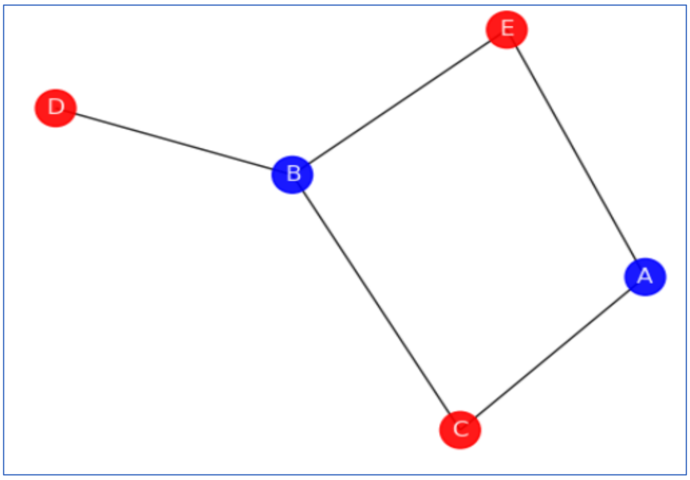
University of Essex

Department of Mathematical Sciences

1905876

MA838 – Capstone Project

# Opinion Forming - A network theory problem




Supervisor – Dr Joseph Bailey

April 15, 2022

# Table of Contents

# 1 Introduction

If we consider a group of individuals as a network, the ways in which opinions are spread and changed can be thought of as a graph colouring problem.

A social network, the graph of relationships among a group of individuals, plays a fundamental role as a medium for the spread of information, ideas, and influence among its members. [3]

Social media such as Facebook, Twitter, and Instagram have served as a crucial tool for communication and information disseminating in today's life. Therefore, studying different social behaviours like how people form their opinion regarding a new product or an election or how the information spreads through a social network have attracted a substantial amount of attention. Many different processes, from bootstrap percolation to rumour spreading, have been introduced to model this sort of social phenomena [7].

A network where each vertex represents a person, the colour of the vertex determines the opinion of that individual and the edge joining two vertices is a relationship between individuals [Figure 1]. It plays an important role in influencing the decisions people make and acts as a medium for the spread of information and ideas.

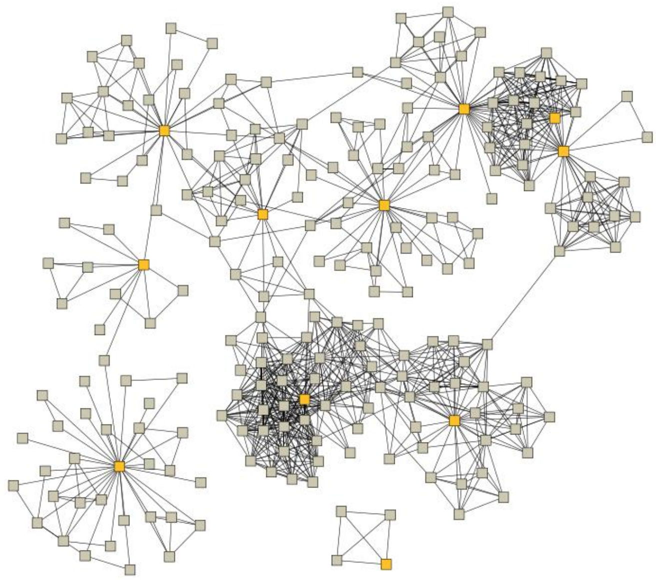


*Figure 1: An example of a social network, in this case of collaborative links. The nodes (squares) represent people and the edges (lines) social ties between them [27].*

Opinion spread can therefore be modelled by considering the influence of an individual's neighbours have upon the individual.

In the simplest case, we consider two opinions (colour vectors with two colours for practical simulations), and opinions are formed/changed based on a simple rule - at each timestep every vertex update, the colour of that vertex updates itself based on the majority of the neighbouring vertices, if at all the number of vertices of the opposing colour is the same, the vertex retains its colour.

It is simple to show how some setups are quite straightforward and settle into stable configurations whilst some settle into a cyclic loop of either distinct or a same colour vector. We discover that there is an increase in the number of steps it takes to finish one iteration as we reach higher vertices, there is also noticeable increase in the loop sizes

In this project we investigate discovering, classifying, and determining dynamics of graphs based on their initial configurations through an approach based on exhaustive simulations backed by theoretical results.

# 2 Implementation of Opinion Forming

The beginning of opinion forming was with people's exposure to relevant information and experiences, they then process the information and come to a judgement with a reasoning behind it. The different reasonings are then aggregated through either informal interactions or opinion polls [1].
The formation of opinions within a group of people has been a subject of interest in many areas [8], few of the most common ones are discussed below.

## 2.1 Sociology and Psychology

Since the study of online opinion formation involves both psychological behaviour and network dynamics, it has been a hot topic in sociology and nonlinear physics [9]. In general, one's opinion represents his attitude or standpoint towards a certain object, and opinion dynamics aims to reveal how social opinions evolve and converge by defining different interaction mechanisms from individual levels [10]. Although there have been many literatures on opinion model, most of them focus on nonlinear physics or statistical physics methods or simulate the opinion interaction merely using the principle of the minority being subordinate to the majority, which falls short on theoretically illustrating how the opinion interaction process is affect by multiple factors relevant to both parties [1].

In fact, the evolution of group opinion is a complex and holistic process, thus the characteristics and thoughts of everyone are necessary to be considered. For instance, some breaking news propagate rapidly on social networking sites and get widely discussed. The background players behind the diffusion are individual netizen who hierarchically forwarding the topics, and such spreading behaviour depends on the psychological attitude (e.g., support, opposition, etc.) of individual. It indicates that opinion formation is a fusion process of individual opinions, where a group of interacting agents continuously fuse their opinions on the same issue based on established rules [11]. Therefore, it is of great significance to consider more psychological factors to model opinion formation at the individual level.

Sociology and psychology theories are also important theory evidence to describe microcosmic individual interaction and macroscopic group behaviour in the process of opinion formation. Specifically, theories of behavioural psychology like stimulus–response theory [12] (Later improved as stimulus–object–response) may be employed to explain individual behaviours in opinion formation: if we regard opinions received as a stimulus, individual response is to decide whether to change his opinion. And the famous attitude change model proposed by Hovland is a theory basis to simulate individual mental state:

stick to the point of view or transform the attitude [13]. Moreover, human behaviour including mental activity is the result of the individual interaction, and is also influenced by group environment, which has been confirmed by the sociology research [1].

Distance plays a great deal in forming connections and is a fundamental element of establishing social links. Geo-social platforms are also highly correlated, and this could be further used to network. One of the great aspects of opinion forming when looking at it from the social point of view, we consider the identification of influential spreaders of the information and the impact of homophily. Homophily is the principle where the contact between similar people occurs faster than among dissimilar people [14].

Sociological and psychological features have also been introduced into opinion models, such as memory, inertia, noise, and conviction, characterizing the way in which these features change individual behaviour and the global dynamics in a specific scenario [15].

## 2.2 Economics and Finance

Within the standard Keynesian multiplier framework, extended by a micro-model of interactive formation of individual consumption propensities, it was demonstrated that socioeconomic interactions can lead to cyclical fluctuations in aggregate economic activity [16]. Each firm has fixed social relations to other firms and is either optimistic or pessimistic. If a firm is optimistic, it expects higher sales and consequently increases its production. A pessimistic firm, however, decreases its production since it fears a reduction in sales. A firm's opinion is influenced by two aspects. A firm tends to be optimistic (pessimistic) (i) if national income increases (decreases) and/or (ii) if more (fewer) firms it interacts with are optimistic. The mood of firms is dynamically updated. This formation model opinion is incorporated into a simple yet consistent macroeconomic framework and establish a robust bi-directional feedback process between firms' sentiments and national income. Accordingly, changes in firms' sentiments cause changes in national income, which, in turn, feedback on firms' sentiments. This phenomenon was observed for both a square lattice network and for a scale-free network [2].

## 2.3 Political

One of the big impacts of opinion forming in politics is during the elections and protests. One can easily model the dynamics of an election using data available on social media and then study the characteristics of distinct group of people who are like minded.

Given that Twitter is increasingly appropriated for both conversation and collaboration [11], and that tweet can be seen as an electronic word-of-mouth communication [13], it is likely that we can learn something about political sentiment by eavesdropping on these conversations. Typical uses of Twitter, including daily chatter, information sharing, reporting news and conversing [13], can all contain indicators of political opinion and sentiment [17].

In this light, the formation of public opinion is understood to be a process that revolves around individuals. It begins with their exposure to politically relevant experiences and information. Each individual processes this information, thereby coming to a judgment that yields an attitude. The attitudes of different individuals are then aggregated, either through informal interactions or more formal mechanisms, such as elections or opinion polls [18].

## 2.4 Public opinion - Media (POV)

Media provide people with cues as to what could ideally lead into formed opinions, but these are usually short lived. One example could be people evaluating the performance of a politician and/or their party based on the issues which are glorified by the media themselves. If the issue stays longer on media's agenda, people start to take sides, and form biases leading to opinions.

For example, in the evaluation of President George Bush in 1991, his overall job approval rating was high, corresponding with the victory in the Gulf War. However, in 1993 his approval rating was far lower than it was in 1991 because the Persian Gulf crisis was overshadowed by intense media coverage of economic recession. During this time, Bush's approval rating was more strongly linked to his performance concerning the economy than to his performance on foreign policy matters [19].

