Iterated Snap Rounding with Bounded Drift

Eli Packer

Department of Computer Sciense, Stony Brook University

November 2, 2005

Abstract

Snap Rounding and its variant, Iterated Snap Rounding, are methods for converting arbitrary-precision arrangements of segments into a fixed-precision representation (we call them SR and ISR for short). Both methods approximate each original segment by a polygonal chain, and both may lead, for certain inputs, to rounded arrangements with undesirable properties: In SR the distance between a vertex and a non-incident edge of the rounded arrangement can be extremely small, inducing potential degeneracies. In ISR, a vertex and a non-incident edge are well separated, but the approximating chain may drift far away from the original segment it approximates. We propose a new variant, Iterated Snap Rounding with Bounded Drift, which overcomes these two shortcomings of the earlier methods.

1 Introduction

Finite-Precision-Approximation is an approach in the area of robust geometric computing whose idea is to convert the representation of the input into finite-precision, such that it guarantees more robustness for algorithms that work on it. Snap Rounding is a well known Finite-Precision-Approximation method, mainly devoted to arrangements of segments in the plane [2]. It converts input segments into polygonal chains in the following way. The plane is tiled with a grid of unit squares, *pixels*, each centered at a point with integer coordinates. A pixel is *hot* if it contains a vertex of the arrangement. The output of each segment s is a polygonal chain through the centers of the hot pixels met by s, in the same order as they are met.

Two major advantages of the Snap Rounding is that the deviation of the input is very small and the topology is preserved. However, Halperin and Packer [3] showed that a vertex of the output may be extremely close to a non-incident edge, inducing potential degeneracies. They proposed an augmented procedure, Iterated Snap Rounding, that is aimed to eliminate this undesirable property. It rounds the arrangement differently from SR, such that any vertex is at least half-the-width-of-a-pixel away from any non-incident edge. ISR, however, may round segments far from their origin. A rounding tight bound of $\Theta(n^2)$ pixels is presented in [4].

Our contribution. We propose a new algorithm, Iterated Snap Rounding with Bounded Drift (ISRBD, for short), which rounds the segments such that both the distance between a vertex and a non-incident edge is at least half-the-width-of-a-pixel, and the approximation is bounded as a user parameter. Thus, it has none of the undesirable properties mentioned above. The algorithm is a simple and efficient modification of ISR. It has demonstrated good performance (both theoretically and in practice) and has produced efficient output. Thus, we believe that ISRBD should be the option to choose when a snap-rounded-like arrangement of segments is required.

2 Algorithm

We use the following notations throughout the abstract. $S = \{s_1, s_2, \ldots, s_n\}$ is the set of input segments. Let ε be the size of the pixel in the grid. Let $s \in S$ be a segment in the input. For each hot pixel h and segment s, let $\Delta(h, s)$ be the right triangle whose hypotenuse lies on the line containing s and its right angle on the center of h. We denote the output of s by $\lambda(s)$.

The main idea of our work is to bound the approximation of the output. Let δ be this bounded distance, given as a parameter. For each segment, s, δ defines a domain, $D(s, \delta)$ which is the area that is δ-close to s, in which we allow s to be approximated.

Our goal, therefore, is to restrict $\lambda(s)$ to $D(s, \delta)$. We show in the full paper that there are two thin strips to the left and to the right of $D(s, \delta)$ (we denote this area by $F(s)$, the *forbidden loci* of s) such that if $\lambda(s)$ drifts from $D(s, \delta)$, there is a hot pixel h whose center is contained in $F(s)$, which is the cause for rounding s beyond $D(s, \delta)$. Hence, it is sufficient to make sure that s will never snap to the center of h. We note without a proof that $\lambda(s)$ is weakly-monotone. It follows that s never snaps to the center of h if we make sure that there is at least one hot pixel centered within or on the triangle $\Delta(h, s)$, such that the segment connecting its center to the center of h is not

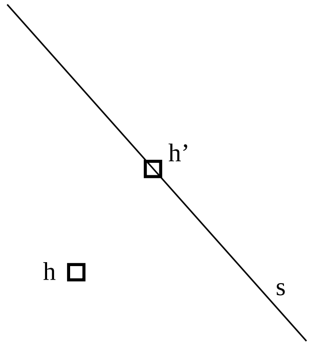


Figure 1: Heating the pixel h' prevents s from snapping to hot pixel h.

