# A Preprocessor for a Magic Venn Diagram Counter

ANJA REMSHAGEN AND JAMIA ECHOLS
DEPARTMENT OF COMPUTING AND MATHEMATICS
UNIVERSITY OF WEST GEORGIA

## Overview

Magic Venn diagrams and k-connected circles

The magic Venn diagram counting problem

Solution algorithm with preprocessor

Computational results and discussion

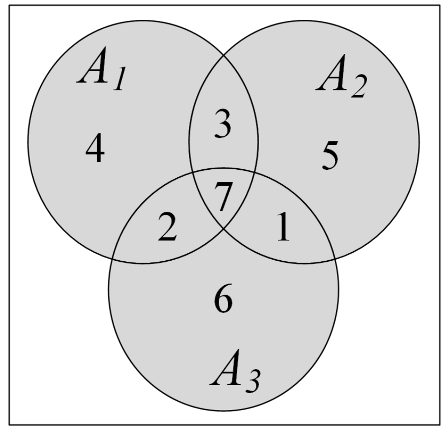
# Magic Venn Diagrams (MVD)

Introduced by David Robinson

A collection of sets $A_1, \ldots, A_n$

Non-empty regions
$R_I = (\bigcap_{i \in I} A_i) \cap (\bigcap_{i \notin I} A_i')$ with $I \subseteq [\![n]\!]$
and $[\![n]\!] = \{1,2, \ldots, n\}$

A labeling of the non-empty regions by consecutive labels



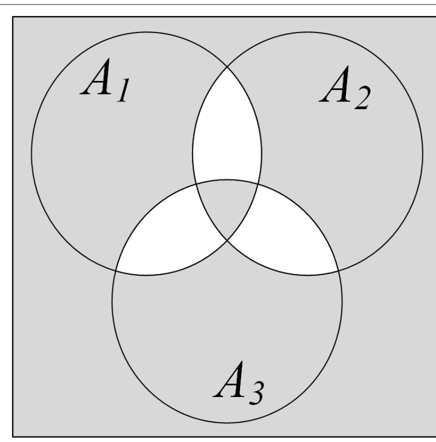

# Regional Support System

The collection $\mathcal{I} = \{I \mid R_I \neq \emptyset\}$ uniquely identifies the regions of a Venn diagram that need to be labeled.

The set $\mathcal{I}$ is the *regional support system* of the Venn diagram.

The regional support system of the Venn diagram in the figure to the right:

$\{\emptyset, \{1\}, \{2\}, \{3\}, \{1, 2, 3\}\}$



# Magic Labeling

Given a support system $\mathcal{I} = \{I \mid R_I \neq \emptyset\}$ of size $|\mathcal{I}|$ = r. A *magic labeling* can be represented by a bijection

$$\lambda : \mathcal{I} \longrightarrow [\![r]\!]$$

such that $\Sigma_{I,\, i \in I}\, \lambda(I) = m$ for a *magic sum* $m$ for all $i \in [\![n]\!]$.

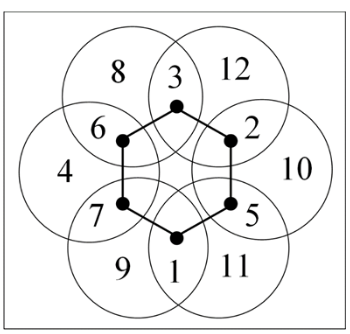
A magic Venn diagram is a pair $(\mathcal{I}, \lambda)$ with regional support system $\mathcal{I}$ and magic labeling $\lambda$.

The order of the MVD is the number $n$ of sets $A_i$.

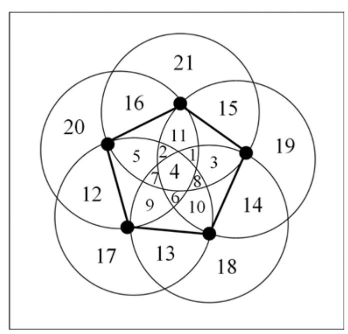
The size of the MVD is the total number $r$ of non-empty regions.

We consider only *regular* magic Venn diagrams $(\mathcal{I}, \lambda)$ with degree $d$ where each set $A_i$ contains exactly $d$ non-empty regions.

# Magic Cycles and Circle Diagrams




Edge-magic graphs by Kotzig and Rosa



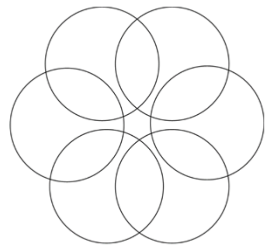

Magic circle diagrams by Robinson, resulting from regular 5-gon.

# k-connected Circle Diagrams

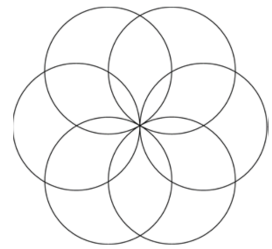
A $k$-connected circle diagram is a circular arrangement of $n$ circles that overlap with its $k$ neighboring circles on each side.

