\mathcal{P}$ touches A. See Figure 3 (a). Recall that a homothetic copy $\mathcal{P}$ is said to be *feasible* if $\mathcal{P}$ is inscribed in Q. Note that a homothetic copy $\mathcal{P}$ is not necessarily feasible even if $\mathcal{P}$ satisfies a contact pair.

3.1 The number of changes for fixed k

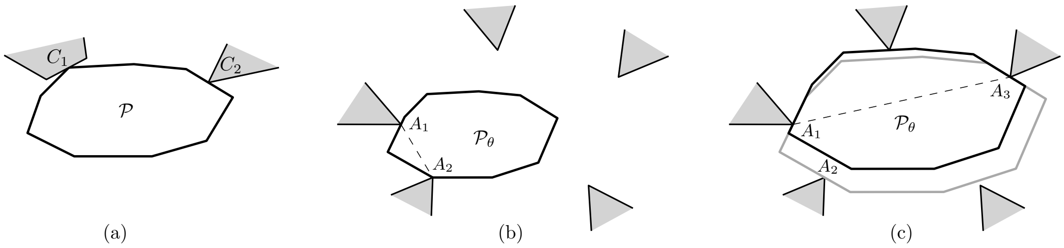

Figure 3 (a) $\mathcal{P}$ satisfies a side contact pair C_1 and a corner contact pair C_2. (b) The segment A_1A_2 is a reported edge with $Q_H = A_2$. (c) The segment A_1A_3 is an unreported edge because no hinge is involved in the segment for a feasible $\mathcal{P}_\theta$.

For a constant k, we improve upon the previously best upper bound $O(n\lambda_3(n))$ by Chew and Kedem [9] to $O(n^2)$. A key idea is to consider the lower envelope of some functions related to the expansion factor, one for each edge and vertex of P, and then to analyze the lower envelope of those lower envelopes. By careful analysis on the complexities of the lower envelopes, we show that the number of combinatorial changes to eDT$_\theta$ for θ increasing from 0 to 2π is $O(n^2)$.

To bound the number of changes tight, we classify the generalized edges into two types and count them separately. Chew and Kedem also used this approach. An ordered pair (Q_H, P_H) is a *hinge* if Q_H is a corner of Q and P_H is a corner of P. For a hinge $H = (Q_H, P_H)$ and a contact pair $C = (A, B)$, the generalized edge connecting Q_H and A is a *reported edge* if there is a feasible $\mathcal{P}_\theta$ for some θ satisfying both H and C. An edge of eDT$_\theta$ is an *unreported edge* if it is not a reported edge. See Figure 3(b,c). We use the numbers of changes to the reported edges and to the labels in counting the changes to the unreported edges.

Changes to the reported edges and the label changes to point sites. We count the changes to the reported edges and the changes to the labels of point sites in eDT$_\theta$ for θ increasing from 0 to 2π. We define the *expansion function* $E_{HC}(\theta)$ for a hinge H and a contact C to be the minimal expansion factor of $\mathcal{P}_\theta$ satisfying H and C. For a hinge $H = (Q_H, P_H)$, let $\mathcal{F}_H$ be the set of all expansion functions satisfying H and another contact pair. An expansion function $E_{HC}(\theta)$ of $\mathcal{F}_H$ for a contact $C = (A, B)$ appears in the lower envelope of $\mathcal{F}_H$ at θ if the generalized edge connecting Q_H and A is a reported edge in eDT$_\theta$.

For a set X of functions, let $\mathsf{B}(X)$ denote the number of breakpoints on the lower envelope of the functions in X. Then the number of changes to the reported edges in eDT_θ which involve H is bounded by $\mathsf{B}(\mathcal{F}_H)$.

Every label change to a point site involves a hinge. See Figure 4(a). An intersection of E_{HC_1} and E_{HC_2} of $\mathcal{F}_H$ for contact pairs C_1 and C_2 appears in the lower envelope of $\mathcal{F}_H$ if a label change to a point site is induced by C_1, C_2 and H. Then the number of label changes to the point sites in eDT_θ which involve H is bounded by $\mathsf{B}(\mathcal{F}_H)$.

