Layout of Non-Planar Venn Graphs with Geometric Constraints

Jacques Eloff
eloff@cs.sun.ac.za

Department of Computer Science
University of Stellenbosch

June 17, 1999

Abstract

This paper describes algorithms for the automatic layout of non-Planar Venn graphs that are constructed using binary composition for the labeling of vertices. The binary composition is also used to describe intersections between the diagrams in the Venn diagram. The Venn graph is then placed onto a grid within the boundaries imposed by certain geometric constraints.

Contents

1 Introduction **4**

2 Venn Diagrams **4**

2.1 Introduction 4

2.2 Definitions and Applications 4

3 Non-Planar Venn Graphs using Binary Composition **5**

3.1 Introduction 5

3.2 Binary Logical Expressions Associated with Venn Diagrams 5

4 Constructing Non-Planar Venn Graphs **6**

4.1 Introduction 6

4.2 Algorithms for Non-Planar Venn Graphs 6

4.3 An Example Venn Graph 7

5 Drawing Venn Diagrams Using Non-Planar Graphs **8**

5.1 Introduction 8

5.2 Grid Drawings of Venn Graphs using Geometric Constraints 9

5.3 Reducing the Number of Crossings in a Non-Planar Venn Graph 11

6 Conclusion **11**

List of Figures

1 A Venn Diagram and its Associated Planar Venn Graph [4] 4
2 A Typical Venn Diagram 5
3 Venn Graph with Binary Composed Vertex Numbering 8
4 Venn Graph with Binary Composed Vertex Numbering after a Vertex Insertion 8
5 Venn Graph with Binary Composed Vertex Numbering 9
6 4-Venn Graph 10
7 6-Venn Graph with Geometric Boundary Placed on a Grid 11
8 4-Venn Diagram with Alternative Drawing 12

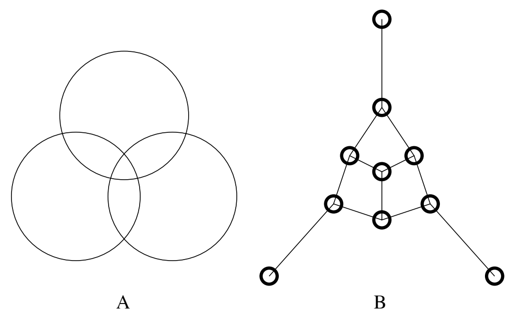

Figure 1: A Venn Diagram and its Associated Planar Venn Graph [4]

1 Introduction

The aim of this paper is to describe methods and algorithms that can be used to visualize Venn diagrams using graph theory as the basis for the layout of these diagrams, and more specific, non-planar graphs.

An important distinction must be made between a Venn diagram $\mathcal{D}(\mathcal{F})$ and its associated planar graph representation $\mathcal{V}(\mathcal{F})$. Little attention is given to the planar representation since this paper focus on non-planar graphs

Section 3 describes a method for a binary composition scheme that is used to label the vertices of a Venn graph. These labels also describe the intersections in the Venn diagram. Section 4 applies this technique in order to construct a Venn graph which can then be used to draw a Venn diagram. The technique for drawing the Venn graph is discussed in Section 5.

2 Venn Diagrams

2.1 Introduction

The aim of this section is to give an overview of Venn diagrams and the planar graph that is used to represent the Venn diagram and to define certain basic concepts for the reader.

2.2 Definitions and Applications

Figure 1 A is an example of a typical simple 3-Venn diagram. A simple Venn diagram is defined as a graph where no more than two curves intersect at any point [4]. The Venn diagram in Figure 1 contains six vertices (each vertex corresponds to an intersection) and twelve edges. Euler's formula relating to the number of faces, edges and vertices for a graph embedded in the plane is given by $f = e - v + 2$ and holds for graphs of simple Venn-diagrams. The associated Venn graph is given by Figure 1 B.

