# A Note to the Structure of Bresenham Slice Algorithm

T. Hrúz* I. Považan†

December 2, 1997

**Abstract**

This paper gives a complete analysis of the structure of segments (spans) of points with a constant
y-coordinate, which are observed in the Bresenham Slice algorithm. In particular, we show that this structure is related to the greatest common divisor of the end-point coordinates.

## 1 Introduction

It is an old problem to find an appropriate picture of an Euclidean line in raster space. A widely used algorithm was proposed by Bresenham [1] in 1965. Further improvements have since been published (See [2] and references there). The number of operations for algorithms mentioned above is linear with respect to the number of points approximating a line. The recent trends in improvement of line interpolation methods consist in new algorithms with smaller number of iterations [4, 3]. They produce more than one point during an iteration, resulting in smaller total number of operations compared with original Bresenham algorithms. There is also another class of algorithms which relies on certain tables containing structural information about the parts of line approximation [11, 12].

The line segment and its raster space approximation is shown in Figure 1. It is intuitively clear that for a line in the first octant (derivative is from the interval $< 0, 1 >$) a good approximation is obtained when exactly one point of the line segment approximation lies on every vertical line of the grid. Its y-coordinate is the y-coordinate of the nearest grid point from the intersection of the vertical line of the grid and the ideal line.

It is a well known observation that the above approximation of line leads to the structure like in Figure 1, consisting of segments of points with constant y-coordinate. A closer look shows that the segments posses rather regular structure. Roughly described, there is a starting short segment, then a sequence of segments with length $l$ or $l + 1$, and finally a short segment at the end. In this article, the structure of these segments is investigated in detail. In particular, it is shown that this structure is related to the greatest common divisor of the end-point coordinates.

We would like to stress, that there is also a different point of view [14, 13] which gives a similar structural information. In particular, so called Freeman sequences can be used to analyse the structure of those segments. The main difference between our approach and Freeman sequences is that they are defined for infinite lines and as far as we know the greatest common divisor structure has not been observed in this setting. Freeman sequences are mainly used for line recognition which is an inverse problem of the one we are investigating here.

The article is organized as follows: in the next section some preliminary definitions and assumptions are given, in the section 3 we derive an algorithm similar to the original Bresenham Slice algorithm [2]. We do not focus on the optimization of this algorithms but rather we prepare a method and

---

*Ivo Považan, Institute of Control Theory and Robotics, Slovak Academy of Sciences, Dubravská cesta 9, 842 37 Bratislava, Slovak Republic, Phone: +42-7-3782985, e-mail: utrrpova@savba.savba.cs

†Tomáš Hrúz, Slovak Technical University, Faculty of Mechanical Engineering, Department of Automatic Control and Measurement Námestie Slobody 17, 812 31 Bratislava, Slovak Republic, Phone: +42-7-3594-571, 497193 e-mail: hruz@vm.stuba.sk Fax: +42-7-495315

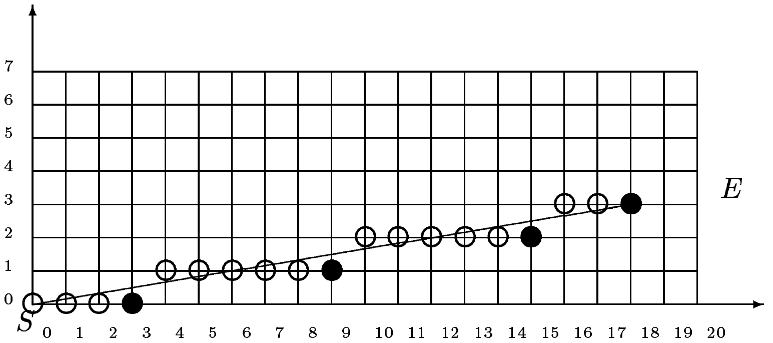

Figure 1:

The transition points are labelled with a bullet. Point $E$ is also a transition point because of a constraint in the inequality (1).

terminology for efficient derivation of the main contribution consisting in the analysis of the structure of segments arising in the Slice algorithm. The main section called Generalized Slice Algorithm starts with some intuitive observation without which the following derivation could look rather "mysterious". Finally, we comment some possible directions of further research.

## 2 Preliminaries

A set $Z^2$ (all integer pairs) is interpreted as embedded in the Euclid plane $R^2$ and is called raster space. The first octant will be a set of ordered pairs $(x, y)$ of integers defined as:

$$FO = \{(x, y) \in Z^2 \subset R^2 \mid 0 \leq y \leq x\}.$$

The basic assumptions of our paper are that line segments under consideration have integral start-points and end-points. Only line segments in the first octant are analysed because a general line segment can be transformed to the first octant with a simple transformation.