## 2.5 Social Influence

Social influence is the process by which individuals adapt their opinion, revise their beliefs, or change their behaviour because of social interactions with other people. In our strongly interconnected society, social influence plays a prominent role in many self-organized phenomena such as herding in cultural markets, the spread of ideas and innovations, and the amplification of fears during epidemics. Yet, the mechanisms of opinion formation remain poorly understood, and existing physics-based models lack systematic empirical validation [5].

Two controlled experiments we reported showing how participants answering factual questions revise their initial judgments after being exposed to the opinion and confidence level of others. Based on the observation of 59 experimental subjects exposed to peer-opinion for 15 different items, an influence map was drawn that described the strength of peer influence during interactions.

A straightforward process model derived from the observations demonstrates how opinions in a group of interacting people can converge or split over repeated interactions. Two major attractors of opinion were identified: (*i*) the *expert effect*, induced by the presence of a highly confident individual in the group, and (*ii*) the *majority effect*, caused by the presence of a critical mass of laypeople sharing similar opinions. Additional simulations reveal the existence of a tipping point at which one attractor will dominate over the other, driving collective opinion in each direction. These findings have implications for understanding the mechanisms of public opinion formation and managing conflicting situations in which self-confident and better-informed minorities challenge the views of a large, uninformed majority [5].

## 2.6 Agent based Models

When group members' actions follow a set of rules, their behaviour may be described by an agent-based model. Agent-based models may be employed to describe a variety of characteristics of the agents involved and the way they interact, allowing us to understand the evolution of the opinions of the individuals, and if and how they reach a final consensus or whether the agents polarize around a small number of different opinions [8].

Many opinion models are based on agent-based modelling as it is a successful method used in social dynamics.

Agent-based models may be employed to describe a variety of characteristics of the agents involved and the way they interact, allowing us to understand the evolution of the opinions of the individuals, and if and how they reach a final consensus or whether the agents polarize around a small number of different opinions.

Extant studies show that opinion formation in social network is of immense importance in various fields such as word of mouth marketing, political election, and social governance [4].

## 3 Introduction to graph theory

A graph is a mathematical structure containing two finite sets V and E and can be denoted by G = (V, E). V represents the vertices and E stands for the edges [20]. Graphs have order and size [Figure 2], the number of vertices in a graph is called the order of the graph and the number of edges is the size.

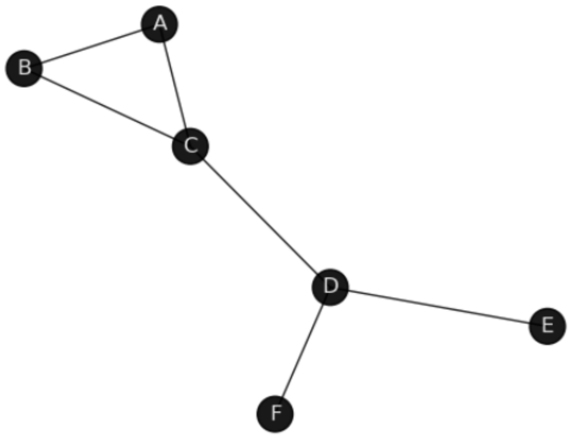



*Figure 2: Order of the graph - 6, size of the graph - 6*

The figure below can be used to show how to name the vertices and the edges of a graph.

The vertex and edge set are represented as $V_A$ = (A, B, E, D, C) and $E_A$= (AB, BE, ED, DC, CA, AD, BD) respectively [Figure 3].

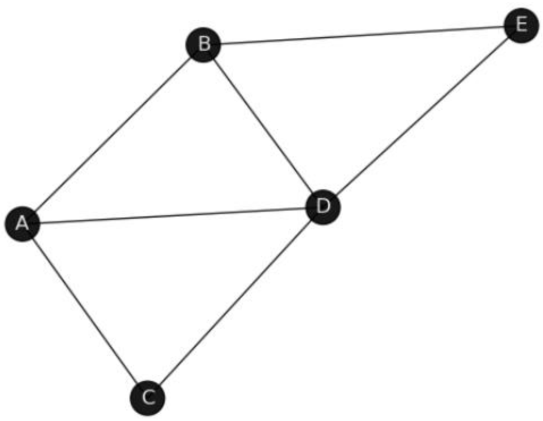



*Figure 3: vertex, edge set*

## 3.1 Diverse kinds of graphs

1. Fully connected Graphs – A graph which has a path(edge) between each pair of vertices is called a fully connected graph [21].

2. Disconnected Graphs – A graph where the vertices are split into 2 or more disjoint groups, such that one cannot link a vertex in one group to the vertex in another group by traversing along through the edges is called a disconnected graph [21].

3. Planar graphs – A graph which can be drawn on a plane such that no edges cross each other and only intersect at the vertices of that graph [21].

4. Circular Graphs – It is an undirected graph whose vertices can be associated with a finite system of chords of a circle such that two vertices are adjacent if and only if the corresponding chords cross each other [21] [Figure 4]. Its simplified version can be observed here [Figure 5].

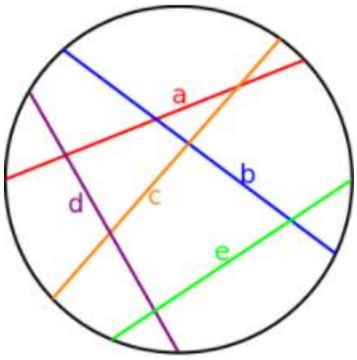



*Figure 4: finite system of chords in a circular graph [46]*

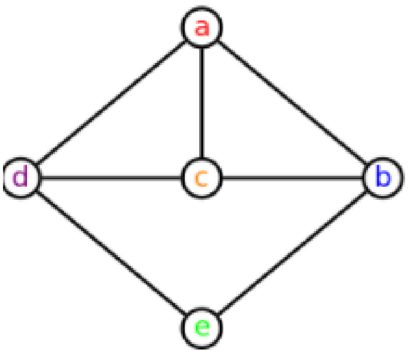



*Figure 5: simplified representation of the circular graph*

5. Bipartite Graphs – A graph G whose vertices can be divided into two disjoint sets M and N such that each edge of the original graph G connects a vertex of set M and set N, such a graph is called a bipartite graph [21] [Figure 6].

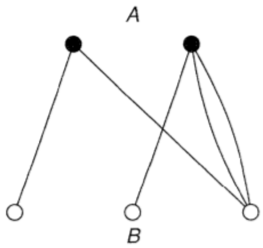



*Figure 6: Bipartite Graphs*

6. Loop less/Simple Graphs – When there is at most one edge joining two vertices, no edge may join a vertex to itself, and the edges are not directed, the graph formed is called a loop less graph or a simple graph [21].

7. Digraphs or Directed Graphs are graphs where the edges are directed between any given two vertices. The direction of the edge is denoted by the arrow on that very edge [21] [Figure 7].

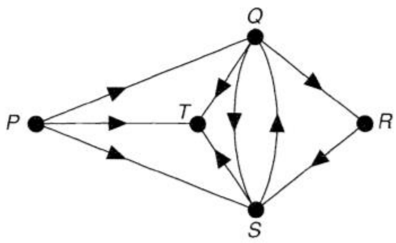



*Figure 7: Directed graphs*

## 3.2 Formulas

The maximum number of edges possible in a single graph with 'n' vertices is $nC2$ where

$$nC2 = n(n-1)/2 \text{ [22].}$$

The number of all possible graphs with 'n' vertices:

$$n = 2nC2 = 2n(n-1)/2 \text{ [23].}$$

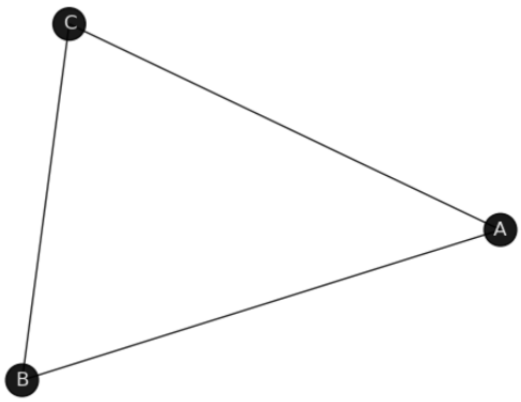



*Figure 8: Example for $n = 3$*

For example, consider the triangle above [Figure 8].

$$n = 3.$$

$$nC2 = 3$$

$$2nC2 = 23 = 8$$

The 8 graphs are as follows [Figure 9]:

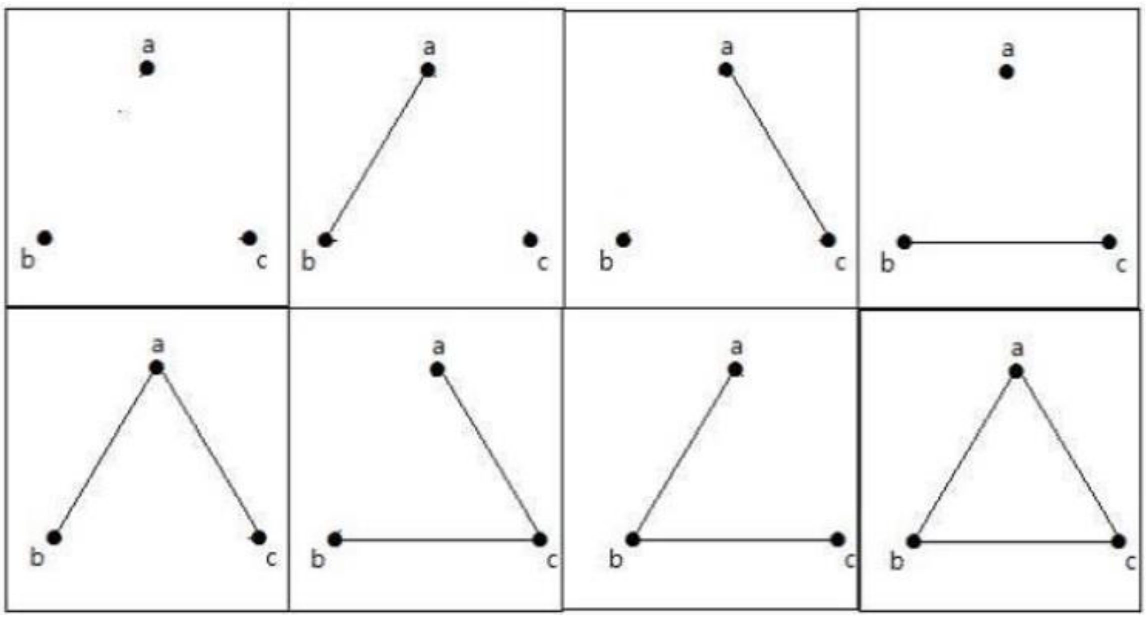



*Figure 9: All unique graphs for $n = 3$*

## 3.3 Centrality

Centrality is the measure of closeness of a node from the other nodes in a graph. It can be calculated using various measures, couple of the most important ones are node prominence and structural importance [24] [Figure 10]. Centrality indices answers the question "What characterizes an important vertex?" The answer is given in terms of the real-valued function on the vertices of the graph, where the values produced are expected to provide a ranking of the most important nodes [25].

The word ‘important’ here has a wide number of meanings, leading to many different definitions of centrality. It can be categorized by either the network flow or by walk structure which further disassociates into various categories.

The **Betweenness centrality** is the measure of a vertex within a graph. It is the number of times a node acts as a bridge along the shortest path between two other nodes. The betweenness centrality index is essential in the analysis of social networks, but costly to compute. Currently, the fastest known algorithms require (n) time and (n) space, where n is the number of actors in the network [26].

The **Closeness centrality** of a node is the average length of the shortest path between the node and all the other nodes in the graph [29].

The **Eigenvector centrality** is the measure of the influence a node has on the network [28].

The **Degree centrality** can be defined as the number of links incident upon a node, which is the number of ties the node has [30].

The **harmonic centrality** reverses the sum and reciprocal operations in the definition of closeness centrality [31].

The **Katz centrality** is a generalization of degree centrality. Degree centrality measures the number of direct neighbours, and Katz centrality measures the number of all nodes that can be connected through a path, while the contributions of distant nodes are penalized [28].

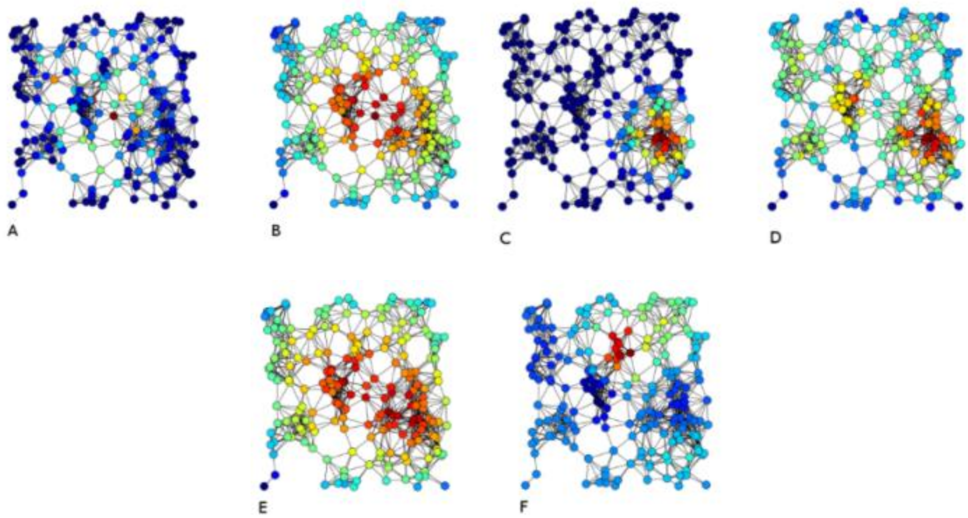



*Figure 10: Same network but different applications of centrality indices [32].*

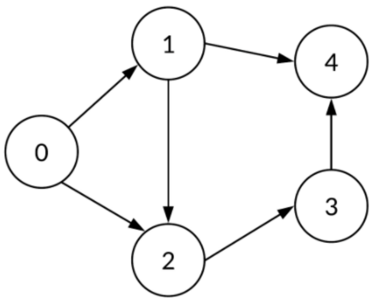
There are a couple of limitations to the centrality indices, one of them being obvious wherein one application of centrality is often sub-optimal for a different application. The more subtle limitation is the commonly held fallacy that vertex centrality indicates the relative importance of vertices [32]. Centrality indices are explicitly designed to produce a ranking which allows indication of the most important vertices. This they do well, under the limitation just noted. They are not designed to measure the influence of nodes in general.

## 3.4 Adjacency Matrix

An adjacency matrix is a square n X n matrix that is used to represent a finite graph by storing the nodes labelled as one if they are adjacent and 0 if they are not.

In case of a simple graph, as in the figure below [Figure 11]. The diagonal will always be made up of zeroes since edges from a vertex to itself (loops) are not allowed in simple graphs.

The adjacency matrix of an undirected simple graph is symmetric, and therefore has a complete set of real eigenvalues and an orthogonal eigenvector basis The set of eigenvalues of a graph is the spectrum of the graph.




Adjacency Matrix

| | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 | 1 |
| 2 | 0 | 0 | 0 | 1 | 0 |
| 3 | 0 | 0 | 0 | 0 | 1 |
| 4 | 0 | 0 | 0 | 0 | 0 |

*Figure 11: Adjacency matrix*

The representation of the matrix is different for undirected and undirected graphs, see [Figure 12].

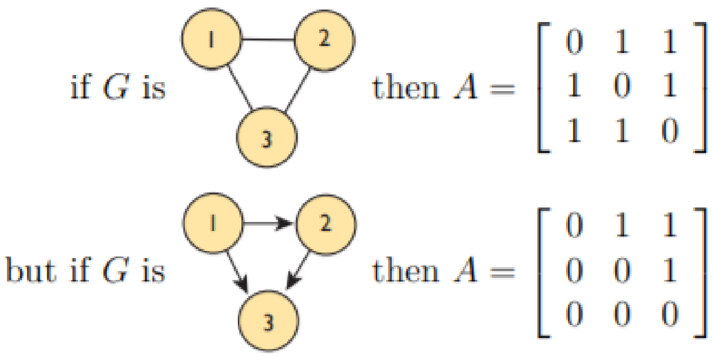



*Figure 12: undirected vs directed adjacency matrix comparision*

## 3.5 Isomorphism

Two graphs are said to be isomorphic if there exists a one-to-one correspondence between their vertices. For graphs $G_1$ and $G_2$, if the number of edges joining any two vertices of $G_1$ is equal to the number of edges joining the corresponding vertices of $G_2$, they are said to be isomorphic graphs [Figure 13].

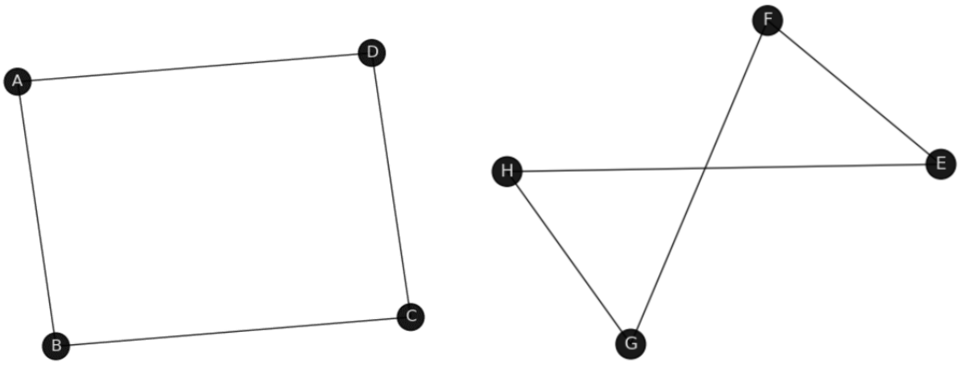



*Figure 13: Isomorphic graphs*

In the diagram above, A has two edges, and its corresponding vertex E has two edges as well. Similarly, B<->F, C<->G and D<->H. All have the same number of edges to their corresponding vertex, making this and isomorphic pair of graphs.

## 3.6 Euler's Analysis of Seven Bridges of Königsberg

The foundation of graph theory started when Leonhard Euler laid negative resolution to '*Seven Bridges of Königsberg*' [Figure 14]. This is an infamous mathematical problem where two mainlands in the city of *Königsberg* in Prussia (now Russia) were set on both sides of Pregel river. These two mainlands were connected by seven bridges.