$k$-connected circle diagrams comprise cycles and circle diagrams:

- Cycles are 1-connected circle diagrams
- Circle diagrams of order $n$ are $(n-1)$-connected circle diagrams

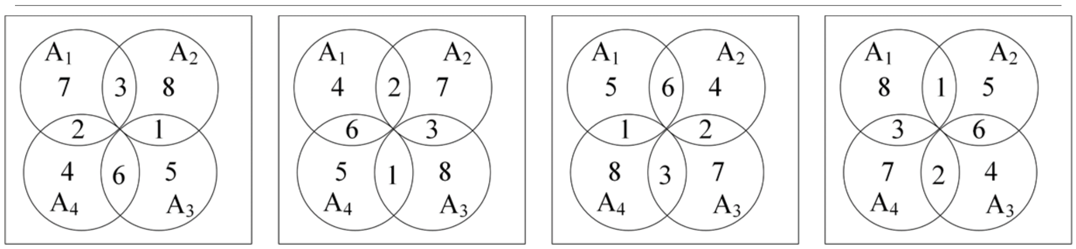


A 1-connected circle diagram of order 6



A 2-connected circle diagram of order 6

# MVDs with the same Structure




Four isomorphic 1-connected circle diagrams (magic cycle graphs) resulting from rotating the labels.

Four additional isomorphic MVDs can be obtained by reflection of the labels.

## Isomorphic MVDs

An isomorphism on a regional support system $\mathcal{I}$ is a bijection h on ⟦n⟧ that maintains the regional support system, i.e.

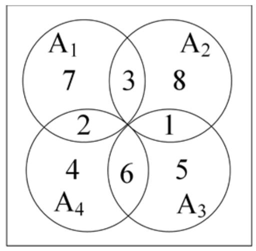
h: ⟦n⟧ → ⟦n⟧

with $h(I) = \{h(i) \mid i \in I\} \in \mathcal{I}$.

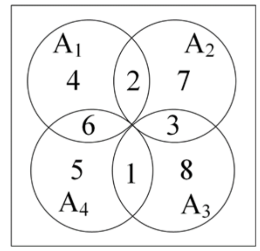
Two MVDs $(\mathcal{I}, \lambda)$ and $(\mathcal{I}, \lambda')$ are isomorphic iff an isomorphism h on $\mathcal{I}$ exists such that

$\lambda(I) = \lambda'(\{h(I)\})$

for all regions $I \in \mathcal{I}$.




h(i) = (i mod 4) + 1




## Properties of Isomorphic MVDs

The isomorphism relation on the MVDs with a given regional support system is an equivalence relation.

The size of each equivalence class equals the number of isomorphisms.

Isomorphic MVDs have the same magic sum.

# The MVD Counting Problem

For a given regional support system, count the *non*-isomorphic MVDs for each magic sum.

Based on the previous discussion, we need to count the number of equivalence classes of the isomorphism relation.

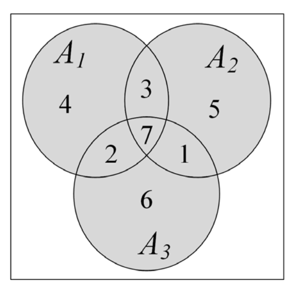
# non-isomorphic MVDs = # isomorphic MVDs / # isomorphisms on $\mathcal{I}$

# Representation of a Labeling

A labeling can be represented by an $n$-tuple where the $i^{th}$ component is a sorted list of the labels of set $A_i$.

For example, the labeling in the image to the right is represented by ((2, 3, 4, 7), (1, 3, 5, 7), (1, 2, 6, 7))

labels of $A_1$ — labels of $A_2$ — labels of $A_3$




The sorted labels can be compared lexicographically.

For example: (1, 3, 5, 7) < (2, 3, 4, 7)
and (1, 2, 6, 7) < (2, 3, 4, 7)
and (1, 2, 6, 7) < (1, 3, 5, 7)

# Selecting a Representative MVD (Case 1)

**Case 1: Every bijection of a support system is an isomorphism**

Then among each group of isomorphic MVDs, there is one where the elements of the n-tuple (i.e. the sorted lists of labels) are listed in lexicographical order.

This is the *standard representation* among all isomorphic MVDs.

For example:

- The labeling ((2, 3, 4, 7), (1, 3, 5, 7), (1, 2, 6, 7)) does not list the element of the triple in lexicographical order.
- The labeling ((2, 3, 4, 7), (1, 3, 5, 7), (1, 2, 6, 7)) is isomorphic to the labeling ((1, 2, 6, 7), (1, 3, 5, 7), (2, 3, 4, 7)) which lists the elements in lexicographical order, i.e. (1, 2, 6, 7) < (1, 3, 5, 7) < (2, 3, 4, 7))

# Selecting a Representative MVD (Case 2)

**Case 2: The regional support system has circular structure**

Then among each group of isomorphic MVDs, there is one where

- the first sorted list of labels is lexicographically smaller than all other sorted label lists and
- the second sorted label list is lexicographically smaller than the last sorted label list.