Although originally designed to represent logical expression, Venn diagrams are typically used to teach set arithmetic. But, Venn diagrams have also been applied in other, more practical areas such as computerized design and geometric modeling [3]. It is therefore important that suitable means are found for displaying Venn diagrams.

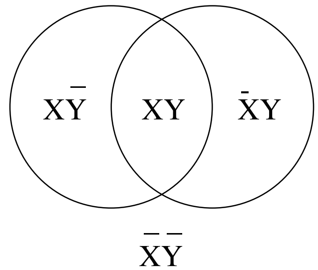

Figure 2: A Typical Venn Diagram

3 Non-Planar Venn Graphs using Binary Composition

3.1 Introduction

Literature such as *Chilakamarri, et. al.* [2] treat Venn graphs as planar graphs. The object of this paper as stated earlier is to present the Venn graph as a non-planar graph rather than a planar one. The method described in this section relies heavily on the fact that Venn diagrams are used to represent logical expressions and that these logical expressions each have a unique binary number. The Venn diagram in Figure 2 illustrates the logical expressions visually.

3.2 Binary Logical Expressions Associated with Venn Diagrams

This section describe some basic algorithms and functions that are used in deriving a binary representation of the intersections of the Venn graph with the plane. These functions are later used in Section 4 to construct a Venn graph.

We start by defining a set called $\mathcal{S}$ (the diagram set) that has as its elements the faces or logical symbols of the Venn diagram. The diagram set for Figure 2 is given by $\mathcal{S} = \{X, Y\}$.

For every n-Venn diagram we have 2^n intersections with the plane [2]. Every intersection can be expressed as a logical expression. The intersection $X \cap Y$ can be written as XY for instance. Binary numbers can be used to describe logical expressions by mapping every element in $\mathcal{S}$ unto a unique binary number.

We know that the number of binary digits required to describe an integer n is given by the function $\log_2 n$. If we substitute n in $\log_2 n$ with 2^n (the total number of intersections in the plane), we have $\log_2 2^n$ which can be reduced to n by applying elementary logarithmic rules. From this, we can derive the following conclusion: *For every n-Venn diagram, n binary digits are required to describe every intersection in the plane.*

We now state that for every $s_i \in \mathcal{S}$, where s_i represents the i -ith element in $\mathcal{S}$, there is a unique decimal number that, when converted into binary notation, yields a unique binary digit that describes s_i. The function $\beta(s_i) = 2^{i-1}$ can be used to determine the decimal number associated with every $s \in \mathcal{S}$. We can illustrate a trivial proof by way of example.

Example 1

Let $\mathcal{S} = \{X, Y\}$ represent the diagram set for a 2-Venn diagram such as the one in Figure 2. Then $s_1 = X$ and $s_2 = Y$ with $\beta(s_1) = 1$ and $\beta(s_2) = 2$. The binary representation is given by 01_2 and 10_2 respectively, ie. X is associated with the binary number 01_2 and Y with 10_2. It can also be noted that the conclusion derived earlier still applies since only two binary digits are used to describe the 2-Venn diagram.

By using logical operations every intersection with the plane can be described. The value 0 is always associated with the exterior face of the Venn graph. Referring to the above example and Figure 2, the exterior face described by $\neg X \neg Y$ is represented by 00_2 while the intersection $X \cap Y$ is described by 11_2.

4 Constructing Non-Planar Venn Graphs

4.1 Introduction

This section describes the algorithms involved in the construction of a non-planar Venn graph. At the end of this section, an illustrated example will be given to assist the reader in understanding the algorithm.

4.2 Algorithms for Non-Planar Venn Graphs

Recall that in Section 3.2 we defined the diagram set $\mathcal{S}$ and an associated function called $\beta(s_i)$. We now define a set of labels L_β containing all the $\beta(s_i)$ values for $1 \leq i \leq n$ where $n = \#\mathcal{S}$ (the number of elements in $\mathcal{S}$) so that for every s_i there exists a $l_i \in L_\beta$. $L_e = \{0\}$ and is the label of the vertex that references the exterior face. $\mathcal{G}(\mathcal{V})$ denotes the non-planar Venn graph that will be constructed for a given n-Venn diagram.