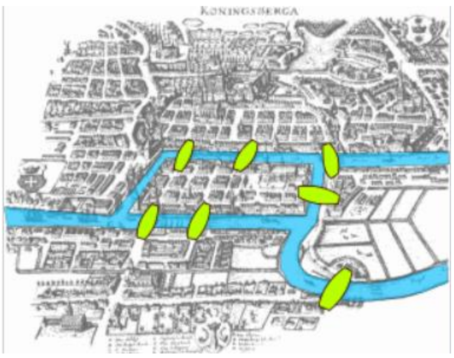


*Figure 14: Seven bridges problem [45]*

The problem was to devise a way wherein one would cross each bridge one and only once. Euler proved that this problem has no solution. How he did this is by first pointing out that choice of route inside either landmass are irrelevant as the only important feature is the sequence in which the bridges are crossed. This enabled him to rephrase the problem in a more abstract way where the landmasses were nodes, and the bridges were the edges. It does not really matter whether the edges are curved or straight. The resulting structure is a graph [Figure 15].

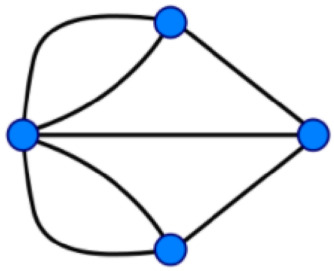


*Figure 15: Euler's abstract representation of seven bridges problem*

## 3.7 Graph Colouring

Graph colouring was first introduced when in 1852, when Francis Guthrie postulated the four-colour conjecture, observed that four colours were sufficient to colour the map of any region such that no region sharing the same border (adjacent borders) have the same colour. Francis' brother approached his teacher (Augustus De Morgan) at the university college, who later wrote to William Hamilton in 1852.

This transformed into the problem of deciding whether it is possible to colour the vertices of every planar graph with four colours such that no two adjacent vertices are assigned the same colour.

**Vertex colouring** [Figure 16] arises more commonly than edge colouring or map colouring. It can be defined as the assignment where $f:V_G\text{->}C$ from its vertex-set onto a k element set C whose elements are called colours (C = 1,2,3…k). For any k, such an assignment is called vertex-colouring.

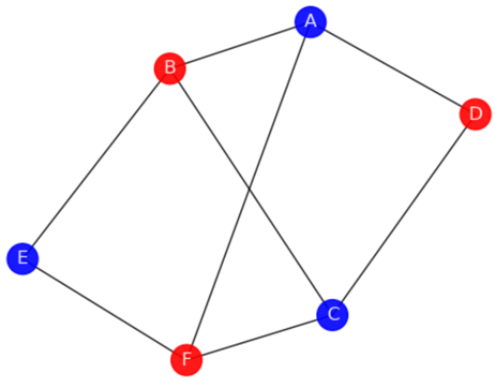



*Figure 16: Vertex-colouring*

A **colour class** in a vertex-colouring of a graph G is a subset of $V_G$ containing all the vertices of a given colour.

A **proper vertex-colouring** is vertex-colouring of the graph is such that the endpoints of each edge are appointed distinct colours [Figure 17].

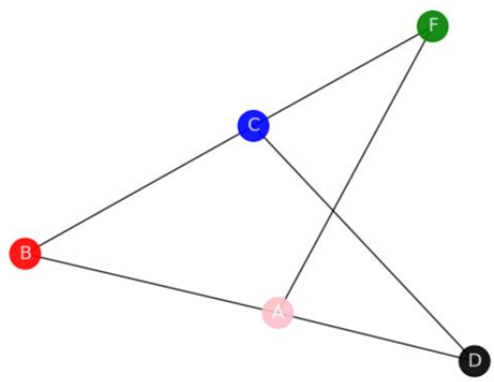



*Figure 17: proper vertex colouring*

The **Chromatic number** of a graph is G is denoted by $_x$(G), is the minimum number of distinct colours required for a proper vertex-colouring of G. A graph G is k-chromatic if $_x$(G) = k.

## 3.8 Popular lemmas in Graph Theory

**Regularity lemma** states that every dense graph can be partitioned into a small number of regular pairs and a few leftover edges. Since regular pairs behave as random bipartite graphs in many ways, the Regularity Lemma provides us with an approximation of an arbitrary dense graph with the union of a constant number of random-looking bipartite graphs.

**Handshaking lemma** states that in every finite undirected graph, the number of vertices that touch an odd number of edges is even [Figure 17].

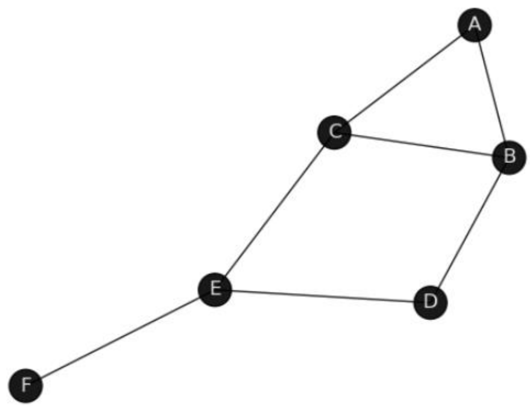



*Figure 18: an example of handshaking lemma*

From the figure above, the vertices E, C, B and F are an even number of vertices (four) that are connected to an odd number of edges.

The sum of the degrees of their edges is 2 + 3 + 2 + 3 + 3 + 1 = 14. This is known as the **degree sum formula**. It states that the sum of the degrees of the vertices in a graph is twice of the number of edges present in the graph.

**Graph removal lemma** states that when a graph contains a few copies of a given subgraph, then all the copies can be eliminated by removing a small number of edges.

# 4 Introduction to network analysis

The study of graphs can be divided into two sections based on their symmetry namely as Symmetric and Asymmetric relations.

Symmetric relations are the binary relations where if a=b true then b=a is also true, where a, b belong to the set X. Set X here can be called symmetric. Asymmetric relations can be better explained with the help of digraphs, Digraphs are graphs made of a set of vertices connected by directed graphs.

Network analysis has many applications in statistics, particle physics, electrical engineering, economics, etc.

## 4.1 Application of Graph Theory in puzzles

1. 4 cube problem
2. Ramsey theory
3. Six people at a party problem
4. The eight circles problem

## 4.2 Bootstrap Percolation

In statistical mechanics, bootstrap percolation is a percolation process in which a random initial configuration of active cells is selected from a lattice or other space, and then cells with few active neighbours are successively removed from the active set until the system stabilizes. The order in which this removal occurs makes no difference to the final stable state. Bootstrap percolation can be interpreted as a cellular automaton, resembling Conway's Game of Life, in which live cells die when they have too few live neighbours. However, unlike Conway's Life, cells that have become dead never become alive again [2].

## 4.3 Majority Bootstrap Percolation

In majority bootstrap percolation on a graph G, an infection spreads according to the following deterministic rule: if at least half of the neighbours of a vertex v are already infected, then v is also infected, and infected vertices remain infected forever. Percolation occurs if eventually every vertex is infected. [3]

## 4.4 Proof for Handshake theorem.

Let V1 be the vertices of even degree and V2 be the vertices of odd degree in an undirected graph G = (V, E) with m edges. Since each edge contributes twice to the total degree count of all vertices [33]. Then

$$\rightarrow 2m = \sum_{v \in V} \deg(v) = \sum_{v \in V_1} deg\,(v) + \sum_{v \in V_2} deg\,(v)$$

or

$$2m = \sum_{v \in V} \deg(v)$$

### 4.5 Proof for max number of edges theorem.

We know that the degree of each vertex (v) in a simple graph (G) is 1 less than the number of vertices(n) [34].

$$deg(v) \ = \ n - 1$$

For n=2,

$$deg(v) \ = \ n - 1 \ = \ 2 - 1 \ = \ 1$$

Since the sum of the degrees is even for a simple undirected graph, we can denote it by

$$\sum deg(v) = 2m \text{ [Proof for Handshake theorem]}$$

For n=2,

$$\sum deg(v) = n - 1$$

For n vertices the total degree is $n(n-1)$,

$$\rightarrow 2m \ = \ n(n-1)$$

$$m \ = \ n(n-1)/2$$

where m denotes the edges of the graph.

### 4.6 Degree for the bipartite graph

For a vertex, the number of adjacent vertices is called the degree of the vertex and is denoted $deg(v)$. The degree sum formula for a bipartite graph state that

$$\sum \deg(v) = \sum \deg(u) = |E|$$

# 5 Modelling opinion forming

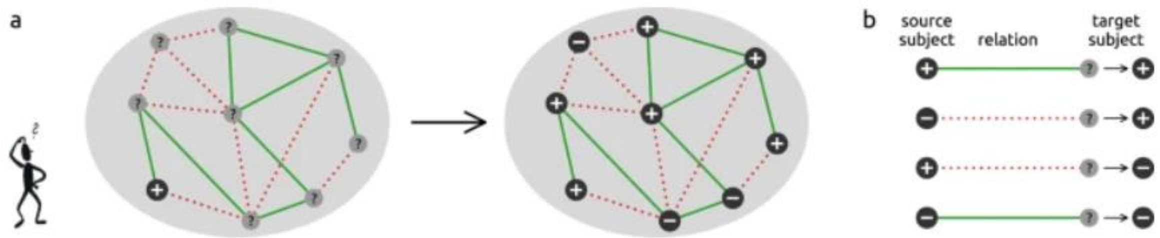



*Figure 19: Opinion formiation model*

An observer interconnected with a set of subjects by mutual relations of trust (Solid green line) or distrust (dotted red line). Starting from a small set of opinions (in this case just one) marked with a + in black and a connection of unknown subjects (grey question marks), the observer gradually forms opinions on all subjects.