▶ **Proposition 3** (Proposition 3 of [9]). *Two expansion functions E_{HC_1} and E_{HC_2} intersect each other in at most one point in their graphs if both C_1 and C_2 are corner contact pairs, or both are side contact pairs. If one is a corner contact pair and the other is a side contact pair, E_{HC_1} and E_{HC_2} intersect each other in at most two points in their graphs.*

Let v_i and e_i denote the vertices and edges of P for $i = 1, \dots, k$. For each $i = 1, \dots, k$, let $\mathcal{C}_{1i} = \{(e, v_i) \mid e \text{ is an edge of } Q\}$ be the set of side contact pairs and let $\mathcal{C}_{2i} = \{(v, e_i) \mid v \text{ is a vertex of } Q\}$ be the set of corner contact pairs. Let $\mathcal{F}_{ji} = \{E_{HC} \mid C \in \mathcal{C}_{ji}\}$ for $j = 1, 2$.

▶ **Lemma 4.** $\mathsf{B}(\mathcal{F}_{ji}) = O(n)$ *for each $j = 1, 2$ and $i = 1, \dots, k$.*

Proof. $\mathsf{B}(\mathcal{F}_{1i}) = O(n)$ for each i since E_{HC_1} and E_{HC_2} intersect each other only at the boundaries of their intervals for $C_1, C_2 \in \mathcal{C}_{1i}$.

Consider now $\mathsf{B}(\mathcal{F}_{2i})$. Two expansion functions E_{HC_1} and E_{HC_2} intersect each other in at most one point for $C_1, C_2 \in \mathcal{C}_{2i}$. Also, E_{HC} has the same length of domain for all $C \in \mathcal{C}_{2i}$. Thus, $\mathsf{B}(\mathcal{F}_{2i}) = O(n)$ by Lemma 1. ◀

From Lemma 2, Proposition 3, and Lemma 4, we achieve an upper bound on $\mathsf{B}(\mathcal{F}_H)$.

▶ **Lemma 5.** $\mathsf{B}(\mathcal{F}_H) = O(\lambda_3(k)n)$.

Proof. Let $\mathcal{F}_j = \{f_{j1}, \dots, f_{jk}\}$ for $j = 1, 2$, where f_{ji} is the lower envelope of $\mathcal{F}_{ji}$ for each $i = 1, \dots, k$ and $j = 1, 2$. Let $\mathcal{L}_j$ denote the lower envelope of $\mathcal{F}_j$. Then, the lower envelope of $\mathcal{F}_H$ is the lower envelope of $\mathcal{L}_1$ and $\mathcal{L}_2$. The number of breakpoints on $\mathcal{L}_j$ is $O(\lambda_3(k)n)$ for $j = 1, 2$ by Lemma 2, Proposition 3, and Lemma 4. Then $\mathsf{B}(\mathcal{F}_H) = O(\lambda_3(k)n)$, because two continuous pieces, one from $\mathcal{L}_1$ and one from $\mathcal{L}_2$, intersect each other in at most two points by Proposition 3. ◀

By Lemma 5, the number of changes to the reported edges and the number of label changes to the point sites are $O(k\lambda_3(k)n^2)$.

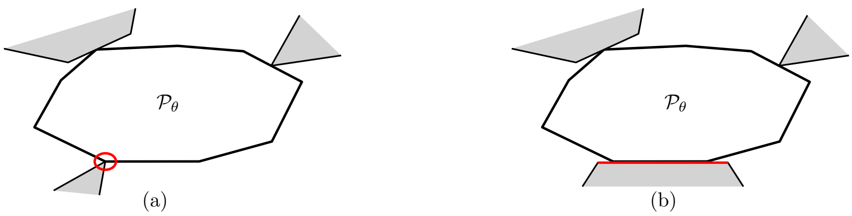

Figure 4 (a) Label change to a point site (hinge). (b) Label change to an edge site.

Label changes to edge sites. We count the changes to the labels of edge sites in eDT_θ for θ increasing from 0 to 2π. Imagine we fix an edge e of Q and an edge g of P. See Figure 4(b). Then, the number of label changes to edge site e with g is $O(n)$ because there are $O(n)$ different $\mathcal{P}_\theta$'s, each associated to a face of eDT_θ while e and g are aligned and touching each other. Thus, the total number of label changes to all edge sites is $O(kn^2)$.