Algorithm 1 is used to create the initial Venn graph of $\mathcal{G}$ containing only vertices with labels from L_β.

Algorithm 1

1. Let $j = 1$ and $\mathcal{G}$ be an empty graph containing no edges or vertices so that $\mathcal{G} = (\phi, \phi)$
2. While $j < n$ Do
   - Create a vertex v_j labeled as l_j
   - Insert v_j into $\mathcal{G}$ as follows:
     - If $\mathcal{G}$ is empty, then $\mathcal{G}$ contains only the vertex v_j and no edges.
     - If $\mathcal{G}$ is not empty, then v_j is inserted after v_{j-1} so that an edge is formed between v_j and v_{j-1}
   - Increment j
3. The loop guard ($j < n$) guarantees that when the *While* loop terminates, $j = n$. We now create a cycle by inserting the vertex v_n and connecting it to v_{n-1} and v_1

Before the second algorithm used in the construction can be given, a few more important sets and functions need to be defined. Let L denote the set of all labels excluding L_e. We also define a set called the *remainder* set denoted by L_r where $L_r = L \setminus L_\beta$. L_r is the set difference between L and L_β. A function $\alpha(v)$ is defined that returns a label for the vertex v.

We also define a function $I(l_a, l_b)$ that returns the label for a vertex describing the intersection of two vertices labeled as l_a and l_b. If $(l_a \vee_b l_b \neq l_a) \wedge (l_a \vee_b l_b \neq l_b)$ then $I = l_a \vee_b l_b$ else $I = 0$, where $\vee_b$ is a binary OR and not a logical OR operation.

An insertion function $P(g, l_1, l_2, v_k)$ is defined which takes a vertex v_k and inserts it into a graph g if g contains a pair of vertices v_x and v_y with labels l_1 and l_2 respectively.

If an edge exists between v_x and v_y, then v_k is placed so that the edge connecting v_x and v_y is removed and an edge is formed between v_x and v_k and v_y and v_k. If no such an edge exists, a new edge is added to v_x that connects to v_k and another edges is placed in v_k that connects to v_y.

Algorithm 2

1. Let $\mathcal{G}' = \mathcal{G}$
2. While $L_r \neq \phi$ Do
   - Let m be the number of vertices in $\mathcal{G}$
   - Let $i = 1$
   - While $i < m$ Do
     - Let $j = i + 1$
     - While $j \leq m$ Do
       (a) Let v_i and v_j denote the i-th and j-th vertex in $\mathcal{G}$ respectively
       (b) Let $l = I(\mathcal{G}, \alpha(v_i), \alpha(v_j))$
       (c) If $l \neq 0$ and $l \in L_r$ Then
         * Remove l from L_r
         * Create a vertex w with label l
         * $P(\mathcal{G}', \alpha(v_i), \alpha(v_j), w)$
       (d) Increment j
     - Increment i
   - Let $\mathcal{G} = \mathcal{G}'$

Upon closer examination of Algorithm 2 it can be seen that the two nested while loops compute all possible combinations of vertex pairs v_i and v_j. This is done in order to compute all possible intersections for the Venn diagram.

4.3 An Example Venn Graph

An example will be given of how to construct the non-planar Venn graph $\mathcal{G}(\mathcal{V})$ for a specific n -Venn diagram.

Example 2

The example presented here will deal with a 3-Venn diagram. The figures used in the example will be based on the final representation of the Venn graph after the methods discussed in Section 5 have been applied.

We define the diagram set $\mathcal{S} = (X, Y, Z)$ and construct $L_\beta = \{1, 2, 4\}$. After applying Algorithm 1, a graph similar to the one in Figure 3 will be obtained.