An accurate formulation of the line segment approximation is done using a definition of a *digitalization scheme* which is a mapping from Euclidean plane $R^2$ to the set $Z^2$ of all integral pairs. The most common digitalization scheme is a scheme called *mid-point*. In the analysis which follows, we do not need to deal with the formalism of digitalization schemes. What is sufficient, is to know that for the mid-point digitalization scheme the following holds: on each vertical line $x = i$ of raster space lies exactly one raster point, and the y-coordinate of this raster point equals $y$ when for the exact intersection of a given line with the vertical line holds $y - 1/2 < y_{exact} \leq y + 1/2$. The comparison of various digitalization schemes together with the corresponding formalism can be found for example in [5], the basic algorithms in mid-point scheme are also described in [10].

¿From the above basic property of the mid-point digitalization scheme the approximation of a line with starting point $S = (0, 0)$ and ending point $E = (H, V)$ in the form of an inequality is:

$$y - \frac{1}{2} < \frac{V}{H}x \leq y + \frac{1}{2} \tag{1}$$

.

The inequality is solved for all integer solutions $(x, y)$ satisfying the condition $x \in \{0, 1, 2, ...., H - 1, H\}$. The linearity implies that also $y \in \{0, 1, 2, ...., V - 1, V\}$. ¿From this inequality two different systems of inequalities can be derived. The problem can be reformulated to the form of a system of $H + 1$ inequalities enumerated by integral values of $x$ running through the set $\{0, 1, 2, ...., H - 1, H\}$ with integral unknown $y$ or to the form of a system of $V + 1$ inequalities enumerated by integral values of $y$ running through the set $\{0, 1, 2, ...., V - 1, V\}$ with integral unknown $x$. The former variation leads to the Bresenham class of algorithms whereas the latter leads to the Bresenham Slice class of algorithms.

## 3 Bresenham Slice Algorithm

Let us modify the basic inequality (1) to the form of $V + 1$ inequalities:

$$\frac{H}{2V}(2y - 1) < x \le \frac{H}{2V}(2y + 1) \tag{2}$$

where the inequalities are solved with respect to the *integral* unknown $x$ and $y = j$, $0 \le j \le V$. A system of inequalities is obtained:

$$\frac{H}{2V}(2.0 - 1) < x \le \frac{H}{2V}(2.0 + 1)$$

$$\dots$$

$$\frac{H}{2V}(2.1 - 1) < x \le \frac{H}{2V}(2.1 + 1)$$

$$\dots$$

$$\frac{H}{2V}(2.i - 1) < x \le \frac{H}{2V}(2.i + 1)$$

$$\dots$$

$$\frac{H}{2V}(2.V - 1) < x \le \frac{H}{2V}(2.V + 1)$$

With the following new notation, $\frac{H}{2V} = c_0 + \frac{r_0}{2V}$, $\frac{2H}{2V} = c_1 + \frac{r_1}{2V}$, $r_0 = H \bmod 2V$, $r_1 = 2H \bmod 2V$, the system above can be written in the form of $V + 1$ inequalities:

$$c_1 y - c + \frac{r_1 y - r}{2V} < x \le c_1 y + c + \frac{r_1 y + r}{2V}$$

where $y$ is running through the set $\{0, 1, 2, ...., V - 1, V\}$ and at the same time $y$ enumerates the inequalities. This system of inequalities is solved sequentially with respect to the *integral* unknown $x$. It holds that each inequality has at least one integral solution $x$, some of them can have more. First of all, the algorithm searches for the greatest integral element $x$ for each inequality. At the same time the algorithm uses the property that the smallest solution for the $(j + 1) - th$ inequality is exactly the largest solution of the $j - th$ inequality plus one.