Changes to unreported edges. We count the changes to the unreported edges using the number of changes to the reported edges and to the labels, and Lemma 6.

▶ **Lemma 6** (Lemma 2 of [9]). *Every edge of eDT_θ is either a reported edge or a diagonal in a convex l-gon, $l \leq 3k$, whose sides are either reported edges or portions of edge sites.*

Let G_θ be the graph whose edges are the reported edges in eDT_θ and portions of edge sites in Lemma 6. We count the changes to the unreported edges which are diagonals in a face of G_θ for an interval of θ with no label change to eDT_θ. Observe that no combinatorial change occurs to G_θ for the interval. Any change to an unreported edge involves four sites lying on a face boundary of G_θ. There are at most four changes for a group of four sites. We describe the details on this bound in Section 4 in the full version. Since each face has at most $3k$ edges by Lemma 6, there are at most $\binom{3k}{4}$ such groups. Thus, $O(k^4)$ changes occur to the unreported edges for the boundary of a face g of G_θ for an interval of θ with no label change to the faces of eDT_θ intersecting g. Since the number of changes to the reported edges and to the labels is $O(k\lambda_3(k)n^2)$, there are $O(k^5n^2\lambda_3(k))$ combinatorial changes to eDT_θ.

▶ **Theorem 7.** *For a polygonal domain Q of size n and a convex k-gon P, the number of combinatorial changes to eDT_θ for θ increasing from 0 to 2π is $O(n^2)$ for a constant k.*

3.2 The number of changes with respect to k

We now consider k as a variable and bound the number of changes to eDT_θ. Since each triangular face in eDT_θ is defined by three elements (edges or vertices) of P, we choose three elements of P and use their convex hull in the counting. Then by Theorem 7, the total number of faces in eDT_θ for all these convex hulls is $O(k^3n^2)$ for θ increasing from 0 to 2π.

Let $\mathcal{T}$ be the set of all faces of eDT_θ for the convex hull of three elements B_1, B_2, B_3 of P such that the contact pairs inducing the face have B_1, B_2, and B_3 as their elements. Consider two faces T and T' of eDT_θ for two distinct orientations θ_1 and θ_2 with $\theta_1 < \theta_2$ that are defined by the same sites. We consider T and T' as distinct faces if there is any change to T or T' in eDT_θ for θ increasing from θ_1 to θ_2. For a face $T \in \mathcal{T}$, let $C(T)$ be the set of contact pairs which defines T, and let $I(T)$ be the interval of θ at which T appears in eDT_θ.

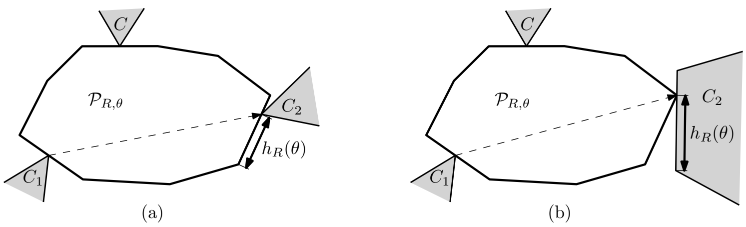

Figure 5 $\mathcal{P}_{R,\theta}$ and $h_R(\theta)$ for a restricted contact pair $R = (C, I)$ and θ. Let $h_R(\theta)$ be the distance from the clockwise endpoint (with respect to the dashed ray) of the side element of C_2 to point element of C_2. (a) $h_R(\theta)$ when C_2 is a corner contact. (b) $h_R(\theta)$ when C_2 is a side contact.

For any two fixed contact pairs (C_1, C_2) with $C_i = (A_i, B_i)$ for $i = 1, 2$ such that $A_1 \neq A_2$ and $B_1 \neq B_2$, we count the combinatorial changes involving (C_1, C_2) and other two contact pairs C, C' given in counterclockwise order C_1, C_2, C, and C' along the boundary of P. The combinatorial changes for the cases that $A_1 = A_2$ or $B_1 = B_2$ will be counted for other choices of fixed contact pairs.

