Reordering Ruppert's Algorithm

Alexander Rand

Fall Workshop on Computational Geometry, 10/31-11/1, 2008.

1 Overview

Ruppert's algorithm [6] is an elegant method for generating size-competitive meshes, but admits a poor worst case run-time. Recent time-efficient Delaunay refinement algorithms [2] rely on bounding the the degree of each intermediate triangulation and thus ensure that all local operations in the Delaunay triangulation are efficient. We propose a simple alternative to Ruppert's algorithm which maintains this additional property that the all intermediate triangulations have bounded degree.

The algorithm combines three main ideas. First, the yielding procedure of Ruppert's algorithm is eliminated by instead deleting a nearby circumcenter whenever a midpoint or input point is inserted in the mesh in the spirit of Chew's second algorithm [1]. Second, quality of the mesh is maintained before conformity in a similar fashion to the SVR algorithm [2]. Finally, the triangles on the priority queue are prioritized by circumradius with the largest simplices processed first.

The resulting algorithm produces a conforming Delaunay, size-competitive quality mesh. Additionally, there is an explicit bound on the degree of each triangulation produced by the algorithm which depends only on the minimum angle acceptable for output triangles, denoted κ. The algorithm is simple and can be implemented by making a few changes to Ruppert's algorithm.

2 Preliminaries

Given a non-acute piecewise linear complex (PLC) $\mathcal{C} = (\mathcal{P}, \mathcal{S})$ composed of sets of points and segments, we seek a refinement $\mathcal{C}' = (\mathcal{P}', \mathcal{S}')$ which conforms to the input and contains no triangles with angles less than $\kappa°$. The algorithm incrementally builds a refinement for this purpose.

Definition 1. Let $q \in \mathcal{P}'$ be a vertex added to the mesh.

- n_q is a **nearest neighbor** to q in the triangulation.
- r_q is the **insertion radius** of q: the distance from q to n_q when q is inserted.
- q is called a **quality point** if q was inserted as the circumcenter of a poor quality triangle.
- q is called a **conformality point** if q is an input point or was inserted as the midpoint of a segment.

Definition 2. Given a PLC $\mathcal{C}$, the **local feature size** of a point p, denoted $\text{lfs}(p, \mathcal{C})$ or $\text{lfs}(p)$, is the radius of the smallest disk centered at p that intersects two disjoint features of $\mathcal{C}$.

Local feature size will always be evaluated with respect to the input PLC $\mathcal{C}$, and the second argument will be omitted.

Definition 3. A simplex is **unacceptable** if it is

- an input point which has not been inserted,
- a segment with a nonempty diametral disk, or
- a triangle with an angle less than $\kappa°$.

Definition 4. Given a triangle t, let

- $\text{LE}(t)$ denote the length of the longest side of t and
- $\text{SE}(t)$ denote the length of the shortest side of t.

The following two propositions are specific restatements of the Delaunay property for our needs.

Proposition 1. *Let $p, q \in \mathcal{P}$. Suppose that the segment between p and q is a chord of a disk with radius R which contains no points of $\mathcal{P}$. Then there is a triangle in the Delaunay triangulation of $\mathcal{P}$ which contains p and q and has a radius of at least R.*

Proposition 2. *Let p and q be neighbors in the Delaunay triangulation of $\mathcal{P}$ and suppose that p and q are vertices of triangle t. If $\mathcal{P}'$ is a subset of $\mathcal{P}$ which contains p and q, then there exists triangle t' in the Delaunay triangulation of $\mathcal{P}'$ with $R_{t'} \geq R_t$.*

Next, two propositions relate the degree of a Delaunay triangulation to certain properties.

Proposition 3. *If a triangulation contains no angles smaller than κ, then the degree of the triangulation is at most $\frac{360}{\kappa}$.*

Proposition 4. *Let D be the degree of the Delaunay triangulation of $\mathcal{P}$ and let $p \in \mathcal{P}$. Then the degree of the Delaunay triangulation of $\mathcal{P} \setminus \{p\}$ is at most $2D - 4$.*

Ruppert's algorithm for generating a quality, conforming mesh is given in Algorithm 1. The basic idea of the algorithm is that encroached segments are split to ensure the mesh conforms and poor quality triangles are split generate a quality mesh.