The formed opinion is determined as a product of the opinion and the sign of the relation between the source subject and the target subject. A positive opinion is formed when the source opinion and the relation are either both positive or both negative; a negative opinion is formed otherwise [Figure 19].

Assumption – We assume, in a Graph g, the vertex does not change its colour if presented in situation with equal number of distinct colours, i. e., it retains its colour.

The most basic model for our problem statement can be thought about in a similar way where a graph (g1) represents a network of people with a pre-existing colour scheme denoted by a colour vector (c1). The introduction of a new colour vector changes the dynamic of the network, in a sense where when the colour vector is multiplied with the graph (g1), it results in another graph (g2) with a new colour scheme of its own denoted by a new colour vector (c2). The colour of the resulting graph's vertices are decided based on the initial colour vector. This can be seen here [Figure 20] [Figure 21] [Table 1] This process is repeated until either the graph comes to a halt or starts looping over previously seen colouring schemes.

Modelling the graphs and colour vectors can be easily done using a programming language. We have used python to model our problem statement. We implement the graph and the colour vector with the help of the data structure like adjacency matrix and a list respectively.

Colour scheme: Blue = 1 and Red = -1.

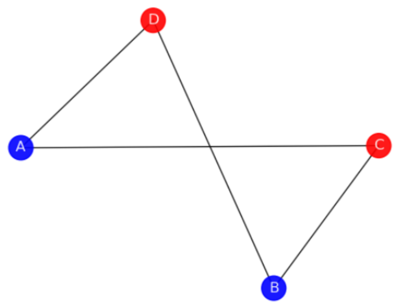



*Figure 20: graph g1 with colour scheme [1, 1, -1, -1]*

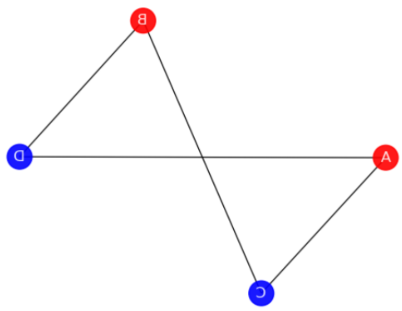



*Figure 21: graph g2 with colour scheme [-1, -1, 1, 1]*

Once we set an example for four vertices, we came up with the list of all the possible graphs and colour vectors using clever functions explained later here [code].

The model is primarily based on 3 for loops [36] followed by an if condition check [38] and ultimately a while do loop [37].

## 5.1 Cyclic loops explained

### 5.1.1 Case 1

We start with a simple graph G made up of four vertices A, B, C, and D and the edge set AB, BC, CD, and DA. Each vertex represents an individual and is assigned a colour to show either a similarity or difference in opinion. We use the colours red and blue to differentiate here.

We start off with A and C being red and B and D being blue. This is a special case wherein the graph would change but remain the same in terms of isomorphism. Let us take vertex A to understand this. Vertex A is of the opinion 'X' on some issue M, and so is vertex C. But vertex B and D are of the opinion 'Y' on the same issue. A is connected to both B and D and hence has A changes its opinion from 'X' to 'Y' since for A both B and D go with opinion Y. Similarly, B, C, and D all change their opinion only to land up in an identical situation. This ends up being a cyclic stage of step time = 2, where graph changes its colour every step of the iteration, ultimately leading into an infinite loop.

In a similar fashion the method explained above, the graphs, when ran for vertices ranging from 2 till 6 portray distinct cyclic loops with the maximum length of cycle = 2. The program was also implemented on vertices 7,8,9 and 10 but with limited graphs due to constraints on the memory and exponential increase in the number of iterations, it did not show much variation in the cycle. But it could be hypothesised that, as the number of vertices increase even further, and computations are done on a more powerful machine, it would eventually lead to a cycle of three or even more.

### 5.1.2 Case 2

Consider the same graph in case 1 except this time, the vertex pair with same opinions are A, B as opposed to C, D. In this case, no vertices would never change their opinion based on the assumption. Hence the graph still ends up in a loop, but the unique property of the graph would be that the state of its colour scheme never changes.

# 6 Code Explanation

The code for running the model and analysing the results has been written in python and Jupyter notebooks. All the code files are present on this link in a maintained GitHub repository [39] and submitted in a zip file along with this report.

Using the code in graph.py, we run simulation for various number of graphs and colour vectors.

Initially we import all the necessary libraries for the project including the infamous ones like NumPy, Pandas and Time. We also import the network library to deal with matrices and graph related components.

Next, we declare all the variables and data structures, followed by a custom list containing all the features for graphs in the format of a dictionary, this reduces complexity when dealing with the key-value pairs during multiple iterations.

We have also implemented the code in the conventional format where it follows object-oriented programming, s. Object oriented programming might be preferred to abstract data types, data encapsulation, information hiding, and modularization [6]. This makes it easier to maintain and modify the code.

We also use two clever functions to generate and manipulate the list of all the possible graphs and all the possible colour vectors for that number of vertices, the interesting part about this function is that while generating the colour vectors for the user input, it halves the list, in a way where all the cases are considered, and we end up saving computational and memory resources. Let us take an example to understand why this is feasible:

For a graph denoted by the adjacency matrix

[[0,1,1],

[1,0,1],

[1,1,0]]

It will produce a similar result when multiplied to either the colour vector [1, -1, 1] or the colour vector [-1, 1, -1] i.e., the next colour vector for both the iterations would be a vector consisting of all the vertices of the same colour denoted by either [1, 1, 1] or [-1, -1, -1]. This indicates that the graphs get infinitely stuck in a loop for the same colour. To avoid redundancy, we eliminate half the colour vector list by inferring results about all the colour vectors.

Since the time for which the program runs increases exponentially after crossing n=5, we also implement a call-back function using a simple if-else check while generating the graph list for similar higher number of vertices [see appendix 1]

We declare various other functions which I will explain briefly, to begin with, we have the function get_zero_matrix [see appendix 2]. Since the list of graphs is generated in a n bit binary string filled with 1s and 0s where n is equal to the number of vertices, we manually assign it to a matrix form for performing operations on it later. Then we have another function named check_connected which does exactly as it name suggests, from the list of all graphs, it checks whether the next graph in the list is connects, if the Boolean returns true, the program proceeds and calculates all the variations against the entire list of colour vectors before it moves on to the next graph in the graph list.

While dealing with the result of the product of the matrix and the colour vector, we encounter 0s in the resulting colour vector which is meant to be used for the next iteration, The way we have modelled the problem, it assumes that in case where a person is exposed to equal number of opposing opinions, their opinion does not change, and they keep their original opinion. When we apply this to our model, it implies that when the result of the product of the matrix and the colour vector sums to give 0, we assign that vertex the colour it previously had.

To put this explanation into practice, we implement a function called check_for_zeroes. This function is responsible for checking each individual indices of the next colour vector, if the sum of the resulting colour vector's index is equal to 0, the function assigns that index the previous colour vector's index's value.

For example, if the vector for iteration 1 had the value [1, 1, -1, -1, 1] and the next vector for iteration 2, post matrix multiplication turns out to be [1, 0, 1, 1, 1], it is converted to [1, 1, 1, 1, 1] in accordance with our assumption.

Next, we have two simple functions responsible for calculating the length of step-time and cycle over which the iterations loop and appending all the results to the table which later get saved as a csv file. These functions are called calculate_length_of_cycle and append_to_table.

Once all the functions are declared, we take input from the user for the number of vertices and run the program on that very input resulting in a data frame finally converted to a csv file, which consists of various columns [see appendix 3].

We have also used the file visualisation.py quite a few times to plot most of the graph images used in this report. Further we also implement the code in the file plotting-graphs.ipynb to plot all possible and unique graphs [see Appendix 4] [ see Appendix 5] [ see Appendix 6] [ see Appendix 7] [ see Appendix 8] [ see Appendix 9] for each number of vertices.

To get the csv files, simple run the program and enter the value for which you want the dataset. Then pass that file through to the pandas data frame in the analysis.ipynb file. Once the data set is loaded, simply run the file to obtain the plots and desired metrics.

# 7 Data Visualisation and Analysis

The file graph.py is ran for assorted vertices ranging from 2 till 10 which produces a csv file for that very vertex. We, then create a custom csv file containing the data for vertices ranging from 2 till 6 and analyse it using a piece of analytic code in analysis.ipynb. This is our preliminary analysis of the entire dataset. We try to find patterns and trends here and establish a basic understanding of the variation in properties of the graphs like step time and cycle of loop across all the vertices [Figure 22] [Figure 23] [Figure 24].

