On Symmetry in Venn Diagrams and Independent Families

By

Mark Richard Nicholas Weston
B.Sc., University of Victoria, 2000

A Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

in the Department of Computer Science

We accept this thesis as conforming
to the required standard

Dr. F. Ruskey, Supervisor (Department of Computer Science)

Dr. W. Myrvold, Departmental Member (Department of Computer Science)

Dr. D. Roelants van Baronaigien, Departmental Member (Department of Computer Science)

Dr. G. MacGillivray, Outside Member (Department of Mathematics and Statistics)

Dr. B. Grünbaum, External Examiner (Department of Mathematics, University of Washington)

University of Victoria

Supervisor: Dr. F. Ruskey

ABSTRACT

An n-Venn diagram is a collection of n coloured simple closed curves in the plane intersecting only at points. There must be 2^n faces defined by the curves, each of which contains a unique set of interiors of the curves. An independent family of n curves is defined similarly except that a subset of curve interiors may be contained by two or more faces. A Venn diagram or independent family is symmetric if it can be constructed by rotating a single closed curve in the plane n times, each time by $2\pi/n$, and changing the curve colour for each rotation. Though symmetric n-Venn diagrams exist only for n prime, symmetric independent families exist for composite n. Via a backtracking technique to construct a "pie-slice" of $1/n$th of the diagram, we enumerate symmetric Venn diagrams for $n = 5$ and 7, construct minimal symmetric independent families for n up to 10, and investigate near-Venn diagrams, which are a natural maximal type of diagram smaller than Venn diagrams. We also create non-symmetric Venn diagrams when n is a prime power that have several nice properties.

Examiners:

Dr. F. Ruskey, Supervisor (Department of Computer Science)

Dr. W. Myrvold, Departmental Member (Department of Computer Science)

Dr. D. Roelants van Baronaigien, Departmental Member (Department of Computer Science)

Dr. G. MacGillivray, Outside Member (Department of Mathematics and Statistics)

Dr. B. Grünbaum, External Examiner (Department of Mathematics, University of Washington)

CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGMENTS

I would like to gratefully acknowledge the support, assistance, patience, and good humour of my supervisor Frank Ruskey – without his help this work would never have been started, let alone completed.

Drs. Ruskey and MacGillivray each helpfully provided a number-theoretic proof on short notice.

I also wish to thank my family, friends and Cathy, who learned more about Venn diagrams than they ever wanted to know.

1. Introduction

Venn diagrams form an important class of combinatorial objects used in set theory and many applied areas, and are often taught at the grade-school level as a method of understanding simple logic. Grünbaum was a pioneer in formalizing the study of Venn diagrams [5], and he and other researchers recently began to study them as mathematical objects worthy of study in their own right. An up-to-date dynamic survey is maintained by Ruskey [12].

Symmetric Venn diagrams are beautiful objects that have recently begun to be studied, and while interest lies primarily in their aesthetic value, they have many interesting combinatorial properties that we will use. Until recently, symmetry in Venn diagrams has only been considered in the case of n being prime, which naturally motivates the question: can we expand on the traditional (so far as tradition can exist in a field so young) notions of symmetry in Venn diagrams?

The contributions of this thesis are three-fold and are based on developing a backtracking technique to generate ”pie-slices” of symmetric independent families. We first consider symmetric independent families for composite values of n and close several open cases of a conjecture of Grünbaum by providing examples of such diagrams for $n = 8, 9$, and 10. Second, we apply the notion of symmetry to diagrams with fewer regions than Venn diagrams, termed “near-Venn” diagrams, and provide examples of such diagrams that are symmetric. Third, we introduce a type of diagram with an interesting structural symmetry that can be constructed when n is not prime but rather a prime power. This construction depends heavily on the construction developed by Savage, et al., in a recent result proving that symmetric n-Venn diagrams exist for all prime n [14].

Chapter 2 provides some definitions and preliminary notions about Venn diagrams and symmetry that are used throughout. In Chapter 3 we outline a backtracking method that can be used to produce both symmetric Venn diagrams and symmetric independent families and near-Venn diagrams, by creating a special labelled planar graph termed a "doodle" from which the entire diagram can be reconstructed. We examine the properties of the doodle and describe our results using this method.

Chapter 4 describes a new type of Venn diagram, termed "pseudo-symmetric", that is not symmetric but has some interesting properties related to symmetry. We examine some properties of these diagrams and then explain a method of constructing these diagrams using doodles. Finally we use the construction from [14] to create doodles that can be used to create pseudo-symmetric diagrams.

Chapter 5 concludes by discussing some further areas of research.

2. Background

Venn diagrams were introduced to popular scientific literature by John Venn in 1880 [16], but were used primarily in applications of set theory until the last thirty years, when research began to focus on them as objects of interest in themselves. We follow Grünbaum's [5] and Ruskey's [12] definitions.

2.1 Basic Definitions Regarding Venn Diagrams

A *simple closed curve* is a curve that is closed and does not intersect itself. Let $\mathcal{C} = \{C_1, C_2, \ldots, C_n\}$ be a family of n simple closed curves in the plane. We say that $\mathcal{C}$ is an *independent family* if all of the 2^n regions $X_1 \cap X_2 \cap \cdots \cap X_n$ are non-empty, where each set X_i is either the bounded interior or the unbounded exterior of the curve C_i. Additionally, $\mathcal{C}$ is a *Venn diagram*, or *n-Venn diagram*, if each region is connected. Note that the term "Venn diagram" is often used informally to refer to diagrams with fewer than 2^n regions, for example in using them to illustrate set inclusion and exclusion, but here we exclusively use the term as defined. The definitions and properties we consider in this chapter can all, unless otherwise noted, be applied to independent families in general.

Each of the 2^n regions in a Venn diagram can be uniquely labelled by a subset of $\{1, 2, \ldots, n\}$, with the members of that subset corresponding exactly to the indices of the curves that contain that region. We often find it convenient to refer to a region's label as a string of bits of length n where the bit at position i is set to 1 if and only if curve i encloses that region. The *weight* of the region is the number of curves containing it, or the number of 1's in the corresponding bitstring.

We define some terminology regarding strings to aid in manipulating the bit-strings labelling regions. A *necklace* is an equivalence class of strings under rotation, so, for example, 01101 is the same necklace as 10101 as they differ by a rotation of two positions. We call a string x of length n a *necklace representative* if we select it as a unique representative from the class of at most n strings equivalent to x under rotation; in other literature necklace representatives are often chosen to have special properties (*i.e.* the string with the lexicographically smallest rotation) but for our purposes this is not the case.

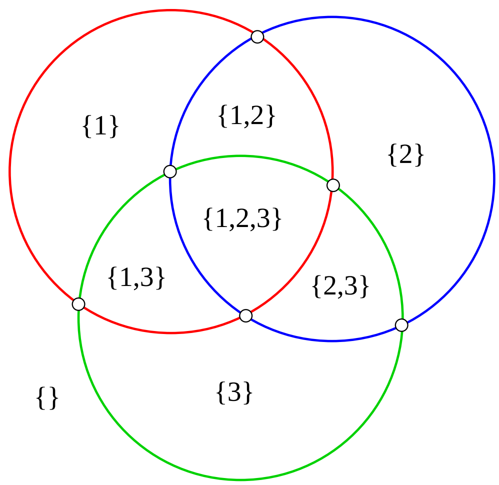
At times the rotational period of the string is important; if a necklace representative does not map onto itself by rotation of some proper factor of n (*i.e.* is aperiodic), then it is termed a *Lyndon word*, so a *non-Lyndon word* is a periodic necklace representative. If n is prime then all necklace representatives are Lyndon words except for 0^n and 1^n.

2.2 Planar Graphs

We define some basic graph terms as independent families are often referred to in a graph-theoretic sense. A *graph* $G = (V, E)$ is a set of vertices $V = \{v_1, v_2, \ldots, v_n\}$ and a set of edges $E = \{e_1, e_2, \ldots, e_m\}$ where each edge is a two-element subset of V. A graph is a *labelled* graph if each edge and/or vertex has a label associated with it. If $e_i = \{v_j, v_k\}$, vertices v_j and v_k are termed *adjacent*. A graph H is a *subgraph* of G if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$.

A graph G is *planar* if it can be drawn in the plane so that its edges do not intersect except at shared vertices; all graphs that we use are assumed to be planar unless obvious from context or stated otherwise. An independent family can be viewed as a labelled graph embedded in the plane by treating curve intersections as vertices and segments of curves between intersections as edges; regions in the diagram thus correspond to faces in this planar graph. A face in the diagram can be called a *k-face* if the face is bounded by exactly k edges. An edge of this planar graph is labelled by the colour of the curve it is a segment of. We overload the terms "independent family" (and similarly "Venn diagram" if the original diagram is one) to also refer to this graph; it will be clear from context which interpretation of the diagram we are using. Note that this interpretation demands that the number of intersection points in the diagram be finite, an assumption that is often not stated in the literature but which we take to be true always.

Figure 2.1a shows a 3-Venn diagram with 3 curves, 6 vertices, 12 edges, and 8 regions. Each region in each graph in Figure 2.1 is labelled by the set of curves that contain it, given that the curves are labelled as red = 1, blue = 2, and green = 3.



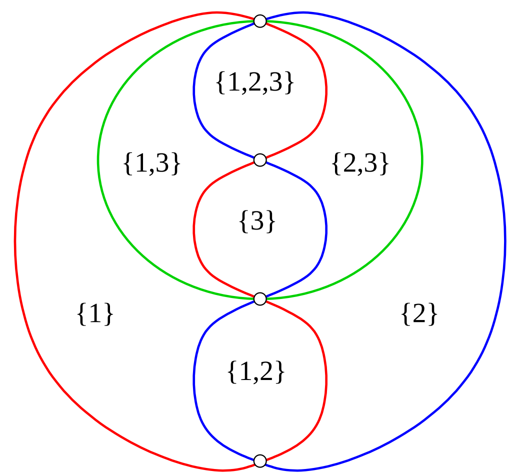


Figure 2.1: Two Venn diagrams with 3 curves.

2.3 Venn Duals

A *plane* graph is an embedding of a planar graph in the plane; every plane graph has a unique plane *dual graph*. For a Venn diagram, the *Venn dual* is constructed as follows: a vertex (also called a *node* in this context) is placed in each region of the Venn diagram and labelled with the subset of $\{1, 2, \cdots, n\}$ enclosed by that region; from our discussion above we can also label them with bitstrings of length n. Edges are drawn between a pair of these vertices exactly when the labels of these vertices differ by the inclusion or exclusion of one curve i, with $1 \leq i \leq n$, and the edge is labelled by i. Label i thus corresponds to the colour of the curve in the Venn diagram that the edge crosses. Note that each vertex in the Venn diagram corresponds to a face in the Venn dual and vice versa.

We also associate with each vertex in the Venn dual a *weight*: recall that the weight of a region is the number of 1's in the binary label of that region, so the weight of a vertex in the Venn dual is the weight of the region enclosing it in the Venn diagram.

Figure 2.2 shows the unlabelled duals of the Venn graphs shown in Figure 2.1; the labels can easily be calculated by converting the region labels in Figure 2.2 to bitstrings.

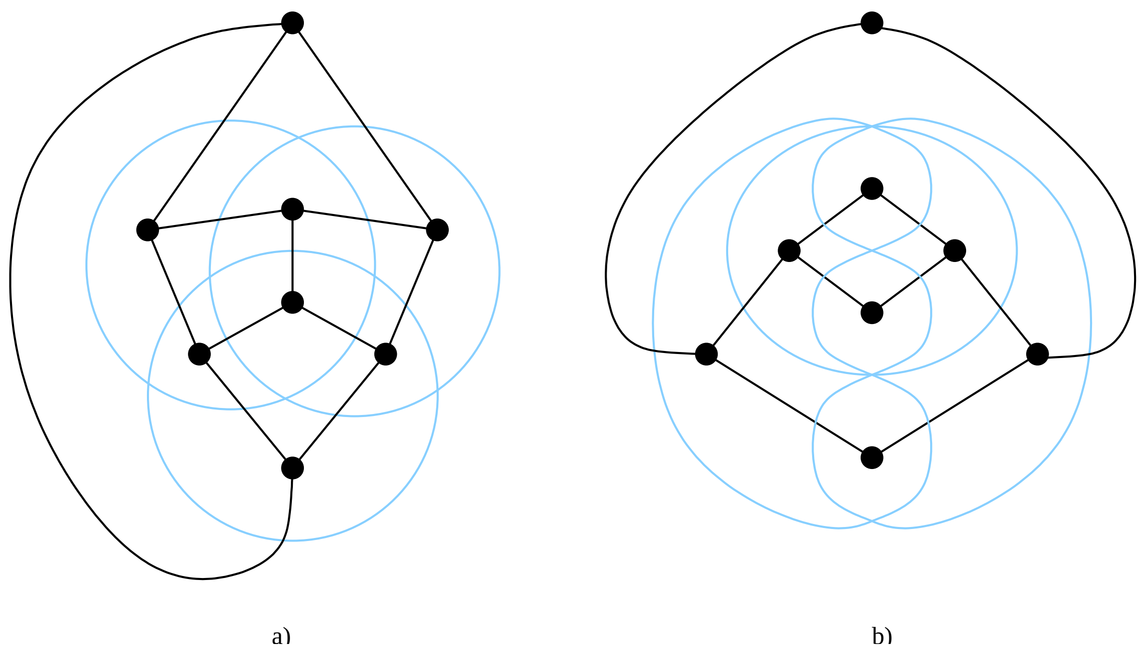


Figure 2.2: Duals of two 3-Venn diagrams, minus their labels.

2.4 Properties of Venn Diagrams

We say that a Venn diagram is *simple* if no three curves intersect in a common point. This condition also tells us that all of the faces of the dual of a simple Venn diagram are 4-faces (faces bordered by exactly 4 edges) if and only if the corresponding Venn diagram is simple, as every vertex in the Venn diagram has degree four if and only if it is simple. Since every vertex in the Venn graph has degree four, Euler's formula ($V - E + F = 2$), combined with the fact that there are 2^n faces, immediately tells us that there are $2^n - 2$ vertices in a simple Venn diagram. The diagram in Figure 2.1a is simple and thus its dual in Figure 2.2a contains only 4-faces, whereas the one in Figure 2.1b is not simple and thus its dual contains some faces bounded by larger numbers of edges than four.

A *vertex traversal* of a vertex v in a Venn diagram is a circular sequence $C_0, C_1, \ldots, C_m$ of the curves adjacent to v, when read in a clockwise rotation around v (m is twice the number of curves passing through v). A vertex is said to be *separable* if its vertex traversal has the property that for some i, there is a j such that the intersection of $A = \{C_i, C_{i+1}, \ldots, C_{j-1}\}$ and $B = \{C_j, C_{j+1}, \ldots, C_{i-1}\}$ consists of exactly one curve and in addition, $|A| > 2$ and $|B| > 2$. Stated informally, v is separable if it can be "pulled apart" into two adjacent vertices in the Venn diagram with no other alteration to the diagram. The separation of a vertex corresponds to the operation of adding an edge between vertices in the Venn dual. A vertex that is not separable in the Venn diagram corresponds in the Venn dual to a region which has the property that no pair of non-adjacent vertices on the cycle bordering the region have labels differing by the inclusion of exactly one curve. For example, vertex v in Figure 2.3 has the vertex traversal *(black, green, red, blue, black, blue, green, red)*, which can be the intersection of $A = \{green, red, black, green, red\}$ and $B = \{blue, black, blue\}$ where the common curve is *black*. This corresponds to the separation into vertices v_1 and v_2 as shown.

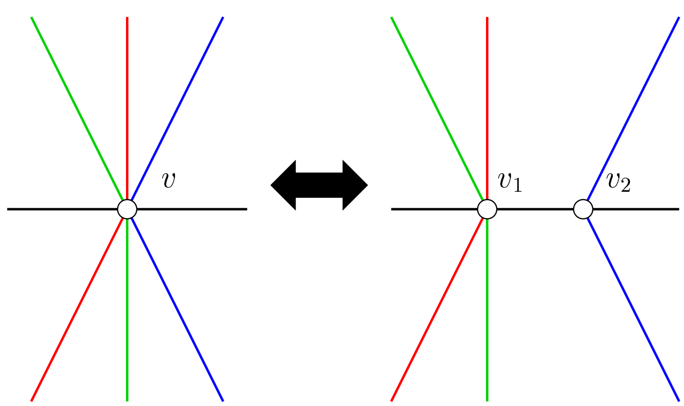


Figure 2.3: An example of the separation of a separable vertex.

A Venn diagram is said to be *rigid* if it has no separable vertices. Rigid diagrams are interesting to study as they cannot be pulled apart into simpler diagrams with more vertices, and thus they are in some sense "canonical". Simple diagrams are trivially rigid, and both diagrams in Figure 2.1 are rigid.

