Minimal enclosing parallelogram with application

Christian Schwarz 1  Jürgen Teich 1  Alek Vainshtein 2  Emo Welzl 3  Brian L. Evans 4

We show how to compute the smallest area parallelogram enclosing a convex n-gon in the plane in linear time, and we describe an application of this result in digital image processing.

Related work has been done on finding a minimal enclosing triangle, see e.g. [OAMB86], a minimal enclosing rectangle [FS75], a minimal enclosing k-gon [ACY85], and a minimal enclosing k-gon that has sides of equal lengths or a fixed-angle sequence [DA84]. Note that whereas e.g. a rectangle would be contained in the latter class of polygons, our problem is different since the angles of the desired enclosing parallelogram are not given in advance. Nevertheless, our method clearly borrows from the techniques developed in the computational geometry literature, and our contribution is to show how these methods can help to obtain the result as requested by the application. In fact, we learned that the linear time algorithm has been previously published in a Russian journal, [Vai90].

There are two key facts which lead to the algorithm. First, let us consider the edges e1, e2, e3 and e4 of an enclosing parallelogram (in counterclockwise order), and let l1, l2, l3 and l4, respectively, be their supporting lines. Then there is an optimal enclosing parallelogram which has at least one of the edges e1 and e3 flush with an edge of the convex polygon, and also one edge of e2 and e4 flush with an edge of the polygon. There are at most n pairs of parallel tangents to an n-gon, where at least one supports an edge. This already leads to an O(n^2) algorithm.

A second stronger condition for any optimal parallelogram reads as follows: There is a line l parallel to l1 (and so to l4), which intersects the polygon in two points touched by edges e2 and e4. Similarly, a symmetric statement holds for a line parallel to l2 and l3. This excludes many pairs of directions for the edges of an optimal parallelogram. There are only a linear number of such combinations possible, and we can scan through these in a “rotating calipers”-fashion in linear time.

The implementation showed that, in fact, the linear-time algorithm is not substantially more difficult to implement than the quadratic algorithm. Furthermore, the linear-time algorithm is faster than the quadratic algorithm for even the smallest problem size of n = 5.

The application that motivated this research is compressing two-dimensional signals (e.g. images) based on their frequency content [ETS94]. Specifically, we are interested in designing rational decimation systems that reduce the number of input samples by a rational factor. A rational decimation system extracts a specific portion of the frequency content (the passband) and resamples the resulting signal at its Nyquist rate. Rational decimation systems are realized by a cascade of four linear operators—modulator, upsampler, filter, and downsampler. In the two-dimensional case, the modulation factor n0, the upsampling matrix L, the filter passband specifications, and the downsampling matrix M can be computed directly from any parallelogram that circumscribes the passband and has vertices which are rational multiples of π. Therefore, in order to optimize the compression ratio of the overall system, we need to find the parallelogram of minimal area that circumscribes the passband.

Our design procedure takes the vertices of the desired passband and returns the decimator system parameters n0, L, and M. The vertices would be sketched with a mouse, typed in, or defined by mathematical formulas. The design algorithm amounts to (1) snapping the vertices of the desired passband to grid points that are rational multiples of π, (2) finding the convex hull of the rational vertices, (3) computing the minimal enclosing parallelogram, and (4) calculating the design parame-
The theoretical upper limit on the compression ratio, computed as the ratio of $4 \pi^2 \over 2$ over the area of the original polygon, is $10 \over 9$ to 1, which is 5.55-to-1.

Packing efficiency by the parallelogram is 80%. The compression ratio is 40-to-9 (4.44-to-1).

The full details of the algorithms to find the minimal enclosing parallelogram are available as a technical report [STWE94]. Details of the digital signal application may be found in [ETS94].

Figure 1: Automatic Design of a Rational Decimation System

References