Simplices on the queue are processed by Ruppert's algorithm based on dimension: input points are given higher priority than segments which are given higher priority than triangles. Simplices of equal dimension can be processed in any order.

3 The Reordered Ruppert Algorithm

The following algorithm is very similar to Ruppert's algorithm: it maintains a queue of unacceptable simplices and processes the front simplex by inserting its circumcenter. The main difference from Ruppert's algorithm is that quality points do not yield to segments they encroach, but when conformality points are inserted, a nearby circumcenter is removed (if it exists).

When processing queued simplices, triangles are given the highest priority, followed by input points and finally segments. Triangles are prioritized by circumradius with larger triangles processed first.

Algorithm 1 Ruppert

```
Initialize the Delaunay triangulation of a bounding box.
Form a priority queue of all encroached simplices.
while the queue is nonempty do
  Propose the circumcenter q of the top simplex s for insertion.
  if s is a triangle and q encroaches a segment s then
    Queue s.
  else
    Insert q.
    Update the priority queue.
  end if
end while
```

Algorithm 2 Reordered Ruppert

```
Initialize the Delaunay triangulation of a bounding box.
Form a priority queue of all unacceptable simplices.
while the queue is nonempty do
  Insert the circumcenter q of the top simplex s.
  if s is not a triangle then
    if n_q is a quality point then
      Remove n_q.
    end if
  end if
  Update the priority queue.
end while
```

4 Results

First, the algorithm is shown to produce meshes with the same desirable properties as Ruppert's algorithm: the resulting Delaunay triangulation conforms to the input PLC, and the output is graded to the local feature size. These first two theorems mirror the standard results for Ruppert's algorithm.

Theorem 1. *For $\kappa < \arcsin\left(\frac{1}{4\sqrt{2}}\right)$, Algorithm 2 terminates. Moreover, there exists $C_\kappa > 0$ such that for each vertex q inserted into the mesh, $\mathrm{lfs}(q) \leq C_\kappa r_q$.*

Proof. This claim will be shown by induction over a number of different cases corresponding to different insertions into the mesh. Our goal is to find C_1, C_2, and C_3 such that for each point p inserted into the mesh,

$$\mathrm{lfs}(p) \leq \begin{cases} C_1 r_p & p \text{ is a circumcenter} \\ C_2 r_p & p \text{ is an input point or a “type 1” midpoint} \\ C_3 r_p & p \text{ is a “ type 2” midpoint} \end{cases}$$

In the above cases, an inserted midpoint p is considered a “type 2” midpoint if

- p is inserted as the midpoint of segment s,
- n_p was inserted as a circumcenter (which is removed),
- the second nearest neighbor to p is an endpoint of s, and
- the circumcenter n_p is older than both endpoints of s.

Otherwise, p is called a “type 1” midpoint.

Let $C_\kappa := \max(C_1, C_2, C_3)$. We will soon see than $C_\kappa = C_3$.

First, consider a circumcenter p inserted into the mesh. Let q be the newer point on the shortest edge of the triangle which is being split.

$$\begin{aligned} \mathrm{lfs}(p) &\leq \mathrm{lfs}(q) + |p - q| \\ &\leq C_M r_q + r_p \\ &\leq (2C_M \sin\kappa + 1) r_p \end{aligned}$$

So, we need $C_1 > 2C_M \sin\kappa + 1$ in order for the desired bound to hold.

Next, consider an input point or midpoint p inserted into the mesh such that the nearest neighbor to p, possibly after deleting a circumcenter, is on a disjoint feature from an input feature containing p. Then $\mathrm{lfs}(p) \leq r_p$. So, we need $C_2 > 1$.

Consider an input point or midpoint p which is inserted such that the nearest two neighbors to p (including the one that is deleted) are both circumcenters. Let q denote the older of these two circumcenters.

$$\begin{aligned} \mathrm{lfs}(p) &\leq \mathrm{lfs}(q) + |p - q| \\ &\leq C_1 r_q + r_p \\ &\leq (2C_1 + 1) r_p \end{aligned}$$

In this case, we require $C_2 \geq 2C_1 + 1$.

Continuing, consider the case that a midpoint p of a segment is inserted such that the encroaching point q on the segment is newer than one of the endpoints of the segment q'.

$$\begin{aligned}
\text{lfs}(p) &\leq \text{lfs}(q) + |p - q| \\
&\leq C_1 r_q + r_p \\
&\leq C_1 |q - q'| + r_p \\
&\leq (2C_1 + 1) r_p
\end{aligned}$$

This gives the same requirement on C_1 as was found on the previous case.