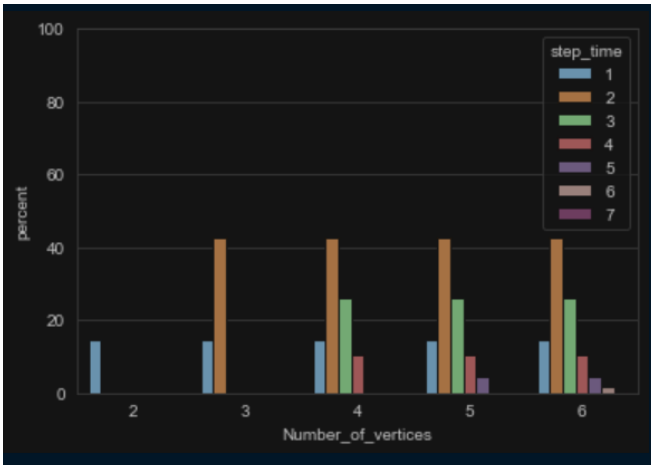



*Figure 22: bar plot (x = number of vertices, y = % step time, hue = step time)*

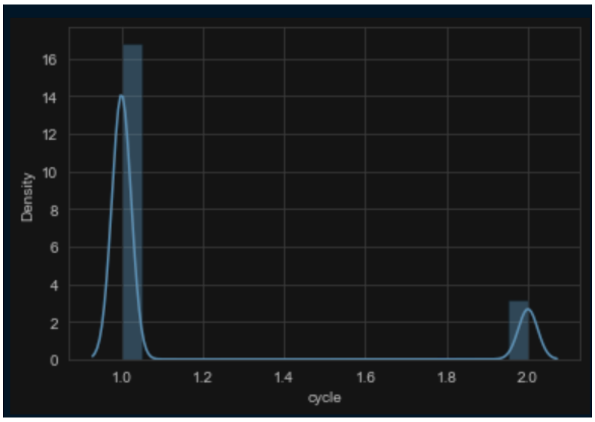



*Figure 23: density plot (x = cycle, y = density)*

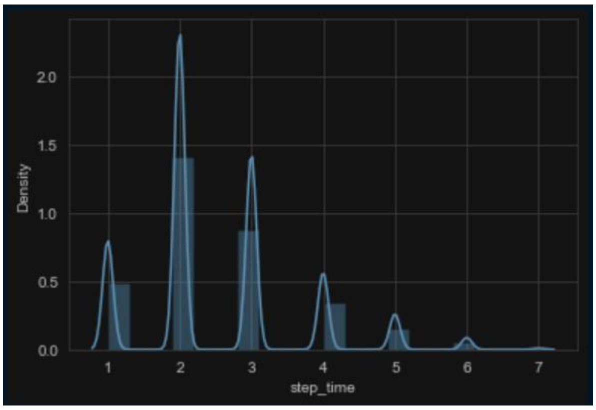



*Figure 24: density plot (x = step time, y = density)*

Moving on further we use another piece of code in finer-analysis.ipynb to analyse the dataset in depth, where we load the csv files for individual vertices and visualise them separately. We then, conclude our analysis by using some statistical measures and noting the difference across the board.

Initially, we begin with loading and cleaning the dataset. Then, we start converting the value counts to percent to obtain percentage columns for each variable which could be further plotted against the number of vertices and compared based on metrics like skewness, kurtosis, mean and variance.

We make use of python's seaborn library to come up with these bar plots [appendix bar plot].

# 8 Interesting insights

From having a look at the bar plots [bar plots in appendix], we can comment on some insights. We can observe the most common vector is [1, 1, 1] for when the number of vertices = 3, with over 60% across all the colour vectors.

The cycle of value 1 occurs more often than cycle of 2, leading almost by 80%, indicating that most of the graphs settle down in a stable state quite rapidly. Also, another factor influencing this is the skewness.

Moving from number of vertices = 2 to number of vertices = 8, We also notice that the skewness of the data fluctuates between positive and negative showing variance in the dataset.

# 9 Conclusion

After considering the code driven approach to model and simulate results for the given problem statement, we can conclude that there all configurations either settle into a stable state, start looping after a certain number of iterations or end up in a cyclic loop.

We also see that the increase in the number of configurations is exponential as we increase the number of vertices, which is an expected outcome as the configurations depend on the graph list and colour vector list which in turn are increasing exponentially. With this in mind, we can predict an increase in the step time and the loops, implying much more complex patterns in opinion distribution.

I would have further added to this project by expanding on the colour vectors from only two colours to more, indicating that the opinions are not just contrasting but there are multiple points of view which resonates more to a real-life situation.

Another approach could be where an individual's opinion weighs more than another individual. A real-life example from the industry of sports, wherein the infamous athlete Cristiano Ronaldo replaces the bottles of Coca-Cola with water, resulting in a whopping $4 billion dollar loss in the market value of the American drink giant [35]. To apply this to the model, we could add weights to specific variables which would alter the entire process of opinion formation.

Furthermore, this project could also be looked at from the point of view of probabilistic approach where we calculate conditional probability based on the value of various kinds of graph components like clique, and centralities and consider them while making operations to avoid redundant calculations and much more optimal results.

Lastly, as we move up to higher vertices, we can implement a check for isomorphic graphs eliminating similar cases and optimizing the time complexity on top of that. Besides that, another check could be implemented for every graph where the iterations are checked and stored while performing, in case of repetition of a similar iteration, they could be then overlooked since we would already know how they would end up.

The most time was spent on understanding, researching, modelling, and coding the problem. Writing efficient functions to cut down on computational time was a real challenge, however I did enjoy working on this project.

# 10 Glossary

Lemma - A subsidiary or intermediate theorem in an argument or proof.

Arc – An arc is a directed line

Edge – A line joining a pair of vertices

Loop – Edge/Arc that joins a vertex to itself

Adjacent – Next to or adjoining something else.

Walk – Series of vertices and edges.

Path – A walk where no repeated vertices.

Clique – In an undirected graph G = (V, E), it is a subset of vertices C ⊆ V, such that every two distinct vertices are adjacent.

Width – In an undirected graph in which every subgraph has a vertex of degree at most k, which is some vertex in the subgraph touches k or fewer of the subgraph's edges. The width of the graph is the smallest value of k for which it is k-degenerate. It measures how sparse the graph is.

Size – The size of the graph is the number of edges in the graph.

Order – The number of vertices in a graph is the order of the graph.

Cycle – an interval of time during which a sequence of a recurring succession of events or phenomena is completed

Step time – The number of steps it takes for the iteration to either come to a halt or start looping.

Colour Vector – Made up of 1s and -1s, this vector indicating two distinct colours, when multiplied with the matrix representing the graph, gives a new colour vector in succession.

Matrix – a set of numbers arranged in rows and columns to represent a graph.

# 11 Appendix

Link to the code - [https://github.com/Sanskar-16/capstone](https://github.com/Sanskar-16/capstone)

```python
def get_graph_int_list(rep):
    counter = 0
    for graph in product([1, 0], repeat=rep):
        graph_int_list.append(list(graph))
        counter = counter + 1
        # additional counter for higher number of vertices
        if counter == 10000:
            break
    print("There are {} graphs in total for {} vertices".format(counter, n))
    print(graph_int_list)
    print('\n')

    return graph_int_list
```

*Appendix 1*

```python
# function that creates the required zero matrix on user input
def get_zero_matrix(x):
    global zero
    zero = [[0] * x for _ in range(x)]
    print("The zero matrix for {} vertices is {}".format(x, zero))
```

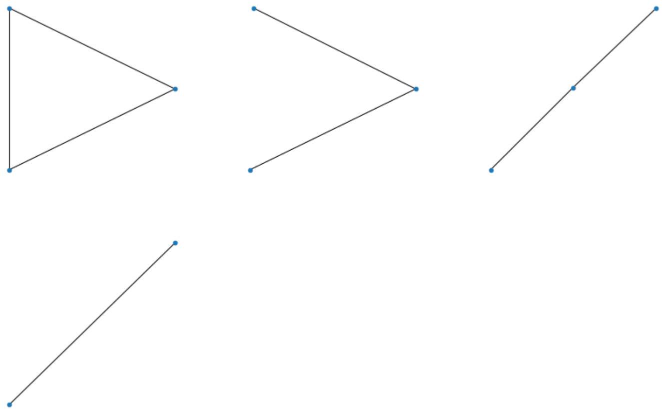
*Appendix 2*

| | Graph_number | Number_of_vertices | starting_colour_vertex | ending_colour_vertex | step_time | cycle |
|---|---|---|---|---|---|---|
| **0** | 1 | 2 | [1, 1] | [1, 1] | 1 | 1 |
| **1** | 1 | 2 | [1, -1] | [1, -1] | 1 | 2 |
| **2** | 1 | 3 | [1, 1, 1] | [1, 1, 1] | 1 | 1 |
| **3** | 1 | 3 | [1, 1, -1] | [1, 1, 1] | 2 | 1 |
| **4** | 1 | 3 | [1, -1, 1] | [1, 1, 1] | 2 | 1 |

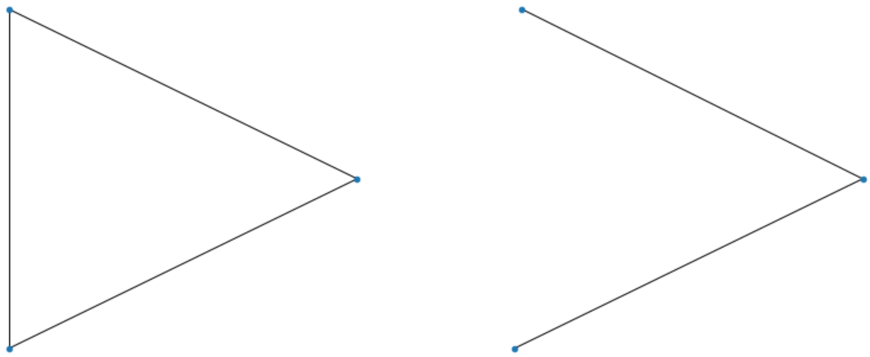
*Appendix 3*

| Graph (adjacency matrix) | Colour Vector |
|---|---|
| g1 = [[0, 0, 1, 1], [0, 0, 1, 1], [1, 1, 0, 0], [1, 1, 0, 0]] | c1 = [1, 1, -1, -1] |
| g2 = [[0, 0, 1, 1], [0, 0, 1, 1], [1, 1, 0, 0], [1, 1, 0, 0]] | c2 = [-1, -1, 1, 1] |

