Algorithms for Clipping Quadratic Arcs

V. Skala

Abstract

New algorithms for 2D quadratic arcs clipping against convex and non-convex windows are presented. Algorithms do not use parametric equations for the arcs description. Therefore only a square root function is needed. The algorithms require information how edges of the given window are oriented. The design, implementation and verification is the first step toward to the quadratic arcs usage in computer graphics as new basic primitives. The presented algorithms have not been published in a literature known to the author.

Keywords: Clipping, Quadratic Arcs, Algorithms

1. Introduction

Clipping is a very important part of all graphics packages. There are many efficient algorithms as Cyrus (1979), Liang (1984), Nicholl (1987) for clipping lines against convex windows or for clipping lines against a window with holes and with the non - linear boundaries, see van Vyk (1984), Skala (1989). Unfortunately no one known algorithm deals with a problem how to clip circles or ellipses against a window. This problem is a fundamental one if we want to introduce quadratic arcs as a basic primitive for 2D graphics packages. The below described algorithms enable to clip general quadratic curves or arcs against convex or non-convex window.

2. Clipping by a Convex Area

Provided a convex area is given by its vertices in the clockwise order and a oriented circle given by its center $\mathbf{x}_w$ and radius r. We want to find those parts of the given circle which lie inside of the given convex window. For easier understanding the following notation will be used:

$\mathbf{x}_w$ circle center $\mathbf{x}_i$ polygon vertex

$\mathbf{x}_k$ instead of $\mathbf{x}_{i+1}$; the + is meant as modulo n addition

T.-S. Chua et al. (eds.), *CG International '90*

$s_i = x_i - x_{i-1}$ $\qquad s_k = x_k - x_i$

${}^i s_w = x_w - x_i$

$\mathbf{t} = [\, y_w - y \,,\, x - x_w \,]^T$ tangent vector at the point (x,y)

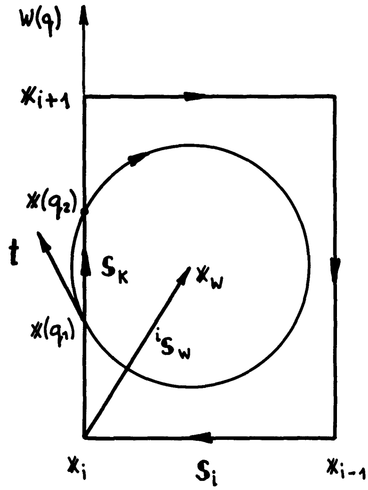

Figure 2.1.

To solve this problem it is necessary to find intersection points of the circle and line w(q) on which the window border lies, i.e. to solve the following equations:

$$\mathbf{x}(q) = \mathbf{x}_i + (\mathbf{x}_{i+1} - \mathbf{x}_i) \cdot q$$

$$(x - x_w)^2 + (y - y_w)^2 - r^2 = 0$$

Solving these equations with regard to the variable q the quadratic equation

$$a\, q^2 + b\, q + c = 0$$

will be obtained, where:

$$a = |\, s_k \,|^2 \qquad b = -2\, {}^i s_w^T \cdot s_k \qquad c = |\, {}^i s_w \,|^2 - r^2$$

In the case that the line w(q) intersects or touches the given circle two solutions are generally obtained that are not necessarily different:

$$q_{1,2} = \frac{-b \pm \sqrt{b^2 - 4\,ac}}{2\,a}$$

The obtained values q_1, q_2 must be ordered so that $q_1 \leq q_2$. For the next processing only points for $q \in \langle 0 \,,\, 1 \,)$ will be considered. Of course some possible situations must be distinguished, see fig. 2.2. For a general case, when $q_1 \neq q_2$, the tangent vector $\mathbf{t}_1$ always points out of the given window while the tangent vector $\mathbf{t}_2$ always points into the given window, see fig. 2.2. cases a and d. Therefore the circular arc $\mathbf{x}(q_1)\ \mathbf{x}(q_2)$ cannot be considered for further processing. The cases b and c from fig. 2.2. are a little bit more complicated because the