We use (C, I) to denote a contact pair C *restricted to* an interval I of θ. For (C_1, C_2), let $\mathcal{R}$ be the set of *restricted contact pairs* (C, I) such that $C(T) = \{C_1, C_2, C\}$ and $I = I(T)$ for a face $T \in \mathcal{T}$, and C_1, C_2, C appear in counterclockwise order along the boundary of P. For a fixed restricted contact pair $R = (C, I) \in \mathcal{R}$ and $\theta \in I$, let $\mathcal{P}_{R,\theta}$ denote the homothet of P_θ which satisfies C_1, C_2, and C. Let $h_R(\theta)$ be the function that denotes the distance from the clockwise endpoint of the side element of C_2 (with respect to the ray from the point element of C_1 to the point element of C_2) to the point element of C_2 with respect to $\mathcal{P}_{R,\theta}$. Observe that h_R is a partially defined continuous function on $R \in \mathcal{R}$. See Figure 5.

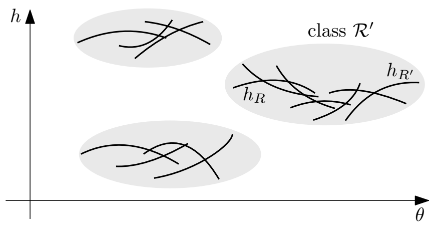

Figure 6 Partitioning $\mathcal{R}$ into classes using the graphs of functions in $\mathcal{F} = \{h_R \mid R \in \mathcal{R}\}$. For two pairs R, R' in class $\mathcal{R}'$, h_R and $h_{R'}$ are connected in the union of the function graphs of $\mathcal{F}$.

Let $\mathcal{F} = \{h_R \mid R \in \mathcal{R}\}$. We partition $\mathcal{R}$ into classes such that two restricted contact pairs R, R' belong to the same class if and only if h_R and $h_{R'}$ are connected in the union of the function graphs of $\mathcal{F}$. Figure 6 illustrates the classes of $\mathcal{R}$.

If a combinatorial change occurs by C_1, C_2, C, and C' at θ, we have $h_R(\theta) = h_{R'}(\theta)$ for two distinct restricted contact pairs $R = (C, I)$ and $R' = (C', I')$. Let $\mathcal{R}'$ be a class of $\mathcal{R}$ and let $\mathcal{F}' = \{h_R \mid R \in \mathcal{R}'\}$. We verify that if $\mathcal{P}_{R,\theta}$ is feasible, then $h_R(\theta)$ appears in the lower envelope or upper envelope of $\mathcal{F}'$.

▶ **Lemma 8.** *Let $R, R', R'' \in \mathcal{R}$ be the restricted contact pairs in the same class. If $h_{R'}(\theta) < h_R(\theta) < h_{R''}(\theta)$, then $\mathcal{P}_{R,\theta}$ is not feasible.*

For a vertex v_i of P, let $\mathcal{R}'_{1i}$ be the set consisting of restricted contact pairs $(C, I) \in \mathcal{R}'$ such that $C = (e, v_i)$ is a side contact pair for some edge $e \in Q$. For an edge e_i of P, let $\mathcal{R}'_{2i}$ be the set consisting of restricted contact pairs $(C, I) \in \mathcal{R}'$ such that $C = (v, e_i)$ is a corner contact pair for some vertex $v \in Q$. Let $|\mathcal{R}'| = m$, and let $|\mathcal{R}'_{1i}| = m_{1i}$ and $|\mathcal{R}'_{2i}| = m_{2i}$ for $i = 1, \ldots, k$. Let $\mathcal{F}'_{ji} = \{h_R \mid R \in \mathcal{R}'_{ji}\}$ for $j = 1, 2$. Recall that for a set X of functions, $\mathsf{B}(X)$ denotes the number of breakpoints on the lower envelope of the functions in X. We have $\mathsf{B}(\mathcal{F}'_{1i}) = O(m_{1i})$ since h_R and $h_{R'}$ intersect each other only at the boundaries of their intervals for $R, R' \in \mathcal{R}'_{1i}$. Let d_i be the number of intersections of the function graphs of $\mathcal{F}'_{2i}$, and let $d = \sum_{i=1}^{k} d_i$. Then $\mathsf{B}(\mathcal{F}'_{2i}) = O(m_{2i} + d_i)$.