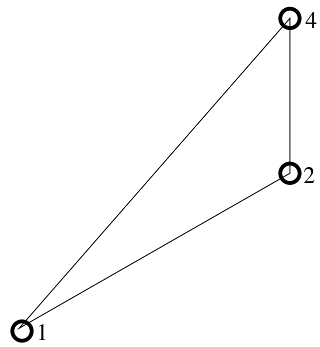

Figure 3: Venn Graph with Binary Composed Vertex Numbering

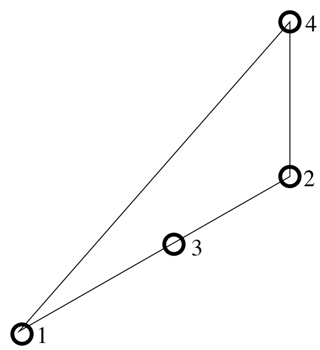

Figure 4: Venn Graph with Binary Composed Vertex Numbering after a Vertex Insertion

Before applying Algorithm 2, we compute $L_r = \{3, 5, 6, 7\}$. For $i = 1$ and $j = i + 1$ vertices v_1 and v_2 will be used to compose the label for the next vertex. $l = 3$ and $l \in L_r$ so the following steps are performed:

- l is removed from L_r so that $L_r = \{5, 6, 7\}$.
- Vertex w is created with label l so that $\alpha(w) = 3$
- w is inserted between v_i and v_j.

Figure 4 illustrates what $\mathcal{G}'$ will look like after vertex w has been inserted into the graph. Upon completion of the algorithm $L_r = \phi$ and $\mathcal{G}$ will look similar to the graph in Figure 5.

5 Drawing Venn Diagrams Using Non-Planar Graphs

5.1 Introduction

Battista, et. al. [1] describes graph drawing as a method to *address the problem of constructing geometric representations of graphs, etc.*. Graph drawing in general sets many goals, and more

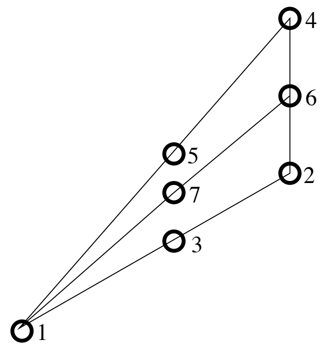

Figure 5: Venn Graph with Binary Composed Vertex Numbering

often than not, these goals implies trade-offs to be made. An example of such a trade-off is the fact that a graph which occupies a minimum area is not always aesthetically pleasing.

The following sub-sections will discuss an algorithm that was designed to deal with non-planar Venn graphs on a grid that is confined by certain geometrical constraints imposed on the graph.

5.2 Grid Drawings of Venn Graphs using Geometric Constraints

An algorithm will now be given for placing the Venn graph $\mathcal{G}$ that was constructed in Section 4 onto a grid. It is very important to note that for a n-Venn diagram where $n < 4$ the algorithm presented in this section still yields a planar graph.

The Venn graph $\mathcal{G}$ is drawn on a grid under certain geometric constraints. These constraints are introduced in order to obtain the following goals:

- Vertices are placed within the boundaries of the geometric shape in order to minimize the area of the graph on the grid
- Where possible, to create a symmetrical graph to present an aesthetically pleasing layout

Figure 6 is an example of a 4-Venn diagram that was drawn using Algorithm 3 presented in this section.

Algorith 3 takes a Venn graph $\mathcal{G}$ as input and returns a graph $\mathcal{G}'$ with its vertices mapped unto a grid. The only other parameter for the algorithm is the degree of the Venn diagram from which $\mathcal{G}$ was constructed. The algorithm relies on certain values that were calculated in Section 4.