Two Venn diagrams are *isomorphic* if, by continuous transformation of the plane or relabelling of the curves, one can be changed into the other or its mirror image. The two diagrams in Figure 2.1 are not isomorphic to each other, as they contain different numbers of vertices.

A Venn diagram is *monotone* if and only if each vertex in the Venn dual with weight k for $0 < k < n$ is adjacent to at least one vertex with weight $k-1$ and at least one vertex with weight $k+1$. A simple closed curve is *convex* if any two points on its interior can be joined by a straight line that does not cross the curve. A Venn diagram is convex if all of its component curves are convex. By a result of Bultena *et al* [1], there is a straightforward connection between the combinatorial property of monotonicity and the geometric property of convexity: a Venn diagram is monotone if and only if it is isomorphic to a convex diagram.

The diagram in Figure 2.1a is simple, monotone, and convex.

2.5 Symmetric Venn Diagrams

An n-Venn diagram is *symmetric* if it has n-fold rotational symmetry. Thus there is a point about which the diagram may be rotated by $2\pi i/n$, $0 \leq i \leq n$, and remain invariant, with the curve colourings shifting according to a *circular* permutation; a permutation is circular if it consists of one cycle of length n (see [3, pages 259–262]). Two curves are called *congruent* if one can be made to coincide with the other by some sequence of translations and rotations; the n curves in a symmetric diagram must all be congruent.[1]

The curve in Figure 2.1a is symmetric: if we rotate the curves $2\pi/3$ clockwise about the obvious center point and permute the curves by the circular permutation

[1] Some authors have called diagrams with congruent curves symmetric, however the extra criteria about curve rotation is important for our purposes: for example Grünbaum in [6] considers some diagrams that he terms *nice*: they are not necessarily symmetric by rotation but are composed of congruent curves.

(*red* $\rightarrow$ *blue* $\rightarrow$ *green* $\rightarrow$ *red*), we obtain the same diagram. We do not say the diagram in Figure 2.1b is symmetric, despite its having mirror-image symmetry.

The dual of a symmetric Venn diagram is symmetric if we ignore the node labelled with the empty set, corresponding to the external unbounded face of the Venn diagram. This is apparent in Figure 2.2a: if we remove the upper-most node and the edges incident to it, the rest of the diagram is symmetric. In the Venn dual the point of rotation is about the center node corresponding to the region enclosed by all curves, and it is apparent that in general, the point of symmetry for a Venn diagram must lie inside the internal (full) face (or external (empty) face, if the roles of inside and outside of the curves are reversed).

Theorem [10]. For an n-Venn diagram to be symmetric, n must be prime.

Proof. Consider the $\binom{n}{k}$ subsets of weight k, $0 < k < n$, and note that the center of rotation is not in any of these subsets as noted above. Then the subsets must be distributed evenly about the center of rotation for any rotation by $2i\pi/n$, $0 \leq i \leq n$ to map them onto themselves. Thus they must be divided into n groups, each containing a whole number of regions, so n must divide $\binom{n}{k}$ for $0 < k < n$. This implies n must be prime, by a theorem of Leibnitz. $\square$

Another type of symmetry that is considered, relevant only to symmetric Venn diagrams, is as follows: consider the action of stereographically projecting the diagram onto a sphere tangent to the plane at the point of symmetry (*i.e.* with the axis running through the point of symmetry). Then project the diagram back onto the plane tangent to the sphere at the antipodal point; if the resulting diagram is isomorphic to the original diagram, the diagram is said to have *polar* symmetry. The diagram in Figure 2.1a is polar symmetric. Another way to view polar symmetric diagrams (mentioned by Grünbaum [7]) is to consider reversing the roles of "inside" and "outside" for each curve in defining which region is "inside" (*i.e.* contained by) each curve: if the new diagram is isomorphic to the old, they are polar symmetric.

Symmetric Venn diagrams have traditionally been studied for their geometric properties, and in fact the first mention of symmetric Venn diagrams was in relation to a 5-Venn diagram constructed of quadrilaterals [10]. Various results in this area have been proved, for example that it is impossible to construct a Venn diagram of more than three circles [5]. Recently, however, research has shifted towards the many interesting combinatorial properties of symmetric diagrams, with a key result being the construction of symmetric Venn diagrams for any prime n [14] - a construction based on symmetric chain decompositions of the boolean lattice. Before that result, the largest symmetric diagram known was a highly non-simple example for $n = 11$ found by Hamburger [8].

3. A Backtracking Method for Symmetric Diagrams

The ideas we use here to construct symmetric Venn diagrams by drawing doodles were also used by Hamburger in constructing his symmetric 11-Venn diagrams in [8], and are explained in detail by Savage, et al, as part of their construction of a symmetric Venn diagram for any prime n, in [14]. Our contribution is the backtracking methodology to create the doodles, and the subsequent use of the doodles to create symmetric independent families and near-Venn diagrams.

General principles for efficient backtracking algorithms can be found in [11].

3.1 An Example

Figure 3.1 shows a symmetric monotone 7-Venn diagram with several interesting properties.

First, consider that there are two adjacent vertices of degree $2n$ on the inner (and outer) face that isolate between them a sector comprising $1/n$th the entire diagram (here $n = 7$), and that this sector's structure repeats n times in the entire diagram. Figure 3.2a shows a blow-up of this sector, with all curves coloured. If the curves are labelled $1, 2 \ldots, n$ by arbitrarily fixing curve 1 (say the red curve in Figure 3.1) and adding one (modulo n) as the curve is rotated clockwise around the centre of the circle, we can label each region in Figure 3.2a by its corresponding bitstring (shown in the dual graph of the sector, which we return to shortly).

From the sector in Figure 3.2a, the entire diagram can be recovered by rotating the sector n times and rotating the label of each curve for each successive copy by adding one (modulo n). This corresponds to rotating the bitstrings in each region by shifting right one (modulo n) positions (recall that bitstrings are indexed $1, 2 \ldots, n$ from left to right).

Since each region appears exactly once in the diagram, the bitstrings labelling each region in Figure 3.2a must be rotationally distinct, as otherwise if bitstring a

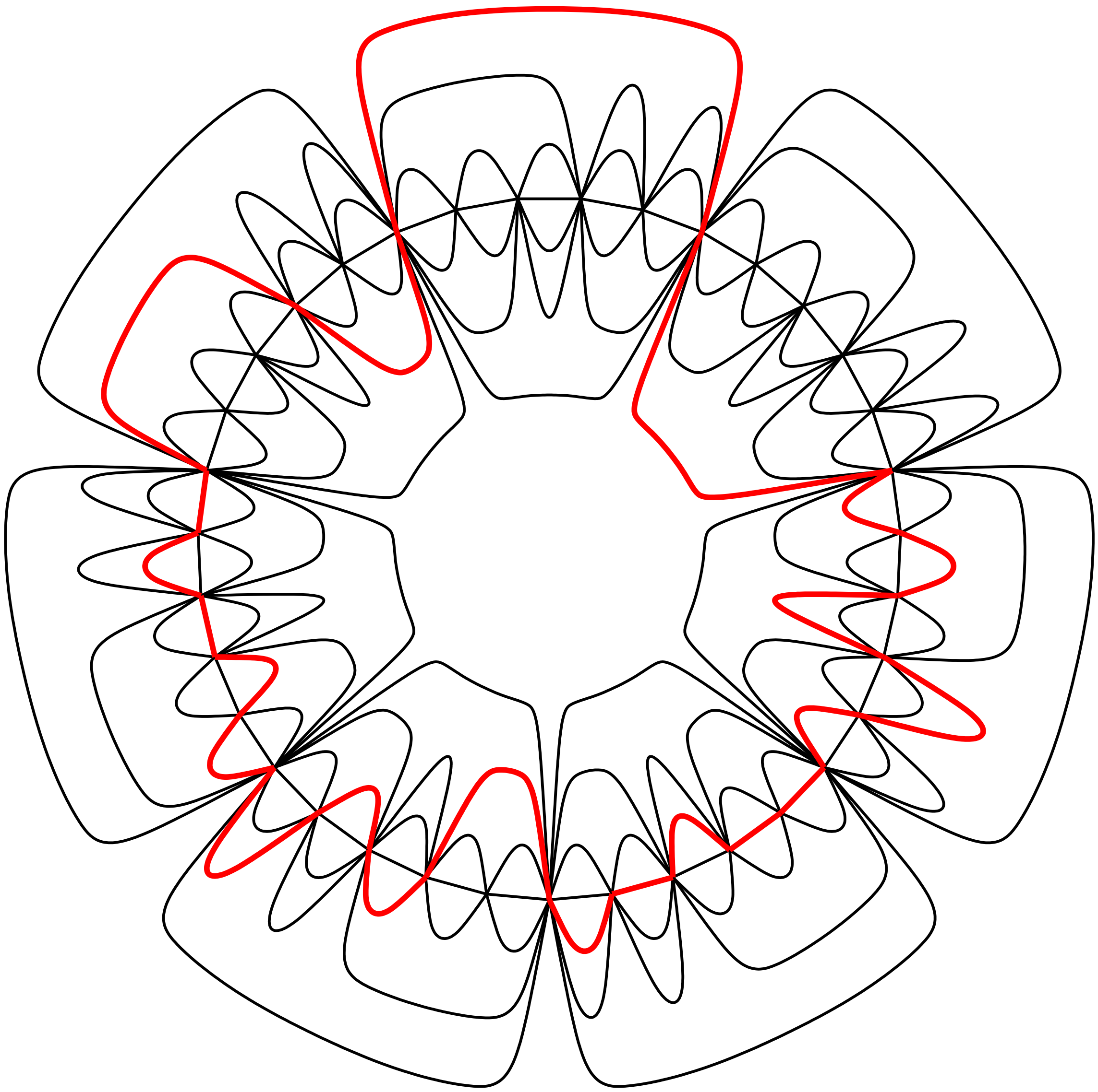
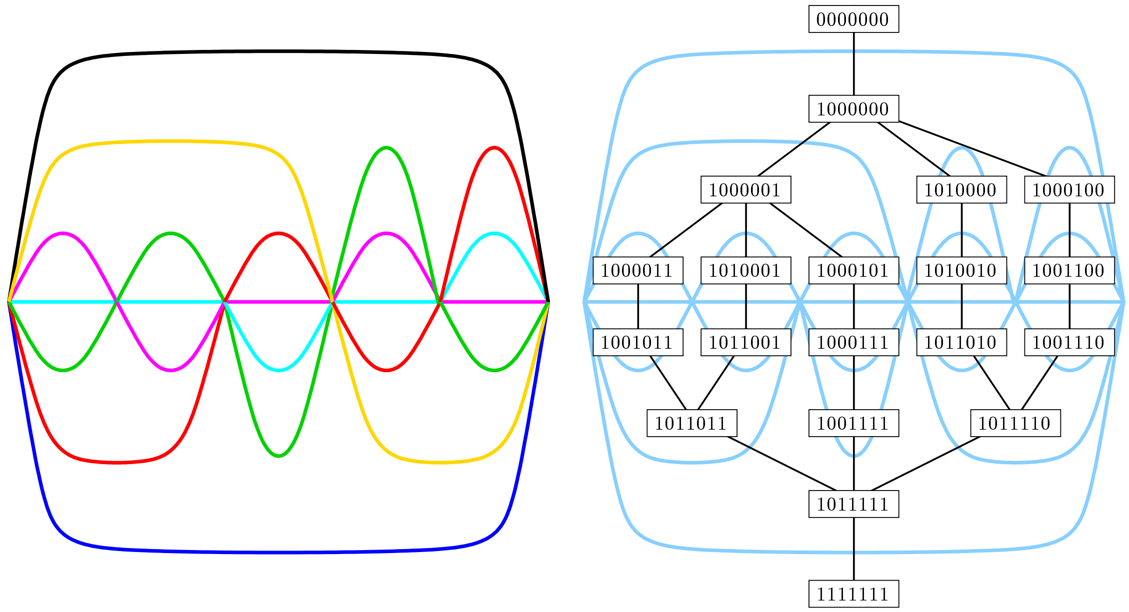


Figure 3.1: Symmetric 7-Venn diagram with one curve coloured.

shifted k times equalled bitstring b, then in the kth rotation of the sector, a second copy of the region labelled b would appear. Also, consider that the n rotations of the sector must give all $2^n - 2$ regions in the diagram if we exclude the single outer face (labelled by 0^n) and single inner face (labelled by 1^n). Thus for any bitstring of length n, there is a region in the sector with a label that is rotationally equivalent to that bitstring (except 0^n and 1^n, which map onto the outer and inner faces respectively, and in the following discussions we ignore the single outer and inner faces unless



a) b)

Figure 3.2: One sector of the sample symmetric 7-Venn diagram and its labelled dual.

otherwise noted.

Recall that a necklace representative is a representative of the equivalence class of strings under rotation. By the above discussion, regions in the sector must be labelled by a representative string (which can be chosen randomly) from every possible distinct necklace of length n.

To summarize, two strings in the sector cannot have the same necklace, as they would map onto each other in some rotation, and a necklace cannot be missing, as that would correspond to some missing region(s) in the full diagram.

The above discussion also gives us the fact that, for n prime, the number of necklace representatives of binary strings of length n is $(2^n - 2)/n$, and the number of necklace representatives of weight k with $1 \leq k \leq n-1$ must be $\binom{n}{k}/n$.

Consider the unique dual of the sector in Figure 3.2a, shown in Figure 3.2b. Nodes in the dual of the sector are labelled by the bitstrings labelling the regions in the sector, and bitstrings of the same weight are next to each other. The sector in

Figure 3.2a can be recovered from this dual, and so the information in Figure 3.2b is sufficient to uniquely identify the resulting diagram. We call such a dual of a sector a *doodle*.[2]

Since the original diagram is monotone, each region is adjacent to a region of higher weight and a region of lower weight, and so edges in the doodle represent a "parent-child" relationship: each bitstring of weight k, $1 \leq k \leq n-1$ has at least one edge to a "parent" bitstring of weight $k-1$, with the edge labelled by the bit that changes from parent to child, and an edge to a "child" bitstring of weight $k+1$. Thus a path from the bottom to the top of the doodle gives a permutation of the edge labels from 1 to n as each curve is crossed exactly once in the corresponding path through the regions in the sector. In some of Hamburger's constructions of symmetric 11-Venn diagrams (for example those in [9]) his diagrams are not monotone, and so some paths from 0^n to 1^n contain more than n edges.

It is clear from the above description that the doodle, together with all of its $n-1$ rotations (creating the dual of the entire Venn diagram), is a spanning subgraph of the n-hypercube.[3]

Finally we note that, though it is not required for the doodle to correspond to a valid Venn diagram, the doodle in Figure 3.2b has the property that each bitstring at level $k \leq \frac{n-1}{2}$ has exactly one parent, each bitstring at level $k \geq \frac{n+1}{2}$ has exactly one child, and the middle two rows form a perfect matching. Viewed another way, the doodle is constructed of two trees embedded in the plane, one rooted at the weight zero node and extending downwards and the other rooted at the weight n node and extending upwards, and joined at their leaves by a perfect matching. This property implies that the doodle has exactly as many faces (as an embedded planar graph) as the number of bitstrings in each of the middle levels (where the depth $k = \frac{n-1}{2}$),

[2]Hamburger, in [8], used the term *doodle* to refer to the sector itself, but he pointed out that it should more properly refer to the *dual* of the sector as that is what is most important in the construction: this is how we will use it.

[3]Set theorists usually draw Hasse diagrams like the one in Figure 3.2b with the empty set at the bottom and the complete set uppermost, but we invert the diagrams so that the node corresponding to the empty set is uppermost and thus the doodle corresponds to the top of the Venn diagram once it is rotated.

which corresponds to the number of vertices in the sector, and there can be no fewer regions. Then the corresponding Venn diagram has $\binom{n}{\lfloor \frac{n}{2} \rfloor}$ vertices, which is minimum for monotone Venn diagrams via a result by Bultena *et al* [1], and the addition of more parent-child edges would create more regions, and thus more vertices in the Venn diagram.

The above discussion suggests an efficient backtracking method of generating diagrams similar to those in Figure 3.1, based on generating a doodle, from which the entire diagram can be generated by recovering the sector and then rotating it and permuting edge labels to obtain the entire diagram. For more detail on the procedure of obtaining the final diagram from the doodle, see [14] and [8].

