Bresenham’s Line Generation Algorithm with Built-in Clipping

Yevgeny P. Kuzmin
IBS Graphics Technologies, 46/2 Dmitrovskoje sh., Moscow, 127238, Russia
kuzmin@dealine.msk.ru

Abstract
One of the most important operations in many graphical systems is the generation of a line segment. This process consists of two stages: clipping and drawing. These two stages are separated in current graphical applications. In this paper a new approach to line generation is proposed, which unifies these stages. The proposed algorithm is based on Bresenham’s line generation algorithm to include necessary line clipping. The line clipping stage is an operation-reduced, integer arithmetic only algorithm. The notion of correctness of line clipping is introduced and correctness of the proposed algorithm is shown. Complete C-notation of the algorithm is included.

Keywords: raster graphics, line drawing, clipping.

1. Introduction
One of the most important and fundamental operations in many graphical systems is generation of a line segment. The efficiency of raster device usage strongly depends on the efficiency of this algorithm.

The process of generation of a line segment consists of two stages. The first is clipping of the segment by a given planar rectangle - the window. At this stage that part of the line segment within the window is determined. In the second stage, a raster representation of the clipped segment is generated.

There are three main methods for line clipping:
- encoding clipping;
- divide clipping;
- parametric clipping;

These methods determine the part of the given segment within the given rectangle, by calculation of intersection points of the segment and the window rectangle.

The clipped segment must then be represented by a sequence of pixels, which closely approximate this segment. Usually, segments are drawn by using some modification of Bresenham’s line generation algorithm (BLGA).

In existing line generation processes these two stages are separated. The first stage produces rounded coordinates of the endpoints of the clipped segment. This may produce incorrect results: raster representation of the clipped segment may differ from raster representation of the original unclipped segment within the window.

This may be demonstrated by drawing several randomized unclipped segments, then setting up the clipping area and again drawing these segments in XOR mode. Usually the unerased points will all be seen within the clipping area. This is as a result the inherent error in the line generation process used.

In this paper we propose an efficient integer-based clipping algorithm based on BLGA with correct output.

After introducing the necessary definitions and notions we briefly review existing methods of clipping and drawing. Then we give our interpretation of BLGA, introduce a notion of correctness and describe our proposed method and its implementations. Finally we analyze the proposed method.

The appendix contains the complete C-notation of the proposed algorithm.