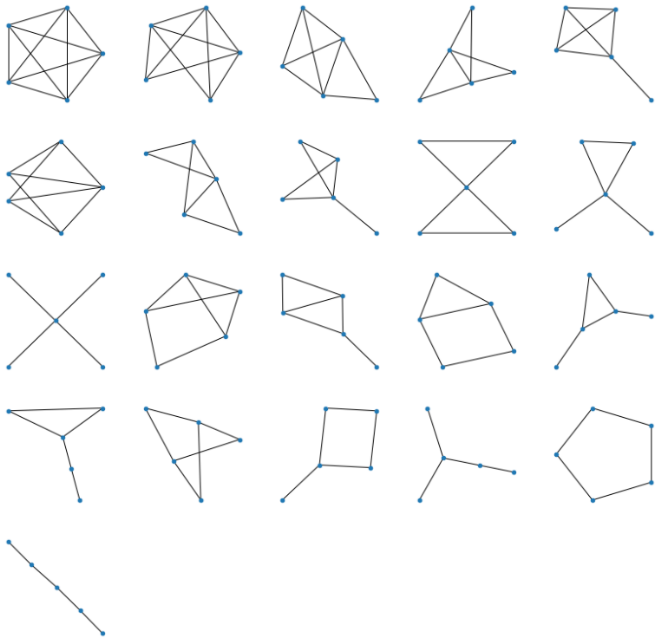
*Table 1: Graph and vertex colouring*



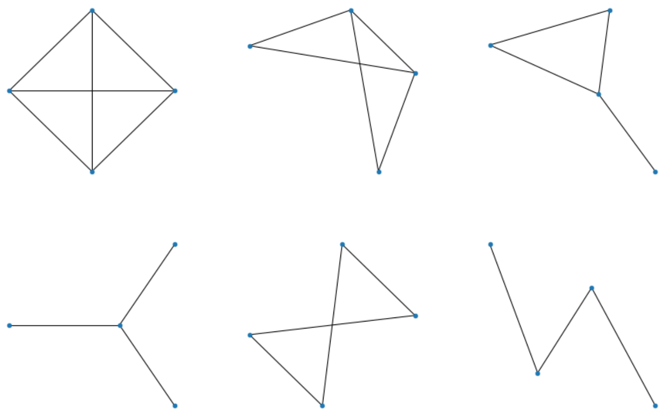
*Appendix 4: All graphs with three vertices*



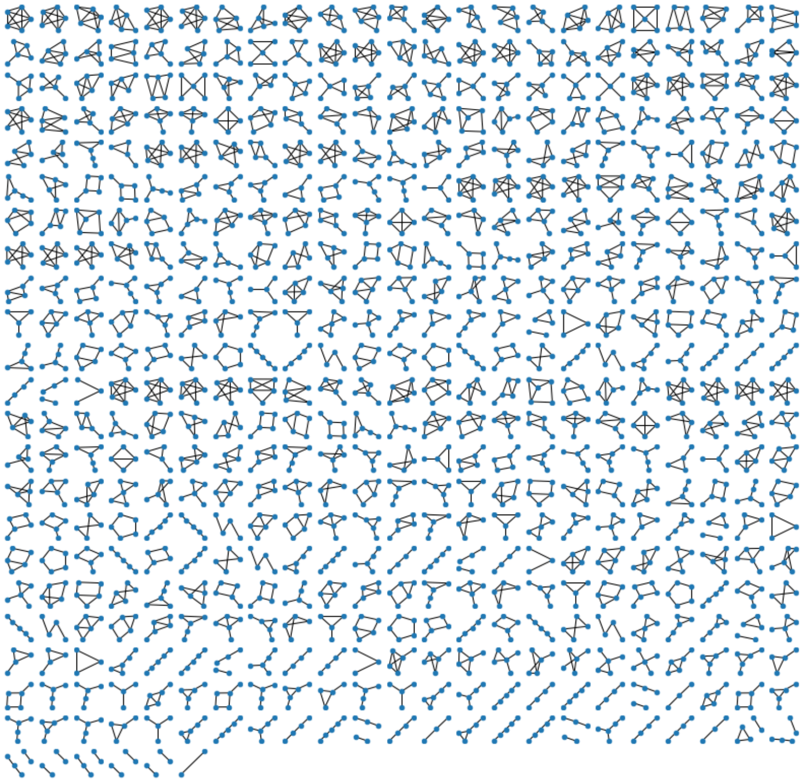
*Appendix 5: Unique graphs with three vertices*



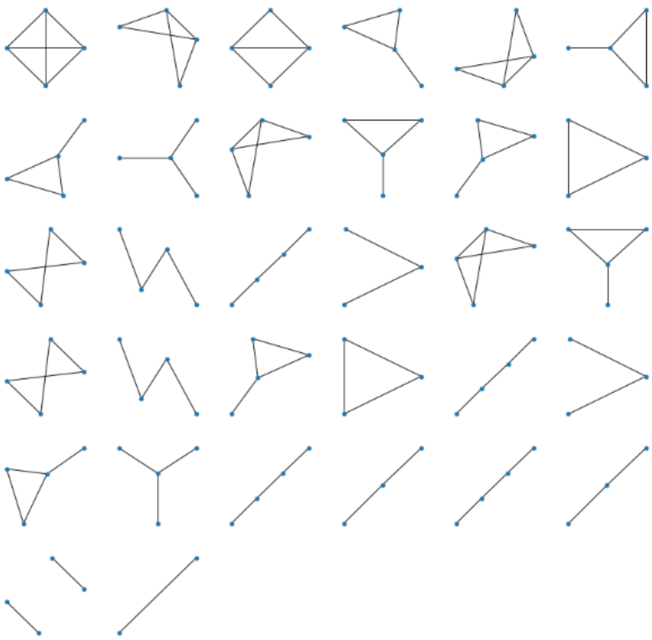
*Appendix 6: All graphs with four vertices*