Let MAP represent a grid of $M_x \times M_y$ where M_x and M_y are the maximum x and y positions on the grid respectively. A position on the grid is indicated by $MAP[x, y]$. $MAP[x, y]$ can assume the values 0 or 1. A 0 indicates that the position on the grid is not occupied while a 1 indicates that the position is occupied by some vertex. Let N be the degree of the Venn diagram.

Algorithm 3

1. Let $\theta = 360° \div N$ if $N \neq 3$ else $\theta = 60°$

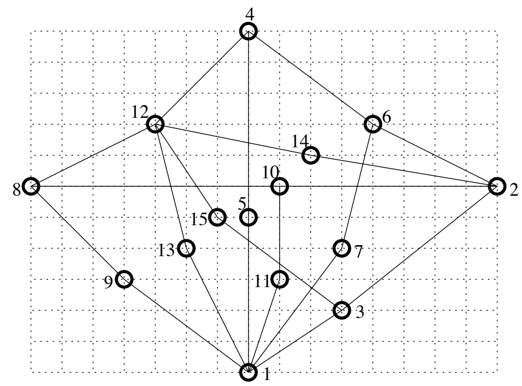

Figure 6: 4-Venn Graph

2. No vertex in $\mathcal{G}$ contains any coordinates. This is indicated by the fact that for any vertex v in $\mathcal{G}$, $v_x = -1$ and $v_y = -1$
3. Let $i = 1$, $a = 0^\circ$ and $\mathcal{G}' = \mathcal{G}$
4. Let $M_{px} = M_x \div 2$ and $M_{py} = M_y \div 2$
5. MAP initially has all positions set to 0 to indicate an empty grid
6. While $i \leq N$ Do
   - Locate vertex v in $\mathcal{G}'$ so that $\alpha(v) = l_i$ where $l_i \in L_\beta$
   - Calculate the vertex grid coordinate as follows: $v_x = 1 + M_{px} + \sin(a) M_{px}$ and $v_y = 1 + M_{py} + \cos(a) M_{py}$
   - Increment a by θ
   - $MAP[v_x, v_y] = 1$
7. Let $i = 1$
8. Let m be the number of vertices in $\mathcal{G}'$
9. While $i \leq m$ Do
   - Locate vertex w_i in $\mathcal{G}'$
   - If $w_{i_x} = -1$ Then
     - Let v_1 be a vertex that has an edge connected to w_i
     - Let v_2 be a vertex such that w_i has an edge connected to v_2
     - If both v_1 and v_2 contains grid coordinates, the grid coordinates for w_i is calculated as $w_{i_x} = v_{1_x} + (|v_{1_x} - v_{2_x}\ | \div 2)$ if $v_{1_x} < v_{2_x}$ else $w_{i_x} = v_{2_x} + (|v_{1_x} - v_{2_x}\ | \div 2)$ and $w_{i_y} = v_{1_y} + (|v_{1_y} - v_{2_y}\ | \div 2)$ if $v_{1_y} < v_{2_y}$ else $w_{i_y} = v_{2_y} + (|v_{1_y} - v_{2_y}\ | \div 2)$
     - If either v_1 or v_2 does not have an assigned coordinate, then $v_{1_{xy}}$ can be substituted by 1 and $v_{2_{xy}}$ by M_{px} and M_{py} respectively

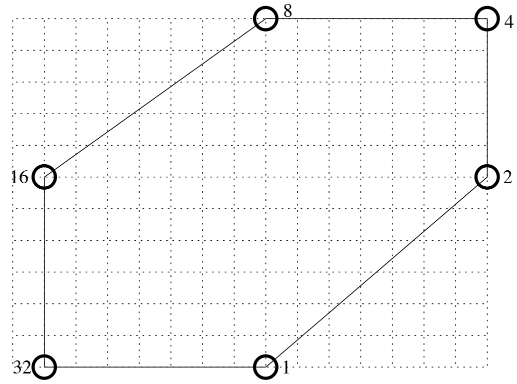

Figure 7: 6-Venn Graph with Geometric Boundary Placed on a Grid

  - If $MAP[w_{i_x}, w_{i_y}]$ is occupied, the closest position to w_{i_x}, w_{i_y} is located and the coordinates are assigned to w
  - $MAP[w_{i_x}, w_{i_y}] = 1$
- Increment i

The first part in the algorithm uses the diagram set to create the bounding geometrical shape. Figure 7 illustrate the boundary mapped onto the grid by the first *While* loop in Algorithm3. All subsequent vertices are placed inside this boundary area on the grid.