▶ **Lemma 9.** $\mathsf{B}(\mathcal{F}'_{1i}) = O(m_{1i})$ *and* $\mathsf{B}(\mathcal{F}'_{2i}) = O(m_{2i} + d_i)$ *for each* $i = 1, \ldots, k$.

Observe that each intersection of the function graphs of $\mathcal{F}'_{2i}$ corresponds to a combinatorial change to eDT for the convex hull of B_1, B_2, and e_i. See Figure 7(a). By Lemma 8, every combinatorial change appears in the lower envelope or upper envelope of $\mathcal{F}'$. Here, we describe the case for the lower envelope of $\mathcal{F}'$. We count the breakpoints of certain types on the lower envelope of $\mathcal{F}'$. We use (a, b)-*change* to denote a combinatorial change induced by a side contact pairs and b corner contact pairs.

Two side contact pairs. We count only $(4, 0)$-changes in this case. We count $(3, 1)$ and $(2, 2)$-changes appearing in the lower envelope in other choices of two pairs, one side contact pair and one corner contact pair. See Figure 7(b). For $\mathcal{F}'_1 = \{f_i \mid i = 1, \ldots, k\}$ such that f_i

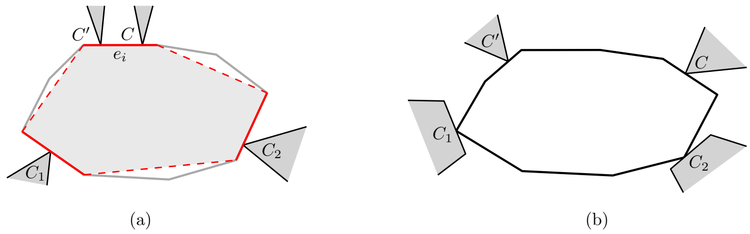

Figure 7 (a) For $R = (C, I)$ and $R' = (C', I')$ in $\mathcal{R}'_{2i}$, and orientation θ with $h_R(\theta) = h_{R'}(\theta)$, there exists a rotated and scaled copy of the convex hull of B_1, B_2, e_i that is feasible and satisfies C_1, C_2, C, and C'. The intersection $h_R(\theta) = h_{R'}(\theta)$ corresponds to a combinatorial change to eDT_θ for the convex hull of B_1, B_2, and e_i. (b) The $(2,2)$-change induced by C_1, C_2, C and C' is counted when C_2 and C are chosen as the fixed pair.

is the lower envelope of $\mathcal{F}'_{1i}$, $\mathsf{B}(\mathcal{F}'_1) = \sum_{i=1}^{k} O(m_{1i}\lambda_4(k)/k) = O(m\lambda_4(k)/k)$ by Lemmas 2 and 9. We show that any two continuous pieces in $\mathcal{F}'_1$ intersect each other in at most two points in Section 4 of the full version. Since each $(4,0)$-change corresponds to a breakpoint on the lower envelope of $\mathcal{F}'_1$, the number of $(4,0)$-changes is $O(m\lambda_4(k)/k)$.

Two corner contact pairs. We count only $(0,4)$-combinatorial changes in this case. Other changes are counted for other choices of the fixed pairs. For $\mathcal{F}'_2 = \{f_i \mid i = 1, \ldots, k\}$ such that f_i is the lower envelope of $\mathcal{F}'_{2i}$, $\mathsf{B}(\mathcal{F}'_2) = \sum_{i=1}^{k} O((m_{2i} + d_i)\lambda_4(k)/k) = O((m + d)\lambda_4(k)/k)$ by Lemmas 2 and 9. We show that any two continuous pieces in $\mathcal{F}'_2$ intersect each other in at most two points in Section 4 of the full version. Since each $(0,4)$-change corresponds to a breakpoint of the lower envelope of $\mathcal{F}'_2$, the number of $(0,4)$-changes is $O((m + d)\lambda_4(k)/k)$.