We call a *transition point* the mesh point $(x_j, y_j)$ where $x_j$ is the greatest solution for the $j - th$ inequality mentioned above. Transition points are coloured black in figure 1. We stress that the $(V + 1) - th$ transition point has x-coordinate $H$ because of a constraint in the basic inequality (1). This analysis directly leads to the algorithm B in the following picture. In fact the algorithm computes x-coordinates of transition points which are equal to $\lfloor c_1 y + c_0 + \frac{r_1 y + r_0}{2V} \rfloor$ where $y = j$, $0 \le j \le V$.

```
Algorithm B
variable Y – y-coordinate of a transition point
variable STARTX, ENDX – starting and ending x-coordinate
of the horizontal segment related to the transition point
variable MOD – contains $r_1 y + r_0 \bmod 2V$
it is supposed that values $r_0, c_0, r_1, c_1$ are known

1. $initialization, Y = 0; STARTX = 0; ENDX = c_0; MOD = r_0;$
2. $if(horizontal\_segment(STARTX, ENDX, Y) == FALSE)$
       finish;
3. $STARTX = ENDX + 1; ENDX = ENDX + c_1;$
$MOD = MOD + r_1;$
4. $if(MOD \geq 2V)$
       $\{ENDX = ENDX + 1;$
       $MOD = MOD - 2V;\}$
5. $Y = Y + 1;$
6. $goto 2;$

procedure $horizontal\_segment(STARTX, ENDX, Y)$
{
$X = STARTX;$
$while(X \leq ENDX)$
       {
       $output(X, Y);$
       $if(X == H)$
            $return(FALSE);$
       $X = X + 1;$
       }
$return(TRUE);$
}
```

The algorithm B is equivalent to the Bresenham's Slice algorithm [2]. We do not focus here on the character of this equivalence. It can be derived from the theory in [8].

The run-length Slice class of linear interpolation algorithms has been developed and improved in various ways [4]. The improvements lowered the number of iterations as well as the overall number of additions and comparisons.

## 4 Generalized Slice Algorithm

We would like the reader to adopt the following point of view. When $H$ divides $V$, it is a perfect situation where each segment has the same length with the augmentation that the first segment is split to two parts. This splitting of the first segment is not due to the mid-point scheme constant term but due to the constraints coming from the finite length of the line considered. Now, let us suppose that we have a situation that $H = 17$ and $V = 5$. We can interpret this situation as getting a certain defect with a value $H \bmod V = 2$ and the defect must be spread along the line in a linear way. So each segment has a basic length 5 and the defect is spread in way that some of the segments have length 6. One way how to do this, is to imagine the set of segments as points on x-coordinate axis and the defect to be spread is put on the y-coordinate axis. To know the distribution a line is drawn from $(0, 0)$ to $(V, H \bmod V)$ and its raster approximation is considered. As is shown later, the digitalization scheme for the defect distribution is not exactly the mid-point scheme but is very similar. It has the following form:

$$y-\alpha \le \frac{V}{H}x < y+(1-\alpha), 0<\alpha<1$$

However, the Slice algorithm can be easily modified to this scheme, so that the structure of segments in the original problem can be thought as to came in a self-replicating structure of segments of segments. Then, the process can be recursively continued to get segments of segments of segments ... etc. The iteration finishes when the sequence $(H, V, H \bmod V, V \bmod (H \bmod V), \ldots)$ stops. But this is the Euclidean algorithm for the greatest common divisor of numbers $H, V$.

Now, let us turn to the accurate derivation of the above intuitive observation. From this point to the end of the article we adopt a slightly different notation. Instead of $H, V$ the notation $H_0, H_1, \ldots, H_k, \ldots$ is used, where $H_0 > H_1 > H_2 > H_3 > \ldots > H_k > \ldots$. The binary function $x - y \bmod x$ is denoted by $y \operatorname{dom} x$. The derivation starts with an inequality:

$$\frac{H_0}{H_1}y - \frac{H_0}{2H_1} < x \le \frac{H_0}{H_1}y + \frac{H_0}{2H_1} \tag{3}$$

where the inequality is solved with respect to the *integral* unknowns $x, y$ and $0 \le x \le H_0$, $0 \le y \le H_1$. It is advantageous for the moment to formulate the problem once again in the form of *one* inequality. Then inequality 3 can be written as:

$$\lfloor\frac{H_0}{H_1}\rfloor y - \lfloor\frac{H_0}{2H_1}\rfloor + \left(\frac{H_0 \bmod H_1}{H_1}y - \frac{H_0 \bmod 2H_1}{2H_1}\right) < x \le \lfloor\frac{H_0}{H_1}\rfloor y + \lfloor\frac{H_0}{2H_1}\rfloor + \left(\frac{H_0 \bmod H_1}{H_1}y + \frac{H_0 \bmod 2H_1}{2H_1}\right)_{e_1}$$