3.2 A Backtracking Technique

It is clear from the preceding section that the number of nodes of weight k in the doodle is fixed at $\binom{n}{k}/n$, and that a bitstring labelling a node of weight k must be rotationally distinct from all others of the same weight (trivially a bitstring of weight k is rotationally distinct from a bitstring of different weight). Finally the nodes in the doodle must satisfy the parent-child relationship explained above for the doodle to correspond to a valid n-Venn diagram, and indeed any doodle satisfying these constraints can be turned into a valid diagram. We can generate such a doodle by filling in the doodle from top to bottom with bitstrings satisfying the above constraints.

Since every doodle must have 0^n as its uppermost node and 1^n as its lowermost node, we can start generating the doodle at the second level, with a node of weight one, and finish at the second-last level with weight $n-1$. Without loss of generality we can assume that the initial bitstring forming the "root" of the doodle is $1 \cdot 0^{n-1}$. Also, since the curves are labelled arbitrarily, we can also assume that the permutation to be applied to the curve colours to obtain the final diagram is $(12 \cdots n)$ as the curves can always be relabelled to change the permutation from any circular permutation to any other. This exploits the fact that two Venn diagrams that differ only in their curve colourings are considered isomorphic.

This suggests a backtracking technique for generating such diagrams by generating valid doodles:

1. Fix the top bitstring at level $k = 1$ to be $1 \cdot 0^{n-1}$.

2. At each subsequent level $k \leq \frac{n-1}{2}$, fill in the row from left to right with bitstrings of weight k such that each bitstring placed:

   (a) is rotationally distinct from previous bitstrings in the row, and

   (b) is a child of either the last bitstring in the previous row that is a parent, or the bitstring immediately to its right in the previous row.

3. Fill in the middle row at level $k = \frac{n+1}{2}$ from left to right with bitstrings of weight k such that each bitstring placed:

   (a) is rotationally distinct from previous bitstrings in the row, and

   (b) is a child of the first bitstring in the previous row that is not already a parent.

4. At each subsequent level $k > \frac{n+1}{2}$, fill in the row from left to right with bitstrings of weight k such that each bitstring placed:

   (a) is rotationally distinct from previous bitstrings in the row, and

   (b) is a child of one or more consecutive bitstrings in the previous row, none of which are already parents.

5. Halt once the single bitstring at level $n - 1$ is fixed.

The algorithm randomly selects bitstrings and attempts to fill in the entire doodle, from level 1 to level $n - 1$, backtracking on the current row or previous rows whenever it cannot continue (because no valid bitstring remains to fill in the current row, or there are no more possible parents in the previous row). Once a row is completed with the last bitstring having as its last parent the last bitstring in the previous row, it then begins filling in the next row.

Steps 1-2 generate the embedded tree forming the top half of the doodle, Step 3 creates the matching in the middle row, and Steps 4-5 create the inverted tree forming the bottom half of the doodle.

3.2.1 Implementation

We have implemented this algorithm in C. In this section we briefly overview the data structures we used to generate the doodle.

Since node labels are elements of the n-hypercube, we store the n-hypercube in a data structure containing elements as follows:

- An array of pointers to linked lists of **HypercubeNode**s, with one linked list per level (from one to $n-1$).
- Each **HypercubeNode** in the linked list at level i contains a bitstring of weight i, pointers to the **HypercubeNode**s containing all possible parent and children bitstrings on levels $i-1$ and $i+1$ respectively, and a pointer to a **Necklace** object.
- Each **Necklace** object contains the lexicographically minimal rotation of a string, along with a boolean flag **IsUsed** which is initially false.

To generate the doodle, we store it in a data structure containing the following elements:

- An array of pointers to linked lists of **DoodleNode**s, with one linked list per level (from one to $n-1$). The linked list for level i has length $\binom{n}{i}/n$.
- Each **DoodleNode** contains a pointer to a **HypercubeNode** and an integer recording the number of children (if the level is less than $n/2$) or parents of the node.

The code recursively traverses the doodle stored in the data structure above from level one to $n-1$. To set a node x at level $i+1$ with $i < n/2$, we decide which **DoodleNode** y in level i is to be the parent of the new node. Dereferencing y's pointer into the **Hypercube**, we then traverse the list of possible children of y and try all possible children whose **Necklace**'s **IsUsed** flag is not set. Once we set some child of y to be x and set the **IsUsed** flag of its **Necklace** to true, we recurse to either x's right-hand neighbour or the start of the next row (as appropriate).

The middle row is filled similarly, except that each node must have exactly one parent. Filling the bottom half of the tree is similar, except that if a node x in level $i+1$ has a parent y, we can test to see if the node next to y can be a parent of x as well, and we do not allow y to be the parent of two successive nodes.

3.3 Properties of Venn Diagrams Generated via Doodles

Given a doodle output from our generation algorithm, it is useful to be able to determine some properties of the resulting Venn diagram, specifically if it is polar symmetric, simple, or rigid. Note that since the diagrams we generate have exactly $\binom{n}{\lfloor n/2 \rfloor}$ vertices, which is minimum for symmetric Venn diagrams, no diagram we generate with the above process can be both simple and rigid, but a non-rigid diagram may be separable to a simple diagram. We can overload the relevant terms and refer to a doodle as *rigid*, *polar*, or *simple*, if it corresponds to a Venn diagram that is rigid, polar symmetric, or separable to a simple diagram respectively.

3.3.1 Simple or Rigid Doodles

As noted in Chapter 2, separating a vertex in the Venn diagram corresponds to adding an edge crossing the cycle bordering the corresponding face in the doodle. Since no edge can be added between node labels differing by more than one bit, in our doodles vertex separations correspond to the addition of edges between levels k and $k+1$ for $1 \le k \le n-1$. In fact, they will always occur in one of the two ways possible in Figure 3.3: between vertex a and d or vertex b and c.

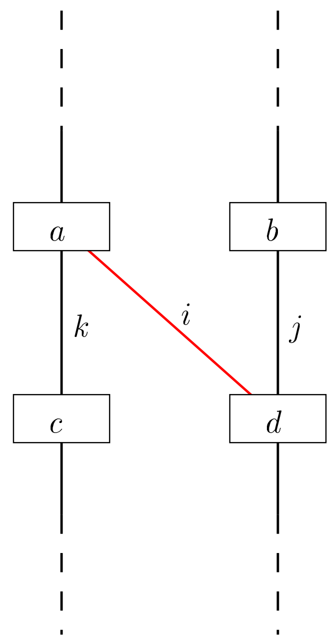


Figure 3.3: Adding an edge $\{a, d\}$ across a non-rigid face in a doodle by setting $a_i = 1$.

Lemma 1 *In the Venn doodle, if an edge can be added between a vertex a in level k and a vertex d in level $k+1$, no edge can be added to cross the new edge.*

Proof. Since a and b are both parents of d they differ in exactly two positions, say i and j with $i \neq j$, as setting a_i to 1 obtains d and also setting b_j to 1 obtains d. Also, a is a parent of c and a differs from c by exactly one bit a_k, with $k \neq i$. Also $k \neq j$ as bit j must already be set in a. Thus b differs from c by exactly three different bit positions, so b cannot also be a parent of c. $\square$

In Figure 3.3, this means if a and d are adjacent, then b and c cannot be adjacent.

Note that not only do we have to consider edges within the interior faces of the doodle, but also edges that correspond to separating the single vertices on either side of the doodle once the dual is taken to turn it into a sector and rotated to obtain the complete diagram; we term these *wrapping* edges. For example in Figure 3.4 the red edges on either side of the doodle in the centre are wrapping edges which can be added.

Lemma 2 *Given a doodle $\mathcal{D}$, there is exactly one rigid diagram corresponding to $\mathcal{D}$,* *formed by separating vertices in the diagram recoverable from $\mathcal{D}$.*

Proof. This is apparent from Lemma 1: no face in the doodle can have edges added in mutually exclusive ways and so there is exactly one way to add as many edges as possible to a given doodle, including wrapping edges. □

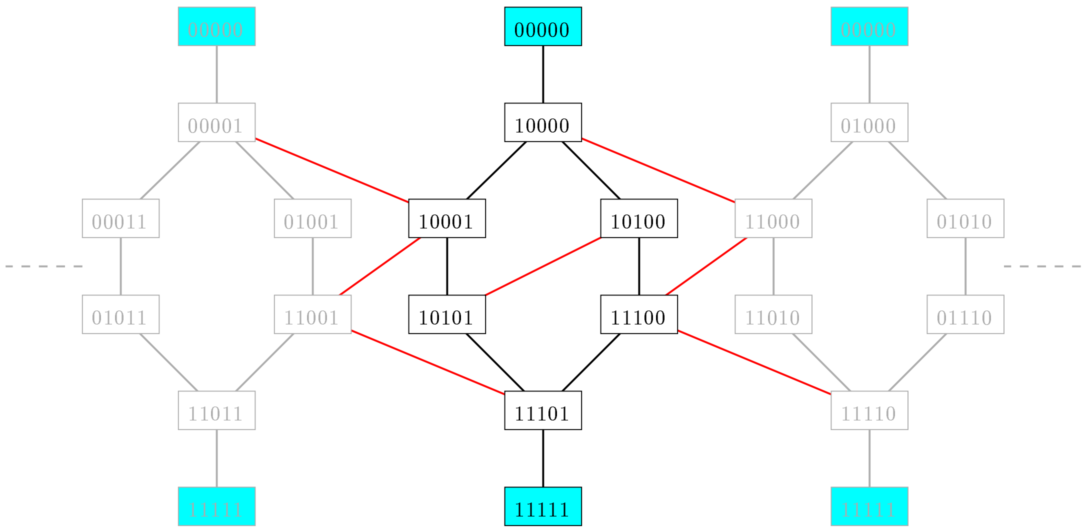


Figure 3.4: A doodle of a 5-Venn diagram separable to a simple diagram; blue vertices of the same weight are identified.

However, a given rigid diagram with more than the minimum number of vertices may be generated by separating vertices in several distinct doodles (for example in Figure 3.3 the doodle may have been generated with either a or b as the parent of d). Thus our backtracking algorithm will generate every doodle corresponding to a rigid minimum-vertex diagram exactly once but other diagrams more than once; in fact the larger the number of vertices in the Venn diagram the greater the number of non-rigid doodles of minimum vertex diagrams will correspond to it. We will see the consequences of this in Section 3.4 when we generate doodles for diagrams of $n = 5$ and 7 curves, cases where the number of different simple symmetric diagrams are already known, and compare how many such diagrams we generate.

Lemma 1 gives us an easy way to tell if a given doodle corresponds to a rigid or a simple diagram: if no extra edges can be added to the doodle, then the diagram is rigid, but if the maximum number of possible extra edges can be added, including wrapping edges (turning every face in the Venn dual into a 4-face), then the resulting doodle corresponds to a Venn diagram that is separable to a simple diagram.

As an example, consider the doodle of the simple 5-Venn diagram in Figure 3.4: the red edges are the extra edges that can be added to turn all faces into 4-faces, and we see that we can add $n-2$ wrapping edges on each side of the doodle to separate the vertices on either side of the doodle in the resulting sector (note that the 0^n and 1^n vertices are identified; we isolate them to emphasize how the doodle is copied and the labels rotated to obtain the full diagram). Thus in Figure 3.4 each face, after adding the red edges, is a 4-face and the resulting Venn diagram is simple.

3.3.2 Polar Symmetry

A polar symmetric diagram can be viewed as being isomorphic to itself if the roles of inside and outside of every curve are reversed, *i.e.* if we "turn the diagram inside-out". In terms of the diagram's doodle, the doodle must be equivalent to itself under transformation as follows: if the bottom half of the doodle (including the middle two levels) is reflected vertically and then horizontally, and the bitstrings inverted (1's changed to 0's and vice versa) and reversed, the result must be rotationally equivalent to the original top half of the doodle.

For example, consider again the doodle in the centre of Figure 3.4: the first label of weight two, 10001, becomes 01110 when inverted and reversed, which is a rotation by one right of the last label of weight three, 11100. Similarly the last label of weight two is 10100 which when inverted and reversed becomes 11010, which is a rotation by one right of the first label of weight three, 10101. The key point is that the amount of rotation must be the same – both strings, after inversion and reversal, must be rotated by the same amount (here one place right) to obtain their polar symmetric companions. This property must hold for all bitstrings in the doodle.

For polar symmetry it must also be the case that the structure of the doodle (the parent-child relationships graph) has the same symmetry, but this will always be the case if the bitstrings satisfy the above property: if the structure does not already have this reflective symmetry then it is apparent that it could be altered to the appropriate structure by simply changing the parent-child relationships, as the operations of inverting, rotating and reversing the bitstrings do not change the relative parent-child relationships.

3.3.3 Exhaustiveness

Finally we note that though all of the generated diagrams will be monotone and symmetric, not every monotone symmetric diagram can be created in this way: Figure 3.5 (from [12]) shows a monotone symmetric minimum-vertex 5-Venn diagram that can easily be seen to not have a doodle of the type shown in Figure 3.2 as it does not have two vertices that denote a sector of the diagram that can be rotated to form the entire diagram: the vertices on the outer face are not on the inner, and vice versa, and the dual does not contain a doodle starting and ending at a single vertex.

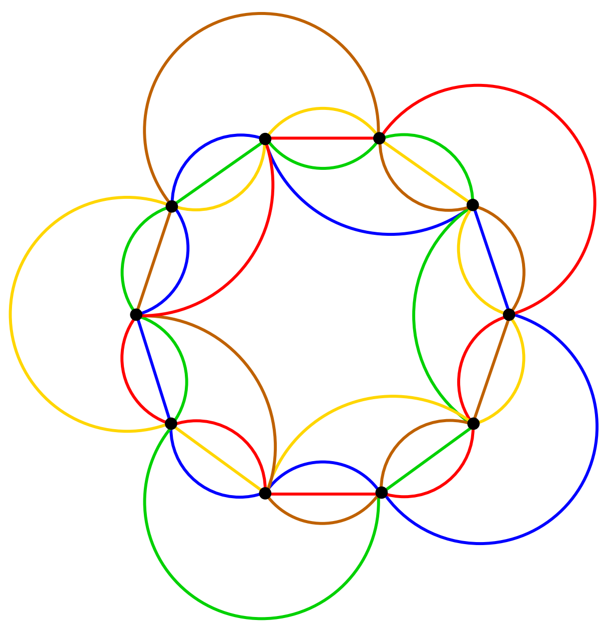


Figure 3.5: [12] Monotone symmetric 5-Venn diagram that cannot be constructed using doodles.

Similarly, it is worthwhile to consider whether every simple symmetric Venn diagram can be constructed in this method: all of the 23 simple symmetric diagrams for $n = 7$ have doodles that can be separated to make the resulting simple diagram. Instead of having two vertices of degree $2n$ that denote a sector of the diagram, the simple diagrams have paths of length n that start at the outer face and end at the inner face, and the vertices along that path can all be compressed together into two such vertices that denote a sector of the diagram.

It appears that it may be possible to create a simple 11-Venn diagram that does not have a doodle – we can imagine that it will look like a much more complex version of Figure 3.5 with simple vertices – as the middle rank of the doodle for $n = 11$ must be very large (42 vertices) and the single top and bottom vertices have high degree (five), so it appears that there is no reason that a simple 11-Venn diagram would always have an 11-path from the outer to the inner faces.

Since, as we shall see, the number of Venn diagrams in general and simple symmetric Venn diagrams in particular increase dramatically as n increases, we can surmise:

Conjecture 1 *There exist many order-11 simple symmetric Venn diagrams.*[4]

Thus by the above discussion, we offer the following:

Conjecture 2 *There exists a simple order-11 symmetric Venn diagram that cannot be constructed by building a doodle for it and then separating vertices.*

3.4 Generation Results for Venn Diagrams

As noted above, this backtracking technique will not generate all symmetric n-Venn diagrams, as we will see by summarizing the output from the algorithm for small n. We did not find any diagrams larger than previously known examples via this method, but there are some interesting aspects of the enumerations.

[4]As partial support for this, many different non-simple Venn diagrams of order 11 have already been found by other researchers, for example Hamburger's diagrams, found by starting from a few "source" diagrams and separating vertices [9], and the diagrams created by the construction due to Savage, et al. [14].

3.4.1 $n = 5$

We summarize in Table 3.1 the different properties of all 243 symmetric 5-Venn diagrams, taken from [12], and then compare them against our generated output. Our construction will only produce a subset of the monotone diagrams (counted in the bottom half of the table).

monotone	**rigid**	**polar**	Total
no	no	no	100
no	no	yes	5
no	yes	no	12
no	yes	yes	2
yes	no	no	89
yes	no	yes	18
yes	yes	no	13
yes	yes	yes	4

Table 3.1: [12] Properties of symmetric 5-Venn diagrams.