Finally, we reach the case of the "type 2" midpoint insertion. Let p be a type 2 midpoint. Notice that this implies that both endpoints of the segment being split are either type 1 midpoints or input points. Let q denote the endpoint which is nearer to the encroaching point q'.

$$\begin{aligned}
\text{lfs}(p) &\leq \text{lfs}(q) + |p - q| \\
&\leq C_2 r_q + r_p \\
&\leq C_2 |q - q'| + r_p \\
&\leq (\sqrt{2} C_2 + 1) r_p
\end{aligned}$$

So the important inequality in this case is that $C_3 \geq \sqrt{2} C_2 + 1$. Combining the required inequalities, it follows that such C_1, C_2 and C_3 exist if

$$(4\sqrt{2} C_1 + 2\sqrt{2} + 2) \sin \kappa_1 \leq C_1$$

Thus, appropriate constants exist if $\sin \kappa < \frac{1}{4\sqrt{2}}$. This bound ensures termination of the algorithm and completes the proof. □

Ruppert's algorithm allows a little more flexibility with the minimum angle parameter, allowing any $\sin \kappa < \frac{1}{2\sqrt{2}}$. The next result simply states that the algorithm successfully generates a quality conforming mesh.

Theorem 2. *The triangulation produced by Algorithm 2 conforms to the input PLC and contains no angles less than κ.*

Proof. When Algorithm 2 terminates, no triangles are unacceptable. This implies both that the resulting mesh conforms to the input and that no poor quality triangles remain. □

Finally, the degree of each Delaunay triangulation is bounded throughout the duration of the algorithm. A similar result in [2] relies on ensuring no small angles occur at any point during the algorithm and used this quality bound to imply the degree bound. While Algorithm 2 allows arbitrarily small angles to occur in intermediate Delaunay triangulations, it is still possible to compute and explicitly bound the degree of these triangulations.

Theorem 3. *There exists D depending only on κ such that the degree of the Delaunay triangulation at any step during Algorithm 2 is bounded by D.*

Proof. The first part of the proof is to show that when points are inserting during Algorithm 2, new triangles which are formed do not contain large angles.

Claim. When a point p is inserted into the mesh, no angle of any triangle containing p is larger than $180 - \kappa$.

Let p be a point inserted by the algorithm. Let t be a triangle in the Delaunay triangulation with p as a vertex. Suppose t contains an angle γ larger than $180 - \kappa$. Let q and q' be the other two vertices of t. In the Delaunay triangulation preceding the insertion of p, q and q' are vertices of a triangle $\bar{t}$ of at least radius R_t, by Proposition 1.

Observe the following string of inequalities.

$$R_{\bar{t}} \geq R_t = \frac{\mathrm{LE}(t)}{2\sin\gamma} \geq \frac{|q - q'|}{2\sin\kappa} \geq \frac{\mathrm{SE}(\bar{t})}{2\sin\kappa}$$

This implies that $\bar{t}$ is a poor quality triangle. This means that p could not be inserted for conformity, as the quality split to add the circumcenter of $\bar{t}$ would get higher priority on the queue. Thus p must have been inserted as the circumcenter of some poor-quality triangle with radius r_p.

$$r_p \leq |p - q| \leq \mathrm{LE}(t) \leq 2R_{\bar{t}}\sin\kappa < R_{\bar{t}}$$

The last inequality requires $\kappa < 30°$, which is a much weaker restrict than was needed in Theorem 1. This now contradicts the priority of the queue, as a triangle smaller than $\bar{t}$ must have split before the poor-quality triangle $\bar{t}$. Conclude that no triangle with angle larger than $180 - \kappa$ is formed when inserting a point into the Delaunay triangulation.

Next, the degree of the triangulation is bounded throughout the algorithm. Proposition 3 ensures that immediately before any conformity insertion, the degree of the triangulation is bounded by $\frac{360}{\kappa}$ (since the triangulation has no small angles). Next, we consider the triangulation immediately after a point is inserted for conformity.

Claim. Following the insertion of a conformity point, the degree of the Delaunay triangulation is at most $\frac{360}{\kappa} + 1$.