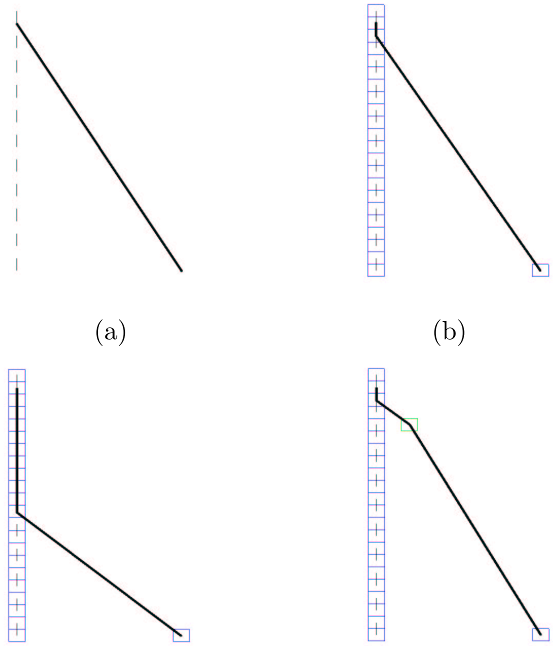
orthogonal. We do that by heating a pixel such that this condition is satisfied (in the full paper we describe this process in detail). See Figure 1 for an illustration. Since the output can be illustrated as a continuous deformation of pixels [2], heating pixels may change the output, but not the topology. This process may cascade as a new hot pixel may lie inside a forbidden loci of another segment.

Our algorithm follows this discussion. The idea is to heat pixels iteratively until all hot pixels in the forbidden loci of segments are blocked from rounding them by the way described above. As a result, the output drift is bounded as required. (We show in the full paper that δ must satisfy $\delta > 1.5\varepsilon$, a magnitude that should be satisfactory for any purpose.) In the last step of our algorithm, we remove redundant degree-2 vertices (slightly differently from [1]). We devise a routine, *ProduceNewHotPixels* which performs all we described above. *ProduceNewHotPixels* is plugged in the ISR algorithm to produce ISRBD. ISRBD produces segments which lie inside the Minkowski sum of the corresponding original segments with a disc of radius δ, centered at the origin. In other words, the drift magnitude of any segment is bounded by δ.

Figure 2 illustrates the differences among SR, ISR and ISRBD where the example demonstrates large drift. The interesting segment which demonstrates the differences is shown in bold. Notice that in the SR output, the polygonal chain of the bold input segment penetrates a hot pixel but does not pass through its center. Thus, generating a possibly non-robust situation (Figure 2(b)). The outcome is not satisfactory with the ISR as well, since the same segment output drifts far (figure 2(c)). Both undesirable features are eliminated when ISRBD is applied (Figure 2(d)).

3 Experimental Results

We implemented and tested the algorithm with many examples under different parameters. The examples demonstrated similar behavior which we next discuss. The number of new hot pixels produced in *ProduceNewHotPixels* was small, which is a positive indication since one generally prefers less hot pixels and shorter output links. The extra time required for ISRBD did not increase the total processing time greatly. We observed an increase of $25 - 30\%$. Another observation is that the majority of the pixels heated by ProduceNewHotPixels are redundant degree-2 vertices which can be removed safely, improving the quality of the output.



(a) (b) (c) (d)

Figure 2: Results of SR, ISR and ISRBD (all squares represent hot pixels): (a) Input segments (b) SR output (c) ISR output (d) ISRBD output

References

[1] M. de Berg, D. Halperin, and M. Overmars. An intersection-sensitive algorithm for snap rounding. Manuscript.

[2] L. Guibas and D. Marimont. Rounding arrangements dynamically. *Internat. J. Comput. Geom. Appl.*, 8:157–176, 1998.

[3] D. Halperin and E. Packer. Iterated snap rounding. *Computational Geometry: Theory and Applications*, 23(2):209–225, 2002.

[4] E. Packer. Finite-precision approximation techniques for planar arrangements of line segments. In *Master's thesis, Dept. Comput. Sci., Tel-Aviv Univ.*, 2002.