Table 3.2, taken from [12], shows the number of 5-Venn diagrams broken down by the number of vertices. Since the diagrams are symmetric, the number of vertices must be divisible by 5. As $2^5 - 2 = 30$, the single rigid diagram with 30 vertices is simple.

vertices	**rigid**	**non-rigid**
10	6	66
15	12	99
20	12	37
25	0	10
30	1	0

Table 3.2: [12] Number of vertices of symmetric 5-Venn diagrams.

Our algorithm generates 32 5-Venn diagrams, all of which have ten vertices and are monotone; Table 3.3 summarizes the properties of them. Interestingly, no rigid diagrams are generated by our method. Recall that a doodle is *simple* if it corresponds to a Venn diagram that is separable to a simple diagram.

simple	**polar**	Total
no	no	20
no	yes	4
yes	no	4
yes	yes	4

Table 3.3: Doodles for symmetric 5-Venn diagrams generated by backtracking algorithm.

Since there are 72 diagrams in total with ten vertices, and 124 monotone diagrams known, we can see that we are generating a non-trivial fraction of the entire class of diagrams of this type, even if we take into account the inevitable duplication of the simple and non-rigid diagrams we mentioned above.

3.4.2 $n = 7$

Our output for $n = 7$ in Table 3.4 is more interesting and bears some examination. The generation algorithm produces a total of 687,120 doodles corresponding to symmetric 7-Venn diagrams.

simple	**rigid**	**polar**	Total
no	no	no	678336
no	no	yes	5602
no	yes	no	1504
no	yes	yes	72
yes	no	no	1472
yes	no	yes	134
yes	yes	no	0
yes	yes	yes	0

Table 3.4: Doodles for symmetric 7-Venn diagrams generated by backtracking algorithm.

We know that there are exactly six simple and polar symmetric diagrams for $n = 7$ and 17 simple non-polar diagrams [12]. Thus we can see that this algorithm must generate each simple polar symmetric diagram many times as there are 134 generated doodles with these properties. Similarly the 17 simple non-polar diagrams are generated 1472 times in total. Also, we can see that the vast majority of generated diagrams are not simple, rigid, or polar, and even bearing in mind the duplication inherent in producing non-rigid diagrams via minimum-vertex doodles, we can safely say there are on the order of thousands of these diagrams.

Some of these types of diagrams are "new" in the sense that no previous examples of diagrams with these characteristics have been published. Figure 3.6 shows a doodle of a rigid (and thus having the minimum number of vertices) polar symmetric 7-Venn diagram; the corresponding diagram is shown in Figure 3.7. Many more examples with these properties can be exhibited.

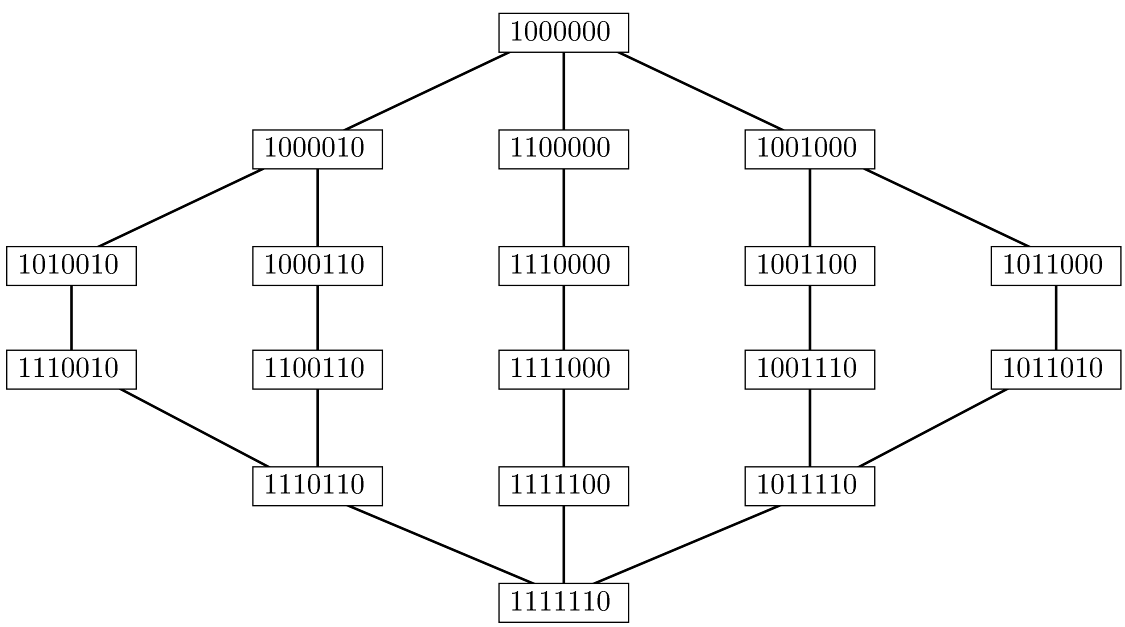


Figure 3.6: Doodle of a rigid minimum-vertex polar symmetric 7-Venn diagram.

Since the number of non-simple symmetric 7-Venn diagrams is unknown, based on the above ratio of generated to total 5-Venn diagrams (about 32/124), we can speculate that there are on the order of millions of symmetric 7-Venn diagrams.

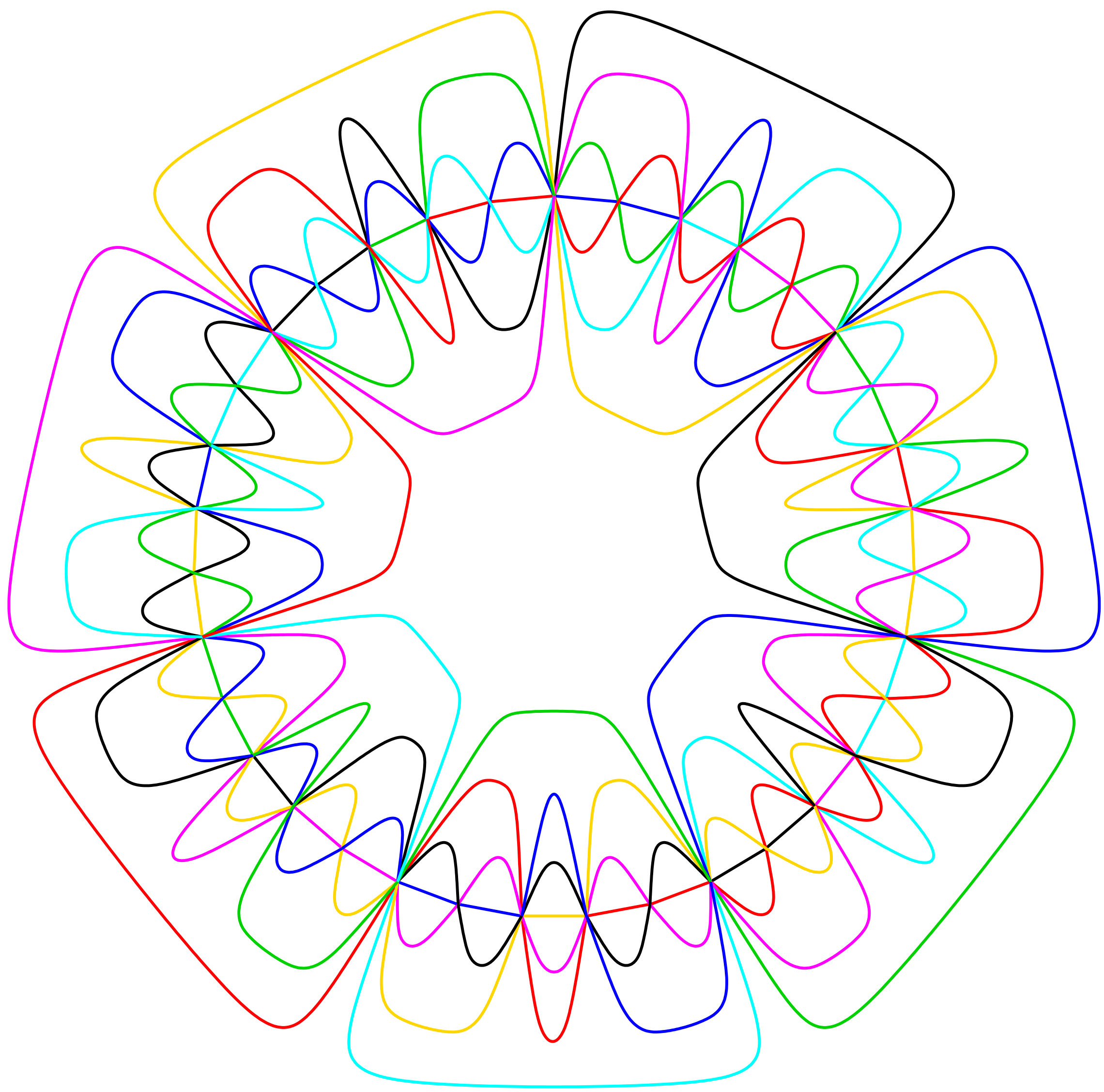


Figure 3.7: A rigid minimum-vertex polar symmetric 7-Venn diagram.

3.4.3 $n = 11$

We attempted to find symmetric Venn diagrams for $n = 11$ using the above backtracking method, but it appears the search space is too vast to succeed using this straight brute-force backtracking approach. The algorithm can be refined in many ways to narrow the search space, and we did indeed achieve some progress in this direction, along with using smart randomization. However it appears a different search paradigm will be needed to generate significant numbers of 11-Venn diagrams using this approach. Some general principles for fast backtracking that we applied to the problem are described briefly in [11].

3.5 Independent Families

In [7], Grünbaum generalized the concept of symmetry to independent families of n sets when n is not prime. He also provided some useful bounds, which we briefly survey before presenting some results.

If n is not prime, some of the regions present in an independent family have labels with rotational period less than n (in addition to the outside region 0^n and inside region 1^n); for example, if $n = 8$ there must be regions corresponding to the label 11001100, with period 4, and 10101010, with period 2. To create a symmetric diagram, the doodle we create (using the same method as described above), must have nodes corresponding to these regions (in addition to regions with rotational period n). As each doodle comprising $1/n$th of the diagram must have a representative for every necklace, there must be repeated regions in the resulting diagram, and the number of necklaces of size n gives us a lower bound $M(n)$ on the number of required regions: $M(n) = 2 + n(C_n - 2)$, where C_n is the number of binary necklaces of length n. The numbers C_n are well-studied, see [3, pages 139–141] for explicit formulae and further references. We say that symmetric independent families that attain this lower bound are minimal, in the sense that they cannot have any regions deleted without either losing their symmetry or losing some regions altogether and ceasing to be independent families.

Table 3.5 gives $M(n)$ and C_n for small non-prime n, along with 2^n for comparison. We can see that in fact symmetric independent families do not require very many repeated regions to be symmetric, as the number of extra regions will be the difference $2^n - M(n)$.

Grünbaum conjectured in [7] that symmetric independent families with only $M(n)$ regions exist for every integer n (this includes the special case that n is prime, and the symmetric independent family is just a symmetric Venn diagram - it has no

n	C_n	$M(n)$	2^n
4	6	18	16
6	14	74	64
8	36	274	256
9	60	524	512
10	108	1062	1024
12	352	4202	4096

Table 3.5: $M(n)$ for small n.

repeated regions), and gave examples for $n = 4$ and $n = 6$ of such diagrams. The rest of this section provides diagrams that provide examples closing the next few open cases, $n = 8, 9$ and 10. [5]

3.5.1 Generation and Results

We can generate such symmetric independent families for a given non-prime n using a slight modification to our backtracking algorithm:

1. At each level in the doodle of depth k, fix the number of nodes to be the number of binary necklaces of length n and weight k.

The rest of the algorithm, including starting and terminating conditions, remains unchanged.

Thus, if the backtracking algorithm reaches the final node, each level has been filled in with rotationally distinct labels, and since a level at depth k must contain exactly the number of necklaces of that weight, each of which will appear in all n possible rotations to create the final diagram, every possible bitstring is represented either in the doodle or in some rotation of that doodle.

Since there seem to be many more symmetric independent families than symmetric Venn diagrams, instead of concentrating on enumerating the diagrams we will

[5] Interestingly, a symmetric independent family of size six composed of quadrangles Grünbaum gave in 1975 in [5, Figure 7] also attains the lower bound. This diagram is also very nice in that, although it is not simple, it has only 6 vertices in which more than two curves cross; by the property of 6-symmetry this is as close as one can get to being simple without actually being simple.

simply state that many examples can easily be generated by the above algorithm for $n = 4, 6, 8, 9$, and 10, and present some interesting examples.

For $n = 4$, several nice minimal examples can be found in the existing literature, such as Figure 1 in [7], and more are fairly easy to produce. For $n = 6$, Figure 3.8 shows a minimal independent symmetric family with numbers marking the repeated regions. All regions labelled i are enclosed by the same subset of curves: for example, the two regions labelled 1 are both enclosed by exactly the black and cyan curves, and (assuming black is colour 1 and cyan is colour 4, corresponding to labelling the colours 1 to 6 moving clockwise on the outer curve starting from black) both regions correspond to the bitstring 100100, which has period three.

A visual illustration of the fact that the diagram in Figure 3.8 is minimal is to consider that the removal of any single one of the numbered regions (which are the repeated regions) would destroy the symmetry.

Figure 3.9 is a symmetric independent family of eight curves with $M(8)$ regions. It closes the first open case mentioned by Grünbaum in [7].

The diagram in Figure 3.9 is drawn with no separable vertices: thus this diagram is as near to simple as it can be. It has only 24 points where three curves cross each other (ever other point has the property, common to every point in simple diagrams, that at most two curves cross at that point). As the diagram has 24 degree six vertices, there are three degree six vertices in the sector corresponding to the dual of the doodle of the diagram; all other vertices have degree four. We refer to diagrams like Figure 3.9 as *near-simple* to emphasize that they have few vertices of degree greater than four. Grünbaum suggested in [7] that we cannot do better than this: he conjectured that if n is not prime, no simple symmetric independent family of n curves exists with $M(n)$ regions. We have been unable to find a doodle for a symmetric independent family of eight curves with fewer than three vertices of degree greater than four, or in fact a simple symmetric independent family for any n not prime.

Note that Grünbaum's conjecture contains the condition that the diagram has $M(n)$ regions: this is because a near-simple diagram like that in Figure 3.9 can easily be turned into a simple symmetric independent family of curves with more than $M(n)$ regions by taking the few vertices of degree greater than four and separating the curves to create a new region (or regions) surrounded by simple vertices in the natural way. In the case of Figure 3.9, each of the degree six vertices in the 1/8th sector of the diagram can be pulled apart to create a triangle bounded by the three curves passing through the separated vertex, adding three extra regions to the sector of the diagram. Thus this would create a simple diagram with $274 + 8 \times 3 = 298$ regions.

Figures 3.10 and 3.11 show minimal symmetric independent families for $n = 9$ and 10 respectively; they have $M(9) = 524$ and $M(10) = 1062$ regions, and have many vertices of high degree.

3.6 Near-Venn Diagrams

In the last section we examined generating diagrams that provide a lower bound on symmetric diagrams that are "larger" (have more regions) than Venn diagrams; it is natural to also consider diagrams with an upper bound on the number of regions that are "smaller" than Venn diagrams, *i.e.* have less than 2^n regions. Thus in some sense we can "bracket" non-symmetric Venn diagrams of non-prime n both from above and below by symmetric (non-Venn) diagrams.

In Section 3.5 we discussed creating symmetric diagrams for non-prime n by adding extra regions where needed to preserve symmetry and thus creating symmetric independent families with few duplicated regions. Conversely, we can create symmetric diagrams by removing as few regions as possible from a Venn diagram to create a figure that is not an independent family (as some regions will be missing) but is symmetric. More precisely, since if n is not prime n does not divide $\binom{n}{k}$ for some k in the range $1 \leq k \leq n-1$, and thus the number of regions of weight k cannot be evenly distributed into n sectors of a symmetric diagram, we could reduce the diagram to only having $n\lfloor \binom{n}{k}/n \rfloor$ regions of depth k. Then the question becomes: which regions to remove? If we simply choose any of the $n\lfloor \binom{n}{k}/n \rfloor$ regions to keep we may still have regions that repeat. More satisfying is to ensure that there are no repeated regions in the diagram, thus preserving the similarity to Venn diagrams - thus we may have even fewer regions than $n\lfloor \binom{n}{k}/n \rfloor$ regions at depth k . As in the previous section we can consider the number of nodes in the doodle: to not repeat any regions, each doodle's nodes must only have labels that are Lyndon words (aperiodic necklaces).

Since these diagrams are not Venn diagrams, or even independent families, but are "close" to being Venn diagrams in that they are missing few regions and every region is connected, we call them *near-Venn* diagrams, and assume they are symmetric unless otherwise noted. Near-Venn diagrams for non-prime n are maximal in the sense that they cannot have any more regions added without either losing their symmetry or repeating regions.

