Minimal enclosing parallelogram with application*

Christian Schwarz[†] Jürgen Teich[‡] Alek Vainshtein[§] Emo Welzl[¶] Brian L. Evans[||]

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 e_1, e_2, e_3 and e_4 of an enclosing parallelogram (in counterclockwise order), and let l_1, l_2, l_3 and l_4, respectively, be their supporting lines. Then there is an optimal enclosing parallelogram which has at least one of the edges e_1 and e_3 flush with an edge of the convex polygon, and also one edge of e_2 and e_4 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 l_1 (and so to l_3), which intersects the polygon in two points touched by edges e_2 and e_4. Similarly, a symmetric statement holds for a line parallel to l_2 and l_4. 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 those 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 n_0, 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 n_0, 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-

*This research has been carried out while C. Schwarz and E. Welzl were at the International Computer Science Institute (ICSI), Berkeley, CA, USA, and while J. Teich was at the Dept. of Electrical Eng. and Computer Sci., Univ. of California, Berkeley, USA.

†Max-Planck-Institut f. Informatik, Im Stadtwald, D-66123 Saarbrücken, Germany. E-mail: schwarz@mpi-sb.mpg.de

‡Institut TIK, ETH Zürich, Gloriastr. 35, CH-8092 Zürich, Switzerland. E-mail: teich@tik.ethz.ch

§School of Math. Sci., Tel Aviv Univ., Tel Aviv, Israel. E-mail: alek@math.tau.ac.il

¶Freie Univ. Berlin, Inst. f. Informatik, Takustr. 9, D-14195 Berlin, Germany. E-mail: emo@inf.fu-berlin.de

||Dept. of Electrical Eng. and Computer Sci., Univ. of California, Berkeley, CA, 94720, USA. E-mail: ble@eecs.berkeley.edu; supported by the Ptolemy Project, which is funded by the ARPA RASSP Program (F33615-93-C-1317), SRC (94-DC-008), NSF (MIP-9201605), ONR, the MICRO Program, and several companies.

```
{n0, L, M} =
  DesignDecimationSystem2D[
    sketchedPolyVertexList,
    Justification -> All, Mod -> 10 ];
```

The theoretical upper limit on the compression ratio, computed as the ratio of 4 π^2 over the area of the original polygon, is $\frac{50}{9}$-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).

$$n_0 = \begin{bmatrix} \frac{-3\pi}{10} \\ \frac{2\pi}{5} \end{bmatrix} \quad L = \begin{bmatrix} 3 & 0 \\ -48 & 3 \end{bmatrix} \quad M = \begin{bmatrix} 5 & -2 \\ -80 & 40 \end{bmatrix}$$

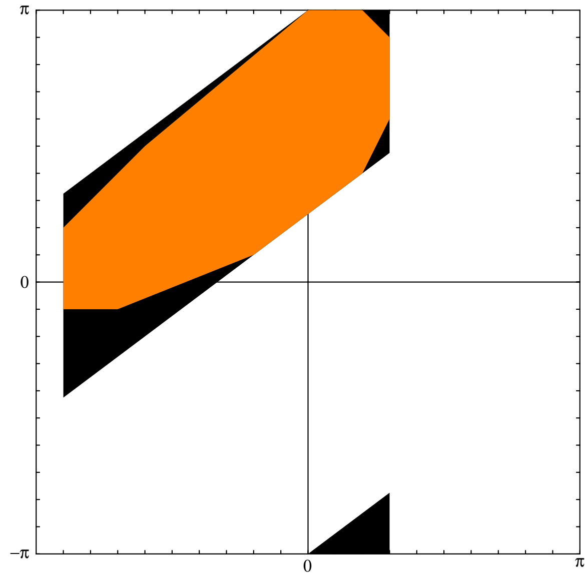

Figure 1: Automatic Design of a Rational Decimation System

ters. Figure 1 shows the automatic design of a rational decimation system from a user's sketch of the desired passband based on our implementation in the Mathematica symbolic mathematics environment. The figure shows one $2\pi \times 2\pi$ period of the frequency domain, which includes most of one circumscribing parallelogram (top) and a small piece of its replica from the period below (bottom).

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].

References

[ACY85] A. Aggarwal, J. S. Chang, and C. K. Yap. Minimum area circumscribing polygons. *The Visual Computer: International Journal of Graphics*, 1:112–117, 1985.

[DA84] A. DePano and A. Aggarwal. Finding restricted k-envelopes for convex polygons. In *Proc. 22nd Allerton Conf. on Comm. Control and Computing*, pp. 81–90, 1984.

[ETS94] B. L. Evans, J. Teich, and C. Schwarz. Automated design of two-dimensional rational decimation systems. In *IEEE Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, Oct. 31 - Nov. 2, 1994, pp. 498-502. Available by FTP to `ptolemy.eecs.berkeley.edu` and by World Wide Web at URL `http://ptolemy.eecs.berkeley.edu`.

[FS75] H. Freeman and R. Shapira. Determining the minimum-area encasing rectangle for an arbitrary closed curve. *Communications of the ACM*, 18:409–413, 1975.

[OAMB86] J. O'Rourke, A. Aggarwal, S. Maddila, and M. Baldwin. An optimal algorithm for finding enclosing triangles. *J. Algorithms*, 7:258–269, 1986.

[STWE94] C. Schwarz, J. Teich, E. Welzl, and B. Evans. On finding a minimal enclosing parallelogram. Tech. Rep. TR-94-036, International Computer Science Institute, 1947 Center Street, Suite 600, Berkeley, CA, Aug. 1994. Available by gopher at `gopher.icsi.berkeley.edu` and by World Wide Web at URL `http://www.icsi.berkeley.edu`.

[Vai90] A. Vainshtein. Finding minimal enclosing parallelograms. *Diskretnaya Matematika* 2:72–81, 1990. In Russian.