Conformity points are only inserted into a quality triangulation so for any point other than the newest point in the mesh, the degree if bounded by $\frac{360}{\kappa} + 1$ by Proposition 3. Let p be the point added to the mesh. Consider any triangle t in the Delaunay triangulation containing p and let s be the line segment opposite p in this triangle. If the angle subtended by s is less than κ, then by Proposition 1, s belonged to a poor quality triangle immediately before p was inserted. This then violates the priority queue which performs priority splits before conformity splits. Since no angle at p is less than κ, the degree of p is lat most $\frac{360}{\kappa}$.

By Proposition 4,observe that following the (possible) deletion of a nearby circumcenter point, the degree of the Delaunay triangulation is at most $\frac{720}{\kappa} - 2$.

For any point q in the triangulation, consider the sequence of angles with vertex q. The degree bound at all steps will follow by showing that the algorithm does not create two consecutive angles with sum less than θ, where θ is defined by

$$\theta = \arctan\left(\frac{\sin\kappa}{2 + \cos\kappa}\right)$$

This immediately leads to a degree bound on the triangulation which is summarized in the next claim.

Claim. Following the insertion of any triangle circumcenter, the degree of the Delaunay triangulation is at most $D = \frac{720}{\theta} + \frac{360}{\kappa} + 1$.

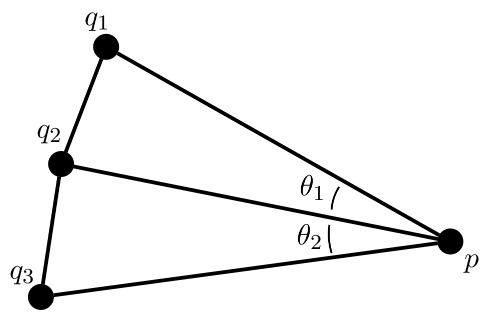

Figure 1: If the degree of p is large, then p is a vertex of two adjacent triangles with small angles at p. These triangles exist such that $\theta_1 + \theta_2 < \theta$.

This will be shown by contradiction. Let p be the first point with degree larger than D. Let q_c be the last point inserted into the triangulation for conformity before the first step at which p has degree larger than D. Consider the triangles containing p at the step when the degree of p has become large. There are at least $\frac{360}{\kappa} + 1$ pairs of adjacent angles on the cavity which sum less than θ as in Figure 1. For one of these pairs of triangles, let q denote the point of q_1, q_2 and q_3 which was added most recently to the mesh and let q' be a neighbor to q from the set $\{q_1, q_2, q_3\}$. Let q'' denote the unlabeled element of this set.

Now, there exists a pair of triangles as before such that q is inserted after q_c. Why is this the case? Suppose that q is inserted before q_c. The circumcircle of q,q', and p cannot be empty immediately before the insertion of q_c, otherwise a quality split with be given priority over the conformity split. But this circumcircle is empty later when the degree of p is large so conclude that a point q_r must be removed following the insertion of q_c which was inside this circle. Now q_r is the only point in the circle so q_r and q must be neighbors preceding the insertion of q_c.

The previous argument holds for every possible q which was inserted before q_c. But the degree of the triangulation is bounded by $\frac{360}{\kappa}$ when q_c is inserted so q_r can have at most $\frac{360}{\kappa}$ neighbors. This means that at least one such q was inserted after q_c. From this point onward, consider a q which was inserted after q_c.

Consider the Delaunay triangulation immediately preceding the insertion of q into the mesh. For purposes of analysis, consider adding p to this triangulation. Then q' and q'' are Delaunay neighbors of p and each are contained in triangles with small angles at p (since $\angle q'pq'' \le \theta < \kappa$). Considering the empty circumball of one of these triangles, conclude that when q is inserted, q' belongs to a skinny triangle t of circumradius at least $\frac{|q'-p|}{2}$.

Now, using the claim bounding angles from above by $180 - \kappa$, $|q - q'|$ can be bounded in terms of $|q' - p|$ using the law of sines.

$$\begin{aligned} |q - q'| &= |q' - p| \frac{\sin(\angle qpq')}{\sin(\angle pq'q)} \\ &< |q' - p| \frac{\sin \theta}{\sin(\kappa - \theta)} \\ &= \frac{|q' - p|}{2} \end{aligned}$$

The value of θ was specifically chosen such that $\frac{\sin \theta}{\sin(\kappa - \theta)} = \frac{1}{2}$, which gives the final equality.