One side contact pair and one corner contact pair. We count all combinatorial changes other than $(4,0)$-changes and $(0,4)$-changes. First, we count the breakpoints on the lower envelope of $\mathcal{F}'_1 = \{h_R \mid R \in \bigcup_{i=1}^{k} \mathcal{R}'_{1i}\}$ and on the lower envelope of $\mathcal{F}'_2 = \{h_R \mid R \in \bigcup_{i=1}^{k} \mathcal{R}'_{2i}\}$, and then count the breakpoints on the lower envelope of $\mathcal{F}'_1 \cup \mathcal{F}'_2$. We show that any two continuous pieces, both from either $\mathcal{F}'_1$ or $\mathcal{F}'_2$, intersect each other in at most two points in Section 4 of the full version. We can compute $\mathsf{B}(\mathcal{F}'_1) = O(m\lambda_4(k)/k)$ and $\mathsf{B}(\mathcal{F}'_2) = O((m+d)\lambda_4(k)/k)$ in the same way as for counting $(4,0)$-changes and $(0,4)$-changes, respectively. $\mathsf{B}(\mathcal{F}') = O((m+d)\lambda_4(k)/k)$ since both $\mathsf{B}(\mathcal{F}'_1)$ and $\mathsf{B}(\mathcal{F}'_2)$ are $O((m+d)\lambda_4(k)/k)$, and any two continuous pieces, one from $\mathcal{F}'_1$ and one from $\mathcal{F}'_2$, intersect each other in at most four points. Details are in Section 4 in the full version. Thus, the number the combinatorial changes for the fixed contact pair is $O((m + d)\lambda_4(k)/k)$.

Consider the sum σ of the complexities $|\mathcal{F}'|$ of $\mathcal{F}' = \{h_R \mid R \in \mathcal{R}'\}$ over all classes $\mathcal{R}'$ for a fixed pair. The total sum of σ's for all enumerations of fixed pairs is $O(k^3n^2)$ since $|\mathcal{T}| = O(k^3n^2)$. Similarly, consider the sum ξ of the numbers of intersections (d in the complexities in the previous paragraphs) over all classes for a fixed pair. The total sum of ξ's for all enumerations of fixed pairs is $O(k^3n^2)$ since ξ is bounded by the number of combinatorial changes to eDT_θ for the convex hulls of three elements of P.

▶ **Theorem 10.** *For a polygonal domain Q with n vertices and a convex k-gon P, the number of combinatorial changes to the edge Delaunay triangulation of Q under P_θ-distance for θ increasing from 0 to 2π is $O(k^2n^2\lambda_4(k))$.*

Theorem 10 directly improves upon the algorithm by Chew and Kedem.

▶ **Corollary 11.** *Given a polygonal domain Q with n vertices and a convex k-gon P, we can find a largest similar copy of P inscribed in Q in $O(k^2n^2\lambda_4(k)\log n)$ time using $O(kn)$ space.*

High-clearance motion planning. For a convex polygonal robot P with k vertices and a polygonal domain Q with n vertices in the plane, we want to find a path of P from an initial position to a final position such that the *clearance* of the path exceeds a given value Δ. The clearance of a path of P is the minimum of P-distance from the boundaries of Q throughout translations and rotations of P moving along the path. Chew and Kedem [9] gave an $O(k^4n\lambda_3(n)\log n)$-time algorithm for the high-clearance motion planning. The running time is dominated by the number of combinatorial changes, and our result directly improves the running time.

▶ **Corollary 12.** *Given a convex polygonal robot P with k vertices, a polygonal domain Q with n vertices, initial and final positions of P in the plane, and a clearance Δ, we can find a path of clearance exceeding Δ for P in Q in $O(k^2n^2\lambda_4(k)\log n)$ time using $O(k^2n^2\lambda_4(k))$ space.*

4 The number of critical orientations for four contact pairs

An orientation θ is a *critical orientation* if a combinatorial change to eDT_θ occurs at θ. We consider the critical orientations θ at which $\mathcal{P}_\theta$ has contact with four contact pairs. Recall that an (a, b)-change is a combinatorial change induced by a side contact pairs and b corner contact pairs. We count the number of critical orientations for each (a, b)-change type. The number of critical orientations of $(0, 4)$-change is 2, which is shown in Appendix B of [9]. So we count the critical orientations for the other types of combinatorial changes. The following table summarizes the results. Details can be found in the full version.