Previous section implies that it is sufficient to focus only on the right hand side inequality. It is easy to see the integral solutions of the above inequality as far as the error term $e_1 < 1$. Just the value of $y$ is taken and there is a segment of solutions $((x_s, y), (x_s + 1, y), \ldots, (x_s + k, y))$ with length $\lfloor\frac{H_0}{H_1}\rfloor$. As far as the error term is also considered, the situation becomes more complicated, but the integrality conditions on the error term $e_1$ can be expressed in the form of the inequality:

$$z \le e_1 < z+1, \qquad 0 \le z \le H_0 \bmod H_1$$

after suitable renaming of variables, reformulation and substitution $H_2 = H_0 \bmod H_1$:

$$\frac{H_1}{H_2}y - \frac{H_0 \bmod 2H_1}{2H_2} \le x < \frac{H_1}{H_2}y + \frac{H_0 \operatorname{dom} 2H_1}{2H_2}$$

where the inequality is solved with respect to the *integral* unknowns $x, y$ and $0 \le x \le H_1$, $0 \le y \le H_2$. When $H_0 \bmod 2H_1$ is denoted by $H_1'$ and $H_0 \operatorname{dom} 2H_1$ by $H_1''$ the following inequality is obtained:

$$\frac{H_1}{H_2}y - \frac{H_1'}{2H_2} \le x < \frac{H_1}{H_2}y + \frac{H_1''}{2H_2} \tag{4}$$

Clearly this inequality has the same form as the inequality 3, but on the other hand it does not represent mid-point digitalization scheme any more. However, the difference is only in the constant term so the Slice algorithm can be easily adapted, just changing the initial condition. There is also another irregularity, which seems to be unavoidable, being the change of inequality sharpness from the right hand side to the left hand side occurring between the first and the second inequality. The remaining inequalities have the same structure from this point of view.

The construction can be inductively generalized to k steps obtaining:

$$\lfloor\frac{H_0}{H_1}\rfloor y - \lfloor\frac{H_0}{2H_1}\rfloor + \left(\frac{H_2}{H_1}y - \frac{H_0 \bmod 2H_1}{2H_1}\right) < x \le \lfloor\frac{H_0}{H_1}\rfloor y + \lfloor\frac{H_0}{2H_1}\rfloor + \left(\frac{H_2}{H_1}y + \frac{H_0 \bmod 2H_1}{2H_1}\right)_{e_1}$$

$$\ldots$$

$$\lfloor\frac{H_1}{H_2}\rfloor y-\lfloor\frac{H_1'}{2H_2}\rfloor+\left(\frac{H_3}{H_2}y-\frac{H_1' \bmod 2H_2}{2H_2}\right)\le x<\lfloor\frac{H_1}{H_2}\rfloor y+\lfloor\frac{H_1''}{2H_2}\rfloor+\left(\frac{H_3}{H_2}y+\frac{H_1'' \bmod 2H_2}{2H_2}\right)_{e_2} \tag{5}$$

$$\dots$$

$$\lfloor\frac{H_{k-1}}{H_k}\rfloor y-\lfloor\frac{H_{k-1}'}{2H_k}\rfloor+\left(\frac{H_{k+1}}{H_k}y-\frac{H_{k-1}' \bmod 2H_k}{2H_k}\right)\le x<\lfloor\frac{H_{k-1}}{H_k}\rfloor y+\lfloor\frac{H_{k-1}''}{2H_k}\rfloor+\left(\frac{H_{k+1}}{H_k}y+\frac{H_{k-1}'' \bmod 2H_k}{2H_k}\right)_{e_k}$$

$$\dots$$

The result is a system of inequalities with the property that the k-th inequality describes the behaviour of the error term of the (k-1)-th inequality and admissible intervals for $x, y$ subsequently shrink according to the sequence $< 0, H_0 >, < 0, H_1 >, < 0, H_2 >, \dots, < 0, H_k > \dots$. Because the sequence $H_0, H_1, \dots, H_k$ is convergent to the greatest common divisor of the numbers $H_0, H_1$, the process stops in this stage. The last inequality has the following form because $H_k$ divides $H_{k-1}$ and the term $\frac{H_{k+1}}{H_k}y$ vanishes from the error term:

$$\lfloor\frac{H_{k-1}}{H_k}\rfloor y-\lfloor\frac{H_{k-1}'}{2H_k}\rfloor+\left(\frac{H_{k-1}' \bmod 2H_k}{2H_k}\right)\le x<\lfloor\frac{H_{k-1}}{H_k}\rfloor y+\lfloor\frac{H_{k-1}''}{2H_k}\rfloor+\left(\frac{H_{k-1}'' \bmod 2H_k}{2H_k}\right)_{e_k}$$

This inequality has a simple structure, it contains exactly $H_k = gcd(H_0, H_1)$ segments with length $H_{k-1}/H_k$. In fact, it contains $H_k + 1$ segments because the first segment is split to the starting and ending part according to the constant term $\lfloor\frac{H_{k-1}''}{2H_k}\rfloor$.