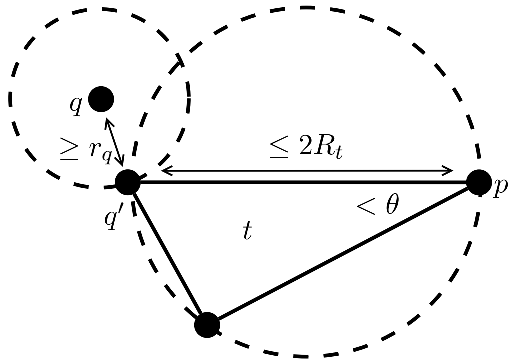

Figure 2: Diagram of the final steps of the proof. The earlier bound on the maximum angle in the mesh when a point is inserted is essential for ensuring that finding that r_q is less than R_t.

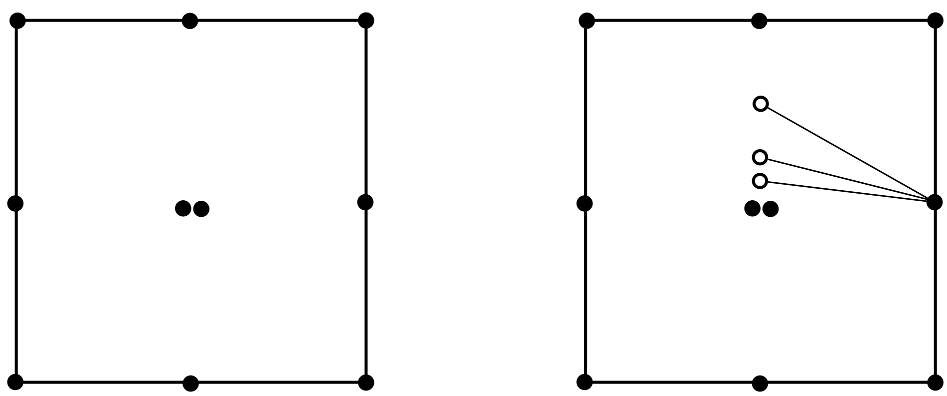

Figure 3: Smallest first priority queue on triangles does not lead to a degree bound on intermediate triangulations.

Now a contradiction has been achieved. Point q must be a circumcenter (as it is inserted after the last conformity pint). Moreover, q must be the circumcenter of a skinny triangle of radius at most $|q - q'| < \frac{|q'-p|}{2}$, but q' belongs to another skinny triangle of radius at least $\frac{|q'-p|}{2}$. This violates the priority queue requiring the largest triangle to be split first.

Conclude that the degree bound of $\frac{720}{\theta} + \frac{360}{\kappa} + 1$ holds for all points in the mesh at any step of the algorithm. $\square$

Only Theorem 3 relies on the specific order of the triangles in the priority queue in the algorithm: the other results hold as long as triangles are processed before segments. Algorithm 2 does not lead to a degree bound if the smallest triangles are processed first. This can be seen with an explicit example. Consider a mesh of a box containing the midpoints of the segments of the box centered at the origin. In addition, consider adding the points $(\epsilon, 0)$ and $(-\epsilon, 0)$. Using a smallest first priority queue leads to an intermediate step in which the point $(1, 0)$ has a degree of $\Theta(\log \frac{1}{\epsilon})$. See Figure 3.

Typically, θ is very near, but slightly less than $\frac{\kappa}{3}$. For $\kappa = 10.2$, the corresponding θ value is $3.4°$. This leads to a bound on the degree of 248. Compared to a rough lower bound on the maximum degree needed of $360/10.2 \approx 35$, the bound in the proof is within a