*Appendix 7: Unique graphs with four vertices*



*Appendix 8: All graphs with five*



*Appendix 9: Unique graphs with five vertices*

## 11.1 Bar plots

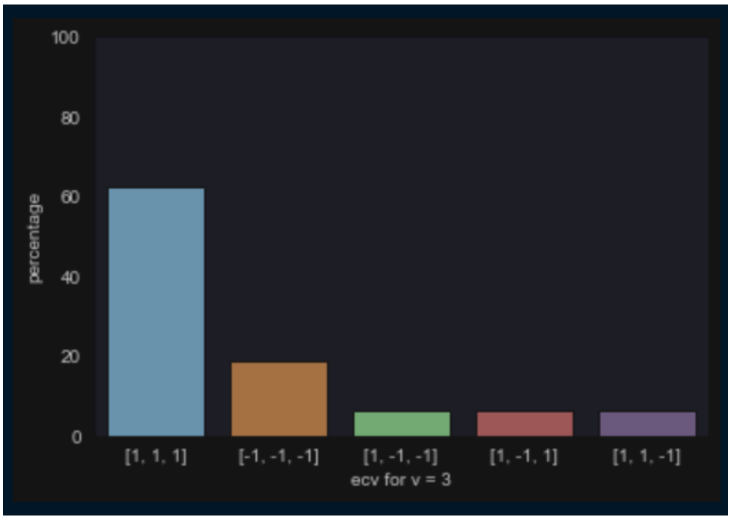



*Figure 25: ending colour vector vs % ending colour vector*

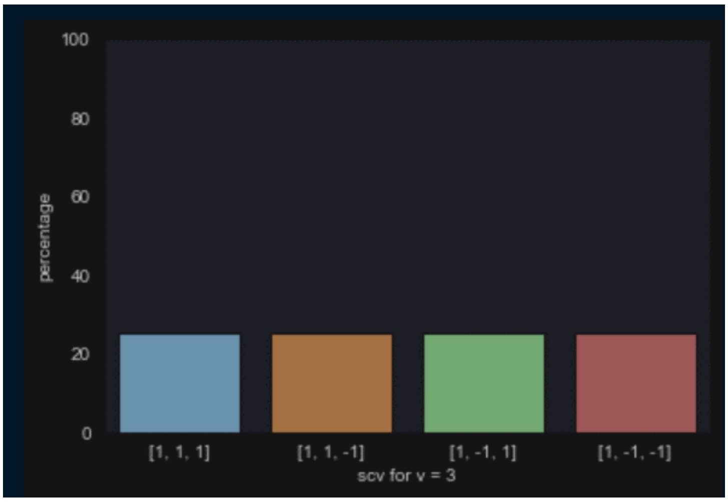



*Figure 26: starting colour vector vs % starting colour vector*

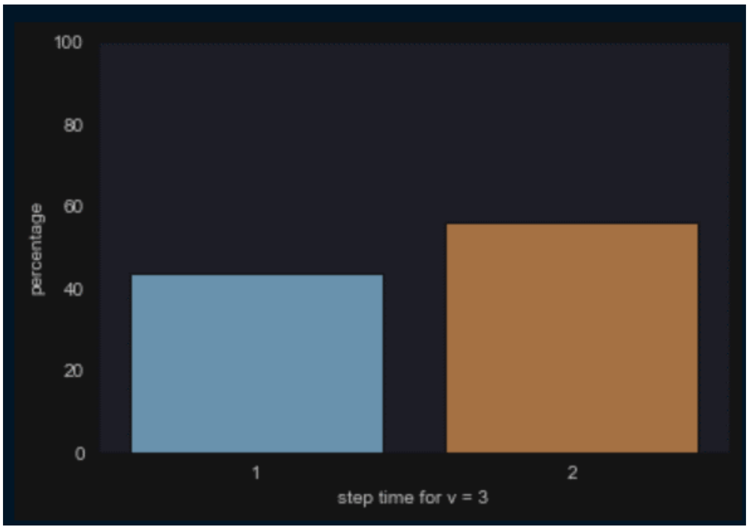



*Figure 27: step time vs % step time*

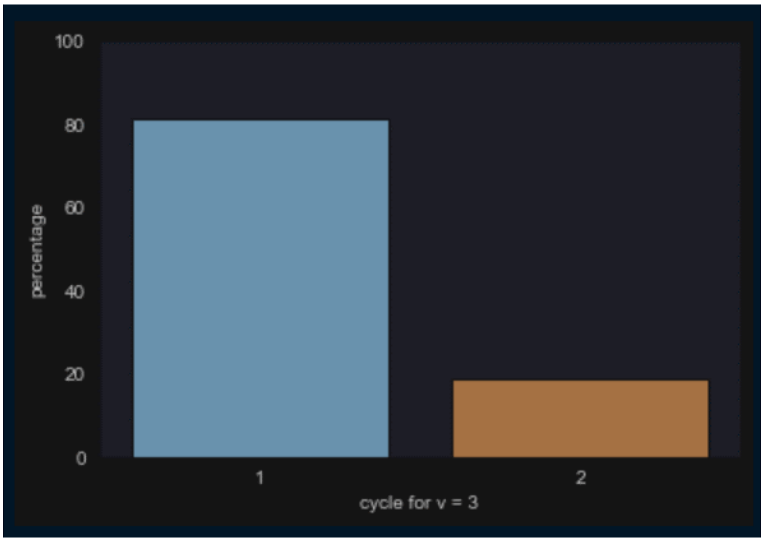



*Figure 28: cycle vs % cycle*

# 12 Acknowledgement

I am grateful to Dr joseph Bailey for their help in preparing the manuscript and providing great supervision and support throughout the project.

# 13 Bibliography

[1] Yin, X., Wang, H., Yin, P., & Zhu, H. (2019). Agent-based opinion formation modelling in social network: A perspective of social psychology. *Physica A: Statistical Mechanics and its Applications*, *532*, 121786.

[2] Westerhoff, F. (2010). An agent-based macroeconomic model with interacting firms, socio-economic opinion formation and optimistic/pessimistic sales expectations. *New Journal of Physics*, *12*(7), 075035.

[3] Zehmakan, A. N. (2020). Opinion forming in Erdős–Rényi random graph and expanders. *Discrete Applied Mathematics*, *277*, 280-290.

[4] Yin, X., Wang, H., Yin, P., & Zhu, H. (2019). Agent-based opinion formation modelling in social network: A perspective of social psychology. *Physica A: Statistical Mechanics and its Applications*, *532*, 121786.

[5] Moussaïd, M., Kämmer, J. E., Analytis, P. P., & Neth, H. (2013). Social influence and the collective dynamics of opinion formation. *PloS one*, *8*(11), e78433.

[6] Rentsch, T. (1982). Object oriented programming. *ACM Sigplan Notices*, *17*(9), 51-57.

[7] https://www.frontiersin.org/articles/10.3389/frai.2020.00045/full

[8] Mastroeni, L., Vellucci, P., & Naldi, M. (2019). Agent-based models for opinion formation: A bibliographic survey. *IEEE Access*, *7*, 58836-58848.

[9] Castellano, C., Fortunato, S., & Loreto, V. (2009). Statistical physics of social dynamics. *Reviews of modern physics*, *81*(2), 591

[10] Ding, F., Liu, Y., Shen, B., & Si, X. M. (2010). An evolutionary game theory model of binary opinion formation. *Physica A: Statistical Mechanics and its Applications*, *389*(8), 1745-1752.

[11] Dong, Y., Zhan, M., Kou, G., Ding, Z., & Liang, H. (2018). A survey on the fusion process in opinion dynamics. *Information Fusion*, *43*, 57-65

[12] Watson, J. B. (1913). Psychology as the behaviourist views it. *Psychological review*, *20*(2), 158

[13] Janis, I. L., & Hovland, C. I. (1959). An overview of persuasibility research. *Personality and persuasibility*, 1-26.

[14] Khanam, K. Z., Srivastava, G., & Mago, V. (2020). The homophily principle in social network analysis. *arXiv preprint arXiv:2008.10383*.

[15] Xiong, F., & Liu, Y. (2014). Opinion formation on social media: an empirical approach. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *24*(1), 013130.

[16] Westerhoff, F., & Hohnisch, M. (2007). A note on interactions-driven business cycles. *Journal of Economic Interaction and Coordination*, *2*(1), 85-91.

[17] Skoric, M., Poor, N., Achananuparp, P., Lim, E. P., & Jiang, J. (2012, January). Tweets and votes: A study of the 2011 Singapore general election. In *2012 45th Hawaii international conference on system sciences* (pp. 2583-2591). IEEE.

[18] Shawn W. Rosenberg, in International Encyclopaedia of the Social & Behavioural Sciences (Second Edition), 2015

[19] Edwards III, G. C., Mitchell, W., & Welch, R. (1995). Explaining presidential approval: The significance of issue salience. *American Journal of Political Science*, 108-134.

[20] Bondy, J. A., & Murty, U. S. R. (1976). *Graph theory with applications* (Vol. 290). London: Macmillan.

[21] Gross, J. L., Yellen, J., & Anderson, M. (2018). *Graph theory and its applications*. Chapman and Hall/CRC.

[22] Murugesan, N., & Nair, D. S. (2011). The domination and independence of some cubic bipartite graphs. *Int. J. Contemp. Math. Sciences*, *6*(13), 611-618.

[23] https://math.stackexchange.com/questions/154941/how-to-calculate-the-number-of-possible-connected-simple-graphs-with-n-labelle

[24] Borgatti, S. P. (2005). Centrality and network flow. *Social networks*, *27*(1), 55-71.

[25] Bonacich, P. (1987). Power and centrality: A family of measures. *American journal of sociology*, *92*(5), 1170-1182.

[26] Brandes, U. (2001). A faster algorithm for betweenness centrality. *Journal of mathematical sociology*, *25*(2), 163-177.

[27] Newman, M. E. (2008). The mathematics of networks. *The new palgrave encyclopaedia of economics*, *2*(2008), 1-12.

[28] Negre, C. F., Morzan, U. N., Hendrickson, H. P., Pal, R., Lisi, G. P., Loria, J. P., ... & Batista, V. S. (2018). Eigenvector centrality for characterization of protein allosteric pathways. *Proceedings of the National Academy of Sciences*, *115*(52), E12201-E12208.

[29] Du, Y., Gao, C., Chen, X., Hu, Y., Sadiq, R., & Deng, Y. (2015). A new closeness centrality measure via effective distance in complex networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *25*(3), 033112.

[30] https://www.sciencedirect.com/topics/computer-science/degree-centrality

[31] de Andrade, R. L., Rêgo, L. C., da Silva, T. L. C., de Macêdo, J. A. F., & Silva, W. C. (2021). Energy disruptive centrality with an application to criminal network. *Communications in Nonlinear Science and Numerical Simulation*, *99*, 105834.

[32] https://en.wikipedia.org/wiki/Centrality

[33] https://homepage.cs.uri.edu/faculty/hamel/courses/2012/fall2012/csc447/lecture-notes/csc447-ln022.pdf

[34] Vasudev, C. (2006). *Graph theory with applications*. New Age International.

[35] https://www.espn.co.uk/football/portugal-por/story/4410619/cristiano-ronaldo-snub-sees-coca-cola-share-price-fall-by-$4bn

[36] https://en.wikipedia.org/wiki/For_loop

[37] https://en.wikipedia.org/wiki/While_loop

[38] https://en.wikipedia.org/wiki/Conditional_(computer_programming)

[39] https://github.com/Sanskar-16/capstone

[40] https://github.com/Sanskar-16/capstone/blob/master/code/graph.py

[41] https://github.com/Sanskar-16/capstone/blob/master/code/analysis.ipynb

[42] https://github.com/Sanskar-16/capstone/blob/master/code/finer-analysis.ipynb

[43] https://github.com/Sanskar-16/capstone/blob/master/code/plotting-graphs.ipynb

[44] https://github.com/Sanskar-16/capstone/blob/master/code/visualisation.py

[45] https://en.wikipedia.org/wiki/Seven_Bridges_of_K%C3%B6nigsberg

[46] https://en.wikipedia.org/wiki/Circle_graph