Using $n = 6$ as an example, there are nine Lyndon words of length six (of the 14 total necklaces). So the doodle for a near-Venn diagram of six curves would have nine labels, which would then rotate six times to create $9 \times 6 + 2 = 56$ regions in the resulting diagram; the extra two is for the exterior and interior faces. A diagram satisfying these constraints is shown in Figure 3.12.

Like the function $M(n)$, which provided a lower bound on the number of regions in a symmetric independent family of order n, we can define a new function $N(n)$, which provides an upper bound on the number of regions in a symmetric near-Venn diagram of order n. By the above discussion, $N(n) = 2 + n(L_n)$ where L_n is the number of binary Lyndon words of length n. Formulae for L_n and further references can be found in [13].

We give $N(n)$ for small n in Table 3.6, and 2^n for comparison. Just as we saw in the case of symmetric independent families, the number of regions in a near-Venn diagram of order n is quite close to the number of regions in a n-Venn diagram.

n	$N(n)$	2^n
4	14	16
6	56	64
8	242	256
9	506	512
10	992	1024
12	4022	4096

Table 3.6: $N(n)$ for small n.

3.6.1 Generation and Results

Our algorithm to generate near-Venn diagrams is again a trivial modification of the algorithm to produce doodles for symmetric Venn diagrams:

1. In the doodle, at each level of depth k, fix the number of nodes to be the number of binary Lyndon words of length n and weight k.

2. Restrict the algorithm to only choose labels of nodes from the set of Lyndon words of length n.

The second condition is needed to ensure that there are no repeated regions in the resulting near-Venn diagram corresponding to the completed doodle.

Again the rest of the algorithm, including starting and terminating conditions, remains unchanged.

Thus, if the backtracking algorithm reaches the final node, each level has been filled in with rotationally distinct labels, all of which are Lyndon words, and so each region in the resulting near-Venn diagram is distinct.

An example of a near-Venn diagram generated in this fashion is shown in Figure 3.12. This diagram has $N(6) = 56$ regions, fewer than the 64 required for a Venn diagram of order six.

We have generated examples of doodles of maximal symmetric near-Venn diagrams for $n = 4, 6, 8,$ and 9 and more can easily be produced; in fact, most doodles for symmetric independent families can easily be turned into doodles for symmetric near-Venn diagrams simply by deleting nodes labelled by non-Lyndon words (as long as the tree structures can be repaired in the doodle - the issue of the *repairability* of doodles is revisited later in the context of pseudo-symmetric diagrams).

Once again we have found that, for composite n, all near-Venn diagrams we can construct using the above method are non-simple. We can extend Grünbaum's conjecture as follows:

Conjecture 3 *For composite n, no symmetric maximal near-Venn diagram with no repeated regions exists that is simple.*

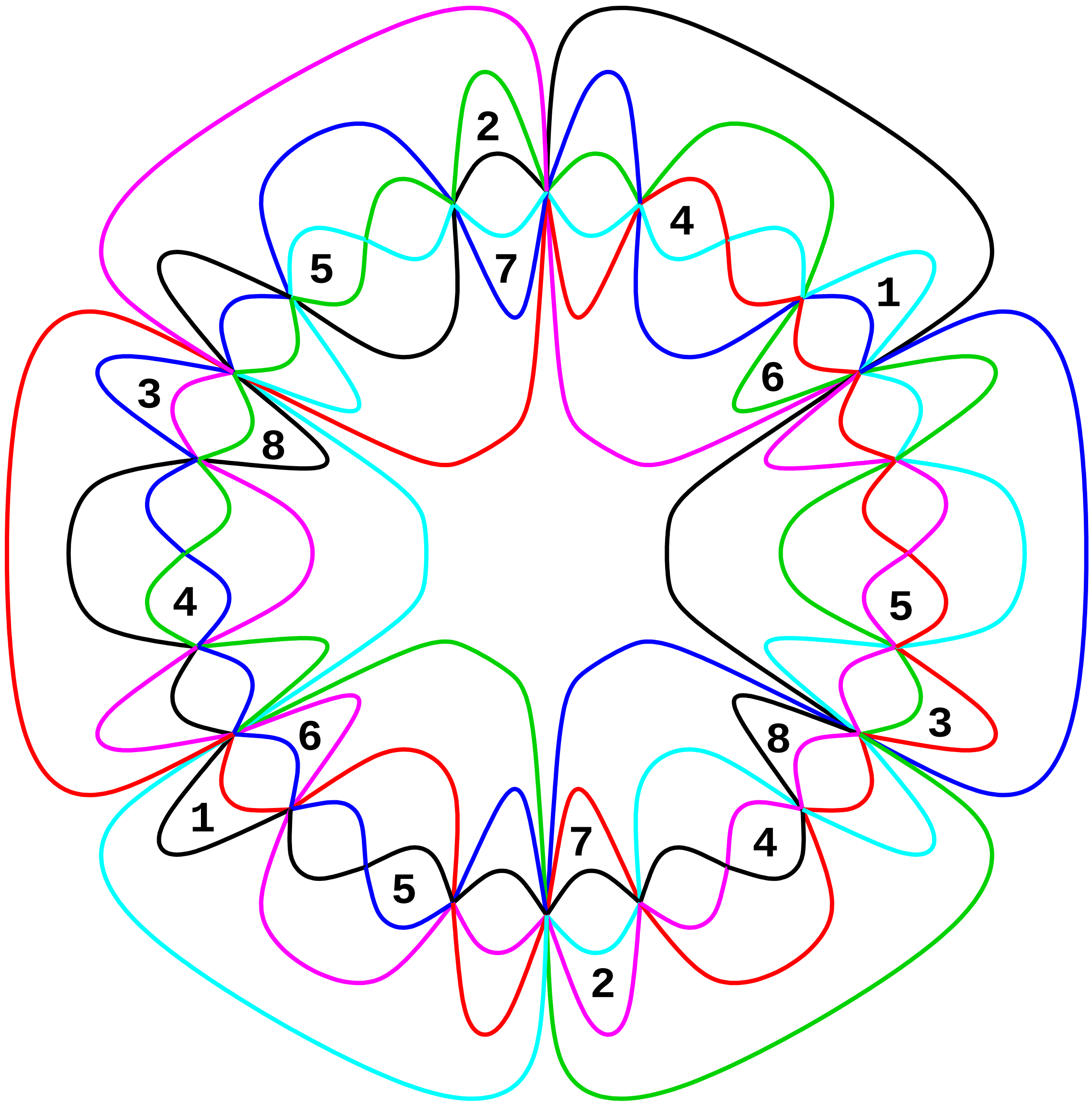


Figure 3.8: Minimal symmetric independent family of 6 curves.

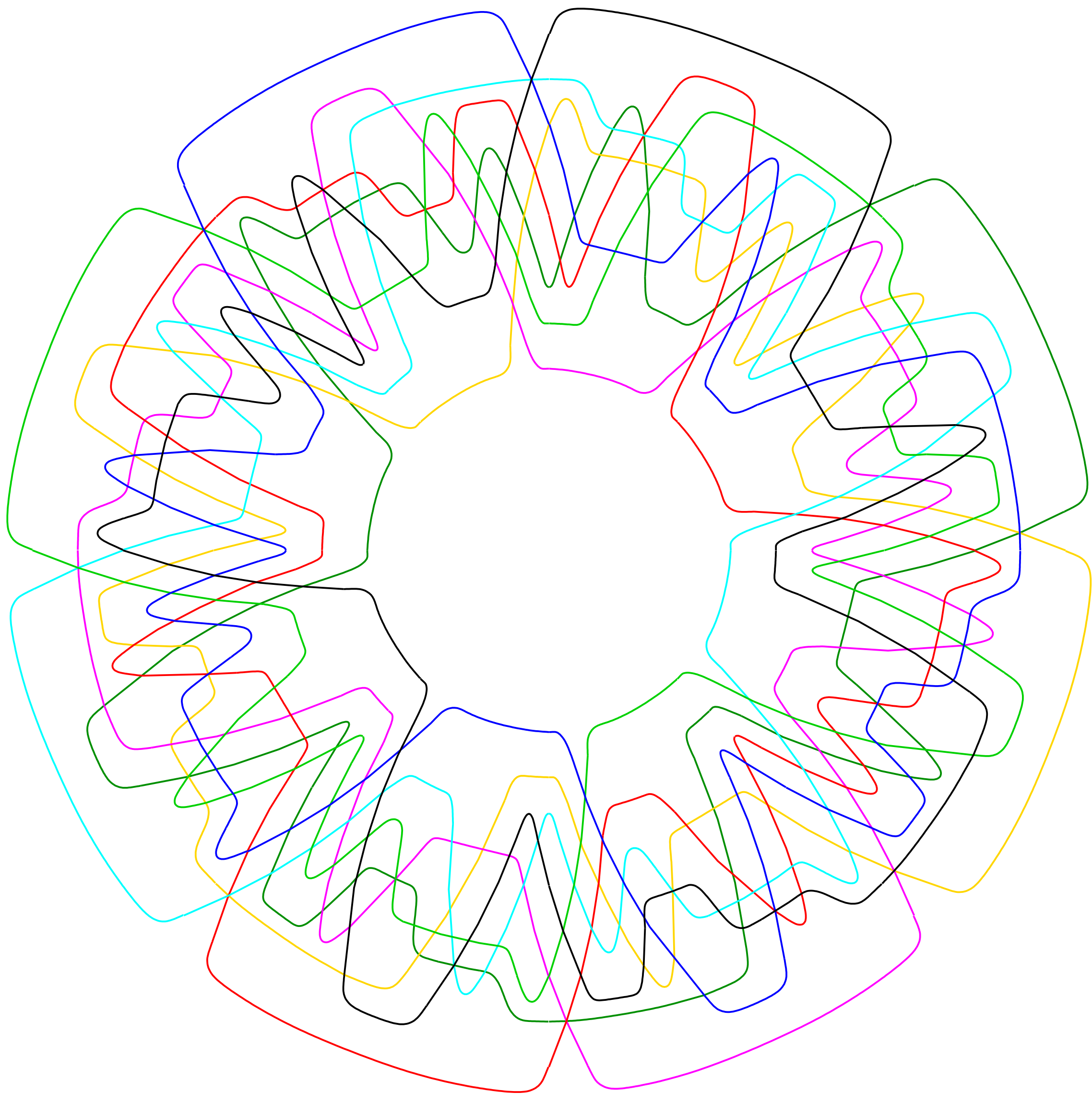


Figure 3.9: Minimal symmetric independent family of 8 curves that is near-simple.

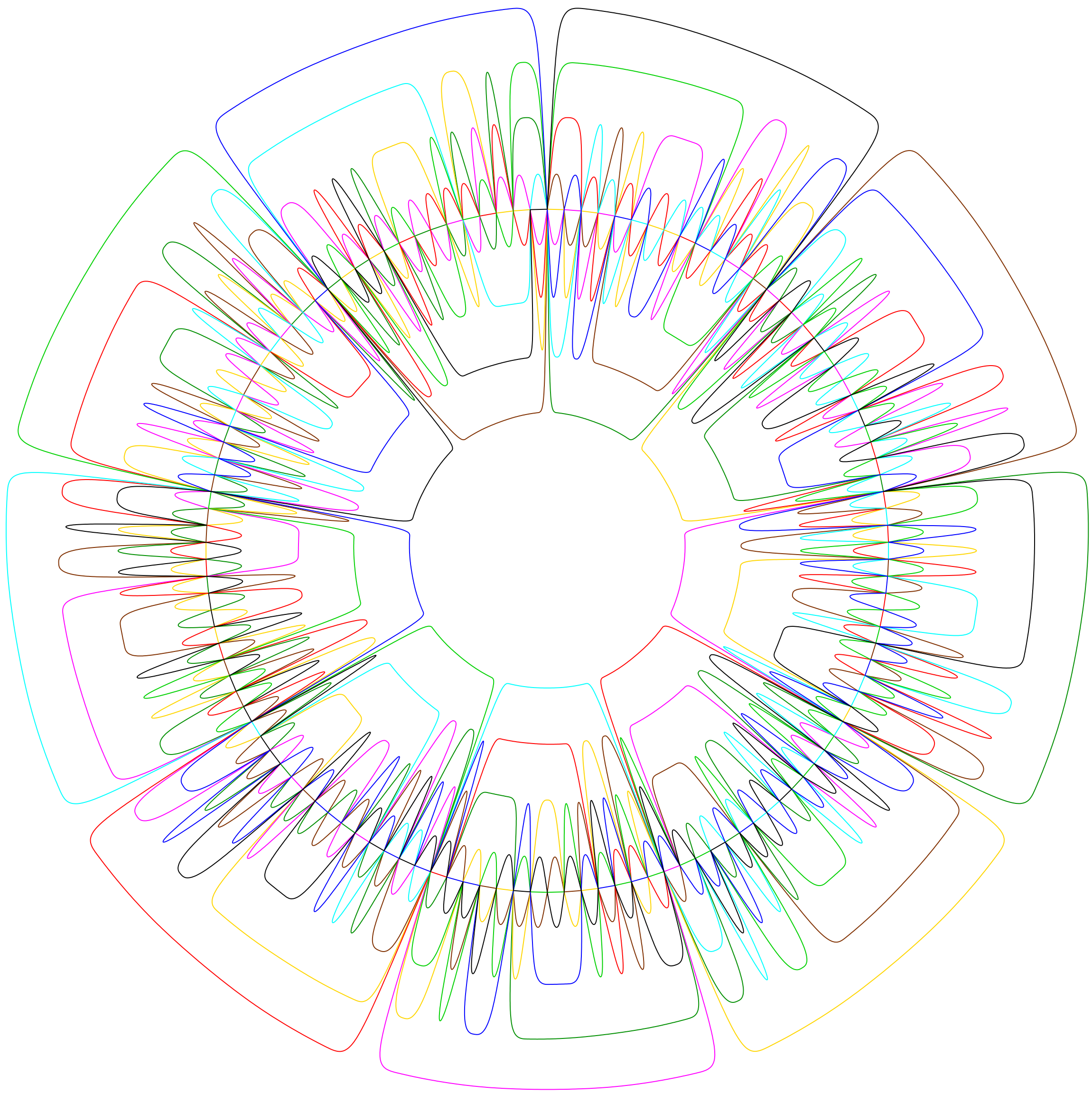


Figure 3.10: Minimal symmetric independent family of 9 curves.

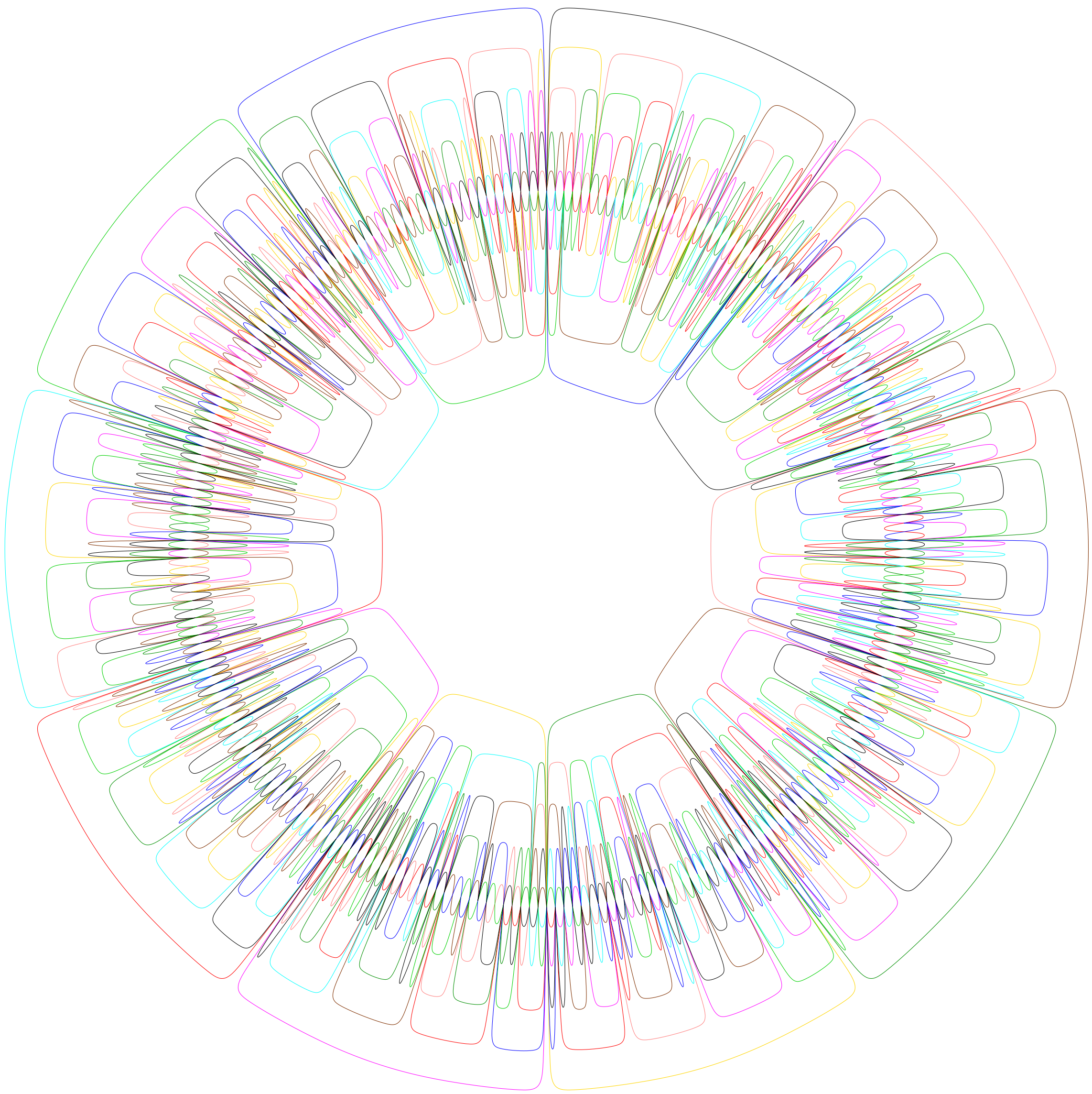


Figure 3.11: Minimal symmetric independent family of 10 curves.