5.3 Reducing the Number of Crossings in a Non-Planar Venn Graph

It is possible to reduce the number of crossing in a non-planar Venn Graph and minimize the area while improving symmetry at the same time. This can be done by allowing redundant binary compositions and by using the boundary shape as the center of the graph except for a few exceptions. The method for creating such a non-planar Venn graph will only be briefly discussed as it is basically the inverse of the algorithms presented earlier.

As in the preceding algorithms, the diagram set $\mathcal{S}$ is used to connect the boundary of the Venn graph on the grid, except that the boundary that is formed is the center of the graph.

We assume that we have two pairs of two vertices in a 4-Venn diagram labeled 1 and 8 and 8 and 4. The expression $1 \vee_b 8$ describes $1 \cap 8$ and a vertex labeled 9 is centered on the outside of 1 and 8. $8 \vee_b 4$ describes $4 \cap 8$ and a vertex labeled 12 is centered on the outside of 8 and 4. If $9 \vee_b 12$ is calculated and redundant binary digits are removed, we obtain a vertex labeled 13 which is centered on the outside of 9 and 12 and describes $1 \cap 4 \cap 8$. If we apply this algorithm in general, we can obtain a highly symmetrical non-planar graph with very few crossings. Figure 8 is an example of a 4-Venn graph drawn using the method described above.

6 Conclusion

The algorithms presented is of little practical importance, but serve as a good indication as to why planar Venn graphs are preferred. No investigation was done as to how the non-planar

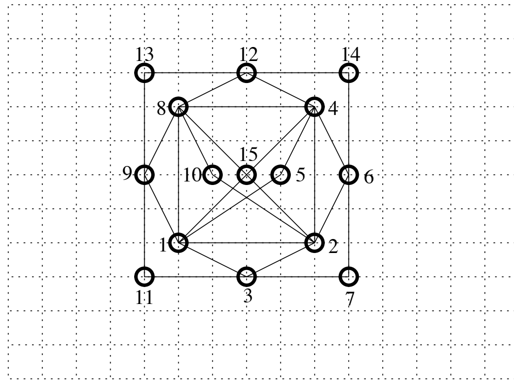

Figure 8: 4-Venn Diagram with Alternative Drawing

Venn graph will affect the actual drawing of the Venn diagram.

The non-planar graph can of course be converted into a planar graph so that standard algorithms can be applied to draw the Venn diagram, but this is extremely expensive in terms of execution time (Up to $O(n^4)$ fictitious vertices can be introduced [1] which will influence the final drawing to a considerable extent). Even if execution time is not important, it will still be more efficient to construct a planar Venn-graph before drawing a Venn diagram instead of using a non-planar representation.

References

[1] Giuseppe Di Battista, Peter Eades, Roberto Tamassia, Ioannis G. Tollis. *Graph Drawing-Algorithms for the Visualization of Graphs.* Prentice Hall, 1999.

[2] Kiran N. Chilakamarri, Peter Hamburger, Raymond E. Pippert. *Venn Diagrams and Planar Graphs.* Geometriae Dedicata. Kluwer Publishers, 1996.

[3] Peter Hamburger, Raymond E Pippert. *Simple Reducible Venn Diagrams on Five Curves and Hamiltonian Cycles.* Geometriae Dedicata. Kluwer Publishers, 1997.

[4] *Graphs Associated with Venn Diagrams.*
http://sue.csc.uvic/~cos/venn/VennGraphEJC.html