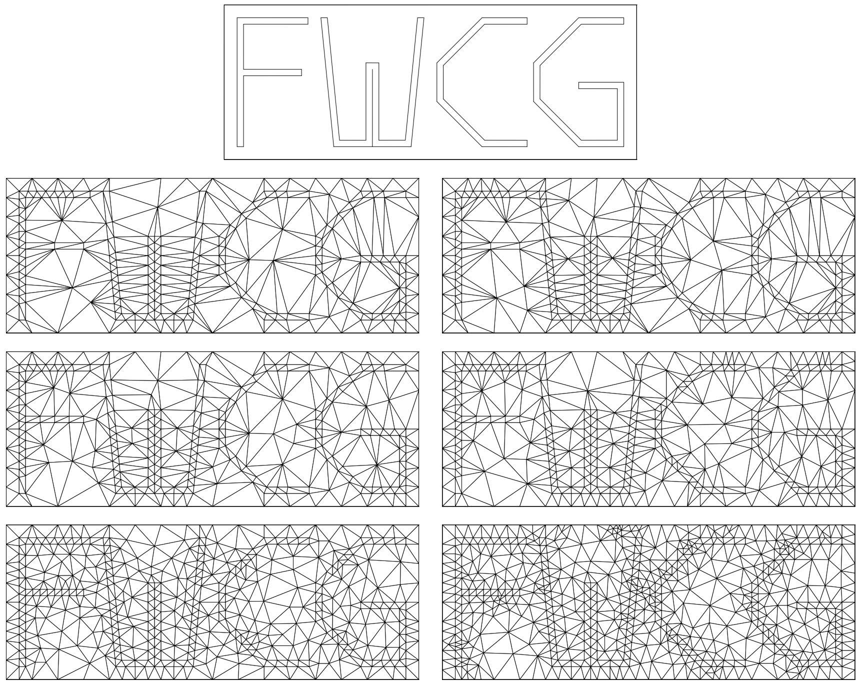

Figure 4: (top) An input PLC. (left) Output of Ruppert's algorithm using $\kappa = 10°$, $15°$, and $25°$. (right) Output of Algorithm 2 using $\kappa = 10°$, $15°$, and $25°$. While Theorem 1 only ensures that Algorithm 2 terminates for κ less than $10.2°$ (compared to $20.7°$ for Ruppert's algorithm), Algorithm 2 terminates in practice for higher κ values.

κ	Algorithm 1	Algorithm 2
5°	266	266
10°	284	297
15°	312	366
20°	352	477
25°	473	647

Table 1: Number of points in the resulting example meshes for different minimum angles.

factor of ten of optimal (and likely much closer than that). Still, the constant in the proof is surely not sharp and the sharp value will lie somewhere in between.

Figure 4 gives an example of a mesh refined using Algorithm 2 and Ruppert's algorithm for several different κ values. Table 1 contains the number of vertices in the resulting meshes.

5 Extensions

This algorithm can be extended to higher dimensions with a slight modification. Simplices queued for mesh quality are processed at a higher priority than those for mesh conformity. As in the 3D extension of Ruppert's algorithm [3], insertions for conformity are prioritized by dimension with the lowest dimension handled first. As in the SVR algorithm [2], insertions for quality are prioritized by dimension with the highest dimension handled first.

This modification also enlarges the allowable range for the minimum angle parameter to $\kappa < \arcsin(\frac{1}{4})$. Ruppert's algorithm still provides a wider range for this minimum angle parameter, accepting any $\kappa < \arcsin(\frac{1}{2\sqrt{2}})$.

We expect that the degree bound on intermediate triangulations will also hold if the largest first priority queue on triangles is replaced with a first in-first out queue. In this case, it is likely that the explicit degree bound on the triangulations is weaker.

We hope to integrate this approach with techniques for applying Delaunay refinement to domains with acute angles [4, 5].

References

[1] L. P. Chew. Guaranteed-quality mesh generation for curved surfaces. In *Symposium on Computational Geometry*, 1993.

[2] B. Hudson, G. Miller, and T. Phillips. Sparse Voronoi refinement. In *15th International Meshing Roundtable*, 2006.

[3] G. Miller, S. Pav, and N. Walkington. Fully incremental 3D Delaunay refinement mesh generation. In *11th International Meshing Roundtable*, 2002.

[4] G. Miller, S. Pav, and N. Walkington. When and why Ruppert's algorithm works. In *12th International Meshing Roundtable*, 2003.

[5] A. Rand and N. Walkington. 3D Delaunay refinement of sharp domains without a local feature size oracle. In *17th International Meshing Roundtable*, 2008.

[6] J. Ruppert. A Delaunay refinement algorithm for quality 2-dimensional mesh generation. *J. Algorithms*, 18(3):548–585, 1995.