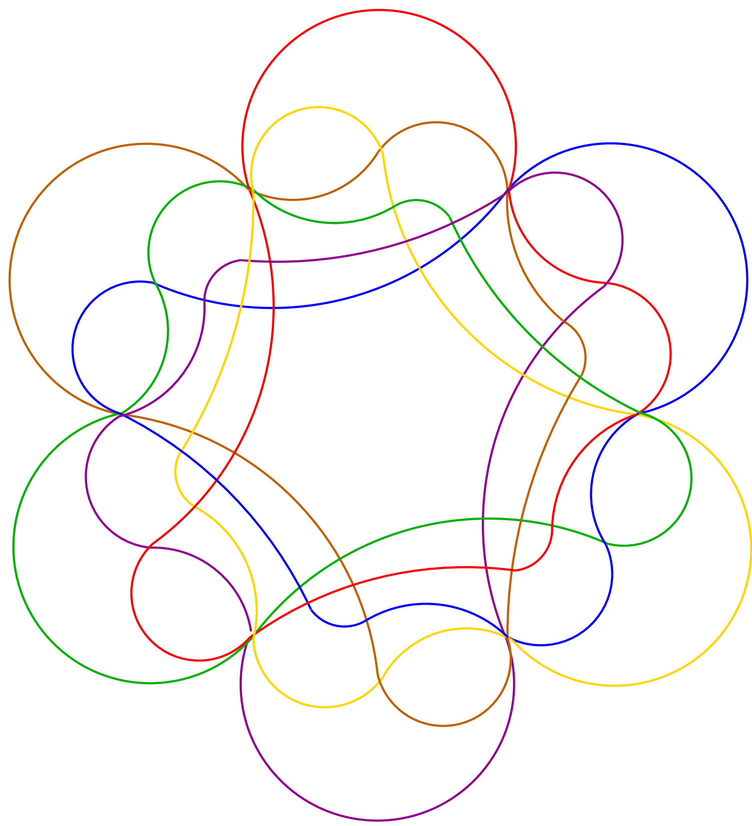


Figure 3.12: Maximal symmetric near-Venn diagram of 6 curves that is near-simple.

4. Pseudo-Symmetric Venn Diagrams

Continuing in the theme of generalizing the concept of symmetry to diagrams with a non-prime number of curves, we now turn our attention to generating Venn diagrams that are not symmetric but contain some pleasing visual symmetries and in fact are very "close" to the symmetric Venn diagrams we have generated in some sense.

Once again we consider an example: Figure 4.1 shows a 4-Venn diagram with some interesting properties. The diagram is obviously not symmetric, however it has a 2-fold "structural symmetry": if we ignore the curve colourings, the diagram maps onto itself by rotating it $180° = \pi$ radians about the centre point. Another way of viewing it is that the diagram is composed of 2 copies of the "pie-slice" forming the top 1/2 of the diagram, with the curves labelled in the second copy from the first by a permutation of the colours in the first pie-slice.

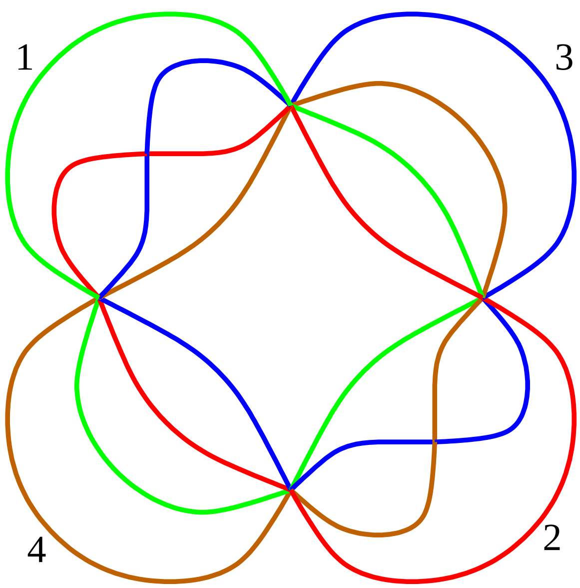


Figure 4.1: 4-Venn diagram with 2-fold pseudo-symmetry.

Definition 1 *An n-Venn diagram $\mathcal{V}$ has k-fold* pseudo-symmetry *with $1 < k < n$ if there is a pie-slice making up $1/k$th of $\mathcal{V}$ so that the rest of $\mathcal{V}$ can be generated* *by repeatedly rotating the pie-slice and recolouring the curves according to a fixed permutation.*

The diagram in Figure 4.1 has 2-fold pseudo-symmetry, and the curves are labelled so that the permutation to be applied is (1234): note that since $k = 2$ the permutation is only applied once, to obtain the bottom half of the diagram from the top half.

This idea forms a natural extension of traditionally symmetric Venn diagrams like the ones we considered in the previous section: there, a single pie-slice forming $1/n$th of the diagram was rotated n times and the permutation $(12 \cdots n)$ was applied to obtain the rest of the diagram. This motivates a generalization of this idea to a Venn diagram with two parameters, namely n and k, and a similar process of generating the rest of the diagram from the pie-slice forming $1/k$th of the diagram; namely copying the pie-slice and rotating the labels. Unlike symmetric Venn diagrams, however, pseudo-symmetric diagrams do not have the property that the last copy of the pie-slice, when rotated once more and the permutation applied to the curve labels, gives the first set of labels again (since $k < n$, applying $(12 \cdots n)$ exactly k times will not return a curve label to itself).

This definition also implies that the diagram maps onto itself by a rotation of $2\pi/k$ ignoring the curve colourings, however the extra condition about the permutation being fixed between successive copies of the pie-slice seems to be necessary, in that it is quite conceivable that there exists a diagram that maps onto itself by a rotation of $2\pi/k$ but that does not use the same permutation between successive copies of the pie-slice before the last and the first: we can refer to this condition as *weak pseudo-symmetry*. Though an interesting idea, we will not consider weak pseudo-symmetry further.

The diagram in Figure 4.1 also maps onto itself, ignoring curve colourings, by reflection about two orthogonal axes, but this property in general will not apply to other pseudo-symmetric diagrams we construct. Note that we do not say that a normal symmetric n-Venn diagram has n-fold pseudo-symmetry, as this would lead to confusion when we consider the differences between pseudo-symmetry and normal symmetry.

4.1 Properties of Pseudo-symmetric Diagrams

Before considering how to generate pseudo-symmetric diagrams, we prove some interesting properties about them. In considering Venn diagrams, we used the fact that the permutation to be applied was always circular and could thus be assumed to be the permutation $(12\cdots n)$ by relabelling the curves. For pseudo-symmetric diagrams, this is also true: that the subsequent copies of the $1/k$th pie-slice are always generated by applying a circular permutation of the curve labels in the first pie-slice.

Lemma 3 *The fixed permutation π applied to the pie-slice making up $1/k$th of a pseudo-symmetric diagram to generate the rest of the diagram must be circular.*

Proof.

Since a pseudo-symmetric diagram with k-fold pseudo-symmetry can be viewed as being composed of k copies of a pie-slice forming $1/k$th of the diagram, the $2^n - 2$ different labels of regions must be divided up into k sets of equal size, one per pie-slice. We can consider all sets but the first as being generated from the first by one or more applications of π to each label, viewed as a bitstring, from the first set.

We assume π is not circular and derive a contradiction. If π is not circular, then consider the first cycle in π of length $j < n$, *i.e.* the first cycle in $\pi = (\pi_1\pi_2\cdots\pi_j)(\pi_{j+1}\cdots)(\cdots\pi_n)$. Since $\mathcal{V}$ is a Venn diagram, it must contain a region corresponding to the label $w = 1^j0^{n-j}$ (regardless of the curve labelling), which, when π is applied to it, would clearly map onto itself. Thus w cannot be in any of the aforementioned sets of region labels but the final set (the kth copy of the pie-slice), since if it were in any previous set it would map onto itself in a subsequent set and create a repeated region in $\mathcal{V}$. However, each label in the non-final sets must have a corresponding label in the final set that they map onto via repeated applications of

π. This implies that w is an extra element in that final set, contradicting the fact that the sets must have the same cardinality.

□

Since the curve labellings are arbitrary, as we did for symmetric Venn diagrams we can relabel the curves so that the permutation to be applied is most convenient for us, and for the rest of the discussion we assume that the permutation π is $(12 \cdots n)$. For example, the curves in Figure 4.1 are labelled so that the top half maps onto the bottom half by applying the permutation (1234).

Lemma 4 *k-fold pseudo-symmetric n-Venn diagrams can only be constructed if k is prime and $n = k^\alpha$ with $\alpha > 1$.*

Proof.

Since a pseudo-symmetric diagram with k-fold pseudo-symmetry can be viewed as being composed of k copies of a pie-slice forming $1/k$th of the diagram, the $2^n - 2$ different labels of regions must be divided up into k sets of equal size, one per pie-slice. Moreover regions of a given weight must also be divided evenly into the k sets as a region of a given weight cannot be shifted by any permutation onto a region of smaller or larger weight. Thus for k-pseudo-symmetric diagrams to exist for n, k must divide $\binom{n}{i}$ for all $i, 0 < i < n$. This condition implies that k is prime and that $n = k^j$ for some $j \geq 1$, via Theorem 13 in [15]. □

In Lemma 4, n being a prime power is not a sufficient condition for the existence of pseudo-symmetric diagrams unless we can exhibit a general construction. Our examples later provide valid diagrams for $n = 8$ and 9.

The following Lemma illustrates the major difference between pseudo-symmetry and symmetry in Venn diagrams.

Lemma 5 *In a pseudo-symmetric n-Venn diagram, all n curves have distinct shapes.*

Proof. Assume the diagram has k-fold pseudo-symmetry, with $1 < k < n$. Consider the pie-slice forming $1/k$th of the entire diagram, which rotates k times to give the entire diagram, and the permutation $\pi = (12 \cdots n)$ as given above that is applied to the curve labels each time to obtain the next pie-slice.

Consider the set of edges of a given colour, say colour 1, in this pie-slice. These edges form a connected path of that colour from one side to the other. We call this path a *curve segment*. There are n different curve segments passing through the pie-slice, one corresponding to each colour, and we can label these n segments by their colour in the first pie-slice: call s_i the segment with colour i. Each segment must be distinct in shape from all others. In the final diagram this pie-slice is rotated k times and the curve labels are incremented each time by applying the permutation π.

Thus for example in the second copy s_1 becomes colour 2 by applying π, in the third copy s_1 becomes colour 3, and in general in the jth rotation of the pie-slice, $1 < j \leq k$, s_1 becomes colour $(j+1) \bmod n$. Moreover, segment s_i must be colour $(i+j-1) \bmod n$ in the jth copy, with $0 \equiv n$.

Consider the inverse permutation function π^{-1} defined by $\pi^{-1}[x] = y$ if and only if $\pi[y] = x$. In general the segment of colour i is s_i in the first copy, $s_{\pi^{-1}[i]}$ in the second copy (*i.e.* it is the segment whose colour shifts to colour i), so the curve segment of colour i in the jth copy of the pie-slice must be $s_{(i-j+1) \bmod n}$, with $1 \leq i \leq n$ and $1 \leq j \leq k$.

Now consider that the curve of colour i is made up of the cycle formed by joining all of the segments of colour i. Define the sequence of segments followed by curve i as the circular sequence $\mathcal{S}(i) = (s_i, s_{(i-1) \bmod n}, \ldots, s_{(i-k+1) \bmod n})$, with $1 \leq i \leq n$. Since $k < n$ it is clear that $\mathcal{S}(i) \neq \mathcal{S}(j)$ for $i \neq j$ given the permutation π. The sequence of segments of length k can be thought of as a "window" of size k moving around the circular list of segments, which has length n. Thus curves i and j are composed of a different circular sequence of distinct segments, and thus are distinct. $\square$.

As an example of Lemma 5, consider the 4-Venn diagram with 2-fold pseudo-symmetry in Figure 4.1. In the top half of the diagram that is rotated to obtain the bottom half, we can label the segments as green = 1, red = 2, blue = 3, and brown = 4. Then $\mathcal{S}(1) = (s_1, s_4)$ (*i.e.* the green curve follows segments 1 and then

4), $\mathcal{S}(2) = (s_2, s_1)$, $\mathcal{S}(3) = (s_3, s_2)$, and $\mathcal{S}(4) = (s_4, s_3)$. The four distinct curves are shown in Figure 4.2.

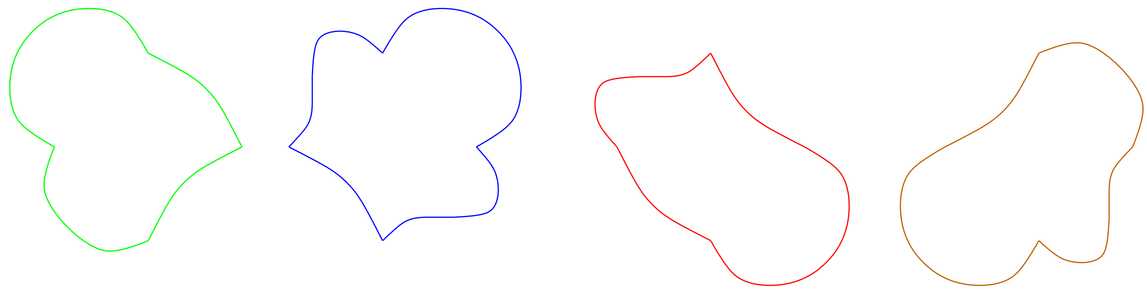


Figure 4.2: The 4 distinct curves of the 4-Venn diagram with 2-fold pseudo-symmetry.

4.2 Pseudo-Symmetry and Doodles

To find a useful method of constructing pseudo-symmetric Venn diagrams, once again the concept of a doodle can be applied. Consider again Figure 4.1: the dual of this diagram contains n/k doodles which can be rotated k times to produce the entire dual; the two doodles are shown in Figure 4.3 (the blue vertices are identified).

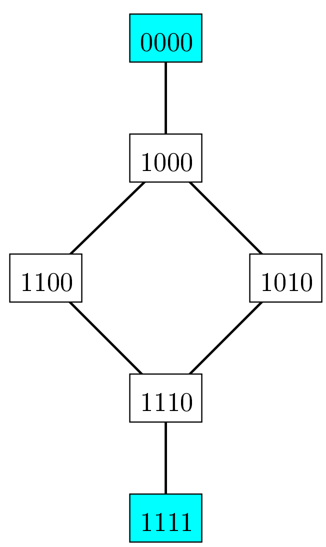


Figure 4.3: Doodles of half of the pseudo-symmetric 4-Venn diagram; blue vertices of equal weight are identified.

As we noted, our curve numbering results from using the permutation $(12 \cdots n)$

to generate successive copies of the $1/k$th pie-slice, and the resulting labelling follows the pattern that the curve labels are incremented by k moving clockwise around the outer face, stopping when we have labelled each curve in the n/k doodles forming the dual of the pie-slice. For example in Figure 4.3 the curves are labelled 1,3,2,4 moving clockwise around the outer face of the diagram in Figure 4.1, with green as colour 1.