So the complete description of the structure of segments in the Slice algorithm has been obtained. An immediate result of this analysis is, for example, the answer to the question what are the coordinates which deliver a most complicated structure. They are exactly the worst case of Euclidean algorithm, so the answer is $(H_0, H_1) = (F_n, F_{n+1})$ where $F_n$ denotes Fibonacci numbers ($F_0 = 0, F_1 = 1, F_n = F_{n-1} + F_{n-2}$).

There are many possibilities how to use the above analysis to construct an algorithm. For example, a recursive algorithm can be constructed in a straightforward way. Another possibility is to use a fixed number of levels, say $l$, from the inequality system (5) in the form of embedded cycles and instead of going in the sequence as far as the greatest common divisor is obtained, Slice algorithm can be used to solve the $l + 1 - th$ inequality.

## 5 Conclusions

Even if the analysis provided in this article do not deliver a faster algorithm from known ones in this stage, we suggest that it can be used to augment Slice-like algorithms. The deeper inside into the structure of mid-point digitalization scheme could be also worthwhile for various algorithms which use tables to speed up their function.

There are some immediate directions of possible research leading from this article. We summarize them in the following notes.

- the table can be constructed containing at the place $(i, j), i > j$ values $(j, i \bmod j, \lfloor i/j \rfloor)$. This table contains all information necessary to avoid the division operations in generalized Slice algorithm. Tabular approach can be also combined with approximate division using only leading bits. There are also other methods how to avoid expensive division operation in Slice-like class of algorithms [4].
- because the "gcd" version of algorithm which would use all structural information until the greatest common divisor of $H_0, H_1$ would contain too many expensive divisions, we suggest to take three resp. four levels and to combine them with tables and approximate division to obtain a very fast algorithm.
- there is also a question how to speed up the computation of quantities $H_k', H_k''$ by relating them to the values of $H_k$.

## References

[1] J. E. Bresenham, "Algorithm for computer control of digital plotter," IBM Systems Journal, Vol. 4, No.1, January 1965, pp. 25-30.

[2] J. E. Bresenham, "Incremental Line Compaction," The Computer Journal, Vol. 25, No.1, 1983, pp.116.

[3] J. Rokne and Y. Rao, "Double-Step Incremental Linear Interpolation", ACM Transactions on Graphics, Vol. 11, No. 2, April 1992, pp. 183-192.

[4] K. Y. Fung, T. M. Nicholl and A. K. Dewdney, "Run-Length Slice Line Drawing Algorithm without Division Operations," Proceedings of EUROGRAPHICS '92, Vol. 11, No. 3, 1992, pp. C-267 - C-277.

[5] C. A. Wutrich and P. Stucki, "An Algorithmic Comparison between Square and Hexagonal Based grids," Graphical Models and Image Processing, Vol. 53, No. 4, 1991, pp. 324-339.

[6] W. E. Wright," Parallelization of Bresenham's Line and Circle Algorithms," IEEE Computer Graphics & Applications, Vol. 10, No. 5, September 1990, pp. 60-67.

[7] A. T. Pang, "Line Drawing Algorithms for parallel machines," IEEE Computer Graphics & Applications, Vol. 10, No. 5, September 1990, pp. 54-59.

[8] S. M. Ecker and J. V. Tucker, "Tools for the Formal Development of Rasterisation Algorithms," New Advances in Computer Graphics. Proceedings of CG International 89, 1989, pp. 53-89.

[9] R. F. Sproull, "Using Program Transformation to Derive Line-Drawing Algorithms," ACM Transactions on Graphics, Vol. 1, No. 4, October 1982, pp. 259-273.

[10] James D. Foley, Andries Van Dam, Steven K. Feiner, John H. Hughes, Computer Graphics, Principles and Practise, second edition, Addison Wesley, Reading, Mass., November 1991.

[11] N. D. Butler, A. C. Gay, J. E. Bresenham. Line Generation in a Display System, US patent 4,996,653, Feb. 26, 1991.

[12] Graeme W. Gill, N - Step Incremental Stright-Line Algorithms. IEEE Computer Graphics & Applications, Vol.10, No. 5, May 1994, pp. 66-72.

[13] P. Sauer, "On the recognition of digital circles in linear time ," Computational Geometry: Theory and Applications, Vol. 2, 1993, pp. 287-302.

[14] H. Freeman, "Boundary encoding and processing," B.S.Lipkin and A. Rosenfeld, eds., Picture Processing and Psychopictorics, Academic Press, New York, 1970, pp. 241-266.