Types of (a, b)-change	$(4, 0)$	$(3, 1)$	$(2, 2)$	$(1, 3)$	$(0, 4)$
Number of critical orientations	1	2	4	2	2

References

1 Pankaj K. Agarwal, Nina Amenta, and Micha Sharir. Largest placement of one convex polygon inside another. *Discrete & Computational Geometry*, 19(1):95–104, 1998.

2 Pankaj K. Agarwal, Boris Aronov, and Micha Sharir. Motion planning for a convex polygon in a polygonal environment. *Discrete & Computational Geometry*, 22(2):201–221, 1999.

3 Pankaj K. Agarwal, Haim Kaplan, and Natan Rubin. Kinetic Voronoi diagrams and Delaunay triangulations under polygonal distance functions. *Discrete & Computational Geometry*, 54:871–904, 2015.

4 Mikhail J. Atallah. Dynamic computational geometry. In *Proceedings of the 24th Annual Symposium on Foundations of Computer Science (FOCS 1983)*, pages 92–99. IEEE, 1983.

5 Francis Avnaim and Jean Daniel Boissonnat. Polygon placement under translation and rotation. In Robert Cori and Martin Wirsing, editors, *STACS 88*, pages 322–333, 1988.

6 Sang Won Bae and Sang Duk Yoon. Empty squares in arbitrary orientation among points. In *Proceedings of the 36th International Symposium on Computational Geometry (SoCG 2020)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2020.

7 Bernard Chazelle. The polygon containment problem. In F.P. Preparata, editor, *Advances in Computing Research, Vol I: Computational Geometry*, pages 1–33. JAI Press Inc., 1983.

8 L Paul Chew and Klara Kedem. Placing the largest similar copy of a convex polygon among polygonal obstacles. In *Proceedings of the 5th Annual Symposium on Computational Geometry (SoCG 1989)*, pages 167–173, 1989.

9 L Paul Chew and Klara Kedem. A convex polygon among polygonal obstacles: Placement and high-clearance motion. *Computational Geometry*, 3(2):59–89, 1993.

10 Rudolf Fleischer, Kurt Mehlhorn, Günter Rote, Emo Welzl, and Chee Yap. Simultaneous inner and outer approximation of shapes. *Algorithmica*, 8(1):365, 1992.

11 Steven Fortune. A fast algorithm for polygon containment by translation. In *Proceedings of the 12th International Colloquium on Automata, Languages, and Programming (ICALP 1985)*, pages 189–198, 1985.

12 Jacob E. Goodman, Joseph O'Rourke, and Csaba D. Tóth, editors. *Handbook of Discrete and Computational Geometry*. CRC Press LLC, 3rd edition, 2017.

13 Klara Kedem and Micha Sharir. An efficient motion-planning algorithm for a convex polygonal object in two-dimensional polygonal space. *Discrete & Computational Geometry*, 5:43–75, 1990.

14 Seungjun Lee, Taekang Eom, and Hee-Kap Ahn. Largest triangles in a polygon. *Computational Geometry*, 98:101792, 2021.

15 Daniel Leven and Micha Sharir. Planning a purely translational motion for a convex object in two-dimensional space using generalized Voronoi diagrams. *Discrete & Computational Geometry*, 2:9–31, 1987.

16 Nimrod Megiddo. Applying parallel computation algorithms in the design of serial algorithms. *Journal of the ACM*, 30(4):852–865, 1983.

17 Colm Ó'Dúnlaing and Chee K Yap. A retraction method for planning the motion of a disc. *Journal of Algorithms*, 6(1):104–111, 1985.

18 Natan Rubin. On kinetic Delaunay triangulations: A near quadratic bound for unit speed motions. *Journal of the ACM*, 62(3):25:1–25:85, 2015.

19 Micha Sharir and Sivan Toledo. Extremal polygon containment problems. *Computational Geometry*, 4(2):99–118, 1994.

20 Endre Szemerédi. On a problem by Davenport and Schinzel. *Acta Arithmetica*, 25:213–224, 1974.

21 Sivan Toledo. Extremal polygon containment problems. In *Proceedings of the 7th Annual Symposium on Computational Geometry (SoCG 1991)*, pages 176–185, 1991.