Note that the first doodle in Figure 4.3 contains a label for every necklace, and so we could use it by itself to generate a symmetric independent family of four curves using the method in the previous section; however we will now use it for a different purpose.

The second doodle in Figure 4.3, consisting of a labelled path, is a copy of the first doodle with all labels rotated by two, but with the node 1010 deleted as it is labelled by a non-Lyndon word: rotating it by two would map it onto itself and cause a repeated region. The two doodles in Figure 4.3 can then be copied to form the other half of the Venn dual, and all labels rotated by one, to create every bitstring in the dual. Thus the region 1010 appears exactly once, and it is rotated to form the region 0101 which also appears exactly once. Thus we have ensured that any region corresponding to a periodic label appears exactly once, and the resulting diagram is a Venn diagram (as regions labelled by Lyndon words also appear exactly once as they do not rotate onto themselves). This is similar to our statement that any doodle for an independent symmetric family can be turned into a doodle for a symmetric near-Venn diagram by deleting nodes labelled by non-Lyndon words; except here we make a copy of the doodle and delete the nodes labelled by non-Lyndon words in the copy.

4.2.1 Generation Method

This gives us a method of constructing pseudo-symmetric diagrams: given the doodle of a symmetric independent family containing nodes labelled by non-Lyndon words of rotational period k, we:

1. Make n/k copies of the doodle, rotating the node labels by k in subsequent

copies from the first.

2. In all copies except the first, delete nodes that pertain to repeated copies of non-Lyndon words of period k, assuming that the structure of the doodle can be repaired (*i.e.* any nodes that are not on a path from 0^n to 1^n can be assigned as children of other nodes, possibly by permuting the structure of the doodle without destroying planarity).

3. If there are non-Lyndon words with period m a multiple of k and $m < n$, delete copies of these nodes in every doodle past the first m/k.

   Step 3 accommodates the fact that for some n, non-Lyndon words may have periods that are multiples of periods of other non-Lyndon words: for example for $n = 8$, the label 10001000 has period four, whereas 10101010 has period two. Thus in the four copies of the doodles we want to keep the labels 10001000 and its first rotation by two, 00100010, but subsequent rotations by two would create repeated labels, so we delete the repeats.

4. To construct the Venn dual, rotate the n/k copies k times, rotating the node labels by one in subsequent copies.

Note that, as a consequence of Lemma 4, this construction depends on there not being non-Lyndon words with two different rotational periods that are relatively prime in the doodle. For example, for $n = 6$ the bitstrings 100100 and 101010 have periods three and two respectively and thus the above construction would not work: the doodle would need to be copied $k = 2\times 3 = 6$ times to accommodate the periods of rotation, which would give rise to a Venn diagram with 1-fold pseudo-symmetry – *i.e.* no pseudo-symmetry at all. This can easily be seen by considering that the symmetric independent family in Figure 3.8 has repeated regions both at points distributed at intervals of $2\pi/3$ around the circle, corresponding to labels of period two, and at points at intervals of π around the circle, corresponding to labels of period three. Thus removing exactly the extra repeated regions could only create a diagram whose structure repeats every six sectors.

As alluded to in Step 2 above, the doodle also has to have the property that the non-Lyndon words can be removed and the result can still be a valid doodle: without loss of generality, assume that the non-Lyndon word in the doodle has weight $k < \lfloor n/2 \rfloor$. Then if it is removed all of its children must have the property that they can also be children of other nodes of weight k. Setting new parents for the orphaned children may entail manually permuting substructures in the doodle to preserve planarity. If the doodle has this property we call it *repairable*.

From the necessary conditions we have discussed, any repairable doodle for a prime power n containing every representative for every necklace can be used to create pseudo-symmetric diagrams. We have previously used such doodles to create symmetric independent families for $n = 8$ and 9, the first prime powers above four, but in most cases the doodles were not repairable. We now outline a construction for prime n that we can easily modify to create doodles for n being a small prime power. Though our modification is manual and has not yet been generalized, it appears that every doodle created in this manner for prime powers can be easily repairable and has other nice properties.

4.3 Savage's Construction

Our backtracking constructions in the previous chapter depended upon creating a doodle that corresponded to a sector that was then rotated to create a symmetric diagram. As we noted above, pseudo-symmetric diagrams can be created from a doodle of a symmetric independent family if n is a prime power and if the doodle is repairable. A recent result by Savage *et al* in [14] produces doodles for symmetric Venn diagrams for any prime n by producing a symmetric chain decomposition of a set of necklace representatives (chosen in a particular way) that can then be embedded in the plane with some extra edges to form the doodle of a symmetric Venn diagram. In the following sections we outline the construction and then apply it to create repairable doodles of pseudo-symmetric Venn diagrams by applying the construction, with minor modifications, when n is a prime power, and then copying and rotating the doodle appropriately.

Savage *et al*'s construction of a valid doodle for n prime is based on constructing a symmetric chain decomposition of a poset of necklace representatives and showing the chains can be combined into a valid doodle. We will simply describe the construction; proofs of correctness can be found in [14]. First we describe how the set of necklace representatives are chosen and then describe the construction of the doodle from these representatives.

With each node label x of length n, associate a *block code* $\beta(x)$ over $\{2, \ldots, n, \infty\}$ as follows:

If

$$x = 1^{a_1}0^{c_1}1^{a_2}0^{c_2}\cdots 1^{a_k}0^{c_k}, k > 0, a_i > 0, c_i > 0, 1 \le i \le k,$$

Then

$$\beta(x) = (a_1 + c_1, a_2 + c_2, \cdots, a_k + c_k).$$

Otherwise we define $\beta(x) = (\infty)$.

For example, consider the block codes of rotations of $x = 00101$: $\beta(00101) = (\infty)$, $\beta(10010) = (3, 2)$, $\beta(01001) = (\infty)$, $\beta(10100) = (2, 3)$, and $\beta(01010) = (\infty)$.

Any string x has n different rotations (which, if n is prime, are all distinct) and we choose one of the rotations $\rho(x)$ to be in our necklace representative poset as follows:

> $\rho(x)$ is the rotation y of x for which the block code $\beta(y)$ is lexicographically minimum.

We let R_n be the set of necklace representatives, removing only the complete set and the empty set as they are not included in our doodle construction:

$$R_n = \{x \in \{0,1\}^* \mid \rho(x) = x\} - \{0^n, 1^n\}.$$

To construct the doodle, we define a set of symmetric chains in R_n with the property that every $x \in R_n$ is included in exactly one chain. These chains are specified by choosing elements to start the chains and then specifying how subsequent elements in the chain are produced, and finally specifying the terminator of each chain. The symmetric chains are constructed using the "parenthesis matching" algorithm of Greene and Kleitman [4] to produce a symmetric chain decomposition of the poset formed by the set R_n and the set inclusion relation.

In a string $y \in \{0,1\}^*$, if we view the 0's as left parentheses and 1's as right parentheses, we can match parentheses in the usual way: proceeding left to right in the string, a 0 is matched to the closest previously unmatched 1 to its right, and vice versa. We will build the chains, starting from a given chain starter $x = x_1 x_2 \cdots x_n$, using the function τ:

$$\begin{aligned} \tau(x) &= x_1 \cdots x_{i-1} 1 x_{i+1} \cdots x_n \\ &\qquad \text{where } i \text{ is the index of the left-most unmatched 0 in } x. \end{aligned}$$

1. Define the set of chain starters as the set of strings for which the only unmatched 1 occurs in position 1.

2. For a chain starter $x \in R_n$, let $k(x)$ be the number of unmatched 0's in x.

3. Define the chain of x, J_x, as

$$J_x = x, \tau(x), \tau^2(x), \ldots, \tau^{k(x)}(x).$$

It can be shown that the chains thus constructed are symmetric about the middle rank $(n-1)/2$, *i.e.* if the weight of a chain starter x is d, the chain J_x will have length $n - 2d$ and thus the weight of the final element in the chain, $\tau^{k(x)}(x)$, is $n - d$. Also, every element in R_n is contained in exactly one chain, and the chain terminators are the strings for which the only unmatched 0 occurs in position n. It is also true that the block code $\beta(x)$ is not changed by the operation of τ, so $\beta(y) = \beta(x)$ for every element $y \in J_x$.

Finally we need to stitch the chains together to form a valid doodle: once the symmetric chains are embedded in the plane with elements of R_n as nodes and edges drawn between elements with the cover relation, edges can be added from chain starters to elements in other chains by the "cover relation":

> For a chain starter $x = x_1x_2 \cdots x_{j-1}100 \cdots 0$, *i.e.* x_j is the 1 with highest index in x, its cover element is the element $x_1x_2 \cdots x_{j-1}000 \cdots 0$.

It can be shown that the cover element for all chain starters (except 10^{n-1}) is also an element of R_n. The chain terminators are connected to the symmetric element in the chain containing the cover element of the chain starter (preserving the symmetry by reflection about a horizontal axis of the structure of the embedding).

For n prime this construction produces a valid doodle that corresponds to a minimum-vertex monotone symmetric n-Venn diagram. For further details see [14].

4.4 Modification of the Construction

Given the construction above, we now apply it to the case where n is non-prime but is a prime power and then use the doodle constructed to produce pseudo-symmetric Venn diagrams. We do our modification of the construction by hand; it is not a general method to construct symmetric chain decompositions of the necklace representative poset R_n for n a prime power.

First we consider which n are useful to look at: for $n = 2^2 = 4$ we have seen an example of a pseudo-symmetric diagram; there is one other pseudo-symmetric diagram for $n = 4$ that differs from Figure 4.1 in the curve segments but has the same structural symmetry. For $n = 2^3 = 8$ and $3^2 = 9$ the diagrams are easily constructed by hand, as we will see. The next prime power, $n = 2^4 = 16$, is far too large to be constructed by hand (consider that the doodle would have $2^{16}/16 = 4096$ vertices), so we will restrict our attention to the $n = 8$ and 9 cases. If n is even ($n = 8$) we can get chains of length one; this poses no particular problems.

By applying the construction for $n = 8$, all of the chains composed of Lyndon words (and thus having Lyndon words as chain starters) are constructed as per the construction explained above, and we must also use the non-Lyndon word 10001000 as a chain starter; this is natural as $\beta(10001000) = (4, 4)$ is the lexicographically lowest block code of its four distinct rotations (for the other three, the block code is ∞), and also as 10001000 has no unmatched 1's except in the first position (like the other chain starters). At the middle level (depth four), it is natural to choose the string 10101010 as a chain starter to form a chain of length one. The necklace representative 10011001, however, does not fit in nicely as it creates a choice of either picking a necklace representative with minimal block code (*i.e.* 11001100), or with the only unmatched 1 in the first position (*i.e.* 10011001). The string 10011001 would cause problems, however, as the final bit is set and it would not fit it into the diagram as a child of 10011000 and parent of 10011110. However 11001100 works as a child of 11001000. The resulting doodle is shown in Figure 4.4 with the non-Lyndon words highlighted: the green labels have period four and the yellow label has period two. Note that this doodle could be used as-is to create a symmetric independent family of eight curves, as in the previous section, but we wish to use it to create a pseudo-symmetric Venn diagram.

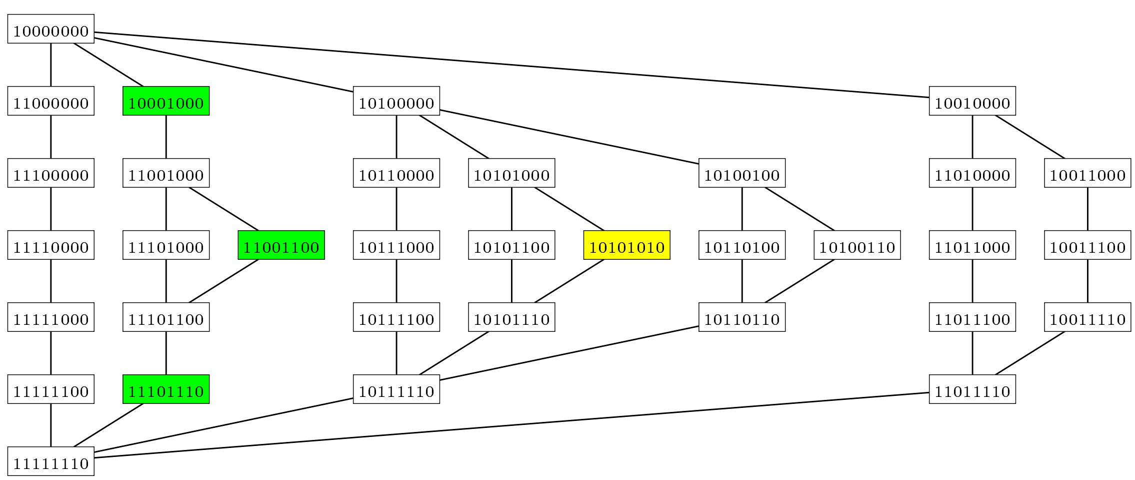


Figure 4.4: Doodle for pseudo-symmetric 8-Venn diagram.

Note that the doodle in Figure 4.4 is obviously repairable: the highlighted labels can all be removed and the resulting doodle is still planar if the node 11001000, orphaned by the removal of its parent 10001000, is made a child of the node 11000000, next to 10001000 (and similarly 11101100 can be made a parent of 11111100). Figure 4.5 shows the four (different) doodles obtained by copying the doodle in Figure 4.4 three times, incrementing the curve labels by two each time, and deleting the appropriate copies of the non-Lyndon words. The doodles are drawn vertically, with the starting doodle at the bottom; the labels are rotated $0, 2, 4$, and then 6 places right in each copy of the doodle going up from the bottom. The non-Lyndon word of period two only appears once in the four doodles and those of period four appear twice. These four doodles form the dual of half of the final 8-Venn diagram; the other half is obtained by copying these four doodles once more and incrementing the curve labels by one (and making the circular dual).

The final diagram that results is shown in Figure 4.6. Though at first glance the diagram looks symmetric, upon closer inspection the several "missing" regions due to the deletion of the labels in the doodles make it clear that the diagram does in fact have only 2-fold pseudo-symmetry.

The construction for $n = 9$ is simpler as there are only two non-Lyndon words in the doodle, both with period three: 100100100 and 110110110. The construction creates three doodles, only one containing the two non-Lyndon words, the other two being rotated three and six places, and the entire set of three doodles is copied three times, incrementing the curve labels by one each time, to create the final diagram. The pseudo-symmetric diagram that results is shown in Figure 4.7. Like the 8-Venn diagram in Figure 4.6, it is difficult to tell that this diagram is not a symmetric Venn diagram: the blown-up areas highlight the parts of the diagram that differ between the "sectors" making up 1/9th of the diagram; the highlighted areas only repeat every three sectors, as the diagram has 3-fold pseudo-symmetry.

Since the doodles have vertical symmetry in their structure, the Venn diagrams that result have a pleasing "inside-out" symmetry (though not polar symmetry), ignoring the curve colourings.

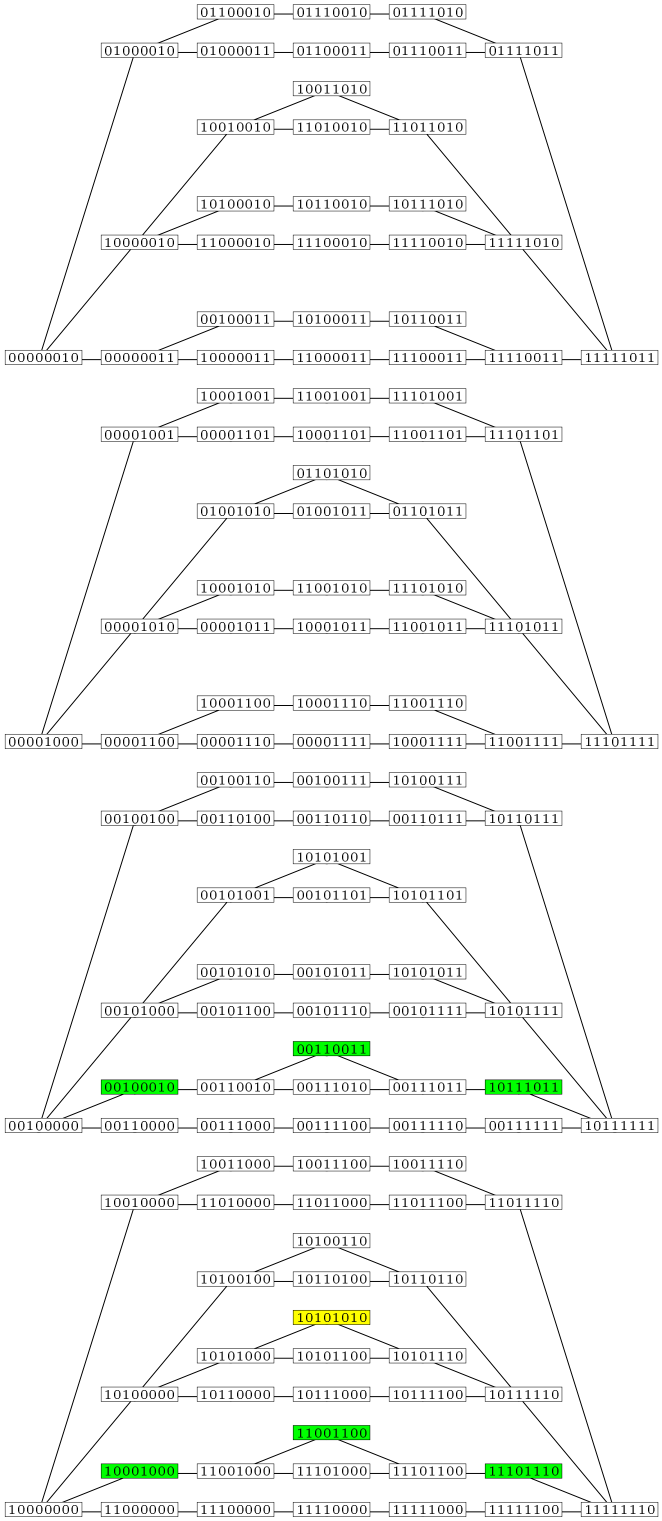


Figure 4.5: Four distinct doodles forming half of the dual of the pseudo-symmetric 8-Venn diagram.