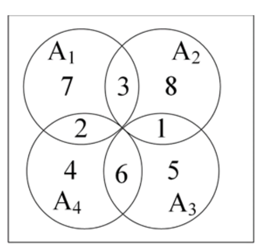
This is the *standard representation* among all isomorphic MVDs.

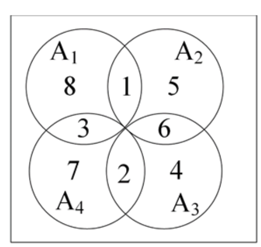
For example:

- Among all isomorphic MVDs of the two examples to the right, the second on with the labeling

  ((1, 3, 8), (1, 5, 6), (2, 3, 7), (2, 4, 6))

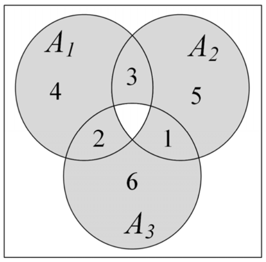
  meets the above criteria.

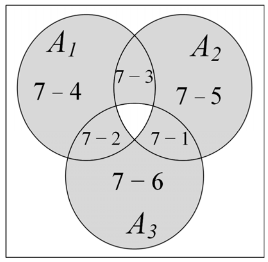


# Duality of MVDs

Due to the duality property, it is sufficient to determine only MVDs with magic sum greater than or equal to the *magic median* $d(r+1)\ /\ 2$.




Two dual MVDs with magic sum 9 and 12, resp.

The magic median is 3(6 + 1) / 2 = 10.5.

# Prior Solution Algorithm

Backtracking search – assigns regions to the labels

- Basically, a brute-force algorithm that tries out all possible assignment of labels to regions.

The backtracking search maintains upper and lower bounds for the magic sum and backtracks if

- the lower bound exceeds the upper bound.
- the upper bound for the magic sum less than to magic median $d(r+1)\ /\ 2$.

Problem

- The algorithm counts isomorphic MVDs.
- The isomorphism factor has to be determined by hand to determine the number non-isomorphic MVDs.

# The Preprocessor

Counts the number of isomorphisms on the regional support system.

- Using Heap's algorithm for generating permutations, generate all bijections and check if the bijection is an isomorphism.

Identifies two types of structures:

1. Every bijection is an isomorphism.
2. Circular structure – exactly the bijections of the form
   $h(i) = ((i + k) \bmod n) + 1$ and
   $h(i) = ((k - i) \bmod n) + 1$
   are isomorphisms.
   The $k$-connected circles have that structure.

# Solution Process

Input: A regional support system $\mathcal{I}$

1. Execute the preprocessor.
2. If the instance has neither structure of Case 1 nor Case 2:
   a) The backtracking search counts all MVDs.
   b) The number of non-isomorphic MVDs are calculated using the isomorphism count.
3. If the instance has structure of Case 1 or Case 2,
   a) The backtracking search prunes the search tree as soon as the current labeling does not meet the standard representation.

# Computational Results

Programming language: Java 17

Desktop computer

- Intel(R) Core(TM) i9-9900 CPU @ 3.10GHzSoftware

Operating system

- Windows 10 Education

Original solver: V1

Updated solver: V2

# Results for k-connected Circles

| k | Order | Isomorphism count | Solver V1 Runtime (sec) | Preprocessor Runtime (sec) | Solver V2 Runtime (sec) | Runime V1 / Runtime V2 |
|---|---|---|---|---|---|---|
| 1 | 5 | 10 | 0.016 | < 0.001 | 0.015625 | 1 |
| 1 | 6 | 12 | 0.031 | 0.016 | < 0.001 | - |
| 1 | 7 | 14 | 0.422 | 0.016 | 0.047 | 9 |
| 1 | 8 | 16 | 8.750 | 0.0625 | 0.656 | 13.33 |
| 1 | 9 | 18 | 237.906 | 0.313 | 13.984 | 17.01 |
| 1 | 10 | 20 | 7248.797 | 3.078 | 386.906 | 18.74 |
| 1 | 11 | 22 | > 24 hrs | 38.031 | 12103.500 | - |
| 2 | 4 | 8 | 0.391 | < 0.001 | 0.078 | 5 |
| 2 | 5 | 10 | 112.672 | < 0.001 | 11.812 | 9.54 |
| 2 | 6 | 12 | 53662.063 | < 0.001 | 4612.047 | 11.64 |
| 3 | 4 | 8 | 3.172 | < 0.001 | 0.516 | 6.15 |

# Discussion

The MVD solver determines the number of non-isomorphic MVDs of any structure.

- At a size of around 22, runtimes exceed 24 hours

The preprocessor is $\Theta(n!)$ in the order $n$ of the MVD. The solution algorithm is still $O(r!)$ in the number $r$ regions of the support system.

- The bottleneck is not the preprocessor, but still the MVD counter.

In case of $k$-connected circles, the runtime reduction of solver V1 compared to V2 is in the order of the isomorphism factor.

- No surprise here

The problem of any algorithm that explicitly enumerates the MVDs is the huge number of existing MVDs. We cannot expect to solve instances with significant larger size $r$ without high performance computing.

# Thank you!