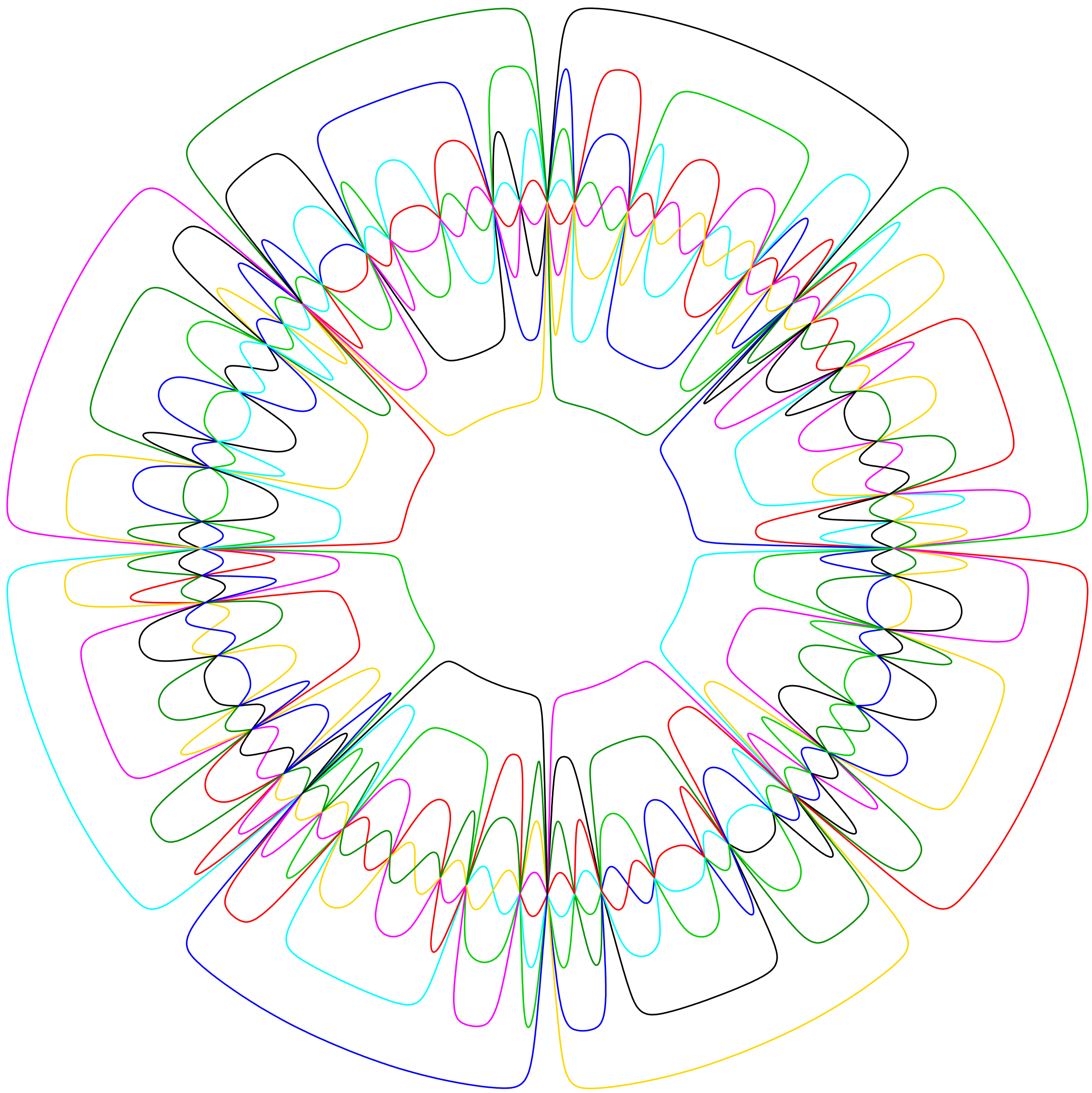


Figure 4.6: Pseudo-symmetric 8-Venn diagram with 2-fold pseudo-symmetry.

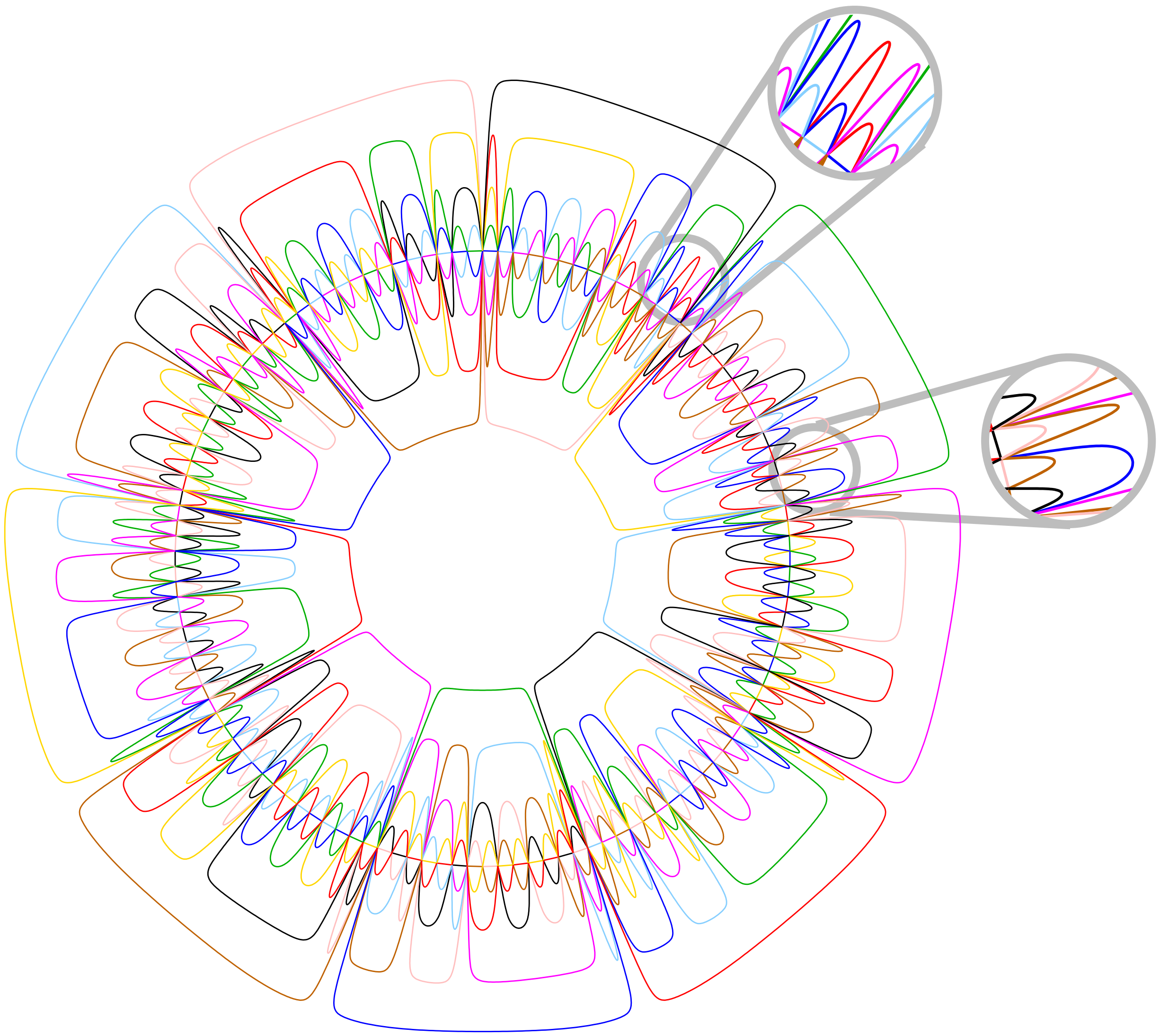


Figure 4.7: Pseudo-symmetric 9-Venn diagram with 3-fold pseudo-symmetry, with blown-up regions to illustrate the small differences between successive sectors.

5. Conclusions

In this thesis, we explored the concept of symmetry in Venn diagrams by utilizing Grünbaum's generalization and a few other ideas, with the goal to extend the possibilities of computation beyond prime values of n and to produce interesting Venn diagrams and non-Venn diagrams that are, in some sense, "close" to Venn diagrams. Our contributions include creating a backtracking methodology for creating symmetric diagrams, closing several cases of an open problem of Grünbaum and extending symmetry to near-Venn diagrams, and characterizing and providing examples of pseudo-symmetric diagrams for $n = 4, 8$ and 9. The diagrams we have created for $n = 8, 9$ and 10, both Venn diagrams and non-Venn diagrams such as symmetric independent families and near-Venn diagrams, are the first diagrams of this size that we know of that have interesting properties – besides diagrams created using the general constructions for all n (such as those outlined in [12]).

5.1 Open Problems

The field of Venn diagrams, as noted in the Introduction, is fairly new and there are many interesting areas of research that can be explored. We highlight just a few that are directly related to this research.

5.1.1 More Enumeration Results

As we noted in Chapter 3, our enumeration results for $n = 5$ and 7 would be more useful if we actually knew how many Venn diagrams there were for either case (though for $n = 5$ at least we know how many symmetric diagrams there are). Counting Venn diagrams in general (not necessarily symmetric) is a difficult problem; results are only known for very small n (see [12] for the latest figures; at the time of writing the total number of diagrams was known up to $n = 4$ and for simple diagrams up to $n = 5$). An approach like the one we have used for symmetric diagrams does not generate all symmetric diagrams; other techniques such as those outlined in [2] could possibly be extended or revisited to count Venn diagrams in general.

Similarly, counting all symmetric Venn diagrams for $n = 7$ would be a non-trivial task: as we noted, our generation technique for them, which creates many duplicates and certainly misses many diagrams, generated almost 700,000 doodles corresponding to nonisomorphic diagrams. Extending this method or exploring a new method to count exactly how many symmetric 7-Venn diagrams exist, and their properties, would be an important result.

5.1.2 Simple Symmetric Independent Families and Near-Venn Diagrams

Grünbaum conjectured in [7] that simple symmetric independent families with the minimal number of regions do not exist, and indeed we were unable to find one for larger values of n than he considered; additionally, we also were unable to a find a symmetric near-Venn diagram with the maximal number of regions that was simple. A proof of this result in general could be illuminating, and it may provide directions towards finding simple Venn diagrams.

5.1.3 Finding larger Venn diagrams

As we noted in Chapter 3, our backtracking method works well for finding doodles for n up to ten, and we were able to generate all of the possible doodles using the backtracking approach for n up to seven. However it appears the case for $n = 11$ is currently out of reach using conventional backtracking approaches; other approaches like those described in [2] are similarly too slow for $n = 11$. It is possible that the backtracking method we use here could be refined enough to produce significant numbers of diagrams for $n = 11$.

5.1.4 Pseudo-symmetric diagrams

Pseudo-symmetric diagrams are a useful way of creating large Venn diagrams, and a general method of constructing these would enable us to quickly create large diagrams for n a prime power. Following our ad hoc method, this would entail creating two related pieces of work:

1. A symmetric chain decomposition for the necklace representative poset for all n (or at least n a prime power).

2. A general method of modifying the symmetric chain decomposition to be able to remove any non-Lyndon word from the embedding of the chain decomposition and still repair the doodle.

As we noted a pseudo-symmetric diagram for $n = 16$ is too large to construct entirely by hand. However, it could be done by using a computer to generate all of the symmetric chains headed by Lyndon words and then manually inserting the chains headed by non-Lyndon words, as there are only 36 non-Lyndon words of length 16. Drawing such a diagram would obviously be impractical.

Many more open problems related to Venn diagrams can be found in [12].

Bibliography

[1] B. Bultena, B. Grünbaum, and F. Ruskey. Convex drawings of intersecting families of simple closed curves. In *11th Canadian Conference on Computational Geometry*, pages 18–21, 1999.

[2] S. Chow and F. Ruskey. Searching for symmetric Venn diagrams. Manuscript, 2 pages, 1998.

[3] R. Graham, D. Knuth, and O. Patashnik. *Concrete Mathematics: A Foundation for Computer Science*. Addison-Wesley, second edition, 1994.

[4] C. Greene and D. Kleitman. Strong versions of Sperner's theorem. *Journal of Combinatorial Theory Ser. A*, 20(1):80–88, 1976.

[5] B. Grünbaum. Venn diagrams and independent families of sets. *Mathematics Magazine*, 48:12–23, 1975.

[6] B. Grünbaum. Venn diagrams I. *Geombinatorics*, 1:5–12, 1992.

[7] B. Grünbaum. The search for symmetric Venn diagrams. *Geombinatorics*, 8:104–109, 1999.

[8] P. Hamburger. Doodles and doilies, non-simple symmetric Venn diagrams. Manuscript, 17 pages, 2001.

[9] P. Hamburger. Pretty drawings. More doodles and doilies, symmetric Venn diagrams. Manuscript, 43 pages, 2001.

[10] D. Henderson. Venn diagrams for more than four classes. *American Mathematical Monthly*, 70(4):424–426, 1963.

[11] B. McKay, W. Myrvold, and J. Nadon. Fast backtracking principles applied to find new cages. In *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms (San Francisco, CA, 1998)*, pages 188–191, New York, 1998. ACM.

[12] F. Ruskey. A survey of Venn diagrams. *The Electronic Journal of Combinatorics*, 2001. http://www.combinatorics.org/Surveys/ds5/VennEJC.html.

[13] F. Ruskey. *Combinatorial Generation*, 2003. Manuscript.

[14] C. Savage, C. E. Killian, and J. Griggs. Venn diagrams and symmetric chain decompositions in the boolean lattice. Preprint, submitted, 29 pages, 2002.

[15] D. Singmaster. Divisibility of binomial and multinomial coefficients by primes and prime powers. In *A collection of manuscripts related to the Fibonacci sequence*, pages 98–113. Fibonacci Assoc., Santa Clara, Calif., 1980.

[16] J. Venn. On the diagrammatic and mechanical representation of propositions and reasonings. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 9:1–18, 1880.

Vita

Surname: Weston Given Names: Mark Richard Nicholas

Place of Birth: Duncan, British Columbia, Canada

Educational Institutions Attended:

University of Victoria	1995-2003

Degrees Awarded:

B.Sc.	University of Victoria	2000

Honours and Awards:

NSERC Postgraduate A Scholarship	2000 - 2001
ASI Graduate Scholarship	2000
Nellie Gillespie Prize	2000
Governor General's Silver Medal	2000
President's Scholarships	1996 - 1999
Howard Petch Scholarship	1999
University of Victoria Faculty Scholarship	1999
John Thipthorpe Memorial Award	1998
Woods Trust Scholarships	1996 - 1997
James Riddell Memorial Scholarship	1996
English Speaking Union Exchange Scholarship	1993
Advanced Placement Scholar with Honours	1993
University of Victoria Entrance Scholarship	1993

University of Waterloo Entrance Scholarship 1993

Publications:

M. Weston, "A Fixed-Parameter Tractable Algorithm for Matrix Domination", accepted for publication by *Information Processing Letters*, 9 pages, 2002.

Partial Copyright License

Title of Thesis:

On Symmetry in Venn Diagrams and Independent Families

Author: ____________________________________

Mark Richard Nicholas Weston

March 25, 2003